INFORMATION-THEORETIC REASONING IN DISTRIBUTED AND AUTONOMOUS SYSTEMS

OLIVER M. CLIFF
BE (HONS)

A THESIS SUBMITTED IN FULFILMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

SCHOOL OF AERONAUTICAL, MECHANICAL AND MECHATRONIC ENGINEERING
FACULTY OF ENGINEERING AND IT
UNIVERSITY OF SYDNEY

SUBMITTED MAY 2018; REVISED MAY 2019

For Pa.

DECLARATION

This is to certify that, to the best of my knowledge, the content of this thesis is my own work. This thesis has not been submitted for any other degree or purposes.

I certify that the intellectual content of this thesis is the product of my own work and that all relevant resources used in preparing this thesis have been acknowledged.

May 2019.

Oliver M. Cliff

ABSTRACT

The increasing prevalence of distributed and autonomous systems is transforming decision making in industries as diverse as agriculture, environmental monitoring, and healthcare. Despite significant efforts, challenges remain in robustly planning under uncertainty. In this thesis, we present a number of information-theoretic decision rules for improving the analysis and control of complex adaptive systems.

We begin with the problem of quantifying the data storage (memory) and transfer (communication) within information processing systems. We develop an information-theoretic framework to study nonlinear interactions within cooperative and adversarial scenarios, solely from observations of each agent's dynamics. This framework is applied to simulations of robotic soccer games, where the measures reveal insights into team performance, including correlations of the information dynamics to the scoreline. We then study the communication between processes with latent nonlinear dynamics that are observed only through a filter. By using methods from differential topology, we show that the information-theoretic measures commonly used to infer communication in observed systems can also be used in certain partially observed systems.

For robotic environmental monitoring, the quality of data depends on the placement of sensors. These locations can be improved by either better estimating the quality of future viewpoints or by a team of robots operating concurrently. By robustly handling the uncertainty of sensor model measurements, we are able to present the first end-to-end robotic system for autonomously tracking small dynamic animals, with a performance comparable to human trackers. We then solve the issue of coordinating multi-robot systems through distributed optimisation techniques. These allow us to develop non-myopic robot trajectories for these tasks and, importantly, show that these algorithms provide guarantees for convergence rates to the optimal payoff sequence.

PUBLICATIONS

Some of the ideas and figures in this thesis have appeared previously in the following publications during the candidature for this degree.

[23] G. Best, O. M. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Decentralised Monte Carlo tree search for active perception," in *Proc. of WAFR*, 2016.

[24] G. Best, O. M. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Dec-MCTS: Decentralized planning for multi-robot active perception," *Int. J. Robot. Res.*, vol. 38, pp. 316–337, 2–3 2018.

[52] O. M. Cliff, M. Prokopenko, and R. Fitch, "An information criterion for inferring coupling in distributed dynamical systems," *Front. Robot. AI*, vol. 3, no. 71, 2016.

[53] O. M. Cliff, R. Fitch, S. Sukkarieh, D. L. Saunders, and R. Heinsohn, "Online localization of radio-tagged wildlife with an autonomous aerial robot system," in *Proc. of RSS*, 2015.

[54] O. M. Cliff, N. Harding, M. Piraveenan, E. Y. Erten, M. Gambhir, and M. Prokopenko, "Investigating spatiotemporal dynamics and synchrony of influenza epidemics in Australia: An agent-based modelling approach," *Simul. Model. Pract. Th.*, vol. 87, pp. 412–431, 2018.

[56] O. M. Cliff, J. T. Lizier, X. R. Wang, P. Wang, O. Obst, and M. Prokopenko, "Quantifying long-range interactions and coherent structure in multi-agent dynamics," *Art. Life*, vol. 23, no. 1, pp. 34–57, 2017.

[57] O. M. Cliff, M. Prokopenko, and R. Fitch, "Minimising the Kullback-Leibler divergence for model selection in distributed nonlinear systems," *Entropy*, vol. 20, no. 2, p. 51, 2018.

[58] O. M. Cliff, D. Saunders, S. Sukkarieh, and R. Fitch, "Robotic ecology: Tracking small animals with an autonomous aerial vehicle," *Science Robot.*, vol. 3, eaat8409, 23 2018.

[59] O. M. Cliff, V. Sintchenko, T. C. Sorrell, K. Vadlamudi, N. McLean, and M. Prokopenko, "Network properties of salmonella epidemics," *Sci. Rep.*, vol. 9, no. 1, p. 6159, 2019.

[101] B. Hefferan, O. M. Cliff, and R. Fitch, "Adversarial patrolling with reactive point processes," in *Proc. of ARAA ACRA*, 2016.

[265] C. Zachreson, K. M. Fair, O. M. Cliff, N. Harding, M. Piraveenan, and M. Prokopenko, "Urbanization affects peak timing, prevalence, and bimodality of influenza pandemics in Australia: Results of a census-calibrated model," *Sci. Adv.*, vol. 4, no. 12, eaau5294, 2018.

AUTHORSHIP ATTRIBUTION STATEMENT

Results from a number of these publications are included as substantial contributions to this thesis.

- Chapter 4 contains material from [55, 56] where I co-designed the study with co-authors, analysed the data and co-wrote the MS.
- Chapter 5 contains material from [52, 57] where I designed the study, devised the theory and code, ran the experiments and analysis, and co-wrote the MS.
- Chapter 6 contains material from [53, 58] where I co-designed the study with co-authors, devised the theory and code, ran the experiments and analysis, and co-wrote the MS.
- Chapter 7 contains material from [23, 24] where I wrote the theoretical analysis and co-wrote the MS.

In addition to the statements above, in cases where I am not the corresponding author of a published item, permission to include the published material has been granted by the corresponding author.

May 2019.

Oliver M. Cliff

As supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct.

May 2019.

Robert Fitch

All stable processes we shall predict.
All unstable processes we shall control.

— John von Neumann

ACKNOWLEDGMENTS

No one creates anything alone. The ideas expressed in this dissertation are a result of the people I have known both within and outside of the academic community. Because of this, I would like to thank a number of people instrumental in producing this work.

To my supervisors. Rob, you have unending support for my ludicrous ideas and limitless patience for my uncanny ability to make the same grammatical errors *ad infinitum*. The time you set aside every day has been invaluable to me and I have no doubt that your sage advice on everything from brackets to bourbon will have an everlasting effect on me. Mikhail, your enthusiasm for science is infectious and is a major reason I started my PhD. Since then, your optimism has kept me interested in subjects as far-reaching as epidemiology to robotic soccer (even if sometimes this optimism means a bug in my code leads us to decide we have solved science).

To my family. Mum and dad, your unconditional guidance and support for me never goes unappreciated. You will always listen to my bizarre technical problems (dad) and force me to eat so that I don't waste away (mum). Tim, Lily, Nick and Christy, you always know how to keep your little brother grounded. Thanks for being the best siblings (and siblings-in-law) that I could wish for.

To my friends. Tom and Hannah, you should probably appear in the previous paragraph for being my surrogate parents for most of my candidature; I hope to one day make it up to you. To my other non-academic (read: normal) friends, your diverse interests always help to distract me from my work and see the forest for the trees. Thanks to the guys from the Australian Centre for Field Robotics for always having time to discuss any problem I have; in particular Graeme, Tim, and Wolfram. Thanks also to everyone from the Complex Systems Research Group for these same conversations, as well as some that are in no way helpful to society; in particular Conor, Nathan, Emanuele, and Leo.

I'd also like to thank all of the collaborators I've had over the last few years. Thanks, Joe, for introducing me to information theory, as well as Jag and Deb, for putting up with all our field trials and the incessant bugs in my code. Finally, some of the theoretical frameworks presented in this thesis were verified by numerous visiting academics; most notably, Nihat Ay and Jürgen Jost.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

2TBN	Two time-slice Bayesian networks
AGC	Automatic gain control
AIC	Akaike information criterion
ASV	Autonomous surface vehicle
BIC	Bayesian information criterion
BN	Bayesian network
CPD	Conditional probability distribution
DAG	Directed acyclic graph
DBN	Dynamic Bayesian network
DCM	Dynamic causal modelling
Dec-MCTS	Decentralised Monte Carlo tree search
Dec-POMDP	Decentralised partially observable Markov decision process
D-UCB	Discounted upper confidence bounds
D-UCT	Discounted upper confidence bounds applied to trees
EM	Expectation-maximisation
GCS	Ground control station
GDS	Graph dynamical system
JIDT	Java information dynamics toolkit
KL	Kullback-Leibler
MAB	Multi-armed bandit
MAP	Maximum *a posteriori*
MCTS	Monte Carlo tree search
MDL	Minimum description length
MDP	Markov decision process
MGV	Mobile ground vehicle
ODE	Ordinary differential equation
PDF	Probability density function

PGM	Probabilistic graphical model
POMDP	Partially observable Markov decision process
POSGDS	Partially observable synchronous graph dynamical system
RF	Radio frequency
ROS	Robotic operating system
RSSI	Received signal strength indicator
UAV	Unmanned aerial vehicle
UCB	Upper confidence bound
UCT	Upper confidence bounds applied to trees
VHF	Very high frequency

1 INTRODUCTION

The ability to make decisions under uncertainty is fundamental to many areas of science and engineering. In general, a model is formulated to explain observed phenomena and then experiments are planned with the aim to validate this model. This procedure can be summed up as reducing the uncertainty over a belief. Information theory addresses this challenge by quantifying the amount of predictability in information processing systems in order to study the storage and transfer of data. In this thesis we subscribe to this paradigm by using information measures in a variety of contexts inspired by artificial and biological systems. We focus on two decision problems: model selection for distributed systems, and planning algorithms for single- and multi-robot information gathering tasks.

1.1 REASONING IN COMPLEX ENVIRONMENTS

Complex systems are broadly defined as systems that comprise interacting nonlinear components [32]. Their wide scope renders the problem of reasoning within these environments a topic of general interest studied in various areas of artificial intelligence. As a consequence, there is significant effort in improving decision making practices whose purpose is to understand and ultimately predict the outcome of complex spatiotemporal phenomena.

1.1.1 *Statistical model selection*

The problem of modelling distributed information processing systems is that of inferring data-driven statistical models, often in the case where each subsystem can be viewed as a nonlinear dynamical system. This encompasses many practical problems of known artificial, biological and chemical systems studied in ecology [227], neuroscience [203, 245], robotics [26, 86, 260], and various other fields [32]. By selecting models that allow for a parsimonious representation of the underlying phenomena, we are able to perform efficient inference and better understand the physical processes being studied.

Although many complex adaptive systems differ physically, they handle information similarly. Distributed computation is, in general, discussed in terms of *mem-*

ory, *communication*, and *processing*. These primatives are concerned with the *storage*, *transfer*, and *modification* of information within and between processes [147]. Information theory provides a model-free approach to identify and account for these component operations when selecting an appropriate model. Although these models are only an approximation to reality, by matching these information primitives we can perform effective inference in complex environments.

1.1.2 *Active perception*

When the goal is to model the environment, these models can be improved by deploying autonomous systems that actively explore the environment and obtain more relevant information than passively recorded data. For this reason, information gathering is a fundamentally important family of problems in robotics that plays a primary role in a wide variety of tasks, ranging from scene understanding to manipulation.

Although the idea of exploiting robot motion to improve the quality of information gathering has been studied for nearly three decades [16], most real robot systems today (both single- and multi-robot) still gather information passively. The motivation for an active approach is that sensor data quality (and hence perception quality) relies critically on an appropriate choice of viewpoints. This is particularly the case in complex perception tasks such as object classification, which is known to be highly viewpoint-dependent [175].

One way to efficiently achieve an improved set of viewpoints is via high fidelity sensor modelling. This allows for the robot to better predict the quality of future observations. A major challenge arises in modelling measurement uncertainty in order to ensure the observations are not too overconfident (leading to an inaccurate belief) or too underconfident (leading to an imprecise belief).

Another way to improve viewpoint selection is through teams of robots, where concurrency allows for scaling up the number of observations in time and space. In multi-agent systems, the main hurdle is to coordinate the behaviour of robots as they actively gather information, ideally in a decentralised manner.

1.1.3 *Multi-agent dynamics*

While explicit communication is typically studied within the formulation of robotic information gathering tasks, the detection and quantification of implicit indirect interactions in distributed systems remains a challenge. This is primarily due to

inaccessibility of the logic and neural processing of the team as well as noise in the environment.

The challenges of distributed control are a result of the shared collective objectives of a multi-agent system, in which multiple autonomous agents must cooperate in making distributed decisions towards optimising the overall team objective. In addition, not all communications occur explicitly within well-defined channels. Instead, complex multi-agent behaviours involve tacit interactions which can be characterised by implicit communications, including spatially long-range interactions with indirect effects. These implicit interactions need to be properly accounted for within specific feedback control loops.

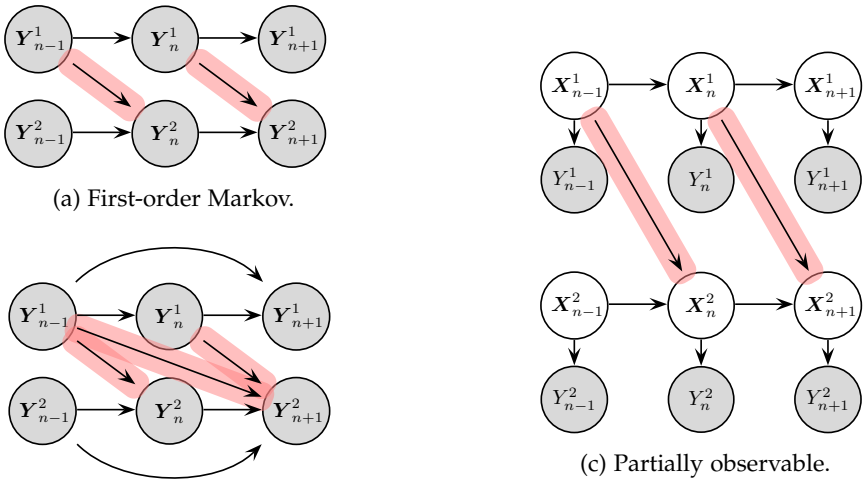
Furthermore, these dynamics are often constrained by either changing and partially unknown environmental factors or competing objectives of adversaries engaged in directly opposing activities. Multi-robot information gathering is a canonical example of a scenario where latent and (often dynamic) environmental states influence team dynamics. The complexity of these dynamics is compounded by adversarial interactions in, e.g., various team sports scenarios, where some of the interactions cannot be simply reduced to algorithmic details of the agents, being affected by a multiplicity of concurrent activities.

Many multi-agent tasks, real and virtual, include rich interactions occurring dynamically and shaping the course of the contest both locally and globally. While the interactions within a team are usually constrained by cooperatively shared plans and tactical schemes, the interactions across the teams are created by opposing objectives of competing players. Generally, the interactions vary in strength spatiotemporally, manifesting some tacit correlations that often are delayed in time and are long-ranged over the environment.

Thus, distributed control of a multi-agent system – for tasks as wide-ranging as information gathering to team sports – demands new techniques for identifying possibilities and features of feedback control loops. For instance, changing team tactics during a contest requires the team to quickly and coherently detect emergent patterns and regularities, quantify their strength and extent, and evaluate the potential impact on the overall performance.

1.2 INFORMATION-THEORETIC DECISION MAKING

The concept of information-theoretic decision making is ubiquitous in the study of distributed and autonomous systems. Entropy is a fundamental utility function for cooperatively or independently optimising a data stream and, moreover, selecting statistical models based on the incoming data.



(a) First-order Markov.

(b) Second-order Markov.

(c) Partially observable.

Figure 1.1: DBNs for model selection problems. The systems comprise hidden $\boldsymbol{X}_n$ and observed $\boldsymbol{Y}_n$ variables. When studying communication of distributed processes, the objective is to infer the coupling (edges) between subsystems, i.e., recovering the highlighted edges or ascertaining their significance in the transfer of information.

1.2.1 *Information in statistical modelling*

One of the most fundamental approaches to inference is the hypothetico-deductive method [90, 159]. Under this framework, hypotheses are iteratively selected or rejected to explain phenomena that are *closer* to the observed truth. For statistical models, this closeness is often quantified by the information gained (via the likelihood ratio tests) or information lost (via the Kullback-Leibler (KL) divergence) when considering competing models. Thus, information theory can be used to reason in statistical inference tasks in general. However, we are more interested in the specific case of modelling communication and memory within distributed systems.

Modelling either fully or partially observable systems as probabilistic graphical models (PGMs) presents a challenge in synthesising these models and capturing their global properties [32]. Figure 1.1 illustrates canonical stochastic processes that are represented in this way. The information transfer is illustrated by the interprocess coupling between subsystems (highlighted edges), which causes a flow of information through the network. The information storage is illustrated by intraprocess coupling, which characterises the dependency of subsystem dynamics on their past.

In many situations, it suffices to model the processes we are studying as Markovian where the state $\boldsymbol{Y}$ is fully observed for each time step. Figure 1.1 illustrate this

case for first-order 1.1a and second-order 1.1b Markov chains through dynamic Bayesian networks (DBNs), where the top process is driving the lower process. If some basic properties of nonlinear systems are satisfied, information theory allows us to accurately investigate the storage and transfer of data without the need for a model. For information-theoretic measures to be suitable, the dynamics should be in steady-state (i.e., the processes are stationary), so that the conditional probability distributions (CPDs) are homogeneous; furthermore, they should be ergodic, so that parameter estimates converge to the true values [19, 206, 207]. We draw on this approach in Chapter 4 and use information-theoretic measures to study the information dynamics of multi-agent systems. This allows us to quantify indirect interactions without knowledge of the actual algorithms underscoring these systems.

In other cases, the dynamics are hidden and observed only through a filter. These systems can be characterised by a transition map (that describes their evolution over time $\boldsymbol{X}$) and a read-out function (through which we observe this latent state to obtain measurements $\boldsymbol{Y}$). The distributed form of this system is where a set of of these subsystems is unidirectionally coupled to one another, as is shown in Fig. 1.1c; in this thesis, models of this type are labelled a partially observable synchronous graph dynamical system (POSGDS). In general, we can only obtain estimates of this coupling through approximation procedures such as expectation-maximisation [66, 128]. However, we show in Chapter 5 that if the dynamics and measurement maps are generic functions (as per [68, 221, 222, 232]), attractor reconstruction can be used to optimally recover the underlying network. Moreover, the resulting measures that are used to reveal this network are the same ones that are commonly used for studying information dynamics in the fully observed case (i.e., Figs. 1.1a and 1.1b).

1.2.2 *Informative path planning*

In robotic information gathering tasks, the objective is to find a sequence of viewpoints such that uncertainty (entropy) over the environment belief is minimised. This sequence can be improved by active informative path planning algorithms. In these algorithms, at each decision step, the robot uses current knowledge to choose future actions that maximise the information gained about the environment.

Active information gathering can be formulated, in general, as a partially observable Markov decision process (POMDP) where the reward at each time step is given by the mutual information between the prior and posterior beliefs. Partial observability here refers to whether or not the reward can be computed at a given

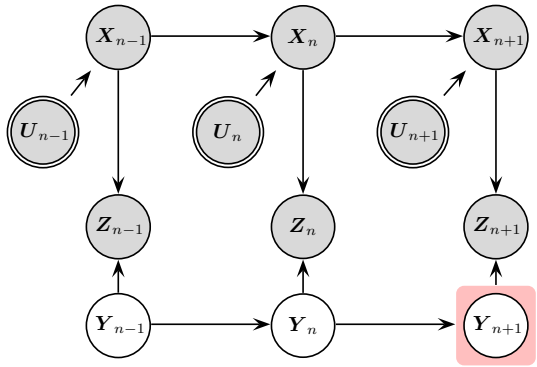


Figure 1.2: DBNs for information gathering problems. The robotic systems comprise a (hidden or observed) state $\boldsymbol{X}_n$, control $\boldsymbol{U}_n$, and measurements $\boldsymbol{Z}_n$ of some hidden quantity of interest $\boldsymbol{Y}_n$. The objective is to infer the hidden state $\boldsymbol{Y}_{n+1}$ or the entire sequence of hidden states $\boldsymbol{Y}_{n+1}^{-}$.

time (based on observability of the underlying state). Thus, the problem can often be simplified to a Markov decision process (MDP) if the robot state is known at any given instant. Figure 1.2 illustrates this case, where the robot state $\boldsymbol{X}$, measurement $\boldsymbol{Z}$, and actions $\boldsymbol{U}$ are all observed and the main concern is to infer the most recent value of the hidden variable of interest $\boldsymbol{Y}$ (the highlighted node).

There are many known techniques for solving the general class of MDPs, but policies are usually computed in advance and executed online (e.g., passive coverage algorithms [235]) or learned iteratively (e.g., reinforcement learning [230]). In order to actively collect data, however, we require efficient algorithms or problem-specific solutions. Near-optimal greedy approximation algorithms exist for the special case where the objective function is monotone submodular [211]. Active information gathering tasks admit this property as long as the energy cost of the path is not taken into account. For instance, in Chapter 6, we study the problem of autonomously tracking wildlife with an aerial vehicle. In this scenario, a small number of observations are able to triangulate the animal and thus we can ignore the cost of travel between observation locations. The major challenge there is to robustly handle measurement uncertainty (i.e., sensor modelling).

Multi-robot active information gathering tasks can be formulated as a decentralised variant of POMDPs known as Dec-POMDP [7]. As in the centralised approach, no efficient distributed planning algorithms exist unless the objective function is monotone submodular [211]. Although this assumption is valid for problems such as continuously monitoring particular environments [87], it is not always the case, particularly in field robotics applications. Approaches such as Monte Carlo tree search (MCTS) are promising for online planning as they provide long time-horizon solutions and admit general purpose objective functions. However, the time complexity of tree search problems scales exponentially with the number of robots and

so we require efficient means for optimising in the joint (team) space. To address this issue, we present a novel algorithm in Chapter 7 that combines MCTS with *probability collectives* [255–257], a game-theoretic approach to distributed optimisation.

1.3 CONTRIBUTIONS OF THE THESIS

The fundamental contribution of this thesis is a framework to study the memory and communication of information in fully and partially observable dynamical systems, coupled with methods to optimise the information gathering of this data with single-robot and multi-robot systems. We detail the individual contributions below.

We give a taxonomy of time series analysis and reasoning by collating surrounding literature and technical background. This includes a general formulation of decision rules for model selection and robotic path planning problems.

We provide a framework for quantifying long-range interactions and implicit communication in multi-agent dynamics. This includes contextualising multivariate transfer entropy and active information storage in terms of network model selection in order to present a model-free approach to quantifying responsiveness and rigidity of agents in team scenarios. This facilitates the use of information-source and information-sink diagrams for identifying driving and driven agents in team scenarios. Furthermore, by using coherence state-space plots, we show that information dynamics correlate to collaborative goals of teams. The framework is experimentally demonstrated on simulated games of robotic soccer.

We mathematically define and investigate the structure learning problem for a general, distributed dynamical system (the POSGDS model). This includes an analytical derivation of expected log-likelihood and KL divergence of a POSGDS for model selection. We further provide a decomposition of KL divergence and log-likelihood that illustrates their relationship to common information dynamics measures. We present scoring functions and algorithms for learning the structure of POSGDSs as well as their computational complexity. Finally, we give experimental validation of these scoring functions for coupled Lorenz and Rössler attractors.

We present and evaluate an end-to-end robotic system for autonomously performing wildlife telemetry tracking. This system includes a novel lightweight two-point phased array antenna that yields unambiguous bearing measurements onboard an aerial vehicle as well as custom electronics to sample and hold the analog output of the receiver. We provide rigorous mathematical derivation of the range-azimuth sensor model as well as the data fusion and informative path planning algorithms from first principles. The system is validated to yield accurate (to

within 30 m) estimates of stationary targets and shown to sufficiently track a common bird species (noisy miners). Through extensive field trials, we show that the system is capable of tracking the critically endangered swift parrots and, moreover, that it performs comparably to human trackers.

Finally, we present a decentralised variant of MCTS (termed Dec-MCTS) for information gathering with a team of robots. The original paper was a collaboration with another PhD candidate, Graeme Best, who initially proposed the general algorithm. The stated contributions to this work in this thesis include a discounted tree search algorithm for MCTS. This approach is based on the well-known algorithm upper confidence bounds applied to trees (UCT) (the discounted variant being termed D-UCT) and designed for scenarios where the reward distributions are changing. We prove that, when D-UCT is used as the tree search policy in MCTS, the regret at the root node grows logarithmically even while the reward distributions are changing. We further propose that the distributed optimisation process of probability collectives converges when combined with an asymptotic, anytime algorithm such as MCTS.

1.4 STRUCTURE OF THE THESIS

This thesis is organised in a hierarchical structure. In each contribution chapter, we investigate decision rules with information-theoretic utility functions. Although each chapter could be considered under the general framework of decision theory, we divide the thesis into two distinct parts (with two chapters each): one on model selection and one on planning algorithms. Within each part, we consider autonomous systems from either the physics perspective (i.e., time-invariant systems) or the computer science perspective (i.e., autonomous agents and multi-agent systems). Moreover, we are explicitly motivated by distributed systems in each chapter excluding Chapter 6. Finally, each chapter will start with a short abstract, reiterating how the chapter contents fit into the thesis narrative. Thus, it is recommended that each abstract is read in succession (before the body text) to get an idea of the flow of the thesis.

To provide background on the contributions of the later chapters, Chapters 2 and 3 cover related work and technical details on topics from complex systems science and robotics. Chapter 2 gives a thorough literature review of the subtopics covered in each chapter. Specifically, we discuss literature surrounding multi-agent dynamics, network inference, information gathering, and wildlife telemetry tracking. Chapter 3 then introduces necessary mathematical foundations of time series analysis and reasoning in distributed and autonomous systems. In particular, we

focus on models for stochastic processes as well as decision rules for model selection and robotic path planning problems.

In Chapters 4 and 5, we demonstrate the use of information theory in model selection problems for distributed systems. In Chapter 4 we consider the problem of studying implicit communication in multi-agent systems. In this context, we are unable to control the robots, but rather observe their behaviour and make tacit inferences about their interactions. We take a traditional statistical perspective, assuming the observed movement of each robot can be modelled as a finite-order Markov chain (as in Figs. 1.1a and 1.1b). In Chapter 5 we show that information measures can further be used for model selection in distributed nonlinear systems. We consider distributed dynamical systems whereby a latent state is observed through a filter (Fig. 1.1c). The objective is then to learn the structure of the system, i.e., the coupling between latent states.

In Chapters 6 and 7, we study robotic information gathering problems, where the autonomous agents take actions that minimise their uncertainty about the environment. The challenge here is to use this formulation to solve specific automation tasks. In Chapter 6, we present an aerial robot system for tracking small dynamic animals. We assume the travel cost is negligible in this application and thus can employ a typical greedy planner and focus on sensor modelling. Chapter 7 extends this analysis to present a decentralised planning algorithm for active perception, Dec-MCTS, where we are tasked with maximising information gain but also taking into account travel cost.

Finally, Chapter 8 concludes the thesis by highlighting the problems and challenges addressed therein. We discuss future work that would further coalesce the more traditional study of information measures in complex systems analysis with the problems faced in robotic active perception tasks.

2 RELATED WORK

2.1 OVERVIEW

Here we present literature that is relevant to the contributions of this thesis. We begin by introducing information dynamics as a topic of information theory that focusses on stochastic processes. The various approaches to studying coupling between complex systems is discussed, from functional connectivity to full multivariate analysis. This includes various subtopics such as Bayesian network (BN) structure learning, attractor reconstruction, and dynamic causal modelling (DCM). We then discuss how the above concepts have previously been applied to multi-agent systems and swarms. Following this, the problem of robotic information gathering is introduced. This primarily involves path planning algorithms for the single-robot and multi-robot case. We then focus on the more specific information gathering task of automated wildlife telemetry tracking. We motivate the use of a robotic system to solve this problem by presenting the current state-of-the-art tracking approaches and conclude the chapter by introducing existing platforms and algorithms intended for this purpose.

2.2 INFORMATION DYNAMICS

Information theory is widely regarded as a systematic approach to designing and studying complex self-organised systems (see the overview in [182]). The field was originally introduced to assess the compression limits of data in terms of the most basic computational primitives: storage, transmission, and modification of information [206]. Traditionally, these concepts are applied to random variables by computing probability distributions over the outcomes (see [157] for an introductory course). However, recently, the same logic has been applied to stochastic processes, where uncertainty can be quantified by considering the (finite or infinite) history of a system. We refer to the framework that explicitly conditions on the past of a process as *information dynamics* (after [145, 147, 149, 153]). Due to their conditional nature, these measures are not only used averaged across the entire realisation of a process, but additionally have meaning when computed pointwise in

time [151, 153]. The two measures that we use in this thesis are the time-averaged *transfer entropy* and *active information storage*.

Transfer entropy was originally introduced by Schreiber [202] to quantify the information transfer between nonlinear (finite-order Markov) systems. Schreiber took a predictive definition of information flow, where he proposed to compute the (predictive) transfer of data from a source to a target process. An intuitive approach to capture the temporal flow (i.e., the dynamics) is to compute the information in the past of a source variable that predicts the future of a target variable; however, this measure is undirected and thus incomplete [202]. Instead, transfer entropy captures the dynamics *and* the direction of information by computing the information contained in the source about the next state of the target that was not already in the target's past. Since its introduction, transfer entropy has found success in numerous fields, e.g., computational neuroscience [146, 245], multi-agent systems [56], financial markets [201], supply-chain networks [196], and biology [62].

Active information storage was introduced by Lizier et al. [153] to quantify the intrinsic computation within a system, i.e., the process' memory. Prior to this work, existing measures captured related but different storage capacities of a process. Specifically, statistical complexity [64] quantifies the total storage relevant to the future of a process. Excess entropy [63] captures the total storage actually used in the future of a process. Although both the above measures quantify the memory of a system in time, active information storage measure differs in that it captures the amount of storage currently in use by a process, rather than that used in the semi-infinite future. In this sense, it presents a Markovian view of information storage. Similar to transfer entropy, this measure has been used in computational neuroscience [74, 249] and biology [160, 247], as well as in the optimisation [67, 183] of artificial systems.

2.2.1 *Coherent structure in distributed communications*

In general, one of the defining features of complex computation is a coherent information structure. This is understood as some pattern or configuration appearing in a state-space formed by information-theoretic quantities, such as transfer entropy and excess entropy [152]. These information dynamics state-space diagrams are known to provide insights which are not immediately visible when the measures are considered in isolation. One example is a structure for a class of systems (such as logistic maps) that can be examined by plotting time-averaged excess entropy versus entropy rate while changing a system parameter [75]. Another example is a characterisation of complexity of distributed computation within the spatiotem-

poral dynamics of cellular automata via state-space diagrams formed by transfer entropy and active information storage of the cellular automata rules [152]. In this example, each point in the state-space quantifies both the communication and memory operations of a cellular automaton.

2.3 CONNECTIVITY ANALYSIS FOR DISTRIBUTED SYSTEMS

In many cases of practical interest, a complex system can be abstracted into arbitrary subsystems, some of which are statistically independent of one another. That is, the multivariate state of the system can be modelled by individual subsystems whose dynamics are given by set of either discrete-time maps or first-order ordinary differential equations (ODEs), called a flow. In the discrete-time formulation, a map can be obtained numerically by integrating differential equations or recording experimental data (observations) at discrete-time intervals [114]. These types of systems have been studied under several names, including complex dynamical networks [32], distributed dynamical systems [114, 203], master-slave configurations (or systems with a skew product structure) [124], and coupled maps [113]. There is significant interest in analysing characteristics such as stability and synchrony of coupled dynamical systems, however we will restrict our attention here to the process of learning statistical independences between subsystems.

