

Reduced latent belief spaces for active perception in robotics

by

Jennifer Wakulicz

A thesis submitted in partial fulfilment of the
requirements for the degree of Doctor of Philosophy

at the

School of Mechanical and Mechatronic Engineering
Faculty of Engineering and Information Technology
University of Technology, Sydney

April 2024

Certificate of Original Authorship

I, Jennifer Wakulicz, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy in the Faculty of Engineering and Information Technology at the University of Technology Sydney. This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signed:

Date:

Reduced latent belief spaces for active perception in robotics

by

Jennifer Wakulicz

Abstract

As robot perception advances, so does robot autonomy. The development of human-like perception capabilities in robots, where robots actively select sensing actions to improve understanding, is the concern of *active perception* research. A common approach in active perception is to develop sensing policies that minimise the uncertainty in a robot's *belief* about the state of the environment or an object, where a belief is a probability distribution over the state. Planning in the space of possible beliefs rather than physical space is then a prudent approach to active perception, as it places uncertainty minimisation at the heart of planning strategies. However, due to the size of belief space, it is generally infeasible to find optimal belief space plans in the real-time requirements of robotics. As such, research in recent decades has focused on finding high quality yet suboptimal solutions to belief space planning in feasible time frames.

Motivated by this research, this thesis proposes a new approach to belief space planning for active perception coined *reduced latent belief space planning*. Here, a partially observable latent variable is introduced that satisfies two properties: first, that the state of the system under estimation can be inferred from it, and second, that its belief space is of reduced complexity compared to the original belief space. Then, plans designed to improve estimation of the latent variable can be found in a simpler belief space, in turn providing efficient estimation of the original state of interest.

Under this new framework we present a suite of reduced latent belief space planning algorithms that address various active perception problems. For each problem setting we propose a latent variable whose selection is grounded in theoretical understanding of the problem structure. The choice of latent variables is careful, and aims not only to reduce the computational complexity of planning but also to maintain or even improve active perception of the original state.

For general active estimation problems where the system under estimation follows linear Gaussian dynamics with some unknown inputs, we introduce the *state belief space* as a reduced latent belief space for solving joint active estimation of state and unknown input. We extend the Reduced Value Iteration algorithm to this new belief space and derive suboptimality bounds for recovering full system estimation given planning is performed in the state belief space only. We evaluate our theoretical findings in a simulated example of active target tracking, where the target is performing unknown, evasive manoeuvres. Our evaluation demonstrates that impressively, non-myopic planning is achievable via our approach, even in the presence of unknown dynamics.

To further explore the active target tracking problem the *homotopic belief space* is introduced. Homotopy classes are proposed as a latent variable that captures the target’s high-level motion. Estimates of the homotopy class can then be used to infer the target’s full, low-level trajectory. We perform this inference by developing a hierarchical Gaussian mixture model. Then, planning is performed over the homotopic belief space. As a sparse and discrete space, computation of homotopic information gain is more efficient than full belief space alternatives. In empirical evaluation on real pedestrian data, our approach achieves equivalent estimation accuracy as full belief space planning in only half the number of measurements on average.

We continue to demonstrate the flexibility of the reduced latent belief space framework by addressing active mapping using sparse Gaussian processes, introducing the *inducing point belief space*. This reduced latent belief space admits computationally efficient planning heuristics and results in sensing trajectories that outperform conventional approaches. Moreover, theoretical analysis shows that planning in the proposed belief space is a direct conduit for planning to minimise estimation error – an exciting result that connects information-theoretic planning to deterministic error minimisation.

The theoretical and empirical evaluations of each proposed belief space demonstrate that the reduced latent belief space planning framework is a promising theoretical advancement in planning for active perception in robotics. We hope that this approach is adopted by others in the future to work toward fast and accurate active perception.

Acknowledgements

First of all I'd like to thank my supervisors, Prof Robert Fitch and A/Prof Teresa Vidal Calleja. Rob, your guidance and leadership has been invaluable. Thank you for giving me the freedom to explore and for always championing my work, even when I'm having doubts. Teresa, it's impossible to leave your office without a smile. Thank you for being such a joy to learn from. A/Prof He Kong and Prof Salah Sukkarieh, thank you for starting me on this journey and for inspiring me to continue. He, if you hadn't taken me under your wing, I would have never recognised my strengths as a researcher in this field.

Thanks to friends old and new who have kept me laughing during even the hardest times. To Steph, Jess, Lauren, Rory, Chris, Terry, Taiga, Nick, Paddy, Wei, Felix, Vera, Billy and Sam. And of course to Haribo, for getting me across this finish line in one piece. If for some unimaginable reason any of you ever open this document, please feel free to spiral about the order in which your names are written.

Finally, my family. Tim, Ant, Peta, Jenny, Arden and Louie, thank you for being my biggest cheerleaders. I'm so grateful for all your support, humour and kindness. Dad, thanks for letting me live in the garage. Mum, thank you for still looking after me, even when I should be looking after you.

Contents

Declaration of Authorship	iii
Abstract	v
Acknowledgements	vii
List of figures	xiii
List of acronyms	xv
General notation	xvii
1 Introduction	1
1.1 Active perception for stronger autonomy	3
1.2 Uncertainty reduction via belief space planning	5
1.3 Scope	7
1.3.1 Problem formulation	8
1.3.2 Reduced latent belief spaces for active perception	10
1.3.3 Active perception problem settings	11
1.4 Outline and Contributions	14
1.5 Publications	16
2 Related Work	17
2.1 Active perception	17
2.1.1 Exploration-based methods	17
2.1.2 Information-based methods	19
2.2 Belief space planning algorithms	20
2.2.1 Latent belief space planning	22
2.3 Applications	23
2.3.1 Active estimation of states with unknown inputs	23
2.3.2 Active target tracking	24
2.3.3 Active mapping	25
3 Background	27

3.1	Information theory	27
3.2	Bayesian filtering	30
3.3	Tree search planning algorithms	34
3.3.1	Reduced Value Iteration	35
3.3.2	Monte Carlo tree search	39
4	The state belief space for active estimation of unknown inputs	41
4.1	Preliminaries and problem formulation	42
4.1.1	Preliminaries of filtering under unknown inputs	42
4.1.2	Problem formulation	45
4.1.3	Planning framework	46
4.2	Suboptimality bounds for active state estimation	47
4.3	Suboptimality bounds for active unknown input estimation	49
4.4	Experiment results	51
4.5	Summary	54
5	The homotopic belief space for active target tracking	57
5.1	Homotopy theory in robotics	58
5.1.1	The h -signature as a homotopy invariant	59
5.2	Homotopy-informed trajectory prediction	60
5.2.1	Problem formulation	60
5.2.2	Partial h -signatures	62
5.2.3	High-level prediction with variable order Markov processes	62
5.2.4	Low-level prediction with hierarchical Gaussian mixture models	67
5.2.5	Evaluation of predictive model	69
5.3	Homotopic belief space planning	72
5.3.1	Belief updates in the online setting	73
5.3.2	Homotopic information gain	74
5.3.3	Planning framework	76
5.4	Evaluation of homotopic belief space planning	77
5.4.1	Experiment setup	77
5.4.2	Results	79
5.5	Summary	81
6	The inducing point belief space for active mapping	83
6.1	Preliminaries and problem formulation	84
6.1.1	Sparse Gaussian process regression	85
6.1.2	Problem formulation	87
6.2	Information gain and worst-case error minimisation	88
6.3	Planning framework	92
6.3.1	Recursive sparse GP regression	92
6.3.2	Receding horizon planning	93
6.4	Experiment results	94
6.4.1	Characterisation of the error bound	94
6.4.2	Flow field case study	95

6.5	Summary	98
7	Conclusions and future work	101
7.1	Thesis summary	102
7.1.1	The state belief space for active estimation of unknown inputs . . .	102
7.1.2	The homotopic belief space for active target tracking	103
7.1.3	The inducing point belief space for active mapping	103
7.2	Future work	104
7.2.1	Active multi-target tracking via the homotopic belief space	105
7.2.2	Topological latent variables for localisation	105
7.2.3	Improving efficiency of active perception solutions	107
7.3	Outlook	108
	Appendices	108
A	Proofs of results in Chapter 4	109
A.1	Proof of Lemma 4.1	109
A.2	Proof of Lemma 4.2	110
A.3	Proof of Lemma 4.3	113
A.4	Proof of Theorem 4.1	114
A.5	Proof of Lemma 4.4	114
A.6	Proof of Lemma 4.5	115
A.7	Proof of Lemma 4.6	116
A.8	Proof of Theorem 4.2	117
B	Proofs of results in Chapter 6	121
B.1	Proof of Theorem 6.1	121
B.2	Proof of Theorem 6.2	122
	Bibliography	125

List of Figures

1.1	Illustration of informative sensing actions in a simple view planning scenario.	4
1.2	Examples of autonomous robotic systems that rely on active perception to make decisions.	5
1.3	Graphical models for the POMDP problem and the latent belief space planning problem proposed in this thesis.	11
4.1	Results for the active target tracking problem solved via latent belief space planning.	53
5.1	Illustrations of homotopy classes and h -signature calculation.	59
5.2	Equivalent methods of representing a VOMP.	66
5.3	VOMP h -signature prediction for a test trajectory over time.	67
5.4	The ATC shopping mall environment and a subset of the training data.	69
5.5	Results and improvement in median of our proposed topological GMM over a naive GMM in each metric.	70
5.6	Snapshots of predictions output by a GMM with h -signature context given by the VOMP and without.	71
5.7	Heatcubes showing homotopic information gain at sensing locations and times.	76
5.8	Results for active target tracking of pedestrians in the ATC shopping mall dataset.	80
5.9	Results for the most typical runs for each planner.	81
6.1	Graphical models for GP regression.	86
6.2	A study of estimation error for one-dimensional sparse Gaussian process regression of a target function in the RKHS.	95
6.3	The ground truth active mapping environment.	96
6.4	Average absolute reconstruction error with varying search horizon.	97
6.5	Example trajectories produced by the reduced belief space planner.	97
6.6	An example trajectory produced by the full belief space planner.	98

List of acronyms

SLAM	Simultaneous localisation and mapping
POMDP	Partially observable Markov process
GP	Gaussian process
GMM	Gaussian mixture model
PBVI	Point-based Value Iteration
MOMDP	Mixed observability Markov decision process
PDF	Probability density function
PMF	Probability mass function
KL divergence	Kullback-Leibler divergence
FVI	Forward Value Iteration
RVI	Reduced Value Iteration
MCTS	Monte Carlo tree search
UCT	Upper confidence bound for trees
VOMP	Variable order Markov process
RKHS	Reproducing kernel Hilbert space

General notation

x	A scalar value
\mathbf{x}	A column vector
I_n	The $n \times n$ identity matrix
\mathbb{R}^n	The space of real n -dimensional vectors
$\mathbb{R}^{n \times m}$	The space of real $n \times m$ matrices
\mathcal{P}^+	The space of real, positive semi-definite matrices, i.e. matrices $X \in \mathbb{R}^{n \times n}$ such that $\forall \mathbf{v} \in \mathbb{R}^n, \mathbf{v}^\top X \mathbf{v} \geq 0$
\succeq	The partial ordering over positive semi-definite matrices, i.e. for $X_1, X_2 \in \mathcal{P}^+, X_1 \succeq X_2$ if and only if $X_1 - X_2 \succeq 0$
$(\cdot)^\top$	The transpose of a vector or matrix
$(\cdot)_t$	The value of a quantity at time step t
$(\cdot)_{1:t}$	The set quantities from time step 1 up to time step t inclusive
$(\cdot)_{t t-1}$	A prior for a quantity at time t given information up to time $t - 1$
$(\cdot)_{t+1 t+1}$	A posterior quantity at time $t + 1$
$(\cdot)^*$	An optimal quantity
$P(a)$	The probability density of a
$P(a b)$	The conditional probability density of a given b
$P(a, b)$	The joint probability density of a and b
$\mathcal{N}(\boldsymbol{\mu}, \Sigma)$	Multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance Σ
$\mathcal{GP}(\boldsymbol{\mu}(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$	Gaussian process with mean function $\boldsymbol{\mu}(\mathbf{x})$ and kernel function $k(\mathbf{x}, \mathbf{x}')$
\sim	follows, e.g. if \mathbf{x} follows a normal distribution, $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$

Chapter 1

Introduction

Robotics researchers have been striving for decades to enable robots to exhibit the same on-the-fly decision making skills that humans do. A crucial component for achieving this higher level of autonomy is the ability to interrogate the environment for information that improves decision making. Endowing robots with human-like intuition around *how* exactly to interrogate the environment with their on-board sensors is therefore an open research problem, referred to as *active perception*.

Active perception research is concerned with the development of theory and algorithms for planning sensing actions that provide high quality information about systems of interest such as other moving entities or environmental phenomena. With this information, the robot can produce accurate state estimates for these systems and make informed, safe decisions to complete its task. However, due to noise, the true state cannot simply be observed via real-world sensors. The robot must instead estimate or infer the state from noisy measurements. As such, in active perception the optimality of sensing actions is typically characterised by their potential to reduce estimation uncertainty incurred by noise.

Estimates can be represented as probability distributions over the state space of interest, referred to as *beliefs*. The probabilistic nature of beliefs allows for the representation of both the estimated state and the uncertainty in this estimate. Rather than planning sensing actions in physical space, one can plan in *belief space* – the space of possible

beliefs. This approach, referred to as *belief space planning*, facilitates planning to ‘visit’ beliefs with minimal uncertainty directly, naturally lending itself to active perception.

The full potential of uncertainty reduction via belief space planning is unfortunately hamstrung by the immense size and stochasticity of belief spaces. Belief spaces are of much larger dimension than their corresponding state space, and measurement noise introduces stochasticity to transitions between beliefs. Even in active perception of features with small state spaces, finding optimal sensing actions in belief space can require more time than a robot can afford in real-world applications. To overcome this hurdle, roboticists commonly trade optimality for computational efficiency, developing belief space planning algorithms that discretise belief space and determinise belief transitions.

In this thesis we propose the *reduced latent belief space planning* framework, an orthogonal approach to achieving tractable belief space planning for active perception. We introduce partially observable latent variables from which the original state under estimation can be inferred. Moreover, these latent variables have belief spaces of reduced size and complexity compared to the belief space of the original state. By planning sensing actions over the reduced latent belief space, we gain improved computational efficiency while maintaining accurate estimation of the desired state.

We present three instances of latent variables for reduced belief space planning in three active perception scenarios. The choice of latent variables is grounded in careful theoretical understanding of each problem setting. Not only do they provide computational efficiency, but the resulting sensing plans are often more richly informative regarding the original state than plans produced over the full belief space. We provide evidence of this throughout the thesis via both empirical study and theoretical performance guarantees, emphasising the exciting potential of this new framework.

In the remainder of this chapter we establish the scope and contributions of this thesis towards developing the proposed framework. First we further motivate the need for our novel approach. Then, the active perception problem and reduced latent belief space planning approach is formalised. Next, the specific active perception problem applications addressed in this thesis are introduced. Finally, the main contributions of this thesis are summarised.

1.1 Active perception for stronger autonomy

In the late 80s, Bajcsy recognised a significant gap in the field of machine perception and disseminated a new perspective on the issue in her paper entitled ‘Active perception’ [1]. This perspective drew inspiration from human cognition, acknowledging that humans do not simply *see* but rather we *look*. Our ability to actively adjust the way in which we look at our surroundings is an adaptive behaviour central to our success and survival in uncertain or adversarial environments. Why then should the robotic systems we design continue to understand their world by analysing data which passively falls upon their sensors? By choosing where, when and how to look, robots and humans alike can extract advantageous information from their environment for completing their tasks.

Since the introduction of this paradigm, the field of active perception has grown exponentially [2, 3]. Research aims to imbue robotic systems with the ability to actively sense their surroundings to gain information that improves understanding. The quality of a sensing action can be considered the *amount* of information it provides to our robot about the state under estimation. Intuitively, if a measurement greatly reduces the robot’s uncertainty in its estimate, the information gained is high. By quantifying information gain mathematically via information-theoretic measures of uncertainty, our robot can autonomously select good sensing actions in any general setting.

To illustrate, consider a view planning scenario depicted in Figure 1.1. Suppose our robot is given the simple perception task of determining whether an object is a square or triangle pyramid using its on-board camera. The front and side views of a square and triangle pyramid, shown in Figure 1.1(a), are identical to one another. From this viewpoint the category of the pyramid cannot be determined, and it is therefore minimally informative for the classification task at hand. However, the top views alone distinguish the two objects completely, as depicted in Figures 1.1(b) and 1.1(c). Thus, the overhead viewpoint provides optimal information gain and the robot should select this viewpoint for sensing.

A key characteristic of information gain is it is a monotone submodular function [4]. This means the more unique measurements one takes, the more information one has regarding the state of interest, and revisiting sensing locations has diminishing returns. Given infinite

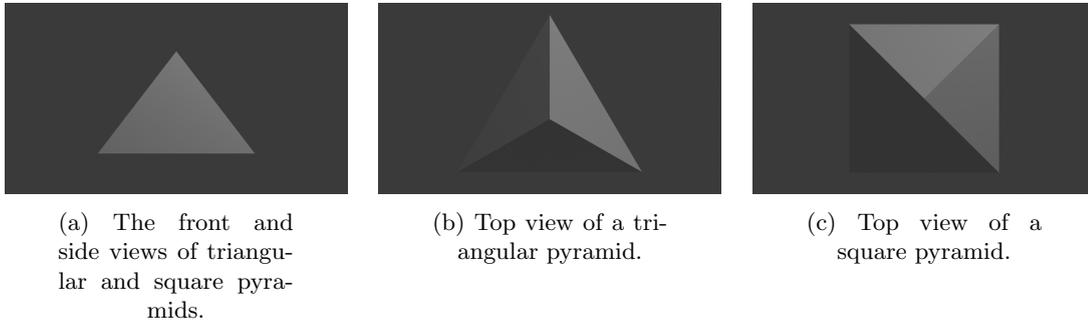


FIGURE 1.1: Illustration of informative sensing actions in a simple view planning scenario.

resources the information-driven active perception task may then seem trivial, as one could plan to take as many unique measurements as possible. However, our robots are typically subject to the harsh realities of the real world: their operation is constrained by computation, time, power, or communication limitations [5–7]. Sensing trajectories must therefore be planned with intention. We wish to select measurement locations carefully to maximise information within budget constraints.

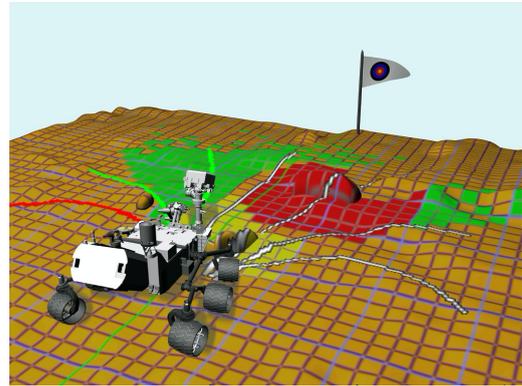
Successful active perception unlocks a level of autonomy in robots that allows us to surpass the goals of yesteryear to automate ‘dull, dirty and dangerous’ tasks with robots. Now, with increasingly efficient active perception we work toward a more ambitious goal, where robots are working and making informed, safe decisions *with autonomy* in unstructured, dynamic environments and even alongside humans. To further motivate this sentiment, consider the scenarios depicted in Figure 1.2.

The development of robotic systems designed to work collaboratively with humans, or *cobots*, is a growing research field [8]. In collaborative settings as in Figure 1.2(a), active perception is crucial for the robot to quickly and accurately estimate the human’s impact on the workspace and make decisions about how to best assist in task completion. Moreover, to ensure safety, the robot cannot afford to take reactive actions after observing its counterpart’s behaviour but should instead anticipate behaviour via its active perception module and make pre-emptive decisions [9, 10].

Autonomous robotic systems have proven useful in planetary exploration (Figure 1.2(b)), where interplanetary communication limitations prohibit tele-operated missions and restrict the amount of data that can be sent back to Earth [5]. Here, active perception



(a) Safe human-robot collaboration requires robots to position themselves to best perceive their changing environments [12].



(b) Extraterrestrial science missions require robots to recognise valuable sensing locations autonomously [13].

FIGURE 1.2: Examples of autonomous robotic systems that rely on active perception to make decisions.

enables the exploratory robot to autonomously seek and collect only the most interesting scientific data to relay to Earth. Additionally, safe navigation while executing this task requires the robot to select landscape viewpoints that aid in building a map of the environment and localising itself – a subproblem of active perception known as active simultaneous localisation and mapping (SLAM) [11].

Despite extensive research and successful applications of active perception, it is far from a solved problem. In the following section we will describe the main challenge in active perception and how it manifests in planning informative sensing actions.

1.2 Uncertainty reduction via belief space planning

A fundamental challenge in active perception will always lie in the physical limitations of real sensors: they provide only partial and noisy information about the observed state. We are then left to infer a probabilistic estimate of the state, or a belief, from these measurements. Beliefs are probability distributions over the true state, providing not just the most likely state but also the uncertainty in this estimate. This is a valuable construct in active perception, as information gain is directly linked to uncertainty. Then,

an intuitive way to plan sensing actions that optimally minimise uncertainty is to plan directly in the space of possible beliefs rather than physical space.

This approach, known as belief space planning, is no simple task. Even a small, discrete physical state space can induce a continuous belief space of much larger dimension. For example, a two-dimensional physical state space of positions in a 10×10 grid has a 100-dimensional belief space. More generally, if there are n possible physical states, the corresponding belief space is n -dimensional.

Furthermore, noise renders measurement outcomes at future sensing locations impossible to predict. Thus, transitions between belief states are stochastic in nature. To find high-quality sensing actions in belief space, planning algorithms have been developed that estimate the expected information gain at a sensing location by sampling the space of possible measurements [14]. Sampling policies dictate the suboptimality of the resulting solutions. In other work, algorithms are proposed to determinise belief state transitions by, for example, considering only the maximum likelihood measurement outcome, possibly incurring great suboptimality in the resulting sensing plans [15, 16].

Another hurdle is in evaluating the information-theoretic metrics that quantify the expected quality of sensing actions. This can be computationally expensive, particularly over long planning horizons. If beliefs are Gaussian in nature, these metrics often have tractable closed-form expressions. As such, confining problem formulations to those that assume Gaussian beliefs is a popular approach to active perception. However, even here, the information-theoretic metrics can become tricky to evaluate. For example, problems with high-dimensional multivariate Gaussian beliefs [17] can have computationally expensive information measures, and problems where the belief is a mixture of Gaussian distributions can have no known closed-form expressions at all [18].

The primary contribution of this thesis is the introduction of a new belief space planning framework that addresses these challenges. Specifically, we propose a framework that facilitates planning over belief spaces of reduced dimensionality – the belief spaces associated with a *latent variable* or feature. To ensure the original active perception problem is solved and an accurate estimate of the desired state is acquired, the full state must be able to be inferred through the latent variable. Then, we can plan sensing actions that optimise

information gain regarding the latent variable and recover an accurate state estimate by proxy.

The secondary contribution of this thesis is therefore the identification and evaluation of specific latent variables that satisfy the aforementioned criteria for reduced belief space planning. For a number of active perception problems, an interrogation of the problem structure uncovers examples of these latent variables.

While existing approaches to belief space planning introduce planning algorithms that temper the curse of dimensionality via discretisation or determinisation [14], our reduced latent belief space planning approach is an entirely orthogonal approach. Though the belief spaces are of reduced size and complexity, existing planning algorithms can still be used to explore them for sensing plans. This thesis shows these existing planning algorithms for active perception tasks are improved by utilising our reduced belief spaces.

The improvements are twofold. First, the information gain measures associated with the latent variables are less computationally taxing than those associated with the original state. Second, plans formulated over the reduced belief spaces often result in measurements that are more richly informative regarding the true state than those planned over the full belief space. These improvements are observed throughout the thesis both via experimental evaluation of the proposed latent variable approach and via theoretical performance guarantees.

1.3 Scope

The work in this thesis aims to improve the efficiency of active perception pipelines. Specifically, this work focuses on finding manageable belief spaces over latent variables that are afforded to us by structures present in specific problem settings. These belief spaces are of lower dimension or complexity than the full belief spaces typically considered. As a result, they admit planning heuristics that are easier to compute. This thesis aims to demonstrate how utilising these reduced latent belief spaces can quantifiably improve both the planning and estimation stages of the active perception pipeline via theoretical and experimental results.

1.3.1 Problem formulation

The problem central to this thesis is the active perception problem. Put simply, this is the task of planning future sensing trajectories for a mobile sensing robot that best improve estimation of the state of a system of interest.

More formally, suppose one has a mobile sensing robot whose state $\mathbf{x}_t \in \mathbb{R}^{n_x}$ at time t evolves according to the following dynamic model given control input $\mathbf{u}_t \in \mathcal{U}$ at time t selected from the space of admissible controls \mathcal{U} ,

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t). \quad (1.1)$$

Throughout this thesis, the robot dynamics f are assumed to be deterministic, that is, the localisation problem is not considered.

The robot must measure and build an estimate of a possibly time-varying target system whose state $\mathbf{y}_t \in \mathbb{R}^{n_y}$ evolution is also governed by a discrete dynamics model

$$\mathbf{y}_{t+1} = g(\mathbf{y}_t, \mathbf{w}_t), \quad (1.2)$$

where \mathbf{w}_t is the outcome of a random process at time t representing process noise. The robot's measurement $\mathbf{z}_t \in \mathbb{R}^{n_z}$ at time t follows the sensor observation model

$$\mathbf{z}_t = h(\mathbf{x}_t, \mathbf{y}_t, \mathbf{v}_t), \quad (1.3)$$

where \mathbf{v}_t is the outcome of a random process at time t representing measurement noise.

The measurements taken by the robot are intended to improve the accuracy of an estimate of the state of interest. The suitability of a measurement for achieving this task can be quantified by the *uncertainty reduction* – or equivalently the *information gain* – it provides regarding the state. One of the most common approaches used to quantify information gain is via concepts from information theory [11], within which there are various metrics.

With the tools introduced above we may now formulate the active perception problem addressed in this thesis as a single optimal control problem.

Problem 1.1. Given a planning horizon $T < \infty$ and an initial belief over \mathbf{y}_0 , choose a sequence of controls $\sigma = \{\mathbf{u}_0, \dots, \mathbf{u}_{T-1}\}$ for the sensing robot that results in the best estimate of the state \mathbf{y}_T , or

$$\max_{\sigma \in \mathcal{U}^T} \mathcal{IG}(\mathbf{y}_T; \mathbf{z}_{1:T}) \quad (1.4)$$

$$\begin{aligned} \text{s.t. } \quad & \mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t), \quad t = 0, \dots, T-1, \\ & \mathbf{y}_{t+1} = g(\mathbf{y}_t, \mathbf{w}_t), \quad t = 0, \dots, T-1, \\ & \mathbf{z}_t = h(\mathbf{x}_t, \mathbf{y}_t, \mathbf{v}_t), \quad t = 1, \dots, T, \end{aligned}$$

where $\mathcal{IG}(\cdot)$ is a measure of information gain and $\mathbf{z}_{1:T} = \{\mathbf{z}_1, \dots, \mathbf{z}_T\}$.

In the above formulation we pose the active perception problem as an open loop feedback control problem rather than a closed loop control problem. The distinction between these two approaches is an ‘action versus strategy’ one. In open loop feedback control a sequence of actions is chosen at planning time and is executed until new information is gained at which point replanning is performed. Meanwhile in closed loop control, optimal control *policies* are found, i.e. functions of \mathbf{y}_t that return an optimal control [19]. Closed loop control is typically superior to open loop, however, for the problems considered in this thesis, where observation models are linear or linearisable and noise is Gaussian distributed, open loop feedback control is provably sufficient [20].

The tools introduced in this problem formulation are written in their most general form and are common to all active perception formulations, though specific instances vary. For example, the sensing robot’s dynamics f may be described by a simple unicycle model as in Chapter 4 or may even include external dynamics such as flow field vectors, as in Chapter 6. Physically, the state under estimation \mathbf{y}_t may represent the pose of another entity such as a human, vehicle, animal or robot, or the state of a non-entity such as spatial maps environmental phenomena as in Chapter 6. The state \mathbf{y}_t may be even more abstract, such as the intent of another agent in the environment or the category of an object. There are many observation models that correspond to physical sensors [21]. Likewise, there are many metrics that are used to quantify information gain, most stemming from the concept of entropy, as detailed in Chapter 3.

1.3.2 Reduced latent belief spaces for active perception

Solving Problem 1.1 is confounded by the noise terms present in Equation (1.3). In particular, measurement noise renders the state under estimation only *partially observable*. Problem 1.1 is thus an instance of a partially observable Markov decision process (POMDP) problem. The true state \mathbf{y}_t may only be inferred from measurement outcomes \mathbf{z}_t at sensing locations \mathbf{x}_t via the calculation of a belief. This inferential relationship is depicted in Figure 1.3(a).

To solve POMDP problems planning is performed in the space $\mathcal{B}_{\mathbf{y}}$ of all possible beliefs held over the state space of \mathbf{y}_t . However, solving POMDPs exactly can be intractable for systems with just tens of states. While approximate solvers can produce good solutions for belief spaces of significant size [14], the approach in this thesis is to further remove computational burden by introducing partially observable latent variables $\boldsymbol{\xi}_t$ through which state inference is performed. That is, the desired state can be inferred given knowledge or observation of this latent variable, and the latent variable can be inferred from measurements. Further, the dynamics of $\boldsymbol{\xi}_t$ are either learnable through partially observed historical data, or are accessible via known analytic models. Thus, we introduce an added layer of inference, depicted in Figure 1.3(b). Then, rather than solving Problem 1.1, we propose a framework that instead solves the following:

Problem 1.2. *Given a planning horizon $T < \infty$, choose a sequence of controls $\sigma = \{\mathbf{u}_0, \dots, \mathbf{u}_{T-1}\}$ for the sensing robot maximises information gained about latent variable $\boldsymbol{\xi}$, or*

$$\max_{\sigma \in \mathcal{U}^T} \mathcal{IG}(\boldsymbol{\xi}_T; \mathbf{z}_{1:T}), \quad (1.5)$$

as a proxy for solving Problem 1.1.

The benefit of introducing this added layer of inference now becomes clear. The latent variables $\boldsymbol{\xi}_t$ are still partially observable. However, now belief space planning can be executed over the belief space associated with $\boldsymbol{\xi}_t$. Thus, $\boldsymbol{\xi}_t$ is chosen such that the corresponding belief space $\mathcal{B}_{\boldsymbol{\xi}}$ is of reduced dimensionality or is sparse compared to $\mathcal{B}_{\mathbf{y}}$. Additionally, information-gain measures are then easier to compute. While it may seem difficult to find a $\boldsymbol{\xi}_t$ that fits the required criteria for reduced latent belief space planning, often problem

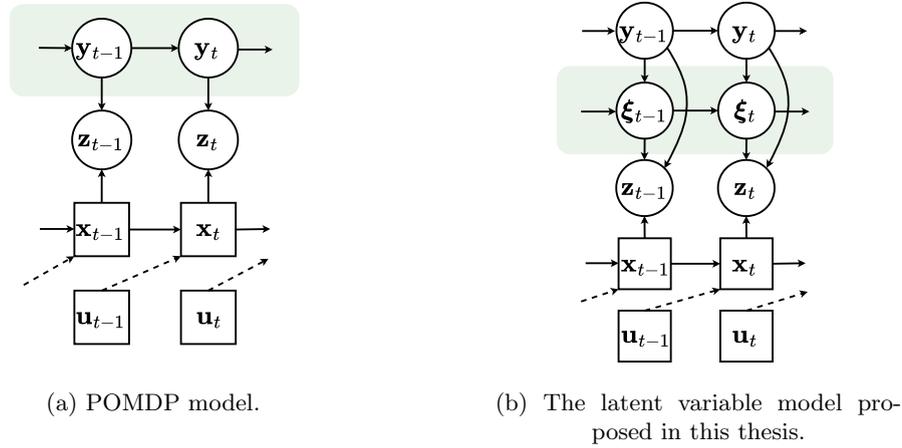


FIGURE 1.3: Graphical models for the POMDP problem and the latent belief space planning problem. Variables that are partially observable are depicted as circles. Variables that are considered fully observable in this thesis are depicted as squares. Arrows from partially observable variables A to B indicate that A is made available via inference given B . Shaded levels indicate belief space used for planning in each approach.

structure guides us to the appropriate latent variable. To build further intuition around ξ_t , specific problems considered in this thesis are outlined in the following section, along with the dynamics and nature of ξ_t used in each problem.

1.3.3 Active perception problem settings

This thesis addresses Problem 1.1 via the framework in Problem 1.2 in various practical settings differentiated by the state of interest in each application. These settings are outlined here, along with the various constraints and challenges inherent to each.

Active estimation of states with unknown inputs

Sometimes the evolution of a state of interest can be adequately modelled as a linear system with additive Gaussian noise, i.e. $g(\mathbf{y}_t, \mathbf{w}_t) = A_t \mathbf{y}_t + \mathbf{w}_t$, with $\mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, Q_t)$. The Gaussian noise captures small disturbances in the evolution that perturb the model away from an exactly linear one. Under these circumstances, a Kalman filter can be used to estimate the state and to facilitate belief space planning. However, these assumptions may be limiting when the state has more complex dynamics that are not captured by the Gaussian noise term. If these dynamics are prohibitively difficult to model explicitly, they

can be treated as an *unknown input* in the motion model and estimated via an unknown input filter [22].

This necessitates the online estimation of two possibly uncorrelated variables: the target state and the input itself, complicating solving Problem 1.1 as the underlying belief space is expanded considerably with the addition of the input. Further, optimal sensing locations for the state and the unknown input may be at odds with each other. Chapter 4 exploits structure in the design of an established unknown input filter to return to a manageable belief space to plan over. The proposed latent variable ξ_t is a subset of the full state being estimated. Specifically, ξ_t is the state variable without the input. The problem structure gives the dynamics of ξ_t via the unknown input filter and dictates that the initial belief over the latent variable is Gaussian. As such, the reduced latent belief space planning problem looks very similar to the original, however here the challenging joint estimation problem is discarded entirely.

Active target tracking

When the state of interest \mathbf{y}_t is the position of another moving entity, we face the active target tracking problem. In its most general form, this problem is confounded by unknown or partially known dynamic models of the target and the robot's environment. However, this thesis assumes an accurate environmental model is given, focusing instead on overcoming the challenges brought by unknown dynamics by learning a dynamic model.

Learning a dynamic model from historical data can be difficult, as moving entities do not often follow a uni-modal distribution. Multi-modal models can be learned where each mode can be considered one hypothesis regarding the target's true dynamics. When tracking a target modelled in this way, we again encounter a complicated belief space planning problem, where sensing paths must minimise the uncertainty in both hypothesis and position together. Furthermore, information gain measures associated with mixture distributions are often intractable. To address this, Chapter 5 introduces the belief space associated with a target's high-level motion. The proposed latent variable ξ_t is the target's *homotopy class*, an identifier of high-level motion. The dynamic model of ξ_t is learned from historical trajectory data and is closely associated with the dynamic model of the original

state, however is much sparser in nature. The learned dynamic model allows us to explore the reduced latent belief space associated with high-level motions. Here, the expected information gain of a sensing action is evaluated much more efficiently than in the original belief space.

Active mapping

In environmental monitoring or 2.5-dimensional mapping applications, the state of interest \mathbf{y}_t in Equation (1.2) is sometimes a static, scalar spatial field. That is, $\mathbf{y}_t = y$ is some continuous function of position \mathbf{x}_D in the domain, $y = s(\mathbf{x}_D)$. Note that \mathbf{x}_D is separate from the sensing robot position \mathbf{x}_t at time t .

One might naively plan to send the robot on a sensing trajectory that covers the entire domain of the environment – collecting all the information required for mapping this spatial field. However, sensing robots often have constraints that limit the number of measurements they may take. We must then plan maximally informative plans under this constraint. The spatial field is a continuous function, and accurate estimation is required not just near observed locations, but across the whole unseen domain. Once again, predicting and updating a belief of this size is computationally intensive.

Chapter 6 employs a sparse Gaussian process (GP) for reconstructing static, scalar spatial fields in environmental monitoring settings. The sparse GP produces a continuous estimate of y via a set of interpolating *inducing points*, reducing the computational complexities faced when using full GPs. We harness this structure of the sparse GP in the planning stage, proposing to plan over the belief space of inducing points rather than the full representation. Then, the latent variable ξ_t is the spatial field estimate held at the inducing point set. The evolution of this estimate is modeled by a Kalman-like belief update, giving a scalable, online framework to explore the reduced latent belief space.

1.4 Outline and Contributions

The main contribution of this thesis is the introduction of reduced latent variable belief space planning as a framework to simultaneously reduce computational complexity and improve estimation quality in active perception problems. To this end, we contribute a suite of reduced latent variable belief spaces and evaluate them. Below is an outline of each chapter with their contributions.

Chapter 2 presents a review of established literature relevant to this thesis, covering previous work on active perception, belief space planning and the specific problem settings studied in this thesis.

Chapter 3 gives a brief introduction to information theory and Bayesian filtering, tools that are central to the work presenting in following chapters. Established belief space planning algorithms used in this thesis are also introduced.

Chapter 4 addresses active perception of systems with arbitrary, unknown inputs in their dynamic model. To avoid complications associated with jointly estimating both the system state and the unknown input, this chapter proposes planning over the state belief space only. The structure of the unknown input filter used to maintain beliefs allows the state to be treated as the latent variable via which inference of the full system description is performed. The contributions of this chapter are as follows:

- Reduced latent belief space planning where ξ_t is the state variable, a subset of the full system description.
- An extension of the Reduced Value Iteration algorithm to the proposed reduced latent belief space for active estimation of linear Gaussian systems with unknown inputs.
- Derivation of suboptimality bounds for full system estimation acquired through planning over the proposed reduced latent belief space with the RVI extension.
- Evaluation of the proposed approach on an adversarial target tracking example.

Chapter 5 further explores active target tracking, where target dynamics are modelled via a multi-modal distribution learned from historical trajectory data. Homotopy classes are introduced as the latent variable used to facilitate reduced belief space planning. This chapter’s contributions are:

- Reduced latent belief space planning where ξ_t is the homotopy class of the target’s trajectory, a discrete random variable with sparse belief space.
- Development of a topological multi-modal trajectory prediction model based on Gaussian mixture models and homotopy classes to facilitate homotopic belief updates based on low-level measurements.
- Evaluation of the proposed belief space planning methodology on real pedestrian data, demonstrating that planning over the homotopic belief space results in impressive estimates of target state with substantially fewer measurements.

Chapter 6 addresses active mapping using sparse Gaussian processes. Sparse GPs are used as an efficient representation for the spatial field under estimation due to scalability challenges associated with full GPs. We additionally leverage the inducing point-based sparse GP for improved planning. Specifically, the contributions of this chapter are:

- Reduced latent belief space planning where ξ_t is the pseudo-measurements held at inducing point locations.
- A theoretical connection between belief space planning and deterministic worst-case estimation error minimisation for active mapping with sparse GPs via derivation of information-theoretic bounds on worst-case estimation error.
- Demonstration of the proposed methodology on a mapping problem for an underwater robot in a flow field.