The seminal work of Granger [95] sparked extensive interest in inferring the coupling between these types of distributed systems. Inspired by the earlier efforts of Wiener [253] on the predictability of multivariate stochastic processes, Granger defined causality in terms of the predictability of one system linearly coupled to another. This definition is now commonly referred to as Wiener-Granger causality [35] in order to differentiate the concepts of predictive causality from mechanistic causality (causal effect). Although a mechanistic function describing causal effect is ideal, inferring such a model is well-known to be intractable without intervening with the data (i.e., physically removing a link between subsystems) [176]. It is possible, however, to obtain an equivalence class of networks with the same Markov structure [50]. In spite of this knowledge, measures such as transfer entropy have received criticism over spuriously identifying causality [109, 140, 214]. As such, it is important to emphasise that information transfer is not akin to causal effect [144].

Depending on the specific application, measures of predictive causality can be used to analyse either directed functional connectivity (used in Chapter 4) or multivariate connectivity (used in Chapter 5).

2.3.1 *Directed functional connectivity*

The concept of *functional connectivity* refers to recovering statistical dependencies [84]. That is, functional connectivity is concerned with the relationship between pairs of variables (multiple bivariate analyses), rather than considering the entire multivariate system. Although functional connectivity often involves undirected connectivity measures, such as Pearson correlation, here we discuss measures that are inherently directed.

Granger causality is popular for identifying coupling however it assumes a linear statistical model and is considered insufficient for inferring coupling between dynamical systems due to inseparability [227]. It was recently shown that Granger causality is a specific case of transfer entropy, where the variables are assumed to be are linearly-coupled Gaussian systems (e.g., Kalman models) [18], rather than generic CPDs. The notion of using transfer entropy to infer functional connectivity has been used extensively in analysis of datasets obtained from neural recordings [72, 73, 146, 158, 223, 226, 244, 245, 250–252]. However, most of these results build on the work of Schreiber [202] by assuming the system is composed of observable finite-order Markov chains.

A more general case is where each subsystem comprises latent dynamics that are only observed through a filter. A number of measures have been proposed to infer coupling between distributed dynamical systems based on reconstruction theorems. Sugihara et al. [227] proposed convergent cross-mapping that involves collecting a history of observed data from one subsystem and uses this to predict the outcome of another subsystem. This history is the delay reconstruction map described by Takens' Delay Embedding Theorem [232]. Similarly, Schumacher et al. [203] used the Bundle Delay Embedding Theorem [221, 222] to infer causality and perform inference via Gaussian processes. Although the algorithms presented in these papers can infer driving subsystems in a spatially distributed dynamical system, the results obtained differ from our analysis in Chapter 5 as inference is not considered for an entire network structure, nor is a formal derivation presented.

2.3.2 *Multivariate connectivity*

A more involved approach to studying the connectivity of a complex system is to consider building a network for the full multivariate system. The task of multivariate connectivity analysis can be formalised by the BN structure learning problem. This task comprises two subproblems: *evaluating* the fitness of a graph (a scoring function), and *identifying* the optimal graph given this fitness criterion (a search

procedure) [50]. Although BNs are typically assumed static, i.e., the problem involves reasoning over random variables rather than stochastic processes, these networks have been generalised to represent arbitrary processes known as DBNs. Unfortunately, even with an accurate scoring function, the problem of structure learning is NP-complete [51].

The problem of evaluating the fitness of a graph represents similar challenges to that of developing measures for directed functional connectivity. However, unlike in function connectivity analysis, these measures must now be conditioned on the parents of existing variables in a graph to eliminate redundant links. A number of theoretically optimal techniques exist for the evaluation problem for BNs with complete data [33, 100, 135], which have been extended to DBNs [81]. However, this problem is particularly challenging in the case of partially observable systems, which include both latent and observed variables. With incomplete data such as this, the common approach in BN structure learning is to resort to approximations that find local optima, e.g., expectation-maximisation (EM) [81, 91].

In neuroscience, the objective of DCM is to infer the parameters of explicit dynamic models that cause (generate) data. In DCM, the set of potential models is specified *a priori*, (typically in the form of ODEs) and then scored via marginal likelihood or evidence. The parameters of these models include *effective connectivity* such that their posterior estimates can be used to infer coupling among distributed dynamical systems [242]. As a consequence these approaches can be used to recover networks that reveal the effective structure[1] of observed systems [174, 218]. In contrast, information-theoretic approaches do not require an explicitly specified model because the scoring function can be computed directly from the data. However, in Chapter 5, we do employ an implicit model, i.e., that our data are generated by generic functions, where the subsystems are coupled to create a directed acyclic graph (DAG).

More recently, researchers have used the additive noise model [107, 178] to infer unidirectional cause and effect relationships with observed random variables and find a unique DAG (as opposed to an equivalence class). These studies have been extended by exploring weakly additive noise models for learning the structure of systems of observed variables with nonlinear coupling [96]. However, the inference requires the models are differentiable (by e.g., using a Gaussian process model [107]), and are thus less general.

1 That is, the structure according to the pre-defined set of ODEs.

2.4 multi-agent systems and swarms

In Chapter 4, we study the behaviour of multi-agent systems using information dynamics. Specifically, we use transfer entropy to build networks of the implicit interactions between players (i.e., directed functional connectivity analysis), as well as study the coherent information structure of these players and how it relates to system performance. This approach is connected to two concepts previously explored in the literature: studying the information processing of swarms; and building explicit interaction networks from team dynamics.

2.4.1 *Information processing in swarms*

Quantitative analysis of information-processing attributes (in particular, swarming behaviour) is a rapidly expanding cross-disciplinary field, ranging from biology [61] and statistical mechanics [29] to swarm engineering [142, 179].

Wang et al. [247] recently used information dynamics to study the information processing within swarms. Their intention was to quantify information cascades within a simulated swarm by considering dynamic synchrony in collective motion of swarm individuals which do not exchange explicit messages [247]. The authors verify the hypothesis that the collective memory within a swarm can be captured by active information storage: higher values of storage are associated with higher levels of dynamic coordination. Furthermore, they show that cascading information waves that correspond to long range communications are captured by conditional transfer entropy. In other words, information transfer was shown to characterise the communication aspect of collective computation distributed within the swarm. A follow-up study compared such collective communications within two different swarms, one of which had a constraint imposed on the speed of its individuals [160]. The authors reported that the constrained swarm generated weaker information cascades and had more difficulties in self-organising into a coherent state. We build on such advances in Chapter 4 to detect and analyse implicit interactions within and between teams which are undertaking a specific collective task.

In other work, information transfer in a swarm of fish was quantified as the normalised angular deviation of group direction, showing that transfer of information and decision-making can occur in an animal group without explicit signals or individual recognition [61]. The maximum entropy model was used to establish that local pairwise interactions between birds are sufficient to correctly predict the propagation of order throughout entire flocks of starlings [29]. An intuitive

measure of information flow was used to identify behavioural strategy within simulated swarms, demonstrating swarm plasticity in response to changing environments [179].

2.4.2 *Networks for team dynamics*

Team sports are increasing being analysed using complex systems theory to better understand and evaluate performance [1, 246], as well as identify networks between players. For example, Fewell et al. [77] analysed basketball games as networks, where players are represented as nodes and passing density as edge weights: the resulting network captures ball movement, at different stages of the game. Their work studies network properties (degree centrality, clustering, entropy and flow centrality) across teams and positions, and attempts to determine whether differences in team offensive strategy can be assessed by their network properties. Strategic networks considered by the authors include only explicit interactions (such as passes) within a team, and not implicit or spatially long-ranged interactions, across teams.

Similar analysis was applied in the context of soccer, using passing data made available by FIFA during the 2010 World Cup [177]. The study constructed a static weighted directed graph for each team (the passing network), with vertices corresponding to players and edges to passes, in order to provide a direct visual inspection of a team's strategy. The passing network was visualised by placing the nodes in positions roughly corresponding to the players' formation on the pitch, and enabling and inspection of play patterns, hot-spots and potential weaknesses. Using different centrality measures, the relative importance of each player in the game was also inferred. This work, as well as the previous study of Duch et al. [70] which constructed and analysed networks with one node for shots on target and one for wide shots, are limited to static passing networks, and again do not reveal spatially long-ranged interactions across teams.

The multi-player dynamics of a soccer game was recently shown to exhibit self-similarities in the time evolution of player and ball positioning [120] (i.e., the dynamics are similar at a number of temporal scales). Specifically, the persistence time below which self-similarity holds has been estimated to be a few tens of seconds, implying that the volatility of soccer dynamics is an intrinsic feature of these games. Taking such volatility into account, the investigation by Vilar et al. [246] proposed a novel method of analysis that captures how teams occupy sub-areas of the field as the ball changes location. This study was important in focusing on the local dynamics of team collective behavior rather than individual player capa-

bilities. When applied to soccer (soccer) matches, the method suggested that players' numerical dominance in some local sub-areas is a key to "defensive stability" and "offensive opportunity". While the method rigorously used an information-theoretic approach (e.g. the uncertainty of the team numerical advantage across sub-areas was determined using Shannon entropy), it was not aimed at and did not produce interaction networks, either explicit or implicit.

2.4.3 *Robotic soccer as a case study*

In Chapter 4, we use RoboCup 2D Simulation League matches to exemplify our approach to detecting and quantifying dynamic interactions in a game. During the last two decades, the RoboCup initiative has essentially superseded chess [42] and, more recently, Go [209], as a benchmark for artificial intelligence. RoboCup (the "Robot Soccer World Cup") was first proposed in 1997 as a standard problem for the evaluation of theories, algorithms and architectures for artificial intelligence, robotics, computer vision, and several other related areas [122], with the overarching RoboCup goal of developing a team of humanoid robots capable of defeating the FIFA World Cup champion team (the "Millennium Challenge"). From the outset of the RoboCup effort it was recognised that RoboCup is different from the previous benchmarks (chess and Go), in several crucial elements: environment (static vs dynamic), state change (turn-taking vs real-time), information accessibility (complete vs incomplete), sensor readings (symbolic vs non-symbolic), and control (central vs distributed) [10]. Since 1997, this ambitious goal has been pursued along two general complementary paths [121]: physical robot league and software agent (simulation) league [168].

RoboCup 2D Soccer Simulation League specifically targets the research question of how the optimal collective dynamics can result from autonomous decision-making under constraints, set by tactical plans and teamwork (collaboration) as well as opponents (competition) [40, 127, 163, 187–189, 194, 225, 234, 248]. In answering this question it becomes important to measure the mechanisms for, and to discover the patterns of, dynamic spatiotemporal interactions between different players.

2.5 INFORMATION GATHERING

The literature presented above is concerned with understanding and modelling systems where observations are obtained passively (or implicitly). Preferably, however, we would obtain only *useful* information about some phenomena in order to

improve these models. When gathering this information with a robotic system, the task is to compute a path or trajectory that gathers the most pertinent information; this problem is known as *informative path planning* [235].

2.5.1 *Single-robot information gathering*

The problem of single-robot information gathering has been studied extensively over the last decade [97]. Informative path planning typically involves optimising actions of a robot over a limited time-horizon (myopic algorithms) or the entire sequence of future observations (nonmyopic algorithms). Thus, the general formulation of information gathering can be viewed as a POMDP, where sequential decision processes in which actions are chosen to maximise an objective function; this is known to be NP-hard [132].

Under certain assumptions, efficient nonmyopic solutions can be designed by exploiting problem-specific characteristics. For instance, analysis of submodular set functions [166] has shown that myopic planning can achieve near-optimal performance for entropy reduction problems [133]. That is, this property can be exploited to obtain an approximately optimal sequence of actions for a single-robot system by greedily selecting the most informative viewpoints at each decision step. Unfortunately, however, this is not suitable in all scenarios, e.g., when the objective function also takes into account the path cost. As a result such approaches are unsuitable when resources such as time or energy is constrained (e.g., search and rescue missions). Occasionally, the full nonmyopic solution can be computed optimally [25, 28] or approximated with little loss in performance [47], even when including such a path cost. However, in general these approaches are not applicable.

For developing general nonmyopic solutions online, MCTS is a promising approach because it efficiently searches over long planning horizons and is anytime (i.e., reasonable solutions can be generated before a computational budget is reached) [38]. In this approach, a tree is incrementally expanded from the current state, and while it does consider long planning horizons, actions in the near future are visited more often by the search. MCTS has the advantage of being anytime, which is useful in time-critical applications. Moreover, the algorithm has also been extended to partially observable environments [210]. The algorithm has been proposed in many different forms [38] but by far the most common is the UCT algorithm [125, 126]. The UCT algorithm performs an asymmetric expansion of a search tree using a best-first policy that generalises the UCB1 policy for multi-armed bandit (MAB) problems [11]. This expansion policy provides theoretical guarantees for

a polynomial bound on regret and therefore is said to balance between exploration (of unknown but potentially optimal paths) and exploitation (of currently promising paths). Several variants to UCT have been proposed, such as for exploiting smoothness of the reward function [60].

In the above approaches, a discrete set of future actions were assumed to be known *a prior*. For large or continuous configuration spaces, sampling-based methods are beginning to be explored [104] for information gathering. These methods typically involve taking samples from the configuration space and testing them for optimality with respect to a local planner.

In Chapter 6, we specifically study the problem of wildlife tracking with a unmanned aerial vehicle (UAV). In this scenario, we are able to exploit submodularity since localisation requires a small number of good quality observations, and thus path cost is ignored. This result is not considered a major contribution, since many challenges have been addressed specifically for UAVs path planning problems [103, 123, 167, 200, 212, 236]. The problem of information gathering with a UAV has further been studied using formal methods [262, 263] and multi-UAV constrained search [86].

2.5.2 *Decentralised information gathering*

The problem of informative path planning in a decentralised manner is compounded because the search space grows exponentially with the number of robots. Similar to single-robot information gathering, the general problem is a Dec-POMDP, and is also NP-hard. Thus, decentralised coordination in these problems is typically solved by maximising the objective function myopically [86, 260]. Unfortunately, the quality of solutions produced by these methods can be arbitrarily poor in the general case. The concept of exploiting submodularity of the objective function has led to considerable interest in their application to information gathering with multiple robots [87, 211]. As with the single-robot scenario, however, while these methods provide theoretical guarantees, they require a submodular objective function, which is not applicable in all cases.

As mentioned in Sec. 2.5.1, MCTS is promising for online planning, however it has not been extended for decentralised multi-agent planning, and that is our focus in Chapter 7. A key component of our proposed Dec-MCTS algorithm is a novel, discounted UCT variant, D-UCT, that accounts for a changing reward distribution by using a new expansion policy that generalises a MAB policy designed for switching bandit problems [88], i.e., problems where the reward distribution is changing.

MCTS is parallelisable [48], and various techniques have been proposed that split the search tree across multiple processors and combine their results. In the multi-robot case, the joint search tree interleaves actions of individual robots and it remains a challenge to effectively partition this tree. A related case is multi-player games, where a separate tree may be maintained for each player [12]; however, a single simulation traverses all of the trees and therefore this approach would be difficult to decentralise. We propose a similar approach, except that each robot performs independent simulations while sampling from a locally stored probability distribution that represents the other robots' action sequences.

As mentioned above, MCTS algorithms have been extended for problems with partial-observability, such as the POMCP [210] and DESPOT [215] algorithms, and Dec-MCTS could be extended in a similar way. In Chapter 7, however, we focus our attention on reasoning over the unknown plans of the other robots, while assuming other aspects of the problem are fully observable. The decentralised information gathering problems we consider are not Dec-POMDP but our algorithm is general enough to be extended to problems with partial observability.

Coordination between robots is achieved in Dec-MCTS by combining MCTS with a framework that optimises a product distribution over the joint action space in a decentralised manner. Our approach is analogous to the classic mean-field approximation and related variational approaches [193, 261]. Variational methods seek to approximate the underlying global likelihood with a collection of structurally simpler distributions that can be evaluated efficiently and independently. These methods characterise convergence based on the choice of product distribution, and work best when it is possible to strike a balance between the convergence properties of the product distribution and the KL divergence between the product and joint distributions. As discussed in the body of work on probability collectives [255–257], such variational methods can also be viewed under a game theoretic interpretation, where the goal is to optimise each agent's choice of actions based on examples of the global reward/utility function. The latter method has been used for solving the multiple travelling salesman problem in a decentralised manner [134]; we propose a similar approach, but we leverage the power of the MCTS to select an effective and compact sample space of action sequences.

2.6 WILDLIFE TELEMETRY TRACKING

An important application of information gathering is *environmental monitoring*, which typically involves robots acting in the field and thus handling a high level of un-

certainty. In Chapter 6, we present an autonomous system for the purpose of understanding the movement patterns of critically endangered birds.

The problem of tracking small radio-tagged animals using an autonomous and lightweight aerial robot has recently gained interest in a variety of academic communities. Conservation management of certain critically endangered species relies on the process of detecting and tracking the position of individual animals in the wild [108, 116, 165]. Aerial robot systems can access rugged areas that are difficult for humans to traverse, and thus are viewed as a potentially revolutionary tool for data collection in wildlife ecology [45, 94]. However, this potential remains largely unrealised. Robot systems have yet to achieve a level of tracking accuracy and speed that is sufficient to legitimise their role as a replacement for human trackers.

Despite recent advances in automated wildlife telemetry tracking, very little is known about the movement of small, dynamic migratory species, of which many have reached critically endangered status. For large to medium animals, the miniaturisation of GPS tags with remote data readout has facilitated a dramatic increase in understanding the movements of a diversity of species [173, 267]. Methods such as satellite telemetry have far reaching applications from investigating migration routes and wintering areas of large migratory birds [21, 118, 171] to studying the dynamics of aquatic predators [31, 181]. Unfortunately, these approaches are still only suitable for about 70% of bird species and 65% of mammal species [116]. In the case of smaller species that return to the same breeding areas seasonally, miniature non-transmitting data loggers can be used [116]; however, retrieving this data requires relocating the animals *in situ*. Due to this challenge, VHF tracking has become one of the most useful techniques in ecology and management [36]. This involves instrumenting animals with small radio transmitters and subsequently tracking the target species. Although scientists have been using VHF tracking since the early 1960s [155], data yielded by this approach is sparse due to the manual labour involved [116]. Thus, researchers are more frequently exploiting the abundance of low-cost UAVs for this type of conservation management and wildlife monitoring [45, 94].

2.6.1 *Robotic tracking*

In recent years, there has been increased interest in end-to-end wildlife telemetry tracking with robotic systems [45], where the robot moves autonomously to track a live target animal. The usefulness of these systems, however, is yet to be proven in direct performance comparison to the traditional manual approach. Most notably, ongoing research is aimed at tracking radio-tagged carp in Minnesotan lakes

using autonomous surface vehicles (ASVs) on the water and mobile ground vehicles (MGVs) when the lake is frozen [20, 106, 169, 237, 238, 243]. While this project has yielded seminal work in the field, the use of ground and surface vehicles is untenable for wildlife situated in rugged habitats.

Small aerial robots such as multirotor platforms are suitable for wildlife tracking because they are easy to deploy and can fly over terrain that is difficult to access on foot, potentially reducing localisation time from hours in the manual case to tens of minutes. Further, small aerial robots can operate from sufficient distance to not disturb wildlife. However, it is difficult to design and model a high-performance antenna system that is light enough to be carried by such systems. Popular loop aerials [106, 237] are known to be inefficient, especially for low frequency signals. Standard horizontally-mounted directional antennas [129, 180] are affected by UAV rotors which cause unpredictable irradiance. As a consequence we designed and built a novel lightweight antenna for use on-board a UAV. This hardware yields unambiguous bearing-only observations and is shown to be sufficiently sensitive for our field experiments.

The majority of research in radio tracking with an aerial vehicle focuses on isolated subsystems. Although these systems are typically motivated by the idea of tracking small animals (e.g., bird [69, 129, 136, 180] and fish species [111, 112]), only simulations or prototypes are presented with limited field testing. Alternatively, when tracking a relatively stationary target, the observations can be considered more robust and thus attention in this field has shifted to optimising planning for single [20, 106, 169] or multi-robot systems [243]. The main assumption the authors make is that the sequential observations are *homoscedastic*, meaning that the uncertainty over each measurement is constant or bounded. However, with a sporadic and unpredictable live target, this assumption is violated due to the resulting wide spectrum of observation quality from noisy to precise. As we show later, this induces *heteroscedastic* observations, where the uncertainty varies with every observation. Failing to distinguish between low and high quality observations can lead to overconfident measurements that cause spurious location estimates, or to highly uncertain location estimates that are of little value.

A mathematically valid observation model is also critical in planning the motion of the robot to improve the location estimate. In robotics this general problem is known as *active perception* [16, 17] and introduces a coupling between data collection and planning. The idea of passively locating transmitting radio sources has been investigated in operations research motivated by search and rescue missions where stationary distress beacons must be recovered rapidly. Hence, the task is a coverage problem solved via offline strategies with an emphasis on minimising path cost over the entire area or teleoperated by humans [143]. Alternatively, when

the wildlife habitat is known and bounded, sensor networks can be placed in order to precisely track the animals location [41, 117]. In our case, we require fast, precise estimates without intervention and thus employ active strategies where the observation quality relies crucially on an appropriate sequence of viewpoints [175]. Our objective is reduce uncertainty (entropy) of the target location; thus, the task of actively tracking targets falls under informative path planning (see Sec. 2.5). The problem of designing an online estimator for radio localisation and tracking is well studied [137–139], with emphasis on ground-based systems and an assumed sensor model. Moreover, extensive research in online estimation is coupled with optimal sensor placement and planning in [80] and [103].

2.7 SUMMARY

This chapter presented methods for studying the communication and storage of multivariate processes (from data). Following this, we introduced numerous methods designed to optimise the quality of a data stream via robotic systems. In doing so, we have presented a comprehensive survey on the multidisciplinary subject of information-theoretic reasoning by various covering topics from robotics, distributed optimisation, and complex systems science. In the next chapter we will elaborate on some of these approaches by providing technical details on time series analysis and decision rules for selecting models and informative paths.

3 BACKGROUND ON TIMES SERIES MODELLING AND DECISION THEORY

3.1 OVERVIEW

This chapter provides the technical background and nomenclature used throughout this thesis. We first describe the two main perspectives on time series analysis: applied statistics and physics (with a particular emphasis on attractor reconstruction theorems). We then introduce information theory for stochastic processes as a means for quantifying uncertainty, given the realisation of the process (data). Following this, using the framework of decision theory, we formally introduce the two main concepts studied in this thesis: model selection and planning under uncertainty. Finally, we present DBNs and discuss learning their network structure from data.

3.2 NOMENCLATURE

We draw on both complex systems and robotics literature in this thesis and thus occasionally use conflicting nomenclature. In general, we consider multivariate time series generated by a discrete-time stochastic process, the convention is that $(\cdot)$ denotes a sequence, $\{\cdot\}$ a set, and $\langle\cdot\rangle$ a vector. We follow typical statistics notation in that upper case letters denote stochastic variables and lower case letters are the associated realisations of these variables.

Given a distributed system, the process $\boldsymbol{Z}$ is typically abstracted into M components (often termed subsystems)[1], i.e., $\boldsymbol{Z} = \{\boldsymbol{Z}^1, \ldots, \boldsymbol{Z}^M\}$. Each subsystem process $\boldsymbol{Z}^i$ comprises a sequence of random variables $(\boldsymbol{Z}^i_1, \ldots, \boldsymbol{Z}^i_N)$ with realisation $(z^i_1, \ldots, z^i_N)$ for countable time indices $n \in \mathbb{N}$. We use bold to denote any variable that is non-scalar (or unspecified) and will often marginalise out indices if it is clear based on context, e.g., $z_n = \{z^1_n, \ldots, z^M_n\}$ is the collection of M realisations at index n. In general, functions will not be bold regardless of their output dimension, this includes parameters or parameter sets (typically being functions of random variables themselves).

1 These components are typically multivariate and of arbitrary dimension abstracted from context (e.g., robots in a multi-robot system or regions in an fMRI scan).

In Chapters 4 and 5, we use the standard DBN notation [164]. That is, $\boldsymbol{X}_n^i$ is a latent (hidden) variable, $\boldsymbol{Y}_n^i$ is an observed variable, and $\boldsymbol{Z}_n^i$ is an arbitrary variable; here, $\boldsymbol{Z}_n = \{\boldsymbol{X}_n, \boldsymbol{Y}_n\}$ is the set of all hidden and observed variables at temporal index n.

In Chapters 6 and 7, we use the standard robotics notation that $\boldsymbol{X}_n$ is the robot state and $\boldsymbol{Y}_n$ is the quantity of interest. This is historically due to the fact that certain robotics applications do not assume complete knowledge of the state or consider the robot position to be intrinsic to the environment that they are observing [235]. In our work, however, it is notationally clearer to distinguish the robot state from the environment state (e.g., the animal or object of interest).

Finally, we typically use $p(\boldsymbol{Z})$ to denote a probability distribution over an arbitrary random variable $\boldsymbol{Z}$, i.e., $p(\boldsymbol{z}) = p(\boldsymbol{Z} = \boldsymbol{z})$. If necessary, we will use $p_\theta(\boldsymbol{Z})$ to specify that the distribution depends on some set of parameters θ. These parameters are often learned, e.g., via maximum likelihood or density estimation techniques [130, 131].

3.3 TIME SERIES ANALYSIS

The traditional school of thought on time series analysis views the data as sample paths of a *stochastic process* and is a branch of applied statistics. In this framework, the mechanisms for generating the process are typically assumed arbitrary and the focus is on modelling the distributions over the dynamics. In physical systems, however, we sometimes have models of mechanistic functions that describe some phenomena over time such that the time series can be considered a distorted realisation of this behaviour. When viewed under the physics paradigm, a *dynamical system* is often used to define the rule for the evolution of the state [205].

In the statistics perspective, measurements are associated with a random variable $\boldsymbol{Z}_n$ that defines a distribution over the outcomes at time n. A stochastic process is a sequence of these variables $(\boldsymbol{Z}_n)$ where n is in the index set (typically a subset of the real line). A single outcome of this sequence of variables is called the realisation or sample path $(\boldsymbol{z}_n)$. To describe models of how these processes are generated, we follow the state space representation.

The *state* $\boldsymbol{x}_n$ is a point in some state-space $\mathcal{M}$ (or *phase space*, in dynamical systems literature). In discrete time, the evolution of this state is described by a *map* $f : \mathcal{M} \times \mathcal{M} \times \mathbb{N} \to \mathcal{M}$, so that the sequence of states $(\boldsymbol{x}_n)$ is given by

$$\boldsymbol{x}_{n+1} = f_n(\boldsymbol{x}_n, \boldsymbol{\omega}_n), \quad n \in \mathbb{N}, \tag{3.1}$$

where, as implied, the map f_n can depend on time n and a sequence of independent variables $(\boldsymbol{\omega}_n)$. In continuous time, the map is replaced by a *flow* $\varphi : \mathcal{M} \times \mathcal{M} \times \mathbb{R} \to \mathcal{M}$ that describes the rate of change of the state

$$\dot{\boldsymbol{x}}_t = \phi_t(\boldsymbol{x}_t, \boldsymbol{\omega}_t), \qquad t \in \mathbb{R}, \tag{3.2}$$

where the flow can change over time. In this thesis we focus mainly on the discrete-time definition (3.1), which can be given by simulation, obtained from (3.2) by integrating the flow, or by sampling a physical system at discrete time intervals. In this framework, the future of x_n is fully determined by the current state and rule. In statistical nomenclature, this is referred to as a *Markov process* and is characterised by the following independence:

$$p\left(\boldsymbol{X}_{n+1} = \boldsymbol{x}_{n+1} \mid \boldsymbol{x}_n, \boldsymbol{\omega}_n, \ldots, \boldsymbol{x}_0, \boldsymbol{\omega}_0\right) = p\left(\boldsymbol{X}_n = f(\boldsymbol{x}_n, \omega_n) \mid \boldsymbol{x}_n, \boldsymbol{\omega}_n\right). \tag{3.3}$$

That is, in physics literature, the Markov property is a direct result of the definition of state. We will assume that f_n is time-invariant (autonomous), i.e., the probability distributions in (3.3) are homogeneous.

In many situations of practical interest, we only have access to a filtered representation of the state, i.e., we observe $\boldsymbol{y}_n$ in some *measurement space* $\mathcal{N}$, which is, in general, a noisy, nonlinear function of the state

$$\boldsymbol{y}_n = \psi(\boldsymbol{x}_n, \boldsymbol{\epsilon}_n). \tag{3.4}$$

Here, $\psi : \mathcal{M} \times \mathcal{N}$ is called the *measurement function*[2] that distorts the observation by some noise process $(\boldsymbol{\epsilon}_n)$.

The typical objective of time series analysis is thus, given the sequence of observations $(\boldsymbol{y}_n)$, determine the phase space $\mathcal{M}$, dynamics f, observation function ψ, and noise processes [205, 232]. In theory this is achievable, however, no general framework has yet been developed. As a result, the study of time series relies on making assumptions about these quantities, predicting the outcomes based on these hypothesises, and validating these predictions for plausibility.

The traditional statistics approach to time series analysis was to assume stationary processes, whereas physics and economics allowed for a degree of non-stationarity. Around the 1970s, these two efforts were combined with methods such as ARIMA, which accounted for the autoregressive (AR), integrated (I), and moving average (MA) components of a time series [34].

Around this time, Lorenz developed a mathematical model for weather forecasting [156]. When simulating this simple nonlinear system, he recognised that small changes in initial conditions can dramatically affect the later state of the system.

2 We sometimes refer to the measurement function as the read-out function or filter.

As a result, a new branch of mathematics emerged known as *chaos theory* (often referred to these days as nonlinear time series analysis [114]). This indicated that methods such as ARIMA are not applicable to these systems. However, in certain cases, we can exploit our *a priori* knowledge of the system (f, ψ) to perform time series analysis.

3.3.1 *The nonlinear approach*

Shortly afterward, Takens [232] produced seminal results in reconstructing the phase space of a dynamical system, given only the observed sequence. In doing so, he founded *embedding theory*: the study of inferring the (hidden) state $\boldsymbol{x}_n \in \mathcal{M}$ of a dynamical system from a sequence of scalar observations $y_n \in \mathbb{R}$. This section will cover reconstruction theorems that define the conditions under which we can use delay embeddings for recovering the original dynamics f from this observed time series.