Chapter 7 summarises and concludes the thesis, providing discussion regarding implications of this and future work.

1.5 Publications

The work presented in this thesis is largely based on the following publications. In each of these publications, Jennifer Wakulicz is the primary contributing author.

1. **Wakulicz, J.**, Kong, H. and Sukkarieh, S., 2021, May. Active information acquisition under arbitrary unknown disturbances. In *2021 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 8429-8435). IEEE.
2. **Wakulicz, J.**, Lee, K.M.B., Yoo, C., Vidal-Calleja, T. and Fitch, R., 2022, May. Informative planning for worst-case error minimisation in sparse gaussian process regression. In *2022 International Conference on Robotics and Automation (ICRA)* (pp. 11066-11072). IEEE.
3. **Wakulicz, J.**, Lee, K.M.B., Vidal-Calleja, T. and Fitch, R., 2023, May. Topological trajectory prediction with homotopy classes. In *2023 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 6930-6936). IEEE.

Chapter 2

Related Work

This chapter gives a review of existing literature relevant to this thesis. First, we review progress in solving the active perception problem. Then, we outline existing solutions to belief space planning and review work adjacent to our latent variable approach. Finally, we cover literature pertaining to the active perception tasks addressed in this thesis.

2.1 Active perception

The active perception problem statement, to actively select sensing actions that improve understanding of the environment, is a broad statement that can be addressed in many ways. As such, within the research there are various degrees to which perception is ‘active’. In this section we review progress toward increasingly sophisticated sensing policies.

2.1.1 Exploration-based methods

Active perception for mapping, inspection, object search, classification or recognition can be framed as a exploration maximisation problem. This is an intuitive approach – to maximise the understanding of the environment or an object, visit all unseen locations.

Frontier-based exploration was introduced for active mapping in [23], where goal locations for sensing are those on the edge of known and unexplored space, or *frontiers*. Then, the

sensing policy is to navigate directly to the nearest frontier and observe new space. Frontier exploration was a pioneering approach in active mapping, but the myopic, distance-based frontier selection in [23] suffers from local minima and is thus not optimal for maximising coverage in minimal time.

Since its introduction, frontier-based exploration has been expanded to introduce various utility functions for improved frontier selection. For mapping and inspection tasks as in [24–27], frontier selection is based on the *utility* of the viewpoint, where utility is captured by how many unseen grid cells are revealed at the frontier. Additionally, the cost of navigation is considered in the frontier selection, promoting more efficient exploratory behaviour. Evaluation of different frontier-based methods in [28] for mapping illustrates that these cost-utility based methods exhibit more rapid initial coverage of the environment than the original distance-based selection policy in [23].

Utility functions can be designed to suit the the perception task at hand, making frontier-based methods more broadly applicable to perception tasks other than mapping. For example, for active search tasks, utility functions can include the probability of successful communication between collaborating robots [29, 30] or the likelihood of detecting a target object [31]. However, in frontier-based research, there is no unified approach to active perception as a whole, as the utility function design is ad-hoc.

Other exploration-based methods treat exploration as a coverage problem, where by the end of a sensing trajectory, the accumulated sensed area must cover the entire domain. These approaches optimise sensing networks or sensing trajectories over a discretisation of the domain such that theoretical coverage guarantees are met [32–34]. More recently, the coverage problem has been addressed for continuous domains, increasing the complexity of the optimisation problem as the search space over continuous paths becomes infinite. Here, *ergodic* search methods are developed, where continuous trajectories are designed whose time-averaged spatial distribution approximate the spatial distribution of information in the domain [35]. This approach has been shown to be optimal for exploration under certain conditions [36] and is not limited by a discretisation of the environment, and thus is an increasingly popular method [37, 38].

2.1.2 Information-based methods

Information-based methods select sensing actions with planning heuristics grounded in information theory. Information theory provides a single, unified approach to any active perception task: to drive down entropy or uncertainty in estimated variables by maximising *information gain*. This approach has been shown to give more accurate estimation results than frontier-based alternatives [28, 39, 40].

Information gain is a monotone submodular function, meaning measurements always provide positive information about a variable, however as more measurements are taken, less information is gained [4]. This property renders greedy approaches to maximising information gain highly attractive. Not only are they efficient, but they have been proven to give at worst a constant-factor approximation to the optimal maximum information gain [41]. Despite this, non-myopic planning for information maximisation can still outperform greedy policies. This is especially the case for active perception tasks complicated by sensing or dynamic constraints, where the constant-factor guarantee may break. Various non-myopic algorithms with performance guarantees have been developed to this end in [20, 42, 43]

Information-driven active perception can additionally encourage behaviour beyond pure exploration. In some cases, the goal of reducing uncertainty manifests as *exploitative* behaviour as well. For example, in active SLAM map uncertainty is minimised not only by exploration, but also by re-visiting sensing locations, referred to as loop closure. By visiting a known location, mapping uncertainty and errors due to sensor drift can be corrected. In active SLAM works such as [25, 44, 45] the trade-off between coverage and loop closing – or exploration and exploitation – is inherently balanced via information gain maximisation.

Information-based methods are closely tied to belief space planning in that they both explicitly handle uncertainty during planning. In fact, much information-based active perception research adopts belief space planning as a planning method.

2.2 Belief space planning algorithms

Active perception is a task complicated by various sources of uncertainty. In realistic circumstances there is stochasticity in a robot’s motion model, sensing model and environment, and the true optimal control sequence for maximising information gain can only be estimated. This problem can be formulated as an instance of a partially-observable Markov decision process (POMDP) problem [46].

A POMDP consists of environmental or robot states s_t at time t , a set of actions $A(s_t)$ for each state, a transition model $P(s_{t+1} | s_t, a_t)$ describing the probability of moving to state s_{t+1} by taking action $a_t \in A(s_t)$ at state s_t (capturing motion or environmental uncertainty) and a set $\Omega(s_{t+1})$ of possible measurements y_{t+1} of new state s_{t+1} , each with a probability $P(y_{t+1} | s_{t+1}, a_t)$ of occurrence (capturing sensing uncertainty). Due to this sensing uncertainty, the robot must maintain a belief $b(s_t)$ over the state. Then, as the robot takes an action a_t and observes y_{t+1} it can calculate an updated belief $b(s_{t+1})$ via Bayesian inference. Each action has associated reward $R(a_t, b(s_{t+1}))$ which quantifies the quality of the belief that follows from taking it. Since the measurement outcome $y_{t+1} \in \Omega(s_{t+1})$ is not known prior to taking an action, the reward for taking an action must be evaluated by taking the expected value over all possible measurement outcomes. A solution to the POMDP problem is thus a mapping from belief space to action space, or a *policy* for selecting actions that maximises the expected reward.

The POMDP framework and belief space planning is a ubiquitous approach in information-driven active perception, as it allows roboticists to place estimation and uncertainty minimisation at the heart of planning strategies. It has been used with great success as an approach to active perception problems such as active SLAM [47–53], active recognition [54–58], active intention inference for human-robot collaboration [59–61] and environmental monitoring [62–65]. However, this broad usage of belief space planning in active perception is only possible thanks to major developments in POMDP solvers.

Finding optimal solutions to POMDP problems is generally intractable due to the high

dimensionality of the belief space and long planning horizons [14]. In fact, it is a PSPACE-hard problem [66]. The first approaches to demonstrate the viability of POMDP formulations in robotics were sampling based, solving the problem over a sampled set of beliefs rather than the full space. The general approach is to expand a belief tree by sampling action space and observation space. For each action, a set of observations are sampled and belief nodes corresponding to each action-observation pair are added to the tree. Sampling beliefs is therefore equivalent to choosing a node to expand with an action-observation pair. The value or quality of sampled beliefs are estimated via back-propagation of the value of their descendants in a value iteration process.

The Point-based Value Iteration (PBVI) [67] algorithm produced successful solutions to POMDP problems an order of magnitude larger than any solver preceding it by sampling only the belief space reachable by the robot and performing value iteration to estimate rewards associated with actions. Perseus [68] improved upon PBVI by performing back-propagation of value estimates over a subset of the sampled beliefs to reduce computation time further. The breakthrough algorithm SARSOP [69] produced better solutions to the same problems as its predecessors in only seconds rather than hours by sampling only from an approximated region of belief space reachable from the *optimal* policy. Advancements continue to be discovered by improving the way sampling and value iteration is performed [70–73].

Other sampling-based methods apply to problem settings where beliefs are assumed to be Gaussian and dynamics to be linear. In these cases the *separation principle* applies, allowing the problem to be reduced to a deterministic control problem where sampling to calculate expectations over measurement outcomes no longer needs to be considered [74]. The belief spaces here are Gaussian in nature, and beliefs expected from a sensing action can be computed via the Kalman filter. Then a search tree in belief space can be expanded deterministically to find a solution, as in [15, 16, 20, 75, 76]. These assumptions have also been used to extend traditional algorithms such as Rapidly-exploring Random Trees and the Probabilistic Roadmap to explore the belief space for belief space planning [77, 78].

2.2.1 Latent belief space planning

At the core of each aforementioned approach is an attempt to reduce the belief space associated with the target state to a more manageable representation for planning. The work in this thesis presents an alternative route: to focus on a feature related to the state that has a naturally smaller belief space for more efficient planning. These latent features with reduced belief spaces must still be informative for the perception task at hand. Our approach is related, but not equivalent, to the following established work.

In [79] the authors identify that some systems exhibit what they refer to as mixed observability. In systems with mixed observability, some elements of the state are either fully observable or can be sensed with enough accuracy to be considered so. The remaining states are partially observable as in typical POMDP formulations. Then the state can be partitioned into fully observable and partially observable components via a factored model. The belief space is then similarly partitioned into two corresponding disjoint subspaces. Belief space planning can be performed over just the partially observable belief space, which is of much lower dimension than the full belief space. The mixed-observability Markov decision process (MOMDP) framework improved performance of SARSOP on a number of benchmark POMDP problems. This approach is similar to the one presented in this thesis. In fact, the MOMDP model can be illustrated in a similar way to Figure 1.3(b), however, our proposed model has no fully observable component and the desired state must be inferred from the latent variable.

An extension of the MOMDP formulation to include partially observable latent features that are not direct subsets of the target state was presented in [61]. Here, the partially observable variable describes a pedestrian’s intent, while the position and velocity of the entity are assumed to be fully observable. Then, intention-aware motion planning is executed according to the current intent belief by solving the MOMDP problem. Similar motion planning work over latent belief spaces is studied in [80, 81]. While related to this thesis as they introduce planning over belief spaces of latent variables, this work is in motion-planning rather than active perception, and still assumes mixed observability.

Latent belief space planning for active perception is not yet well established in the literature beyond work presented in this thesis. However, the power of estimating a related

latent variable for improving estimation of a target state has been demonstrated numerous times before. For example, in trajectory prediction and estimation, estimation of social interactions between entities has improved accuracy demonstrably [82–84], as has estimation of the entity’s goal or intention [85–87]. In these papers, planning for perception is performed over the *full* belief space associated with the state or via other methods all together. Perception of the latent variable is passive. On the other hand, in this thesis planning is performed over the smaller latent belief space and *active* perception of the latent variable is used to inform estimation of the full state.

2.3 Applications

As described in Chapter 1, active perception is a general problem with many specific applications. In the remainder of this chapter, we review related work surrounding the applications studied in this thesis.

2.3.1 Active estimation of states with unknown inputs

Often in robotics there are systems whose evolution involve some arbitrary, unknown disturbances that are difficult or even impossible to statistically interpret or model. These unknown disturbances are of great importance for understanding the system, and thus must be estimated. For example, understanding the behaviour of faults or attackers on cyber-physical systems is crucial for safeguarding against them [88–90]. When tracking and localising moving objects or entities, we must be able to estimate any abrupt manoeuvres they perform or are subject to [91–95]. Complex interactions in advanced vehicle applications [96–99], and grasping and manipulation [100, 101] are also important to estimate in many robotics applications, though they may not be easy to model.

Active estimation of these unknown inputs is thus an attractive approach to improving our understanding of these systems. However, since they are difficult to statistically interpret or model, belief space planning for improving estimation is near impossible. Thankfully, unknown input filtering or Simultaneous Input and State Estimation (SISE) frameworks [22, 102–105] have been designed that can facilitate this. Specifically, [22]

decouples the unknown input from the system dynamic model, maintaining an estimate and associated error covariance for the two components separately. Then, as in Chapter 4, active estimation via belief space planning is possible. However, it is a *joint* estimation problem, where both the state and the unknown input must be estimated together.

Joint active estimation is not straightforward. Often the sensing actions that maximise information gain regarding one component of the joint state is at odds with optimal sensing actions for the other. A famous instance of this is in active SLAM, where exploratory behaviour to maximise mapping information is at odds with loop closing to maximise localisation information [106]. Balancing these two goals is important for good active perception, but bloats the size of the already large belief space. In Chapter 4 we demonstrate that for joint active estimation of states with unknown inputs, one can focus on perception of a *single* variable and rely on the structure of the unknown input filter to recover good joint estimation.

2.3.2 Active target tracking

Tracking the dynamics and behaviour of other agents is important for autonomous robots to operate harmoniously in environments with moving entities such as pedestrians, vehicles, animals or human collaborators [9, 82, 86, 107–112]. It is also central to target localisation and tracking problems that can benefit from mobile sensing robots such as surveillance and search and rescue [93, 113–115].

Understanding a target’s motion model is necessary for active target tracking, as predictive models of motion facilitate the prediction and update of beliefs required for belief space planning. For targets whose motion models can be sufficiently described by linear dynamics with additive Gaussian noise, a Kalman filter is optimal for this task as long as measurement noise is also Gaussian distributed [20, 75, 92, 116–120]. Extensions are easily made to non-linear target and sensing dynamics via linearisation with an extended Kalman filter in [121–123]. For active target tracking in cases where noise is also non-Gaussian, beliefs can be approximated via sampling using particle filters [113, 124–126].

Sometimes target dynamics are complex enough to warrant learning a model for them. These learned models must again support belief space planning. Previously, recurring neural networks and long-short term memory networks have been implemented for learning motion models with success [127–130]. Other approaches have introduced mixture models such as Gaussian mixture models (GMMs) and the kernel trajectory map [131] to capture the multi-modal nature of an agent’s possible future paths, showing greater prediction accuracy over single-mode approaches [107, 132–134].

Multi-modal or multi-hypothesis models can be overwhelming in belief space planning scenarios. With each measurement and belief update, the number of possible hypotheses grow. For beliefs represented by GMMs, there has been plentiful work in best addressing this combinatorial expansion via, for example, pruning non-dominant or redundant components [135–139]. In Chapter 5, we propose to fix the number of possible motion hypotheses to the number of possible high-level motions or homotopy classes. Then a simplified, component-wise belief update can be performed via Gaussian conditioning as in [132].

2.3.3 Active mapping

The online mapping of scalar spatial fields from measurement data is often required in environmental monitoring applications in oceanography [140, 141], agriculture [142] and even navigation [65, 143]. Here, the state under estimation is a representation of space and can be prohibitively large to estimate. Much work has been directed toward addressing this, designing efficient representations [144–146] and filtering algorithms [147–149]. A particular representation of interest in this thesis is provided by Gaussian processes (GPs).

GP regression [150] is a powerful machine learning technique for modelling spatially correlated phenomena. GPs have been widely used in the robotics community to estimate a variety of spatial fields as in [65, 140, 142, 143, 151, 152]. However, few robotics papers have studied GP regression beyond its use as a tool. A full understanding of its theoretical power in robotics is yet to be achieved. In [153–155] the connection between GP regression and kernel interpolation theory is studied, providing error bounds given *noise-free* observations. However, this is a noise assumption rarely applicable to active perception

applications. In Chapter 6 we extend these results to cases where observation noise is *bounded*, a more realistic assumption in robotics.

Another property of GPs pertinent to robotics applications is that the computational complexity of GP regression scales cubically with the size of the input data. This can be problematic in real-time active perception scenarios. Sparse GP approaches mitigate this computational challenge by adopting simplifying approximations [147, 156–158]. One such approximation is the *inducing point* formulation [147, 156, 159], where the target function is assumed to be conditionally independent given the function values at a fixed set of inducing points. A recent advance in inducing point-based sparse GP regression is that incoming measurements can be ‘fused’ via recursive Bayesian estimation of a latent Gaussian state of fixed dimensionality, reminiscent of Kalman filtering [142, 148, 151, 160–162].

Path planning for optimal reconstruction of a GP is typically posed as an information gain or marginal entropy maximisation problem [17, 65, 163, 164]. However, this involves planning heuristics whose computational cost grows as the size of the measurement set grows. In Chapter 6 we instead propose to treat the sparse GP estimate held at the inducing points as the latent variables via which the full estimate is inferred. Then, planning over this belief space is simplified as the computational cost is fixed throughout the planning horizon.

Chapter 3

Background

This chapter presents the mathematical formulation for several of the tools and concepts foundational to this thesis. First, information theory is introduced along with measures of information gain. Then, Bayesian filtering is introduced in the context of maintaining beliefs based on measurements. Finally, established belief space planning algorithms used throughout this thesis are detailed.

3.1 Information theory

Information theory is the mathematical framework through which the quantification and transmission of information is studied. Thus, information theory provides the tools required for quantifying the expected information gain that a robot's measurements will provide. In Chapter 1 the state y^1 is introduced as a random variable due to noise processes. The information content of a state y – an outcome of the discrete random variable Y – can be intuited as a value which increases as the probability of the state outcome decreases. That is, low probability states have high information content \mathcal{I} ,

$$\mathcal{I}(y) = \log \left(\frac{1}{p(y)} \right), \quad (3.1)$$

¹For ease of notation, we drop time-dependence and boldfacing for $\mathbf{y}_t, \mathbf{z}_t$ introduced in Chapter 1, as concepts in this section are applicable to states of any dimension and can be extended to the time-varying case by considering random processes rather than random variables.

where $p(y)$ is the probability of outcome y given by the probability mass function p .

The information content of the random variable itself is then the expected value of \mathcal{I} over the space of possible states \mathcal{Y} ,

$$H(Y) = \mathbb{E}_{\mathcal{Y}}[\mathcal{I}(Y)] = - \sum_{y \in \mathcal{Y}} p(y) \log p(y). \quad (3.2)$$

H is referred to as the Shannon entropy, or simply the entropy, of Y . The higher the entropy of a random variable, the higher its information content or uncertainty.

Measurements z are outcomes of random variable Z that inform us about Y . To quantify how much information a measurement provides, we calculate the entropy of Y when conditioned on outcome z ,

$$H(Y \mid Z = z) = - \sum_{y \in \mathcal{Y}} p(Y = y \mid Z = z) \log p(Y = y \mid Z = z), \quad (3.3)$$

where the conditional probability $p(y \mid Z = z) = \frac{p(y, z)}{p(z)}$ is understood as the probability that random variable Y takes value y given we know outcome z occurred.

Then, just as before, we can calculate the expected information that Z contains about Y , known as the *conditional entropy*,

$$H(Y \mid Z) = - \sum_{z \in \mathcal{Z}} \sum_{y \in \mathcal{Y}} p(Y = y \mid Z = z) \log p(Y = y \mid Z = z). \quad (3.4)$$

So far we have assumed Y is a *discrete* random variable. This means the notion of the probability of an event $p(y)$ is well defined, allowing us to understand the intuition of entropy by considering the information content of discrete events. However, in the context of robotics, Y and Z are typically *continuous* random variables described by continuous *probability density functions* (PDFs), distinguished notationally from probability mass functions via capitalisation. While $p(y)$ refers to the absolute probability of outcome y , $P(y)$ refers to the PDF over outcome y . Here, probabilities are evaluated by integrating PDFs over intervals. Henceforth, all random variables in this thesis will be continuous unless stated otherwise.

We thus require a continuous analogue for entropy, known as differential entropy, defined by replacing the sum in Equation (3.3) with an integral over the now continuous space of outcomes \mathcal{Y} ,

$$H(Y) = - \int_{\mathcal{Y}} P(y) \log P(y) dy. \quad (3.5)$$

Similarly for conditional entropy,

$$H(Y | Z) = - \int_{\mathcal{Z}} \int_{\mathcal{Y}} P(y | z) \log P(y | z) P(z) dy dz, \quad (3.6)$$

where $P(y | z) = \frac{P(y, z)}{P(z)}$ with $P(y, z)$ the joint distribution of outcomes y and z .

Intuitively, the conditional entropy is a measure of how much information remains about Y given what is known about Z . If $H(Y | Z) = 0$, then the value of Y is completely determined by Z . If Y and Z are independent, then Z contains no information about Y and $H(Y | Z) = H(Y)$. Conditioning on measurements can never *increase* entropy, that is, $H(Y | Z) \leq H(Y)$, and entropy is always non-negative. Further, since this is an expectation over all possible measurements, it is independent of specific measurement outcomes. These are important remarks in the context of Problem 1.1.

With an understanding of entropy and conditional entropy, measures of information gain can be introduced. The *mutual information* of Y relative to Z quantifies the information extracted or uncertainty reduced about Y given our knowledge of Z , defined as

$$I(Y; Z) = H(Y) - H(Y | Z). \quad (3.7)$$

Mutual information can also be expressed in terms of a comparison between the probability density function of Y conditioned on measurement variable Z and the unconditioned density,

$$I(Y; Z) = \mathbb{E}_{\mathcal{Z}} [D_{\text{KL}}(P(y | z), P(y))] \quad (3.8)$$

$$= \int_{\mathcal{Z}} \int_{\mathcal{Y}} P(y | z) \log \left(\frac{P(y | z)}{P(y)} \right) dy dz, \quad (3.9)$$

where D_{KL} is the Kullback-Leibler (KL) divergence, often referred to as the *relative entropy*, a non-negative function of PDFs which takes value 0 when the two PDFs are exactly

identical. However, it is not a true metric as it does not satisfy symmetry or the triangle inequality.

For some distributions the above measures of uncertainty and information gain do not have closed analytic form and are difficult to compute. However, for k -dimensional multivariate normal distributions, i.e. $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, the PDF has analytic form

$$P(\mathbf{y}) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2} (\mathbf{y} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{y} - \boldsymbol{\mu})\right). \quad (3.10)$$

Then, the measures have closed analytic form for k -dimensional normally distributed random variables Y and Z ,

$$H(Y | Z) = \frac{k}{2} \ln 2\pi e + \ln \det \Sigma_{Y|Z}, \quad (3.11)$$

$$D_{\text{KL}}(P(\mathbf{y}), P(\mathbf{z})) = \frac{1}{2} \left[\tilde{\boldsymbol{\mu}}^\top \Sigma_Z^{-1} \tilde{\boldsymbol{\mu}} + \text{Tr}(\Sigma_Z^{-1} \Sigma_Y) - \ln \frac{\det \Sigma_Y}{\det \Sigma_Z} - k \right], \quad (3.12)$$

where $\tilde{\boldsymbol{\mu}} = (\boldsymbol{\mu}_Z - \boldsymbol{\mu}_Y)$ and $\Sigma_{Y|Z} = \Sigma_Y - \Sigma_{YZ} \Sigma_Z^{-1} \Sigma_{ZY}$ is the covariance of Gaussian conditional distribution $P(y|z)$.

3.2 Bayesian filtering

The core of the work in this thesis relies on calculating and updating a robot's belief over a state of interest with measurement data. Bayesian estimation facilitates precisely this. In this section we formalise the notion of a belief and review Bayesian estimation and filtering, tools used throughout the following chapters for this purpose.

A belief is a probabilistic estimate of a state conditioned on measurements²,

$$b_{t|t} = P(y_t | z_{1:t})^3. \quad (3.13)$$

This probability density function is referred to as the *posterior* estimate, posterior belief or simply the posterior. The notation $b_{t|t}$ indicates that it is an estimate of the state at time

²In this thesis we consider conditioning only on $z_{1:t}$ and do not include the control history $u_{1:t}$ as the robot's dynamics are assumed to be deterministic throughout.

³Again states and measurements are notated as scalar, though they can be easily replaced with vectors.

t made with data up to time t . In Bayesian estimation we are concerned with calculating this belief via Bayes' theorem.

Theorem 3.1. (*Bayes' theorem*) *The conditional distributions of two continuous random variables Y and Z with probability density functions $P(y)$ and $P(z)$ are related as follows:*

$$P(y | z) = \frac{P(z | y)P(y)}{P(z)}. \quad (3.14)$$

In the context of Problem 1.1, calculating $b_{t|t} = P(y_t | z_{1:t})$ is a straightforward application of Theorem 3.1. Here, the observation model in Equation (1.3) provides the 'inverse' conditional distribution $P(z_{1:t} | y_t)$, which describes in some sense how measurements are generated by y_t . $P(y_t)$ is the prior distribution reflecting the initial belief before any measurements are taken. $P(z_{1:t})$ is known as the evidence, marginal likelihood or simply the normalisation factor.

In active perception tasks, measurements are taken online and the above calculation is performed recursively with each new z_t to keep the state estimate up to date and enable intelligent online planning. Recursive Bayesian filtering is a framework for *predicting* a prior belief $b_{t|t-1}$ from $b_{t-1|t-1}$ and *updating* it with each new measurement z_t to recover $b_{t|t}$. Each recursion is facilitated by Bayes' theorem.

The prior belief is the estimate at time t before incorporation of information from most recent measurement z_t ,

$$b_{t|t-1} = P(y_t | z_{1:t-1}), \quad (3.15)$$

sometimes also referred to as the prediction. The notation $b_{t|t-1}$ indicates that this distribution is an estimate of the state at time t made with data up to time $t - 1$.

Updating prior $b_{t|t-1}$ with measurement z_t is a simple application of Bayes' theorem,

$$b_{t|t} = P(y_t | z_{1:t}) = \frac{P(z_t | y_t)P(y_t | z_{1:t-1})}{P(z_t | z_{1:t-1})}. \quad (3.16)$$

The denominator $P(z_t | z_{1:t-1}) = \int P(z_t | y_t)P(y_t | z_{1:t-1})dy_t$ is often an unwieldy integral to compute. Thankfully, it is a constant with respect to the state y_t and can thus be

treated as a normalisation factor in practice. After the numerator is evaluated it can be normalised, giving a valid probability distribution.

Prediction of the state at t given data up to $t - 1$ is calculated via

$$b_{t|t-1} = P(y_t | z_{1:t-1}) = \int P(y_t | y_{t-1})P(y_{t-1} | z_{1:t-1})dy_{t-1}, \quad (3.17)$$

where $P(y_t | y_{t-1})$ is the dynamic model in Equation (1.2)). Note we assume dynamics are Markov in nature, such that $P(y_t | y_{1:t-1}) = P(y_t | y_{t-1})$.

Of course, for recursion to be fully enabled an initial belief $b_0 = b_{0|0}$ is required. The choice of distribution for this initial belief impacts the design of the filter, as does the measurement model and dynamic model. As seen above, all three of these factors influence the resulting posterior distribution greatly. These factors additionally influence the computational efficiency of the filter, determining whether posteriors are analytically tractable, or whether they must be approximated. Clearly, Bayesian filtering is a general framework implementable in many different ways. To conclude this section and consolidate understanding a particular implementation relevant to this thesis is given.

The Kalman filter

The Kalman filter is perhaps the most well known of all Bayes filters. It applies the Bayesian filtering framework to systems with linear Gaussian dynamics and observation models. That is to say, state dynamics (recall Equation (1.2)) take the form⁴

$$y_{t+1} = A_{t+1}y_t + w_t, \quad (3.18)$$

with additive Gaussian noise variable $w_t \sim \mathcal{N}(0, Q_t)$. Similarly, the sensor observation model in Equation (1.3) follows

$$z_t = C_t(x_t)y_t + v_t(x_t), \quad (3.19)$$

⁴Sometimes the system can be modelled with an additional input vector u_t . As long as dynamics are linear in this additional argument, the Kalman filter holds.

with $v_t(x_t) \sim \mathcal{N}(0, R_t(x_t))$. Along with these linear Gaussian assumptions, a Gaussian initial belief $b_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$ is assumed. The dependence of C_t and v_t on the robot state x_t will henceforth be dropped for brevity, however it is important to remember sensing location influences measurement outcomes.

The Kalman filter recursively updates the mean and covariance of the belief, beginning from μ_0 and Σ_0 . Linear Gaussian assumptions mean random variables Y_t and Z_t follow Gaussian distributions for all t . The Kalman filter is therefore an instance of a *Gaussian filter*, which reappear frequently in this thesis. Since normal distributions are described completely by their mean and covariance, a Gaussian filter needs only to maintain these two statistics. Moreover, their update has closed form and is generally tractable.

Substituting the relevant distribution functions into Equations (3.16) and (3.17) give the resulting mean and covariances for the prior and posterior beliefs, stated below.

Predict: Following Equation (3.17) the prior belief is given by

$$b_{t|t-1} = \int \underbrace{P(y_t | y_{t-1})}_{\sim \mathcal{N}(A_{t+1}y_t, Q_t)} \underbrace{P(y_{t-1} | z_{1:t-1})}_{\sim \mathcal{N}(\mu_{t|t}, \Sigma_{t|t})},$$

where $b_{t|t-1} \sim \mathcal{N}(\mu_{t|t-1}, \Sigma_{t|t-1})$ with

$$\mu_{t|t-1} = A_t \mu_{t-1|t-1}, \quad (3.20)$$

$$\Sigma_{t|t-1} = A_t \Sigma_{t-1|t-1} A_t^\top + Q_t. \quad (3.21)$$

Update: Following Equation (3.16) and disregarding the normalisation factor we have

$$b_{t|t} \propto \underbrace{P(z_t | y_t)}_{\sim \mathcal{N}(C_t y_t, R_t)} b_{t|t-1},$$

where $b_{t|t} \sim \mathcal{N}(\mu_{t|t}, \Sigma_{t|t})$ with

$$\mu_{t|t} = \mu_{t|t-1} + K_t(z_t - C_t \mu_{t|t-1}), \quad (3.22)$$

$$\Sigma_{t|t} = (\mathbb{I} - K_t C_t) \Sigma_{t|t-1}. \quad (3.23)$$

Here, $K_t = \Sigma_{t|t-1} C_t^\top (C_t \Sigma_{t|t-1} C_t^\top + R_t)^{-1}$ is known as the optimal Kalman gain. It is guaranteed to minimise the mean square error of the estimate.

Note that update and prediction of the covariance of the belief is independent of any future measurements. This means a recursive update of the covariance is possible based only on future sensing positions x_{t+1} reached via control u_t with the *Riccati mapping*

$$\Sigma_{t+1} = \rho_{u_t}(\Sigma_t) = \bar{A}_{t+1}(A_t \Sigma_t A_t^\top + Q_t) \bar{A}_{t+1}^\top + F_{t+1} R_{t+1} F_{t+1}^\top, \quad (3.24)$$

where the shorthand $\Sigma_{t|t} = \Sigma_t$ is used for brevity, $\bar{A}_{t+1} = (I - K_{t+1} C_{t+1})(A_t \Sigma_t A_t^\top + Q_t)$ and $F_{t+1} = K_{t+1}(A_t \Sigma_t A_t^\top + Q_t)$. As covariance is strongly linked to uncertainty and information gain, this is an important result.

3.3 Tree search planning algorithms

There is no dearth of planning algorithms available for use in belief space planning, but those used in this thesis fall under the general category of *tree search* planners. In the context of this thesis – with deterministic robot transition models – tree search algorithms generally represent the robot’s state space as a discrete space enabled by a set of discrete controls. Nodes represent states and edges represent actions taken to transition between states. The expansion and search of such a tree can be performed via the Forward Value Iteration (FVI) algorithm as follows. Starting from a root node one calculates the set of possible next states from the discrete action space, adding new nodes to the next level of the tree. This continues for each node in each new level of the tree until the planning horizon is reached. After the tree is fully expanded, the total cost or reward of any path through the tree can be calculated from root to leaf, and an exhaustive search of all possible paths guarantees the optimal path is recovered.

FVI requires a search over a tree of size $\mathcal{O}(|\mathcal{U}|^T)$ where \mathcal{U} is the action space and T the planning horizon. For large action spaces or long planning horizons this search is impossible complete within the real-time requirements of robotics applications. Consequently, various non-exhaustive search algorithms have been developed to recover sub-optimal solutions in feasible time frames. Two such algorithms are outlined below.

3.3.1 Reduced Value Iteration

The Reduced Value Iteration (RVI) algorithm was introduced in [117] for the purpose of sensor scheduling and studied further in [20] for sensor trajectory planning with linear Gaussian systems. In both cases RVI is introduced in the context of active perception, where a solution represents a policy for sensor scheduling or planning that suboptimally minimises a cost function $J(\Sigma_T)$ of the state estimation covariance at the end of the planning horizon.

RVI expands a tree \mathcal{T} over possible robot states and corresponding beliefs, that is, each node n_t at level t encodes data (x_t, Σ_t) . Expansion of the tree thus requires both a deterministic dynamic model for the sensing robot and a method for propagating belief covariances based only on future sensing states, such as a Riccati mapping. Then, expansion is as per FVI, but with pruning of ‘non-informative’ nodes, reducing the size of the final tree. The extent of pruning is dictated by user-defined tuning parameters which trade computational complexity for optimality of the resulting solution. Moreover, the manner in which pruning is performed allows for derivation of analytical suboptimality bounds, giving an understanding of solution quality – a desirable feature in a planning algorithm.

Pruning is based on the identification of informationally redundant nodes. These nodes have covariance matrices which are dominated by other covariance matrices on the same level of the tree. Under certain assumptions regarding the map used for propagating belief covariances in time, dominated nodes are ensured to remain so for all future times, rendering their future trajectories effectively useless. These assumptions are given below.

Assumption 3.1. *For any $\Sigma_1, \Sigma_2 \in \mathcal{P}^+$, the Riccati mapping ρ is:*

1. *Monotone: if $\Sigma_1 \preceq \Sigma_2$ then $\rho(\Sigma_1) \preceq \rho(\Sigma_2)$*
2. *Concave: $\forall \alpha \in [0, 1], \rho(\alpha\Sigma_1 + (1 - \alpha)\Sigma_2) \succeq \alpha\rho(\Sigma_1) + (1 - \alpha)\rho(\Sigma_2)$.*

From the monotone property one can intuit that if a belief has larger covariance than another at the same time step, it will continue to have larger covariance for all future times. From concavity, if a covariance can be written as the linear combination of two other covariance matrices Σ_1, Σ_2 , its update will be larger than the linear combination of

the updates of Σ_1 and Σ_2 . These properties thus motivate the pruning policy RVI is based on. If at the same sensing location a node has a larger covariance than another or can be written as a linear combination of others, it is considered non-informative or *algebraically redundant* and will remain so for all time. Such nodes are therefore pruned.

Definition 3.1. (*Algebraic redundancy [117]*) Let $\{\Sigma^i\}_{i=1}^K$ be a finite set with $\Sigma^i \in \mathcal{P}^+$ $\forall i$. Then a matrix $\Sigma \in \mathcal{P}^+$ is algebraically redundant with respect to $\{\Sigma^i\}_{i=1}^K$ if there exists a set of non-negative constants $\{\alpha_i\}_{i=1}^K$ such that

$$\sum_{i=1}^K \alpha_i = 1, \quad \Sigma \succeq \sum_{i=1}^K \alpha_i \Sigma^i. \quad (3.25)$$

Pruning algebraically redundant nodes that cross each other (i.e., share a x_t) after each level is expanded can reduce the size of the final tree substantially. Moreover, if the Kalman filter Riccati mapping in Equation (3.24) is used and the cost function J is monotone and concave, the final tree is guaranteed to still have the optimal solution within it [20, 117]. However, further computational savings can be made by introducing flexibility in the definition of algebraic redundancy.

Definition 3.2. (*ϵ -Algebraic redundancy [117]*) Let $\epsilon \geq 0$ and $\{\Sigma^i\}_{i=1}^K$ be a finite set with $\Sigma^i \in \mathcal{P}^+$ $\forall i$. Then a matrix $\Sigma \in \mathcal{P}^+$ is ϵ -algebraically redundant with respect to $\{\Sigma^i\}_{i=1}^K$ if there exists a set of non-negative constants $\{\alpha_i\}_{i=1}^K$ such that

$$\sum_{i=1}^K \alpha_i = 1, \quad \Sigma + \epsilon I \succeq \sum_{i=1}^K \alpha_i \Sigma^i. \quad (3.26)$$

Introducing tuning parameter ϵ no longer guarantees optimality of solutions. However, an upper bound on the resulting suboptimality is known [20, 117]. These bounds depend on ϵ , where optimality is recovered when $\epsilon = 0$.

When planning trajectories in continuous space it may be that no robot states land on precisely the same x_t . Then no nodes will be considered for ϵ -algebraic redundancy and one is faced again with the computational complexity of FVI. In [20] this is managed by introducing further flexibility in the definition of trajectory crossings, broadening the set of nodes checked for ϵ -algebraic redundancy to those that come within δ distance of one another.

Algorithm 1 Reduced Value Iteration

Inputs: $\mathcal{L}_0 = (x_0, \Sigma_0)$, $\epsilon \geq 0$, $\delta \geq 0$, planning horizon T , cost function J .

```

1:  $\mathcal{L}_t = \emptyset \forall t = 1, \dots, T$ 
2: for  $t = 1 : T$  do
3:   for  $(x, \Sigma) \in \mathcal{L}_{t-1}$  do
4:     for  $u \in \mathcal{U}$  do
5:        $\mathcal{L}_t = (f(x, u), \rho_u(\Sigma)) \cup \mathcal{L}_t$        $\triangleright$  Expand new level according to dynamics
6:     end for
7:   end for
8:    $(x^*, \Sigma^*) = \arg \min_{\mathcal{L}_t} J(\Sigma_t)$        $\triangleright$  Find the lowest cost node
9:    $\mathcal{L}'_t = \{(x^*, \Sigma^*)\}$        $\triangleright$  Initialise set of nodes to keep in tree
10:  for  $(x, \Sigma) \in \mathcal{L}_t \setminus (x^*, \Sigma^*)$  do
11:    for  $(x', \Sigma') \in \mathcal{L}'_t$  do
12:       $Q = \{\Sigma \mid d_{\mathcal{X}}(x, x') \leq \delta\}$        $\triangleright$  Collect nodes in  $\mathcal{L}'_t$  that  $\delta$ -cross  $x$ 
13:    end for
14:    if  $Q = \emptyset$  or not ISREDUNDANT( $\Sigma, Q, \epsilon$ ) then
15:       $\mathcal{L}'_t = (x, \Sigma) \cup \mathcal{L}'_t$        $\triangleright$  If no  $\delta$ -crossings or not redundant, keep the node
16:    end if
17:  end for
18:   $\mathcal{L}_t = \mathcal{L}'_t$ 
19: end for
20: return  $\min_{\mathcal{L}_T} J(\Sigma_T)$ 

```

Definition 3.3. (*δ -Crossing trajectories [20]*) Two sensor trajectories δ -cross at time $t \in \{1, \dots, T\}$ if for some $\delta \geq 0$ and metric $d_{\mathcal{X}}$,

$$d_{\mathcal{X}}(x_t^1, x_t^2) \leq \delta. \quad (3.27)$$

For statements of suboptimality to be made under this less stringent pruning policy, continuity assumptions regarding the robot dynamics and sensor observation model are introduced.

Assumption 3.2. [20] *The sensor motion model f is Lipschitz continuous in x with Lipschitz constant $L_f \geq 0$ for every fixed control $u \in \mathcal{U}$, i.e.*

$$d_{\mathcal{X}}(f(x_1, u), f(x_2, u)) \leq L_f d_{\mathcal{X}}(x_1, x_2).$$

In other words, sensing trajectories that come close to one another remain close to one another if the same control is applied. Applying this recursively implies that under identical

control sequences, such sensing trajectories remain close for all time.

Assumption 3.3. [20] For any two nodes $(x_{t-1}^1, \Sigma_{t-1}^1)$, $(x_{t-1}^2, \Sigma_{t-1}^2)$, let Σ_t^1 , Σ_t^2 be the updated state estimation covariances after applying control $u \in \mathcal{U}$ to each node. Then

$$\begin{aligned}\Sigma_t^1 &\succeq \gamma \Sigma_t^2 + (1 - \gamma)Q_{t-1}, \\ \Sigma_t^2 &\succeq \gamma \Sigma_t^1 + (1 - \gamma)Q_{t-1},\end{aligned}$$

$\forall t \in \{1, \dots, T\}$, where $\gamma = (1 + L_m d_{\mathcal{X}}(x_t^1, x_t^2))^{-1} < 1$ for some $L_m > 0$. Note for any $\delta > 0$, if $d_{\mathcal{X}}(x_{t-1}^1, x_{t-1}^2) < \delta$ then $\gamma = (1 + L_m L_f \delta)^{-1} < 1$.

That is, sensing from similar locations gives similar covariance belief updates. Together these assumptions formalise the intuition that two trajectories that come close to one another will remain close in both physical distance and informativeness. Based on this intuition the goal is then to reduce search complexity by pruning nodes that are likely to produce similar trajectories in the future.

Pseudocode for the full Reduced Value Iteration algorithm with ϵ, δ redundancy pruning is given in Algorithm 1. New levels \mathcal{L}_t for $t = 1, \dots, T$ are expanded by applying each control u in the admissible set \mathcal{U} to each parent node in level \mathcal{L}_{t-1} (lines 3 to 7). After expansion, a set \mathcal{L}'_t is created to store all nodes that will be kept in the tree, initialised with the lowest cost node (lines 8 to 9). Then, δ -crossing nodes are collected and checked for ϵ -algebraic redundancies. Any non-redundant nodes are added to \mathcal{L}'_t (lines 11 to 14). The remaining are pruned.

Discarding trajectories in this manner reduces the complexity of the search algorithm vastly while incurring a minimal loss in optimality. Once again, the size of this loss is characterised by an upper bound derived for the Kalman filter Riccati mapping in [20]. In Chapter 4 suboptimality bounds are derived for another Riccati mapping suited for motion models that are not purely linear Gaussian, but rather have an unidentified input in their dynamics.

3.3.2 Monte Carlo tree search

Some tree search methods opt to de-prioritise the expansion of certain nodes rather than pruning them all together. In Monte Carlo tree search (MCTS) the prioritisation of nodes for expansion is most commonly based on a balance of exploring new nodes and exploiting existing nodes with promising expected return [165]. MCTS has been utilised for planning in a number of robotics applications such as solving mixed integer problems, the travelling salesman problem and belief space planning [166]. Here the algorithm is outlined in the context of active perception.

The core concept of MCTS is to build a search tree iteratively until some user-defined termination condition such as a time or iteration budget is met. Nodes again represent a robot and belief state and edges represent actions taken to transition between nodes. With each iteration of the algorithm, a new node is added to the tree whose value is estimated via a simulation of a possible future sensing trajectory, or a *rollout*. This estimated value is then used in future iterations to guide future expansion. In the context of this thesis, the expected amount of information gathered throughout the simulation is used to estimate the expected value or reward of the node.

Expansion of the tree is facilitated by the upper confidence bound for trees (UCT) policy for selection of nodes. A child node v' of parent v is selected if it maximises the UCT,

$$\text{UCT} = \underbrace{\frac{R(v')}{N(v')}}_{\text{exploitation}} + c \underbrace{\sqrt{\frac{2 \ln N(v)}{N(v')}}}_{\text{exploration}}, \quad (3.28)$$

where $R(v')$ is the *cumulative* reward collected over $N(v')$ rollouts from v' , and $N(v)$ is the number of times the parent node has been visited. $c > 0$ is a constant that dictates how much exploration is allowed. Sampling is then guided by a balance of selecting children with high expected reward and children that have not been visited often or at all (a child with $N(v') = 0$ has infinite UCT). Re-sampling incorporates more rollout evaluations into the expected value of a node, improving accuracy.

In more detail, the four key steps of the algorithm pseudocode in Algorithm 2 are:

Algorithm 2 Monte Carlo Tree Search

Inputs: Root node v_0 , $c > 0$, reward function r .

- 1: **while** termination condition not met **do**
 - 2: $v_l = \text{SELECTION}(v_0)$
 - 3: $v^* = \text{EXPANSION}(v_l)$
 - 4: $\Delta r = \text{ROLLOUT}(v^*)$
 - 5: $\text{BACKPROPAGATION}(\Delta r)$
 - 6: **end while**
 - 7: **return** $\arg \max_{\mathcal{L}_1} R(v_1)/N(v_1)$
-

1. *Selection.* Starting from the root, traverse the existing tree according to UCT until a leaf node v_l is found.
2. *Expansion.* Expand the children of v_l , adding new leaves to the tree by applying actions $u \in \mathcal{U}$ to v_l . Select from these children a node v^* to rollout at random.
3. *Rollout.* Simulate a future sensing trajectory by randomly sampling actions from U until a terminal node is reached. Evaluate the reward Δr associated with this simulation according to reward function r . For active perception tasks, this is a measure of information gained or uncertainty reduced along the trajectory.
4. *Backpropagation.* Update the cumulative reward $R(v)$ of all tree nodes v traversed in the selection step by adding rollout reward Δr . Similarly, increment the number of visits $N(v)$ to these selected nodes by 1.

Generally, when the termination condition is met, the path through the expanded tree with highest reward is returned. However, in active perception settings replanning is required with each new belief update. Then, the child of the root node with highest expected reward is returned and executed.

Chapter 4

The state belief space for active estimation of unknown inputs

To begin exploring reduced latent belief spaces for active perception we study active estimation of linear Gaussian systems with unknown inputs. Such systems are common in robotics where there are spurious state dynamics which cannot be captured by the inclusion of Gaussian noise in the dynamic model. These unknown inputs can be present in, for example, cyber-physical systems undergoing faults or attacks, targets performing abrupt manoeuvres, or any system with spurious signals due to environmental disturbances. In these cases, a full description of the system under estimation can be written $\mathbf{y}_t^S = [\mathbf{y}_t, \mathbf{d}_t]^\top$ where \mathbf{y}_t is the state vector and \mathbf{d}_t is the unknown input. Rather than estimating these two components passively, we wish to actively plan sensing actions to obtain the best estimates possible.

This is a joint estimation problem that brings with it a joint belief space of larger dimensionality and complexity than single component estimation. One may imagine that optimal sensing locations for unknown input estimation may diverge from the optimal sensing locations for state estimation and vice versa. In this chapter we address this challenge by focusing on the reduced belief space associated with only the state \mathbf{y}_t , avoiding planning over the joint belief space entirely. That is, Problem 1.1 where information must

be collected regarding \mathbf{y}_T^S is reduced to Problem 1.2, where $\boldsymbol{\xi}_T = \mathbf{y}_T$, a sub-component of the full system description \mathbf{y}_T^S . We show this approach is made feasible via the structure of the *unknown input filter* introduced in [22] which allows \mathbf{d}_t to be inferred from \mathbf{y}_t , in turn facilitating inference of the full system description \mathbf{y}_t^S as in Figure 1.3(b).

First, linear Gaussian systems with unknown inputs and the unknown input filter are introduced. Then, our extension of the Reduced Value Iteration algorithm for planning in the reduced state belief space is presented. Suboptimality bounds for state estimation resulting from the proposed planning framework are derived. To study the efficacy of the state-optimised plan as a solution for estimation of the full system description \mathbf{y}_T^S , we also derive suboptimality bounds for the resulting unknown input estimate. Finally, the approach is evaluated in a simulated active target tracking application. The work in this chapter was first presented in [167].

4.1 Preliminaries and problem formulation

4.1.1 Preliminaries of filtering under unknown inputs

To begin, we introduce linear Gaussian systems with unknown inputs and the filtering framework established in [22]. In such systems, the motion model in Equation (1.2) takes form

$$\mathbf{y}_{t+1} = A_t \mathbf{y}_t + G_t \mathbf{d}_t + \mathbf{w}_t, \quad (4.1)$$

where $\mathbf{y}_t \in \mathbb{R}^{n_y}$ is the state vector and the process noise $\mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, Q_t)$ is normally distributed with zero-mean and covariance $Q_t \in \mathcal{P}^+$. A_t, G_t are known matrices. Most importantly, $\mathbf{d}_t \in \mathbb{R}^{n_d}$ represents arbitrary, unknown inputs whose models or statistical properties are *not assumed to be known*.

While in operation, the sensing robot has motion model according to Equation (1.1) and observation model:

$$\mathbf{z}_t = C_t(\mathbf{x}_t) \mathbf{y}_t + \mathbf{v}_t(\mathbf{x}_t), \quad (4.2)$$

where $\mathbf{z}_t \in \mathbb{R}^{n_z}$ is the measurement, $C_t(\mathbf{x}_t) \in \mathbb{R}^{n_z \times n_y}$ is a known measurement matrix, and the measurement noise $\mathbf{v}_t(\mathbf{x}_t) \sim \mathcal{N}(\mathbf{0}, R_t(\mathbf{x}_t))$ with $R_t(\mathbf{x}_t) \in \mathcal{P}^+$. Throughout this chapter it is assumed that (C_t, A_t) is observable and $\text{rank}(C_t G_{t-1}) = \text{rank}(G_{t-1}) = n_d$ holds true for all t . Observability of the state is required for existence of the estimator, while the rank constraint ensures the estimator is unbiased in the presence of the unknown input. For brevity we drop dependence of C_t, R_t, \mathbf{v}_t on the sensing robot state \mathbf{x}_t in the remainder of this chapter, however it is important to introduce here in order to emphasise the impact of sensing locations on measurements.

In Chapter 3 the Kalman filter was described as a tool for facilitating belief space planning for linear Gaussian systems. Here, it is unsuitable due to the presence of the unknown input \mathbf{d}_t . Fortunately, extensions of the Kalman filter have been proposed to handle unknown inputs [22, 88, 104]. In this chapter, we adopt the unknown input filter in [22]. This filter deconstructs the full system description into two decoupled components: the state and the unknown input. Then, a belief is maintained over each component separately via the following steps:

1. Time update:

$$\hat{\mathbf{y}}_{t|t-1} = A_t \hat{\mathbf{y}}_{t-1|t-1}, \quad (4.3)$$

2. Unknown input estimation:

$$\hat{\mathbf{d}}_{t-1} = M_t(\mathbf{z}_t - C_t \hat{\mathbf{y}}_{t|t-1}), \quad M_t \in \mathbb{R}^{n_d \times n_z}, \quad (4.4)$$

3. Measurement update:

$$\begin{aligned} \hat{\mathbf{y}}_{t|t}^* &= \hat{\mathbf{y}}_{t|t-1} + G_{t-1} \hat{\mathbf{d}}_{t-1} \\ \hat{\mathbf{y}}_{t|t} &= \hat{\mathbf{y}}_{t|t}^* + K_t(\mathbf{z}_t - C_t \hat{\mathbf{y}}_{t|t}^*), \quad K_t \in \mathbb{R}^{n_y \times n_z}. \end{aligned} \quad (4.5)$$

Define

$$\begin{aligned} \tilde{\mathbf{d}}_{t-1} &= \mathbf{d}_{t-1} - \hat{\mathbf{d}}_{t-1}, \quad \Sigma_{t-1}^d = \mathbb{E}[\tilde{\mathbf{d}}_{t-1} \tilde{\mathbf{d}}_{t-1}^T], \\ \tilde{\mathbf{y}}_{t|t} &= \mathbf{y}_t - \hat{\mathbf{y}}_{t|t}, \quad \Sigma_t = \mathbb{E}[\tilde{\mathbf{y}}_{t|t} \tilde{\mathbf{y}}_{t|t}^T], \end{aligned} \quad (4.6)$$

as the unknown input estimation error, the state estimation error, and their respective covariances.

As shown in [22], $\hat{\mathbf{d}}_{t-1}$ in Equation (4.4) is an unbiased estimate if and only if the state estimate in Equation (4.5) is unbiased, and the unknown input gain M_t satisfies

$$M_t C_t G_{t-1} = I_{n_d}. \quad (4.7)$$

However, this condition alone is not enough to ensure the input estimate is a *minimum variance* unbiased (MVU) estimate, in the sense that it does not ensure minimisation of the trace of the input error covariance introduced in Equation (4.6). For minimum variance to be achieved, the input gain must take the form

$$M_t^*(\Sigma_{t-1}) = (F_t^\top \tilde{R}_t^{-1}(\Sigma_{t-1}) F_t)^{-1} F_t^\top \tilde{R}_t^{-1}(\Sigma_{t-1}), \quad (4.8)$$

where $F_t = C_t G_{t-1}$, $\tilde{R}_t(\Sigma_{t-1}) = C_t(A_{t-1}\Sigma_{t-1}A_{t-1}^\top + Q_{t-1})C_t^\top + R_t \in \mathcal{P}^+$ and $\Sigma_{t-1} = \Sigma_{t-1|t-1}$ is the filtered state error covariance at time step $t-1$. Given Equation (4.8), one may transform the state estimation problem into a standard Kalman filtering problem and find a resulting MVU state gain matrix K_t^* . The resulting optimal gain K_t^* is in general non-unique [22]. For simplicity, in this chapter we take the choice

$$K_t^*(\Sigma_{t-1}) = (A_{t-1}\Sigma_{t-1}A_{t-1}^\top + Q_{t-1})C_t^\top \tilde{R}_t^{-1}. \quad (4.9)$$

The optimal filter gains in Equations (4.8) and (4.9) admit the following state and unknown input error covariance update maps respectively:

$$\Sigma_t = \rho(\Sigma_{t-1}, M_t^*, K_t^*) = \tilde{A}_t \Sigma_{t-1} \tilde{A}_t^\top + \tilde{F}_t Q_{t-1} \tilde{F}_t^\top + \tilde{W}_t R_t \tilde{W}_t^\top, \quad (4.10)$$

$$\Sigma_{t-1}^d = \rho^d(\Sigma_{t-1}) = (F_t^\top \tilde{R}_t^{-1}(\Sigma_{t-1}) F_t)^{-1}, \quad (4.11)$$

where

$$\begin{aligned} \tilde{A}_t &= (I - K_t^* C_t)(I - G_{t-1} M_t^* C_t) A_{t-1}, \\ \tilde{F}_t &= -(I - K_t^* C_t)(I - G_{t-1} M_t^* C_t), \\ \tilde{W}_t &= G_{t-1} M_t^* - K_t C_t G_{t-1} M_t^* + K_t^*. \end{aligned}$$

Note that ρ, ρ^d are functions of \mathbf{x}_t via C_t and R_t and thus of the control input \mathbf{u}_{t-1} . We therefore denote $\rho_{\mathbf{u}_t}(\Sigma_{t-1}), \rho_{\mathbf{u}_t}^d(\Sigma_{t-1})$ to refer to updates applied under the control $\mathbf{u}_t \in \mathcal{U}$.

4.1.2 Problem formulation

We wish to solve Problem 1.1 to best track the evolution of the full system description, $\mathbf{y}_t^S = [\mathbf{y}_t, \mathbf{d}_t]^\top$. Owing to the decoupling of \mathbf{y}_t and \mathbf{d}_t in during unknown input filtering, we can plan over the belief space of each component separately rather than considering the joint belief space. Then, the problem can be written as a minimisation of the total uncertainty,

Problem 4.1. *Given an initial belief of the target state \mathbf{y}_0 , find a sequence of admissible controls $\sigma = \{\mathbf{u}_0, \dots, \mathbf{u}_{T-1}\}$ that minimises the total uncertainty in target state and unknown input. That is,*

$$\min_{\sigma \in \mathcal{U}^T} \log \det(\Sigma_T) + \log \det(\Sigma_{T-1}^d) \quad (4.12)$$

$$\begin{aligned} \text{s.t. } \quad & \mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t), \quad t = 0, \dots, T-1, \\ & \Sigma_{t+1} = \rho_{\mathbf{u}_t}(\Sigma_t), \quad t = 0, \dots, T-1, \\ & \Sigma_t^d = \rho_{\mathbf{u}_t}^d(\Sigma_t), \quad t = 1, \dots, T-1, \end{aligned}$$

where $\rho_{\mathbf{u}_t}(\Sigma_t)$, $\rho_{\mathbf{u}_t}^d(\Sigma_t)$ are the state and unknown input error covariance update maps defined in Equations (4.10) and (4.11), with the first measurement taken at $t = 1$.

Although the unknown inputs are not assumed to follow any specific probability distribution, we ideally wish to solve the above in a Gaussian belief space belief transitions are determined. To this end, one could give some statistical interpretation of the optimisation problem similar to existing work for the case without unknown inputs [20, 116]. This can be done by following concepts in [168] to firstly pose the unknown input as a Gaussian noise process with variance D to recover Gaussian belief space planning. The lack of prior information regarding the unknown input can then be expressed by taking D to infinity.

Finding the optimal solution to Problem 4.1 amounts to exploring each Gaussian belief space separately and finding an optimal sensing path. Although the belief spaces are decoupled, finding an optimal sensing path is still challenging, as one must decide how to balance each component in the information-theoretic cost function in Equation (4.12) at each time step.

However, there is a close relationship between the state and unknown input estimates in Equations (4.4) and (4.5). Namely, the unknown input estimate in Equation (4.4) lags behind the state estimate by one time step. Further, the unknown input error covariance update map Equation (4.11) is not a true Riccati mapping, as it is not a recursive function. Instead, it is a function of the *state* covariance matrix Equation (4.6) at the previous time step. These connections imply that the unknown input belief and thus the full system belief at $t - 1$ can be inferred from measurements up to time t and the prior belief $\hat{\mathbf{y}}_{t|t-1}$. Thus, we may plan over the state belief space *only* while still maintaining good unknown input estimation.

The remaining sections of this chapter therefore address the following belief space planning problem, an instance of the general reduced latent belief space planning problem introduced in Chapter 1.

Problem 4.2. *Given an initial belief of the target state \mathbf{y}_0 , find a sequence of admissible controls $\sigma = \{\mathbf{u}_0, \dots, \mathbf{u}_{T-1}\}$ that minimises the uncertainty about the state only. That is,*

$$\min_{\sigma \in \mathcal{U}^T} \log \det(\Sigma_T) \quad (4.13)$$

$$\begin{aligned} \text{s.t. } \quad & \mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t), \quad t = 0, \dots, T-1, \\ & \Sigma_{t+1} = \rho_{\mathbf{u}_t}(\Sigma_t), \quad t = 0, \dots, T-1, \end{aligned}$$

Further, suboptimality bounds for both state and unknown input tracking that result from this simplified approach are derived.

4.1.3 Planning framework

To search the reduced belief space for an optimal sensing trajectory we extend the RVI algorithm introduced in [20, 117] and detailed in Chapter 3 to our proposed belief space. The implementation is as in Algorithm 1, with the unknown input filter's state covariance update map in Equation (4.10) used to expand the belief tree in lines 3 to 7 rather than the Kalman filter's Riccati map. Additionally, the cost function J in this implementation is $\log \det(\Sigma_T)$.

We make use of RVI for its reduced computational complexity and intelligent pruning strategy that facilitates the derivation of suboptimality bounds. These derivations are strongly tied to the form of the Kalman filter Riccati mapping in Equation (3.24) and rely on its concavity and monotonicity (recall Assumption 3.1). As such, the existing suboptimality bounds in [20, 117] derived for exploring linear Gaussian belief spaces no longer hold for exploring our proposed belief space. In the following sections we prove concavity and monotonicity of Equations (4.10) and (4.11) and derive new suboptimality bounds for state and unknown input estimation associated with solutions to Problem 4.2 found via our extension of RVI to this new belief space.

4.2 Suboptimality bounds for active state estimation

The suboptimality of state estimation incurred by pruning algebraically redundant nodes according to Equation (3.26) can be upper bounded via a worst-case analysis as in [20]. This worst-case analysis assumes the optimal node is pruned from the search tree early and thus the optimal path is not expanded any further. Thus the suboptimality bounds quantify how this error propagates as each new level of the search tree is expanded. This analysis is outlined below, with all proofs of results available in Appendix A.

The following properties of the state estimation covariance update map in Equation (4.10) are key in deriving suboptimality bounds.

Lemma 4.1. *The unknown input filter state estimation error covariance update map is:*

1. *Monotone: if $\Sigma_1 \preceq \Sigma_2$ then $\rho(\Sigma_1) \preceq \rho(\Sigma_2)$*
2. *Concave: $\forall \alpha \in [0, 1], \rho(\alpha\Sigma_1 + (1 - \alpha)\Sigma_2) \succeq \alpha\rho(\Sigma_1) + (1 - \alpha)\rho(\Sigma_2)$.*

It is important in our worst-case analysis to consider recursive update of the error covariance over a long horizon. We therefore introduce the ‘ t -horizon’ mapping $\phi_t : \Sigma_0 \mapsto \Sigma_t$, which maps the state error covariance matrix at time 0 to time t according to the first t elements of the control sequence σ :

$$\phi_t^\sigma(\Sigma_0) = \rho_{\mathbf{u}_{t-1}}(\dots \rho_{\mathbf{u}_1}(\rho_{\mathbf{u}_0}(\Sigma_0))) = \Sigma_t. \quad (4.14)$$

Monotonicity and concavity of the t -horizon mapping naturally follow from Lemma 4.1 and the definition in Equation (4.14). As a direct result of concavity, the t -horizon mapping ϕ_t is bounded by its first order Taylor approximation, i.e.

$$\phi_t^\sigma(\Sigma + \epsilon X) \preceq \phi_t^\sigma(\Sigma) + \epsilon g_t^\sigma(\Sigma, X), \quad (4.15)$$

where

$$g_t^\sigma(\Sigma, X) = \left. \frac{d\phi_t^\sigma(\Sigma + \epsilon X)}{d\epsilon} \right|_{\epsilon=0}$$

is the directional derivative of the t -horizon mapping ϕ_t at $\Sigma \in \mathcal{P}^+$ along an arbitrary direction $X \in \mathcal{P}^+$. Intuitively, the directional derivative $g_t^\sigma(\Sigma, X)$ can be interpreted as the impact an early perturbative error will have on the covariance at a later time t . This interpretation is pertinent for our worst-case analysis if one frames the ϵI term in Equation (3.26) as a perturbative error causing pruning of the optimal node. This motivates the study of the directional derivative.

Lemma 4.2. *The directional derivative of the state estimation covariance update map at $\Sigma \in \mathcal{P}^+$ along the arbitrary direction $X \in \mathcal{P}^+$ is given by*

$$\left. \frac{d\rho_u(\Sigma + \epsilon X)}{d\epsilon} \right|_{\epsilon=0} = \tilde{A}(\Sigma) X \tilde{A}(\Sigma)^\top,$$

where $\tilde{A}(\Sigma)$ is defined as in Equation (4.10). Further, the directional derivative of the t -horizon mapping ϕ_t at $\Sigma \in \mathcal{P}^+$ along an arbitrary direction $X \in \mathcal{P}^+$ is given by

$$g_t^\sigma(\Sigma, X) = \prod_{k=0}^{t-1} (\tilde{A}_{t-k}) X \prod_{k=0}^{t-1} (\tilde{A}_k)^\top,$$

$\forall t \in \{1, \dots, T\}$, with $g_0^\sigma(\Sigma, X) = X$.

The trace of the directional derivative of the t -horizon mapping can quantify the impact of such a perturbation at a later time step.

Lemma 4.3. *Suppose $\exists \beta < \infty$ such that $\Sigma_t \preceq \beta I \forall t \in \{0, \dots, T\}$, then we have*

$$\text{Tr}\{g_t^\sigma(\Sigma, X)\} \leq \beta \eta^k \text{Tr}\{\Sigma^{-1} X\},$$

where $\eta = \frac{\beta}{\beta + \lambda_Q} < 1$ and $\underline{\lambda}_Q$ is the minimum eigenvalue of $\tilde{F}_t Q_{t-1} \tilde{F}_t^\top \forall t \in \{0, \dots, T\}$.

As in [20, 117], the above bound implies that provided the state error covariance is bounded for all time, the effect of a perturbation at an early time step *decays exponentially* as time evolves. The culmination of utilising the above results in the worst-case performance analysis is an upper bound on the suboptimality of the state error covariance $\Sigma_T^{\epsilon, \delta}$ found by RVI.

Theorem 4.1. *Let $\beta^* < \infty$ be the peak state estimation error of the optimal trajectory, i.e. $\Sigma_t^* \preceq \beta^* I \forall t \in \{1, \dots, T\}$. Then we have*

$$0 \leq J(\Sigma_T^{\epsilon, \delta}) - J(\Sigma_T^*) \leq (\zeta_T - 1) (J(\Sigma_T^*) - J(\lambda_Q I)) + \epsilon \left(\frac{n_y}{\lambda_Q} + \Delta_T \right), \quad (4.16)$$

where $\zeta_t := \prod_{\tau=1}^{t-1} \left(1 + \sum_{s=1}^{\tau} L_f^s L_m \delta \right) \geq 1$, $\Delta_T := \frac{n_y}{\lambda_Q} \beta^* \sum_{\tau=1}^{T-1} \frac{\zeta_T}{\zeta_\tau} \eta_*^{T-\tau}$, $\eta_* = \frac{\beta^*}{\beta^* + \lambda_Q} < 1$.

This bound is similar to the state estimation bound in [20], but derived for the unknown input filter Riccati map in Equation (4.10) and time-varying Q_t . Similar to [20, 117], the performance bound in Theorem 4.1 grows with δ and ϵ , the tunable parameters that dictate pruning. For $\epsilon, \delta = 0$ we recover the optimal solution.

4.3 Suboptimality bounds for active unknown input estimation

Now we exploit the connection between the state and unknown input to derive concrete suboptimality bounds for unknown input estimation resulting from taking a state-optimised sensing plan found via our RVI extension. Thus, we complete the characterisation of the full system description estimate that results from solving the reduced latent belief space planning problem in Problem 4.2. Proofs are again provided in Appendix A.

Once again we introduce a ‘ t -horizon’ update map for unknown input estimation error,

$$\phi_t^d : \Sigma_0 \mapsto \Sigma_{t-1}^d$$

$$\phi_t^{d\sigma}(\Sigma_0) = \rho^d(\phi_{t-1}^\sigma(\Sigma_0)) = (F_t^\top \tilde{R}_t^{-1}(\Sigma_{t-1}) F_t)^{-1}, \quad (4.17)$$

where ϕ_{t-1}^σ is as in Equation (4.14). From this definition the connection between the unknown input and state belief spaces is further highlighted. We see the control sequence $\sigma^* \in \mathcal{U}^T$ which solves the reduced problem should give Σ_{T-1}^* which minimises Equation (4.17). However, as $F_T(\mathbf{x}_T) = C_T(\mathbf{x}_T)G_{T-1}$, the unknown input error covariance in Equation (4.17) has two arguments Σ_{T-1} and \mathbf{x}_T . The sensor state \mathbf{x}_T^* found by solving Problem 4.2 with the proposed RVI extension may not coincide with the sensor state \mathbf{x}_T^{d*} required to optimise unknown input estimation at the final time step of the planning horizon. This is an important observation that has direct impact on the performance of unknown input tracking under a control sequence tailored for state evolution tracking optimization.

Monotonicity and concavity of the unknown input error covariance update map are again crucial properties for describing the evolution of nodes.

Lemma 4.4. *The unknown input error covariance update map $\rho^d(\cdot)$ is monotone and concave.*

These properties extend to the t -horizon input estimation error update map in Equation (4.17). Thus, $\phi_t^{d\sigma}$ is bounded from above by its first order Taylor approximation. We can again characterize the directional derivative $g_{t-1}^{d\sigma}(\Sigma, X)$.

Lemma 4.5. *The directional derivative of $\phi_t^{d\sigma}$ at $\Sigma \in \mathcal{P}^+$ in the direction $X \in \mathcal{P}^+$ is given by*

$$\begin{aligned} g_{t-1}^{d\sigma}(\Sigma, X) &= \left. \frac{d}{d\epsilon} \phi_{t-1}^d(\Sigma + \epsilon X) \right|_{\epsilon=0} \\ &= M_t^* C_t A_{t-1} g_{t-1}^\sigma(\Sigma, X) A_{t-1}^\top C_t^\top M_t^{*\top}, \end{aligned}$$

where $g_{t-1}^\sigma(\Sigma, X)$ is the directional derivative of the state t -horizon update map.

As in Lemma 4.3, we find the effect of a perturbation in the state error covariance on the unknown input error covariance dampens with time provided Σ_t^d and Σ_t are bounded for all t .

Lemma 4.6. *Suppose $\exists \beta^d < \infty$ such that $\Sigma_t^d \preceq \beta^d I \forall t \in \{1, \dots, T\}$. Then*

$$\text{Tr}\{g_{t-1}^{d\sigma}(\Sigma, I)\} \leq (n_d)^2 (\beta^d)^2 \bar{\lambda}_{\tilde{G}} \text{Tr}\{g_{t-1}^\sigma(\Sigma, I)\},$$

where $\bar{\lambda}_{\tilde{G}}$ is the maximum eigenvalue of $G_{t-1}^\top H_t A_{t-1} A_{t-1}^\top H_t G_{t-1} \in \mathbb{R}^{n_d \times n_d}$.

The propagated error incurred on the unknown input covariance by a perturbation in the state covariance is therefore a multiple of that found for the state. Hence, given the state result in Lemma 4.3, the unknown input analogue can also be found. We now provide an upper bound on the final unknown input error covariance found by solving the reduced Problem 4.2 via RVI.

Theorem 4.2. *Let $\beta^* < \infty$, $\beta^{d*} < \infty$ be the peak state and input estimation errors of the optimal trajectory respectively. That is, $\Sigma_t^* \preceq \beta^* I$ and $\Sigma_{t-1}^{d*} \preceq \beta^{d*} I \forall t \in \{1, \dots, T\}$. Then*

$$0 \leq J(\Sigma_{T-1}^{d,(\epsilon,\delta)}) - J(\Sigma_{T-1}^{d*}) \leq (\zeta_T - 1) \left(J(\Sigma_{T-1}^{d*}) + J(\gamma^{d*} I) - J(\bar{\lambda}_H^{-1} I) \right) + \epsilon(\Delta_T^d),$$

where $\Delta_T^d := (\gamma^{d*})^{-1} (n_d)^2 \bar{\lambda}_H \bar{\lambda}_{\tilde{G}} (\beta^{d*})^2 \underline{\lambda}_Q \Delta_T$, $\gamma^{d*} = (1 + L_m d(\mathbf{x}_T^*, \mathbf{x}_T^{d*}))^{-1}$ and $\bar{\lambda}_H$ is the maximum eigenvalue of $G_{T-1}^\top H_T G_{T-1}$.

We observe the same behaviours of the unknown input bounds with respect to δ, ϵ as the state bounds found in Theorem 4.1. Here Δ_T^d is a factor of Δ_T from the state bounds, again highlighting the close relationship between the two bounds. Most notably, we see the bound grows with $(\gamma^{d*})^{-1}$. This result is expected – if the distance $d(\mathbf{x}_T^*, \mathbf{x}_T^{d*})$ between optimal sensor positions for state estimation and unknown input estimation is large, the performance of unknown input estimation under a policy optimised for state estimation via RVI worsens.

4.4 Experiment results

To evaluate the capability of the proposed reduced belief space for the joint active estimation problem we validate our theoretical findings via simulation. We simulate a

two-dimensional active target tracking problem where the target is performing unmodelled evasive manoeuvres. To limit test our proposed approach, we inject an additional non-myopic element to the tracking problem. Our theoretical findings indicate that by optimising for state estimation only we can recover accurate estimates of the full system description. However, if in reality the resulting unknown input estimation is poor, then active target tracking performance can suffer in non-myopic conditions. For example, the target may perform an evasive manoeuvre in a low-visibility or occluded area. If the robot's sensing plan does not anticipate this, the target may be lost. On the other hand, if the proposed latent variable $\boldsymbol{\xi}_t = \mathbf{y}_t$ and corresponding problem reduction is indeed a suitable proxy for Problem 4.1, the RVI robot should be able to accurately infer \mathbf{d}_t via its state-optimised sensing plan and track the target even in non-myopic conditions.

Suppose a sensing robot with state \mathbf{x}_t defined by its position-velocity vector has the following constant-velocity dynamic model:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t) := \begin{pmatrix} x_t^1 \\ x_t^2 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} u_t^1 \cos(u_t^2) \tau \\ u_t^1 \sin(u_t^2) \tau \\ u_t^1 \cos(u_t^2) \\ u_t^1 \sin(u_t^2) \end{pmatrix}, \quad (4.18)$$

with control input $\mathbf{u}_t \in \mathcal{U}$, where $\mathcal{U} = \{(u_t^1, u_t^2) \mid u_t^1 \in \{0, 1, 2\}, u_t^2 \in \{0, \pm\pi/2, \pi\}\}$ and τ is a small time translation. The goal of the robot is to track and estimate the position and velocity of a constant-velocity vehicle driven by Gaussian noise and an unknown input \mathbf{d}_t in the form of abrupt accelerations:

$$\mathbf{y}_{t+1} = \begin{bmatrix} I_2 & \tau I_2 \\ \mathbf{0} & I_2 \end{bmatrix} \mathbf{y}_t + \begin{bmatrix} \tau^2/2I_2 \\ \tau I_2 \end{bmatrix} \mathbf{d}_t + \mathbf{w}_t, \quad (4.19)$$

$$\mathbf{w}_t \sim \mathcal{N}\left(\mathbf{0}, q \begin{bmatrix} \tau^3/3I_2 & \tau^2/2I_2 \\ \tau^2/2I_2 & \tau I_2 \end{bmatrix}\right),$$

where $\mathbf{y}_t = [y_t^1, y_t^2, \dot{y}_t^1, \dot{y}_t^2]^\top$ is the position-velocity vector of the target state at time t and q is a diffusion strength scalar. The tracking takes place over 51 time steps. At $t \in \{4, 9, 19, 24, 34, 39\}$, \mathbf{d}_t is a maneuver that takes form of a sharp acceleration in a random direction.

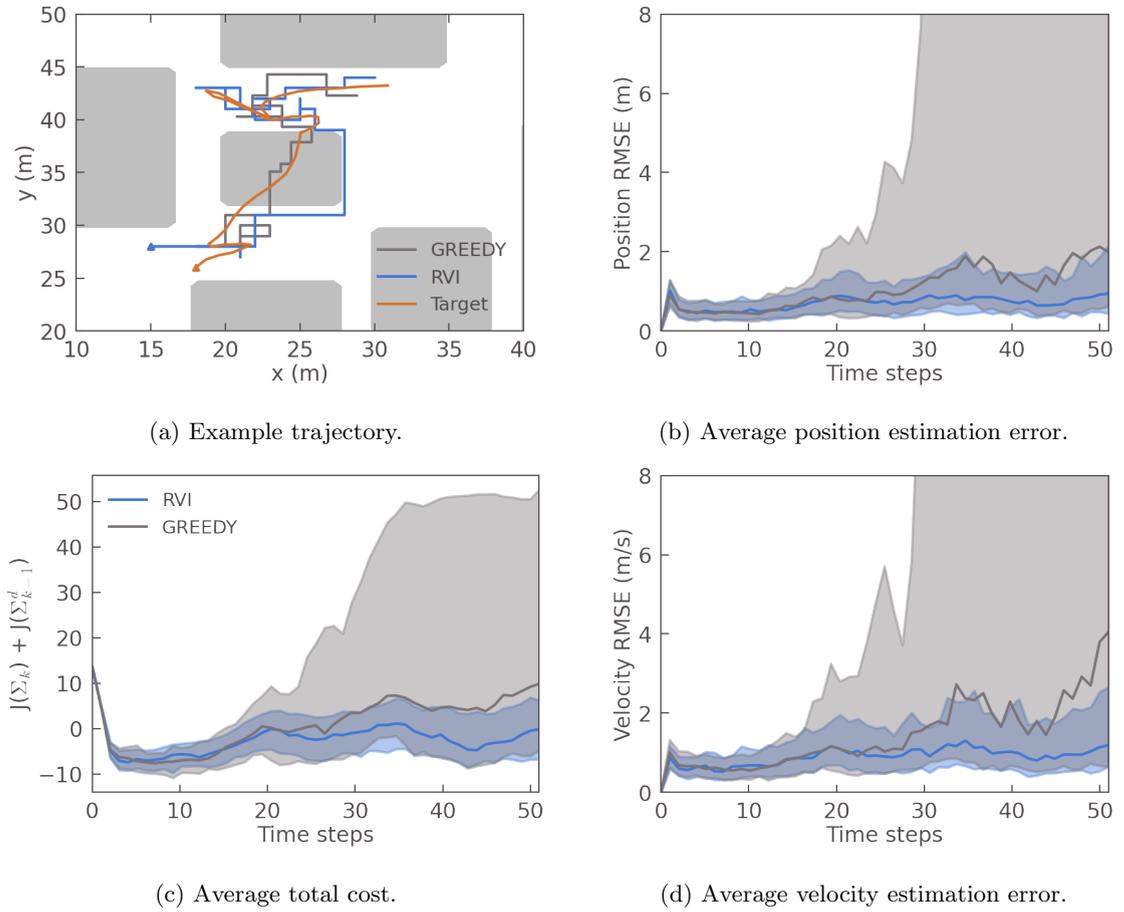


FIGURE 4.1: Results for the active target tracking problem solved via latent belief space planning. Results are averaged over 150 Monte Carlo simulations. The left panel shows an example trajectory (initial robot positions marked by triangles) and the median cost with interquartile range of each policy’s calculated trajectory. The right panel shows the median RMSE and interquartile range of each policy’s target position and velocity estimates.

The sensor takes noisy position measurements of the target and uses them to obtain the target’s velocity by differentiation. For simplicity, the sensor observation model in Equation (4.2) is given by $C_t = I_4$ with the measurement noise increasing linearly with the distance between robot and target. To introduce non-myopia, certain areas of the environment are ‘cloudy’, depicted as grey areas in Figure 4.1(a), and increase the robot’s measurement noise. Upon entering a cloud, the robot should slow down drastically to maintain safety under poor visibility. Beyond a maximum range of 20 metres the measurement noise is effectively infinite.

For 150 Monte-Carlo simulations, Algorithm 1 with implementation detailed as in Section 4.1.3 is used to track the target with $\tau = 1$, $T = 5$, $q = 0.1$, $\epsilon = 0.1$ and $\delta = 1$. The performance of our proposed algorithm is compared to a greedy approach in Figure 4.1(a). The RVI algorithm's long planning horizon successfully predicts the target will enter and remain in a cloudy area in future time steps and thus prioritises taking measurements from outside this area at an increased distance rather than staying close to the target. On the contrary, the greedy algorithm prioritises minimising the cost function at each time step and therefore lacks the foresight to avoid this cloudy area, following the target in. This behaviour often leads the greedy robot to 'lose' the target, unable to re-locate it as it exits a cloud or makes an unknown maneuver, as is evident by the large interquartile range seen for the greedy policy in Figures 4.1(c) to 4.1(d).