In order to introduce this theory, we require some additional notation for embedding a time series with a time delay. For process $\boldsymbol{Y} = (Y_1, \ldots, Y_N)$ with realisation $\boldsymbol{y} = (y_1, \ldots, y_N)$, we define a delay vector $\Psi^{\kappa}_{\tau,n} : \mathcal{M} \to \mathbb{R}^{\kappa}$ that maps the realisation of the the process at a given time to a vector:

$$\Psi^{\kappa}_{\tau,n}(\boldsymbol{y}) := (y_n, y_{n-\tau}, y_{n-2\tau}, \ldots, y_{n-(\kappa-1)\tau}),$$

for some *time delay* $\tau \in \mathbb{N}$ and *embedding dimension* $\kappa \in \mathbb{N}$ (the *embedding parameters*). To simplify notation, we will assume the time delay $\tau = 1$; however, the case of arbitrary τ can be treated equivalently. Henceforth, we drop the time delay subscript and let $\Psi^{\kappa}_{n}(\boldsymbol{y}) = \Psi^{\kappa}_{1,n}(\boldsymbol{y})$. For a collection of M processes $\boldsymbol{Y} = \{Y^1, \ldots, Y^M\}$ with realisation $\boldsymbol{y} = \{\boldsymbol{y}^1, \ldots, \boldsymbol{y}^M\}$, let

$$\Psi^{\{\kappa_i\}}_{n}(\boldsymbol{y}) := (\Psi^{\kappa_1}_{n}(\boldsymbol{y}^1), \ldots, \Psi^{\kappa_M}_{n}(\boldsymbol{y}^M)),$$

where the set $\{\kappa_i\} = \{\kappa_1, \ldots, \kappa_M\}$. Occasionally, it will be more convenient to use a specific scalar value, e.g., $\Psi^{k}_{n}(\boldsymbol{y})$ to denote a constant embedding of k for all processes, i.e., $\kappa_1 = k, \kappa^2 = k, \ldots, \kappa^M = k$.

In differential topology, an *embedding* refers to a smooth map $\Phi : \mathcal{M} \to \mathcal{N}$ between manifolds $\mathcal{M}$ and $\mathcal{N}$ if it maps $\mathcal{M}$ diffeomorphically onto its image. In Takens' seminal work on turbulent flow [232], he proposed a map $\Phi_{f,\psi} : \mathcal{M} \to \mathbb{R}^{\kappa}$, that is composed of delayed observations, can be used to reconstruct the dynamics for typical (f, ψ). That is, fix some κ (the embedding dimension) and τ (the time delay), the *delay embedding map*, given by

$$\Phi_{f,\psi}(\boldsymbol{x}_n) = \boldsymbol{y}^{(\kappa)}_n = (y_n, y_{n+\tau}, y_{n+2\tau}, \ldots, y_{n+(\kappa-1)\tau}), \tag{3.5}$$

is an embedding. Here, we have introduced the shorthand notation $\boldsymbol{y}_n^{(\kappa)}$ which will be used commonly throughout this thesis. More formally, denote $\Phi_{f,\psi}$, $\mathcal{D}^r(\mathcal{M},\mathcal{M})$ as the space of C^r-diffeomorphisms on $\mathcal{M}$ and $C^r(\mathcal{M},\mathbb{R})$ as the space of C^r-functions on $\mathcal{M}$, then the theorem can be expressed as follows.

Theorem 3.1 (Delay Embedding Theorem for Diffeomorphisms [232]). *Let $\mathcal{M}$ be a compact manifold of dimension $d \geq 1$. If $\kappa \geq 2d+1$ and $r \geq 1$, then there exists an open and dense set $(f,\psi) \in \mathcal{D}^r(\mathcal{M},\mathcal{M}) \times C^r(\mathcal{M},\mathbb{R})$ for which the map $\Phi_{f,\psi}$ is an embedding of $\mathcal{M}$ into $\mathbb{R}^\kappa$.*

The implication of Theorem 3.1 is that, for typical (f,ψ), the image $\Phi_{f,\psi}(\mathcal{M})$ of $\mathcal{M}$ under the delay embedding map $\Phi_{f,\psi}$ is completely equivalent to $\mathcal{M}$ itself, apart from the smooth invertible change of coordinates given by the mapping $\Phi_{f,\psi}$. An important consequence of this result is that we can define a map $\mathrm{F} = \Phi_{f,\psi} \circ f \circ \Phi_{f,\psi}^{-1}$ on $\Phi_{f,\psi}$, such that $y_{n+1}^{(\kappa)} = \mathrm{F}(y_n^{(\kappa)})$ [220]. That is, the bound for the open and dense set referred to in Theorem 3.1 is given by a number of technical assumptions. Denote $(Df)_{\boldsymbol{x}}$ as the derivative of function f at a point $\boldsymbol{x}$ in the domain of f. The set of periodic points A of f with period less than τ has finitely many points. In addition, the eigenvalues of $(Df)_{\boldsymbol{x}}$ at each $\boldsymbol{x}$ in a compact neighbourhood A are distinct and not equal to 1.

Theorem 3.1 was established for diffeomorphisms $\mathcal{D}^r$; by definition the dynamics are thus invertible in time. So the time delay τ in (3.5) can be either positive (delay lags) or negative (delay leads). Takens later proved a similar result for endomorphisms, i.e., non-invertible maps that restricts the time delay to a negative integer. Denote by $\mathcal{E}(\mathcal{M},\mathcal{M})$ the set of the space of $\mathcal{C}^r$-endomorphisms on $\mathcal{M}$, then the reconstruction theorem for endomorphisms can be expressed as the following.

Theorem 3.2 (Delay Embedding Theorem for Endomorphisms [233]). *Let $\mathcal{M}$ be a compact d dimensional manifold. If $\kappa \geq 2d+1$ and $r \geq 1$, then there exists an open and dense set $(f,\psi) \in \mathcal{D}^r(\mathcal{M},\mathcal{M}) \times C^r(\mathcal{M},\mathbb{R})$ for which there is a map $\pi_\kappa : \mathcal{X}_\kappa \to \mathcal{M}$ with $\pi_\kappa \Phi_{f,\psi} = f^{\kappa-1}$. Moreover, the map π_κ has bounded expansion or is Lipschitz continuous.*

As a result of Theorem 3.2, a sequence of κ successive measurements from a system determines the system state *at the end* of the sequence of measurements [233]. That is, there exists an endomorphism $\mathrm{F} = \Phi_{f,\psi} \circ f \circ \Phi_{f,\psi}^{-1}$ to predict the next observation if one takes a negative time delay (lead) τ in (3.5).

In Chapter 5, we consider two important generalisations of the Delay Embedding Theorem for Diffeomorphisms 3.1. Both of these theorems follow similar proofs to the original and have thus been derived for diffeomorphisms, not endomorphisms. However, encouraging empirical results in [203] support the conjecture that they

can both be generalised to the case of endomorphisms by taking a negative time delay, as is done in Theorem 3.2 above. This would allow for not only distributed flows that are used in Chapter 5, but endomorphic maps, e.g., the well-studied coupled map lattice structure [154].

The first generalisation is by Stark et al. [220] and deals with a skew-product system. That is, f is now forced by some second, independent system $g : \mathcal{N} \to \mathcal{N}$. The dynamical system on $\mathcal{M} \times \mathcal{N}$ is thus given by the set of equations

$$x_{n+1} = f(x_n, \omega_n) \tag{3.6}$$

$$\omega_{n+1} = g(\omega_n). \tag{3.7}$$

In this case, the delay map is written as

$$\Phi_{f,g,\psi}(x, \omega) = (y_n, y_{n+\tau}, y_{n+2\tau}, \ldots, y_{n+(\kappa-1)\tau}), \tag{3.8}$$

and the theorem can be expressed as follows.

Theorem 3.3 (Bundle Delay Embedding Theorem [220])**.** *Let $\mathcal{M}$ and $\mathcal{N}$ be compact manifolds of dimension $d \geq 1$ and e respectively. Suppose that $\kappa \geq 2(d+e)+1$ and the periodic orbits of period $\leq d$ of $g \in \mathcal{D}^r(\mathcal{N})$ are isolated and have distinct eigenvalues. Then, for $r \geq 1$, there exists an open and dense set of $(f, \psi) \subset \mathcal{D}^r(\mathcal{M} \times \mathcal{N}, \mathcal{M}) \times \mathcal{C}^r(\mathcal{M}, \mathbb{R})$ for which the map $\Phi_{f,g,\psi}$ is an embedding of $\mathcal{M} \times \mathcal{N}$ into $\mathbb{R}^\kappa$.*

Finally, all theorems up until now have assumed a single read-out function for the system in question. Recently, Deyle et al. [68] showed that multivariate mappings also form an embedding, with minor changes to the technical assumptions underlying Takens' original theorem. That is, given $M \leq 2d+1$ different observation functions, the delay map can be written as

$$\Phi_{f,\{\psi^i\}}(\boldsymbol{x}) = (\Phi_{f,\psi^1}(\boldsymbol{x}), \Phi_{f,\psi^2}(\boldsymbol{x}), \ldots, \Phi_{f,\psi^M}(\boldsymbol{x})), \tag{3.9}$$

where each delay map Φ_{f,ψ^i} is as per (3.5) for individual embedding dimension $\kappa^i \leq \kappa$. The theorem can then be stated as follows.

Theorem 3.4 (Delay Embedding Theorem for Multivariate Observation Functions [68])**.** *Let $\mathcal{M}$ be a compact manifold of dimension $d \geq 1$. Consider a diffeomorphism $f \in \mathcal{D}^r(\mathcal{M}, \mathcal{M})$ and a set of at most $2d+1$ observation functions $\{\psi^i\}$ where each $\psi^i \in C^r(\mathcal{M}, \mathbb{R})$ and $r \geq 2$. If $\sum_i \kappa^i \geq 2d+1$, then, for generic $(f, \{\psi^i\})$, the map $\Phi_{f,\{\psi^i\}}$ is an embedding.*

3.4 INFORMATION THEORY

Information theory was originally introduced by Claude Shannon for the purposes of basic signal processing and investigating the limits of data compression [206,

207]. We will first introduce the fundamental information-theoretic measures, and then discuss quantities used for studying information dynamics of distributed computation that are used in Chapters 4 and 5 of this thesis.

3.4.1 *Entropy and KL divergence*

Consider an *observed* random variable $\boldsymbol{X}$. The *entropy* quantifies the amount of uncertainty over the outcome $\boldsymbol{x}$ of that random variable:

$$\begin{aligned} H(\boldsymbol{X}) &= -\sum_{\boldsymbol{x}} p(\boldsymbol{x}) \log p(\boldsymbol{x}) \\ &= \mathbf{E}\left[-\log p(\boldsymbol{X})\right]. \end{aligned} \tag{3.10}$$

The uncertainty of these outcomes may be reduced depending on the outcome of another (observed) random variable $\boldsymbol{Y}$. All logarithms in this thesis are taken by convention in base 2, giving the units of entropy (3.10) and quantities in bits. To quantify this notion, the *conditional entropy* $H(\boldsymbol{X} \mid \boldsymbol{Y})$ then represents the uncertainty of $\boldsymbol{X}$ after taking into account the outcomes of another random variable $\boldsymbol{Y}$:

$$\begin{aligned} H(\boldsymbol{X} \mid \boldsymbol{Y}) &= -\sum_{\boldsymbol{x},\boldsymbol{y}} p(\boldsymbol{x},\boldsymbol{y}) \log p(\boldsymbol{x} \mid \boldsymbol{y}) \\ &= \mathbf{E}\left[-\log p(\boldsymbol{X} \mid \boldsymbol{Y})\right]. \end{aligned} \tag{3.11}$$

Moreover, in order to capture the mutual dependence (nonlinear correlation) between these two variables, we define the *mutual information* as

$$\begin{aligned} I(\boldsymbol{X};\boldsymbol{Y}) &= \mathbf{E}\left[-\log \frac{p(\boldsymbol{X} \mid \boldsymbol{Y})}{p(\boldsymbol{X})}\right] \\ &= H(\boldsymbol{X}) - H(\boldsymbol{X} \mid \boldsymbol{Y}). \end{aligned} \tag{3.12}$$

Another fundamental concept in information theory that we will use is the KL divergence. Given two probability distributions p and p_θ defined over the same sample space, the KL divergence from p_θ to p is

$$D_{\mathrm{KL}}\left[p \parallel p_\theta\right] = -\sum_{\boldsymbol{x}} p(\boldsymbol{x}) \log \frac{p_\theta(\boldsymbol{x})}{p(\boldsymbol{x})} \tag{3.13}$$

$$= \mathbf{E}\left[-\log \frac{p_\theta(\boldsymbol{X})}{p(\boldsymbol{X})}\right] \tag{3.14}$$

Typically, information-theoretic quantities can be defined in terms of the KL divergence. For example, Shannon entropy (3.10) can be expressed as the KL divergence of p from the uniform distribution plus a constant. As a result, KL divergence is often referred to as *relative entropy* or *information gain*, depending on context.

The above measures can be defined over either sequences of random variables z, or simply the variables themselves z_n. In the next section we are specifically concerned with measures applied to a sequence of variables.

3.4.2 *Information dynamics of distributed systems*

The following measures are, in general, used to investigate the information dynamics of distributed computation [147] where the processes are assumed to be stationary. For notational convenience, each subsystem process $\boldsymbol{Y}^i$ below is considered univariate with realisation $Y_n^i \in \mathbb{R}$, however the measures described generalise to variables of arbitrary dimension (see, e.g., [46, 73, 146, 150]).

One of the most intuitive information measures for stochastic processes is *excess entropy* [63], which indicates the amount of predictability of the future of a process from its past (thus, it is sometimes referred to as *predictive information*). Consider the process $\boldsymbol{Y}^i$, the excess entropy of this process is quantified by the mutual information between the (semi-infinite) past $\boldsymbol{y}_n^- = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$ and the (semi-infinite) future $\boldsymbol{y}_n^+ = (\boldsymbol{y}_{n+1}, \ldots, \boldsymbol{y}_N)$ of a system:

$$
\begin{aligned}
E_{\boldsymbol{Y}} &= \mathbf{E}\left[\log \frac{p(\boldsymbol{Y}_n^+, \boldsymbol{Y}_n^-)}{p(\boldsymbol{Y}_n^+)p(\boldsymbol{Y}_n^-)}\right] \\
&= I(\boldsymbol{Y}_n^+; \boldsymbol{Y}_n^-) && (3.15)\\
&= H(\boldsymbol{Y}_n^+) - H(\boldsymbol{Y}_n^+ \mid \boldsymbol{Y}_n^-). && (3.16)
\end{aligned}
$$

Alternatively, if the system is assumed κ-order Markov, we can consider a finite history length:

$$
\begin{aligned}
E_{\boldsymbol{Y}}(\kappa) &= \mathbf{E}\left[\log \frac{p(\boldsymbol{Y}_n, \boldsymbol{Y}_n^{(\kappa)})}{p(\boldsymbol{Y}_n)p(\boldsymbol{Y}_n^{(\kappa)})}\right] \\
&= I(\boldsymbol{Y}_{n+1}; \boldsymbol{Y}_n^{(\kappa)}) && (3.17)\\
&= H(\boldsymbol{Y}_{n+1}) - H(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^{(\kappa)}). && (3.18)
\end{aligned}
$$

Here, as in the previous section, the shorthand notation $\boldsymbol{Y}_n^{(\kappa)} = (\boldsymbol{Y}_n, \boldsymbol{Y}_{n-1}, \ldots, \boldsymbol{Y}_{n-\kappa+1})$, i.e., the sequence of κ previous values taken by random vector/variable $\boldsymbol{Y}_n$. All remaining information measures can be computed in this way and, in this thesis, we will remove the dependence on κ unless it is necessary.

For univariate processes, the *active information storage* $A_{\boldsymbol{Y}}$ quantifies the information storage component that is directly in use in the computation of the next value of a sequence [153]. More precisely, active information storage is the average mu-

tual information between the (semi-infinite) past state of the process and its next value:

$$A_{\boldsymbol{Y}^i} = \mathbf{E}\left[\log \frac{p(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n)}{p(Y^i_{n+1})}\right] \tag{3.19}$$

$$= I(Y^i_{n+1}; \boldsymbol{Y}^{i,-}_n) \tag{3.20}$$

$$= H(Y^i_{n+1}) - H(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n). \tag{3.21}$$

As was the case in quantifying excess entropy, we can assume a finite Markov-order $A_{\boldsymbol{Y}^i}(\kappa)$. Active information storage is thus a specific case of excess entropy (3.15), whereby the process is modelled as κ-order Markov. The implications of applying one measure over the other is discussed in Sec. 2.2.

Transfer entropy is designed to detect asymmetry in the interaction of subsystems by distinguishing between "driving" and "responding" elements [202]. Specifically, transfer entropy captures information transmission from a source (or multiple source) $\boldsymbol{Y}^j$ process(es) to a destination (or multiple destination) $\boldsymbol{Y}^i$ process(es) as the average information provided by the source variable(s) $\boldsymbol{Y}^-_n$ about the next destination variable Y^i_{n+1} in the context of the past state of the destination $\boldsymbol{Y}^{j,-}_n$ [151, 202]. Transfer entropy is computed as:

$$T_{\boldsymbol{Y}^j \to \boldsymbol{Y}^i} = \mathbf{E}\left[\log \frac{p(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n, \boldsymbol{Y}^{j,-}_n)}{p(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n)}\right] \tag{3.22}$$

$$= I(Y^i_{n+1}; \boldsymbol{Y}^{j,-}_n \mid \boldsymbol{Y}^{i,-}_n) \tag{3.23}$$

$$= H(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n) - H(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n, \boldsymbol{Y}^{j,-}_n). \tag{3.24}$$

Again, in practice one can consider finite-κ history $T_{\boldsymbol{Y}^j \to \boldsymbol{Y}^i}(\kappa)$.

It is important to realise that information transfer between two variables does not require an explicit communication channel, it rather indicates a high degree of directional synchrony or nonlinear correlation between the source and the destination. It characterises a degree of *predictive* information transfer, i.e., "if the state of the source is known, how much does that help to predict the state of the destination?" [151].

Sometimes it is useful to condition the local information transfer on another contributing process $\boldsymbol{Y}^k$. This results in the *conditional transfer entropy* [145, 151]:

$$T_{\boldsymbol{Y}^j \to \boldsymbol{Y}^i \mid \boldsymbol{Y}^k} = \mathbf{E}\left[\log \frac{p(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n, \boldsymbol{Y}^{j,-}_n, \boldsymbol{Y}^{k,-}_n)}{p(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n, \boldsymbol{Y}^{k,-}_n)}\right] \tag{3.25}$$

$$= I(Y^i_{n+1}; \boldsymbol{Y}^{j,-}_n \mid \boldsymbol{Y}^{i,-}_n, \boldsymbol{Y}^{k,-}_n) \tag{3.26}$$

$$= H(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n, \boldsymbol{Y}^{k,-}_n) - H(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n, \boldsymbol{Y}^{j,-}_n, \boldsymbol{Y}^{k,-}_n). \tag{3.27}$$

where $T_{\boldsymbol{Y}^j \to \boldsymbol{Y}^i \mid \boldsymbol{Y}^k}(\kappa)$ denotes the κ-order Markov assumption.

Finally, stochastic interaction measures the complexity of dynamical systems by quantifying the excess of information processed, in time, by the system beyond the information processed by each of the subsystems [13–15, 71]. The stochastic interaction of the collection of processes $\boldsymbol{Y}$ is computed as:

$$S_{\boldsymbol{Y}} = \mathbf{E}\left[\log \frac{\prod_i p(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n)}{p(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}^-_n)}\right] \tag{3.28}$$

$$= -H(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}^{i,-}_n) + \sum_{i=1}^{M} H(Y^i_{n+1} \mid \boldsymbol{Y}^{i,-}_n), \tag{3.29}$$

Or $S_{\boldsymbol{Y}}(\kappa)$ under the κ-order Markov assumption. Although, the original definition assumes a first-order Markov process [13, 15], we first introduce it above for the (semi-infinite) past.

The measures introduced in this section can be temporally localised in order to trace the information dynamics over time, e.g. for identifying peaks during specific moments (see [149]). However, in this thesis, we only use the average quantities.

3.5 DECISION RULES FOR MODEL SELECTION AND PLANNING

The problems of model selection and planning both fall under the general formulation of *decision theory*. The following description of decision rules is not derived from any one place but a combined view of statistical inference [128, 157] and planning under uncertainty [235].

Given an observable random variable (or stochastic process) $\boldsymbol{Z} \in \mathcal{Z}$, determined by a parameter $\theta \in \Theta$, and a set $\mathcal{U}$ of possible actions, a decision rule is a function $\pi : \mathcal{Z} \to \mathcal{U}$. For model selection, the set of actions are the parameter space, i.e., $\mathcal{U} = \Theta$. For developing a policy in robotic path planning, the actions are generally the next configuration $\mathcal{U} \subseteq \mathbb{R}^d$ in some d-dimensional configuration space. Decision rules π involve minimising some loss function $\mathcal{L}$ generally of the form:

$$\pi(\boldsymbol{Z}) = \arg\min_{\boldsymbol{u} \in \mathcal{U}} \max_{\theta \in \Theta} \mathcal{L}(\theta; \boldsymbol{u}), \tag{3.30}$$

$$\pi(\boldsymbol{Z}) = \arg\min_{\boldsymbol{u} \in \mathcal{U}} \mathbf{E}_{\Theta}[\mathcal{L}(\theta; \boldsymbol{U})], \tag{3.31}$$

or,

$$\pi(\boldsymbol{Z}) = \arg\min_{\boldsymbol{u} \in \mathcal{U}} \mathcal{L}(\theta; \boldsymbol{u}) \quad \text{subject to} \quad \theta \subset \Theta. \tag{3.32}$$

Here, (3.30) is the minimax loss function, (3.31) is the expected loss function; and (3.32) is the loss function within a subset of the parameter space. In this thesis we use the expected loss (3.31) in Chapters 4-6 and the bounded loss (3.32) in Chapter 7.

3.5.1 *Model selection*

In model selection we are interested in selecting a statistical model from a candidate set of models, given data. The general problem is that of selecting a parameter (or set of parameters) θ^* that minimises a loss function $\mathcal{L}(\theta; z)$, given data z:

$$\theta^* = \arg\min_{\theta} \mathcal{L}(\theta; z).$$

Most model selection approaches are based on either information theory or Bayesian statistics.[3] In the context of information theory, an established technique is to evaluate the encoding length of the data, given the model [3, 43, 195]. The simplest model should aim to minimise code length [135]. In this formulation, the problem is to minimise the KL divergence of the data from the model. In contrast, the Bayesian approach is to place a prior on the graph and compute the posterior probability, instead of looking at the information gain over a null hypothesis.

As a starting point, consider the simplest loss function: the likelihood of the model, given data $p_\theta(z) = p(z \mid \theta)$. Since it is often easier to work in log-space, we instead use the log-likelihood $L(\theta; z) = \log p_\theta(z)$ and negate it to make a loss function $\mathcal{L}(\theta; z) = -L(\theta; z)$. Given the data z are generated by random variables $\mathbf{Z}$, the log-likelihood is a random quantity and we are thus interested in the expected value. This equates the negative log-likelihood loss function with entropy:

$$\begin{aligned}\mathcal{L}(\theta; z) &= \mathbf{E}_{\mathbf{Z}}[-\ell(\theta; \mathbf{Z})] \\ &= H(\mathbf{Z}).\end{aligned}$$

Alternatively, the KL divergence loss function is defined as:

$$\begin{aligned}\mathcal{L}(\theta; z) &= D_{\mathrm{KL}}\left[p(\mathbf{Z}) \parallel p_\theta(\mathbf{Z})\right] \\ &= H(\mathbf{Z}) - \mathbf{E}_{\mathbf{Z}}\left[\log p_\theta(\mathbf{Z})\right].\end{aligned}$$

The minimum information loss is the uncompressed dataset, and thus minimising these functions will always lead to overfitting. Hence, it is common to regularise or penalise the loss function according to the number of parameters $d(\theta)$ and features of the dataset $c(z)$, e.g.,

$$\theta^* = \arg\min_{\theta} \left\{\mathcal{L}(\theta; z) - c(z)d(\theta)\right\}.$$

3 Technically, the dichotomy is actually between hypothetico-deductive and Bayesian approaches in statistical inference. The former refers to the process of formulating and testing a hypothesis and, as mentioned in Chapter 1, this process is formalised by information theory. However, it is recently being suggested that Bayesian statistics is often considered an advanced form of hypothetico-deductive reasoning [90] and so this abstraction is simply for convenience here.

The regulariser depends on the particular loss function. In Chapter 5, we use the Akaike information criterion (AIC) [4], the Bayesian information criterion (BIC) [204], χ^2-distributions and surrogate-distributions to penalise the loss functions.

Given this is a data-driven approach, we do not actually have access to the true parameter θ governing the distribution p_θ and instead use an (unbiased) maximum likelihood estimate $\hat{\theta}_n$ computed at time n. Moreover, we compute the average of the expected log-likelihoods $\hat{\mathbf{E}}[\ell(\hat{\theta}_n; \boldsymbol{Z})]$, rather than the expected values themselves. Under certain assumptions, these estimates asymptotically converge to the true values. This is commonly discussed in statistics literature and we provide a sketch below for how the estimates converge to the true values. The following discussion is adapted from Barnett [19], who specifically focused on transfer entropy.

In general, we assume uniqueness of the true parameter set θ and thus the maximum likelihood estimate $\hat{\theta}_n \xrightarrow{\text{a.s.}} \theta$ as $n \to \infty$. Moreover, processes studied in Chapters 4 and 5 are assumed ergodic with maps f that are (at least) endomorphisms. Due to these assumptions, the Birkhoff-Khinchin ergodic theorem [30] then applies so that the average of the expected log-likelihood $\hat{\mathbf{E}}[\ell(\theta; \boldsymbol{Z})] \xrightarrow{\text{a.s.}} \mathbf{E}[\ell(\theta; \boldsymbol{Z})]$. Combining these results, we have that computing the expected log-likelihoods via averaging and taking the maximum likelihood estimates of the parameters $\hat{\mathbf{E}}[\ell(\hat{\theta}_n; \boldsymbol{Z})] \xrightarrow{\text{a.s.}} \mathbf{E}[\ell(\theta; \boldsymbol{Z})]$.

3.5.2 *Planning under uncertainty*

In robotic path planning, the general problem is to select a sequence of measurement locations $\boldsymbol{x}$ that maximises the cumulative reward $\mathcal{R}(\boldsymbol{x}; \boldsymbol{z})$ based on the corresponding set of measurements $\boldsymbol{z}$ [235]:

$$\boldsymbol{X}^* = \arg\max_{\boldsymbol{X}} \mathbf{E}_{\boldsymbol{Z}} \left[\mathcal{R}(\boldsymbol{X}; \boldsymbol{Z}) \right].$$

In a vast number of scenarios, robots are tasked with understanding or modelling their environment. In general, some quantity of interest $\boldsymbol{Y}$ is typically inferred from the measurements $\boldsymbol{Z}$ at given locations $\boldsymbol{X}$. We want to find the sequence of actions $\boldsymbol{x}$ that tells us the most about the environment in question. That is, our reward is a function of the final entropy over our posterior:

$$\mathcal{R}(\boldsymbol{X}; \boldsymbol{Z}) = -H(\boldsymbol{Y}).$$

This cost function is used in Chapter 6, where the travel cost can be ignored and we can use a greedy algorithm to select future viewpoints. In typical planning scenarios, the robot has full access to the history of its actions and observations,

and plans over a limited time horizon. In this scenario, at every decision step, the task is to optimise the reward

$$\begin{aligned}\mathcal{R}(\boldsymbol{X}_{n+1}^{-};\boldsymbol{Z}_{n+1}^{-}) &= -\left(H(\boldsymbol{Y}_n) - H(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n)\right). \\ &= -I(\boldsymbol{Y}_{n+1};\boldsymbol{Y}_n). \end{aligned} \tag{3.33}$$

In other words, the objective of the robot is to choose an action that minimises the active information storage (3.19); this ensures new information reduces future uncertainty.

In general active perception tasks, however, it is critical to include an additional cost for the path $\boldsymbol{X}$ in the reward function, e.g.,

$$\boldsymbol{X}^* = \arg\max_{\boldsymbol{X}} \left\{H(\boldsymbol{Y}) - c(\boldsymbol{X})\right\}.$$

In Chapter 7, we explore this problem with a team of robots, where the path cost $c(\boldsymbol{X})$ must be below a certain bound.

3.6 DYNAMIC BAYESIAN NETWORKS

Throughout this thesis, we use the framework of DBNs to model the nonlinear processes in question. These models were introduced by Murphy [164] as more expressive than the established approaches of hidden Markov models or Kalman filters (linear dynamical systems). Moreover, they admit a suite of general purpose algorithms for inference, e.g., prediction, filtering, and fixed-lag smoothing.

The DBN is a general graphical representation of a temporal model, representing a probability distribution over infinite trajectories of random variables $(\boldsymbol{Z}_1, \boldsymbol{Z}_2, \ldots)$ compactly. As mentioned above, we denote $\boldsymbol{Z}_n = \{\boldsymbol{X}_n, \boldsymbol{Y}_n\}$ as the set of hidden and observed variables, respectively, where $n \in \{1, 2, \ldots\}$ is the temporal index. A BN $B = (G, \theta)$ represents a joint distribution $p_B(\boldsymbol{z})$ graphically and consists of: a DAG G and a set of CPD parameters θ corresponding to that DAG. Given a graph G, the P^i parents of variable $\boldsymbol{Z}_{n+1}^i$ are given by the parent set $\mathrm{Pa}_G(\boldsymbol{Z}_{n+1}^i) = \{\boldsymbol{Z}_n^{ij}\}_j = \{\boldsymbol{Z}_n^{i1}, \ldots, \boldsymbol{Z}_n^{iP^i}\}$.

The DBN model $B = (B_1, B_{\rightarrow})$ extends BNs to account for temporal processes and comprise two parts: the prior BN $B_1 = (G_1, \theta_1)$, which defines the joint distribution $p_{B_1}(\boldsymbol{z}_1)$; and the *two-time-slice BN (2TBN)* $B_{\rightarrow} = (G_{\rightarrow}, \theta_{\rightarrow})$, which defines a first-order Markov process $p_{B_{\rightarrow}}(\boldsymbol{z}_{n+1} \mid \boldsymbol{z}_n)$ [81]. This formulation allows for a variable to be conditioned on its respective parent set $\mathrm{Pa}_{G_{\rightarrow}}(Z_{n+1}^i)$ that can come from the preceding time slice or the current time slice, as long as $G_{\rightarrow}$ forms a DAG. The 2TBN probability distribution factorises according to $G_{\rightarrow}$ with a local CPD estimated from an observed dataset. That is, given a set of stochastic processes $(\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_N)$, the

realisation of which constitutes a dataset $z = (z_1, \ldots, z_N)$, we obtain the 2TBN distribution as

$$p_{B_\rightarrow}(z_{n+1} \mid z_n) = \prod_i p_{B_\rightarrow}(z^i_{n+1} \mid \mathrm{pa}_{G_\rightarrow}(\boldsymbol{Z}^i_{n+1})), \tag{3.34}$$

where $\mathrm{pa}_{G_\rightarrow}(\boldsymbol{Z}^i_{n+1})$ denotes the (index-ordered) set of realisations $\{z^j_o : \boldsymbol{Z}^j_o \in \mathrm{Pa}_{G_\rightarrow}(\boldsymbol{Z}^i_{n+1})\}$.

In this thesis, we are not concerned with learning the prior network (or, equivalently, assume it is uniform) and hence drop the temporal subscript, i.e., $B = B_\rightarrow$. If this was desired, one can learn the prior network B_1 independent of the 2TBN network B [81]. Moreover, we drop the dependence of the parent set on the graph structure if it is clear from context, i.e., $\mathrm{Pa} = \mathrm{Pa}_G$.

3.6.1 *Learning graph structure from data*

In this section we give background on the theory for learning DBNs from data. Specifically, we focus on data-driven methods for learning the edges between nodes in the graph, known as *structure learning*. Learning the parameters for a given structure are outside the scope of this work and often assumed to be learned through maximum likelihood as a subroutine of the model selection process. For more detail on the subject of parameter learning and PGMs in general, see Koller and Friedman [128].

We will focus on techniques for learning graphical models using the *score and search* paradigm [128]; there are other less common approaches that we will not cover here [66], e.g., constraint-based methods. Given a dataset $z = (z_1, \ldots, z_N)$, the objective is to find a DAG G^* such that

$$G^* = \arg\max_{G \in \mathcal{G}} g(B : \boldsymbol{Z}), \tag{3.35}$$

where $g(B : \boldsymbol{Z})$ is a scoring function measuring the degree of fitness of a candidate DAG G to the data set $\boldsymbol{Z}$, and $\mathcal{G}$ is the set of all DAGs. Note that we could turn (3.35) into minimising a loss function by negation, i.e., $\mathcal{L}(\hat{\theta}_G; \boldsymbol{Z}) = -g(B : \boldsymbol{Z})$, however we opt for a scoring function as this is common in BN structure learning literature.

As mentioned in Sec. 2.3.2, finding the optimal graph G^* in (3.35) requires solutions to the two subproblems that comprise structure learning: the evaluation problem and the identification problem [50]. The main problem we focus on in Chapter 5 is the evaluation problem, i.e., determining a score that quantifies the quality of a graph, given data.

An intuitive scoring function involves computing the likelihood of the graph given data $L(\hat{\theta}_G; z) = p_{\hat{B}}(z_{n+1} \mid z_n)$. Let $\hat{\theta}_G$ be the set of parameters that maximise

$p_{\hat{B}}(\mathbf{Z} = z) = p(z \mid \hat{\theta}_G)$ for a given graph G, i.e., $\ell(\hat{\theta}_G; z) = \log L(\hat{\theta}_G; \mathbf{Z} = z)$. Using the log-likelihood gives the most naive scoring function, the *maximum likelihood score*:

$$g_{\text{L}}(G : \mathbf{Z}) = \mathbf{E}_{\mathbf{Z}} \left[\ell(\hat{\theta}_G; \mathbf{Z}) \right]. \tag{3.36}$$

3.6.1.1 *Information-theoretic approach*

The KL divergence can be used to quantify the information loss of a *factorised distribution* $p_{\hat{B}}$ from the complete distribution p_{K_M}. The factorised distribution is given by including conditional independences exhibited in $\hat{B}$, whereas the complete distribution exhibits no conditional independences, i.e., complete graph K_M. Mathematically, this is expressed as:

$$\begin{aligned} D_{\text{KL}} \left[p_{\hat{K}} \parallel p_{\hat{B}} \right] &= D_{\text{KL}} \left[p_{\hat{K}}(\mathbf{Z}_{n+1} \mid \mathbf{Z}_n^-) \parallel p_{\hat{B}}(\mathbf{Z}_{n+1} \mid \mathbf{Z}_n^-) \right] \\ &= \sum_{z_n^-} p_{\hat{K}}(z_n^-) \sum_{z_{n+1}} p_{\hat{K}}(z_{n+1} \mid z_n^-) \log \frac{p_{\hat{K}}(z_{n+1} \mid z_n^-)}{p_{\hat{B}}(z_{n+1} \mid z_n^-)} \\ &= \mathbf{E}_{\mathbf{Z}} \left[\log \frac{p_{\hat{K}}(\mathbf{Z}_{n+1} \mid \mathbf{Z}_n^-)}{p_{\hat{B}}(\mathbf{Z}_{n+1} \mid \mathbf{Z}_n)} \right]. \end{aligned} \tag{3.37}$$

Here, we have dropped the subscript M from K_M, since the context is clear, i.e., $p_K = p_{K_M}$. It is common in model selection to decompose the KL divergence as

$$D_{\text{KL}} \left[p_{\hat{K}} \parallel p_{\hat{B}} \right] = \mathbf{E} \left[\log p_{\hat{K}}(\mathbf{Z}_{n+1} \mid \mathbf{Z}_n^-) \right] - \mathbf{E} \Big[\ell(\hat{\theta}_G; \mathbf{Z}) \Big]. \tag{3.38}$$

In this form, $p_{\hat{K}}$ is often identical for all models considered and, in practice, it suffices to ignore this term and thus avoid the problem of computing distributions of latent variables. The resulting simpler expression can be viewed as equivalent to the log-likelihood maximisation in (3.36). However, $p_{\hat{K}}$ is *not* equivalent when considering different families of models (as we elaborate on later in Chapter 5).

Again, the KL divergence alone will not yield a parsimonious model, and so we must penalise it. This results in a scoring function in the following form:

$$g_{\text{KL}}(B : \mathbf{Z}) = D_{\text{KL}} \left[p_{\hat{K}} \parallel p_{\hat{B}} \right] - c(N) d(G). \tag{3.39}$$

One way of approximating and regularising this divergence was derived by Akaike [4], who presented what is now known as the AIC:

$$\lim_{N \to \infty} D_{\text{KL}} \left[p_{\hat{K}} \parallel p_{\hat{B}} \right] \simeq \mathbf{E}[\ell(\hat{\theta}_G; \mathbf{Z})] - d(G), \tag{3.40}$$

where $d(G)$ is the model dimension (i.e., number of parameters needed for the graph G [81]).

3.6.1.2 *Bayesian approach*

The Bayesian approach to structure learning is to compute the posterior probability of the network structure G, given data $\boldsymbol{z}$. Using Bayes' rule, we can express this distribution as $p(G \mid \boldsymbol{z}) \propto p(\boldsymbol{z} \mid G)p(G)$, where $p(G)$ encodes any prior assumptions made about the network G. Thus, the problem becomes that of computing the likelihood of the data, given the model, $p(\boldsymbol{z} \mid G)$. This likelihood can be written in terms of distributions over network parameters [81]:

$$p(\boldsymbol{z} \mid G) = \int_{\theta} L(\hat{\theta}_G; \boldsymbol{z})\ p(\boldsymbol{z} \mid \hat{\theta}_G)\ p(\theta \mid G)\ d\theta. \tag{3.41}$$

A common approach to compute (3.41) in closed form is by using Dirichlet priors. This leads to the BD (Bayesian-Dirichlet) score and variants [81, 100]. However, to obtain this analytic solution, we require discrete variables and counts of the tuples $(z_n^i, \mathrm{pa}(z_n^i))$, which can involve latent states. Schwarz [204] derived an asymptotic approximation of the posterior distribution, known as the BIC, which states that

$$\lim_{N\to\infty} p(\boldsymbol{z} \mid G) \xrightarrow{\text{a.s.}} \mathbf{E}[\ell(\hat{\theta}_G; \boldsymbol{Z})] - \frac{\log N}{2} d(G) + \mathcal{O}(1), \tag{3.42}$$

where $\mathcal{O}(1)$ is a constant bounded by the number of potential models. The approximation of the posterior (3.42) requires that data come from an exponential family of likelihood functions with conjugate priors over the model G, and the parameters given the model $\hat{\theta}_G$ [204].

3.6.1.3 *A general information criterion*

We can compute AIC or BIC in terms of the expected log-likelihood $\mathbf{E}[\ell(\hat{\theta}_G; \boldsymbol{Z})]$ and the model dimension $d(G)$, and thus the problem can be generalised to that of deriving an information criterion for scoring the graph of the form [43, 66]

$$g_{\text{IC}}(B : \boldsymbol{Z}) = \mathbf{E}\left[\ell(\hat{\theta}_G; \boldsymbol{Z})\right] - c(N)d(G). \tag{3.43}$$

When $c(N) = 1$, we have the AIC score [4]; $c(N) = \log(N)/2$ yields the BIC score [204][4], and $c(N) = 0$ gives the maximum likelihood score.

3.6.1.4 *The log-likelihood ratio*

The (expected) log-likelihood ratio is the difference between a model $\hat{B}$ and the null model $\hat{B}_0$, i.e.,

$$\mathbf{E}\left[\ell(\hat{B}; \boldsymbol{Y}) - \ell(\hat{B}_0; \boldsymbol{Y})\right] = \mathbf{E}\left[\log \frac{p_{\hat{B}}(\boldsymbol{Y})}{p_{\hat{B}_0}(\boldsymbol{Y})}\right].$$

4 This definition is equivalent to the minimum description length (MDL) scoring function presented in [231], whereas Eq. (3.43) is the general form of the MDL according to [43].

For a nested model from the exponential family, the likelihood ratio is asymptotically χ^2-distributed. A nested model refers to cases where the more complex model $\hat{B}$ can be transformed into the simpler model $\hat{B}_0$ by imposing a set of constraints on the parameters. As a result, in structure learning, the null model $\hat{B}_0$ is generally considered to be the independent network, i.e., where $\mathrm{pa}(X_n^i) = \emptyset$ for any i and n.

Note that the expected log-likelihood ratio is equivalent to the KL divergence. However, typically the KL divergence will be taken from the complete model K_M, whereas the log-likelihood ratio quantifies the improvement over the empty model $\hat{B}_0$ due to the ratio test mentioned above. Thus, although they admit equivalent formulas, we will distinguish them for this reason.

3.7 SUMMARY

This chapter presented the nomenclature and technical detail required for the remainder of the thesis. In particular, we formally introduced stochastic processes, time series modelling, and informative path planning. We provided a decision-theoretic problem statement that succinctly captures the objective underlying the following chapters of this thesis. This general formulation further coalesces the concepts of time series modelling and robotic information gathering. In the next chapter, we examine multi-agent dynamics *post hoc*, where the aim is to investigate the memory and communication of the system by observation alone, i.e., without knowing the internal logic of the agents.

4 INFORMATION MEASURES FOR MULTI-AGENT DYNAMICS

In this chapter we model multi-agent dynamics as coupled finite-order Markov chains. Given that we do not explicitly model the underlying dynamics, the approaches used here take the more traditional statistical perspective on time series analysis. The methods are therefore generally applicable to any fully observed system that is assumed to follow this model.

4.1 OVERVIEW

This chapter introduces techniques for quantifying long-range interactions and communication in multi-agent dynamics. We first consider the information dynamics measures we use in this chapter under the model selection paradigm. Then, using these measures, we investigate the transfer and storage of information in multi-agent dynamics.

The first problem we address is to identify interaction networks that link together autonomous agents. This is achieved without reconstructing the agents' logic and neural processing and using only the observational data, such as positional (e.g. planar) coordinates and their changes. The problem is difficult as some of the dependencies between agents are not discernible simply by correlating their corresponding locations over time — one needs to take into account the possibly directed nature of such correlations, where dynamics of one of the agents affects the positioning of another.

The second problem we address is classifying coherent dynamic situations within the multi-agent games, in the context of distributed communications. For example, during a game, each player (dependent on their tactical role) is engaged in dynamics which are affected both by (i) the player's history of actions (persistence or rigidity), and (ii) spatially long-ranged effects of other players' actions (sensitivity or responsiveness). Therefore, we may want to form an abstract state-space with variables quantifying these features, and consider a structure of this space, aiming to classify the games and game situations by identifying coherent regions within the space.

As a canonical example of team-based dynamics, we use game instances produced within the RoboCup simulation environment. That is, we aim to identify implicit interaction networks and adopt the methods of coherent information structure in classifying repeatable collective dynamics in game situations.

4.2 PROBLEM STATEMENT

In this chapter, for each game g, we have a dataset $\boldsymbol{y}$ as the realisation of a multivariate process $\boldsymbol{Y}$. Each component $\boldsymbol{y}^i$ of this process describes the dynamics of a player i in the two-dimensional RoboCup simulation, however could be abstracted from any multivariate sequence. Recall from Sec. 3.4.2 that we assume these processes are autonomous (such that the CPDs are homogeneous) and ergodic (such that estimators approach the true parameter values).

A game g contains N time steps and is played between two teams $\mathcal{P} = \{P_1, P_2, \ldots, P_M\}$ and $\mathcal{Q} = \{Q_1, Q_2, \ldots, Q_M\}$ with M agents each. The dynamics of the game is captured by the realisation of two sets of stochastic processes $\boldsymbol{Y}^{\mathcal{P}} = \{\boldsymbol{Y}^{P_1}, \ldots, \boldsymbol{Y}^{P_M}\}$ and $\boldsymbol{Y}^{\mathcal{Q}} = \{\boldsymbol{Y}^{Q_1}, \ldots, \boldsymbol{Y}^{Q_M}\}$, i.e., the movements of players in teams $\mathcal{P}$ and $\mathcal{Q}$, respectively. The measurements of each temporal process $\boldsymbol{Y}^i$ is therefore a sequence of positional data $(\boldsymbol{y}^i_1, \ldots, \boldsymbol{y}^i_N)$; in this chapter we consider observations $\boldsymbol{y}^i_n$ as the *change* in the 2D positional vector of the agent.[1] In the remainder of the chapter, we often use the terms process, agent and player interchangeably depending on context.

In this case, the player movements can be represented as a DBN $B = (G, \theta_G)$ given by graph G and a set of parameters θ_G that define the conditional probability distribution $p_{\hat{B}}$. As such, each player is a node in the network. We aim to study this network in terms of the storage and transfer of information and relate these concepts to the collective objective of goal scoring.

4.3 INFORMATION DYNAMICS FOR MODEL SELECTION

First, we more rigorously ground (multivariate) transfer entropy and active information storage (introduced in Sec. 3.4.2) in the context of DBN model selection (introduced in Sec. 3.6.1). The following results are extensions of the derivation of transfer entropy as a log-likelihood ratio considered in [19].

1 This ensures the process is stationary by removing low-frequency components.

4.3.1 *Collective transfer entropy as a log-likelihood ratio*

The transfer of information between subsystems is obtained through coupling in a directed network. First, assuming the data are generated by an adapted process with independent noise, we get the following decomposition:

$$p_{\hat{B}}(\boldsymbol{y}) = \prod_n p_{\hat{B}}(\boldsymbol{y}_{n+1} \mid \boldsymbol{y}_n^-), \tag{4.1}$$

If we further model the data as a collection of coupled processes, then

$$p_{\hat{B}}(\boldsymbol{y}_{n+1} \mid \boldsymbol{y}_n^-) = \prod_i p_{\hat{B}}(\boldsymbol{y}_{n+1} \mid \boldsymbol{y}_n^{i,-}, \mathrm{pa}(\boldsymbol{Y}_{n+1}^i)), \tag{4.2}$$

As mentioned in Chapter 3, we are actually interested in the expected log-likelihood since the data $\boldsymbol{y}$ are realisation of a process $\boldsymbol{Y}$:

$$\begin{aligned}
\mathbf{E}[\ell(\hat{B}; \boldsymbol{Y})] &= \mathbf{E}\left[\log \prod_n p_{\hat{B}}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^{(n)})\right] \\
&= \sum_n \mathbf{E}\left[\log p_{\hat{B}}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^-)\right] \\
&= \sum_i \sum_n \mathbf{E}\left[\log p_{\hat{B}}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^{i,-}, \mathrm{Pa}(\boldsymbol{Y}_{n+1}^i))\right] .
\end{aligned} \tag{4.3}$$

Then the log-likelihood ratio can be decomposed as:

$$\begin{aligned}
\mathbf{E}[\ell(\hat{B}; \boldsymbol{Y}) - \ell(\hat{B}_0; \boldsymbol{Y})] &= \mathbf{E}\left[\log \frac{p_{\hat{B}}(\boldsymbol{Y})}{p_{\hat{B}_0}(\boldsymbol{Y})}\right] \\
&= \sum_i \sum_n \mathbf{E}\left[\log \frac{p_{\hat{B}}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^{i,-}, \mathrm{Pa}(\boldsymbol{Y}_n^i))}{p_{\hat{B}}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^{i,-})}\right],
\end{aligned} \tag{4.4}$$

where, as in Sec. 3.6.1, $\hat{B}_0$ is the null hypothesis of no transfer. Note that we can remove the sum over all n expected values, since the average expectation $\hat{\mathbf{E}}$ will converge to the true expectation $\mathbf{E}$ (see discussion in Chapter 3). Finally, due to stationarity, we can say that the parent set for each node at each time $\mathrm{Pa}(\boldsymbol{Y}_n^i)$ is constant and (4.4) becomes a sum over transfer entropies:

$$\mathbf{E}[\ell(\hat{B}; \boldsymbol{Y}) - \ell(\hat{B}_0\ ;\ \boldsymbol{Y})] = \sum_i T_{\mathrm{Pa}(\boldsymbol{Y}^i) \to \boldsymbol{Y}^i} . \tag{4.5}$$

When the graph contains cycles (e.g., $\mathrm{Pa}(Y^i) = Y^j$ and $\mathrm{Pa}(Y^j) = Y^i$), estimating the marginals (4.2) will be biased (since information in agent the past of agent i already contains information in the past of agent j and vice versa). As such, in the case of a soccer match, with constant feedback between players, we consider transfer entropy to reveal first-order approximations to the effective network. This is discussed further in Sec. 4.5.3.

4.3.2 *Active information storage as a log-likelihood ratio*

We can consider the process of storing information as the degree to which individual processes are simply a sequence of independent random variables (i.e., uncorrelated in time). We thus get the same decomposition as (4.1) for an adapted process. Now, we are testing to what degree this process diverges from an independent sequence of variables, i.e., $p_{\hat{B}_0}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^-) = p_{\hat{B}_0}(\boldsymbol{Y}_{n+1})$. Hence, the log-likelihood ratio decomposes as:

$$\mathbf{E}[\ell(\hat{B};\boldsymbol{Y}) - \ell(\hat{B}_0;\boldsymbol{Y})] = \sum_n \mathbf{E}\left[\log \frac{p_{\hat{B}}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{Y}_n^-)}{p_{\hat{B}}(\boldsymbol{Y}_{n+1})}\right] . \tag{4.6}$$

By the same logic as above, under the assumptions of ergodicity and stationarity, Eq. (4.6) becomes a sum over the active information storage of each subsystem:

$$\mathbf{E}[\ell(\hat{B};\boldsymbol{Y}) - \ell(\hat{B}_0;\boldsymbol{Y})] = A_{\boldsymbol{Y}} . \tag{4.7}$$

Both of the above results trivially extend to finite-order Markov chains as discussed in Sec. 3.4.2.

4.4 TACTICAL INFORMATION DYNAMICS

In order to estimate the strength of directed coupling between two agents we compute the average transfer entropy between them during any given game. In this section we will describe the information-theoretic measures above in the context of robotic soccer as classifying the responsiveness and rigidity of a team.

We will be using the notion of a tactical formation which describes how the players in a team are generally positioned on the field in terms of their roles (the number of: defenders-midfielders-attackers), e.g., "4-3-3" formation with four defenders, three midfielders, and three attackers. Of course, during a game, any player may be drawn to a position fairly remote from its area of responsibility defined by the role (for instance, a defender may join a particular attack), but in general the players tend to stay within their distinct areas, specified by some prior configurations or distinguishable by spatial pattern matching, see Fig. 4.1. Hence, any dynamic coherence observed in the motion of players which are spatially separated on the field due to their tactical roles (e.g., a midfielder of one team and an opponent's defender) can be interpreted as spatially long-range implicit interactions.

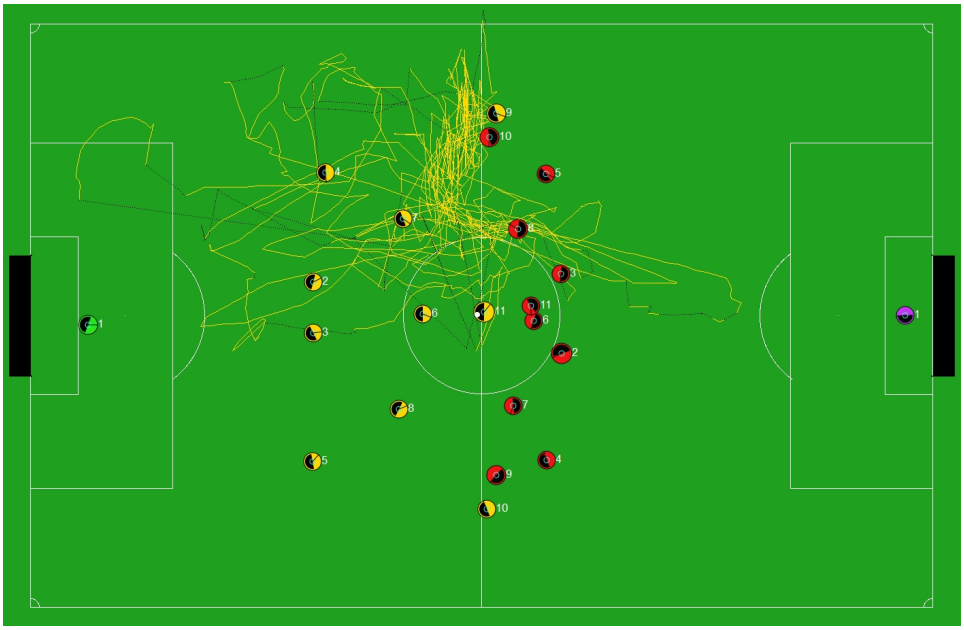


Figure 4.1: Motion trace diagram. A trace curve represents the motion of the left midfielder (player 7) of the left team, during an entire game (solid yellow for regular "play-on" time points, and dotted black for not "play-on" times, e.g. free kicks). The role of player 7 is distinguishable as the left midfielder.

4.4.1 *Transfer entropy as player responsiveness*

For each game g, the transfer entropy is calculated between each source agent $\boldsymbol{Y}^i$ and destination agent $\boldsymbol{Y}^j$, in the context of some other dynamics $\boldsymbol{B}$, denoted $T^g_{\boldsymbol{Y}^i \to \boldsymbol{Y}^j | \boldsymbol{B}}$. In the remainder of this chapter, the relative position of the ball is always conditioned upon in order to compute the transfer entropy in the context of the game, since this context is greatly affected by the ball trajectories in soccer matches. We also define the average transfer entropy over a range of source-destination pairs, targeting subsets $\boldsymbol{Y}^{\alpha(\mathcal{Q})} \subseteq \boldsymbol{Y}^{\mathcal{Q}}$ and $\boldsymbol{Y}^{\beta(\mathcal{P})} \subseteq \boldsymbol{Y}^{\mathcal{P}}$:[2]

$$T^g_{\boldsymbol{Y}^{\alpha(\mathcal{Q})} \to \boldsymbol{Y}^{\beta(\mathcal{P})} | \boldsymbol{B}} = \frac{1}{|\boldsymbol{Y}^{\alpha(\mathcal{Q})}| |\boldsymbol{Y}^{\beta(\mathcal{P})}|} \sum_{\boldsymbol{Y}^i \in \boldsymbol{Y}^{\alpha(\mathcal{Q})}} \sum_{\boldsymbol{Y}^j \in \boldsymbol{Y}^{\beta(\mathcal{P})}} T^g_{\boldsymbol{Y}^i \to \boldsymbol{Y}^j | \boldsymbol{B}} \,. \tag{4.8}$$

The average transfer entropy, defined for specific subsets of team processes, is useful in considering distributed communications across agents with specific roles (e.g. attackers and defenders in soccer).

2 We note a subtle distinction here: $T^g_{\boldsymbol{Y}^{\alpha(\mathcal{Q})} \to \boldsymbol{Y}^{\beta(\mathcal{P})} | \boldsymbol{B}}$ is not equal to the multivariate transfer entropy [150] from the set $\boldsymbol{Y}^{\alpha(\mathcal{Q})}$ to $\boldsymbol{Y}^{\beta(\mathcal{P})}$ (conditioned on $\boldsymbol{B}$) as a whole in general, because of dependencies within and across the sets. That is, $T^g_{\boldsymbol{Y}^{\alpha(\mathcal{Q})} \to \boldsymbol{Y}^{\beta(\mathcal{P})} | \boldsymbol{B}}$ could be viewed as an approximation to multivariate transfer entropy (ignoring these dependencies) in order to avoid dimensionality issues.

Building upon the information dynamics measures, it is possible to investigate role-based behavior with complex interactions. In applying information dynamics to the RoboCup 2D Simulation League we use the following definition:

Definition 4.1. *Responsiveness of player i to player j during the game g is defined as the information transfer $T^g_{\boldsymbol{Y}^j \to \boldsymbol{Y}^i | \boldsymbol{B}}$ from the source j (e.g. dynamics of player $\boldsymbol{Y}^j$) to the destination i (e.g., dynamics of another player $\boldsymbol{Y}^i$), in the context of some other dynamics $\boldsymbol{B}$ (e.g., the movement of the ball).*

That is, the "destination" player $\boldsymbol{Y}^i$ responds, for example, by repositioning, to the movement of the "source" player $\boldsymbol{Y}^j$. This may apply to many situations on the field. For instance, when one team's forwards are trying to better avoid their opponent's defenders, we consider the information transfer $T^g_{\boldsymbol{Y}^{d(\mathcal{Q})} \to \boldsymbol{Y}^{a(\mathcal{P})} | \boldsymbol{B}}$ from defender-agent processes $\boldsymbol{Y}^i \in \boldsymbol{Y}^{d(\mathcal{Q})}$ to forward-agent processes $\boldsymbol{Y}^j \in \boldsymbol{Y}^{a(\mathcal{P})}$, where roles of the agents are determined by their placements in a given tactical formation. Henceforth, we omit the game index g and the condition variable $\boldsymbol{B}$ when there is no ambiguity. Vice versa, the dynamics of the opponent's defenders, who are trying to better mark our team's forwards, are represented in the information transfer $T_{\boldsymbol{Y}^{a(\mathcal{P})} \to \boldsymbol{Y}^{d(\mathcal{Q})}}$ from forward-agent processes $\boldsymbol{Y}^j \in \boldsymbol{Y}^{a(\mathcal{P})}$ to defender-agent processes $\boldsymbol{Y}^i \in \boldsymbol{Y}^{d(\mathcal{Q})}$. These two examples specifically consider a coupling between the attack line $\boldsymbol{Y}^{a(\mathcal{P})}$ of our team and the defense line $\boldsymbol{Y}^{d(\mathcal{Q})}$ of opponent's team.

4.4.2 *Active information storage as player rigidity*

Our analysis also involves computation of the active information storage within the teams. We can define the average active information storage over a range of agents in a game g, targeting subsets $\boldsymbol{Y}^{\beta(\mathcal{P})} \subseteq \boldsymbol{Y}^{\mathcal{P}}$:[3]

$$A^g_{\boldsymbol{Y}^{\beta(\mathcal{P})}} = \frac{1}{|\boldsymbol{Y}^{\beta(\mathcal{P})}|} \sum_{\boldsymbol{Y}^j \in \boldsymbol{Y}^{\beta(\mathcal{P})}} A^g_{\boldsymbol{Y}^j} \,. \tag{4.9}$$

We characterise a team's rigidity $A_{\boldsymbol{Y}^{\mathcal{P}}}$ as the average of information storage values for all players of the team, according to the following definition.

Definition 4.2. *The rigidity of player i is defined as the information storage $A_{\boldsymbol{Y}^i}$ within the process $\boldsymbol{Y}^i$.*

The average information storage, or rigidity, within a team $A_{\boldsymbol{Y}^{\mathcal{P}}}$ is high whenever one can predict the motion of some players based on the movements of their past.

3 As per footnote 2, $A^g_{\boldsymbol{Y}^{\beta(\mathcal{P})}}$ is not equal to the collective active information storage as defined for the multivariate set $\boldsymbol{Y}^{\beta(\mathcal{P})}$ in general, due to dependencies between the variables. Again, $A^g_{\boldsymbol{Y}^{\beta(\mathcal{P})}}$ could be seen as an approximation to such a collective quantity (ignoring these dependencies) which avoids dimensionality issues.

Table 4.1: Tactical information dynamics measures.

Primitive	Metric	Equation	Description
Transmission	$\delta T(\mathcal{P},\mathcal{Q})$	$T_{\boldsymbol{Y}^{\mathcal{Q}}\to\boldsymbol{Y}^{\mathcal{P}}} - T_{\boldsymbol{Y}^{\mathcal{P}}\to\boldsymbol{Y}^{\mathcal{Q}}}$	Rel. responsiveness (team → team)*
	$\delta T_{a\to d}(\mathcal{P},\mathcal{Q})$	$T_{\boldsymbol{Y}^{a(\mathcal{Q})}\to\boldsymbol{Y}^{d(\mathcal{P})}} - T_{\boldsymbol{Y}^{a(\mathcal{P})}\to\boldsymbol{Y}^{d(\mathcal{Q})}}$	Rel. responsiveness (defenders → attackers)
	$\delta T_{m\to m}(\mathcal{P},\mathcal{Q})$	$T_{\boldsymbol{Y}^{m(\mathcal{Q})}\to\boldsymbol{Y}^{m(\mathcal{P})}} - T_{\boldsymbol{Y}^{m(\mathcal{P})}\to\boldsymbol{Y}^{m(\mathcal{Q})}}$	Rel. responsiveness (midfielders → midfielders)
	$\delta T_{d\to a}(\mathcal{P},\mathcal{Q})$	$T_{\boldsymbol{Y}^{d(\mathcal{Q})}\to\boldsymbol{Y}^{a(\mathcal{P})}} - T_{\boldsymbol{Y}^{d(\mathcal{P})}\to\boldsymbol{Y}^{a(\mathcal{Q})}}$	Rel. responsiveness (attackers → defenders)
Storage	$\delta A(\mathcal{P},\mathcal{Q})$	$A_{\boldsymbol{Y}^{\mathcal{P}}} - A_{\boldsymbol{Y}^{\mathcal{Q}}}$	Rel. team rigidity
	$\delta A_d(\mathcal{P},\mathcal{Q})$	$A_{\boldsymbol{Y}^{d(\mathcal{P})}} - A_{\boldsymbol{Y}^{d(\mathcal{Q})}}$	Rel. defender rigidity
	$\delta A_m(\mathcal{P},\mathcal{Q})$	$A_{\boldsymbol{Y}^{m(\mathcal{P})}} - A_{\boldsymbol{Y}^{m(\mathcal{Q})}}$	Rel. midfielder rigidity
	$\delta A_a(\mathcal{P},\mathcal{Q})$	$A_{\boldsymbol{Y}^{a(\mathcal{P})}} - A_{\boldsymbol{Y}^{a(\mathcal{Q})}}$	Rel. attacker rigidity

All measures are computed for one team relative (rel.) to the other by deduction.

In these cases, the players are not as independent of their previous movements as a complex or swarm behavior may warrant, making the dynamics less versatile.

How much does a team's rigidity and responsiveness contribute to a game's scoreline? To answer this question, one can analyse the correlation between a number of measures and the scoreline $\delta S^g = S^g_{\mathcal{P}} - S^g_{\mathcal{Q}}$, where $S^g_{\mathcal{P}}$ is the number of goals scored by team $\mathcal{P}$.