The joint estimation cost in Equation (4.12) for the original problem is shown in Figure 4.1(c), elucidating the impact of RVI's non-myopic planning on the performance of the found solution. We see RVI incurs lower total cost with much lower standard deviation than the greedy policy, indicating that planning over the state belief space via our RVI extension is indeed an appropriate reduction of the original problem.

Additionally, comparison of the average root mean square error (RMSE) of each policies' full state estimate (Figures 4.1(b) and 4.1(d)) shows for all time steps, our algorithm tracks target evolution in the presence of unknown inputs more successfully. These results demonstrate that though the planned sensing actions do not directly consider the unknown inputs, our proposed approach is still able to plan non-myopically in their presence, a promising confirmation of the implication of our theoretical results.

4.5 Summary

In this chapter we proposed reduced latent belief space planning for active estimation of linear Gaussian systems subject to arbitrary, unknown disturbances. For such systems, a proper estimate of the full description requires solving a joint estimation problem to recover both the state and the unknown input. However, we reduced this joint estimation problem to one that considers state estimation only. By doing so we facilitated belief

space planning over a smaller, deterministic belief space, easing the computation required to solve the active estimation problem. Further, we demonstrated theoretically that this reduction has minimal impact on recovering the original full system description through a complete characterisation of suboptimality bounds associated with the final estimates acquired. The theoretical results were supported by simulation results, where a sensing robot was tasked with tracking a target performing unknown evasive manoeuvres in non-myopic conditions.

Here, the latent variable $\boldsymbol{\xi}_t = \mathbf{y}_t$ used to infer the full state $\mathbf{y}_t^S = [\mathbf{y}_t, \mathbf{d}_t]^\top$ was an element of the full state itself. The following chapters will extend the application of latent belief space planning to active perception problems where $\boldsymbol{\xi}_t$ is instead a variable explicitly introduced to enable reduced belief space planning.

Chapter 5

The homotopic belief space for active target tracking

In this chapter we present the homotopic belief space for active target tracking, building upon the experimental application studied in the previous chapter. Here, however, the target’s dynamics are learned from historical data, avoiding the difficulties of modelling altogether. With a learned dynamic model one loses access to the neat properties of the Kalman or unknown input filter covariance update maps that facilitate deterministic belief space planning. We present a solution to the challenges this introduces in two parts: first, a hierarchical learning architecture for learning predictive dynamic models from historical data and second, the homotopic belief space and corresponding planning algorithm to plan sensing trajectories.

When solving the trajectory prediction problem in cluttered environments, there are an infinite number of possible trajectories to consider. To simplify the space of trajectories under consideration, we utilise *homotopy classes* to partition the space into countably many equivalent classes. All members within a class demonstrate identical high-level motion with respect to the environment, i.e., travelling above or below an obstacle. This allows high-level prediction of a trajectory via the prediction of a label identifying its homotopy class, referred to as a *h-signature*. Then, by training a low-level predictive

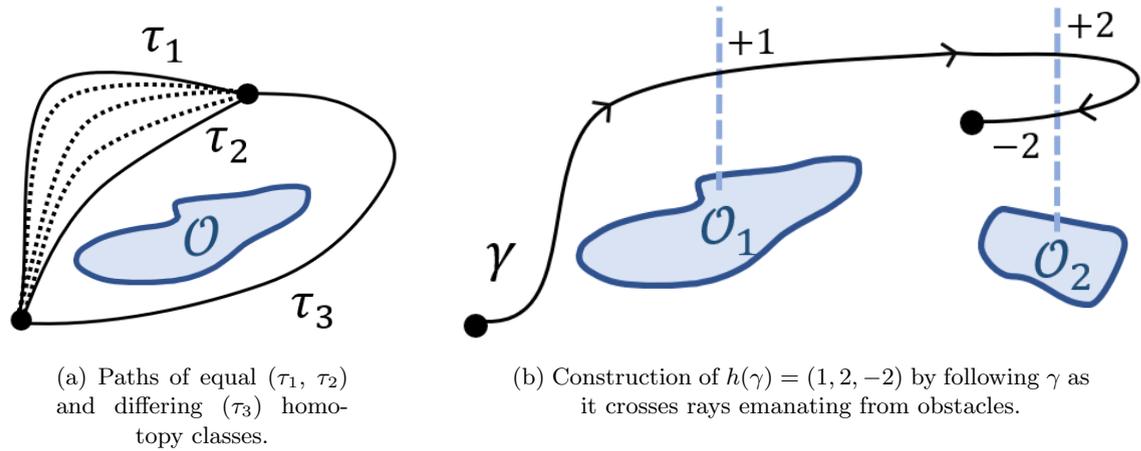
model for each homotopy class the full trajectory can be inferred from the predicted h -signature.

The h -signature or homotopy class of a trajectory is thus the latent variable ξ used to solve reduced Problem 1.2 for active target tracking. The homotopy class prediction module provides a full probability mass function (PMF) over possible future high-level motions given the observed partial trajectory. We refer to these PMFs as *homotopic beliefs* and utilise them to perform belief space planning over homotopy classes. Since the homotopy class of a target is determined by how it manoeuvres around obstacles, information in this belief space is as sparsely distributed as obstacles. Moreover, h -signatures are discrete random variables, meaning the proposed belief space is discrete rather than continuous. These are significant advantages over the full belief space associated with the target's position. Planning over this sparse, discrete latent belief space together with a hierarchical predictive model allows for more efficient active target tracking that is still reliable, opening avenues to overcome resource constraints in multi-target problems in future work.

This chapter builds upon our previous work [169] where we introduced the topological predictive model for trajectory prediction. We present our extension to this publication, closing the perception loop and introducing homotopic belief space planning for active target tracking. At the time of thesis submission, this extension was yet to be published.

5.1 Homotopy theory in robotics

First we introduce the concept of homotopy classes for summarising the high-level motions of a moving entity. Two paths τ_1, τ_2 in a topological space \mathcal{D} with common start and ending points are homotopic if there exists a continuous transformation or deformation from one to the other [170]. Sets of paths homotopic to one another are named homotopy classes. Non-trivial homotopy classes arise as a result of obstacles in the space, as deforming some paths into others would require moving through an obstacle and breaking the continuity requirement. This notion is depicted in Figure 5.1(a). Paths τ_1 and τ_2 are homotopic as they can be continuously deformed into each other, indicated by the dotted paths between

FIGURE 5.1: Illustrations of homotopy classes and h -signature calculation.

them. These paths are however not homotopic to τ_3 , as any deformation into τ_3 would require moving through the obstacle \mathcal{O} .

In the context of robotics, obstacles in the environment split the space of an agent's possible trajectories between points from a single homotopy class into a countable number of homotopy classes. The obstacles therefore dictate the number of unique ways an agent may travel through the space from one point to another. For example, in Figure 5.1(a) the obstacle \mathcal{O} splits the space of trajectories into three broad categories: those that move 'above' the obstacle, those that move 'below', and those that wind around the obstacle any number of times before moving to the end point. Such abstraction of the high-level motions available to an agent is a powerful tool for navigation, prediction and tracking tasks often encountered in robotics [171–175].

5.1.1 The h -signature as a homotopy invariant

Attempting to categorise trajectories in a dataset according to homotopy class can become cumbersome if one considers the formal definition of existence of a continuous transformation between paths. A pairwise comparison of all paths in the dataset would be required. Thankfully, unique identifiers for homotopy classes that avoid this have been proposed. These identifiers are referred to as homotopy invariants because their value does not change for any path within a single homotopy class.

The homotopy invariant used in this work is the h -signature [170, 176]. The h -signature is a unique identifier $h(\tau)$ of a trajectory’s homotopy class such that $h(\tau_1) = h(\tau_2)$ if and only if τ_1 and τ_2 are homotopic. To compute the h -signature of a path, obstacles are enumerated from 1 to N , with N being the number of obstacles in the environment. Non-intersecting rays are drawn from within each obstacle to the boundary of the environment. Then a ‘word’ is constructed by following the path and appending letter ‘ n ’ if the path crosses the ray corresponding to the n -th obstacle from left to right, and ‘ $-n$ ’ if it crosses from right to left. The construction of $h(\gamma)$ is given in Figure 5.1(b) to illustrate. The path γ crosses obstacle ray 1 and 2 from left to right, then returns to cross obstacle ray 2 again from right to left. $h(\gamma)$ is therefore $(1, 2, -2)$. The final h -signature may then be reduced by cancelling all consecutive appearances of n and $-n$. Then, $h(\gamma)$ in Figure 5.1(b) reduces from $(1, 2, -2)$ to (1) .

5.2 Homotopy-informed trajectory prediction

In this section we develop our hierarchical predictive model for target motion. This model facilitates inference of the full target trajectory via the h -signature latent variable $\xi = h$, and is thus a key ingredient for latent belief space planning.

5.2.1 Problem formulation

Consider an agent traversing through a planar environment \mathcal{D} containing N obstacles $O = \{\mathcal{O}_1, \dots, \mathcal{O}_N\}$. The agent’s trajectory is denoted $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_T\}$. We assume the agent begins at a start location on the boundary $\delta\mathcal{D}$ of the environment and travels to an end location on $\delta\mathcal{D}$. We are given a dataset of K historical, fully observed trajectories $\mathcal{Y} = \{Y_{\text{obs}}^1, \dots, Y_{\text{obs}}^K\}$ and full knowledge of obstacles. It is assumed that the historical data is representative of future or unseen trajectories. That is, the training and test data are drawn from the same distribution. Further, in the online setting, we have a sequence of partial noisy measurements of a test trajectory $\mathbf{z}_{1:t} = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$ up to time t , with a known observation model $P(\mathbf{z}_t | \mathbf{y}_t)$.

We are then interested in predicting the agent’s high-level motion through the environment. There is no well-established methodology for abstracting an agent’s high-level motion from its trajectory. We propose the best abstraction is given by a trajectory’s homotopy class and thus prediction of a trajectory’s h -signature is the most appropriate approach to predicting high-level motion. Formally, the problem is:

Problem 5.1. (*High-level prediction*) *Given the partial, noisy measurements of the trajectory $\mathbf{z}_{1:t}$, predict the h -signature h of the robot’s future full trajectory.*

While the solution to Problem 5.1 can be used as a tool in many robotics problems, here we are interested in how the h -signature may be used to produce a topology-informed low-level prediction for a trajectory:

Problem 5.2. (*Low-level prediction*) *Given the predicted h -signature h associated with an agent’s partially observed measurements $\mathbf{z}_{1:t}$, predict the full trajectory Y .*

Problems 5.1 and 5.2 are challenging to solve directly without modification. In a pedantic Bayesian formulation one would first predict the underlying trajectory for the entire duration and subsequently extract a prediction of the corresponding h -signature. This is because the measurements are conditionally independent of the h -signature given the trajectory and the only given relationship between the low-level trajectory and the h -signature is the computation process outlined in Section 5.1.1. In other words, in this view, low-level trajectory prediction precedes high-level prediction, limiting its effectiveness.

We circumvent low-level prediction by introducing the notion of the partial h -signature. Unlike the usual h -signature, partial h -signatures can be obtained from an incomplete trajectory, as we detail in Section 5.2.2. Given the partial h -signature we predict the full h -signature. This is achieved with a variable order Markov process (VOMP) model trained on the collection of h -signatures observed in the historical dataset (Section 5.2.3). The full h -signature can then be used to predict the low-level trajectory with a model trained on the historical dataset. As we readily have access to trajectories and their associated h -signature, this can be as simple as learning a mixture of experts for low-level trajectory prediction within each homotopy class. To this end, we demonstrate the use of a hierarchical Gaussian mixture model in Section 5.2.4.

5.2.2 Partial h -signatures

As described in Section 5.1, homotopy classes exist only for paths between two fixed start and end points. In the context of this work trajectories are assumed to start and end on boundary points of the environment. Similar to [175], we ensure our description of homotopy classes is valid by applying a quotient map, mapping all boundary points to a single quotient point while preserving the topology of the space¹. Then, the set of all homotopy classes here is over paths between the quotient point and itself.

With this in mind, the notion of a *partial h -signature* is ill-defined in a topological sense. Much like a full h -signature, it is calculated by constructing a ‘word’ according to a path’s ray crossings. However, it is extracted from an incomplete trajectory, i.e. one that has not yet returned to the quotient point. It is thus crucial to note the partial h -signature is not introduced as an identifier of a homotopy class but rather as a *predictor*.

To predict full h -signatures from partial ones, a notion of compatibility between the two is needed. For a given partial h -signature p , compatible full h -signatures are those whose prefix is p . In other words, the set of all h -signatures compatible with p is $\mathcal{H}(p) = \{h \mid \exists p', h = pp'\}$.

5.2.3 High-level prediction with variable order Markov processes

As h -signatures in an environment with N obstacles are simply ‘words’ constructed from an alphabet $\mathcal{A} = \{1, \dots, N, -1, \dots, -N\}$, prediction of a full h -signature given partial h -signature can be viewed as a sequence completion problem. The sparse nature of the h -signature as a representation of high-level motion allows for relatively simple techniques to be used for sequence generation. We propose a VOMP [177] for this purpose. VOMPs are compact probabilistic automata reminiscent of Weighted Finite Automata (WFA), capable of producing a probabilistic predictions for sequence completion. However, VOMPs are designed to learn varying length dependencies in data – an advantage over WFA in situations where sequence context can improve prediction. Compared to LSTMs that share

¹In this process we consider all boundary points of the domain to be equivalent. Then, applying the quotient map can be imagined as ‘gluing’ the boundary points to one another, deforming the planar domain into the surface of a sphere in the process.

this capability, VOMPs are capable of efficiently producing probability distributions over possible outcomes. This is crucial for our approach, as a probability distribution over full h -signatures is required to produce probabilistic low-level predictions.

We represent a VOMP with a probabilistic suffix automaton (PSA) as in [177]. Here, a PSA state is a partial h -signature p of up to some maximum length $L > 0$ constructed from alphabet \mathcal{A} as $p = (a_1, \dots, a_l)$, $0 \leq l \leq L$. Transitions between two states p and p' are allowed only if there exists some $a \in \mathcal{A}$ such that p' is a *suffix* of ap . Allowed transitions have associated with them a probability that the transition will occur. Transition probabilities learned offline can be used to produce probability distributions over future states online. An example PSA is drawn in Figure 5.2(b).

We follow [177] closely to train a VOMP over h -signatures offline, and adapt their online prediction process to better suit the prediction of h -signatures. Our algorithms for online prediction and offline learning are detailed below.

Online Prediction

The trained VOMP outputs probabilities for arbitrarily long h -signatures. To find the probability of any h -signature $P(h)$ occurring, one simply takes a walk through the PSA, multiplying transition probabilities from state to state. However, we are interested in finding the conditional probabilities $P(h | p)$ for all possible h -signatures given the partial h -signature p . That is, we are interested in calculating the *homotopic belief*. Letting $\mathcal{H}(p)$ be the set of all full h -signatures compatible with p , the conditional probability $P(h | p)$ of any outcome $h \in \mathcal{H}(p)$ can be calculated by normalising the VOMP output $P(h)$ with the sum of all other possible h -signature outcomes $h' \in \mathcal{H}(p)$,

$$P(h | p) = \frac{P(h)}{\sum_{h' \in \mathcal{H}(p)} P(h')}. \quad (5.1)$$

However, the set $\mathcal{H}(p)$ of compatible h -signatures is of infinite size. Just as agents may walk paths of varying lengths through an environment, the lengths of compatible full h -signatures varies. For example, if an agent is observed passing from left to right above the first obstacle in an environment their partial h -signature is (1). Later, the agent may

Algorithm 3 PSA offline learning**Inputs:** $\epsilon, L, \mathcal{A}, h$ -signature data**Output:** Trained PSA

```

1: initialise tree  $\mathcal{T}$ 
2:  $\mathcal{P} \leftarrow \{a \mid a \in \mathcal{A}, P(a) \geq \epsilon\}$ 
3: while  $\mathcal{P}$  not empty do:
4:    $p \leftarrow \mathcal{P}.\text{pop}()$ 
5:   if  $\mathcal{E}(p, \text{suffix}(p)) \geq \epsilon$  then:
6:     add path to  $p$  to  $\mathcal{T}$ 
7:   end if
8:   if  $|p| \leq L$  then:
9:      $\mathcal{P} \leftarrow \mathcal{P} \cup \{ap \mid a \in \mathcal{A}, P(ap) \geq \epsilon\}$ 
10:  end if
11: end while
12: for all leaves  $r$  in  $\mathcal{T}$  do:
13:   if longest prefix( $r$ ) not in  $\mathcal{T}$  then:
14:     add path to  $r$  to  $\mathcal{T}$ 
15:   end if
16: end for
17: PSA  $\leftarrow$  leaves of  $\mathcal{T}$ 

```

or may not pass above any other obstacle. Their full h -signature may be (1) or (1, ...) depending on this.

To handle these nuances, we assume the longest possible h -signature an agent will take through the environment is the maximum length h -signature present in training data, h^* . Then, for any $h \in \mathcal{H}(p)$, $P(h) = 0$ if $|h| > |h^*|$. We then weigh h -signature probabilities in Equation (5.1) by the probability of observing any h -signature of that length in data. Thus, the VOMP is queried to make the adjusted calculation

$$P(h \mid p) = \frac{P(h) \cdot P(|h|)}{\sum_{h' \in \mathcal{H}(p)} P(h') \cdot P(|h'|)}. \quad (5.2)$$

Offline Learning

To facilitate online belief calculation given partial h -signatures, a prediction suffix tree (PST) (Figure 5.2(a)) is built over the alphabet \mathcal{A} . The PST is then converted to a PSA (Figure 5.2(b)) as per [177]. A high-level overview of this process is provided in Algorithm 3.

To construct the PST, it is first initialised with root node corresponding to the ‘empty’ h -signature labelled (). Paths to suffixes are then successively added to the tree if the suffix has sufficiently strong predictive power. Specifically, a child node labelled with partial h -signature ap is added to parent node p if some measure of statistical difference \mathcal{E} between the PMFs $P(h | p)$ and $P(h | ap)$ is above a user-defined threshold ϵ . The metric used is the KL divergence scaled by the probability of observing ap ,

$$\mathcal{E}(ap, p) = P(ap)D_{\text{KL}}(P(h | ap), P(h | p)). \quad (5.3)$$

The scaling factor $P(ap)$ serves to avoid the addition of suffixes with low probability of occurrence, yet high KL divergence. Suffixes up to length L are tested for their predictive power and added.

To calculate $P(a | p)$ and $P(p)$ from training data, Laplace’s rule of succession is used. Denoting $\mathcal{F}(p)$ as the frequency with which partial h -signature p appears in data, $\mathcal{F}(p^C)$ is then used to denote the frequency of the complement event; the frequency of observing any other partial h -signature of length $|p|$. Then,

$$P(p) \approx \frac{\mathcal{F}(p) + 1}{\mathcal{F}(p^C) + |\mathcal{A}|}. \quad (5.4)$$

Similarly, denoting the frequency with which letter a follows p in observations by $\mathcal{F}(a | p)$ and the frequency of observing any other letter after p by $\mathcal{F}(a^C | p)$,

$$P(a | p) \approx \frac{\mathcal{F}(a | p) + 1}{\mathcal{F}(a^C | p) + |\mathcal{A}|}. \quad (5.5)$$

After construction, taking the leaves of the PST gives the states of the corresponding PSA. However, this simple action may not always admit a valid transition between all states. In this case, leaves must be added to the PST to ensure a complete PSA. Nodes are added to the PST until, for every leaf in the PST, the longest prefix of the leaf exists in the PST. When this condition is true, the leaves of the PST are guaranteed to give a complete PSA [177]. Transition probabilities between states of the PSA correspond directly to the

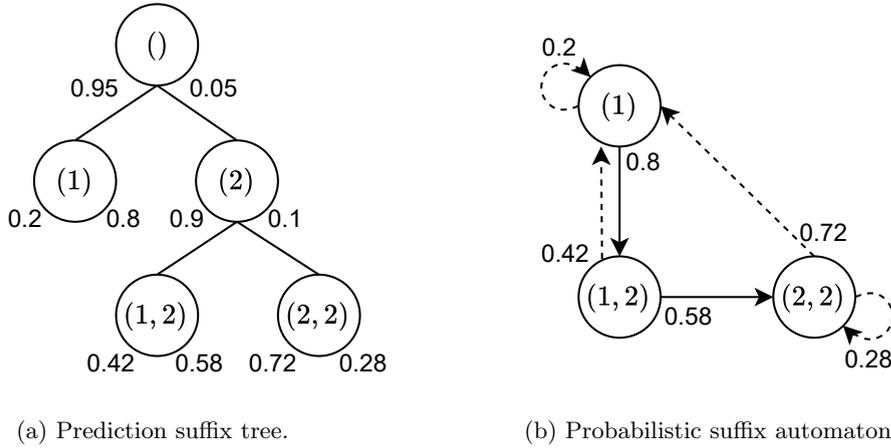


FIGURE 5.2: Equivalent methods of representing a VOMP. Values along edges between states are transition probabilities.

transition probabilities of the PST found via Equation (5.5) and are used to calculate probabilities online.

Illustration of homotopic belief updates

To illustrate the topological VOMP's online belief update we depict the process on a toy example. A simplistic environment and dataset was created, shown in Figure 5.3(a). Trajectory data was created by running Dijkstra's algorithm on a graph over the environment to find shortest-distance paths from randomly selected points on the left boundary to random points on all other boundaries. Thus, in this simple dataset all trajectories move from left to right, and the set of possible homotopy classes includes h -signatures $\{(), (1), (1, 2)\}$.

Prediction of h -signatures over time is shown in Figures 5.3(b) to 5.3(d). At time $t = 8$, the partial h -signature is $p = ()$, and the VOMP predicts that all h -signature trajectories are possible in the future, with probabilities $P(h = () \mid p) = 0.28$, $P(h = (1) \mid p) = 0.37$, $P(h = (1, 2) \mid p) = 0.35$. At $t = 17$ the $()$ class is predicted with probability 0 now the partial h -signature is $p = (1)$, and $P(h = (1) \mid p) = 0.52$, $P(h = (1, 2) \mid p) = 0.48$. In the final time step the VOMP correctly assesses that the trajectory will have full h -signature $(1, 2)$ with probability 1 as the trajectory has passed the second obstacle. Coloured regions correspond to regions in which one may expect the agent to be in the

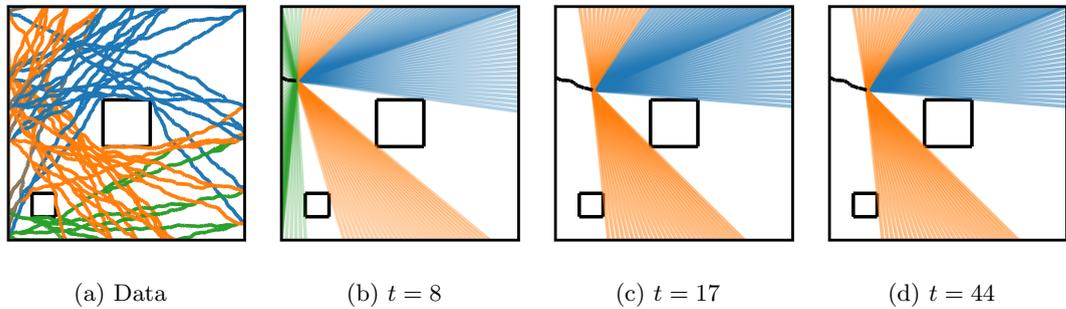


FIGURE 5.3: VOMP h -signature prediction for a test trajectory over time. Black trajectory is the observed test trajectory. Coloured regions correspond to straight-line path completions that lie within predicted homotopy classes. Colour coding of homotopy classes is as in the training set. Alpha of the regions reflects the VOMP output probability for that class.

future, with probabilities indicated by opacity. These regions demonstrate the predictive power of the VOMP output.

5.2.4 Low-level prediction with hierarchical Gaussian mixture models

We present a hierarchical GMM as an example implementation of the low-level prediction algorithm for retrieving a probability distribution $P(Y | h)$ over the *full* trajectory $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_T\}$ given the h -signature h . To this end, we simply cluster the trajectories from the training dataset into their homotopy classes (i.e. those that have the same h -signature h), and fit a GMM with N_C components for each class,

$$P(Y | h) = \sum_c w^{(c,h)} \mathcal{N}(\boldsymbol{\mu}^{(c,h)}, \Sigma^{(c,h)}). \quad (5.6)$$

Here, for a given homotopy class h , $w^{(c,h)}$, $\boldsymbol{\mu}^{(c,h)} \in \mathbb{R}^{2T}$ and $\Sigma^{(c,h)} \in \mathbb{R}^{2T \times 2T}$ are the weight, mean vector and covariance respectively of component $c \in \{1, \dots, N_C\}$. Each component is a multivariate normal distribution over Y . While a single Gaussian per homotopy class can be used, it may not always be sufficient. Within high-level motions there may exist multi-modal behaviour. For example, one can take a sharp or a wide left turn at an intersection. In this case, choosing $N_C = 2$ can capture both these low-level motions within the same homotopy class.

With this representation, the low-level patterns in the trajectories can be captured by computing the full covariance matrix Σ , which captures correlations between positions at particular times. Further, this weighted sum of multivariate normal distributions is tractable and we can derive a fully probabilistic prediction of the trajectory given the measurements $\mathbf{z}_{1:t}$ and corresponding partial h -signature p_t .

To derive this prediction, we treat the the partial h -signature as a random variable that is observable and conditionally independent of Y given the full h -signature h . Then, we can easily condition Equation (5.6) on the partial h -signature as a weighted sum

$$P(Y | p_t) = \sum_h P(Y | h)P(h | p_t). \quad (5.7)$$

Thus, this independence assumption facilitates incorporation of topological information in low-level prediction by simply scaling the weights of the GMM by the homotopic belief $P(h | p_t)$ maintained by the VOMP. Subsequently, we can further condition on the actual low-level measurements $\mathbf{z}_{1:t}$ via

$$P(Y | p_t, \mathbf{z}_{1:t}) = \sum_{c,h} \hat{w}_t^{(c,h)} \mathcal{N}(\hat{\boldsymbol{\mu}}_t^{(c,h)}, \hat{\Sigma}_t^{(c,h)}), \quad (5.8)$$

where hats ($\hat{\cdot}$) signify the *posterior* statistics of the GMM after conditioning on measurements. Statistics without hats indicate the prior statistics. The conditional mean and covariance $\hat{\boldsymbol{\mu}}_t^{(c,h)}$, $\hat{\Sigma}_t^{(c,h)}$ are calculated in the same manner as standard conditional Gaussian distribution [178, Section 8.1.3] given measurements. The conditional weights are calculated as

$$\hat{w}_t^{(c,h)} \propto w_t^{(c,h)} P(h | p_t) \mathcal{N}(\boldsymbol{\mu}^{(c,h)}, \Sigma^{(c,h)} + \sigma_Z^2 I), \quad (5.9)$$

followed by normalisation. The last term is the marginal likelihood of observing $\mathbf{z}_{1:t}$ within each mixture component, where σ_Z is the measurement noise covariance.

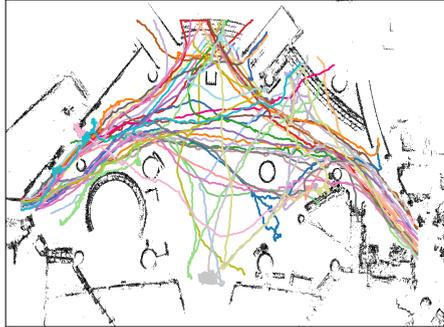


FIGURE 5.4: The ATC shopping mall environment and a subset of the training data.

5.2.5 Evaluation of predictive model

Experiment setup

In this section the efficacy of full trajectory inference via homotopy class as a latent variable is validated experimentally by comparison against a standard GMM without topological knowledge. We evaluate each method on the ATC shopping mall dataset [179] shown in Figure 5.4. From the dataset we retrieved 17558 pedestrian trajectories. From this, 9230 trajectories that satisfied the border crossing assumption were selected. Among these trajectories, we randomly selected 5538 trajectories to create a training dataset and 1000 trajectories for testing. All trajectories are interpolated over 100 timesteps. Thus, the predictive means $\hat{\boldsymbol{\mu}}_t^{(c,h)}$ of the GMMs are 100×2 arrays flattened to 200×1 vectors with corresponding total covariance $\hat{\Sigma}_t^{(c,h)} \in \mathbb{R}^{200 \times 200}$. For fairness, we use the same number of mixtures for the naive GMM as in the topology-informed GMM. All GMMs used are trained using the implementation in the SCIPY library, using the default arguments.

Three metrics are used for comparison. Most immediately, we consider the standard average displacement error (ADE), defined as the displacement of the highest weighted mean from the ground truth, averaged over all $T = 100$ time steps,

$$\text{ADE} = \frac{1}{T} \sum_k^T \|\mathbf{y}_k - \hat{\boldsymbol{\mu}}_{t,k}^{(c^*,h^*)}\|, \quad (5.10)$$

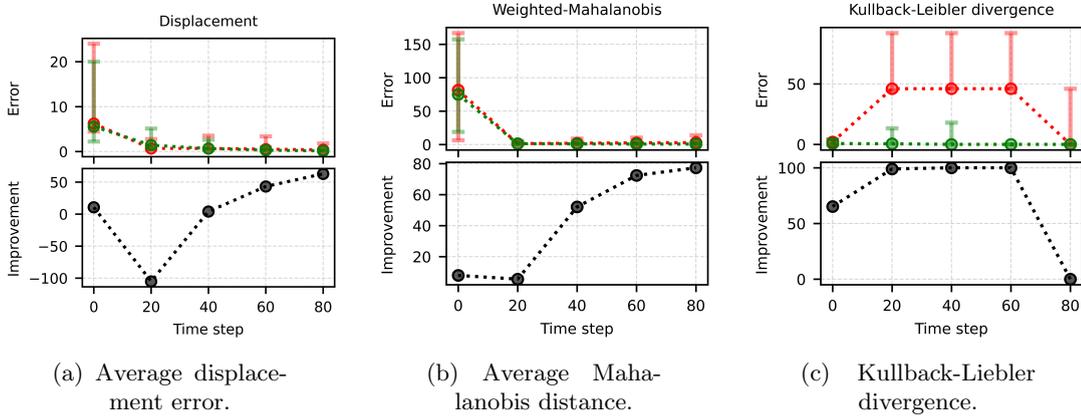


FIGURE 5.5: Results and improvement in median of our proposed topological GMM over a naive GMM in each metric. Green: Our approach. Red: Naive GMM. Markers show the median. Error bars show 25% and 75% quantiles.

where $\hat{\boldsymbol{\mu}}_{t,k}^{(c^*,h^*)}$ is the posterior mean of the highest weighted component at the k -th time step.

Since the framework is fully probabilistic, we also need to account for the uncertainty estimates produced. We then consider the average Mahalanobis distance (AMD), which accounts for the weights and covariances in the GMM:

$$\text{AMD} = \sum_{c,h} \frac{\hat{w}_{\text{obs}}^{(c,h)}}{T} \sum_k \Theta_k^{(c,h)\top} (\hat{\Sigma}_{t,k}^{(c,h)})^{-1} \Theta_k^{(c,h)}, \quad (5.11)$$

where $\Theta_k^{(c,h)} = (\mathbf{y}_k - \hat{\boldsymbol{\mu}}_{t,k}^{(c,h)})$ and $\hat{\Sigma}_{t,k}^{(c,h)}$ is the k -th block diagonal of $\hat{\Sigma}_t^{(c,h)}$.

Lastly, we would like to characterise the performance of the high-level prediction made. To do so, we compare the KL divergence (KLD) between the weights of two posterior GMMs, one conditioned on the full trajectory acting as a ground truth distribution and the other conditioned on $\mathbf{z}_{1:t}$. That is,

$$\text{KLD} = \sum_{c,h} \hat{w}_T^{(c,h)} (\log \hat{w}_T^{(c,h)} - \log \hat{w}_t^{(c,h)}). \quad (5.12)$$

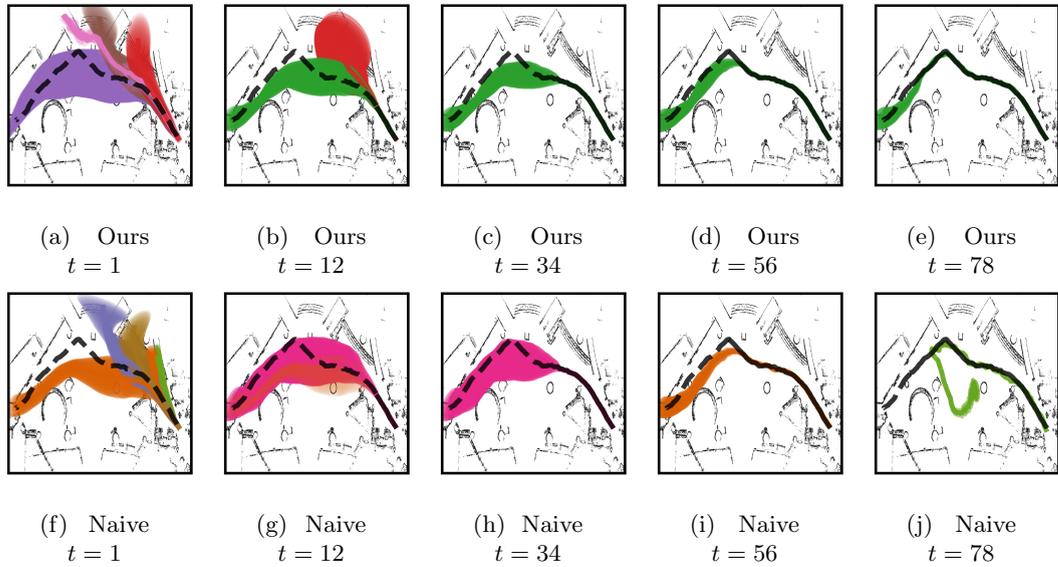


FIGURE 5.6: Snapshots of predictions output by a GMM with h -signature context given by the VOMP (a) - (e) and without (f) - (j). Black solid path is the elapsed ground truth test trajectory at the given time. Dotted path indicates future ground truth trajectory remaining. Shaded regions show the variance of around the mean trajectory in solid colour. Transparency is proportional to weight. Colours in (a) - (e) indicate h -signatures, while colours in (f) - (j) indicate mixture components.

Results

Figure 5.5 shows the results of comparisons between the naive GMM (red) and our topology-informed approach (green) in terms of ADE (left), AMD (middle), and KLD (right). In terms of ADE, our approach outperforms the naive approach after some time, namely showing a 50% improvement in the median just after the halfway point of the trajectory ($t = 60$), and up to 69.4% over all time. This increase in performance is because the topology-informed approach gains more information as the target crosses obstacles, and this topological information improves the predictive power of the VOMP.

Since the naive GMM was given the same number of components as the topology-informed GMM – calculated based on the number of homotopy classes – it is unsurprising that the naive GMM performs relatively well. In practice, a major challenge in deploying GMMs is selecting the right number of mixture components. In fact, initialising the naive GMM with a number of components directly informed by the number of homotopy classes present in the data provides a small topological cue to the baseline method.

Nonetheless, the AMD metric shows our approach performs significantly better than the naive approach. Specifically, we see a 72.3% improvement over the naive halfway through the trajectory, with a maximum of 80.3% improvement over time. This implies the uncertainty predictions from the topology-informed approach are more *consistent* with the actual error compared to the naive approach. This is because our approach pre-clusters trajectories that are ‘similar’, improving the fitness of the GMM model within each cluster.

Further, our approach consistently outperforms in terms of KLD by a great margin, with 100% improvement achieved by $t = 20$. The naive GMM weights do not converge to the ground truth weights until $t = 80$. This means the early posterior GMM weights of our approach better represented the final GMM weights. Viewing GMM components as an alternative description of high-level motion to h -signatures, this implies h -signatures are indeed better descriptors.

Figure 5.6 shows the behaviour of our approach and the baseline GMM. Early on, we see both methods give highest weight to components that reasonably predict the general motion of the trajectory. However, the naive approach begins to produce poor predictions in an area of the environment where many training paths diverge. This ‘crossroads’ is visible in the sample data plotted in Figure 5.4. Additionally, we see the naive GMM re-weights the highest probability component often, in-line with its performance in the KLD metric. In comparison, our approach continues to correctly predict the same h -signature across time, even when reaching the aforementioned crossroads. This demonstrates the power of topological information in imbuing low-level predictions with robustness to deviations in trajectories.

5.3 Homotopic belief space planning

Evaluation of the hierarchical predictive model in the previous section confirms the h -signature is a suitable latent variable for solving Problem 1.2. We therefore continue on to close the active target tracking loop, re-introducing the sensing robot whose task is to collect measurements online and improve its estimate of the target’s full trajectory $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_T\}$.

5.3.1 Belief updates in the online setting

The homotopic and low-level belief updates presented in Equations (5.2) and (5.8) respectively were introduced for a scenario without a sensing robot actively searching for the target. Now, in closing the perception loop and introducing a sensing robot, there is an added uncertainty beyond measurement noise: whether or not the target will be detected at \mathbf{x}_t and a measurement will be received.

To adjust to this online scenario we adjust the low-level belief update in Equation (5.8) in a similar vein to the GM-PHD filter proposed in [135]. However, we have the following simplifying assumptions. Firstly, data association is not a concern. We have knowledge that the source of each measurement is the target itself. Secondly, we do not assume targets spawn or despawn during the tracking scenario. The main implications of these assumptions are that we do not need to consider clutter, and we can run many motion hypotheses in parallel for our target, updating the beliefs of each Gaussian component in the GMM independently.

In our online belief updates, we need only consider the probability of successful detection $P(D = \text{True} \mid (c, h), \mathbf{x}_t)$ given sensing location \mathbf{x}_t for each hypothesis indexed by (c, h) . To this end, we introduce a Bernoulli random field, where at each $\mathbf{x} \in \mathcal{D}$, probability of detection is modelled by Bernoulli random variable $D \sim \text{Bern}(\gamma^{(c,h)}(\mathbf{x}))$ with success parameter $\gamma^{(c,h)}(\mathbf{x}) = P(D = \text{True} \mid (c, h), \mathbf{x})$. The belief update is scaled by the two detection outcomes, success or failure. That is, Equation (5.8) becomes

$$P(Y \mid p_t, \mathbf{z}_{1:t}, \mathbf{x}_t) = \sum_{c,h} \gamma^{(c,h)}(\mathbf{x}_t) \hat{w}_t^{(c,h)} \mathcal{N}(\hat{\boldsymbol{\mu}}_t^{(c,h)}, \hat{\boldsymbol{\Sigma}}_t^{(c,h)}) \quad (5.13)$$

if a target is detected. Otherwise,

$$P(Y \mid p_t, \mathbf{z}_{1:t}, \mathbf{x}_t) = \sum_{c,h} (1 - \gamma^{(c,h)}(\mathbf{x}_t)) \hat{w}_{t-1}^{(c,h)} \mathcal{N}(\hat{\boldsymbol{\mu}}_{t-1}^{(c,h)}, \hat{\boldsymbol{\Sigma}}_{t-1}^{(c,h)}). \quad (5.14)$$

Since $P(Y \mid p_t, \mathbf{z}_{1:t}, \mathbf{x}_t)$ is a distribution over the full trajectory, these belief updates provide future trajectory predictions based on the observations $\mathbf{z}_{1:t}$.