The utilised measures are relative, e.g., the relative team responsiveness

$$\delta T^g = T^g_{\boldsymbol{Y}^{\mathcal{Q}}\to\boldsymbol{Y}^{\mathcal{P}}|\boldsymbol{B}} - T^g_{\boldsymbol{Y}^{\mathcal{P}}\to\boldsymbol{Y}^{\mathcal{Q}}|\boldsymbol{B}}$$

is calculated by comparing the transfer from team $\boldsymbol{Y}^{\mathcal{Q}}$ to team $\boldsymbol{Y}^{\mathcal{P}}$ against the transfer in the other direction. Table 4.1 summarises different relative measures, specified for different tactical roles in a typical soccer formation. We would like to point out that we introduce here new relative measures, expanding on the ones analysed in [55]. Specifically, the previous study [55] compared attacking vs defending lines, that is, analysed $T_{\boldsymbol{Y}^{a(\mathcal{Q})}\to\boldsymbol{Y}^{d(\mathcal{P})}} - T_{\boldsymbol{Y}^{d(\mathcal{P})}\to\boldsymbol{Y}^{a(\mathcal{Q})}}$, while in this work we compare attacking vs attacking lines on the one hand $T_{\boldsymbol{Y}^{d(\mathcal{Q})}\to\boldsymbol{Y}^{a(\mathcal{P})}} - T_{\boldsymbol{Y}^{d(\mathcal{P})}\to\boldsymbol{Y}^{a(\mathcal{Q})}}$ and defending vs defending lines $T_{\boldsymbol{Y}^{a(\mathcal{Q})}\to\boldsymbol{Y}^{d(\mathcal{P})}} - T_{\boldsymbol{Y}^{a(\mathcal{P})}\to\boldsymbol{Y}^{d(\mathcal{Q})}}$ on the other hand. This change is addressing a different question of evaluating relative performance of a specific tactical line (role). Note also that we use averaged pairwise calculations in (4.8) and (4.9), as opposed to a multivariate approach (as in [146, 150]). These two approaches are only equivalent if the individual player processes are independent.

4.5 INTERACTION DIAGRAMS

We describe here another information dynamics tool, interaction diagrams, which provide a simplified view of the strongest pairwise interactions into (Sec. 4.5.1) or out of (Sec. 4.5.2) each agent. Note that the following diagrams are computed for

both teams $\mathcal{P}$ and $\mathcal{Q}$, however, we only describe the algorithms from the perspective of team $\mathcal{P}$.

4.5.1 *Information-sink diagrams*

Once the game's average transfer entropy $T^{g}_{\boldsymbol{Y}^i \to \boldsymbol{Y}^j|\boldsymbol{B}}$ is determined for each pair of agents $\{\boldsymbol{Y}^i, \boldsymbol{Y}^j\}_{i\in\mathcal{P}, j\in\mathcal{Q}}$, we identify the *source* opposing agent $\hat{\imath} \in \mathcal{Q}$ (described by the process $\boldsymbol{Y}^{\hat{\imath}} \in \boldsymbol{Y}^{\mathcal{Q}}$) that transfers maximal information to the given agent $j \in \mathcal{P}$ (i.e., process $\boldsymbol{Y}^j$), i.e.,

$$\hat{\imath}(j, g) = \arg\max_{i\in\mathcal{Q}} \left\{ T^{g}_{\boldsymbol{Y}^i \to \boldsymbol{Y}^j|\boldsymbol{B}} \right\}. \tag{4.10}$$

Over a number of games G, we select the source agent $\hat{\imath}(j)$ that transfers maximal information to $\boldsymbol{Y}^j$ most frequently, as the mode of the set $\{\hat{\imath}(j, 1), \ldots, \hat{\imath}(j, G)\}$. Then, we consider the average information transfer between the processes $\boldsymbol{Y}^{\hat{\imath}}$ and $\boldsymbol{Y}^j$ across all games:

$$T_{\boldsymbol{Y}^{\hat{\imath}(j)} \to \boldsymbol{Y}^j|\boldsymbol{B}} = \frac{1}{G} \sum_{g=1}^{G} T^{g}_{\boldsymbol{Y}^{\hat{\imath}(j,g)} \to \boldsymbol{Y}^j|\boldsymbol{B}}. \tag{4.11}$$

Intuitively, the movement of the source agent $\hat{\imath}(j)$ affected the agent j more than movement of any other agent in team $\mathcal{Q}$. That is, the agent j was responsive most to movement of the source agent $\hat{\imath}(j)$. Crucially, when we use the notion of *responsiveness* to another (source) agent, we do not load it with such semantics as being dominated by, or driven by that other agent. Higher responsiveness may in fact reflect either useful reaction to the opponent's movements (e.g., good marking of the source), or a helpless behaviour (e.g., constant chase after the source). Vice versa, generating a high responsiveness from another agent may result in either a useful dynamic (e.g., positional or even tactical dominance over the responding agent), or a wasteful motion (e.g., being successfully marked by the responding agent). In short, responsiveness captured in the maximal transfer $T_{Y^{\hat{\imath}} \to \boldsymbol{Y}^j|\boldsymbol{B}}$ detects a directed coupling from the source process $Y^{\hat{\imath}}$ to the responding process $\boldsymbol{Y}^j$ and at face value alone should not be interpreted in general as a simple index for comparative performance. It is, however, a useful identifier of the opponents' source player that was affecting a given agent j most.

Given a series of games, we identify the pairs "source-responder" by finding the source agent for each of the agents on both teams (always choosing the source among the opponents). The pairs $(\hat{\imath}(j), j)$ identified for each agent j in team $\mathcal{P}$ treated as a destination are combined in an "information-sink diagram" $\hat{G}_{\mathcal{P},\mathcal{Q}} = (\mathcal{P}, \mathcal{Q}, \hat{\mathcal{E}}_{\mathcal{P}}, \hat{\mathcal{E}}_{\mathcal{Q}})$, where the edge set $\hat{\mathcal{E}}_{\mathcal{P}} = \{\hat{\imath}(j) \to j : j \in \mathcal{P}\}$. The information-sink

interaction diagram $\hat{G}_{\mathcal{P},\mathcal{Q}}$ visualises a directed graph with $2M$ nodes representing players, with the edges representing all source-responder pairs, where a single edge is incoming to every agent from the corresponding source. One may extend the diagrams by specifying the weight of each edge with the corresponding transfer entropy.

4.5.2 *Information-source diagrams*

Similarly, having obtained the average transfer entropy during a game, for all pairs, we identify the *responder* agent $\check{\jmath} \in \mathcal{Q}$ described by the process $\boldsymbol{Y}^{\check{\jmath}} \in \boldsymbol{Y}^{\mathcal{Q}}$ that "received" maximal information from process $\boldsymbol{Y}^i$ for the given agent $i \in \mathcal{P}$. Formally, for any game g:

$$\check{\jmath}(i,g) = \arg\max_{j \in \mathcal{Q}} \left\{ T^g_{\boldsymbol{Y}^i \to \boldsymbol{Y}^j | \boldsymbol{B}} \right\} . \tag{4.12}$$

Over a number of games G, we select the responder agent $\check{\jmath}(i)$ to whom maximal information was transferred by $\boldsymbol{Y}^i$ most frequently, as the mode of the series $\{\check{\jmath}(i,1), \ldots, \check{\jmath}(i,G)\}$. Finally, we consider the average information transfer between the two processes $\boldsymbol{Y}^i$ and $\boldsymbol{Y}^{\check{\jmath}(i)}$ across all games:

$$T_{\boldsymbol{Y}^i \to \boldsymbol{Y}^{\check{\jmath}(i)} | \boldsymbol{B}} = \frac{1}{G} \sum_{g=1}^{G} T^g_{\boldsymbol{Y}^i \to \boldsymbol{Y}^{\check{\jmath}(i)} | \boldsymbol{B}} . \tag{4.13}$$

The pairs $(i, \check{\jmath}(i))$ identified for each agent i in team $\mathcal{P}$ treated as a source are combined in an "information-source diagram" $\check{G}_{\mathcal{P},\mathcal{Q}} = (\mathcal{P}, \mathcal{Q}, \check{\mathcal{E}}_{\mathcal{P}}, \check{\mathcal{E}}_{\mathcal{Q}})$ where the edge set $\check{\mathcal{E}}_{\mathcal{P}} = \{i \to \check{\jmath}(i) : i \in \mathcal{P}\}$.

The intuition in this case is the same as in the previous subsection — the difference is that now we identify the highest responder agent, having selected a source. In general, the agent i in team $\mathcal{Q}$ may be the most informative source for the agent j in team $\mathcal{P}$, but the agent j may be not the best responder to the agent i among all possible responders in team $\mathcal{P}$, and vice versa.

While an information-sink diagram reflects more where the information tends to be transferred to, an information-source diagram tends to depict where the information is transferred from.

4.5.3 *Information-sink and -source diagrams as efficient simplifications*

Neither of the diagrams presents a complete story, highlighting only a small part of the overall information dynamics. That is, they are a representation of directed functional connectivity as discussed in Sec. 2.3.1. In Sec. 2.3.2 we discussed more

comprehensive network diagrams derived from some form of multivariate connectivity analysis, which seek to infer a circuit model which can replicate and indeed explain the time-series of the nodes in the network [82, 219]. Furthermore, we note that information-sink and -source diagrams ignore interactions within teams, and of course both these and full multivariate analysis represents observational correlations rather than strict causation (by specifically using a Wiener-Granger interpretation of causality, see Sec. 2.3 for detail).

Nevertheless, we believe that the interaction diagrams presented here are valuable, as a simplified view of the full effective network representation of the set of agents influencing and influenced by each other agent: they are particularly simple and easy to interpret, and crucially are computationally efficient. Specifically, for an information-sink diagram every agent has an incoming edge, and for an information-source diagram every agent has an outgoing edge, representing the strongest respective in or outgoing interactions for that agent. Also, these diagrams provide a significantly more efficient analysis than full effective network inference, computing only $\mathcal{O}(M^2)$ transfer entropies rather than additionally examining higher-order interactions, and avoiding additional computations for statistical significance measurements. Such efficiency is a particularly important consideration if such a method is to be used online during Robocup games in the future.

4.6 STATE-SPACE COHERENCE DIAGRAMS

The study of Lizier et al. [152] diagrammatically demonstrated that more coherent structures in state-space plots can be observed in systems (cellular automata) with higher degrees of complexity. Motivated by these methods, we investigate coherent information structures observed as patterns in a state-space formed by tactical information dynamics measures, aiming to reveal structure in the relationship between the team's rigidity and responsiveness. Positional dynamics of each agent depends in general on their tactical role in the game and are quantified by their responsiveness (measured by information transfer) and rigidity (measured by information storage). These two measures will specifically be used in forming the two-dimensional state-space, where relative responsiveness is plotted as a function of relative rigidity (see 4.1 for definitions).

Identifying a coherent structure in the relationship between responsiveness and rigidity allows us to classify coherent dynamic situations, in the context of distributed communications. For example, dynamics of agents performing in a specific role, such as attackers, may be characterised by both lower rigidity and lower

responsiveness to opponent's defenders than dynamics of other agents. Coherence diagrams are intended to visualise such dynamic clustering in the state-space formed by the corresponding information-theoretic measures. Furthermore, once these dynamic clusters are highlighted as sub-regions of the space, it is possible to "zoom in" by considering correlations of the points within these regions with the scorelines of the corresponding games, and identifying which regions (clusters) map to more successful games.

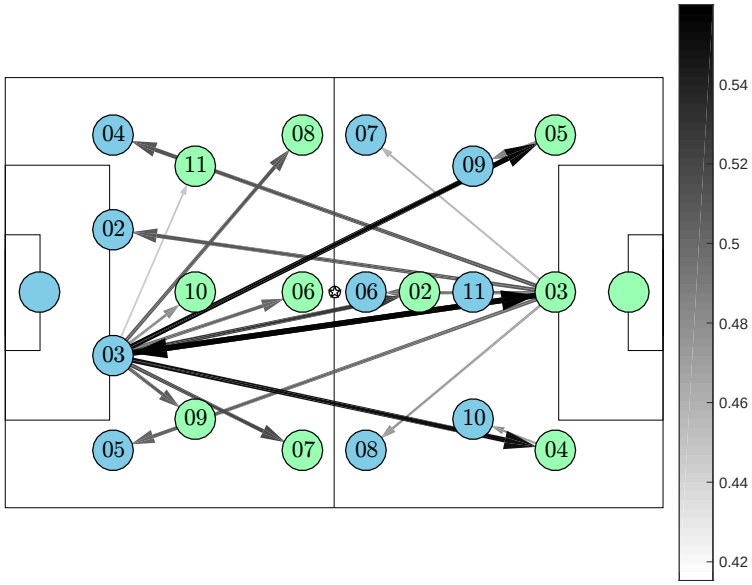
In particular, we introduce two different state-space plots (coherence diagrams) intended to capture different spatio-temporal interactions across teams: *1)* tactical information dynamics in relation to tactical roles (defender, midfielder, attacker); and *2)* information dynamics partitions correlated with the scorelines. The state-space diagrams for each team are produced by computing the following state-space points: $(\delta A_d, \delta T_{a \rightharpoonup d})$, $(\delta A_m, \delta T_{m \rightharpoonup m})$ and $(\delta A_a, \delta T_{d \rightharpoonup a})$. Then, the first coherence diagram is given by plotting these points on the respective axes with a distinct colour for each tactical role, i.e. defenders, midfielders and attackers (cf. Fig. 4.6.). Another coherence diagram is given for each tactical role by selecting the points corresponding to the roles, and colour-mapping them with the corresponding scoreline δS (Figs. 4.7 and 4.7).

4.7 RESULTS

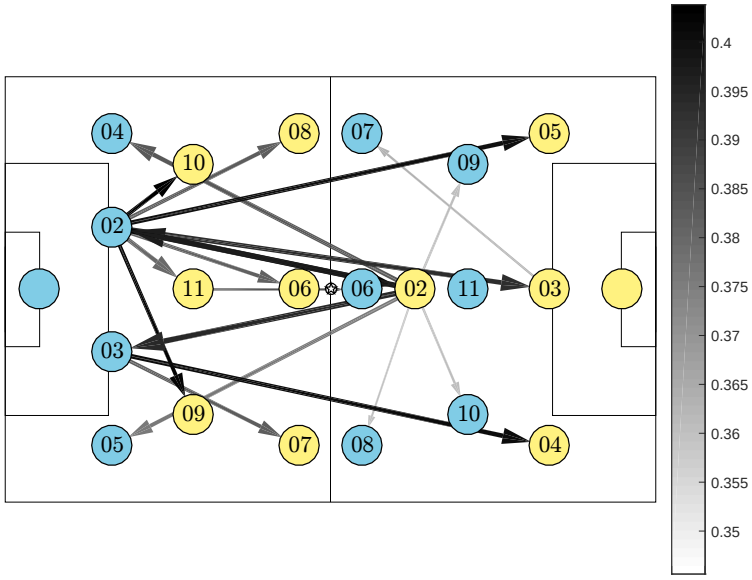
To compute the measures described in previous sections, and produce interaction diagrams and state-space coherence diagrams, we carried out multiple iterative experiments using simulated RoboCup data. The agents in these simulations are distributed, where each client is fed partial information from a server and makes decisions autonomously.

The experiments below match up team Gliders [186] against teams Cyrus [119] and HELIOS [5], denoted by $\mathcal{G}$, $\mathcal{C}$ and $\mathcal{H}$, respectively. Gliders were the winner (champion) of RoboCup-2016 and the runner-up (vice-champion) team for RoboCup-2014, while HELIOS and Cyrus were fourth and fifth ranked teams in 2014.

All information-theoretic measures were computed using the Java information dynamics toolkit (JIDT) [148], with finite history lengths $\kappa = 1$. For the information-sink and base diagrams, kernel estimation was used with a kernel width of 0.4 standard deviations of the data. For the state-space coherence diagrams, Kraskov-Stögbauer-Grassberger estimation [93, 131] was used with four nearest neighbours.

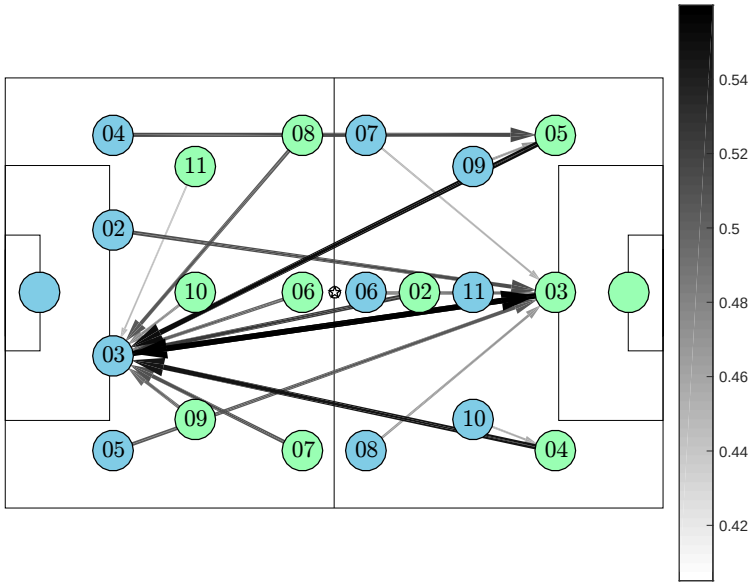


(a) Information-sink diagram for Gliders (blue) and Cyrus (green).

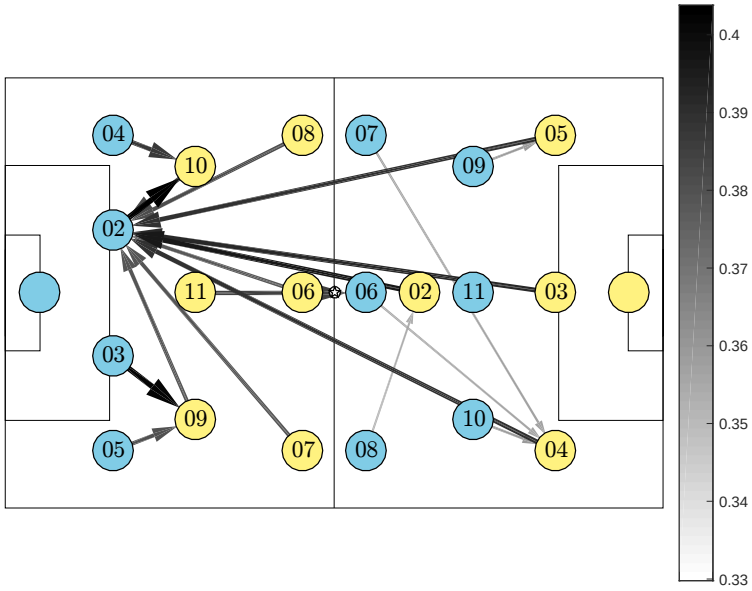


(b) Information-sink diagram for Gliders (blue) and HELIOS (yellow).

Figure 4.2: Information-sink diagrams. Arrows represent highest information transfer between players. Grayscale colormap is used to indicate the strength of transfer, varying smoothly from white (weakest) to black (strongest). Example of the most pronounced interactions: all Cyrus players strongly respond to the motion of the right centre-back of Gliders (player 03), indicating the strong asymmetry of Cyrus tactics in preferring to play on their left wing.



(a) Information-source diagram for Gliders (blue) and Cyrus (green).



(b) Information-source diagram for Gliders (blue) and HELIOS (yellow).

Figure 4.3: Information-source diagrams. Arrows represent highest information transfer between players. Grayscale is used to indicate the strength of transfer, varying smoothly from white (weakest) to black (strongest). Example of the most pronounced interactions: all HELIOS players strongly drive the left centre-back of Gliders (player 02), indicating the strong asymmetry of HELIOS dynamics in preferring to play on their right wing.

4.7.1 *Interaction diagrams*

Figure 4.2 presents the information-sink interaction diagram $\hat{G}_{\mathcal{C},\mathcal{G}}$ and the information-source interaction diagram $\check{G}_{\mathcal{C},\mathcal{G}}$, built over 400 games between Cyrus and Gliders. Analogously, Fig. 4.3 shows the information-sink interaction diagram $\hat{G}_{\mathcal{H},\mathcal{G}}$ and the information-source interaction diagram $\check{G}_{\mathcal{H},\mathcal{G}}$, built over 400 hundred games between HELIOS and Gliders. The nodes in each diagram are shown in positions roughly corresponding to the players' formation on the field, e.g., Gliders follow the 4-3-3 formation with four defenders playing line defence, three midfielders and three attackers, whereas Cyrus and HELIOS utilise one of the defenders (the player with the number 02) as a defensive midfielder, thus loosely following 3-4-3 formation with four midfielders.

Several interesting observations can be made. To some extent, the interaction diagrams exhibit lateral symmetry, which is expected given the symmetric formations of the teams. However and perhaps more importantly, there are some clearly asymmetric connections. For example, the most pronounced interactions are observed with all Cyrus players strongly responding to the motion of the right centre-back of Gliders (player 03), which reveals the strong asymmetry of Cyrus dynamics in preferring to play on the their left wing. This is a feature which has been successfully exploited by Gliders in allocating suitable defensive resources on this wing, resulting in a statistically significant performance gain (an increase in the average goal difference from 1.55 ± 0.03 to 1.80 ± 0.02, over more than 6000 games, i.e., an improvement of 16%). Similarly, all HELIOS players strongly "drive" the left centre-back of Gliders (player 02), also highlighting the strong asymmetry of HELIOS dynamics in preferring to play on their right wing. Again, this can be tactically exploited.

In Fig. 4.2a it is evident that the defenders are the most responsive of both teams, showing that the games between Gliders and Cyrus unfold outside of the midfield, see Fig. 4.4. On the other hand, Fig. 4.2b reveals a more disordered responsiveness between the teams, indicating that a lot of interactions occur in midfield during the games between Gliders and HELIOS. We also point out that the highest information transfer value computed over the games between Gliders and HELIOS (~ 0.4 bits in Fig. 4.2b and Fig. 4.3b) is less than the lowest value computed over the games between Gliders versus Cyrus (~ 0.42 bits in Fig. 4.2a and Fig. 4.3a). This means that the Gliders and HELIOS players are more independent in their respective motions on average.

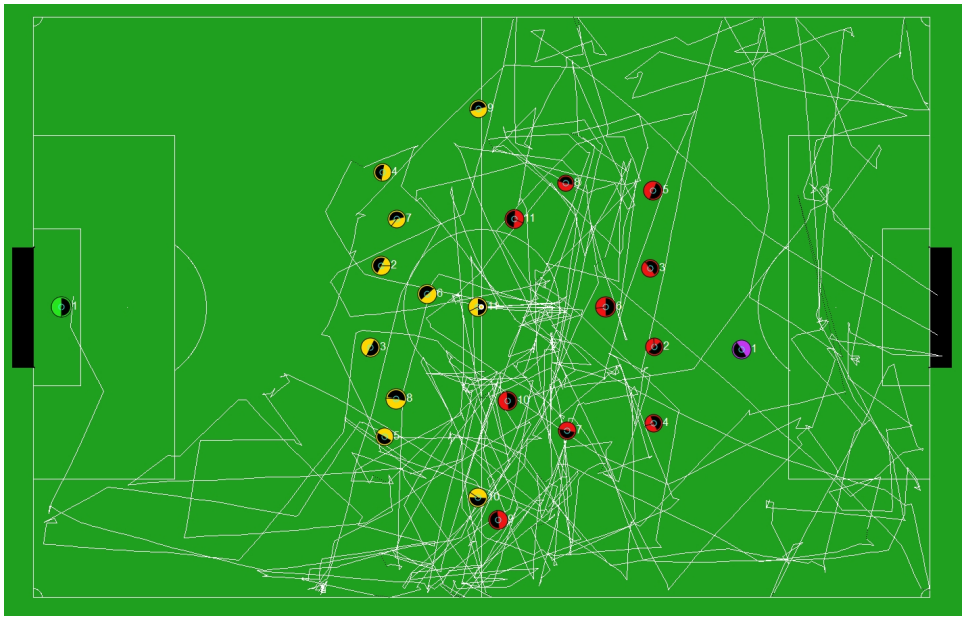


Figure 4.4: A white curve traces the ball motion during an entire game between Gliders (left) versus Cyrus (right). Note the asymmetry in Cyrus attack, as well as a significant ball trace outside of the midfield.

Specifically, Gliders attackers mostly respond to Cyrus defenders, and Gliders midfielders and defenders respond most to the Cyrus central defender (player 03) which is typically moving across wider areas, often playing a "sweeper" role.[4]

This coupling is similar to patterns observed in Gliders and HELIOS dynamics; however, the interactions are generally weaker and are spread amongst more players than just one central defender, because both HELIOS central defenders take an active part in defending the area.

In summary, the findings demonstrate applicability of the information dynamics measures to analysis of dynamics of multi-agent teams, revealing the player pairs with most intense interactions and the extent of the resultant dependencies.

4.7.2 *Correlation with performance*

In this subsection, we correlate measures of relative responsiveness (either tactical role-by-role or team overall), as well as rigidity, with the game scorelines, and identify the tactical roles which impacted on the games more. That is, we compute the correlation between a series of game scorelines and a series of information

4 The sweeper (or libero) is a more versatile centre-back who "sweeps up" the ball if an opponent manages to breach the defensive line. This position is more fluid than that of other defenders who man-mark their designated opponents.

Table 4.2: Correlation of information dynamics measures with scoreline.

Team ($\mathcal{Q}$)	$\delta T(\mathcal{G},\mathcal{Q})$	$\delta T_{a\rightharpoonup d}(\mathcal{G},\mathcal{Q})$	$\delta T_{m\rightharpoonup m}(\mathcal{G},\mathcal{Q})$	$\delta T_{d\rightharpoonup a}(\mathcal{G},\mathcal{Q})$
Cyrus ($\mathcal{C}$)	-0.601	0.607	0.338	-0.427
HELIOS ($\mathcal{H}$)	-0.466	0.455	0.616	0.211

(a) Transfer based measures.

Team ($\mathcal{Q}$)	$\delta A(\mathcal{G},\mathcal{Q})$	$\delta A_d(\mathcal{G},\mathcal{Q})$	$\delta A_m(\mathcal{G},\mathcal{Q})$	$\delta A_a(\mathcal{G},\mathcal{Q})$
Cyrus ($\mathcal{C}$)	-0.613	0.223	-0.558	-0.616
HELIOS ($\mathcal{H}$)	-0.683	0.380	-0.703	-0.642

(b) Storage based measures.

dynamics values for a game. For clarity, we discuss mainly the interpretation of the correlations in the context of the Gliders performance.

Table 4.2 presents the correlation coefficients between scorelines and various information-based measures which were summarised in Tab. 4.1. Generally, the observed correlations are consistent for all measures across both opponent teams, with the exception of $\delta T_{d\rightharpoonup a}(\mathcal{G},\mathcal{Q})$, which differ in sign. All of the correlations displayed in Tab. 4.2 are statistically significant at $p = 0.01$ with a one-tailed test Bonferroni corrected for 16 comparisons. We begin our analysis with the measures based on information transfer.

Overall, the higher responsiveness of a team *to all opponent players* is detrimental to their winning chances, that is, the more responsive Gliders are on average to opponents, indicated by the higher $\delta T(\mathcal{G},\mathcal{Q})$, the less positive the scoreline δS. However, when looking at tactical lines role-by-role, e.g. comparing the relative responsiveness of defenders to attackers between two teams, we observe in general the opposite effect: higher responsiveness is an indication of winning. In particular, if the Gliders defenders are more responsive to their immediate opposing line of Cyrus (or HELIOS) attackers than Cyrus (or HELIOS) defenders are to Gliders attackers, i.e., $T_{\mathbf{Y}^{a(\mathcal{Q})}\rightarrow\mathbf{Y}^{d(\mathcal{G})}} > T_{\mathbf{Y}^{a(\mathcal{G})}\rightarrow\mathbf{Y}^{d(\mathcal{Q})}}$, then team Gliders has a higher chance of winning. Similarly, the Gliders tend to win if its midfielders are more responsive than their midfield opposition (either Cyrus or HELIOS), i.e. positive $\delta T_{m\rightharpoonup m}(\mathcal{G},\mathcal{Q})$ can be used as a precursor for a winning prediction. This means that high relative responsiveness across tactical lines is indicative of a behaviour positively contributing to the performance (e.g., defenders are successfully marking the opponent attackers, or midfielders are successfully finding open zones amongst opponent midfielders in anticipation of teammate passes using Voronoi diagrams [186]), while

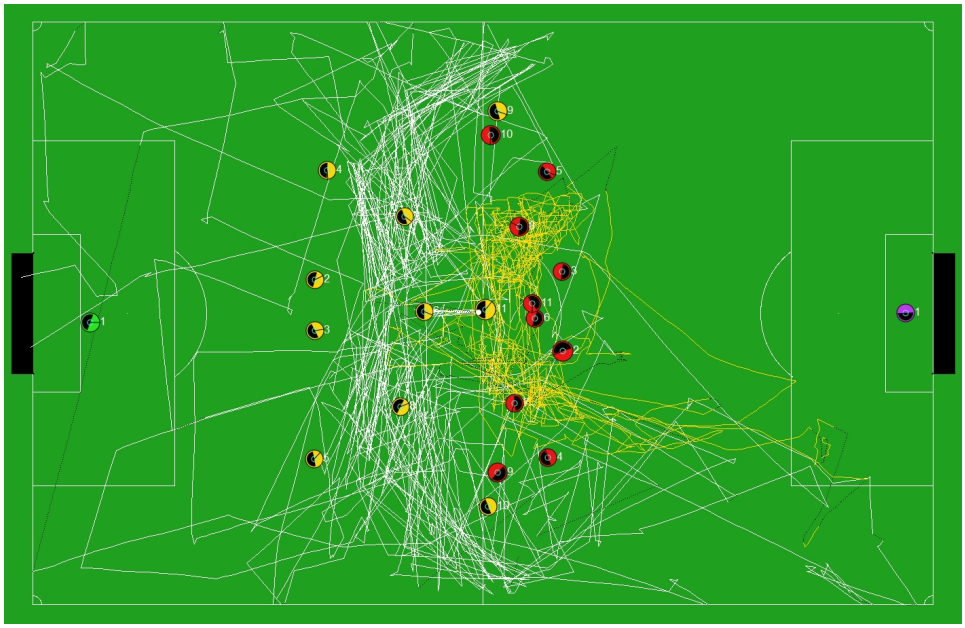


Figure 4.5: Motion trace diagram. A yellow-black trace curve represents the motion of the centreforward (player 11) of the left team (HELIOS), during an entire game (solid yellow for regular "play-on" time points, and dotted black for not "play-on" times, e.g. free kicks). A white curve traces the ball motion during the game. Note that the majority of both traces lies in midfield.

the overall high relative responsiveness across all players $\delta T(\mathcal{G}, \mathcal{Q})$ may suggest an adverse outcome, due to an excessive unstructured dependence on the opposition.

The relative responsiveness $\delta T_{d \rightharpoonup a}(\mathcal{G}, \mathcal{C})$ is negatively correlated with the scoreline, and deserves a separate explanation. The lower responsiveness $T_{\boldsymbol{Y}^{d(\mathcal{C})} \to \boldsymbol{Y}^{a(\mathcal{G})}} < T_{\boldsymbol{Y}^{d(\mathcal{G})} \to \boldsymbol{Y}^{a(\mathcal{C})}}$ means that Gliders attackers are less predictable in their response to the opponent defenders than Cyrus attackers, and this opens up more scoring opportunities. In other words, unpredictability of attackers' motion is positive and characteristic of an opportunity-seeking behaviour. This is in contrast to the responsive tracking behaviour of defenders which are typically engaged in trying to actively mark the opponents attackers.

The relative responsiveness for attackers $\delta T_{d \rightharpoonup a}(\mathcal{G}, \mathcal{H})$ is still positively correlated with the scoreline, and should be interpreted in the context of the interaction diagrams which indicated that in the games between Gliders and HELIOS, most of the action occurred in midfield anyway, and so the attackers are mostly engaged in midfielder-like behaviours, as can be seen in Fig. 4.5. Hence it may be expected that the high relative responsiveness of attackers in this contest is still positively related to performance.

The rigidity of a team *as a whole* is also detrimental to their goal scoring capabilities, as shown by δA negatively correlated to the scorelines in both considered

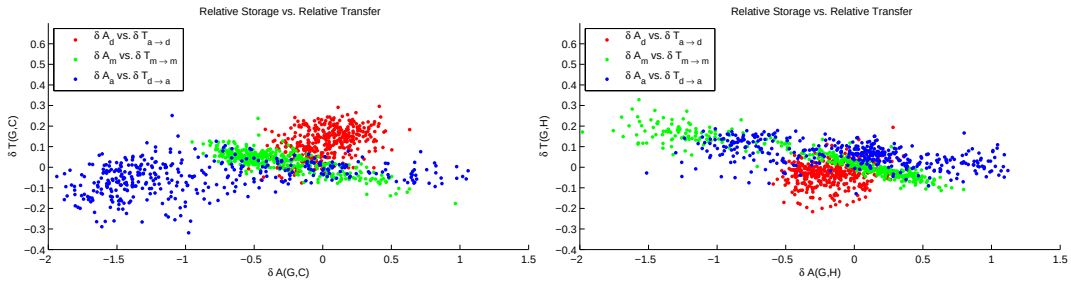
contests. This is an expected result for the team players which are moving highly predictably with respect to their positional histories. An analysis for each role reveals that rigidity of either midfielders (δA_m) or attackers (δA_a) is also negatively correlated with performance. However, rigidity of defenders movements (δA_d) is a positive feature, consistent across both match-ups. This can be explained by a specific tactical behaviour, employed by the team Gliders: "line defense" which is highly dependent on an ability to create offside traps by a simultaneous motion of all four defenders. This defensive tactic produces more synchronous actions and results in a successful but predictable behaviour of each player (on average), captured in turn by their rigidity. As long as this rigidity is not exploited by the opponents, the performance is likely to remain positively correlated.

The notion that the scoreline is correlated with a team's information dynamics is an important consequence of this research. Considering Reichenbach's theorem, we can deduce that either: the scoreline causes information dynamics; the information dynamics causes the scoreline; or, there is a common cause for the two measures of performance. It is unlikely for the former two cases and thus we conjecture that the information dynamics and scoreline are proxy to an underlying cause. Further, our results support the hypothesis of intrinsic motivation in psychology and reinforcement learning [49], whereby it is shown that an embodied agent that is both intrinsically and extrinsically motivated is more adept at problem solving. In the case of team dynamics, information dynamics is an intrinsic reward and scoring goals is an extrinsic reward. This relates to the work of Zahedi et al. [266], who used a linear combination of predictive information to speed up the learning process of an embodied agent.

4.7.3 *State-space coherence diagrams*

Figure 4.6 shows tactical information dynamics, i.e., state-space coherence diagrams for all tactical roles, while Figs. 4.7 and 4.7 shows partitioned information dynamics: state-space coherence diagrams for specific tactical roles, colour-coded with the scorelines.

Both state-space coherence diagrams in Fig. 4.6 clearly show separation among three tactical roles: defenders, midfielders and attackers. Each tactical role is clustered well in each of the contests. Defenders (shown in red) tend to have low relative rigidity and low relative responsiveness. That is, defenders of the competing teams in each contest (Gliders vs Cyrus and Gliders vs HELIOS) do not differ much in their rigidity and responsiveness, except that the Gliders' defenders are more responsive than Cyrus' defenders. Midfielders (shown in green) consistently

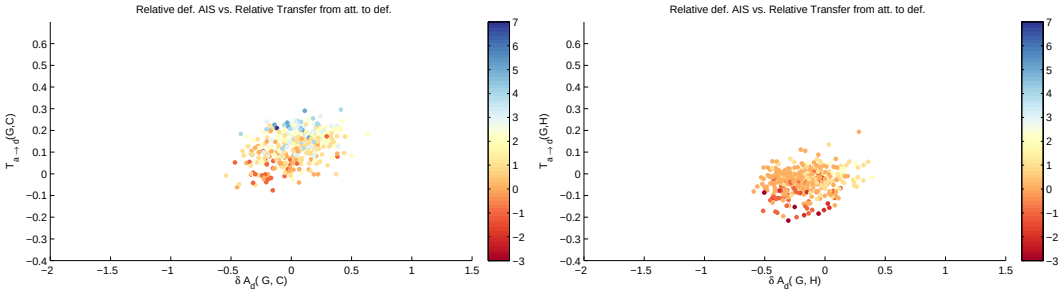


(a) $\delta A(\mathcal{G}, \mathcal{C})$ vs $\delta T(\mathcal{G}, \mathcal{C})$ (b) $\delta A(\mathcal{G}, \mathcal{H})$ vs $\delta T(\mathcal{G}, \mathcal{H})$

Figure 4.6: Tactical information dynamics: state-space coherence diagrams of relative responsiveness $\delta T(\mathcal{G}, \mathcal{Q})$ as a function of relative rigidity $\delta A(\mathcal{G}, \mathcal{Q})$, with two opponents of team Gliders ($\mathcal{G}$): team Cyrus ($\mathcal{C}$) and team HELIOS ($\mathcal{H}$). In the diagrams, red points are used for relative responsiveness versus relative rigidity with respect to Gliders' defenders, with green points for Gliders' midfielders and blue for attackers.

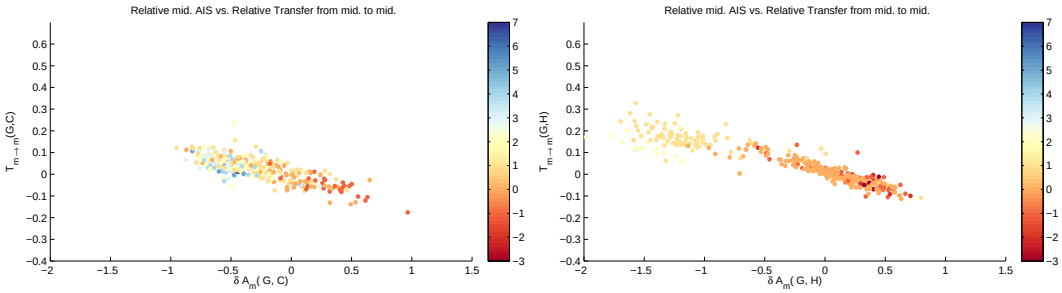
occupy a well-defined narrow region showing that an increase in relative rigidity is correlated with a decrease in relative responsiveness, in both contests. Gliders' midfielders appear to be slightly more responsive and less rigid than HELIOS' midfielders. Finally, attackers (shown in blue) are clustered differently in two contests. In games between Gliders' and Cyrus' low relative rigidity is correlated with a wider spread of relative responsiveness which tends to be negative. In other words, when Gliders' attackers are less rigid than Cyrus attackers, they are also less responsive: this is indicative of their more explorative behavior around and within their opponent's penalty area. In contrast, this feature is not observed in the diagram for Gliders vs HELIOS; moreover, there is a correlation between relative rigidity and responsiveness similar to the one in the midfielders' cluster. This reinforces an earlier observation that in the games between Gliders and HELIOS, the attackers often play in the midfield. Importantly, these state-space coherence diagrams allow us to examine average role-based multi-agent dynamics across games. They can be considered as a means to cluster dynamic processes in an abstract state-space and identify salient features of competing tactical formations.

Now we turn our attention to information dynamics partitioned for each tactical role and their correlation with the scorelines. The partitioned diagrams in Figs. 4.7 and 4.7 reveal how the differences in rigidity and responsiveness are consistently related to the performance, across both contests. For example, there is a clear correlation between better performance and higher responsiveness and higher rigidity of defenders, as shown in Fig. 4.7a and 4.7b. As mentioned earlier, a positive contribution of the higher rigidity is not counter-intuitive as it results from synchronous, and hence more predictable on average, movement of each defender following the



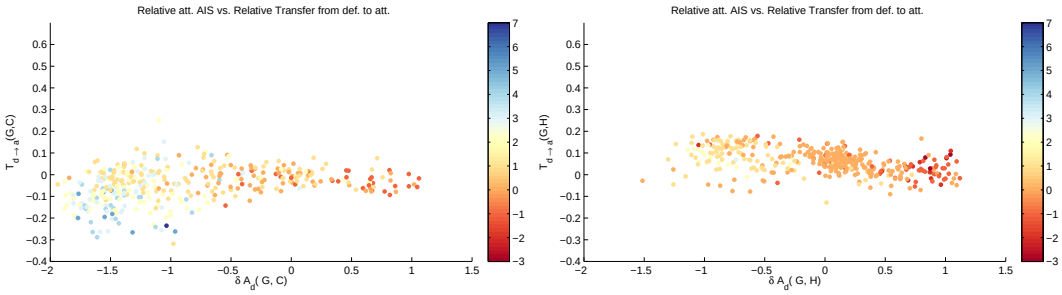
(a) Attackers to Defenders:
$\delta A_d(\mathcal{G},\mathcal{C})$ vs $\delta T_{a\rightharpoonup d}(\mathcal{G},\mathcal{C})$ vs $S_{\mathcal{G}}$

(b) Attackers to Defenders:
$\delta A_d(\mathcal{G},\mathcal{H})$ vs $\delta T_{a\rightharpoonup d}(\mathcal{G},\mathcal{H})$ vs $S_{\mathcal{G}}$



(c) Midfielders to Midfielders:
$\delta A_m(\mathcal{G},\mathcal{C})$ vs $\delta T_{m\rightharpoonup m}(\mathcal{G},\mathcal{C})$ vs $S_{\mathcal{G}}$

(d) Midfielders to Midfielders:
$\delta A_m(\mathcal{G},\mathcal{H})$ vs $\delta T_{m\rightharpoonup m}(\mathcal{G},\mathcal{H})$ vs $S_{\mathcal{G}}$



(e) Defenders to Attackers:
$\delta A_a(\mathcal{G},\mathcal{C})$ vs $\delta T_{d\rightharpoonup a}(\mathcal{G},\mathcal{C})$ vs. $S_{\mathcal{G}}$

(f) Defenders to Attackers:
$\delta A_a(\mathcal{G},\mathcal{H})$ vs $\delta T_{d\rightharpoonup a}(\mathcal{G},\mathcal{H})$ vs $S_{\mathcal{G}}$

Figure 4.7: Left column: Partitioned information dynamics: state-space coherence diagrams for specific tactical roles, with the colour-mapping showing the scoreline difference (positive means the Gliders won). Relative responsiveness $\delta T(\mathcal{G},\mathcal{C})$ is a function of relative rigidity $\delta A(\mathcal{G},\mathcal{C})$ for Gliders ($\mathcal{G}$) versus Cyrus ($\mathcal{C}$). Right column: Partitioned information dynamics: state-space coherence diagrams for specific tactical roles, with colour-mapping of the correlation with scorelines. Relative responsiveness $\delta T(\mathcal{G},\mathcal{H})$ is a function of relative rigidity $\delta A(\mathcal{G},\mathcal{H})$ for Gliders ($\mathcal{G}$) versus HELIOS ($\mathcal{H}$).

"line defense" tactic, enabling efficient offside traps for the opposition. On the other hand, for the midfielders, there is a clear correlation between better scorelines and lower rigidity as well as higher responsiveness, as shown in Fig. 4.7c and 4.7d. That is, when Gliders' midfielders are less rigid or more responsive than their opponent's midfielders, the Gliders team tends to win. Finally, it is evident that when Gliders' attackers are less rigid and less responsive than Cyrus' attackers, Fig. 4.7e, the team benefits, while in the games vs HELIOS the correlation with performance is mostly observed for lower rigidity, Fig. 4.7f. The difference between two contests is again due to the fact that Gliders' attackers are typically restrained to playing in midfield in the games vs HELIOS. These partitioned diagrams provide another useful tool in clustering the multi-agent dynamics and classifying the games in terms of tactical behaviour.

4.8 SUMMARY

In this chapter we used information theory to examine multi-agent dynamics where the system is assumed to be fully observable. In particular, we used information dynamics in studying the implicit interactions in simulated robotic soccer teams. This was achieved by using interaction diagrams, which represent the implicit communication between agents, as well as coherence plots, which showed correlation between the information dynamics and the scoreline. At the start of this chapter, we introduced these information-theoretic measures in the context of DBN model selection.

The autonomous agents in this chapter reason over their actions in order to score goals. However, our focus here was to provide general-purpose analysis techniques that study this process by simply observing trajectories. Thus, reasoning in the context of this work involves formulating a logical model about the memory and communication within a multivariate process, rather than studying and optimising the underlying (causal) mechanisms for achieving an objective (this is left for later chapters). The following chapter further explores this idea by performing DBN model selection where the system is partially observable.

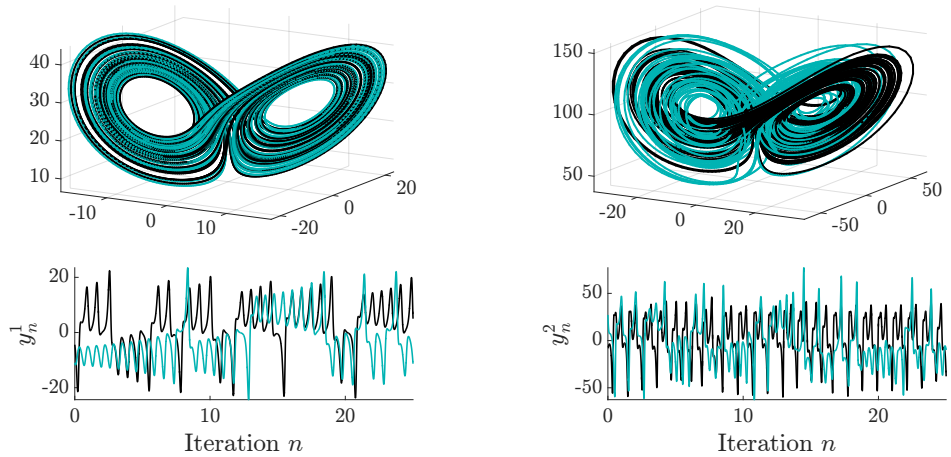
5 INFORMATION-THEORETIC MODEL SELECTION IN DISTRIBUTED NONLINEAR SYSTEMS

In the previous chapter we investigated observed multivariate stochastic processes where each component was assumed to be finite-order Markovian. Here, we study the problem of learning the graphical structure of a distributed system where each subsystem comprises a latent process observed through a filter. Given that certain assumptions are made about the dynamics of the univariate processes, the theory presented here is less related to classical statistics and more to nonlinear systems (physics) literature.

5.1 OVERVIEW

In this chapter we exploit the properties of discrete-time multivariate dynamical systems in inferring coupling between latent variables in a DAG. Our main focus is to analytically derive a measure (score) for evaluating the fitness of a candidate DAG, given data. We assume that the data are generated by a certain family of multivariate dynamical systems and are thus able to overcome the issue of latent variables faced by established structure learning algorithms. That is, under certain assumptions of the dynamical system, we are able to employ time delay embedding theorems (see Sec. 3.3.1) to compute our scores.

As mentioned in Sec. 3.6.1, structure learning for DBNs is commonly expressed via either information theory or Bayesian statistics. Exact methods are known for fully observable systems; however, these are not applicable in the more expressive case when the state variables are latent. Drawing on the information-theoretic perspective, our main result in this chapter is a tractable form of the KL divergence function for certain distributed nonlinear dynamical systems. We establish this result by first representing a family of discrete-time multivariate dynamical systems as DBNs (termed a POSGDS). In this form, both the complete and factorised distributions cannot be directly computed due to the hidden system state. Thus, we employ state space reconstruction methods from differential topology to reformulate the KL divergence in terms of computable distributions.



(a) $\sigma = 10, \beta = 8/3, \rho = 28$

(b) $\sigma = 10, \beta = 8/3, \rho = 90$

Figure 5.1: Trajectory of a pair of coupled Lorenz systems. *Top row*: original state of the subsystems. *Bottom row:* time-series measurements of the subsystems. In each figure, the black lines represent an uncoupled simulation ($\lambda = 0$), and teal lines illustrate a simulation where the first (leftmost) subsystem was coupled to the second ($\lambda = 10$).

Computing the KL divergence involves evaluating the expected log-likelihood of the graph. We begin this chapter by showing that the log-likelihood and log-likelihood ratio can be expressed in terms of collective transfer entropy (3.25). By virtue of this, we are also able to directly compute the BIC [204] and the AIC [4] scoring functions, which could be used to achieve globally optimal approximations to quantify the quality of a candidate graph under certain assumptions. Following from this, we show that the KL divergence can be decomposed as the difference between stochastic interaction (3.28) and collective transfer entropy. Using this expression, we show that the maximum transfer entropy graph is the most likely to have generated the data, and build on this result to present a scoring function for evaluating candidate graphs based on a dataset. This is then experimentally validated using the toy examples of a Lorenz-Rössler system and a network of coupled Lorenz attractors (Fig. 5.1) of up to four nodes.

5.2 PROBLEM STATEMENT

We model multivariate dynamical systems as POSGDSs. With this model, we can express the time evolution of the dynamical system as a stationary DBN, and perform inference and learning on the subsequent graph. We formally state the net-

work of dynamical systems as a special case of the sequential graph dynamical system (GDS) [162][1] with an observation function for each vertex.

Definition 5.1 (Partially observable synchronous graph dynamical system POSGDS)**.** *A POSGDS is a tuple* $(G, \boldsymbol{x}_n, \boldsymbol{y}_n, \{f^i\}, \{\psi^i\})$ *that consists of:*

- *a finite, directed graph* $G = (\mathcal{V}, \mathcal{E})$ *with edge-set* $\mathcal{E} = \{E^i\}$ *and* M *vertices comprising the vertex set* $\mathcal{V} = \{V^i\}$*;*
- *a multivariate state* $\boldsymbol{x}_n = \{\boldsymbol{x}_n^i\}$*, composed of states for each vertex* V^i *confined to a* d^i*-dimensional manifold* $\boldsymbol{x}_n^i \in \mathcal{M}^i$*;*
- *an* M*-variate observation* $\boldsymbol{y}_n = \{y_n^i\}$*, composed of scalar observations for each vertex* $y_n^i \in \mathbb{R}$*;*
- *a set of local maps* $\{f^i\}$ *of the form* $f^i : \mathcal{M} \to \mathcal{M}^i$*, which update synchronously and induce a global map* $f : \mathcal{M} \to \mathcal{M}$*; and*
- *a set of local observation functions* $\{\psi^1, \psi^2, \ldots, \psi^M\}$ *of the form* $\psi^i : \mathcal{M}^i \to \mathbb{R}$*.*

Without loss of generality, we can use local functions to describe the time evolution of the subsystems:

$$\boldsymbol{x}_{n+1}^i = f^i(\boldsymbol{x}_n^i, \mathrm{pa}(\boldsymbol{X}_n^i)) + \boldsymbol{v}_{f^i} \tag{5.1}$$

$$y_{n+1}^i = \psi^i(\boldsymbol{x}_{n+1}^i) + v_{\psi^i}. \tag{5.2}$$

Here, $\boldsymbol{v}_{f^i}$ is strictly i.i.d. additive noise and v_{ψ^i} is noise that is either i.i.d. or dependent on the state, i.e., $v_{\psi^i}(\boldsymbol{x}_{n+1}^i)$. The subsystem dynamics (5.1) are therefore a function of the subsystem state $\boldsymbol{x}_n^i$ and the subsystem parents' state $\mathrm{pa}(\boldsymbol{X}_n^i)$ at the previous time index such that $f^i : \mathcal{M}^i \times_j \mathcal{M}^{ij} \to \mathcal{M}^i$. Each subsystem observation is given by (5.2). We assume the functions $\{f^i\}$ and $\{\psi^i\}$ are invariant w.r.t. time and thus the graph G is stationary.

The time evolution of a POSGDS can be modelled as a DBN. First, each subsystem vertex V^i has an associated state variable $\boldsymbol{X}_n^i$ and observation variable Y_n^i; the parents of subsystem V^i are denoted $\mathrm{Pa}(V^i)$. Since the graph $G_\to$ is stationary and synchronous, parents of $\boldsymbol{X}_{n+1}^i$ come strictly from the preceding time slice, and additionally $\mathrm{Pa}_{G_\to}(Y_{n+1}^i) = \boldsymbol{X}_{n+1}^i$. Thus, we can build the edge set $\mathcal{E} = \{E^1, E^2, \ldots, E^M\}$ in the POSGDS by means of the DBN. That is, each edge subset E^i is built by the DBN edges

$$E^i = \{V^j \to V^i : X_n^j \in \mathrm{Pa}_{G_\to}(\boldsymbol{X}_{n+1}^i) \wedge V^j \in \mathcal{V} \setminus V^i\},$$

1 In the original manuscripts [52, 57], these were termed a synchronous update GDS, however have added *partially observable* to clarify that each subsystem has a latent state and to use terminology more in line with decision-theoretic nomenclature, such as POMDPs.

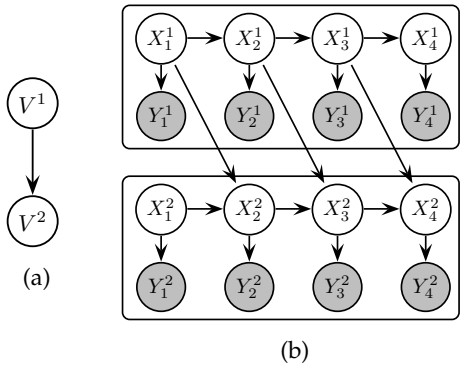


Figure 5.2: Representation of (5.2a) the POSGDS with two vertices (V^1 and V^2), and (5.2b) the rolled-out DBN of the equivalent structure. Subsystems V^1 and V^2 are coupled by virtue of the edge $\boldsymbol{X}_n^1 \to \boldsymbol{X}_{n+1}^2$.

so long as G forms a DAG.

As an example, consider the POSGDS in Fig. 5.2. The subsystem V^3 is coupled to both subsystem V^1 and V^2 through the edge set $\mathcal{E} = \{V^1 \to V^3, V^2 \to V^3\}$, shown in Fig. 5.2a. The time-evolution of this network is shown in Fig. 5.2b, where the top two rows (processes $\boldsymbol{X}^1$ and $\boldsymbol{Y}^1$) are associated with subsystem V^1, and similarly for V^2 and V^3. Recall that the BN $B = (G, \theta)$ comprises a graph G and parameter set θ. The distributions for the state (5.1) and observation (5.2) of M arbitrary subsystems can therefore be factorised according to (3.34):

$$p_B(\boldsymbol{z}_{n+1} \mid \boldsymbol{z}_n) = \prod_{i=1}^{M} p_B(\boldsymbol{x}_{n+1}^i \mid \boldsymbol{x}_n^i, \mathrm{pa}(\boldsymbol{X}_n^i))\; p_B(\boldsymbol{y}_{n+1}^i \mid \boldsymbol{x}_{n+1}^i). \tag{5.3}$$

The problem is then to derive a scoring function $g(B; \boldsymbol{Z})$ to learn the DBN based on the constrained conditional distributions (5.3). Since B is stationary, learning B is equivalent to learning the POSGDS. However, deriving this measure is not straightforward because the dataset $\boldsymbol{z}$ includes hidden variables $\boldsymbol{x}_n^i$. Thus, we rely on reconstruction theorems.

In the rest of the chapter we use simplified notation, given this constrained graph structure. Firstly, since our focus is on learning coupling between distributed systems, the superscripts refer to individual *subsystems*, not variables. Thus, although the 2TBN B is constrained such that $\mathrm{Pa}_G(Y_n^i) = \boldsymbol{X}_n^i$, the notation Y_n^{ij} denotes the *measurement variable* of the jth parent of subsystem i, e.g., in Fig. 5.2 an arbitrary ordering of the parents gives $Y_n^{3,1} = Y_n^1$ and $Y_n^{3,2} = Y_n^2$.

5.3 RECONSTRUCTION THEOREMS FOR LOG-LIKELIHOOD

As mentioned in Chapter 3, computing the KL divergence involves computing the expected log-likelihood. From (5.3) the expected log-likelihood decomposes as

$$\mathbf{E}_{\mathbf{Z}}[\ell(\hat{\theta}_G;\mathbf{Z})] = -N\sum_{i=1}^{M}\Bigg[\sum_{\boldsymbol{x}_{n+1}^i}\sum_{\mathrm{pa}(X_n^i)} p_{\hat{B}}(\boldsymbol{x}_{n+1}^i,\boldsymbol{x}_n^i,\mathrm{pa}(X_n^i))\log p_{\hat{B}}(\boldsymbol{x}_{n+1}^i \mid \boldsymbol{x}_n^i,\mathrm{pa}(X_n^i)) + \sum_{\boldsymbol{x}_{n+1}^i}\sum_{y_{n+1}^i} p_{\hat{B}}(y_{n+1}^i,\boldsymbol{x}_{n+1}^i)\log p_{\hat{B}}(y_{n+1}^i \mid \boldsymbol{x}_{n+1}^i)\Bigg]. \tag{5.4}$$

The log-likelihood function (5.4) involves distributions over latent variables, and thus we resort to state-space (attractor) reconstruction. First, Lemma 5.1 shows that a future observation from a given subsystem can be predicted from a sequence of past observations. Building on this result, we present a computable formulation of the 2TBN distribution (5.3) via Lemma 5.2. We then derive a tractable form of the log-likelihood function, presented in Lemma 5.1. It is then shown in Theorem 5.3 that these lemmas allow us to compute the general information criterion (3.43) discussed in Sec. 3.6.1.3.

In the following proofs and theorems, we will drop the dependence of a delay embedding map on the functions (f,ψ) (e.g., $\Phi(\boldsymbol{x}_n^i) = \Phi_{f^i,\psi^i}(\boldsymbol{x}_n^i)$) if it is clear based on context. However, it is important to note that, in general, we allow different dynamics f and observation functions ψ (and thus delay embedding parameters).

Lemma 5.1. *Consider a POSGDS $(G,\boldsymbol{x}_n,\boldsymbol{y}_n,\{f^i\},\{\psi^i\})$, where the graph G is a DAG. Each subsystem state follows the dynamics $\boldsymbol{x}_{n+1}^i = f^i(\boldsymbol{x}_n^i,\mathrm{pa}(X_n^i))$ and emits an observation $y_{n+1}^i = \psi^i(\boldsymbol{x}_{n+1}^i)$; the subsystem observation can be estimated, for some map T^i, by*

$$y_{n+1}^i = T^i\left(\Phi(\boldsymbol{x}_n^i),\Phi(\mathrm{pa}(\mathbf{X}_n^i))\right). \tag{5.5}$$

Proof. Consider a forced system $\boldsymbol{x}_{n+1} = f(\boldsymbol{x}_n,\boldsymbol{w}_n)$ with forcing dynamics $\boldsymbol{w}_{n+1} = h(\boldsymbol{w}_n)$ and observation $y_n = \psi(\boldsymbol{x}_{n+1})$. The bundle delay embedding theorem [221, 222] states that the delay map $\Phi(\boldsymbol{x}_n,\boldsymbol{w}_n) = y_n^{(\kappa)}$ is an embedding for generic f, ψ, and h. Stark [221] proved this result in the case of forcing dynamics h that are independent of the state x.[2] Moreover, the noise can be considered an additional forcing system so long as $\boldsymbol{v}_f$ is i.i.d and v_ψ is either i.i.d or dependent on the state [222].

2 Stark [221] conjectures that the theorem should generalise to functions h that are not independent of x. To the best of our knowledge, this result remains to be proven.

Given a DAG G, any ancestor of the subsystem V^i is not dependent on V^i. As such, the sequence

$$\Phi\left(\boldsymbol{x}_n^i, \mathrm{pa}(\boldsymbol{X}_n^i)\right) = \Phi(\boldsymbol{x}_n^i) \tag{5.6}$$

and is an embedding, since the realisation $\mathrm{pa}(\boldsymbol{x}_n^i)$ is independent of $\boldsymbol{x}_n^i$. Let $\{\boldsymbol{X}_n^{ijk}\}_k$ be the index ordered set of parents of node $\boldsymbol{X}_n^{ij}$ (which itself is the jth parent of the node $\boldsymbol{X}_n^i$). Under the constraint that G is a DAG, where the state $\boldsymbol{x}_{n+1}^i = f^i(\boldsymbol{x}_n^i, \{\boldsymbol{x}_n^{ij}\}_j) + \boldsymbol{v}_{f^i}$, it follows from the bundle delay embedding theorem [221, 222] that there exists a map F^i that is well defined and a diffeomorphism between observation sequences. From (5.6) we can write this map

$$\begin{aligned}
\Phi(\boldsymbol{x}_{n+1}^i) &= \Phi\left(f^i\left(\boldsymbol{x}_n^i, \{\boldsymbol{x}_n^{ij}\}_j\right), \left\{f^{ij}\left(\boldsymbol{x}_n^{ij}, \{\boldsymbol{x}_n^{ijk}\}_k\right)\right\}_j\right) \\
&= \Phi\left(f^i\left(\Phi^{-1} \circ \Phi(\boldsymbol{x}_n), \Phi^{-1}\left(\Phi(\mathrm{pa}(\boldsymbol{X}_n^i))\right)\right)\right). \\
&= F^i(\Phi(\boldsymbol{x}_n), \Phi(\mathrm{pa}(\boldsymbol{X}_n^i))),
\end{aligned} \tag{5.7}$$

where the last $\kappa^i + \sum_j \kappa^{ij}$ components of F^i are trivial. Denote the first component as $T^i : \mathbb{R}^{\kappa^i} \times_j \mathbb{R}^{\kappa^{ij}} \to \mathbb{R}$, then we arrive at (5.5). □

Lemma 5.2. *Given an observed dataset $\boldsymbol{z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_N)$ where $\boldsymbol{y}_n \in \mathbb{R}^M$ are generated by a directed and acyclic POSGDS $(G, \boldsymbol{x}_n, \boldsymbol{y}_n, \{f^i\}, \{\psi^i\})$, the 2TBN distribution can be written as*

$$\prod_{i=1}^M p_{\hat{B}}(\boldsymbol{x}_{n+1}^i \mid \boldsymbol{x}_n^i, \mathrm{pa}(X_n^i)) \cdot p_{\hat{B}}(y_{n+1}^i \mid \boldsymbol{x}_{n+1}^i) = \frac{\prod_{i=1}^M p_{\hat{B}}(y_{n+1}^i \mid \Phi(\boldsymbol{x}_n^i), \Phi(\mathrm{pa}(\boldsymbol{X}_n^i)))}{p_{\hat{B}}(\boldsymbol{x}_n \mid \Phi(\boldsymbol{x}_n))}. \tag{5.8}$$

Proof. Let each subsystem (local) map $\Phi^i = \Phi_{f^i,\psi^i} : \mathcal{M} \to \mathbb{R}^{\kappa^i}$. The generalised time delay embedding theorem [68] states that, under certain technical assumptions, and given M inhomogeneous observation functions $\{\psi^1, \psi^2, \ldots, \psi^M\}$, the map

$$\Phi(\boldsymbol{x}_n) = (\Phi^1(\boldsymbol{x}_n), \Phi^2(\boldsymbol{x}_n), \ldots, \Phi^M(\boldsymbol{x}_n)) \tag{5.9}$$

is an embedding, where, at time index n, the local map is described by

$$\Phi^i(\boldsymbol{x}_n) = \boldsymbol{y}_n^{i,(\kappa^i)} = (\psi^i(\boldsymbol{x}_n), \psi^i(\boldsymbol{x}_{n-\tau^i}), \psi^i(\boldsymbol{x}_{n-2\tau^i}), \ldots, \psi^i(\boldsymbol{x}_{n-(\kappa^i-1)\tau^i})) \tag{5.10}$$

and $\sum_i \kappa^i = 2d + 1$ [68].[3] Therefore, the global map (5.9) is given by $\Phi(\boldsymbol{x}_n) = (\boldsymbol{y}_n^{i,(\kappa^i)})$ and there must exist an inverse map $\boldsymbol{x}_n = \Phi^{-1} \circ \Phi(\boldsymbol{x}_n)$. Given Lemma 5.1, the

3 The original proof [68] uses positive lags, however the authors note that the use of negative lags also applies (and should be used in the case of endomorphisms, see Sec. 3.3.1).

existence of Φ^{-1}, and since $\{\Phi^i(\boldsymbol{x}_n^i), \Phi^i(\mathrm{pa}(\boldsymbol{X}_n^i))\} \subseteq \Phi(\boldsymbol{x}_n)$ for all i, we arrive at the following equation:

$$\begin{aligned}\prod_{i=1}^{M} p_{\hat{B}}\left(Y_{n+1}^i = T^i\left(\Phi^i(\boldsymbol{x}_n^i), \Phi^i(\mathrm{pa}(\boldsymbol{X}_n^i))\right) \mid \Phi^i(\boldsymbol{x}_n^i), \Phi^i(\mathrm{pa}(\boldsymbol{X}_n^i))\right) \\ = p_{\hat{B}}\left(\boldsymbol{X}_n = \Phi^{-1} \circ \Phi(\boldsymbol{x}_n) \mid \Phi(\boldsymbol{x}_n)\right) \\ \times \prod_{i=1}^{M} p_{\hat{B}}\left(\boldsymbol{X}_{n+1}^i = f^i(\boldsymbol{x}_n^i, \mathrm{pa}(X_n^i)) \mid \boldsymbol{x}_n^i, \mathrm{pa}(X_n^i)\right) \\ \times \prod_{i=1}^{M} p_{\hat{B}}\left(Y_{n+1}^i = \psi^i(\boldsymbol{x}_{n+1}^i) \mid \boldsymbol{x}_{n+1}^i\right).\end{aligned} \tag{5.11}$$

Rearranging (5.11) gives the equality in (5.8). □

Lemma 5.2 shows that the distributions can be reformulated by conditioning on delay vectors. The RHS of (5.8) can be used to perform inference in the 2TBN (5.3). The numerator is a product of local CPDs of scalar variables, and can thus be computed by either counting (for discrete variables) or density estimation (for continuous variables). The denominator is used to compute the probability that the hidden state occured, given an observed delay vector; fortunately, Casdagli [44] established methods to compute this CPD for a variety of practical scenarios. Therefore, Lemma 5.2 provides a method to perform exact inference.