To model $\gamma^{(c,h)}(\mathbf{x}_t)$ we introduce a probabilistic detection model that decays exponentially as the distance between the robot and the target increases

$$P(D = \text{True} \mid \mathbf{x}_t, \mathbf{y}_t) = A \exp\left(-\frac{\|\mathbf{x}_t - \mathbf{y}_t\|^2}{2r^2}\right), \quad (5.15)$$

where r is the sensing radius of the robot and A controls the peak of the model. However, since the target's state \mathbf{y}_t is unknown, marginalisation using the target motion model provided by the GMM is necessary to obtain a tractable result,

$$\gamma^{(c,h)}(\mathbf{x}_t) = \beta \exp\left(-\frac{1}{2}(\mathbf{x}_t - \hat{\boldsymbol{\mu}}_t^{(c,h)})^\top (r^2 I + \hat{\Sigma}_t^{(c,h)})^{-1} (\mathbf{x}_t - \hat{\boldsymbol{\mu}}_t^{(c,h)})\right), \quad (5.16)$$

where $\beta = \frac{A}{\sqrt{\det(\hat{\Sigma}_t^{(c,h)} r^{-2} + I)}}$ is a normalisation factor.

5.3.2 Homotopic information gain

In a conventional approach, the expected information gain at sensing locations can be calculated via information-theoretic measures such as entropy and mutual information over the expected measurement GMM. However, there is no analytical form for the entropy of a GMM [18] and so approximations must be made. There are many proposed approximations [180, 181]. In [181] for example, bounds for mixture entropy are given via calculation of pairwise distances between components. For large GMMs with many components a pairwise calculation can become computationally expensive.

By planning over the belief space associated with the latent variable h we avoid all such computational issues. Then we solve Problem 1.2 where $\xi = h$ as a proxy for producing an accurate estimate of Y . Here, information gain can be posed as a discrete sum over sparse elements of the homotopic belief. Furthermore, planning in this reduced latent belief space has the added benefit of encoding real-world intuition into sensing plans. Intuitively, to track a target through a cluttered environment one might wait at a decision point where the target must choose which high-level motion to take. For example, one might wait at an obstacle to observe whether a target continues left or right. This corresponds to sensing in locations where the partial h -signature, and thus the homotopic belief, is most likely to change.

We propose a heuristic over the homotopic belief space that captures this intuition by comparing the differences between the robot’s current and future homotopic beliefs. To predict a future homotopic belief at some time t given the robot’s current belief, we calculate the likelihood that the target crosses a ray at time t , causing a transition from the current observed partial h -signature to the next, $p_{t-1} \rightarrow p_t$. This is performed by evaluating the cumulative distribution function of each component in the GMM at t and $t + 1$ over the domain beyond the ray. Thus, we have the probability of a ray crossing given a component, $P(p_t | (c, h))$.

With likelihoods of future partial h -signatures, the expected homotopic information gain at t can be evaluated. Recall from Equation (3.8) that the information gained by observing p_t can be captured via the Kullback-Liebler divergence between the observed and next homotopic belief. The expected homotopic information gain taken over all possible partial h -signature outcomes Ω_t at t is then given by

$$\mathbb{E}_{p_t \sim \Omega_t} [D_{\text{KL}}(b_{t-1}, b_t)] = \sum_{p_t} \sum_{c, h} P((c, h)) P(p_t | (c, h)) D_{\text{KL}}(P(h | p_{t-1}), P(h | p_t)), \quad (5.17)$$

where $P((c, h))$ is the weight of the GMM component.

The sensing robot does not take measurements of the target’s partial h -signature p_t directly, but rather of the target’s position \mathbf{y}_t . Although one may consider sensor and filter designs to enable such minimalist sensing and corresponding expected information gain directly (see for example [21, 182]), we wish to maintain measurements of position to help inform our overall estimate of Y . We merely plan over h as a computationally sparse proxy for sensing Y . Then, we must consider the sensing model in the calculation of expected homotopic information gain. To this end, we introduce again the probability of target detection given a sensing location, $P(D | \mathbf{x}_t)$ to relate information gain to sensing positions \mathbf{x}_t ,

$$\mathcal{IG}(h; \mathbf{x}_t) = P(D | \mathbf{x}_t) \cdot \mathbb{E}_{p_t \sim \Omega} [D_{\text{KL}}(b_{t-1}, b_t)]. \quad (5.18)$$

Here, $P(D | \mathbf{x}_t) = \sum_{c, h} P((c, h)) P(D | (c, h), \mathbf{x}_t)$ is given by marginalising Equation (5.16) over components. Furthermore, just as in Equation (5.3), this term prevents high KL events with low probability of detection being considered as informative.

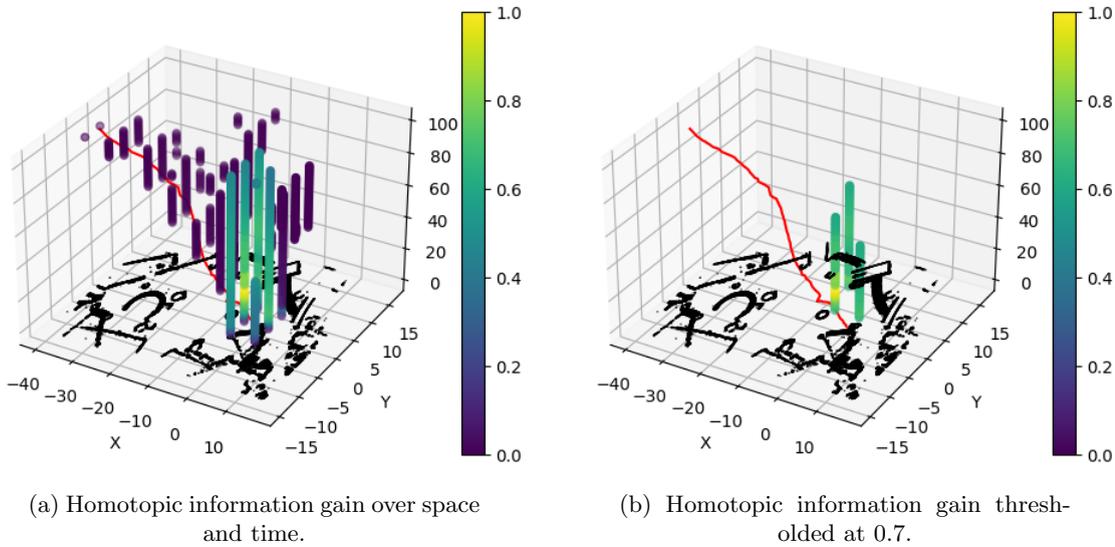


FIGURE 5.7: Heatcubes showing homotopic information gain at sensing locations and times. Yellow is higher information gain, blue is lower. Red line shows the test trajectory. z -axis is time.

The homotopic information gain at a sensing location can thus be calculated without knowledge of any measurement outcomes by simply evaluating the discrete sum in Equation (5.17) and the tractable probability in Equation (5.16). This is computationally advantageous compared to conventional approaches taken over the low-level belief space that involve intractable information-theoretic terms [180, 181].

5.3.3 Planning framework

To plan sensing trajectories given the homotopic information gain developed in the previous section, we calculate the information gain at all sensing locations for all future times, creating a heatcube depicted in Figure 5.7(a). Note that here the domain is discretised into a set of finite sensing locations, as we do not consider the continuous case. The information gain is sparsely distributed in space and time – it is highest around the first obstacle during the time window that the target is likely to move past this obstacle. The sensing robot thus does not need to consider all sensing locations in planning. Instead, we can threshold the heatcube to only the highest values. For example, in Figure 5.7(b) the heatcube is thresholded at 0.7 leaving only three sensing locations that persist along the time axis.

The remaining features in the thresholded heatcube and the time windows within which they exist are used to define an orienteering problem with time windows (OPTW) [183]. A solution to the OPTW selects locations to visit, and an order to visit them in, such that the time-constraints are satisfied and the expected reward is maximised. To find a solution, we construct a search tree over possible paths. The sensing robot is assumed to be an airborne robot which travels with constant velocity through the environment, unimpeded by obstacles. Then, if the robot can travel from node a to b before b 's time window closes given its maximum velocity, b is added to the tree as a child of a . The robot's initial position is the root node of the tree.

To explore the search tree and estimate the information gained along paths through the tree we implement MCTS (Chapter 3, Algorithm 2). This allows for non-myopic planning over the most informative sensing locations and times. The best immediate next action found via MCTS is taken and a measurement is acquired. With this measurement, the belief can be updated via Equations (5.13) and (5.14) and replanning over new nodes of interest can be performed.

5.4 Evaluation of homotopic belief space planning

5.4.1 Experiment setup

To evaluate the impact of planning over the proposed belief space via Equation (5.18), we compare to a conventional approach that maximises conditional entropy. This approach corresponds to planning over the full belief space and solving the original Problem 1.1. To avoid the overhead associated with the aforementioned pairwise distance estimates of entropy [181], we implement a naive measure of low-level information gain. We implement the weighted sum of conditional entropies of each Gaussian in the measurement GMM scaled by the probability of detection at \mathbf{x}_t . That is,

$$\mathcal{IG}(\mathbf{y}_t; \mathbf{x}_t) = P(D | \mathbf{x}_t) \cdot \sum_{c,h} \hat{w}_t^{(c,h)} \left(\log \det(\hat{\Sigma}_t^{(c,h)} + R^{(c,h)}(\mathbf{x}_t)) + \log(2\pi e) \right), \quad (5.19)$$

where $R^{(c,h)}(\mathbf{x}_t)$ is the measurement noise covariance, modelled as a function of the distance between \mathbf{x}_t and the mean of component (c, h) .

For both belief space approaches, heatcube features as in Figure 5.7(b) are calculated and the planning framework detailed in Section 5.3.3 is implemented. We compare each method across 250 random test trajectories in the shopping mall dataset.

Again, three metrics are used for comparison. The first is the ADE in Equation (5.10) to measure the success of each proposed solution to Problem 1.1. The remaining two are measures of the low- and high-level information gained per measurement. To measure information gain we introduce a ‘ground truth’ GMM distribution, calculated for each test by conditioning on the full trajectory data. Thus, the ground truth distribution represents the result if full information was able to be gained regarding the target. We compare the mutual information between the ground truth GMM and the posterior GMM conditioned on the measurement set after each new measurement is added. Then, the reduction in mutual information between measurements quantifies the information gained per measurement.

To measure low-level information gain we compute the KL divergence between the ground truth GMM and the posterior GMMs. The KL divergence is a measure of low-level mutual information between the distributions, however, as it is not analytically tractable for GMMs we utilise the variational approximation presented in [184]:

$$D_{\text{var}}(P(Y | h, \mathbf{z}_{1:T}), P(Y | p_t, \mathbf{z}_{1:t})) = \sum_{c,h} \hat{w}_T^{(c,h)} \log \frac{\sum_{c',h'} \hat{w}_T^{(c',h')} \exp D_{\text{KL}}(\mathcal{N}_T^{(c,h)}, \mathcal{N}_T^{(c',h')})}{\sum_{\tilde{c},\tilde{h}} \hat{w}_t^{(\tilde{c},\tilde{h})} \exp D_{\text{KL}}(\mathcal{N}_T^{(c,h)}, \mathcal{N}_t^{(\tilde{c},\tilde{h})})}. \quad (5.20)$$

Here $\mathcal{N}_T^{(c,h)}$ and $\mathcal{N}_t^{(c,h)}$ indicate the (c, h) -th component of the ground truth and posterior GMM at time t respectively. To measure the high-level information gain we again compute the KL divergence between the *weights* of the ground truth GMM and the posterior GMMs, as in Equation (5.12).

5.4.2 Results

Investigation of the number of measurements taken by each method for each test trajectory reveals the homotopic belief space planner took a median of 6 measurements per test trajectory, while the conventional belief space planner took 11. The full distributions over measurement number are shown in Figure 5.8(a). Due to the sparse spatial distribution of homotopic information gain, the proposed approach takes on average fewer measurements.

With more measurements taken by the conventional approach, one would expect a lower ADE, as more low-level information should improve the low-level estimate. Yet, comparison of the ADE in Figure 5.8(b) shows the mean for the proposed method is still within the interquartile range of the conventional. To investigate whether experiments with high numbers of measurements in the tail of the distribution for the proposed approach in Figure 5.8(a) is responsible for biasing the mean ADE downwards, we study the set of the most ‘typical’ runs for each method in Figure 5.9. That is, we compare all test trajectories where the median number of measurements for each method were taken.

For the proposed approach, the results shown in Figure 5.9 are averaged over the 47 test trajectories where the sensing robot took 6 measurements. The conventional approach results are averaged over the 58 test trajectories where 11 measurements were taken. This direct comparison of the most typical results for each method reveals a much more competitive ADE result between the two, shown in Figure 5.9(a). With only 6 measurements, the proposed approach gives statistically similar ADE to the conventional approach. This suggests homotopic belief-space planning provides more information per measurement at a low-level, despite being designed to maximise high-level motion information.

To interrogate this claim, we quantify high- and low-level information gain per measurement and compare the two methods. Boxplots of average mutual information reduction over typical runs for each approach are shown in Figure 5.9. We note that extracting meaning from comparisons of information gain per measurement is only possible because each test represented in the boxplots has an identical number of measurements. If boxplots in Figure 5.9 were presented for full results, each boxplot would have a varying number of data points.

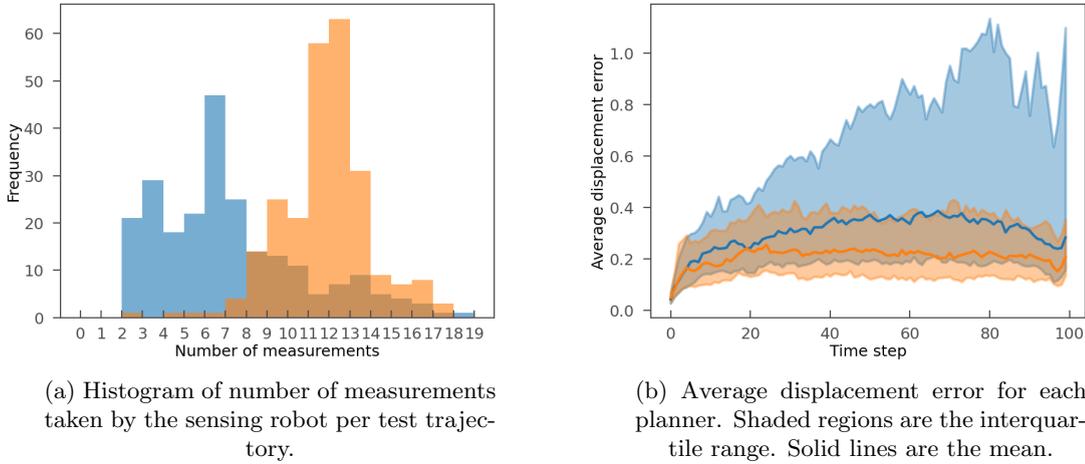
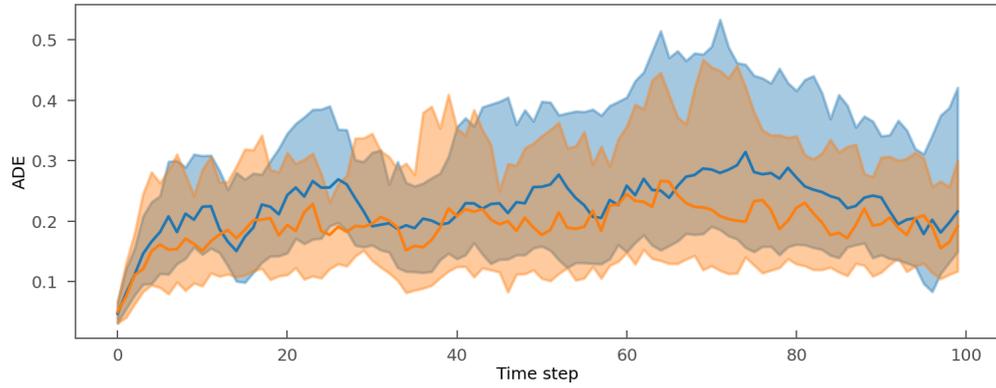


FIGURE 5.8: Results averaged over 250 test trajectories from the ATC shopping mall dataset. Blue is the proposed approach. Orange is the conventional approach.

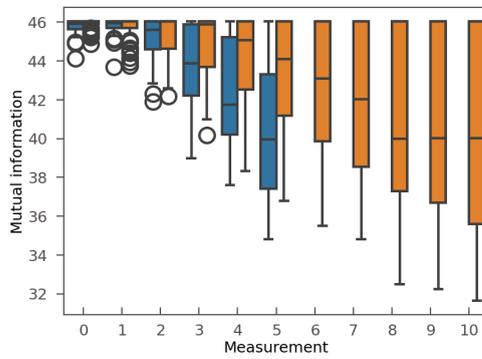
Nevertheless, evidenced by the median dataset, the proposed homotopic belief space planner indeed gathers more richly informative measurements for low-level estimation than the conventional belief space planner. As seen in Figure 5.9(b), the proposed approach achieves similar median mutual information to the conventional approach after only 6 measurements. Further, the interquartile range is tighter than the conventional approach, indicating greater consistency in measurement quality across these test trajectories.

Comparing the high-level information gain in Figure 5.9(c) confirms the homotopic belief space is superior for discerning high-level motions. The weights of the posterior GMM converge to the ground truth weights rapidly, reaching 0 KL divergence by the second measurement. Meanwhile the conventional approach lags behind in this regard, even diverging from the true weights in measurements 2 – 5.

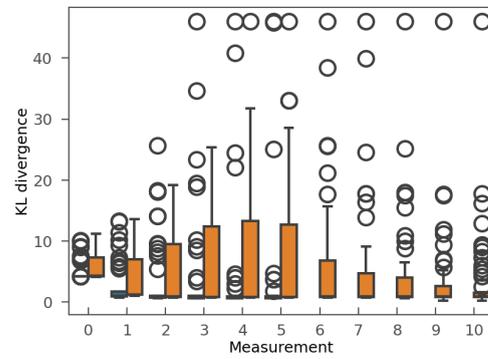
These results indicate that with less measurements, the homotopic belief space planning approach is indeed able to perform competitively in active target tracking scenarios compared to a conventional approach. Further investigation and work towards improving the proposed approach is underway to reduce variance over the full test dataset (Figure 5.8).



(a) Average displacement error (ADE) for each planner. Shaded regions are the interquartile range. Solid lines are the mean.



(b) Mutual information between posterior GMMs and ground truth GMM.



(c) Kullback-Liebler divergence between weights of posterior GMMs and ground truth GMM.

FIGURE 5.9: Results for the most typical runs for each planner. Blue is the proposed approach. Orange is the conventional approach.

5.5 Summary

In this chapter we introduced the h -signature as a latent variable to improve planning for active target tracking. The formulation of homotopic beliefs facilitated planning over a discrete belief space, where information gain was sparsely distributed throughout space. This circumvented the main challenge of planning over the full belief space associated with the target's low-level trajectory – there are no closed-form analytic expressions for entropy or mutual information for GMMs. Our tractable information gain metric associated with the homotopic belief space is a discrete sum over sparse PMFs, a clearly computationally efficient metric. However, in future work we still aim to fully characterise this efficiency.

Beyond computational simplicity, the homotopic belief space proved to be promising for planning sensing paths to estimate the full trajectory. Evaluation on real pedestrian data revealed promising low-level estimation results, even though the belief space is designed to capture high-level information. Comparison to a naive implementation of low-level information gain to represent a full belief space planning approach demonstrates competitive results despite taking almost half the number of measurements. Future work will aim to further improve the consistency of these results and utilise the sparsity of the measurement set to handle more resource-constrained tracking problems such as multi-target tracking.

In the next chapter we move away from active target tracking and propose a new latent variable and corresponding reduced belief space to improve solutions to another active perception problem: active mapping.

Chapter 6

The inducing point belief space for active mapping

In this chapter we present a reduced latent belief space planning framework for active mapping with a sensing robot. In the active mapping setting considered, the robot must plan sensing trajectories under dynamic constraints to estimate a static, scalar spatial field of interest. An inducing point-based sparse Gaussian process is used to build a representation of the field recursively as each noisy measurement is taken.

Conventionally this task is posed as a conditional entropy maximisation problem [17] requiring planning over the full, high-dimensional belief space associated with the spatial field estimate held at all possible sensing locations. Evaluation of the conditional entropy of the GP at a sensing location requires study of the GP covariance, which grows in dimension with each new measurement added. Thus, as more measurements are taken, planning over this belief space becomes increasingly cumbersome.

We circumvent this by leveraging the inducing points of the sparse GP framework to our advantage. We propose to plan in the much smaller latent belief space over the estimates held at the sparse GP's inducing points, referred to as the *pseudo-measurements*. That is, the latent variable ξ introduced for reduced belief space planning is the set of pseudo-measurements at inducing points. Planning in this reduced belief space involves minimising the conditional entropy at inducing points via study of a covariance matrix of constant

dimensionality. Furthermore, we prove that planning over this reduced belief space is in fact equivalent to taking a worst-case estimation error minimisation approach to the active mapping problem. This exciting theoretical development reveals a direct relationship between typical information gathering practices in robotics to deterministic estimation error minimisation for sparse GP regression.

First, we present our theoretical result: that the conditional entropy of pseudo-measurements given sensing measurements upper bounds the worst-case estimation error of a sparse GP regressor. We then corroborate the error bound in an abstract 1D example. Then, we evaluate our proposed reduced belief space approach on a simulated example of active mapping with an underwater robot operating in a flow field [185, 186]. For comparison, we demonstrate the behaviour of a typical measurement entropy maximisation approach equivalent to planning over the full belief space. This chapter builds upon our work published in [187].

6.1 Preliminaries and problem formulation

In the active mapping instance considered in this chapter, the state under estimation is a static, scalar spatial field in some domain $\mathcal{D} \subset \mathbb{R}^2$,

$$y = s(\mathbf{x}), \tag{6.1}$$

where $\mathbf{x} \in \mathcal{D}$.

The mobile sensing robot with dynamic model as in Equation (1.1) takes online scalar measurements $z_t \in \mathbb{R}$ of the spatial field at sensing location $\mathbf{x}_t \in \mathcal{D}$ with additive noise ϵ_t according to the following measurement model,

$$z_t = s(\mathbf{x}_t) + \epsilon_t. \tag{6.2}$$

The measurement noise ϵ_t is assumed to be bounded for all t . Given these noisy measurements we generate an estimate \hat{s} of the spatial phenomenon s across the entire domain $\mathbf{x} \in \mathcal{D}$ using GP regression with sparse approximation.

6.1.1 Sparse Gaussian process regression

A GP is a generalisation of multivariate Gaussian random variables (RVs) to random functions. It is an infinite collection of random variables such that any finite subset taken is jointly Gaussian distributed. A GP is characterised by a mean function $\mu(\mathbf{x})$ and a covariance function $k(\mathbf{x}, \mathbf{x}')$ which specifies the covariance between function values at different points \mathbf{x} and \mathbf{x}' :

$$\mathbb{E}[s(\mathbf{x})] = \mu(\mathbf{x}), \quad \text{Cov}[s(\mathbf{x}), s(\mathbf{x}')] = k(\mathbf{x}, \mathbf{x}'). \quad (6.3)$$

We impose a zero-mean GP prior on the scalar field of interest, that is, our initial belief of s is $\hat{s}(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}'))$, with $k(\mathbf{x}, \mathbf{x}')$ specified by the user. Let \mathbf{Z} be the vectorised representation of the complete set of noisy measurements $z_{1:T} = \{z_1, \dots, z_T\}$ up to T . That is, \mathbf{Z} is a vector with elements $[\mathbf{Z}]_t = z_t, \forall t \in \{1, \dots, T\}$. Similarly, let \mathbf{X} be a vector of the corresponding sensing locations $[\mathbf{X}]_t = \mathbf{x}_t, \forall t \in \{1, \dots, T\}$. Then, with the zero-mean prior, the estimate given the measurements $\hat{s}(\mathbf{x} | \mathbf{Z})$ is represented by another GP:

$$\begin{aligned} \hat{s}(\mathbf{x} | \mathbf{Z}) &\sim \mathcal{GP}(\mu(\mathbf{x} | \mathbf{Z}), \sigma^2(\mathbf{x}, \mathbf{x}' | \mathbf{Z})), \\ \mu(\mathbf{x} | \mathbf{Z}) &= \mathbf{k}_{\mathbf{X}}^{\top}(\mathbf{x}) K_{\mathbf{X}}^{-1} \mathbf{Z}, \\ \sigma^2(\mathbf{x}, \mathbf{x}' | \mathbf{Z}) &= \mathbf{k}(\mathbf{x}, \mathbf{x}') - \mathbf{k}_{\mathbf{X}}^{\top}(\mathbf{x}) K_{\mathbf{X}}^{-1} \mathbf{k}_{\mathbf{X}}(\mathbf{x}'), \end{aligned} \quad (6.4)$$

with vector $[\mathbf{k}_{\mathbf{X}}(\mathbf{x})]_t = k(\mathbf{x}, \mathbf{x}_t)$, and Gram matrix $[K_{\mathbf{X}}]_{t_1, t_2} = k(\mathbf{x}_{t_1}, \mathbf{x}_{t_2})$.

Inversion of the Gram matrix $K_{\mathbf{X}}$ in Equation (6.4) is computationally taxing at $\mathcal{O}(T^3)$. We use the *inducing point*-based approximation of the regression above, introduced in [147, 156] for its reduced computational complexity. Intuitively, this formulation introduces a small set of inducing points $\mathbf{M} = \{\mathbf{m}_i\}_{i=1}^{|\mathbf{M}|}$, $\mathbf{m}_i \in \mathcal{D}$ where $|\mathbf{M}| \ll T$. Then, one can assert that the measurements are conditionally independent (CI) given the function value estimates at the inducing points, or the vector of *pseudo-measurements* $[\mathbf{y}_{\mathbf{M}}]_i = \hat{s}(\mathbf{m}_i)$. A graphical depiction of this notion is given in Figure 6.1. In a full GP, all measurements are modelled with statistical dependence to capture spatial correlations and predict function values at unseen locations $s(\mathbf{x})$. In an inducing point-based sparse GP, the only statistical dependencies are between the pseudo-measurements and the measurement set. Thus,

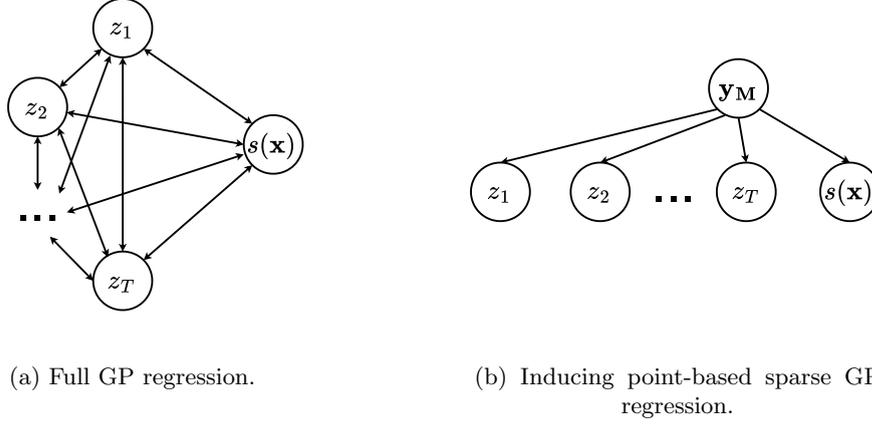


FIGURE 6.1: Graphical models for GP regression.

the pseudo-measurements \mathbf{y}_M ‘summarise’ the spatial correlations between measurements collected by the robot and predictions at unseen locations are *inferred* via the set of latent variables \mathbf{y}_M .

Mathematically, the CI property holds if and only if:

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{k}_M^T(\mathbf{x}) K_M^{-1} \mathbf{k}_M(\mathbf{x}') \quad \forall \mathbf{x} \neq \mathbf{x}', \quad (6.5)$$

where $[\mathbf{k}_M(\mathbf{x})]_i = k(\mathbf{x}, \mathbf{m}_i)$ and $[K_M]_{i,j} = k(\mathbf{m}_i, \mathbf{m}_j)$. This follows from asserting that the conditional cross-covariance vanishes given pseudo-measurements \mathbf{y}_M , i.e. $\sigma^2(\mathbf{x}, \mathbf{x}' | \mathbf{y}_M) = 0$. In other words, the correlation between any two measurements is indirect and is limited by their correlation to the pseudo-measurements.

We consider two popular inducing point-based approximations that satisfy CI: the subset of regressors (SoR) and fully independent conditional (FIC) approximations. As noted in [156], the SoR and FIC approximations are equivalent to replacing the kernel $k(\mathbf{x}, \mathbf{x}')$ in Equation (6.3) with approximate ones as follows:

$$\begin{aligned} \hat{k}_{\text{SoR}}(\mathbf{x}, \mathbf{x}') &= \mathbf{k}_M^T(\mathbf{x}) K_M^{-1} \mathbf{k}_M(\mathbf{x}'), \\ \hat{k}_{\text{FIC}}(\mathbf{x}, \mathbf{x}') &= \hat{k}_{\text{SoR}}(\mathbf{x}, \mathbf{x}') + \delta(\mathbf{x}, \mathbf{x}') (k(\mathbf{x}, \mathbf{x}') - \hat{k}_{\text{SoR}}(\mathbf{x}, \mathbf{x}')), \end{aligned} \quad (6.6)$$

where $\delta(\cdot)$ is the Kronecker delta function. Then, complexity reduces to $\mathcal{O}(|\mathbf{M}|^2 T)$ where $|\mathbf{M}| \ll T$, a significant reduction compared to the full GP regression in Equation (6.4).

Notably, complexity is now linear in the number of measurements, making online regression feasible even for large and complex spatial fields. While this complexity reduction makes inducing point sparse GPs popular for use in mapping to represent spatial field estimates, we are most interested in leveraging the approximations in Equation (6.6) in the *planning* stage of active mapping. In the following section we treat the pseudo-measurements as the latent variable ξ in Problem 1.2 whose belief space we plan over. Thus we exploit the inducing point approximation not just to obtain a sparse representation of s but also for efficient planning of high-quality sensing trajectories.

6.1.2 Problem formulation

Given a prior zero-mean GP representing the initial belief of s , the robot's task is to solve Problem 1.1. That is, to plan a sensing trajectory that minimises the uncertainty of the spatial field estimate \hat{s} across the whole domain \mathcal{D} . For a Gaussian process s , uncertainty is captured by a generalisation of entropy known as the *entropy rate*. The entropy rate captures the infinite nature of the collection of random variables in the Gaussian process, defined as $\bar{H}(s) = \lim_{n \rightarrow \infty} \frac{1}{n} H(X_1, \dots, X_n)$. With this in mind, the full belief space planning problem is written

Problem 6.1. *Given an initial belief of s and a planning horizon $T < \infty$ find a sequence of admissible controls $\sigma = \{\mathbf{u}_0, \dots, \mathbf{u}_{T-1}\}$ that minimises uncertainty in s at all locations, or the entropy rate*

$$\min_{\sigma \in \mathcal{U}^T} \bar{H}(s(\mathbf{x}) \mid \mathbf{Z}). \quad (6.7)$$

A sensing policy to this end is the selection of future sensing locations where the GP uncertainty is high given the current measurement set, or *conditional entropy maximisation* [17]. By sensing in these locations, the robot drives down the total entropy in Equation (6.7). As GPs are extensions of multivariate Gaussian RVs, the conditional entropy at a single location \mathbf{x} given the vector of measurements $\mathbf{z}_{1:t}$ up to time $t < T$ ¹ takes the closed form for a finite Gaussian distribution given in Equation (3.11) where $k = t$ and $\Sigma_{Y|Z} = \sigma^2(\mathbf{x}, \mathbf{x} \mid \mathbf{z}_{1:t})$. Then, choosing a future sensing location with maximal conditional

¹Note we distinguish the vector of online measurements $\mathbf{z}_{1:t}$ taken up to time $t < T$ from the complete vector of measurements \mathbf{Z} up to time T notationally.

entropy amounts to maximising $\log \det \sigma^2(\mathbf{x}, \mathbf{x} \mid \mathbf{z}_{1:t+1})$. Inspection of Equation (6.4) shows this is equivalent to maximising $\log \det K_{\mathbf{x}_{1:t+1}}$, where $\mathbf{x}_{1:t+1}$ is the vector of sensing locations up to time $t + 1$.

As we can see, the heuristic associated with this policy involves the Gram matrix $K_{\mathbf{x}_{1:t+1}}$ that grows in dimension with each measurement added until reaching its maximum size at the final sensing time T . Then for large T this approach can become computationally taxing. To avoid this growing dimensionality we propose to instead plan over the reduced belief space associated with the pseudo-measurements. This also allows us to minimise *entropy* rather than *entropy rates*, as the inducing point locations are a finite subset of the infinite, continuous domain.

Problem 6.2. *Given an initial belief of s and a planning horizon $T < \infty$ find a sequence of admissible controls $\sigma = \{\mathbf{u}_0, \dots, \mathbf{u}_{T-1}\}$ that minimises uncertainty in s at inducing point locations*

$$\min_{\sigma \in \mathcal{U}^T} H(\mathbf{y}_M \mid \mathbf{Z}). \quad (6.8)$$

Here we need only to compute and minimise the conditional entropy of a fixed number of latent variables \mathbf{y}_M – the pseudo-measurements. In Section 6.3 we further exploit this benefit by using an efficient recursive sparse GP algorithm presented in [148] that maintains a Gaussian belief over \mathbf{y}_M . We therefore once again present an instance of reduced latent belief space planning, here for application in active mapping problems. In the following sections we prove that the above problem reduction is equivalent to worst-case estimation error minimisation, connecting information-theoretic active perception approaches to deterministic error reduction for the first time.

6.2 Information gain and worst-case error minimisation

The deterministic error between the true spatial phenomenon s and the sparse GP estimate \hat{s} is the total absolute point-wise difference, defined as

$$\int_{\mathcal{D}} |s(\mathbf{x}) - \mathbb{E}[\hat{s}(\mathbf{x} \mid \mathbf{Z})]| d\mathbf{x}. \quad (6.9)$$

To minimise this estimation error, one would require access to the ground truth field s . As s is unavailable in realistic robotics applications, deterministic error minimisation is not generally considered as a heuristic for planning active mapping trajectories and information-theoretic heuristics are used instead. By studying sparse GP regression through a kernel-based interpolation theoretic lens, we show that the latent variable belief space planning problem proposed in Problem 6.2 is in fact a proxy for minimising deterministic error.

A connection between GPs and kernel-based interpolation on reproducing kernel Hilbert spaces (RKHSs) follows from recognising that the covariance function in Equation (6.3) is a *positive definite kernel*. A kernel k is positive definite if, for any choice of \mathbf{X} , the Gram matrix $K_{\mathbf{X}}$ is positive definite. Any positive definite kernel k uniquely defines an RKHS, \mathcal{H}_k . An RKHS is a space of real-valued functions equipped with an inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$ such that the positive definite kernel satisfies:

1. $k(\cdot, \mathbf{x}) \in \mathcal{H}_k \quad \forall \mathbf{x} \in \mathcal{D}$,
2. $\langle f, k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}_k} = f(\mathbf{x}) \quad \forall \mathbf{x} \in \mathcal{D}, f \in \mathcal{H}_k$.

Intuitively, this means that the kernel function $k(\cdot, \mathbf{x})$ is itself an element of \mathcal{H}_k , and that it ‘reproduces’ all other functions in \mathcal{H}_k at \mathbf{x} . Further, the inner product induces a norm $\|f\|_{\mathcal{H}_k} = \sqrt{\langle f, f \rangle_{\mathcal{H}_k}}$.

With the RKHS framework at hand, one may draw connections between kernel-based interpolation and GP regression to bound deterministic error of a full GP regressor. Just as in GP regression, kernel-based interpolation is concerned with estimating function values at unseen locations given observations at locations $\{\mathbf{x}_1, \dots, \mathbf{x}_T\}$. Specifically, the aim is to find the *minimum-norm interpolant* $f \in \mathcal{H}_k$ of a function s by solving the optimisation problem,

$$\begin{aligned} f &:= \arg \min_{g \in \mathcal{H}_k} \|g\|_{\mathcal{H}_k} \\ \text{s.t. } & g(\mathbf{x}_i) = s(\mathbf{x}_i), \quad \forall \mathbf{x}_i \in \{\mathbf{x}_1, \dots, \mathbf{x}_T\}. \end{aligned} \tag{6.10}$$

For noise-free observations the unique solution to this optimisation problem can be written as a sum of the kernel functions at measurement locations,

$$f(\mathbf{x}) = \sum_{i=1}^T \alpha_i k(\mathbf{x}, \mathbf{x}_i). \quad (6.11)$$

where $\alpha_i = [K_{\mathbf{X}}^{-1}]_{ij} [\mathbf{Z}]_j$ using Einstein notation. This form of α_i is familiar. In fact, we can see that the above minimum-norm interpolant is equivalent to the posterior GP mean given in Equation (6.4). This equivalence allows one to directly apply error analysis developed for the minimum-norm interpolant in kernel-based interpolation literature to GP regression with noise-free observations as in [153, 155].

Inspired by this work, we derive the following worst-case error bound for both full and sparse GP regression with measurements containing bounded noise, which is representative of robotics applications. All proofs of results are available in Appendix B.

Theorem 6.1. *Suppose $s \in \mathcal{H}_k$ with arbitrary positive definite kernel k . For bounded measurement noise $\epsilon^2 < \sigma_\epsilon^2 < \infty$, we have for any measurement set \mathbf{Z} ,*

$$|s(\mathbf{x}) - \mathbb{E}[\hat{s}(\mathbf{x} | \mathbf{Z})]| \leq \|s\|_{\mathcal{H}_k} P_{\mathbf{X}}(\mathbf{x}) + \sqrt{\sigma_\epsilon^2 T \Lambda_k^2(\mathbf{x})}, \quad (6.12)$$

where $\mathbb{E}[\hat{s}(\mathbf{x} | \mathbf{Z})]$ is the mean of the posterior GP, $P_{\mathbf{X}}(\mathbf{x}) = \sqrt{\frac{\det K_{\mathbf{X} \cup \{\mathbf{x}\}}}{\det K_{\mathbf{X}}}}$ is called the power function of \mathbf{X} and $\Lambda_k(\mathbf{x}) = \|K_{\mathbf{X}}^{-1} \mathbf{k}_{\mathbf{X}}(\mathbf{x})\|$.