5.3.1 *Information-theoretic interpretation*

Using the delay vector representation of Lemma 5.2, we arrive at the following theorem.

Theorem 5.1. *Consider a POSGDS $(G, \boldsymbol{x}_n, \boldsymbol{y}_n, \{f^i\}, \{\psi^i\})$, where the graph G is a DAG. Each subsystem state follows the dynamics $\boldsymbol{x}_{n+1}^i = f^i(\boldsymbol{x}_n^i, \mathrm{pa}(X_n^i))$ and generates an observation $y_{n+1}^i = \psi^i(\boldsymbol{x}_{n+1}^i)$; a complete dataset is given by the sequence of observations $\boldsymbol{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N)$. The expected log-likelihood of the data given a network structure can be computed in terms of conditional entropy:*

$$\mathbf{E}_{\boldsymbol{Z}}[\ell(\hat{\theta}_G; \boldsymbol{Z})] = N\, H(\boldsymbol{X}_n \mid \{\boldsymbol{Y}_n^{i,(\kappa^i)}\}) - N \sum_{i=1}^{M} H(Y_{n+1}^i \mid \boldsymbol{Y}_n^{i,(\kappa^i)}, \{\boldsymbol{Y}_n^{ij,(\kappa^{ij})}\}_j) \tag{5.12}$$

Proof. Substituting (5.8) into (5.4) gives the expected log-likelihood $\mathbf{E}[\ell(\hat{\theta}_G; D)]$ as

$$\begin{aligned}N \sum_{i=1}^{M} \sum_{y_{n+1}^i} \sum_{\boldsymbol{y}_n^{i,(\kappa^i)}} \sum_{(\boldsymbol{y}_n^{ij,(\kappa^{ij})})_j} p_{\hat{B}}(y_{n+1}^i, \boldsymbol{y}_n^{i,(\kappa^i)}, (\boldsymbol{y}_n^{ij,(\kappa^{ij})})_j) \log p_{\hat{B}}(y_{n+1}^i \mid \boldsymbol{y}_n^{i,(\kappa^i)}, (\boldsymbol{y}_n^{ij,(\kappa^{ij})})_j) \\ - N \sum_{\boldsymbol{x}_n} \sum_{(\boldsymbol{y}_n^{i,(\kappa^i)})} p_{\hat{K}}(\boldsymbol{x}_n, (\boldsymbol{y}_n^{i,(\kappa^i)})) \log p_{\hat{K}}(\boldsymbol{x}_n \mid (\boldsymbol{y}_n^{i,(\kappa^i)})).\end{aligned} \tag{5.13}$$

In (5.13) we have removed arguments of the joint distributions that will be nullified when multiplied with the CPD. Expressing (5.13) in terms of conditional entropy (3.11), we arrive at (5.12). □

We now look at the log-likelihood in the context of information transfer. First, rearranging the terms of collective transfer entropy (4.4) we can rewrite the log-likelihood function (5.12), leading to the following result.

Corollary 5.1.1. *The log-likelihood function for the POSGDS (5.12) decomposes as follows:*

$$\mathbf{E}[\ell(\hat{\theta}_G; \boldsymbol{Z})] = N\, H(\boldsymbol{X}_n \mid (\boldsymbol{Y}_n^{i,(\kappa^i)})) - N \sum_{i=1}^{M} H(Y_{n+1}^i \mid \boldsymbol{Y}_n^{i,(\kappa^i)}) + N \sum_{i=1}^{M} T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}. \quad (5.14)$$

The first two terms in (5.14) do not depend on the proposed graph structure, and thus maximising log-likelihood is equivalent to maximising collective transfer entropy. This becomes clear when we consider the *log-likelihood ratio*. This ratio quantifies the gain in likelihood by modelling the data $\boldsymbol{Z}$ by a candidate network B instead of the empty network $\hat{B}_0$, i.e.,

$$\ell(\hat{\theta}_G; \boldsymbol{Z}) - \ell(\hat{\theta}_{G_0}; \boldsymbol{Z}) \propto \log \frac{p_{\hat{B}}(\boldsymbol{Z})}{p_{\hat{B}_0}(\boldsymbol{Z})}. \quad (5.15)$$

Recall from Sec. 3.6.1 that the null DAG G_0 for log-likelihood testing is one with no parents for all vertices $\forall i, \mathrm{Pa}(V^i) = \emptyset$. Substituting this definition into (5.12) (or, alternatively (5.14)) gives the following result.

Corollary 5.1.2. *The ratio of the log-likelihood (5.12) of a candidate DAG G to the empty network G_0 can be expressed as*

$$\mathbf{E}_{\boldsymbol{Z}}[\ell(\hat{\theta}_G; \boldsymbol{Z}) - \ell(\hat{\theta}_{G_0}; \boldsymbol{Z})] = N \sum_{i=1}^{M} T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}. \quad (5.16)$$

5.4 RECONSTRUCTION THEOREMS FOR KL DIVERGENCE

We now continue this analysis by considering the KL divergence of the candidate graph G from the complete graph K_M. That is, $p_{\hat{K}}$ is the joint distribution yielded by assuming no factorisation (the complete graph K_M). The distribution is expressed as:

$$p_{\hat{K}}(\boldsymbol{z}_{n+1} \mid \boldsymbol{z}_n^-) = p_{\hat{K}}(\boldsymbol{x}_{n+1} \mid \boldsymbol{x}_n)\, p_{\hat{K}}(\boldsymbol{y}_{n+1} \mid \boldsymbol{x}_{n+1}). \quad (5.17)$$

Substituting the factorisations of the candidate graph (5.3) and the complete graph (5.17) into (3.37), we get

$$D_{\mathrm{KL}}[p_{\hat{K}} \parallel p_{\hat{B}}] = \mathbf{E}_{\boldsymbol{Z}}\left[\log \frac{p_{\hat{K}}(\boldsymbol{X}_{n+1} \mid \boldsymbol{X}_n)\, p_{\hat{K}}(\boldsymbol{Y}_{n+1} \mid \boldsymbol{X}_{n+1})}{\prod_{i=1}^{M} p_{\hat{B}}(\boldsymbol{X}_{n+1}^i \mid \boldsymbol{X}_n^i, \mathrm{Pa}(X_n^i))\, p_{\hat{B}}(Y_{n+1}^i \mid \boldsymbol{X}_{n+1}^i)}\right]. \quad (5.18)$$

However, as in the log-likelihood case, the numerator in Eq. (5.18) still comprises maximum likelihood distributions with unobserved (latent) states $\boldsymbol{x}_n$. In order to compute the distributions in (5.18), we leverage the results from Sec. 5.3 to reformulate the factorised distribution (denominator), and then employ the Delay Embedding Theorem for Multivariate Observation Functions [68] for the joint distribution (numerator).

We present a method for computing the joint distribution (numerator) in Lemma 5.4. As an intermediate step, Lemma 5.3 restates part of the delay embedding theorem in [68] in terms of subsystems of a POSGDS and establishes existence of a map E for predicting future observations from a history of observations.

Lemma 5.3. *Consider a diffeomorphism $f : \mathcal{M} \to \mathcal{M}$ on a d-dimensional manifold $\mathcal{M}$, where the multivariate state $\boldsymbol{x}_n$ consists of M subsystem states $(\boldsymbol{x}_n^1, \ldots, \boldsymbol{x}_n^M)$. Each subsystem state $\boldsymbol{x}_n^i$ is confined to a submanifold $\mathcal{M}^i \subseteq \mathcal{M}$ of dimension $d^i \leq d$, where $\sum_i d^i = d$. The multivariate observation is given, for some map E, by $\boldsymbol{y}_{n+1} = E(\Phi(\boldsymbol{x}_n))$.*

Proof. The proof restates part of the proof of Theorem 2 of Deyle and Sugihara [68] in terms of subsystems.

Recall from (5.9) and (5.10), we have the global map

$$\Phi(\boldsymbol{x}_n) = (y_n^{1,(\kappa^1)}, \ldots, y_n^{m,(\kappa^M)}).$$

Now, since Φ is an embedding, it follows that the map $F = \Phi \circ f \circ \Phi^{-1}$ is well defined and a diffeomorphism between two observation sequences $F : \mathbb{R}^{2d+1} \to \mathbb{R}^{2d+1}$, i.e.,

$$\begin{aligned}\Phi(\boldsymbol{x}_{n+1}) &= \Phi(f(\boldsymbol{x}_n)) \\ &= \Phi\left(f\left(\Phi^{-1} \circ \Phi(\boldsymbol{x}_n)\right)\right) \\ &= F(\Phi(\boldsymbol{x}_n)).\end{aligned}$$

The last $2d+1$ components of F are trivial, i.e., the set $\Phi(\boldsymbol{x}_n)$ is observed; denote the first M components by $E : \Phi_{f,\psi} \to \mathbb{R}^M$, then we have $\boldsymbol{y}_{n+1} = E(\Phi(\boldsymbol{x}_n))$. □

We now use the result of Lemma 5.3 to obtain a computable form of the KL divergence.

Lemma 5.4. *Consider a discrete-time multivariate dynamical system with generic (f, ψ) modelled as a directed and acyclic POSGDS $(G, \boldsymbol{x}_n, \boldsymbol{y}_n, \{f^i\}, \{\psi^i\})$ with M subsystems. The KL divergence of a candidate graph G from the complete graph K_M can be computed from tractable probability distributions:*

$$D_{\mathrm{KL}}\left[p_{\hat{K}} \parallel p_{\hat{B}}\right] = \mathbf{E}_{\boldsymbol{Y}}\left[\log \frac{p_{\hat{K}}(\boldsymbol{Y}_{n+1} \mid \Phi(\boldsymbol{X}_n))}{\prod_{i=1}^M p_{\hat{B}}(Y_{n+1}^i \mid \Phi^i(\boldsymbol{X}_n^i), \Phi(\mathrm{Pa}(\boldsymbol{X}_n^i)))}\right]. \tag{5.19}$$

Proof. From Lemma 5.2, we can substitute (5.8) into (5.18), and express the KL divergence $D_{\mathrm{KL}}\left[p_{\hat{K}} \parallel p_{\hat{B}}\right]$ as

$$D_{\mathrm{KL}}\left[p_{\hat{K}} \parallel p_{\hat{B}}\right] = \mathbf{E}_{\mathbf{Z}}\left[\log \frac{p_{\hat{K}}\left(\boldsymbol{X}_{n+1} \mid \boldsymbol{X}_n\right) p_{\hat{K}}\left(\boldsymbol{Y}_{n+1} \mid \boldsymbol{X}_{n+1}\right) p_{\hat{K}}(\boldsymbol{X}_n \mid \Phi(\boldsymbol{X}_n))}{\prod_{i=1}^{M} p_{\hat{B}}(Y_{n+1}^i \mid \Phi(\boldsymbol{X}_n^i), \Phi(\mathrm{Pa}(\boldsymbol{X}_n^i)))}\right]. \tag{5.20}$$

Recall fom Lemma 5.3 that global equations for the entire system state $\boldsymbol{x}_n$ and observation $\boldsymbol{y}_n$ are

$$\boldsymbol{x}_{n+1} = f(\boldsymbol{x}_n) + \boldsymbol{v}_f = f\left(\Phi^{-1} \circ \Phi(\boldsymbol{x}_n)\right) + \boldsymbol{v}_f, \tag{5.21}$$

$$\boldsymbol{y}_{n+1} = \psi(\boldsymbol{x}_{n+1}) + \boldsymbol{v}_\psi = E(\Phi(\boldsymbol{x}_n)) + \boldsymbol{v}_\psi. \tag{5.22}$$

Given the assumption of i.i.d noise on the function f, from (5.21), we express the probability of the dynamics $\boldsymbol{x}_{n+1}$, given by the embedding, as

$$\begin{aligned} p_{\hat{K}}\left(\boldsymbol{x}_{n+1} \mid \Phi(\boldsymbol{x}_n)\right) &= p_{\hat{K}}\left(\boldsymbol{X}_{n+1} = f\left(\Phi^{-1} \circ \Phi(\boldsymbol{x}_n)\right) \mid \Phi(\boldsymbol{x}_n)\right) \\ &= p_{\hat{K}}\left(\boldsymbol{X}_n = \Phi^{-1} \circ \Phi(\boldsymbol{x}_n) \mid \Phi(\boldsymbol{x}_n)\right) \\ &\qquad \times p_{\hat{K}}\left(\boldsymbol{X}_{n+1} = f(\boldsymbol{x}_n) \mid \boldsymbol{x}_n\right). \end{aligned} \tag{5.23}$$

By assumption, the observation noise is i.i.d or dependent only on the state $\boldsymbol{x}_{n+1}$, and thus the probability of observing $\boldsymbol{y}_{n+1}$, from (5.22) is

$$\begin{aligned} p_{\hat{K}}\left(\boldsymbol{y}_{n+1} \mid \Phi(\boldsymbol{x}_n)\right) &= p_{\hat{K}}\left(\boldsymbol{Y}_{n+1} = E(\Phi(\boldsymbol{x}_n)) \mid \Phi(\boldsymbol{x}_n)\right) \\ &= p_{\hat{K}}\left(\boldsymbol{X}_{n+1} = f\left(\Phi^{-1} \circ \Phi(\boldsymbol{x}_n)\right) \mid \Phi(\boldsymbol{x}_n)\right) \\ &\qquad \times p_{\hat{K}}\left(\boldsymbol{Y}_{n+1} = \psi(\boldsymbol{x}_{n+1}) \mid \boldsymbol{x}_{n+1}\right). \end{aligned} \tag{5.24}$$

By (5.23) and (5.24), we have that

$$p_{\hat{K}}(\boldsymbol{x}_{n+1} \mid \boldsymbol{x}_n)\, p_{\hat{K}}(\boldsymbol{y}_{n+1} \mid \boldsymbol{x}_{n+1}) = \frac{p_{\hat{K}}(\boldsymbol{y}_{n+1} \mid \Phi(\boldsymbol{x}_n))}{p_{\hat{K}}(\boldsymbol{x}_n \mid \Phi(\boldsymbol{x}_n))} \tag{5.25}$$

Finally, substituting (5.25) into (5.20) yields the statement of the theorem. □

Given that all variables in (5.19) are observed, it is now straightforward to compute KL divergence; however, as we will see, it is more convenient to express (5.19) as a function of known information-theoretic measures.

5.4.1 *Information-theoretic interpretation*

The main theorem of this chapter, presented below, states KL divergence in terms of transfer entropy and stochastic interaction.

Theorem 5.2. *Consider a discrete-time multivariate dynamical system with generic* (f, ψ) *represented as a directed and acyclic* POSGDS $(G, \boldsymbol{x}_n, \boldsymbol{y}_n, \{f^i\}, \{\psi^i\})$ *with* M *subsystems. The* KL *divergence* $D_{\mathrm{KL}}[p_{\hat{K}} \parallel p_{\hat{B}}]$ *of a candidate graph* G *from the dataset* $\boldsymbol{z}$ *can be expressed as the difference between stochastic interaction* (3.28) *and collective transfer entropy* (3.22)*, i.e.,*

$$D_{\mathrm{KL}}[p_{\hat{K}} \parallel p_{\hat{B}}] = S_{\boldsymbol{Y}} - \sum_{i=1}^{M} T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}. \tag{5.26}$$

Proof. We can reformulate the KL divergence in (5.19) as

$$\begin{aligned} D_{\mathrm{KL}}[p_{\hat{K}} \parallel p_{\hat{B}}] &= \mathbf{E}_{\boldsymbol{Y}}\left[\log\left(p_{\hat{K}}(\boldsymbol{Y}_{n+1} \mid \Phi(\boldsymbol{X}_n))\right)\right] \\ &\quad - \mathbf{E}_{\boldsymbol{Y}}\left[\log\left(\prod_{i=1}^{M} p_{\hat{B}}(Y^i_{n+1} \mid \Phi(\boldsymbol{X}^i_n), \Phi(\mathrm{Pa}(\boldsymbol{X}^i_n)))\right)\right] \\ &= -H(\boldsymbol{Y}_{n+1} \mid \{\boldsymbol{Y}_n^{(\kappa^i)}\}) + \sum_{i=1}^{M} H(Y^i_{n+1} \mid \boldsymbol{Y}_n^{i,(\kappa^i)}, \{\boldsymbol{Y}_n^{ij,(\kappa^{ij})}\}_j) \\ &= -H(\boldsymbol{Y}_{n+1} \mid \{\boldsymbol{Y}_n^{(\kappa^i)}\}) + \sum_{i=1}^{M} H(Y^i_{n+1} \mid \boldsymbol{Y}_n^{i,(\kappa^i)}) \\ &\quad + \sum_{i=1}^{M}\left(H(Y^i_{n+1} \mid \boldsymbol{Y}_n^{i,(\kappa^i)}, \{\boldsymbol{Y}_n^{ij,(\kappa^{ij})}\}_j) - H(Y^i_{n+1} \mid \boldsymbol{Y}_n^{i,(\kappa^i)})\right). \end{aligned} \tag{5.27}$$

Substituting in the definitions of transfer entropy (3.22) and stochastic interaction (3.28) completes the proof. □

We conclude this section by presenting the following corollary showing that, when we assume a maximum or fixed embedding dimension κ^i and time delay τ^i, it suffices to maximise the collective transfer entropy alone in order to minimise KL divergence for a POSGDS.

Corollary 5.2.1. *Fix an embedding dimension* κ^i *and time delay* τ^i *for each subsystem* $V^i \in \mathcal{V}$*. Then, the graph* G *that minimises the* KL *divergence* $D_{\mathrm{KL}}[p_{\hat{K}} \parallel p_{\hat{B}}]$ *is equivalent to the graph that maximises transfer entropy, i.e.,*

$$\arg\min_{G \in \mathcal{G}} D_{\mathrm{KL}}[p_{\hat{K}} \parallel p_{\hat{B}}] = \arg\max_{G \in \mathcal{G}} \sum_{i=1}^{M} T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}. \tag{5.28}$$

Proof. The first term of (5.26) is constant, given a constant vertex set $\mathcal{V}$, time delay τ and embedding dimension κ and is thus unaffected by the parent set $\mathrm{Pa}(V^i)$ of a variable. As a result, $S_{\boldsymbol{Y}}$ does not depend on the graph G being considered and therefore we only need to consider transfer entropy when optimising KL divergence (5.26). □

5.5 APPLICATION TO STRUCTURE LEARNING

We now employ the results above in selecting a POSGDS that best fits data generated by a multivariate dynamical system. The most natural way to find an optimal model based on Theorem 5.2 is minimise KL divergence. Here we assume constant embedding parameters and use Corollary 5.2.1 to present the *transfer entropy score* and discuss some attributes of this score. We then use this scoring function as a subroutine for learning the structure of coupled Lorenz and Rössler attractors.

As mentioned above, Corollary 5.2.1 is, in practice, equivalent to the maximum log-likelihood and log-likelihood ratio approaches. However, the statement only holds for constant embedding parameters. In the general case, where these parameters are unknown, one requires Theorem 5.2 to perform structure learning. Given this result, we can now confidently derive scoring functions from Corollary 5.2.1.

From either Corollary 5.2.1 or 5.1.2, the log-likelihood scoring function can be defined as

$$g_{\text{TE}}(B : \mathbf{Z}) = \sum_{i=1}^{M} T_{\{\mathbf{Y}^{ij}\}_j \to \mathbf{Y}^i}. \tag{5.29}$$

Given parameterised probability distributions, this score is insufficient, since the sum of transfer entropy in (5.29) is non-decreasing when including more parents in the graph [146]. Thus, we use statistical significance tests in our scoring functions to mitigate this issue.

5.5.1 *Model complexity penalty functions*

We can obtain penalisations based on the AIC and BIC loss functions if the observations are assumed to follow distributions from the exponential family of functions.

Theorem 5.3. *The information criterion* (3.43) *for synchronous GDS can be computed as:*

$$\begin{aligned} g_{\text{IC}}(B : \mathbf{Z}) = -N \sum_{i=1}^{M} H(Y_{n+1}^i \mid \mathbf{Y}_n^{i,(\kappa^i)}, (\mathbf{Y}_n^{ij,(\kappa^{ij})})_j) \\ - c(N) \sum_{i=1}^{M} \Big(|Y_n^i|^{\kappa^i} \, (|Y_n^i| - 1) \prod_{V^p \in \text{Pa}(V^i)} |Y_n^p|^{\kappa^p} \Big). \end{aligned} \tag{5.30}$$

Proof. The distributions for the first term in (5.13) do not depend on the parents of a subsystem and thus are independent of the graph G being considered. Therefore, we have the following equation for maximimum log-likelihood:

$$\max_G \mathbf{E}[\ell(\hat{\theta}_G; \mathbf{Z})] = \mathcal{O}(N) - \min_G N \sum_{i=1}^{M} H(Y_{n+1}^i \mid \mathbf{Y}_n^{i,(\kappa^i)}, (\mathbf{Y}_n^{ij,(\kappa^{ij})})_j). \tag{5.31}$$

We can now compute the number of parameters needed to specify the model as [81]

$$d(G) = \sum_{i=1}^{M} \left(|Y_n^i|^{\kappa^i} \left(|Y_n^i| - 1 \right) \prod_{V^p \in \mathrm{Pa}(V^i)} |Y_n^p|^{\kappa^p} \right), \tag{5.32}$$

where $|\cdot|$ specifies the number of parameters needed to encode the distribution.

Since we are searching for the graph $G^* = \max_G g(B : \boldsymbol{Z})$, holding N constant, we can substitute (5.31) and (5.32) into (3.43) and ignore the constant term $\mathcal{O}(N)$ in (5.31). □

In practice, we do not necessarily have observations from the exponential family. In this case, it is often convenient to use methods based on surrogate populations as we discuss below.

5.5.2 *Independence test penalty functions*

Building on the maximum likelihood score (5.29), we propose to use independence tests to define two new scores of practical value. Here, we draw on the result of Campos [43], who derived a scoring function for BN structure learning based on conditional mutual information and statistical significance tests. The central idea is to use collective transfer entropy $T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}$ to measure the degree of interaction between each subsystem V^i and its parent subsystems $\mathrm{Pa}(V^i)$, but also to penalise this term with a value based on significance testing. As with the MIT score, this gives a principled way to re-scale the transfer entropy when including more edges in the graph.

To develop our scores, we form a *null hypothesis* H_0 that there is no interaction $T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}$, and then compute a test statistic to penalise the measured transfer entropy. To compute the test statistic, it is necessary to consider the measurement distribution in the case where the hypothesis is true. Unfortunately, this distribution is only analytically tractable in the case of discrete and linear-Gaussian systems, where $2NT_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}$ is known to asymptotically approach the χ^2-distribution [19]. Since this distribution is a function of the parents of $\boldsymbol{Y}^i$, we let it be described by the function $\chi^2(\{l^{ij}\}_j)$. Now, given this distribution, we can fix some *confidence level* α and determine the value $\chi_{\alpha,\{l^{ij}\}_j}$ such that $p(\chi^2(\{l^{ij}\}_j) \leq \chi_{\alpha,\{l^{ij}\}_j})$. This represents a conditional independence test: if $2NT_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i} \leq \chi_{\alpha,\{l^{ij}\}_j}$, then we accept the hypothesis of conditional independence between $\boldsymbol{Y}^i$ and $\{\boldsymbol{Y}^{ij}\}_j$; otherwise, we reject it. We express this idea as the TEA score:

$$g_{\mathrm{TEA}}(B : \boldsymbol{Z}) = \sum_{i=1}^{M} \left(2NT_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i} - \chi_{\alpha,\{l^{ij}\}_j} \right). \tag{5.33}$$

In general we only have access to *continuous* nonlinear measurements of dynamical systems, and so are limited by the discrete or linear-Gaussian assumption. We can, however, use *surrogate* measurements $T_{\{\boldsymbol{Y}^{ij}\}_j^s \to \boldsymbol{Y}^i}$ to empirically compute the distribution under the assumption of H_0 [148]. This same technique has been used by Lizier et al. [146] to derive a greedy structure learning algorithm for effective network analysis. Here, $\{\boldsymbol{Y}^{ij}\}_j^s$ are surrogate sets of variables for $\{\boldsymbol{Y}^{ij}\}_j$, which have the same statistical properties as $\{\boldsymbol{Y}^{ij}\}_j$, but the correlation between $\{\boldsymbol{Y}^{ij}\}_j^s$ and $\boldsymbol{Y}^i$ is removed. Let the distribution of these surrogate measurements be represented by some general function $T(s^i)$ where, for the discrete and linear-Gaussian systems, we could compute $T(s^i)$ analytically as an independent set of χ^2-distributions $\chi^2(\{I^{ij}\}_j)$. When no analytic distribution is known, we use a resampling method (i.e., permutation or bootstrapping), creating a large number of surrogate time-series pairs $\{\{\boldsymbol{Y}^{ij}\}_j^s, \boldsymbol{Y}^i\}$ by shuffling (for permutations, or redrawing for bootstrapping) the samples of $\boldsymbol{Y}^i$ and computing a population of $T_{\{\boldsymbol{Y}^{ij}\}_j^s \to \boldsymbol{Y}^i}$. As with the TEA score, we fix some confidence level α and determine the value T_{α,s^i}, such that $p(T(s^i) \leq T_{\alpha,s^i}) = \alpha$. This results in the TEE scoring function as

$$g_{\text{TEE}}(B : \boldsymbol{Z}) = \sum_{i=1}^{M} \left(T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i} - T_{\alpha,s^i} \right). \tag{5.34}$$

We can obtain the value T_{α,s^i} by 1) drawing S samples $T_{\{\boldsymbol{Y}^{ij}\}_j^s \to \boldsymbol{Y}^i}$ from the distribution $T(s^i)$ (by permutation or bootstrapping), 2) fixing $\alpha \in \{0, 1/S, 2/S, \ldots, 1\}$, then (3) taking T_{α,s^i} such that

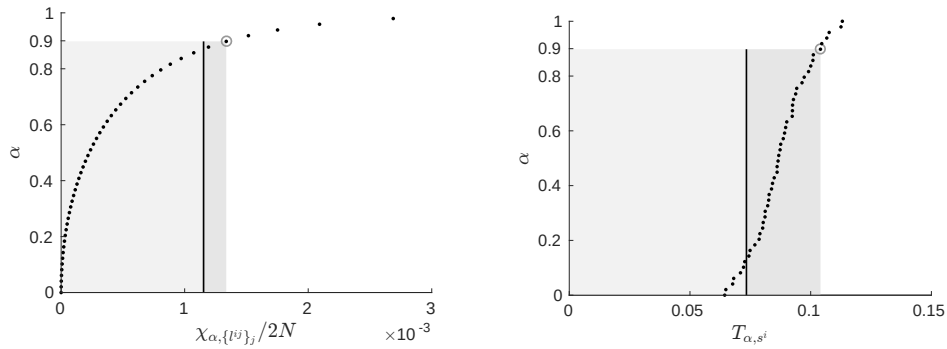
$$\alpha = \frac{1}{S} \sum_{T_{\{\boldsymbol{Y}^{ij}\}_j \to \boldsymbol{Y}^i}} \mathbb{1}_{T_{\{\boldsymbol{Y}^{ij}\}_j^s \to \boldsymbol{Y}^i} \leq T_{\alpha,s^i}}.$$

We can alternatively limit the number of surrogates S to $\lceil \alpha/(1-\alpha) \rceil$ and take the maximum as T_{α,s^i} [114], however taking a larger number of surrogates will improve the validity of the distribution $T(s^i)$.

Both the analytical (TEA) and empirical (TEE) scoring functions are illustrated in Fig. 5.3. Note that the approach of significance testing is functionally equivalent to considering the log-likelihood ratio, where, as stated, nested log-likelihoods (and thus transfer entropy) follows the above χ^2-distribution [19].

5.5.3 *Implementation details and algorithm analysis*

The two main implementation challenges that arise when performing structure learning are: 1) computing the score for every candidate network and 2) obtaining a sufficient number of samples to recover the network. The main contributions of this chapter are theoretical justifications for measures already in use and, fortunately,



(a) TEA penalty (b) TEE penalty

Figure 5.3: Distributions of the 5.3a TEA penalty function (5.33) and the 5.3b TEE penalty function (5.33). Both distributions were generated by observing the outcome of 1000 samples from two Gaussian variables with a correlation of 0.05. The figures illustrate: the distribution as a set of 100 sampled points (black dots); the area considered independent (grey regions); the measured transfer entropy (black line); and the difference between measurement and penalty term (dark grey region). Both tests use a value of $\alpha = 0.9$ (a p-value of 0.1). The distribution in Fig. 5.3a was estimated by assuming variables were linearly-coupled Gaussians, and the distribution in Fig. 5.3b was computed via a kernal box method (computed by the JIDT, see [148] for details).

algorithmic performance has already been addressed extensively using various heuristics. Here, we present an exact, exhaustive implementation for the purpose of validating our theoretical contributions.