Error bounds for GP regression have been studied in the past. In [188], probabilistic error bounds assuming bounded noise are derived based on a multi-arm bandit analysis. In contrast, our bounds are deterministic. In [189] bounds are derived for Gaussian measurement noise, a more general case than the one considered in this thesis. However, these bounds require that the unknown function and the GP kernel function are Lipschitz continuous, a requirement that we do not enforce. Most interestingly, our bounds link GP regression error to *information theoretic* active perception regimes. From Theorem 6.1 we see that Equation (6.9) can be minimised by choosing measurement locations \mathbf{X} that minimise $P_{\mathbf{X}}(\mathbf{x})$ for all possible \mathbf{x} . In fact, via $P_{\mathbf{X}}(\mathbf{x}) = \sqrt{\frac{\det K_{\mathbf{X} \cup \{\mathbf{x}\}}}{\det K_{\mathbf{X}}}}$ we find a close relationship between the bound in Equation (6.12) and the information theoretic approach

in Problem 6.1 – both aim to choose sensing locations where the conditional entropy or uncertainty, characterised by $\det K_{\mathbf{X}}$, is maximal.

We can further prove equivalence between the proposed latent belief space planning approach in Problem 6.2 by exploiting the CI property of the sparse approximations presented in Equation (6.5). Because CI kernels can be viewed as interpolants to the true kernel [159], the interpolation of $s(\mathbf{x})$ given \mathbf{Z} can be decomposed into two stages: 1) the interpolation of pseudo-measurements $\mathbf{y}_{\mathbf{M}}$ given \mathbf{Z} , and 2) the interpolation of $s(\mathbf{x})$ given $\mathbf{y}_{\mathbf{M}}$. Then, it is natural to ask if the deterministic error in Equation (6.12) or the power function admits a similar decomposition. The following theorem confirms that there is such a decomposition.

Theorem 6.2. *Suppose a kernel k satisfies the CI assumption (Equation (6.5)). Then, the power function $P_{\mathbf{X}}(\mathbf{x})$ satisfies:*

$$P_{\mathbf{M}}(\mathbf{x}) \leq P_{\mathbf{X}}(\mathbf{x}) \leq P_{\mathbf{M}}(\mathbf{x}) \exp H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}), \quad (6.13)$$

where $P_{\mathbf{M}}(\mathbf{x}) = \sqrt{\frac{\det K_{\mathbf{M} \cup \{\mathbf{x}\}}}{\det K_{\mathbf{M}}}}$.

Moreover, assuming $s \in \mathcal{H}_k$, the deterministic error can be further bounded:

$$|s(\mathbf{x}) - \mathbb{E}[\hat{s}(\mathbf{x} | \mathbf{Z})]| \leq \|s\|_{\mathcal{H}_k} P_{\mathbf{M}}(\mathbf{x}) \exp H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}) + \sqrt{\sigma_e^2 T \Lambda_k^2(\mathbf{x})}. \quad (6.14)$$

When considering only the control-dependent term $H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z})$ in the upper bound in Equation (6.14), Theorem 6.2 draws direct equivalence between deterministic error minimisation and the latent belief space active perception problem proposed in Problem 6.2. Further, through comparison of Equations (6.12) and (6.14) we note that the solving the reduced Problem 6.2 serves as a closer proxy for minimising deterministic error than the conventional Problem 6.1 since $H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}) \rightarrow 0$ as more measurements are added, or with better solution quality. This is owing to the ‘tightness’ of the inequality in Equation (6.13) in that $P_{\mathbf{X}}(\mathbf{x}) \rightarrow P_{\mathbf{M}}(\mathbf{x})$ as $H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}) \rightarrow 0$.

6.3 Planning framework

Beyond its equivalence to error minimisation, a further benefit of latent belief space planning problem as posed in Problem 6.2 is that we can recursively maintain a Gaussian belief over the pseudo-measurements via a Kalman filtering-like process. This Gaussian representation allows utilisation of the RVI algorithm detailed in Chapter 3 for solving the reduced active mapping problem.

6.3.1 Recursive sparse GP regression

Given measurement vector $\mathbf{z}_{1:t}$ up to time t , the recursive sparse GP regression algorithm [148] permits equivalent calculations as in Equation (6.4), while only storing the posterior mean and covariance of the pseudo-measurements \mathbf{y}_M :

$$\mu_t = \mathbb{E}[\mathbf{y}_M \mid \mathbf{z}_{1:t}], \quad \Sigma_t = \text{Cov}[\mathbf{y}_M \mid \mathbf{z}_{1:t}]. \quad (6.15)$$

Importantly, the posterior entropy Equation (6.8) can be calculated as a function of Σ_t :

$$H(\mathbf{y}_M \mid \mathbf{z}_{1:t}) = \frac{|\mathbf{M}|}{2} \log \det 2\pi e \Sigma_t. \quad (6.16)$$

The recursive update procedure is analogous to a Kalman filter. Initially, the belief is set to $\mu_0 = 0$ and $\Sigma_0 = K_M$. Given measurement z_t at \mathbf{x}_t , we generate the predictive mean, variance and cross-covariance at future time step:

$$\begin{aligned} \hat{z}_{t+1} &= \mu(\mathbf{x}_{t+1} \mid z_t), \\ \Sigma_{t+1}^{zz} &= \sigma_*^2(\mathbf{x}_t, \mathbf{x}_t) + \sigma_\epsilon^2, \\ \Sigma_{t+1}^{zM} &= \mathbf{k}_M^\top(\mathbf{x}_t) K_M^{-1} \Sigma_t, \end{aligned} \quad (6.17)$$

where $*$ is a placeholder for the CI kernels SoR or FIC and $\sigma_\epsilon < \infty$ is an upper bound on

the peak measurement noise. Using the predictions in Equation (6.17), and new measurement z_{t+1} at \mathbf{x}_{t+1} we can perform a belief update:

$$\begin{aligned}\mu_{t+1} &= \mu_t + \Sigma_{t+1}^z \mathbf{M} (\Sigma_{t+1}^{zz})^{-1} (z_{t+1} - \hat{z}_{t+1}), \\ \Sigma_{t+1} &= \Sigma_t - \Sigma_{t+1}^z \mathbf{M} (\Sigma_{t+1}^{zz})^{-1} (\Sigma_{t+1}^z \mathbf{M})^\top.\end{aligned}\tag{6.18}$$

Finally, we can recover the *full* estimate given any location $\mathbf{x} \in \mathcal{D}$ (i.e., perform regression) given the pseudo-measurement belief μ_t and Σ_t :

$$\begin{aligned}\mu(\mathbf{x} \mid \mathbf{z}_{1:t}) &= \mathbf{k}_M^\top(\mathbf{x}) K_M^{-1} \mu_t, \\ \sigma_{SoR}^2(\mathbf{x}, \mathbf{x}' \mid \mathbf{z}_{1:t}) &= \mathbf{k}_M^\top(\mathbf{x}) K_M^{-1} \Sigma_t K_M^{-1} \mathbf{k}_M^\top(\mathbf{x}'), \\ \sigma_{FIC}^2(\mathbf{x}, \mathbf{x}' \mid \mathbf{z}_{1:t}) &= \sigma_{SoR}^2(\mathbf{x}, \mathbf{x}' \mid \mathbf{z}_{1:t}) + \delta(\mathbf{x}, \mathbf{x}') (k(\mathbf{x}, \mathbf{x}') - \hat{k}_{SoR}(\mathbf{x}, \mathbf{x}')).\end{aligned}\tag{6.19}$$

Thus, we have a Gaussian belief space of fixed dimension, from which we can infer the full spatial field estimate. The pseudo-measurement covariance prediction and update Equations (6.17) to (6.18) are measurement independent, enabling efficient belief space planning to minimise the posterior entropy in Equation (6.16) over non-myopic horizons.

6.3.2 Receding horizon planning

Given the pseudo-measurement belief maintained by the recursive GP updates, we solve the reduced latent belief space planning problem in Problem 6.2 using an adapted version of the RVI algorithm detailed in Chapter 3. With each expansion of the belief tree facilitated by Equation (6.17) we extract and execute the best sensing action found via search for the lowest-cost leaf node, where the cost is given by Equation (6.16). Then, the belief is updated via Equation (6.18). A new plan is generated according to the updated belief. However, since the spatial field is static, we re-use the subtree rooted at the selected sensing node in the previous planning stage. Because the RVI iteration adds a new layer of leaves, the depth of the tree is always equal to horizon T .

As noted in [190], the ϵ -algebraic redundancy check in line 14 of Algorithm 1 is an instance of a linear matrix inequality feasibility problem, and poses computational challenge as the number of inducing points grows. This can be circumvented by setting $\epsilon = \infty$. In this

case, the RVI iteration only adds the lowest cost nodes that are not within δ distance of each other. While there are no finite bounds in this case, it produces practically viable and fast solutions. For further detail on how pruning impacts tree size in RVI see [20, 117].

6.4 Experiment results

6.4.1 Characterisation of the error bound

We first corroborate the error bound proposed in Theorem 6.2 with an example. As the RKHS norm required for computing the error bound may be hard to compute, particularly in higher dimensional settings, we consider a one-dimensional regression problem where the target function is designed with simplified form $s(x) = \sum_{i=1}^n \alpha_i k(x, x_i)$. This way we ensure $s \in \mathcal{H}_K$, and the RKHS norm is easily reduced to the Euclidean norm. Such simplifications are enough to illustrate the connection between our latent belief space planning problem formulation and worst-case deterministic error minimisation.

Figure 6.2 depicts the outcome of regression with such a target function. With sparse and noisy measurements, the GP regressor is able to reconstruct the target function well, as evidenced in the top panel. Importantly, here we see that the bound in Equation (6.14) is reasonably tight on the predicted mean and follows the intuitive behaviour of decreasing near measurement locations. When compared to the 1σ -confidence interval obtained from the posterior covariance, the proposed bounds have overall higher value. However, in certain regions of sparse or no measurement the target function is greater than the 1σ -confidence interval and yet remains within our error bound. This demonstrates that while confidence intervals may be broken, the error bound may not.

In the lower panel of Figure 6.2, we show the deterministic error against the proposed upper bound in Equation (6.14), confirming the deterministic error lies below the bound for all x in the domain. This result corroborates our theoretical finding that the proposed latent belief space planning approach does indeed minimise deterministic error reasonably tightly.

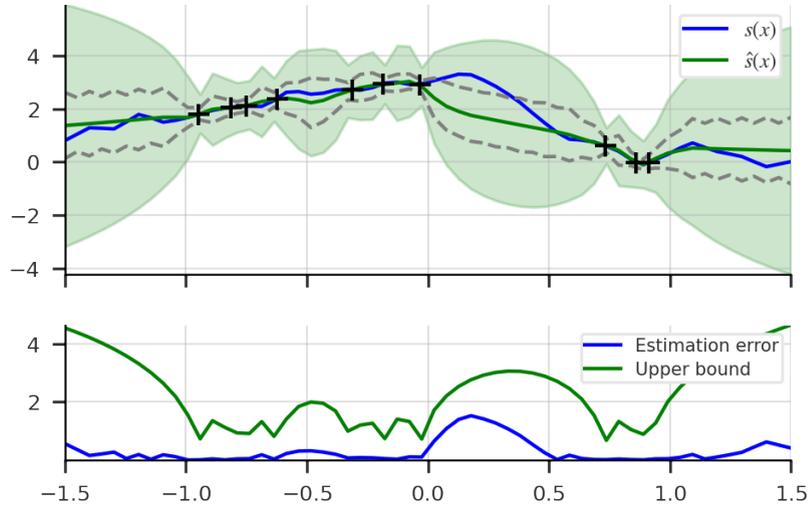


FIGURE 6.2: A study of estimation error for one-dimensional sparse Gaussian process regression of a target function in the RKHS. *Top*: The sparse GP estimate in solid blue compared to the ground truth function in solid green. Measurement locations are indicated with black crosses. The proposed bound on deterministic error is shown via the green shaded area. The area enclosed by the grey dotted lines is the standard 1σ -confidence interval. *Bottom*: The deterministic estimation error of the estimate in the top panel in blue compared to the upper bound in green.

6.4.2 Flow field case study

To demonstrate capability of the proposed algorithm and problem formulation for active mapping, we consider a simplified underwater glider operating in a double-gyre flow field. The dynamics are given by [140]

$$\begin{bmatrix} \mathbf{x}_{t+1}^1 \\ \mathbf{x}_{t+1}^2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_t^1 \\ \mathbf{x}_t^2 \end{bmatrix} + \Delta t (V_g \begin{bmatrix} -\sin(\pi \mathbf{x}^1) \cos(\pi \mathbf{x}^2) \\ \cos(\pi \mathbf{x}^1) \sin(\pi \mathbf{x}^2) \end{bmatrix} + V \begin{bmatrix} \cos u_t \\ \sin u_t \end{bmatrix}), \quad (6.20)$$

where δt is a small time translation, V_g is the velocity of the flow field, and V is the control velocity. The robot aims to solve reduced Problem 6.2 equipped with the planning framework described in Section 6.3 in order to reconstruct a scalar field of interest (such as level of salinity) over the flow field. The scalar field is shown in Figure 6.3 as a heatmap superimposed on the flow field.

To verify that the latent belief space planning algorithm simultaneously solves the deterministic error minimisation problem, we evaluate the average absolute error sampled over a 30×30 grid. This amounts to querying the posterior GP at 900 test locations and

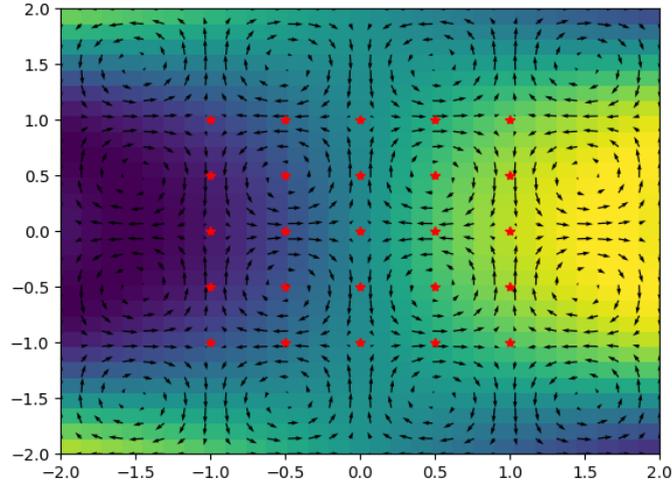


FIGURE 6.3: The ground truth active mapping environment. Double-gyre flow field vectors are shown in black. Yellow indicates a higher value for the scalar field, blue indicates lower values. Inducing point locations used for sparse GP regression are shown in red.

calculating error compared to ground truth at these locations. To verify the advantage of planning over the reduced latent belief space compared to the full belief space we also examine the behaviour of the entropy maximisation solution to Problem 6.1, which is a well-established approach to active mapping with GPs [17]. This was implemented by setting the cost in Algorithm 1 as $-\log \det K_{\mathbf{X}}$. For each approach, we vary the planning horizon T between $\{1, 5, 10\}$ time steps and examine the absolute error over time.

Figure 6.4(a) shows the performance of our proposed planner over 20 randomised initial starting locations in the same environment (Figure 6.3). It can be seen that for all choices of search horizon, the average absolute error decreases over time. The rate of reduction is greater with larger search horizon. Meanwhile, Figure 6.4(b) shows that the error for the full belief space planner does not decrease over time, and actually increases with larger search horizons ($T = 10$).

To better understand this behaviour, we show example trajectories produced via the proposed latent belief space planner after 100 time steps in Figure 6.5 and compare to an example trajectory produced via full belief space planning in Figure 6.6. For plans produced via our proposed approach with myopic greedy horizon $T = 1$, shown in Figures 6.5(a) and 6.5(d), we observe poor coverage of the spatial domain. Additionally, the myopic nature of this planner results in trajectories that often get ‘stuck’ in an attracting region

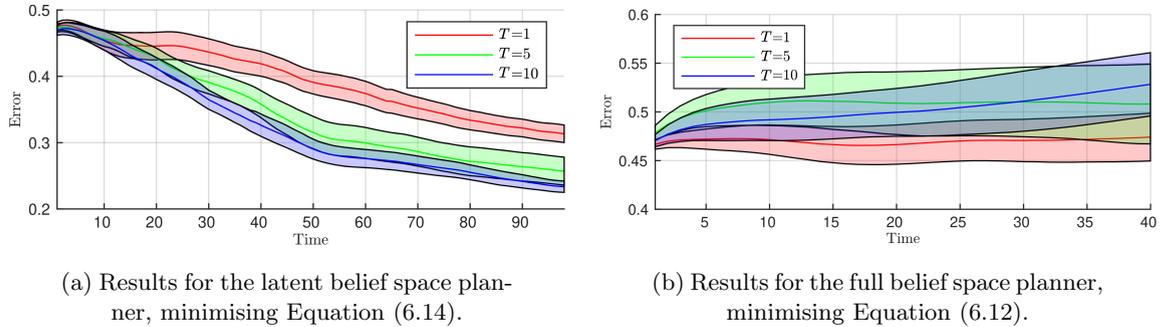


FIGURE 6.4: Average absolute reconstruction error with varying search horizon. Shaded areas represent 95% confidence interval over 20 trials.

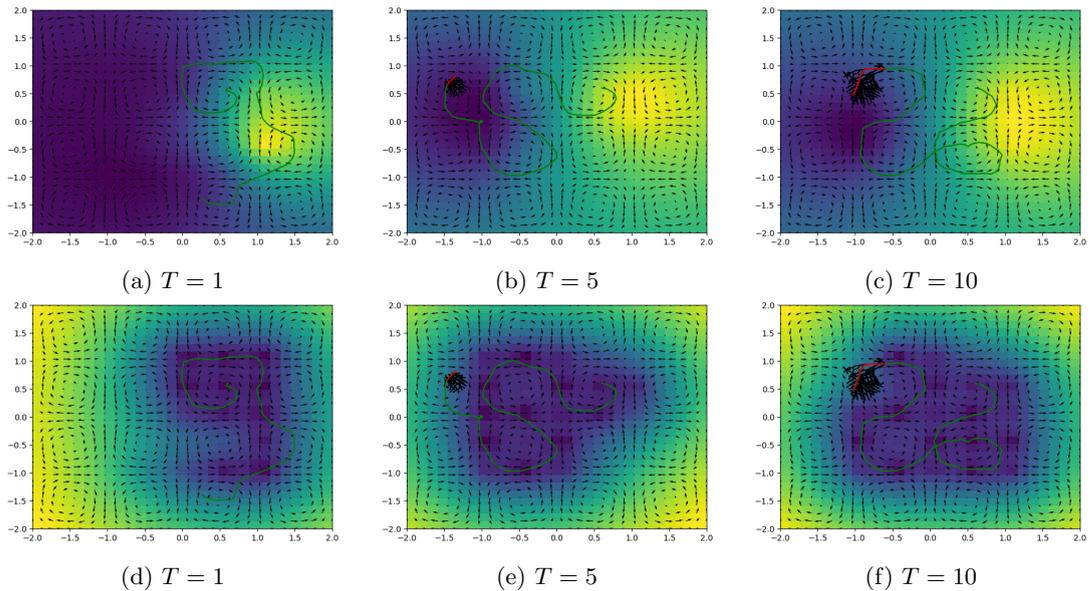


FIGURE 6.5: Example trajectories produced by the reduced belief space planner after 100 time steps with varying search horizon. Green: executed trajectory. Heatmaps in the top row depict the sparse GP mean after each sensing trajectory. Heatmaps in the bottom row depict sparse GP variances after each sensing trajectory. Red: current plan. Black: belief search tree.

of a gyre. The robot is unable to use the flow field dynamics to its advantage to explore, and reconstruction of the spatial field is thus poor.

With longer planning horizons the robot under our proposed framework successfully manoeuvres through the flow field to increase coverage of the domain, giving improved estimation of the spatial field as seen in Figures 6.5(b) and 6.5(c), with best estimation and coverage given by the longest horizon $T = 10$. Figure 6.5 further demonstrates the influence of planning over the reduced latent belief space. For all horizon lengths, the robot preferentially takes measurements near inducing points over exploring regions further away,

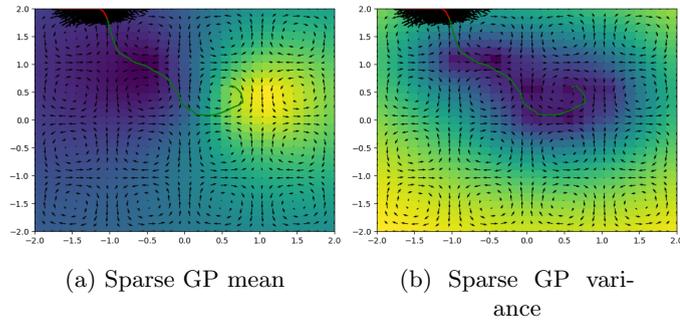


FIGURE 6.6: An example trajectory after 40 time steps produced by the full belief space planner with $T = 5$. Green: executed trajectory. Red: current plan. Black: belief search tree.

such that variance is minimised at inducing points. This is exemplified in Figure 6.5 where with increasing horizon, broader coverage of the region around inducing points and greater minimisation of variance is achieved.

Meanwhile, a sample trajectory for moderate horizon $T = 5$ produced via the full belief space planning or entropy maximisation approach is shown in Figure 6.6. Here, the robot explores outwards, making use of the ambient flow field. This is expected because the measurement entropy maximisation formulation demands the robot to simply move as far away from its previous trajectory as possible. However, this exploratory behaviour is problematic when using inducing point-based sparse GPs. As the robot gets further from the inducing points, the measurements do not make a significant contribution to improving estimation quality. These results corroborate our theoretical finding that the inducing point belief space is more suitable for deterministic error minimisation than the full belief space.

6.5 Summary

In this chapter we introduced the inducing point belief space for more efficient and accurate active mapping with sparse GPs, a common task in active perception. The dimensionality of the inducing point belief space is fixed throughout the active mapping task, providing a clear computational advantage over planning over the full belief space where dimensionality grows with each measurement taken.

Furthermore, we derived an exciting new theoretical connection between belief space planning and deterministic error minimisation for sparse GP regression. The proposed reduced latent belief space was proven to be a closer proxy for planning to reduce deterministic regression error than the full belief space. To validate our theoretical results we evaluated the proposed belief space plans in a one-dimensional illustrative regression example, demonstrating that minimising entropy at inducing point locations minimises worst-case error reasonably tightly. Evaluation in a more realistic active mapping scenario highlighted that our proposed method does indeed provide more accurate map reconstructions in shorter time than a full belief space planning approach.

Chapter 7

Conclusions and future work

In this thesis we introduced reduced latent belief planning as a new theoretical framework for active perception. The key idea is to introduce latent variables that can be inferred from measurements, and that the original system under estimation can be inferred from. By adding this layer of inference to the perception problem we take advantage of the latent variable's reduced belief space for planning sensing actions, thus easing the computational overhead of planning in active perception pipelines.

This framework was developed as an orthogonal approach to established belief space planning research, providing alternative belief spaces to apply existing algorithms to. The proposed belief spaces are of reduced size and admit information-theoretic planning heuristics that are more computationally manageable than traditional belief spaces.

To demonstrate the power of this framework, we proposed a suite of reduced latent belief spaces for different active perception problem instances. The proposed belief spaces were evaluated both empirically and theoretically, and it was shown throughout the thesis that our careful selection of latent variables resulted in belief space plans that improve estimates of the original system state when compared to plans formulated over the full belief space.

In the remainder of this chapter, we summarise these findings in more detail and present new avenues for future work unlocked by this framework. To conclude the thesis, a short discussion on the outlook for active perception is given.

7.1 Thesis summary

Here we summarise the contributions the technical chapters of the thesis, where each active perception problem instance was addressed. In each chapter, an example of a reduced latent belief space is proposed and its suitability for the active perception task at hand is evaluated.

7.1.1 The state belief space for active estimation of unknown inputs

In Chapter 4 active estimation of general systems with linear Gaussian dynamics under unknown inputs was addressed. Here, the sensing robot was required to jointly estimate both the state and the unknown input together. This would typically require planning over the joint belief space of these two variables.

We presented the state without unknown input as the latent variable of interest. Thus, the latent variable was simply a subset of the full system description and the problem was reduced from a joint estimation problem to a single variable estimation problem. This choice of latent variable was facilitated by an understanding of the unknown input filter used for belief updates. Recognising that the filter decouples state and unknown input estimation allows one to infer the full system description from the latent variable alone.

Planning over the latent belief space is simpler than planning over the full, joint belief space. Further, the state belief space is Gaussian in nature, allowing utilisation of existing Gaussian belief space planning algorithms such as RVI. Crucially, this admits derivation of suboptimality bounds for estimation of the full, original state description that results from the sensing trajectory planned via RVI. We derived and presented these bounds for the proposed extension of the RVI algorithm to this joint estimation problem. All proofs for these results are made available in Appendix A.

Finally, the proposed approach was evaluated on an example application of active estimation of linear Gaussian systems with unknown inputs: active target tracking. Here the target was moving with known linear Gaussian dynamics, however was performing unknown evasive manoeuvres at random time steps. We demonstrated that our proposed

approach was able to plan non-myopically in these conditions with our extension of RVI, giving accurate estimation of the full state.

7.1.2 The homotopic belief space for active target tracking

In Chapter 5 the active target tracking problem was further explored, and a new latent variable was introduced. Here, the target’s dynamics were learned via a GMM, a tool commonly used to represent various hypotheses over possible trajectories. However, for GMMs, information gain measures describing the expected uncertainty reduction in the belief given by a measurement have no closed-form analytic expressions. These metrics are crucial to guide selection of sensing actions. Then, to plan over the full belief space one must use estimates of the metrics, involving pairwise comparison of each GMM component. For large GMMs, this can become prohibitively difficult.

We instead proposed the target trajectory’s homotopy class as a latent variable from which the full trajectory could be inferred. This inference was enabled via the development of a hierarchical GMM, which acted as a predictive model over the full trajectory given predictions of the homotopy class. Evaluation of the proposed predictive model demonstrated promising prediction results compared to a standard GMM, confirming that the homotopy class is a suitable latent variable for inferring the full state.

The proposed homotopic belief space is both discrete and sparse, simplifying computation of information gain compared to a traditional approach tremendously. Further, sensing trajectories planned in the homotopic belief space demonstrated estimation accuracy competitive with those planned in the belief space. However, the proposed method took half the number of measurements on average to achieve this accuracy.

7.1.3 The inducing point belief space for active mapping

In Chapter 6 we addressed the active mapping problem. A sensing robot was tasked with reconstruction of a spatial field from noisy measurements via sparse GP regression. As more measurements are taken, the covariance matrix of the sparse GP increases in dimension. Thus, computing the information gain for long horizons and long sensing trajectories

can become cumbersome. Additionally, a typical approach that aims to minimise the final uncertainty of the GP estimate via selection of maximum entropy sensing locations was shown to demonstrate exploratory behaviour not conducive to estimation via inducing point-based sparse GPs.

As an alternative, we proposed the pseudo-measurements or the estimates held at inducing points as latent variables. Then, the covariance matrix representing uncertainty in these latent variables is of fixed dimension. Furthermore, information gain and belief updates can be calculated efficiently via a recursive Kalman-like filtering procedure. Most importantly, sensing trajectories planned in this reduced belief space only explored regions relevant for inducing point-based sparse GP regression, producing more accurate estimates of the spatial field in shorter time.

The proposed reduced latent belief space was also analysed theoretically. Most notably, we proved a direct connection between the typical belief space planning approach to active perception of uncertainty minimisation and explicit minimisation of worst-case estimation error. This connection was made via derivation of an information-theoretic upper bound on worst-case estimation error. We proved this connection for planning over both the full belief space and the proposed reduced belief space. For the reduced belief space we noted that the upper bound was tighter, and thus the reduced belief space was a closer conduit for error minimisation. All proofs for these theoretical developments are made available in Appendix B.

7.2 Future work

The reduced latent belief space planning framework presented in this thesis opens many new avenues for future work. In particular, the most exciting insight provided by the reduced latent belief space planning framework is in the introduction of *topological* latent variables such as homotopy classes. Specifically, we believe that these latent variables are prime candidates for improving active perception pipelines where metric information is unavailable, untrustworthy or cumbersome to work with – as is often the case in robotics.

Below, we expand on this insight by presenting specific avenues for future work in topological latent belief space planning. However, we note that this is not a complete treatment of future work as the applicability of this framework is broad beyond topological latent variables.

7.2.1 Active multi-target tracking via the homotopic belief space

Tracking multiple targets with fewer robots is a highly resource-constrained problem of interest in robotics settings such as search and rescue and surveillance. In realistic scenarios, sensing robots have time, velocity and compute constraints that restrict how often and when they may observe each target to maintain accurate estimates of their trajectories. In fact, it has been shown that under certain resource constraints such as number of sensing robots and maximum velocity, optimal multi-target tracking is impossible, as is tracking at a constant factor of optimality [125]. Active multi-target tracking is thus a difficult optimisation problem where the total information gain must be balanced between all targets. Previous work has explored solutions where target dynamics are constrained such that all targets remain within bounded distance of one another [191]. Other work has considered unconstrained target dynamics, but without sensing robot constraints [192], proposing to always focus on sensing the target with largest estimation uncertainty.

In Chapter 5 the homotopic belief space proved to be very promising for active target tracking. In particular, it was shown that accurate estimation could be achieved with very few measurements, due to the sparse nature of the homotopic belief space. This behaviour is highly desirable in the multi-target tracking setting, where resource constraints limit the number of measurements a robot may take of a single target. In future work we therefore wish extend homotopic belief space planning to address the active multi-target tracking problem.

7.2.2 Topological latent variables for localisation

In this thesis the sensing robot's state was considered fully known or observable, and robot dynamics were deterministic. In practice, actuation errors and unmodelled external forces

introduce stochasticity into the robot's dynamic model. Then, roboticists must tackle the challenge of localisation, where the robot's position must be inferred from sensing data. Metric information provided by odometry and global navigation satellite systems (GNSS) cannot be trusted alone due to drift and drop-out. Thus, additional sensing data from visual sensors or LiDAR is often leveraged to improve localisation.

Again, active localisation rather than passive is preferred. In active localisation, it has been shown that *loop closures* are highly informative locations to visit [11]. These are locations where the robot has potentially previously visited and sensed the environment. By visiting a potential loop closure, the robot can readjust its location estimate and resolve errors caused by accumulated odometry/GNSS drift or modelling inaccuracies. Identifying potential loop closure locations with LiDAR and visual sensing data brings its own challenges and avenues for future work, outlined below.

LiDAR loop closure detection

LiDAR sensors provide point cloud measurements representing three-dimensional range data. Where LiDAR sensors are used to assist in localisation, detecting loop closures therefore requires solving a point cloud registration problem. Measurements taken at identical sensing locations must be consolidated by finding a transformation that aligns the point clouds. The most common method for point cloud registration is the iterative closest point (ICP) approach [193–195], where in each iteration, every point in a reference point cloud is matched to a target point cloud via closest distance and a transformation is estimated via a minimisation problem. Iterations are completed until convergence is achieved. However, this method suffers from computational issues, as it requires a point-to-point or point-to-plane comparison between each point cloud. To circumvent this, a topological representation of the LiDAR scans may be considered instead. The *homology* of a manifold gives a categorisation of equivalent manifolds according to their structure. As each point cloud represents a surface or manifold, homology could therefore be used as a latent variable to guide sensing to locations where point cloud alignment is possible – or loop closure locations.

Visual loop closure detection

Localisation using visual sensors often involves feature-based methods, where highly recognisable features are extracted from sensing data and used to confirm loop closures [106]. However, if the environment is largely featureless – for example in Mars-like terrains, where visual sensors capture images of largely indistinguishable rock formations – this can be near impossible. Then, it may be prudent to study the *topology* of rock formations to identify loop closures. Once again, various topological latent variables capturing the homology of a rock formation can be explored to this end, opening avenues for planning over new topological latent belief spaces to visit potential loop closures.

7.2.3 Improving efficiency of active perception solutions

The planning algorithms used in this thesis were sampling-based tree search algorithms that suffer computational issues. Even with reductions made such as the pruning in RVI, these methods can still be slow. Our proposed reduced latent belief spaces mitigate this by providing a smaller, more informative search space to plan over, yet improvements are still required to have fast and accurate active perception on live robots. We therefore wish to address trajectory optimisation approaches in our future work on active perception, avoiding discretisation and sampling all together. In similar vein to ergodic search methods [37, 38], optimising continuous trajectories to gather information may be a fruitful next step. For example, we wish to further explore the use of homotopy classes in information gathering. The full set homotopy classes of a space describe the complete set of ways in which one may traverse through it, providing a natural solution to the *coverage* problem. Linking homotopy classes to ergodic search, or utilising them to reduce the complexity of the ergodic search problem is a promising future avenue for planning continuous active perception trajectories.

7.3 Outlook

To truly endow robots with human-like intuition regarding how to *look* rather than *see*, we must continue developing increasingly sophisticated frameworks for active perception. Importantly, roboticists must ensure that theoretical and algorithmic developments work toward expedient planning without compromising accurate estimation.

Research to this end has begun moving robots out of factory floors and away from repetitive, structured tasks. In the current state of research, we are already on the eve of the proliferation of cobots [8, 196], self-driving vehicles [197], and autonomous extra-terrestrial exploration [5]. The continuation of dedicated research to active perception could see robots exhibiting a level of autonomy previously only imagined. Fast and accurate active perception could enable the development of real-time assistive technology to empower autonomy in visually impaired people, or even enhance human perception with non-biological sensing capabilities.

Our introduction of reduced latent belief space planning for active perception is a contribution toward this goal. We have demonstrated that the intelligent selection of latent variables provides computational efficiency, theoretical guarantees and impressive empirical results. We hope that this new approach to active perception inspires the discovery and research of more reduced latent belief spaces and further application of this ideology.

Appendix A

Proofs of results in Chapter 4

A.1 Proof of Lemma 4.1

Lemma 4.1. *The unknown input filter state estimation error covariance update map is:*

1. *Monotone: if $\Sigma_1 \preceq \Sigma_2$ then $\rho(\Sigma_1) \preceq \rho(\Sigma_2)$*
2. *Concave: $\forall \alpha \in [0, 1], \rho(\alpha\Sigma_1 + (1 - \alpha)\Sigma_2) \succeq \alpha\rho(\Sigma_1) + (1 - \alpha)\rho(\Sigma_2)$.*

We first prove monotonicity. M_t^* and K_t^* are the optimal gains in the sense that they minimise Σ_t . Thus, $\forall M_t, K_t \neq M_t^*, K_t^*$, we have

$$\rho(\Sigma_{t-1}, M_t^*, K_t^*) \preceq \rho(\Sigma_{t-1}, M_t, K_t).$$

For any $\Sigma_1, \Sigma_2 \in \mathcal{P}^+$ with $\Sigma_1 \preceq \Sigma_2$ and any fixed non-optimal M_t, K_t , we have from Equation (4.10),

$$\rho(\Sigma_2, M_t, K_t) - \rho(\Sigma_1, M_t, K_t) = \tilde{A}_t \Sigma_2 \tilde{A}_t^\top - \tilde{A}_t \Sigma_1 \tilde{A}_t^\top \succeq 0.$$

Proving that ρ is affine in its first argument. Recalling that the optimal gains are functions of Σ , it then follows from the optimality of M_t^* and K_t^* and affinity of ρ that

$$\rho(\Sigma_1, M_t^*(\Sigma_1), K_t^*(\Sigma_1)) \preceq \rho(\Sigma_1, M_t^*(\Sigma_2), K_t^*(\Sigma_2)) \preceq \rho(\Sigma_2, M_t^*(\Sigma_2), K_t^*(\Sigma_2))$$

We next prove concavity. Let $\chi = \alpha\Sigma_1 + (1 - \alpha)\Sigma_2 \forall \alpha \in [0, 1]$ where again $\Sigma_1, \Sigma_2 \in \mathcal{P}^+$.

From Equation (4.10) and optimality of gains we have the result:

$$\begin{aligned} \rho(\chi, M_t^*(\chi), K_t^*(\chi)) &= \alpha\rho(\Sigma_1, M_t^*(\chi), K_t^*(\chi)) + (1 - \alpha)\rho(\Sigma_2, M_t^*(\chi), K_t^*(\chi)) \\ &\succeq \alpha\rho(\Sigma_1, M_t^*(\Sigma_1), K_t^*(\Sigma_1)) + (1 - \alpha)\rho(\Sigma_2, M_t^*(\Sigma_2), K_t^*(\Sigma_2)). \end{aligned}$$

A.2 Proof of Lemma 4.2

Lemma 4.2. *The directional derivative of the state estimation covariance update map at $\Sigma \in \mathcal{P}^+$ along the arbitrary direction $X \in \mathcal{P}^+$ is given by*

$$\left. \frac{d\rho_u(\Sigma + \epsilon X)}{d\epsilon} \right|_{\epsilon=0} = \tilde{A}(\Sigma)X\tilde{A}(\Sigma)^\top,$$

where $\tilde{A}(\Sigma)$ is defined as in Equation (4.10). Further, the directional derivative of the t -horizon mapping ϕ_t at $\Sigma \in \mathcal{P}^+$ along an arbitrary direction $X \in \mathcal{P}^+$ is given by

$$g_t^\sigma(\Sigma, X) = \prod_{k=0}^{t-1} (\tilde{A}_{t-k})X \prod_{k=0}^{t-1} (\tilde{A}_k)^\top,$$

$\forall t \in \{1, \dots, T\}$, with $g_0^\sigma(\Sigma, X) = X$.

We prove the result for a general gain matrix $K_t^* = (\Sigma_t^*(\epsilon)C_t^\top + S_t^*)\alpha_t^\top (\alpha_t \tilde{R}_t^*(\epsilon)\alpha_t^\top)^{-1}\alpha_t$ where α_t can be chosen as in [22] Theorem 7. Here,

$$\begin{aligned} S_t^* &= -G_{t-1}\hat{M}_t R_t, \\ \tilde{R}_t^*(\Sigma_{t-1}) &= (I - C_t G_{t-1} M_t) \tilde{R}_t(\Sigma_{t-1}) (I - C_t G_{t-1} M_t)^\top, \\ \Sigma_t^*(\Sigma_{t-1}) &= (I - G_{t-1} M_t C_t) (A_{t-1} \Sigma_{t-1} A_{t-1}^\top + Q_{t-1}) (I - G_{t-1} M_t C_t)^\top + G_{t-1} M_t R_t M_t^\top G_{t-1}^\top. \end{aligned}$$

For any $\Sigma, X \in \mathcal{P}^+$, let $\tilde{R}_t(\epsilon) = C_t A_{t-1} (\Sigma_{t-1} + \epsilon X) A_{t-1}^\top C_t^\top + C_t Q_{t-1} C_t^\top + R_t$. Then $\tilde{R}_t(0) = C_t A_{t-1} \Sigma_{t-1} A_{t-1}^\top C_t^\top + C_t Q_{t-1} C_t^\top + R_t$. Similarly, let $K_t(\epsilon) = (\Sigma_t^*(\epsilon)C_t^\top + S_t^*)\alpha_t^\top (\alpha_t \tilde{R}_t^*(\epsilon)\alpha_t^\top)^{-1}\alpha_t$ and $M_t(\epsilon) = (F_t^\top \tilde{R}_t^{-1}(\epsilon) F_t)^{-1} F_t^\top \tilde{R}_t^{-1}(\epsilon)$. Then it is simple derive the following directional

derivatives for each of the components in Equation (4.10).

$$\begin{aligned}
\left. \frac{d\tilde{R}_t^{-1}(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= -\tilde{R}_t^{-1}(0)C_tA_{t-1}XA_{t-1}^\top C_t^\top \tilde{R}_t^{-1}(0). \\
\left. \frac{dM_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= M_t(0)C_tA_{t-1}XA_{t-1}^\top C_t^\top M_t^\top(0)F_t^\top \tilde{R}_t^{-1}(0) - M_t(0)C_tA_{t-1}XA_{t-1}^\top C_t^\top \tilde{R}_t^{-1}(0). \\
\left. \frac{dK_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= (I - K_t(0)C_t)A_{t-1}XA_{t-1}^\top C_t^\top \tilde{R}_t^{-1}(0). \\
\left. \frac{d\tilde{W}_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= (I - K_t(0)C_t)G_{t-1} \left. \frac{dM_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} + \left. \frac{dK_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} (I - C_tG_{t-1}M_t(0)). \\
\left. \frac{d\tilde{F}_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= \left. \frac{d\tilde{W}_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} C_t. \\
\left. \frac{d\tilde{A}_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= - \left. \frac{d\tilde{W}_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0} C_tA_{t-1}.
\end{aligned}$$

Then the directional derivative of the Riccati map given in Equation (4.10) can be written in terms of $\Gamma_t = \left. \frac{d\tilde{W}_t(\epsilon)}{d\epsilon} \right|_{\epsilon=0}$ only:

$$\begin{aligned}
\left. \frac{d\rho(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= \Gamma_t \tilde{R}_t(0) \tilde{W}_t^\top(0) + \tilde{W}_t(0) \tilde{R}_t(0) \Gamma_t^\top - \Gamma_t C_t \Sigma_{t|t-1} - \Sigma_{t|t-1} C_t^\top \Gamma_t^\top \\
&\quad + (I - \tilde{W}_t(0)C_t)A_{t-1}XA_{t-1}^\top (I - C_t^\top \tilde{W}_t^\top(0)).
\end{aligned}$$

Expanding further,

$$\begin{aligned}
\left. \frac{d\rho(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= \Gamma_t (C_t \Sigma_t^* + S_t^{*\top}) + \Gamma_t C_t G_{t-1} M_t(0) \tilde{R}_t(0) K_t^\top(0) \\
&\quad - \Gamma_t C_t G_{t-1} M_t(0) \tilde{R}_t(0) M_t^\top(0) G_{t-1}^\top C_t^\top K_t^\top(0) + \Gamma_t \tilde{R}_t(0) M_t^\top(0) G_{t-1}^\top \\
&\quad + (\Sigma_t^* C_t^\top + S_t^*) \Gamma_t^\top + K_t(0) \tilde{R}_t(0) M_t^\top(0) G_{t-1}^\top C_t^\top \Gamma_t^\top \\
&\quad - K_t(0) C_t G_{t-1} M_t(0) \tilde{R}_t(0) M_t^\top(0) G_{t-1}^\top C_t^\top \Gamma_t^\top + G_{t-1} M_t(0) \tilde{R}_t(0) \Gamma_t^\top \\
&\quad - \Gamma_t C_t \Sigma_{t|t-1} - \Sigma_{t|t-1} C_t^\top \Gamma_t^\top + (I - \tilde{W}_t(0)C_t)A_{t-1}XA_{t-1}^\top (I - C_t^\top \tilde{W}_t^\top(0)). \quad (\text{A.1})
\end{aligned}$$

Noting that

$$C_t G_{t-1} M_t(0) \tilde{R}_t(0) M_t^\top(0) G_{t-1}^\top = \tilde{R}_t(0) M_t^\top(0) G_{t-1}^\top,$$

and

$$C_t G_{t-1} M_t(0) \tilde{R}_t(0) = \tilde{R}_t(0) M_t^\top(0) G_{t-1}^\top C_t^\top$$

simplifies Equation (A.1). Specifically, these relations allow one to observe that the second and third, fifth and sixth, fourth and ninth, and eight and tenth terms cancel one another

out. Next, by examining $(\Sigma_t^*(0)C_t^\top + S_t^*)$, we find

$$(\Sigma_t^*(0)C_t^\top + S_t^*) = \Sigma_{t|t-1}C_t^\top(I - M_t^\top(0)G_{t-1}^\top C_t^\top).$$

Equation (A.1) now reduces to

$$\begin{aligned} \left. \frac{d\psi(\epsilon)}{d\epsilon} \right|_{\epsilon=0} &= -\Gamma_t C_t G_{t-1} M_t(0) C_t \Sigma_{t|t-1} \\ &\quad - \Sigma_{t|t-1} C_t^\top M_t^\top(0) G_{t-1} C_t \Gamma_t^\top \\ &\quad + (I - \widetilde{W}_t(0)C_t)A_{t-1}X A_{t-1}^\top (I - C_t^\top \widetilde{W}_t^\top(0)) \end{aligned}$$

Recognising that due to the assumption in Equation (4.7),

$$C_t G_{t-1} M_t(0) C_t G_{t-1} M_t(0) C_t = C_t G_{t-1} M_t(0) C_t$$

makes clear that the directional derivative of ρ can be written in the form given in Lemma 4.2.

To prove the result for the t -horizon mapping, we proceed by induction. For $t = 0$, we have $\phi_0(\Sigma + \epsilon X) = \Sigma + \epsilon X$. It therefore follows immediately that $g_t(\Sigma, Q) = X$.

Assuming that the result holds for any $t \leq T - 1$, then:

$$\phi_t(\Sigma + \epsilon X) = \phi_t(\Sigma) + \left(\prod_{k=0}^{t-1} (\widetilde{A}_{t-k}) X \prod_{k=0}^{t-1} (\widetilde{A}_k)^\top \right) \epsilon + \mathcal{O}(\epsilon)$$

where $\mathcal{O}(\epsilon)/\epsilon \rightarrow 0$ as $\epsilon \rightarrow 0$.

Then for $t + 1$, we have

$$\begin{aligned} \phi_{t+1}(\Sigma + \epsilon X) &= \rho(\phi_t(\Sigma + \epsilon X)) \\ &= \rho \left(\phi_t(\Sigma) + \left(\prod_{k=0}^{t-1} (\widetilde{A}_{t-k}) X \prod_{k=0}^{t-1} (\widetilde{A}_k)^\top \right) \epsilon + \mathcal{O}(\epsilon) \right) \end{aligned}$$

Applying the result for the directional derivative of ρ in Lemma 4.2 completes the proof.

The result for the gain matrix chosen in Equation (4.9) follows from the above derivation, where α_t is as in Theorem 8 of [22].

A.3 Proof of Lemma 4.3

Lemma 4.3. *Suppose $\exists \beta < \infty$ such that $\Sigma_t \preceq \beta I \forall t \in \{0, \dots, T\}$, then we have*

$$\text{Tr}\{g_t^\sigma(\Sigma, X)\} \leq \beta \eta^k \text{Tr}\{\Sigma^{-1}X\},$$

where $\eta = \frac{\beta}{\beta + \lambda_Q} < 1$ and λ_Q is the minimum eigenvalue of $\tilde{F}_t Q_{t-1} \tilde{F}_t^\top \forall t \in \{0, \dots, T\}$.

$\forall t$, we have $\beta I \succeq \Sigma_t$ and $\lambda_Q I \preceq \tilde{F}_t Q_t \tilde{F}_t^\top$. Notice from Equation (4.10) that

$$\Sigma_{t+1} \succeq \tilde{A}_t \Sigma_t \tilde{A}_t^\top + \tilde{F}_t Q_t \tilde{F}_t^\top.$$

So

$$(\beta - \lambda_Q)I \succeq \tilde{A}_t \Sigma_t \tilde{A}_t^\top$$

and

$$\tilde{F}_t Q_t \tilde{F}_t^\top \succeq \lambda_Q I \succeq \frac{\lambda_Q}{\beta - \lambda_Q} \tilde{A}_t \Sigma_t \tilde{A}_t^\top \quad (\text{A.2})$$

$\forall t$. Now,

$$\begin{aligned} \Sigma_{t+1} - \left(1 - \frac{\lambda_Q}{\beta}\right)^{-1} \tilde{A}_t \Sigma_t \tilde{A}_t^\top &\succeq \tilde{A}_t \Sigma_t \tilde{A}_t^\top + \tilde{F}_t Q_t \tilde{F}_t^\top - \left(1 - \frac{\lambda_Q}{\beta}\right)^{-1} \tilde{A}_t \Sigma_t \tilde{A}_t^\top \\ &= \left(1 - \frac{\beta}{\beta - \lambda_Q}\right) \tilde{A}_t \Sigma_t \tilde{A}_t^\top + \tilde{F}_t Q_t \tilde{F}_t^\top \\ &= -\left(\frac{\lambda_Q}{\beta - \lambda_Q}\right) \tilde{A}_t \Sigma_t \tilde{A}_t^\top + \tilde{F}_t Q_t \tilde{F}_t^\top \\ &\succeq 0, \end{aligned}$$

where the last inequality comes from Equation (A.2). Then we can apply the Schur complement lemma for

$$0 \preceq \left(1 - \frac{\lambda_Q}{\beta}\right) S_t - \tilde{A}_t^\top S_{t+1} \tilde{A}_t$$

where $S_t = \Sigma_t^{-1}$. Now, using the same $\zeta_t^{(l)}$ and Lyapunov function as is used in [20], we follow the same working from this point to find the result.

A.4 Proof of Theorem 4.1

Theorem 4.1. *Let $\beta^* < \infty$ be the peak state estimation error of the optimal trajectory, i.e. $\Sigma_t^* \preceq \beta^* I \forall t \in \{1, \dots, T\}$. Then we have*

$$0 \leq J(\Sigma_T^{\epsilon, \delta}) - J(\Sigma_T^*) \leq (\zeta_T - 1) (J(\Sigma_T^*) - J(\lambda_Q I)) + \epsilon \left(\frac{n_y}{\lambda_Q} + \Delta_T \right), \quad (4.16)$$

where $\zeta_t := \prod_{\tau=1}^{t-1} \left(1 + \sum_{s=1}^{\tau} L_f^s L_m \delta \right) \geq 1$, $\Delta_T := \frac{n_y}{\lambda_Q^2} \beta^* \sum_{\tau=1}^{T-1} \frac{\zeta_T}{\zeta_\tau} \eta_*^{T-\tau}$, $\eta_* = \frac{\beta^*}{\beta^* + \lambda_Q} < 1$.

Lemma 7 in Appendix C of [20] holds the unknown input filter state Riccati map Equation (4.10) with small adjustment for time-varying process noise Q_t . Then, the proof of the bounds follows the steps of the proof of Theorem 4 in [20]. However, here we make use of the proven result in Lemma 4.3 and the fact that $\phi_{t-1-\tau}^{\sigma^*}(Q_\tau) \succeq \tilde{F}_{\tau+t-1} Q_{\tau+t-2} \tilde{F}_{\tau+t-1}^\top \forall t, \tau$ and $\tilde{F}_t Q_{t-1} \tilde{F}_t^\top \succeq \lambda_Q I \forall t$.

A.5 Proof of Lemma 4.4

Lemma 4.4. *The unknown input error covariance update map $\rho^d(\cdot)$ is monotone and concave.*

To prove Lemma 4.4, we require some preparatory results:

Lemma A.1. *Let $\alpha \in [0, 1]$ be a constant. Then $\forall \Sigma \in \mathcal{P}^+$, we have $\alpha \rho^d(\Sigma) \preceq \rho^d(\alpha \Sigma)$.*

Proof: Denote $\bar{R}_t(\Sigma) = C_t A_{t-1} \Sigma A_{t-1}^\top C_t^\top + \alpha^{-1} C_t Q_{t-1} C_t^\top + \alpha^{-1} R_t$. We have

$$\bar{R}_t(\Sigma) - \tilde{R}_t(\Sigma) = (\alpha^{-1} - 1)(C_t Q_{t-1} C_t^\top + R_t) \succeq 0,$$

when $\alpha \in (0, 1]$ and $R_t, \Sigma \in \mathcal{P}^+$. Thus, $\bar{R}_t(\Sigma) \succeq \tilde{R}_t(\Sigma)$. Since $X \mapsto X^{-1}$ is order reversing for any matrix X , we have $\tilde{R}_t^{-1}(\Sigma) \succeq \bar{R}_t^{-1}(\Sigma)$. Additionally, note that

$$\begin{aligned} \rho^d(\alpha \Sigma) &= (F_t' \tilde{R}_t^{-1}(\alpha \Sigma) F_t)^{-1} \\ &= \alpha (F_t' \bar{R}_t^{-1}(\Sigma) F_t)^{-1} \succeq \alpha (F_t' \tilde{R}_t^{-1}(\Sigma) F_t)^{-1} = \alpha \rho^d(\Sigma). \end{aligned}$$

For $\alpha = 0$, we have $\alpha\rho^d(\Sigma) = 0 \preceq \rho^d(\alpha\Sigma)$.

Lemma A.2. $f(\Sigma) = F_t^\top \tilde{R}_t^{-1}(\Sigma) F_t$ is a convex function of $\Sigma \in \mathcal{P}^+$.

Proof: We note that $\tilde{R}_t(\cdot)$ is monotone. Then, by Corollary V.2.6 in [198], $\tilde{R}_t^{-1}(\cdot)$ is operator convex. For $\Sigma_1, \Sigma_2 \in \mathcal{P}^+$ and $\alpha \in [0, 1]$, let $\chi = \alpha\Sigma_1 + (1 - \alpha)\Sigma_2$. We can prove that

$$F_t^\top (\alpha \tilde{R}_t^{-1}(\Sigma_1) + (1 - \alpha) \tilde{R}_t^{-1}(\Sigma_2) - \tilde{R}_t^{-1}(\chi)) F_t \succeq 0.$$

So $f(\Sigma)$ is also operator convex.

Proof of Lemma 4.4: We note that $\tilde{R}_t(\cdot)$ is monotone. Then for any $\Sigma_1, \Sigma_2 \in \mathcal{P}^+$ with $\Sigma_1 \preceq \Sigma_2$, we have $\tilde{R}_t(\Sigma_1) \preceq \tilde{R}_t(\Sigma_2)$. As matrix multiplication is order preserving, and matrix inversion is order reversing, it immediately follows that $\rho^d(\Sigma_1) \preceq \rho^d(\Sigma_2)$. Hence, monotonicity is proved.

We next prove concavity. For $\forall \alpha \in [0, 1]$ and $\forall \Sigma_1, \Sigma_2 \in \mathcal{P}^+$, let $\chi = \alpha\Sigma_1 + (1 - \alpha)\Sigma_2$. Then, from Lemma A.2, and since $\alpha\tilde{R}_t^{-1}(\Sigma) \preceq \tilde{R}_t^{-1}(\alpha\Sigma)$ we have $F_t^\top \tilde{R}_t^{-1}(\chi) F_t \preceq F_t^\top \tilde{R}_t^{-1}(\alpha\Sigma_1) F_t + F_t^\top \tilde{R}_t^{-1}((1 - \alpha)\Sigma_2) F_t$. Inverting this expression, utilising Lemma A.1, and remembering that $X \mapsto X^{-1}$ is a convex operation [198] gives

$$\begin{aligned} & \rho^d(\chi) - \alpha\rho^d(\Sigma_1) - (1 - \alpha)\rho^d(\Sigma_2) \\ & \succeq [F_t^\top \tilde{R}_t^{-1}(\chi) F_t]^{-1} - [F_t^\top \tilde{R}_t^{-1}(\alpha\Sigma_1) F_t + F_t^\top \tilde{R}_t^{-1}((1 - \alpha)\Sigma_2) F_t]^{-1} \\ & \succeq 0, \end{aligned}$$

thus proving concavity.

A.6 Proof of Lemma 4.5

Lemma 4.5. The directional derivative of $\phi_t^{d\sigma}$ at $\Sigma \in \mathcal{P}^+$ in the direction $X \in \mathcal{P}^+$ is given by

$$\begin{aligned} g_{t-1}^{d\sigma}(\Sigma, X) &= \left. \frac{d}{d\epsilon} \phi_{t-1}^d(\Sigma + \epsilon X) \right|_{\epsilon=0} \\ &= M_t^* C_t A_{t-1} g_{t-1}^\sigma(\Sigma, X) A_{t-1}^\top C_t^\top M_t^{*\top}, \end{aligned}$$

where $g_{t-1}^\sigma(\Sigma, X)$ is the directional derivative of the state t -horizon update map.

Denoting $\tilde{R}_t^{-1}(\phi_{t-1}^\sigma(\Sigma)) = \tilde{R}_t^{-1}$, it is simple to show that

$$\begin{aligned} \left. \frac{d}{d\epsilon} \phi_{t-1}^{d\sigma}(\Sigma + \epsilon X) \right|_{\epsilon=0} &= -\phi_{t-1}^{d\sigma}(\Sigma) F_t^\top \left. \frac{d}{d\epsilon} \tilde{R}_t^{-1}(\phi_{t-1}^\sigma(\Sigma + \epsilon X)) \right|_{\epsilon=0} F_t \phi_{t-1}^{d\sigma}(\Sigma), \\ \left. \frac{d}{d\epsilon} \tilde{R}_t^{-1}(\phi_{t-1}^\sigma(\Sigma + \epsilon X)) \right|_{\epsilon=0} &= -\tilde{R}_t^{-1} \left. \frac{d}{d\epsilon} \tilde{R}_t(\phi_{t-1}^\sigma(\Sigma + \epsilon X)) \right|_{\epsilon=0} \tilde{R}_t^{-1}. \end{aligned}$$

Putting these together and simplifying gives the result.

A.7 Proof of Lemma 4.6

Lemma 4.6. *Suppose $\exists \beta^d < \infty$ such that $\Sigma_t^d \preceq \beta^d I \forall t \in \{1, \dots, T\}$. Then*

$$\text{Tr}\{g_{t-1}^{d\sigma}(\Sigma, I)\} \leq (n_d)^2 (\beta^d)^2 \bar{\lambda}_{\tilde{G}} \text{Tr}\{g_{t-1}^\sigma(\Sigma, I)\},$$

where $\bar{\lambda}_{\tilde{G}}$ is the maximum eigenvalue of $G_{t-1}^\top H_t A_{t-1} A_{t-1}^\top H_t G_{t-1} \in \mathbb{R}^{n_d \times n_d}$.

Looking into the square of the Frobenius norm $\|\cdot\| = \sqrt{\text{Tr}\{\cdot\}}$ of $g_{t-1}^d(\Sigma, I)$:

$$\text{Tr}\{g_{t-1}^d(\Sigma, I)\} = \text{Tr}\{M_t C_t A_{t-1} g_{t-1}(\Sigma, I) A_{t-1}^\top C_t^\top M_t^\top\}$$

By the cyclical property of trace operator, and submultiplicity of the Frobenius norm:

$$\begin{aligned} \text{Tr}\{g_{t-1}^d(\Sigma, I)\} &= \text{Tr}\{\prod_{k=0}^{t-2} \tilde{A}_k^\top A_{t-1}^\top C_t^\top M_t^\top M_t C_t A_{t-1} \prod_{k=0}^{t-2} \tilde{A}_{t-2-k}\} \\ &= \|\|M_t C_t A_{t-1} \prod_{k=0}^{t-2} \tilde{A}_{t-2-k}\|^2 \\ &\leq \|\|M_t C_t A_{t-1}\|^2 \|\prod_{k=0}^{t-2} \tilde{A}_{t-2-k}\|^2 \\ &= \text{Tr}\{A_{t-1}^\top C_t^\top M_t^\top M_t C_t A_{t-1}\} \text{Tr}\{g_{t-1}(\Sigma, I)\} \end{aligned}$$

Since $\text{Tr}\{g_{t-1}(\Sigma, I)\}$ is given in Lemma 4.3, we need only analyse $\text{Tr}\{A_{t-1}^\top C_t^\top M_t^\top M_t C_t A_{t-1}\}$.

$$\begin{aligned} \text{Tr}\{A_{t-1}^\top C_t^\top M_t^\top M_t C_t A_{t-1}\} &= \|M_t C_t A_{t-1}\|^2 \\ &= \|\phi_{t-1}^d(\Sigma) F_t^\top \tilde{R}_t^{-1}(\phi_{t-1}(\Sigma)) C_t A_{t-1}\|^2 \\ &\leq \|\phi_{t-1}^d(\Sigma)\|^2 \|F_t^\top \tilde{R}_t^{-1}(\phi_{t-1}(\Sigma)) C_t A_{t-1}\|^2 \\ &\leq \|\beta^{d*} I\|^2 \|F_t^\top \tilde{R}_t^{-1}(\phi_{t-1}(\Sigma)) C_t A_{t-1}\|^2 \\ &= n_d (\beta^d)^2 \|F_t^\top \tilde{R}_t^{-1}(\phi_{t-1}(\Sigma)) C_t A_{t-1}\|^2, \end{aligned}$$

where $\beta^d > 0$ is some constant such that the unknown input estimate covariance is bounded from above. That is, $\Sigma_{t-1}^d \preceq \beta^d I \forall t$. Now, turning to $\|F_t^\top \tilde{R}_t^{-1}(\phi_{t-1}(\Sigma)) C_t A_{t-1}\|^2$,

$$\begin{aligned} \|G_{t-1}^\top C_t^\top \tilde{R}_t^{-1}(\phi_{t-1}(\Sigma)) C_t A_{t-1}\|^2 &\leq \|G_{t-1}^\top C_t^\top R_t^{-1} C_t A_{t-1}\|^2 \\ &= \|G_{t-1}^\top H_t A_{t-1}\|^2 \\ &= \text{Tr}\{G_{t-1}^\top H_t A_{t-1} A_{t-1}^\top H_t^\top G_{t-1}\} \\ &\leq n_d \bar{\lambda}_{\tilde{G}} \end{aligned}$$

where $H_t = C_t^\top R_t^{-1} C_t$ and $\bar{\lambda}_{\tilde{G}}$ is the maximum eigenvalue of $G_{t-1}^\top H_t A_{t-1} A_{t-1}^\top H_t^\top G_{t-1}$. Putting the above together gives the result.

A.8 Proof of Theorem 4.2

We again need to prove some preparatory results.

Lemma A.3. *There exists a real constant $L_m \geq 0$ such that $\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$:*

$$\begin{aligned} H_t(\mathbf{x}_1) &\preceq (1 + L_m d_{\mathcal{X}}(\mathbf{x}_1, \mathbf{x}_2)) H_t(\mathbf{x}_2), \\ H_t(\mathbf{x}_2) &\preceq (1 + L_m d_{\mathcal{X}}(\mathbf{x}_1, \mathbf{x}_2)) H_t(\mathbf{x}_1) \end{aligned}$$

where $H_t(\mathbf{x}) = C_t^\top(\mathbf{x}) R_t^{-1}(\mathbf{x}) C_t(\mathbf{x})$.

Proof: Consider any two nodes $(\mathbf{x}_{t-1}^1, \Sigma_{t-1})$, $(\mathbf{x}_{t-1}^2, \Sigma_{t-1})$. Then, applying control $\mathbf{u} \in \mathcal{U}$ to each node we have $\rho_{\mathbf{x}_t^1}(\Sigma_{t-1}) \succeq \gamma \rho_{\mathbf{x}_t^2}(\Sigma_{t-1})$ from Assumption 3.3. Hence,

$$\gamma^{-1} \rho_{\mathbf{x}_t^2}^{-1}(\Sigma_{t-1}) \succeq \rho_{\mathbf{x}_t^1}^{-1}(\Sigma_{t-1}) \succeq \rho_{\mathbf{x}_t^1}^{-1}(\gamma^{-1} \Sigma_{t-1}), \quad (\text{A.3})$$

where the last inequality follows from monotonicity of ρ and $\gamma^{-1} > 1$. Denote

$$\bar{\Sigma}_{t+1}^{-1}(\Sigma_t) = A_t^{-T} \Sigma_t^{-1} A_t^{-1} - A_t^{-T} \Sigma_t^{-1} A_t^{-1} (A_t^{-T} \Sigma_t^{-1} A_t^{-1} + Q_t^{-1})^{-1} A_t^{-T} \Sigma_t^{-1} A_t^{-1},$$

then it is simple to show

$$\gamma \bar{\Sigma}_{t+1}^{-1} \preceq (\overline{\gamma^{-1} \Sigma_{t+1}})^{-1}. \quad (\text{A.4})$$

Now, the information form covariance update map for the unknown input filter is $\rho^{-1}(\cdot)$ [102]:

$$\Sigma_t^{-1} = \bar{\Sigma}_t^{-1} + H_t - \bar{\Sigma}_t^{-1} G_{t-1} (G_{t-1}^\top \bar{\Sigma}_t^{-1} G_{t-1})^{-1} G_{t-1}^\top \bar{\Sigma}_t^{-1}.$$

Using Equations (A.3) and (A.4) and denoting $H_t(\mathbf{x}_t^n) = H_t^n$ for $n = 1, 2$, it follows that

$$\begin{aligned} \gamma^{-1} (\bar{\Sigma}_t^{-1} + H_t^2 - \bar{\Sigma}_t^{-1} G_{t-1} (G_{t-1}^\top \bar{\Sigma}_t^{-1} G_{t-1})^{-1} G_{t-1}^\top \bar{\Sigma}_t^{-1}) \\ \succeq \gamma^{-1} \bar{\Sigma}_t^{-1} + H_t^1 - \gamma^{-1} \bar{\Sigma}_t^{-1} G_{t-1} (G_{t-1}^\top \bar{\Sigma}_t^{-1} G_{t-1})^{-1} G_{t-1}^\top \bar{\Sigma}_t^{-1}. \end{aligned}$$

Reducing gives the desired result $H_t^1 \preceq (1 + L_m d_{\mathcal{X}}(\mathbf{x}_t^1, \mathbf{x}_t^2)) H_t^2$. Following identical working for $H_t(\mathbf{x}_t^2)$ completes the proof.

Lemma A.4. *Suppose $(\mathbf{x}_{t-1}^1, \Sigma_{t-1}, \Sigma_{t-2}^d)$, $(\mathbf{x}_{t-1}^2, \Sigma_{t-1}, \Sigma_{t-2}^d)$ are two sensing locations and corresponding beliefs with $d(\mathbf{x}_{t-1}^1, \mathbf{x}_{t-1}^2) \leq \delta$. Let $\Sigma_{t-1}^{d,1}$, $\Sigma_{t-1}^{d,2}$ be the input estimation error covariances after updating both nodes under the control $\mathbf{u} \in \mathcal{U}$. Then*

$$\Sigma_{t-1}^{d,1} \succeq \gamma \Sigma_{t-1}^{d,2}, \quad \Sigma_{t-1}^{d,2} \succeq \gamma \Sigma_{t-1}^{d,1}$$

$\forall t \in \{1, T\}$.

Proof: Denote $R_t(x_i) = R_t^i$ for $i = 1, 2$, and consider the inverse of the update map applied to node $(\mathbf{x}_{t-1}^1, \Sigma_{t-1}, \Sigma_{t-2}^d)$:

$$(\rho_{\mathbf{x}^1}^d(\Sigma_{t-1}))^{-1} = F_t^\top \tilde{R}_t^{-1}(\Sigma_{t-1}) F_t = F_t^\top \left(C_t A_{t-1} \Sigma_{t-1} A_{t-1}^\top C_t^\top + C_t Q_{t-1} C_t^\top + R_t^1 \right)^{-1} F_t.$$

By applying the matrix inversion lemma and Lemma A.3, we get $(\Sigma_{t-1}^{d,1})^{-1} \preceq \gamma^{-1} (\rho_{\mathbf{x}^2}^d(\gamma^{-1} \Sigma_{t-1}))^{-1}$.

Then, taking the inverse and applying monotonicity of the update map gives the result.

Following the same reasoning for updating $(\mathbf{x}_{t-1}^2, \Sigma_{t-1}, \Sigma_{t-2}^d)$ completes the proof.

Proof of Theorem 4.2:

Theorem 4.2. *Let $\beta^* < \infty$, $\beta^{d*} < \infty$ be the peak state and input estimation errors of the optimal trajectory respectively. That is, $\Sigma_t^* \preceq \beta^* I$ and $\Sigma_{t-1}^{d*} \preceq \beta^{d*} I \forall t \in \{1, \dots, T\}$. Then*

$$0 \leq J(\Sigma_{T-1}^{d,(\epsilon,\delta)}) - J(\Sigma_{T-1}^{d*}) \leq (\zeta_T - 1) \left(J(\Sigma_{T-1}^{d*}) + J(\gamma^{d*} I) - J(\bar{\lambda}_H^{-1} I) \right) + \epsilon(\Delta_T^d),$$

where $\Delta_T^d := (\gamma^{d*})^{-1} (n_d)^2 \bar{\lambda}_H \bar{\lambda}_{\tilde{G}} (\beta^{d*})^2 \lambda_Q \Delta_T$, $\gamma^{d*} = (1 + L_m d(\mathbf{x}_T^*, \mathbf{x}_T^{d*}))^{-1}$ and $\bar{\lambda}_H$ is the maximum eigenvalue of $G_{T-1}^\top H_T G_{T-1}$.

Applying $\rho_{\mathbf{u}_{T-1}^*}^d(\cdot)$ to Appendix C Lemma 7 of [20], where \mathbf{u}_{T-1}^* is the state-optimized control found by the algorithm, rather than an input-optimized control \mathbf{u}_{T-1}^{d*} . Noting again that $\phi_{t-1-\tau}^{\sigma_\tau^*}(Q_\tau) \succeq \tilde{F}_{\tau+t-1} Q_{\tau+t-2} \tilde{F}_{\tau+t-1}^\top \forall t$, and that $\sum_{\tau=1}^{T-1} \Gamma_\tau (1 - \gamma_\tau) = 1 - \Gamma_{T-1}$, by concavity of ρ^d ,

$$\begin{aligned} \rho_{\mathbf{u}_{T-1}^*}^d(\Sigma_{N-1}^*) + \epsilon g_1^{d\sigma_{T-1}^*}(\Sigma_{T-1}^*, \sum_{\tau=1}^{T-2} \Gamma_\tau g_{T-1-\tau}^{\sigma_\tau^*}(\Sigma_\tau^*, I) + \Gamma_{T-1} I) \\ \succeq \Gamma_{T-1} \sum_{i=1}^K \alpha_i \rho_{\mathbf{u}_{T-1}^*}^d(\Sigma_{T-1}^{i*}) + (1 - \Gamma_{T-1}) \rho_{\mathbf{u}_{T-1}^*}^d(\lambda_Q I). \end{aligned}$$

Using Lemma A.4 with $\gamma^{d*} = (1 + L_m d(\mathbf{x}_T^*, \mathbf{x}_T^{d*}))^{-1}$, and $\rho_{\mathbf{u}_{T-1}^*}^d(\lambda_Q I) \succeq (G_{T-1}^\top H_T G_{T-1})^{-1} \succeq \bar{\lambda}_H^{-1} I$, we have

$$\begin{aligned} \gamma^{d*} \Sigma_{T-1}^{d*} + M_T C_T A_{T-1} \sum_{\tau=1}^{T-1} \Gamma_\tau g_{T-\tau}^{\sigma_\tau^*}(\Sigma_\tau^*, I) A_{T-1}^\top C_T^\top M_T^\top \\ \succeq \Gamma_{T-1} \sum_{i=1}^K \alpha_i \gamma^{d*} \Sigma_{T-1}^{di} + (1 - \Gamma_{T-1}) \bar{\lambda}_H^{-1} I. \end{aligned}$$

The proof then follows using monotonicity and concavity of $J(\cdot)$, and Lemma 4.6 after applying $J(\cdot)$ to the above result.

Appendix B

Proofs of results in Chapter 6

B.1 Proof of Theorem 6.1

Theorem 6.1. *Suppose $s \in \mathcal{H}_k$ with arbitrary positive definite kernel k . For bounded measurement noise $\epsilon^2 < \sigma_\epsilon^2 < \infty$, we have for any measurement set \mathbf{Z} ,*

$$|s(\mathbf{x}) - \mathbb{E}[\hat{s}(\mathbf{x} \mid \mathbf{Z})]| \leq \|s\|_{\mathcal{H}_k} P_{\mathbf{X}}(\mathbf{x}) + \sqrt{\sigma_\epsilon^2 T \Lambda_k^2(\mathbf{x})}, \quad (6.12)$$

where $\mathbb{E}[\hat{s}(\mathbf{x} \mid \mathbf{Z})]$ is the mean of the posterior GP, $P_{\mathbf{X}}(\mathbf{x}) = \sqrt{\frac{\det K_{\mathbf{X} \cup \{\mathbf{x}\}}}{\det K_{\mathbf{X}}}}$ is called the power function of \mathbf{X} and $\Lambda_k(\mathbf{x}) = \|\mathbf{K}_{\mathbf{X}}^{-1} \mathbf{k}_{\mathbf{X}}(\mathbf{x})\|$.

Letting $[\mathbf{S}]_t = s(\mathbf{x}_t)$, $t \in \{1, \dots, T\}$ be the vector of true function evaluations at \mathbf{X} , and similarly $[\boldsymbol{\epsilon}_T]_t = \epsilon_t$, $t \in \{1, \dots, T\}$ the noise outcomes, we have:

$$\begin{aligned} |s(\mathbf{x}) - \mathbb{E}[\hat{s}(\mathbf{x} \mid \mathbf{Z})]| &\leq |s(\mathbf{x}) - \mathbf{k}_{\mathbf{X}}^\top(\mathbf{x}) \mathbf{K}_{\mathbf{X}}^{-1} \mathbf{S}| + |\mathbf{k}_{\mathbf{X}}^\top(\mathbf{x}) \mathbf{K}_{\mathbf{X}}^{-1} \mathbf{S} - \mathbf{k}_{\mathbf{X}}^\top(\mathbf{x}) \mathbf{K}_{\mathbf{X}}^{-1} [\mathbf{S} + \boldsymbol{\epsilon}_T]| \\ &= |\langle s, k(\cdot, \mathbf{x}) - \mathbf{k}_{\mathbf{X}}^\top(\mathbf{x}) \mathbf{K}_{\mathbf{X}}^{-1} \mathbf{k}_{\mathbf{X}}(\cdot) \rangle_{\mathcal{H}_k}| + |\mathbf{k}_{\mathbf{X}}^\top(\mathbf{x}) \mathbf{K}_{\mathbf{X}}^{-1} \boldsymbol{\epsilon}_T| \\ &\leq \|s\|_{\mathcal{H}_k} P_{\mathbf{X}}(\mathbf{x}) + \|\mathbf{k}_{\mathbf{X}}^\top(\mathbf{x}) \mathbf{K}_{\mathbf{X}}^{-1} \boldsymbol{\epsilon}_T\| \\ &\leq \|s\|_{\mathcal{H}_k} P_{\mathbf{X}}(\mathbf{x}) + \sqrt{\sigma_\epsilon^2 T} \|\mathbf{k}_{\mathbf{X}}^\top(\mathbf{x}) \mathbf{K}_{\mathbf{X}}^{-1}\| \end{aligned}$$

where the final inequality follows from $|\cdot| = \sqrt{\langle \cdot, \cdot \rangle}$, Cauchy-Schwarz inequality and the assumption $\epsilon_t^2 < \sigma_\epsilon^2 \forall t \in \{1, \dots, T\}$.

B.2 Proof of Theorem 6.2

Theorem 6.2. *Suppose a kernel k satisfies the CI assumption (Equation (6.5)). Then, the power function $P_{\mathbf{X}}(\mathbf{x})$ satisfies:*

$$P_{\mathbf{M}}(\mathbf{x}) \leq P_{\mathbf{X}}(\mathbf{x}) \leq P_{\mathbf{M}}(\mathbf{x}) \exp H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}), \quad (6.13)$$

where $P_{\mathbf{M}}(\mathbf{x}) = \sqrt{\frac{\det K_{\mathbf{M} \cup \{\mathbf{x}\}}}{\det K_{\mathbf{M}}}}$.

Moreover, assuming $s \in \mathcal{H}_k$, the deterministic error can be further bounded:

$$|s(\mathbf{x}) - \mathbb{E}[\hat{s}(\mathbf{x} | \mathbf{Z})]| \leq \|s\|_{\mathcal{H}_k} P_{\mathbf{M}}(\mathbf{x}) \exp H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}) + \sqrt{\sigma_{\epsilon}^2 T \Lambda_k^2(\mathbf{x})}. \quad (6.14)$$

We exploit the fact that the power function $P_{\mathbf{X}}(\mathbf{x})$ can be linked to conditional entropy as:

$$H(s(\mathbf{x}) | \mathbf{Z}) = \frac{1}{2} (\log P_{\mathbf{X}}(\mathbf{x}) + \log(2\pi e)).$$

To bound $H(s(\mathbf{x}) | \mathbf{Z})$, we expand the joint entropy $H(\mathbf{y}_{\mathbf{M}}, \mathbf{Z}, s(\mathbf{x}))$ according to the chain rule:

$$\begin{aligned} H(\mathbf{y}_{\mathbf{M}}, \mathbf{Z}, s(\mathbf{x})) &= H(\mathbf{y}_{\mathbf{M}} | s(\mathbf{x}), \mathbf{Z}) + H(s(\mathbf{x}) | \mathbf{Z}) + H(\mathbf{Z}) \\ &= H(\mathbf{Z} | \mathbf{y}_{\mathbf{M}}) + H(s(\mathbf{x}) | \mathbf{y}_{\mathbf{M}}) + H(\mathbf{y}_{\mathbf{M}}), \end{aligned}$$

where the second equivalence follows from the CI condition and symmetry of conditional entropy. Rearranging yields:

$$\begin{aligned} H(s(\mathbf{x}) | \mathbf{Z}) &= H(\mathbf{Z} | \mathbf{y}_{\mathbf{M}}) + H(s(\mathbf{x}) | \mathbf{y}_{\mathbf{M}}) + H(\mathbf{y}_{\mathbf{M}}) - H(\mathbf{Z}) - H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}, s(\mathbf{x})) \\ &= H(s(\mathbf{x}) | \mathbf{y}_{\mathbf{M}}) + H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}) - H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}, s(\mathbf{x})) \\ &= H(s(\mathbf{x}) | \mathbf{y}_{\mathbf{M}}) + \mathcal{I}(\mathbf{y}_{\mathbf{M}}, s(\mathbf{x}) | \mathbf{Z}), \end{aligned}$$

where the final equivalences follow from the chain rule property $H(\mathbf{Z} | \mathbf{y}_{\mathbf{M}}) - H(\mathbf{Z}) = H(\mathbf{y}_{\mathbf{M}} | \mathbf{Z}) - H(\mathbf{y}_{\mathbf{M}})$ and the definition of mutual information \mathcal{I} respectively. From the

above we see:

$$\begin{aligned} H(s(\mathbf{x})|\mathbf{y}_M) &\leq H(s(\mathbf{x})|\mathbf{Z}) \\ &= H(s(\mathbf{x})|\mathbf{y}_M) + \mathcal{I}(\mathbf{y}_M, s(\mathbf{x}) | \mathbf{Z}) \\ &\leq H(s(\mathbf{x})|\mathbf{y}_M) + H(\mathbf{y}_M|\mathbf{Z}). \end{aligned} \tag{B.1}$$

the claimed bound is recovered by taking the exponential of (B.1).

Bibliography

- [1] R. Bajcsy, “Active perception,” *Proceedings of the IEEE*, vol. 76, no. 8, pp. 966–1005, 1988.
- [2] S. Chen, Y. Li, and N. M. Kwok, “Active vision in robotic systems: A survey of recent developments,” *The International Journal of Robotics Research*, vol. 30, no. 11, pp. 1343–1377, 2011.
- [3] R. Bajcsy, Y. Aloimonos, and J. K. Tsotsos, “Revisiting active perception,” *Autonomous Robots*, vol. 42, no. 2, pp. 177–196, 2018.
- [4] A. Krause and C. Guestrin, “Submodularity and its applications in optimized information gathering,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 4, 2011.
- [5] R. Francis, T. Estlin, G. Doran, S. Johnstone, D. Gaines, V. Verma, M. Burl, J. Frydenvang, S. Montaña, R. C. Wiens, S. Schaffer, O. Gasnault, L. DeFlores, D. Blaney, and B. Bornstein, “AEGIS autonomous targeting for ChemCam on Mars Science Laboratory: Deployment and results of initial science team use,” *Science Robotics*, vol. 2, no. 7, p. eaan4582, 2017.
- [6] S. Bai, T. Shan, F. Chen, L. Liu, and B. Englot, “Information-Driven Path Planning,” *Current Robotics Reports*, vol. 2, no. 2, pp. 177–188, 2021.
- [7] N. D. Wallace, H. Kong, A. J. Hill, and S. Sukkarieh, “Energy Aware Mission Planning for WMRs on Uneven Terrains,” *IFAC Conference on Sensing, Control and Automation Technologies for Agriculture*, vol. 52, no. 30, pp. 149–154, 2019.

- [8] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, “Progress and prospects of the human–robot collaboration,” *Autonomous Robots*, vol. 42, no. 5, pp. 957–975, 2018.
- [9] D. Song, N. Kyriazis, I. Oikonomidis, C. Papazov, A. Argyros, D. Burschka, and D. Kragic, “Predicting human intention in visual observations of hand/object interactions,” in *IEEE International Conference on Robotics and Automation*, pp. 1608–1615, 2013.
- [10] L. Bascetta, G. Ferretti, P. Rocco, H. Ardo, H. Bruyninckx, E. Demeester, and E. Di Lello, “Towards safe human-robot interaction in robotic cells: An approach based on visual tracking and intention estimation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2971–2978, 2011.
- [11] J. A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, and J. A. Castellanos, “A survey on active simultaneous localization and mapping: State of the art and new frontiers,” *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 1686–1705, 2023.
- [12] “Dual-arm yumi collaborative robot.” <https://www07.abb.com/images/librariesprovider89/default-album/robots/yumi/d34r4217.jpg?sfvrsn=1>. [Accessed 22-03-2024].
- [13] “JPL Robotics: Surface System Software and Rover Navigation.” <https://www-robotics.jpl.nasa.gov/what-we-do/flight-projects/mars-science-laboratory/surface-system-software-and-rover-navigation/>. [Accessed 22-03-2024].
- [14] H. Kurniawati, “Partially Observable Markov Decision Processes and Robotics,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, no. 1, pp. 253–277, 2022.
- [15] R. Platt Jr, R. Tedrake, L. P. Kaelbling, and T. Lozano-Perez, “Belief space planning assuming maximum likelihood observations,” in *Robotics: Science and Systems*, vol. 2, 2010.

-
- [16] L. P. Kaelbling and T. Lozano-Pérez, “Integrated task and motion planning in belief space,” *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1194–1227, 2013.
- [17] A. Krause, A. Singh, and C. Guestrin, “Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies,” *Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.
- [18] M. Carreira-Perpinan, “Mode-finding for mixtures of Gaussian distributions,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1318–1323, 2000.
- [19] D. Bertsekas, *Dynamic programming and optimal control: Volume I*, vol. 4. Athena scientific, 2012.
- [20] N. Atanasov, J. L. Ny, K. Daniilidis, and G. J. Pappas, “Information acquisition with sensing robots: Algorithms and error bounds,” in *IEEE International Conference on Robotics and Automation*, pp. 6447–6454, 2014.
- [21] S. M. LaValle, “Sensing and Filtering: A Fresh Perspective Based on Preimages and Information Spaces,” *Foundations and Trends in Robotics*, vol. 1, no. 4, pp. 253–372, 2010.
- [22] S. Gillijns and B. De Moor, “Unbiased minimum-variance input and state estimation for linear discrete-time systems,” *Automatica*, vol. 43, no. 1, pp. 111–116, 2007.
- [23] B. Yamauchi, “A frontier-based approach for autonomous exploration,” in *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 146–151, 1997.
- [24] H. H. González-Banos and J.-C. Latombe, “Navigation strategies for exploring indoor environments,” *The International Journal of Robotics Research*, vol. 21, no. 10-11, pp. 829–848, 2002.
- [25] C. Stachniss, D. Hahnel, and W. Burgard, “Exploration with active loop-closing for FastSLAM,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1505–1510, 2004.

-
- [26] W. Gao, M. Booker, A. Adiwahono, M. Yuan, J. Wang, and Y. W. Yun, “An improved frontier-based approach for autonomous exploration,” in *International Conference on Control, Automation, Robotics and Vision*, pp. 292–297, 2018.
- [27] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, “Receding horizon path planning for 3D exploration and surface inspection,” *Autonomous Robots*, vol. 42, no. 2, pp. 291–306, 2018.
- [28] M. Juliá, A. Gil, and O. Reinoso, “A comparison of path planning strategies for autonomous exploration and mapping of unknown environments,” *Autonomous Robots*, vol. 33, no. 4, pp. 427–444, 2012.
- [29] A. Visser and B. A. Slamet, “Including communication success in the estimation of information gain for multi-robot exploration,” in *International symposium on modeling and optimization in mobile, ad hoc, and wireless networks and workshops*, pp. 680–687, 2008.
- [30] N. Basilico and F. Amigoni, “Exploration strategies based on multi-criteria decision making for searching environments in rescue operations,” *Autonomous Robots*, vol. 31, no. 4, pp. 401–417, 2011.
- [31] F. Giuliani, A. Castellini, R. Berra, A. D. Bue, A. Farinelli, M. Cristani, F. Setti, and Y. Wang, “POMP++: Pomcp-based Active Visual Search in unknown indoor environments,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1523–1530, 2021.
- [32] M. A. Batalin and G. S. Sukhatme, “Coverage, Exploration and Deployment by a Mobile Robot and Communication Network,” *Telecommunication Systems*, vol. 26, no. 2-4, pp. 181–196, 2004.
- [33] E. U. Acar, H. Choset, A. A. Rizzi, P. N. Atkar, and D. Hull, “Morse Decompositions for Coverage Tasks,” *The International Journal of Robotics Research*, vol. 21, pp. 331–344, Apr. 2002.
- [34] H. Choset, “Coverage of Known Spaces: The Boustrophedon Cellular Decomposition,” *Autonomous Robots*, vol. 9, pp. 247–253, Dec. 2000.

-
- [35] G. Mathew and I. Mezić, “Metrics for ergodicity and design of ergodic dynamics for multi-agent systems,” *Physica D: Nonlinear Phenomena*, vol. 240, no. 4, pp. 432–442, 2011.
- [36] L. Dressel and M. J. Kochenderfer, “On the optimality of ergodic trajectories for information gathering tasks,” in *2018 Annual American Control Conference (ACC)*, pp. 1855–1861, 2018.
- [37] M. Sun, A. Gaggar, P. Trautman, and T. Murphey, “Fast ergodic search with kernel functions,” *arXiv preprint arXiv:2403.01536*, 2024.
- [38] D. Lee, C. Lerch, F. Ramos, and I. Abraham, “Stein variational ergodic search,” *Robotics: Science and Systems*, 2024.
- [39] F. Amigoni and V. Caglioti, “An information-based exploration strategy for environment mapping with mobile robots,” *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 684–699, 2010.
- [40] M. Lauri and R. Ritala, “Planning for robotic exploration based on forward simulation,” *Robotics and Autonomous Systems*, vol. 83, pp. 15–31, 2016.
- [41] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, “An analysis of approximations for maximizing submodular set functions—I,” *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.
- [42] G. A. Hollinger and G. S. Sukhatme, “Sampling-based robotic information gathering algorithms,” *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1271–1287, 2014.
- [43] G. Best, O. Cliff, T. Patten, R. R. Mettu, and R. Fitch, “Dec-mcts: Decentralized planning for multi-robot active perception,” *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 316–337, 2019.
- [44] R. Valencia, J. Valls Miro, G. Dissanayake, and J. Andrade-Cetto, “Active Pose SLAM,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1885–1891, 2012.

-
- [45] L. Carlone, J. Du, M. Kaouk Ng, B. Bona, and M. Indri, “Active SLAM and Exploration with Particle Filters Using Kullback-Leibler Divergence,” *Journal of Intelligent & Robotic Systems*, vol. 75, no. 2, pp. 291–311, 2014.
- [46] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, “Planning and acting in partially observable stochastic domains,” *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.
- [47] L. Carlone, A. Censi, and F. Dellaert, “Selecting good measurements via l1 relaxation: A convex approach for robust estimation over graphs,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2667–2674, 2014.
- [48] J. M. Walls, S. M. Chaves, E. Galceran, and R. M. Eustice, “Belief space planning for underwater cooperative localization,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2264–2271, 2015.
- [49] N. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas, “Decentralized active information acquisition: Theory and application to multi-robot SLAM,” in *IEEE International Conference on Robotics and Automation*, pp. 4775–4782, 2015.
- [50] V. Indelman, L. Carlone, and F. Dellaert, “Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments,” *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 849–882, 2015.
- [51] S. M. Chaves, J. M. Walls, E. Galceran, and R. M. Eustice, “Risk aversion in belief-space planning under measurement acquisition uncertainty,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2079–2086, 2015.
- [52] S. Pathak, A. Thomas, and V. Indelman, “A unified framework for data association aware robust belief space planning and perception,” *The International Journal of Robotics Research*, vol. 37, no. 2-3, pp. 287–315, 2018.
- [53] A. Kitanov and V. Indelman, “Topological belief space planning for active SLAM with pairwise Gaussian potentials and performance guarantees,” *The International Journal of Robotics Research*, vol. 43, no. 1, pp. 69–97, 2024.

-
- [54] S. Dutta Roy, S. Chaudhury, and S. Banerjee, “Active recognition through next view planning: a survey,” *Pattern Recognition*, vol. 37, no. 3, pp. 429–446, 2004.
- [55] T. Patten, W. Martens, and R. Fitch, “Monte Carlo planning for active object classification,” *Autonomous Robots*, vol. 42, no. 2, pp. 391–421, 2018.
- [56] G. Best, J. Faigl, and R. Fitch, “Online planning for multi-robot active perception with self-organising maps,” *Autonomous Robots*, vol. 42, no. 4, pp. 715–738, 2018.
- [57] N. Atanasov, B. Sankaran, J. Le Ny, G. J. Pappas, and K. Daniilidis, “Nonmyopic View Planning for Active Object Classification and Pose Estimation,” *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1078–1090, 2014.
- [58] R. Eidenberger and J. Scharinger, “Active perception and scene modeling by planning with probabilistic 6D object poses,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1036–1043, 2010.
- [59] H. Hu, D. Isele, S. Bae, and J. F. Fisac, “Active uncertainty reduction for safe and efficient interaction planning: A shielding-aware dual control approach,” *The International Journal of Robotics Research*, 2023.
- [60] D. Sadigh, S. S. Sastry, S. A. Seshia, and A. Dragan, “Information gathering actions over human internal state,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 66–73, 2016.
- [61] T. Bandyopadhyay, K. S. Won, E. Frazzoli, D. Hsu, W. S. Lee, and D. Rus, “Intention-Aware Motion Planning,” in *Algorithmic Foundations of Robotics X*, vol. 86, pp. 475–491, 2013.
- [62] S. Bai, J. Wang, F. Chen, and B. Englot, “Information-theoretic exploration with Bayesian optimization,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1816–1822, 2016.
- [63] F. Chen, J. D. Martin, Y. Huang, J. Wang, and B. Englot, “Autonomous Exploration Under Uncertainty via Deep Reinforcement Learning on Graphs,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 6140–6147, 2020.

-
- [64] K.-C. Ma, Z. Ma, L. Liu, and G. S. Sukhatme, “Multi-robot Informative and Adaptive Planning for Persistent Environmental Monitoring,” *Distributed Autonomous Robotic Systems*, vol. 6, pp. 285–298, 2018.
- [65] M. G. Jadidi, J. V. Miro, and G. Dissanayake, “Gaussian processes autonomous mapping and exploration for range-sensing mobile robots,” *Autonomous Robots*, vol. 42, pp. 273–290, 2020.
- [66] C. H. Papadimitriou and J. N. Tsitsiklis, “The Complexity of Markov Decision Processes,” *Mathematics of Operations Research*, vol. 12, no. 3, pp. 441–450, 1987.
- [67] J. Pineau, G. Gordon, and S. Thrun, “Point-based value iteration: an anytime algorithm for POMDPs,” in *International Joint Conference on Artificial Intelligence*, IJCAI’03, pp. 1025–1030, 2003.
- [68] M. T. Spaan and N. Vlassis, “Perseus: Randomized Point-based Value Iteration for POMDPs,” *Journal of Artificial Intelligence Research*, vol. 24, pp. 195–220, 2005.
- [69] H. Kurniawati, D. Hsu, and W. S. Lee, “Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces,” *Robotics: Science and Systems IV*, pp. 65–72, 2009.
- [70] T. Smith and R. Simmons, “Heuristic search value iteration for pomdps,” in *Conference on Uncertainty in Artificial Intelligence*, UAI ’04, p. 520–527, 2004.
- [71] A. Somani, N. Ye, D. Hsu, and W. S. Lee, “DESPOT: Online POMDP Planning with Regularization,” in *Advances in Neural Information Processing Systems*, vol. 26, 2013.
- [72] H. Bai, D. Hsu, W. S. Lee, and V. A. Ngo, “Monte Carlo Value Iteration for Continuous-State POMDPs,” *Algorithmic Foundations of Robotics IX*, vol. 68, pp. 175–191, 2010.
- [73] G. A. Hollinger and G. S. Sukhatme, “Sampling-based robotic information gathering algorithms,” *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1271–1287, 2014.
- [74] K. J. Åström, *Introduction to stochastic control theory*. Courier Corporation, 2012.

-
- [75] M. P. Vitus and C. J. Tomlin, “Closed-loop belief space planning for linear, Gaussian systems,” in *IEEE International Conference on Robotics and Automation*, pp. 2152–2159, 2011.
- [76] J. Van Den Berg, S. Patil, and R. Alterovitz, “Motion planning under uncertainty using iterative local optimization in belief space,” *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1263–1278, 2012.
- [77] S. Prentice and N. Roy, “The Belief Roadmap: Efficient Planning in Belief Space by Factoring the Covariance,” *The International Journal of Robotics Research*, vol. 28, no. 11-12, pp. 1448–1465, 2009.
- [78] A. Bry and N. Roy, “Rapidly-exploring Random Belief Trees for motion planning under uncertainty,” in *IEEE International Conference on Robotics and Automation*, pp. 723–730, 2011.
- [79] S. C. W. Ong, Shao Wei Png, D. Hsu, and Wee Sun Lee, “Planning under Uncertainty for Robotic Tasks with Mixed Observability,” *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 1053–1068, 2010.
- [80] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, “Planning with Trust for Human-Robot Collaboration,” in *ACM/IEEE International Conference on Human-Robot Interaction*, pp. 307–315, 2018.
- [81] D. Qiu, Y. Zhao, and C. L. Baker, “Latent Belief Space Motion Planning under Cost, Dynamics, and Intent Uncertainty,” in *Robotics: Science and Systems*, 2020.
- [82] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, “Social LSTM: Human Trajectory Prediction in Crowded Spaces,” in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 961–971, 2016.
- [83] Z. N. Sunberg, C. J. Ho, and M. J. Kochenderfer, “The value of inferring the internal state of traffic participants for autonomous freeway driving,” in *American Control Conference (ACC)*, pp. 3004–3010, 2017.

-
- [84] W. Schwarting, A. Pierson, J. Alonso-Mora, S. Karaman, and D. Rus, “Social behavior for autonomous vehicles,” *Proceedings of the National Academy of Sciences*, vol. 116, no. 50, pp. 24972–24978, 2019.
- [85] N. Rhinehart, R. McAllister, K. Kitani, and S. Levine, “Precog: Prediction conditioned on goals in visual multi-agent settings,” in *IEEE/CVF International Conference on Computer Vision*, 2019.
- [86] Y. Yao, E. Atkins, M. Johnson-Roberson, R. Vasudevan, and X. Du, “BiTraP: Bi-Directional Pedestrian Trajectory Prediction With Multi-Modal Goal Estimation,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1463–1470, 2021.
- [87] K. Katuwandeniya, S. H. Kiss, L. Shi, and J. Valls Miro, “Multi-modal scene-compliant user intention estimation in navigation,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1001–1006, 2021.
- [88] S. Z. Yong, M. Zhu, and E. Frazzoli, “A unified filter for simultaneous input and state estimation of linear discrete-time stochastic systems,” *Automatica*, vol. 63, pp. 321–329, 2016.
- [89] Y. Li, D. Shi, and T. Chen, “Secure analysis of dynamic networks under pinning attacks against synchronization,” *Automatica*, vol. 111, p. Article 108576, 2020.
- [90] M. Showkatbakhsh, Y. Shoukry, S. N. Diggavi, and P. Tabuada, “Securing state reconstruction under sensor and actuator attacks: Theory and design,” *Automatica*, vol. 116, p. Article 108920, 2020.
- [91] A. P. Dani, Z. Kan, N. R. Fischer, and W. E. Dixon, “Structure estimation of a moving object using a moving camera: An unknown input observer approach,” in *IEEE Conference on Decision and Control and European Control Conference*, pp. 5005–5011, 2011.
- [92] H. Jeong, H. Hassani, M. Morari, D. D. Lee, and G. Pappas, “Learning to track dynamic targets in partially known environments,” *arXiv preprint*, 2020.
- [93] S. Engin and V. Isler, “Active localization of multiple targets from noisy relative measurements,” in *Algorithmic Foundations of Robotics XIV*, 2020.

-
- [94] H. Hur and H. S. Ahn, “Unknown input h_∞ observer-based localization of a mobile robot with sensor failure,” *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 6, pp. 1830–1838, 2014.
- [95] A. Ansari and D. S. Bernstein, “Input estimation for nonminimum-phase systems with application to acceleration estimation for a maneuvering vehicle,” *IEEE Transactions on Control Systems Technology*, vol. 27, no. 4, pp. 1596–1607, 2019.
- [96] E. Hashemi, R. Zarringhalam, A. Khajepour, W. Melek, A. Kasaiezadeh, and S. K. Chen, “Real-time estimation of the road bank and grade angles with unknown input observers,” *Vehicle System Dynamics*, vol. 55, no. 5, pp. 648–667, 2017.
- [97] N. Wallace, H. Kong, A. Hill, and S. Sukkarieh, “Receding horizon estimation and control with structured noise blocking for mobile robot slip compensation,” in *IEEE International Conference on Robotics and Automation*, pp. 1169–1175, 2019.
- [98] N. Wallace, H. Kong, A. Hill, and S. Sukkarieh, “Experimental validation of structured receding horizon estimation and control for mobile ground robot slip compensation,” in *Conference on Field and Service Robotics*, pp. 411–426, 2021.
- [99] H. Guo, Z. Yin, D. Cao, H. Chen, and C. Lv, “A review of estimation for vehicle tire-road interactions toward automated driving,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 1, pp. 14–30, 2019.
- [100] L. D. Phong, J. Choi, and S. Kang, “External force estimation using joint torque sensors for a robot manipulator,” in *IEEE International Conference on Robotics and Automation*, pp. 4507–4512, 2012.
- [101] Q. Li, O. Kroemer, Z. Su, F. F. Veiga, M. Kaboli, and H. J. Ritter, “A review of tactile information: perception and action through touch,” *IEEE Transactions on Robotics*, pp. 1–16, 2020.
- [102] S. Gillijns and B. De Moor, “Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough,” *Automatica*, vol. 43, no. 5, pp. 934–937, 2007.

-
- [103] H. Kong and S. Sukkarieh, “An internal model approach to estimation of systems with arbitrary unknown inputs,” *Automatica*, vol. 108, 2019.
- [104] H. Kong, M. Shan, D. Su, Y. Qiao, A. Al-Azzawi, and S. Sukkarieh, “Filtering for systems subject to unknown inputs without a priori initial information,” *Automatica*, vol. 120, pp. 1–12, 2020.
- [105] M. A. Abooshahab, M. M. J. Alyaseen, R. R. Bitmead, and M. Hovd, “Simultaneous input & state estimation, singular filtering and stability,” *Automatica*, vol. 137, p. 110017, 2022.
- [106] K. A. Tsintotas, L. Bampis, and A. Gasteratos, “The Revisiting Problem in Simultaneous Localization and Mapping,” in *Online Appearance-Based Place Recognition and Mapping*, vol. 133, pp. 1–33, 2022.
- [107] S. Eiffert, K. Li, M. Shan, S. Worrall, S. Sukkarieh, and E. Nebot, “Probabilistic crowd gan: Multimodal pedestrian trajectory prediction using a graph vehicle-pedestrian attention network,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5026–5033, 2020.
- [108] S. Eiffert, N. D. Wallace, H. Kong, N. Pirmarzashti, and S. Sukkarieh, “Experimental evaluation of a hierarchical operating framework for ground robots in agriculture,” in *International Symposium on Experimental Robotics*, pp. 151–160, 2021.
- [109] D. Su, H. Kong, S. Sukkarieh, and S. Huang, “Necessary and sufficient conditions for observability of slam-based tdoa sensor array calibration and source localization,” *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1451–1468, 2021.
- [110] K. Li, S. Eiffert, M. Shan, F. Gomez-Donoso, S. Worrall, and E. Nebot, “Attentional-gcnn: Adaptive pedestrian trajectory prediction towards generic autonomous vehicle use cases,” in *2021 IEEE International Conference on Robotics and Automation*, pp. 14241–14247, 2021.
- [111] K. H. Dinh, O. Oguz, G. Huber, V. Gabler, and D. Wollherr, “An approach to integrate human motion prediction into local obstacle avoidance in close human-robot collaboration,” in *IEEE International Workshop on Advanced Robotics and its Social Impacts*, pp. 1–6, 2015.

-
- [112] A. Vemula, K. Muelling, and J. Oh, “Social Attention: Modeling Attention in Human Crowds,” in *IEEE International Conference on Robotics and Automation*, pp. 4601–4607, 2018.
- [113] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe, “A boosted particle filter: Multitarget detection and tracking,” *Lecture Notes in Computer Science*, 2004.
- [114] K. M. B. Lee, F. Kong, R. Cannizzaro, J. L. Palmer, D. Johnson, C. Yoo, and R. Fitch, “Decentralised intelligence, surveillance, and reconnaissance in unknown environments with heterogeneous multi-robot systems,” in *ICRA 2019 Workshop on Robot Swarms in the Real World: From Design to Deployment*, 2019.
- [115] L. Zhou and V. Kumar, “Robust Multi-Robot Active Target Tracking Against Sensing and Communication Attacks,” in *American Control Conference (ACC)*, pp. 4443–4450, 2022.
- [116] J. L. Ny and G. Pappas, “On trajectory optimization for active sensing in gaussian process models,” in *IEEE Conference on Decision and Control*, pp. 6286–6292, 2009.
- [117] M. Vitus, W. Zhang, A. Abate, J. Hu, and C. Tomlin, “On efficient sensor scheduling for linear dynamical systems,” *Automatica*, vol. 48, no. 10, pp. 2482–2493, 2012.
- [118] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, “Sensor placement for optimal Kalman filtering: Fundamental limits, submodularity, and algorithms,” in *American Control Conference (ACC)*, pp. 191–196, 2016.
- [119] B. Schlotfeldt, D. Thakur, N. Atanasov, V. Kumar, and G. Pappas, “Anytime planning for decentralized multirobot active information gathering,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1025–1032, 2018.
- [120] Y. Kantaros, B. Schlotfeldt, N. Atanasov, and G. Pappas, “Asymptotically optimal planning for non-myopic multi-robot information gathering,” in *Robotics: Science and Systems*, pp. 22–26, 2019.

-
- [121] Ke Zhou and S. I. Roumeliotis, “Multirobot Active Target Tracking With Combinations of Relative Observations,” *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 678–695, 2011.
- [122] Z. Zhang and P. Tokekar, “Non-myopic target tracking strategies for non-linear systems,” in *IEEE Conference on Decision and Control*, pp. 5591–5596, 2016.
- [123] R. Zahroof, J. Liu, L. Zhou, and V. Kumar, “Multi-Robot Localization and Target Tracking with Connectivity Maintenance and Collision Avoidance,” in *American Control Conference (ACC)*, pp. 1331–1338, 2023.
- [124] A. Ryan and J. K. Hedrick, “Particle filter based information-theoretic active sensing,” *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 574–584, 2010.
- [125] P. Tokekar, V. Isler, and A. Franchi, “Multi-target visual tracking with aerial robots,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3067–3072, 2014.
- [126] J. Fischer and O. S. Tas, “Information particle filter tree: An online algorithm for POMDPs with belief-based rewards on continuous domains,” in *International Conference on Machine Learning*, vol. 119 of *Proceedings of Machine Learning Research*, pp. 3177–3187, PMLR, 2020.
- [127] F. Altche and A. de La Fortelle, “An LSTM network for highway trajectory prediction,” in *IEEE 20th International Conference on Intelligent Transportation Systems*, pp. 353–359, 2017.
- [128] Z. Shi, M. Xu, Q. Pan, B. Yan, and H. Zhang, “LSTM-based Flight Trajectory Prediction,” in *International Joint Conference on Neural Networks*, pp. 1–8, 2018.
- [129] S. Eiffert, H. Kong, N. Pirmarzashti, and S. Sukkarieh, “Path Planning in Dynamic Environments using Generative RNNs and Monte Carlo Tree Search,” in *IEEE International Conference on Robotics and Automation*, pp. 10263–10269, 2020.
- [130] J. Bi, H. Yuan, K. Xu, H. Ma, and M. Zhou, “Large-scale Network Traffic Prediction With LSTM and Temporal Convolutional Networks,” in *International Conference on Robotics and Automation*, pp. 3865–3870, 2022.

-
- [131] W. Zhi, L. Ott, and F. Ramos, “Kernel Trajectory Maps for Multi-Modal Probabilistic Motion Prediction,” in *Proceedings of the Conference on Robot Learning*, vol. 100, pp. 1405–1414, 2020.
- [132] J. Wiest, M. Hoffken, U. Kresel, and K. Dietmayer, “Probabilistic trajectory prediction with Gaussian mixture models,” in *IEEE Intelligent Vehicles Symposium*, pp. 141–146, 2012.
- [133] B. Ivanovic and M. Pavone, “The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.
- [134] A. Zyner, S. Worrall, and E. Nebot, “Naturalistic Driver Intention and Path Prediction Using Recurrent Neural Networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1584–1594, 2020.
- [135] B.-N. Vo and W.-K. Ma, “The Gaussian Mixture Probability Hypothesis Density Filter,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, 2006.
- [136] Ruijie He, A. Bachrach, and N. Roy, “Efficient planning under uncertainty for a target-tracking micro-aerial vehicle,” in *IEEE International Conference on Robotics and Automation*, pp. 1–8, 2010.
- [137] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, “Multiple Hypothesis Tracking Revisited,” in *IEEE International Conference on Computer Vision*, pp. 4696–4704, 2015.
- [138] F. Zheng, Y. Tian, W. Zhan, J. Yu, and K. Liu, “A Gaussian mixture multiple-model belief propagation filter for multisensor-multitarget tracking,” *Signal Processing*, vol. 220, p. 109473, 2024.
- [139] M. Shienman and V. Indelman, “Nonmyopic Distilled Data Association Belief Space Planning Under Budget Constraints,” *Robotics Research*, vol. 27, pp. 102–118, 2023.
- [140] K. M. B. Lee, J. J. H. Lee, C. Yoo, B. Hollings, and R. Fitch, “Active perception for plume source localisation with underwater gliders,” in *ARAA Australasian Conference on Robotics and Automation*, 2018.

-
- [141] G. D'urso, J. J. H. Lee, O. Pizarro, C. Yoo, and R. Fitch, "Hierarchical mcts for scalable multi-vessel multi-float systems," in *IEEE International Conference on Robotics and Automation*, pp. 8664–8670, 2021.
- [142] M. Popovic, T. Vidal-Calleja, G. Hiltz, J. J. Chung, I. Sa, R. Siegwart, and J. Nieto, "An informative path planning framework for UAV-based terrain monitoring," *Autonomous Robots*, vol. 44, pp. 889 – 911, 2020.
- [143] L. Wu, K. M. B. Lee, L. Liu, and T. Vidal-Calleja, "Faithful Euclidean distance field from log-Gaussian process implicit surfaces," *Robotics and Automation Letters*, vol. 6, pp. 2461–2468, 2021.
- [144] F. Ramos and L. Ott, "Hilbert maps: Scalable continuous occupancy mapping with stochastic gradient descent," *The International Journal of Robotics Research*, vol. 35, no. 14, pp. 1717–1730, 2016.
- [145] L. Jin, J. Ruckin, S. H. Kiss, T. Vidal-Calleja, and M. Popovic, "Adaptive-Resolution Field Mapping Using Gaussian Process Fusion With Integral Kernels," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7471–7478, 2022.
- [146] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [147] E. Snelson and Z. Ghahramani, "Sparse Gaussian Processes using Pseudo-inputs," in *Advances in Neural Information Processing Systems*, vol. 18, 2005.
- [148] M. Schürch, D. Azzimonti, A. Benavoli, and M. Zaffalon, "Recursive estimation for sparse Gaussian process regression," *Automatica*, vol. 120, p. 109127, 2020.
- [149] M. Montemerlo, S. Thrun, D. Roller, and B. Wegbreit, "FastSLAM 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence, IJCAI'03*, pp. 1151–1156, 2003.
- [150] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

-
- [151] L. Sun, T. Vidal-Calleja, and J. V. Miro, “Bayesian fusion using conditionally independent submaps for high resolution 2.5D mapping,” in *IEEE International Conference on Robotics and Automation*, 2015.
- [152] K.-C. Ma, L. Liu, H. K. Heidarrsson, and G. S. Sukhatme, “Data-driven learning and planning for environmental sampling,” *Journal of Field Robotics*, vol. 35, pp. 643–661, 2018.
- [153] M. Kanagawa, P. Hennig, D. Sejdinovic, and B. K. Sriperumbudur, “Gaussian processes and kernel methods: A review on connections and equivalences,” 2018.
- [154] T. Karvonen, G. Wynne, F. Tronarp, C. Oates, and S. Sarkka, “Maximum likelihood estimation and uncertainty quantification for gaussian process approximation of deterministic functions,” *SIAM/ASA Journal on Uncertainty Quantification*, vol. 8, no. 3, pp. 926–958, 2020.
- [155] T. Karvonen, S. Särkkä, and K. Tanaka, “Kernel-based interpolation at approximate Fekete points,” *Numerical Algorithms*, vol. 87, no. 1, pp. 445–468, 2021.
- [156] J. Quinonero-Candela and C. E. Rasumussen, “A unifying view of sparse approximate Gaussian process regression,” *Journal of Machine Learning Research*, vol. 6, pp. 1939–1959, 2005.
- [157] M. Titsias, “Variational learning of inducing variables in sparse Gaussian processes,” in *Proceedings of Artificial intelligence and statistics*, pp. 567–574, 2009.
- [158] K. M. B. Lee, W. Martens, J. Khatkar, R. Fitch, and R. Mettu, “Efficient updates for data association with mixtures of Gaussian processes,” in *IEEE International Conference on Robotics and Automation*, pp. 335–341, IEEE, 2020.
- [159] A. G. Wilson and H. Nickisch, “Kernel interpolation for scalable structured Gaussian processes (KISS-GP),” *International Conference on Machine Learning*, vol. 3, pp. 1775–1784, 2015.
- [160] K. Y. C. To, F. H. Kong, K. M. B. Lee, C. Yoo, S. Anstee, and R. Fitch, “Estimation of spatially correlated ocean currents from ensemble forecasts and online measurements,” in *IEEE International Conference on Robotics and Automation*, 2021.

-
- [161] H. A. Kingravi, H. Maske, and G. Chowdhary, “Kernel observers: Systems-theoretic modeling and inference of spatiotemporally evolving processes,” in *Advances in Neural Information Processing Systems*, pp. 3990–3998, 2016.
- [162] T. Vidal-Calleja, D. Su, F. De Bruijn, and J. V. Miro, “Learning spatial correlations for Bayesian fusion in pipe thickness mapping,” in *IEEE International Conference on Robotics and Automation*, pp. 683–690, 2014.
- [163] R. Mishra, M. Chitre, and S. Swarup, “Online Informative Path Planning Using Sparse Gaussian Processes,” in *OCEANS Conference*, pp. 1–5, 2018.
- [164] A. Viseras, D. Shutin, and L. Merino, “Robotic Active Information Gathering for Spatial Field Reconstruction with Rapidly-Exploring Random Trees and Online Learning of Gaussian Processes,” *Sensors*, vol. 19, no. 5, p. 1016, 2019.
- [165] L. Kocsis and C. Szepesvári, “Bandit Based Monte-Carlo Planning,” *Machine Learning*, vol. 4212, pp. 282–293, 2006.
- [166] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, “A Survey of Monte Carlo Tree Search Methods,” *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 1–43, 2012.
- [167] J. Wakulicz, H. Kong, and S. Sukkarieh, “Active information acquisition under arbitrary unknown disturbances,” in *IEEE International Conference on Robotics and Automation*, 2021.
- [168] R. R. Bitmead, M. Hovd, and M. A. Abooshab, “A kalman-filtering derivation of simultaneous input and state estimation,” *Automatica*, vol. 108, p. Article 108478, 2019.
- [169] J. Wakulicz, K. M. Brian Lee, T. Vidal-Calleja, and R. Fitch, “Topological Trajectory Prediction with Homotopy Classes,” in *IEEE International Conference on Robotics and Automation*, pp. 6930–6936, 2023.
- [170] A. Hatcher, *Algebraic Topology*. 2002.

-
- [171] S. Bhattacharya, V. Kumar, and M. Likhachev, “Search-based path planning with homotopy class constraints,” in *AAAI Conference on Artificial Intelligence*, pp. 1230–1237, 2010.
- [172] S. Bhattacharya, R. Ghrist, and V. Kumar, “Persistent Homology for Path Planning in Uncertain Environments,” *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 578–590, 2015.
- [173] F. T. Pokorny, M. Hawasly, and S. Ramamoorthy, “Topological trajectory classification with filtrations of simplicial complexes and persistent homology,” *International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 204–223, 2016.
- [174] J. Frederico Carvalho, M. Vejdemo-Johansson, F. T. Pokorny, and D. Kragic, “Long-term Prediction of Motion Trajectories Using Path Homology Clusters,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 765–772, 2019.
- [175] S. McCammon and G. A. Hollinger, “Topological path planning for autonomous information gathering,” *Autonomous Robots*, 2021.
- [176] S. Bhattacharya, S. Kim, H. Heidarsson, G. S. Sukhatme, and V. Kumar, “A topological approach to using cables to separate and manipulate sets of objects,” *The International Journal of Robotics Research*, vol. 34, no. 6, pp. 799–815, 2015.
- [177] D. Ron, Y. Singer, and N. Tishby, “The Power of Amnesia,” in *Advances in Neural Information Processing Systems*, vol. 6, 1993.
- [178] K. B. Petersen and M. S. Pedersen, *The Matrix Cookbook*. Technical University of Denmark, 2007.
- [179] D. Brscic, T. Kanda, T. Ikeda, and T. Miyashita, “Person position and body direction tracking in large public spaces using 3D range sensors,” *IEEE Transactions on Human-Machine Systems*, vol. 43, pp. 522 – 534, 2013.
- [180] Tian Lan, D. Erdogmus, U. Ozertem, and Yonghong Huang, “Estimating Mutual Information Using Gaussian Mixture Model for Feature Ranking and Selection,” in

- IEEE International Joint Conference on Neural Network Proceedings*, pp. 5034–5039, 2006.
- [181] A. Kolchinsky and B. Tracey, “Estimating Mixture Entropy with Pairwise Distances,” *Entropy*, vol. 19, no. 7, p. 361, 2017.
- [182] B. Tovar, F. Cohen, and S. M. LaValle, “Sensor Beams, Obstacles, and Possible Paths,” *Algorithmic Foundation of Robotics VIII*, vol. 57, pp. 317–332, 2009.
- [183] M. G. Kantor and M. B. Rosenwein, “The Orienteering Problem with Time Windows,” *The Journal of the Operational Research Society*, vol. 43, no. 6, p. 629, 1992.
- [184] J. R. Hershey and P. A. Olsen, “Approximating the Kullback Leibler Divergence Between Gaussian Mixture Models,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. IV–317–IV–320, 2007.
- [185] K. C. To, J. J. H. Lee, C. Yoo, S. Anstee, and R. Fitch, “Streamline-based control of underwater gliders in 3d environments,” in *IEEE Conference on Decision and Control*, pp. 8303–8310, 2019.
- [186] K. C. To, C. Yoo, S. Anstee, and R. Fitch, “Distance and steering heuristics for streamline-based flow field planning,” in *IEEE International Conference on Robotics and Automation*, pp. 1867–1873, 2020.
- [187] J. Wakulicz, K. M. Brian Lee, C. Yoo, T. Vidal-Calleja, and R. Fitch, “Informative Planning for Worst-Case Error Minimisation in Sparse Gaussian Process Regression,” in *International Conference on Robotics and Automation*, pp. 11066–11072, 2022.
- [188] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, “Information-theoretic regret bounds for gaussian process optimization in the bandit setting,” *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.
- [189] A. Lederer, J. Umlauft, and S. Hirche, “Uniform error bounds for gaussian process regression with application to safe control,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.

-
- [190] N. Atanasov, *Active Information Acquisition with Mobile Robots*. PhD thesis, University of Pennsylvania, 2015.
- [191] N. R. Gans, G. Hu, K. Nagarajan, and W. E. Dixon, “Keeping Multiple Moving Targets in the Field of View of a Mobile Camera,” *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 822–828, 2011.
- [192] Y. Sung and P. Tokekar, “GM-PHD Filter for Searching and Tracking an Unknown Number of Targets With a Mobile Sensor With Limited FOV,” *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 2122–2134, 2022.
- [193] Z. Zhang, “Iterative point matching for registration of free-form curves and surfaces,” *International Journal of Computer Vision*, vol. 13, no. 2, pp. 119–152, 1994.
- [194] Y. Chen and G. Medioni, “Object modelling by registration of multiple range images,” *Image and Vision Computing*, vol. 10, no. 3, pp. 145–155, 1992.
- [195] P. Besl and N. D. McKay, “A method for registration of 3-D shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.
- [196] F. Sherwani, M. M. Asad, and B. Ibrahim, “Collaborative Robots and Industrial Revolution 4.0 (IR 4.0),” in *International Conference on Emerging Trends in Smart Technologies*, pp. 1–5, 2020.
- [197] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. M. Paixão, F. Mutz, L. De Paula Veronese, T. Oliveira-Santos, and A. F. De Souza, “Self-driving cars: A survey,” *Expert Systems with Applications*, vol. 165, p. 113816, 2021.
- [198] R. Bhatia, *Matrix analysis*. 2013.