First, for computing collective transfer entropy for the score (5.34), we require CPDs to be estimated from data. Given these CPDs, collective transfer entropy (3.22) decomposes as a sum of P conditional transfer entropy terms, where $P = |\{\boldsymbol{Y}^{ij}\}_j|$ is the size of the parent set. Since most observations of dynamical systems are expected to be continuous, we employ a non-parametric, nearest-neighbour based approach to density estimation called the Kraskov-Stögbauer-Grassberger (KSG) estimator [131] (the same estimator that was used in Chapter 4). For any arbitrary decomposition of collective transfer entropy (i.e., any ordering of the parent set), this density estimation can be computed in time $O(\kappa(P+1)KN^{\kappa(p+1)}\log(N))$, where K is the number of nearest neighbours for each observation in a dataset of size N, and κ is the embedding dimension [148]. We upper bound this as $O(\kappa MKN^{\kappa M}\log(N))$ since the maximum P is $M-1$.

Now, the above density estimation was described for an arbitrary ordering of the parent set. In the case of parametric (discrete or linear-Gaussian) density estimation, every permutation of the parent set yields equivalent results, with potentially different $\chi_{\alpha,\{I^{ij}\}_j}$ values for each permutation [43]; however, this is not the case for non-parametric density estimation techniques, e.g., the KSG estimator. Hence, as

a conservative estimate of the score, we compute all $P!$ permutations of the parent set and take the minimum collective transfer entropy. In order to obtain the surrogate distribution, we require S uncorrelated samples of the density. Since the surrogate distributions decompose in a similar manner, the score for a candidate network can be computed in time $O(S \cdot M! \cdot \kappa M K N^{\kappa M} \log(N))$, where, again, we have upper bounded $P!$ as $M!$.

Using this approach, we can now compute the score (5.34), and thus the optimal graph G^* can be found using any search procedure over DAGs. Exhaustive search, where all DAGs are enumerated, is typically intractable because the search space is super-exponential in the number of variables (about $2^{O(M^2)}$), and so heuristics are often applied for efficiency. We restrict our attention to a relatively small network (a maximum of $M = 4$ nodes) and thus we are able employ the dynamic programming approach of Silander and Myllymaki [208] to search through the space of all DAGs efficiently. This approach requires first computing the scores for all local parent sets, i.e., 2^M scores. Once each score is calculated, the dynamic programming algorithm runs in time $o(M \cdot 2^{M-1})$ and the entire search procedure run in time $O(M \cdot 2^{M-1} + 2^M \cdot S \cdot M! \cdot \kappa M K N^{\kappa M} \log(N))$. As a consequence, the time complexity of the exhaustive algorithm is dominated by computing the 2^M scores and, in smaller networks, most of the time is spent on density estimation for surrogate distributions.

Finally, the problem of inferring optimal embedding parameters is well studied in the literature. In our experimental evaluation, we set the embedding dimension to the maximum, i.e., $\kappa = 2d + 1$, where d is the dimensionality of the entire latent state space (e.g., if $M = 3$ and $d^i = 3$ for each subsystem, then $\kappa = 2\sum_i d^i + 1 = 19$). However, determining these parameters would give more insight into the system and reduce the number of samples required for inference. There are numerous criteria for optimising these parameters [191]; most notably, the work of Small et al. [213] suggests an information-theoretic approach that could be integrated into the scoring function (5.34) to search over the embedding parameters and DAG space simultaneously.

5.6 EXPERIMENTAL VALIDATION

The dynamics (5.1) and observation (5.2) maps can be obtained by either differential equations, discrete-time maps, or real-world measurements. To validate our approach, we use the toy example of distributed flows, whereby the dynamics of each node are given by either the Lorenz [156] or the Rössler system of ODEs [198]. The discrete-time measurements are obtained by integrating these ODEs over con-

stant intervals. In this section, we formally introduce this model, study the effect of changing the parameters of a coupled Lorenz-Rössler system, and finally apply our scoring function to learn the structure of up to four coupled Lorenz attractors with arbitrary graph topology. To compute the scores, we use the JIDT [148], which includes both the KSG estimator and methods for generating the surrogate distributions.

5.6.1 *Distributed Lorenz and Rössler attractors*

The Lorenz attractor exhibits chaotic solutions for certain parameter values and has been used to describe numerous phenomena of practical interest [65, 98, 156]. Each Lorenz system comprises three components ($d^i = 3$), which we denote $\boldsymbol{x} = \langle u, v, w \rangle$; the state dynamics are given by:

$$\dot{\boldsymbol{x}} = h(\boldsymbol{x}) = \begin{cases} \dot{u} = \zeta(v - u) \\ \dot{v} = u(\rho - w) - v \\ \dot{w} = uv - \beta w, \end{cases} \tag{5.35}$$

with free parameters $\{\zeta, \rho, \beta\}$. Similarly, the Rössler attractor has state dynamics given by:

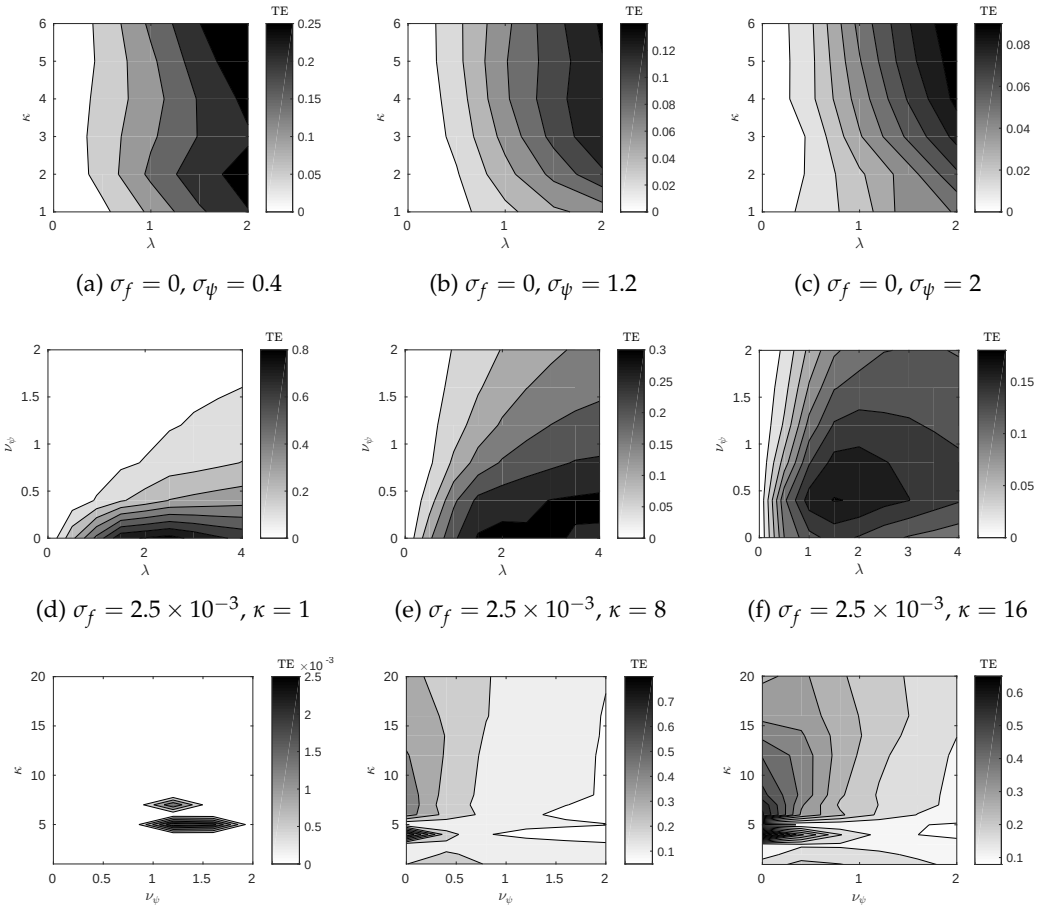
$$\dot{\boldsymbol{x}} = h(\boldsymbol{x}) = \begin{cases} \dot{u} = -v - w \\ \dot{v} = w + av \\ \dot{w} = b + w(u - c), \end{cases} \tag{5.36}$$

with free parameters $\{a, b, c\}$ [198].

In the distributed case, the components of each state vector $\boldsymbol{x}_t^i$ are also driven by components of another subsystem. A number of different schemes have been proposed for coupling these variables, e.g., using the product [99, 124] and the difference [85, 199] of components. Our model uses the latter approach of linear differencing between one or more subsystem variables to couple the network. Let λ denote the coupling strength, C denote a 3-dimensional vector of binary values, and A denote an adjacency (coupling) matrix (i.e., an $M \times M$ matrix of zeros with $A_{ij} = 1$ iff $V^i \in \mathrm{Pa}(V^j)$). Then, the state equations for M spatially distributed systems can be expressed as

$$\dot{\boldsymbol{x}}_t^i = h^i(\boldsymbol{x}_t^i) + \boldsymbol{\nu}_f + \lambda C \sum_{j=1}^{M} A_{ij}(\boldsymbol{x}_t^j - \boldsymbol{x}_t^i), \tag{5.37}$$

where $h^i(\cdot)$ represents the ith chaotic attractor and ν_f is additive noise. In our simulations, we use $\lambda = 2$, $C = (1, 0, 0)$ (each subsystem is coupled via variable u),



(a) $\sigma_f = 0$, $\sigma_\psi = 0.4$ (b) $\sigma_f = 0$, $\sigma_\psi = 1.2$ (c) $\sigma_f = 0$, $\sigma_\psi = 2$

(d) $\sigma_f = 2.5 \times 10^{-3}$, $\kappa = 1$ (e) $\sigma_f = 2.5 \times 10^{-3}$, $\kappa = 8$ (f) $\sigma_f = 2.5 \times 10^{-3}$, $\kappa = 16$

(g) $\sigma_f = 2.5 \times 10^{-3}$, $\lambda = 0$ (h) $\sigma_f = 2.5 \times 10^{-3}$, $\lambda = 2$ (i) $\sigma_f = 2.5 \times 10^{-3}$, $\lambda = 4$

Figure 5.4: Transfer entropy as a function of the parameters of a coupled Lorenz-Rössler system. These components are: coupling strength λ and embedding dimension κ in the top row (Figs. 5.4a-5.4c); coupling strength λ and observation noise σ_ψ in the middle row (Figs. 5.4d-5.4f); and observation noise σ_ψ and embedding dimension κ in the bottom row (Figs. 5.4g-5.4i).

and the adjacency matrices shown in Fig. 5.5. In our experiments we use common parameters for both attractors, i.e., $\zeta = 10$, $\beta = 8/3$, $\rho = 28$ for Lorenz ODEs; and $a = 0.1$, $b = 0.1$, $c = 14$ for Rössler ODEs. For the observation y_t^i it is common to use one component of the state as the read-out function [221, 222, 227]; we therefore let $y_t^i = u_t^i + \nu_\psi$. The noise terms are normally distributed with $\boldsymbol{\nu}_f \sim \mathcal{N}(0, \sigma_f)$ and $\nu_\psi \sim \mathcal{N}(0, \sigma_\psi)$. Figure 5.1 illustrates example trajectories of Lorenz-Lorenz attractors coupled via this model.

5.6.2 *Coupled Lorenz-Rössler system*

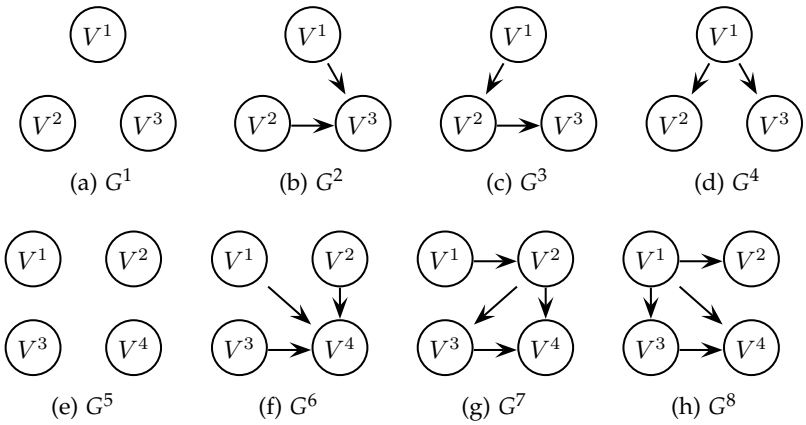
In order to characterise the effect of coupling on our score, we begin our evaluation by measuring the transfer entropy of a coupled Lorenz-Rössler attractor. In this setup, $M = 2$, $\mathrm{Pa}(V^1) = \emptyset$, and $\mathrm{Pa}(V^2) = V^1$, $h^1(x)$ was given by (5.35), and $h^2(x)$ was given by (5.36). The transfer entropy was computed with a finite sample size of $N = 100,000$.

Figure 5.4 shows the transfer entropy as a function of numerous parameters. In particular, the figure illustrates the effect of varying the coupling strength λ, embedding dimension κ, dynamics noise σ_f, and observation noise σ_ψ. As expected, increasing λ, or reducing either noise σ, increases the transfer entropy. The embedding dimension, however, increases to a set point, remains approximately constant, then decreases. The κ-value above which transfer entropy remains constant illustrates the embedding dimension at which the dynamics are reconstructed; the decrease in transfer entropy after this point, however, is likely due to the finite sample size used for density estimation.

There are two interesting features in Fig. 5.4 due to the dynamical systems studied. First, in the bottom row (Fig. 5.4g-5.4i), there is a bifurcation around $\kappa = 6$. The theoretical embedding dimension for this system is $\kappa = 2(d^1 + d^2) + 1 = 7$, and, in this case, for $\kappa < 6$ the embedding does not suffice to reconstruct the dynamics. Second, in Fig. 5.4i, the transfer entropy decreases after about $\lambda = 2$. This appears to be the case of synchrony due to strong coupling, where the dynamics of the forced variable become subordinate to the forcing [227], thus reducing the information transferred between the two subsystems.

5.6.3 *Network of Lorenz attractors*

In this section we evaluate the score (5.29) in learning the structure of distributed dynamical systems. We study systems of three and four nodes of coupled Lorenz subsystems with arbitrary topologies. Unfortunately, significantly higher number



(a) G^1 (b) G^2 (c) G^3 (d) G^4

(e) G^5 (f) G^6 (g) G^7 (h) G^8

Figure 5.5: The network topologies used for structure learning. The top row (Figs. 5.5a-5.5d) are four arbitrary networks with three nodes ($M = 3$) and the bottom row (Figs. 5.5e- 5.5h) are four arbitrary networks with four nodes ($M = 4$).

of nodes become computationally expensive due to an increased embedding dimension κ, number of data points N, and number of permutations required to calculate the collective transfer entropy. To evaluate the performance of the score (5.29), the coupling strength $\lambda = 2$ and dynamics noise $\sigma_f = 0.01$ are constant whereas the observation noise σ_ψ and the number of observations taken N are varied. We selected the theoretical maximum embedding dimension $\kappa = 2d + 1$ and $\tau = 1$ as is common given discrete-time measurements [114]. It should be noted that from the results of Sec. 5.6.2 that transfer entropy is sensitive to the numerous parameters used to generate the data, and thus depending on the scenario, a significant sample size can be required for recovering the underlying graph structure. We do not make an effort to reduce this sample size and instead show the effect of using a different number of samples on the accuracy of the structure learning procedure.

In order to evaluate the scoring function, we compute the recall (R, or true positive rate), fallout (F, or false positive rate), and precision (P, or positive predictive value) of the recovered graph. Let TP denote the number of true positives (correct edges); TN denote the number of true negatives (correctly rejected edges); FP denote the number of false positives (incorrect edges); and FN denote the number of false negatives (incorrectly rejected edges). Then, $\mathrm{R} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$, $\mathrm{F} = \mathrm{FP}/(\mathrm{FP} + \mathrm{TN})$, and $\mathrm{P} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$. Finally, the F_1-score gives the harmonic mean of precision and recall to give a measure of the tests accuracy, i.e., $F_1 = 2 \cdot \mathrm{R} \cdot \mathrm{P}/(\mathrm{R} + \mathrm{P})$. Note that the ideal recall, precision and F_1-score is 1, and ideal fallout is 0. Furthermore, a ratio of $\mathrm{R}/\mathrm{F} > 1$ suggests the classifier is better than random. As a summary statistic, Tab. 5.1 and 5.2 presents the F_1-scores for all networks illustrated in Fig. 5.5, and the full classification results (e.g., precision, recall, and fallout) are given in Appendix A.1. The F_1-scores are thus a measure of

how relevant the recovered network is to the original (generating) network from our data-driven approach.

Table 5.1: F_1-scores for three-node ($M = 3$) networks of coupled Lorenz systems represented by Figure 5.5a-5.5d (network G^1 has no edges and thus an undefined F_1-score). The p-value of the TEE score is given in the top row of each table, with ∞ signifying using no significance testing, i.e., score (5.29).

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph	N	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^2	$5K$	0.8	0.5	0.8	0.5	0.8	0.5	0.8	0.5
	$25K$	1	0.8	1	0.5	1	0.5	1	0.8
	$100K$	1	0.5	1	1	1	1	1	0.8
G^3	$5K$	1	0.67	1	1	1	1	1	0.67
	$25K$	1	1	1	0.5	1	1	1	1
	$100K$	1	1	1	1	1	1	1	1
G^4	$5K$	0.8	-	0.8	0.8	0.8	0.5	0.8	-
	$25K$	1	1	1	1	1	0.5	1	1
	$100K$	1	1	1	1	1	1	1	1

In general, the results of Tab. 5.1 and 5.2 show that the scoring function is capable of recovering the network with high precision and recall, as well as low fallout. In the table, the cell colours are shaded to indicate higher (white) to lower (black) F_1 scores. The best performing score is that with a p-value of 0.01, whereas no penalisation (a p-value of ∞) has the second highest classification results. As expected, the graphs recovered from data with low observational noise ($\sigma_\psi = 1$) are more accurate than those inferred from noisier data ($\sigma_\psi = 10$). The results for three-node networks (shown in Tab. 5.1) yields mostly full recovery of the structure for a higher number of observations $N \geq 75K$; whereas, the four-node networks (shown in Tab. 5.2) are more difficult to classify.

Interestingly, the statistical significance testing does not have a strong effect on the results. It is unclear if this is due to the use of the non-parametric density estimators, which, in effect, are parsimonious in nature since transfer entropy will likely reduce when conditioning on more variables with a fixed samples size. One challenging case is the empty networks G^1 and G^5; this is shown in Appendix A.1, where the fallout is rarely 0 for any of the p-values or sample sizes (although a large number of observations $N = 100K$ appears to reduce spurious edges). It would be expected that significance testing on these networks would outperform the naive score (5.29) given that a non-zero bias is introduced for a finite number of observations, although this is not the case in our experiments.

Table 5.2: F_1-scores for four-node ($M = 4$) networks of coupled Lorenz systems represented by Figure 5.5e-5.5h (network G^5 has no edges and thus an undefined F_1-score).

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph	N	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^6	$5K$	0.57	0.5	0.57	0.29	0.57	0.29	0.57	-
	$25K$	0.75	0.33	0.75	0.33	0.75	0.29	0.75	0.33
	$100K$	1	0.33	1	0.57	1	0.4	1	0.33
G^7	$5K$	1	0.25	1	0.29	0.75	0.25	0.75	0.57
	$25K$	1	0.5	1	0.86	1	0.86	1	0.5
	$100K$	1	0.86	1	0.86	1	0.86	1	0.86
G^8	$5K$	1	0.25	1	0.57	1	0.75	1	0.25
	$25K$	1	0.86	1	0.86	1	0.86	1	0.86
	$100K$	1	0.86	1	0.86	1	0.57	1	0.86

5.7 SUMMARY

In this chapter we explored the concept of structure learning for partially observable systems, which we model as a type of DBN known as a POSGDS. We provided exact approaches for computing the likelihood of a certain structure, given data, using state space reconstruction methods. Moreover, these approaches can be highly efficient if the data are generated by functions from the exponential family. We concluded the chapter by performing structure learning on a network of coupled Lorenz attractors, showing the approach works in practice.

This chapter explored (information-theoretic) reasoning in the sense of formulating a logical model of a distributed system, where the data were obtained passively. In the remainder of the thesis, we focus on another decision-theoretic problem: actively gathering data for improved environmental modelling. This is achieved by robotic systems recording data and optimising their data stream through informative path planning algorithms. That is, we now begin to explore reasoning in terms of optimising the data input about a phenomena, rather than just the model of this input.

6 INFORMATION GATHERING FOR ROBOTIC WILDLIFE TRACKING

In the previous two chapters, we used information theory as a tool to study the quality of statistical models given data. In the remainder of this thesis, we consider autonomous systems whereby the objective is to better understand (explore) the environment. That is, in addition to taking measurements of the environment, autonomous systems can execute actions in order to influence future observations. In this context, entropy is the *de facto* measure for quantifying belief uncertainty about the environment and thus the robots decisions are typically made by maximising information gain at each decision step.

6.1 OVERVIEW

In this chapter, we present and validate a complete system for autonomous wildlife telemetry tracking of small, dynamic animals. We show that this system addresses the associated theoretical and engineering challenges to a degree that is sufficient to match or surpass the performance of skilled human trackers. Moreover, we present preliminary experiments that show an earlier iteration of the system is capable of autonomous localisation of stationary radio tags and live radio-tagged birds.

First, we provide a rigorous derivation for our data-driven sensor model. In addition to estimating the targets' current location, this range-azimuth model is further used to predict the quality of future viewpoints in planning an approximately optimal sequence of observations. The observations are obtained by a novel two-point phased array, designed for use on-board a lightweight multirotor platform. This phased array antenna comprises two monopole antennas, mounted to a carrier rail (shown carried by the platform in Fig. 6.1). The robot performs a full rotation to produce unambiguous range-azimuth measurements with associated observation uncertainty. Although the time duration of a single observation is roughly 45 s, we found that reasonable localisation does not require a large number of observations. This motivates a greedy information-based planning approach for planning the next observation point online.

Target location estimates are represented by a grid-based filter, recursively updated following each observation. The measurement bearing (and uncertainty) is obtained by determining the phase shift between an observed gain pattern and the expected Fourier series radiation pattern model through a sliding-correlation technique. We show that the observation bearing error is normally distributed about a bearing measurement with variable uncertainty (heteroscedasticity). Bayesian data fusion is then used to incorporate the likelihood of each observation into the target belief. Estimation in the plane is sufficient in our case because the resulting estimate will generally be used to either confirm the presence of an animal in an area, or to visually locate and catch the animal for sample collection.

We performed preliminary experiments in order to learn and subsequently validate a bearing-only sensor approach. We present results from 22 flights comprising 131 observations and spanning nearly three hours of accumulated flight time. Of these, we performed eight manual flights for system identification and six autonomous flights localising stationary tags in three different areas. These results validate that the estimation process can locate stationary targets to within 30 m. Following this, we performed three flight trials using live noisy miners (*Manorina melanocephala*) where the robot localises the target while a human tracks its position visually. This evaluation demonstrates the feasibility of localising birds in the field with low-power radio frequency (RF) tags and a small multirotor with limited flight time.

We then directly compare the full range-azimuth tracking system against human operators in the problem of tracking the critically endangered swift parrot (*Lathamus discolor*) species in the wild. The full system includes range estimates of the target to improve planning. Across eight field trials, the estimated bird locations are precise to within 50 m, which is sufficient for recapture or data readout. Moreover, the time taken to achieve these estimates is comparable to, and often faster than, experienced human trackers. This result is significant because it is the first time in over 50 years of wildlife telemetry tracking research that a robotic system has been validated, in direct comparison with humans themselves, as an autonomous or human-assistive device. This milestone paves the way for the widespread use of robots in migration ecology and conservation management for small, dynamic species.

6.2 PROBLEM STATEMENT

Consider a single robot taking a sequence of observations in some workspace $\mathcal{S}$ to locate a target animal. By time t, the robot has obtained observations of the animal

Figure 6.1: The aerial robot system is designed to track small animals that are instrumented with lightweight radio collars, e.g., swift parrots (*Lathamus discolor*) 6.1a, brush-tailed rock-wallabies (*Petrogale penicillata*) 6.1b, and noisy miners (*Manorina melanocephala*) 6.1c. This work demonstrates that the robot is able to track swift parrots and yield comparable performance to an expert human operator performing traditional wildlife telemetry tracking 6.1d. The multirotor platform 6.1e-6.1f includes a lightweight directional antenna system and payload that receives the signal strength from the tag. This data is then transmitted to a ground control station for processing and online decision making.

at a set of times $0 \leq t_1 \leq \ldots \leq t_n \leq t$. Now, for a given observation time t_n, denote $\boldsymbol{X}_n = \boldsymbol{X}(t_n) \in \mathcal{S}$ as the robot location, $\boldsymbol{Y}_n = \boldsymbol{Y}(t_n) \in \mathcal{S}$ as the target's location and $\boldsymbol{Z}_n = \boldsymbol{Z}(t_n) \in \mathcal{H}$ as the observation of the target in some measurement space $\mathcal{H}$ (i.e., bearing and azimuth to the target, defined in Sec. 6.4). We will occasionally use the variable $\boldsymbol{U}_n = \{\boldsymbol{X}_n, \boldsymbol{Z}_n\}$ to denote the combined state-observation pair. Further, true (or optimal) quantities are denoted with an asterisk (e.g., $\boldsymbol{y}_n^*$ is the true location of the bird at time n) and estimates are denoted with a hat (e.g., $\hat{\boldsymbol{y}}_n$ is the target estimate at time n).

The above variables form stochastic processes by which we can estimate the target location at a given time. That is, the robot path is denoted $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$ with $\boldsymbol{z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N)$ denoting the associated sequence of observations. From these sequences, at any given time t_n, the robot is tasked with estimating the target location $\hat{\boldsymbol{Y}}_n$ from the entire history of observations $\hat{\boldsymbol{y}}_n = \hat{Y}(\boldsymbol{x}_n^-, \boldsymbol{z}_n^-)$. This requires a sensor model that converts raw sensor data to instantaneous estimates of the target.

Our overall objective is to know where the target animal is and with what certainty. Thus, the problem can be considered under the framework of information gathering [235] where the goal is to reduce uncertainty about the final estimate $\hat{\boldsymbol{Y}}_N$. To quantify uncertainty, we use the *de facto* measure of Shannon entropy. In this context, we aim to choose a sequence $\boldsymbol{u} = \{\boldsymbol{x}, \boldsymbol{z}\}$ of state-measurement pairs such that the final entropy of the belief $H(\hat{\boldsymbol{Y}}_N)$ is minimised. That is, let $\mathcal{U} = (\mathcal{S} \times \mathcal{H})$ and fix the measurement space $\mathcal{H}$, we can formally state the objective as that of minimising the final entropy based on the sequence of locations and observations:

$$\boldsymbol{U}^* = \underset{\boldsymbol{U} \subseteq \mathcal{U}}{\arg\min}\, \mathbf{E}_{\boldsymbol{U}} \left[H\left(\hat{Y}(\boldsymbol{U})\right) \right] \ . \tag{6.1}$$

6.3 SENSOR AND SENSOR DATA

This section describes the antenna and the raw data collected for each observation. First, the radio tag emits an on-off keyed pulse signal; this transmission is received by the payload on-board the UAV and the received signal strength indicator (RSSI) values of the signal are captured. These filtered RSSI values are linearly related to the power received during a transmission and are the raw sensor data used for the observation.

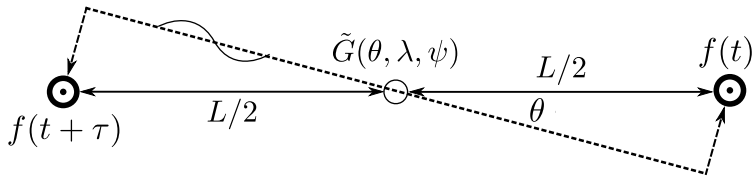


Figure 6.2: Two-point phased array antenna. Two monopole antennas are separated by a spacing L. This spacing causes a phase offset τ between the fore and aft antenna as a function of azimuth angle of arrival θ. The two signals are summed with a combiner circuit that has an additional (constant) phase offset ψ; this then generates a gain pattern $g(\theta) \propto 1 + \cos(\tau)$.

6.3.1 *Two-point phased array*

We designed a two-point phased array antenna with a front lobe and back null radiation pattern to account for weight and precision requirements. Shown in Fig. 6.2, the array consists of two quarterwave monopole antennas situated in front and behind the vehicle centre of gravity with a spacing $L(\lambda) < \lambda/4$, where λ is the transmitter wavelength. This spacing lags the aft antenna by a phase difference τ as a function of azimuth angle of arrival Θ

$$\tau(\Theta, \lambda, \psi) = \frac{2\pi L(\lambda)}{\lambda} \cos(\Theta) + \psi \,. \tag{6.2}$$

In Eq. (6.2), ψ is introduced by an RF combiner with a passive phase offset ψ between the fore and aft antenna. In accordance with (6.2), if the angle of arrival is perpendicular to the antenna array, the aft antenna phase lag $\tau(\Theta = \pi, \lambda, \psi) = \psi$. The interference pattern from this phase difference is then simply $1 + \cos(\tau(\theta, \lambda, \psi))$. From (6.2), the asymmetric gain pattern as a function of angle of arrival in dBi is

$$\tilde{G}(\Theta, \lambda, \psi) \text{ dBi } = 20 \log_{10} [1 + \cos(\tau(\Theta, \lambda, \psi))] + 1 \,. \tag{6.3}$$

Equation (6.3) is the ideal case; dBi is gain relative to a standard half-wave dipole antenna, however with a non-infinite ground plane and inaccuracy in manufacturing and vibrations, the actual pattern is distorted. Further, in Sec. 6.7 we introduce an analog circuit to directly sample the RSSI. This causes a distortion of the gain pattern $\tilde{G}(\Theta, \lambda, \psi)$. Thus, for a given antenna setup, i.e., fixing λ and ψ, we compute the expected gain pattern $\mathbf{E}[G \mid \Theta]$ empirically, as discussed in Sec. 6.3.2. The theoretical gain pattern (presented later in Fig. 6.4a) is sampled from Eq. (6.3) and shows the directionality of the antenna in the fore-aft asymmetry.

6.3.2 *Observed and expected sensor data*

In order to reduce noise and spurious readings due to multipath propogation, the UAV remains stationary while yawing through a full rotation. During this rotation, the continuous RSSI values are filtered and sampled at a constant rate to give a scalar value g_k associated with the bearing of the kth value ϕ_k. These values are then transmitted to a base station, giving the recorded gain pattern $\boldsymbol{g} = (g_1, \ldots, g_K)$. As a result, the random vector $\boldsymbol{G}_n = \boldsymbol{g}$ is a function of $(\boldsymbol{X}_n, \boldsymbol{Y}_n)$. If it is clear based on context, we drop the subscript n for notational convenience such that $\boldsymbol{G} = \boldsymbol{G}_n$.

Further, let $b(\boldsymbol{x}_n, \boldsymbol{y}_n)$ denote the bearing from $\boldsymbol{x}_n$ to $\boldsymbol{y}_n$. The true bearing to the target from the robot location $\boldsymbol{x}_n$ is then $\theta_n^* = b(\boldsymbol{x}_n, \boldsymbol{y}_n^*)$. We assume that the error for each recorded RSSI value is normally distributed with unknown variance $\sigma^2(\theta_n^*)$ that remains constant throughout an observation, i.e., for arbitrary $g_k \in \boldsymbol{g}$

$$g = \mathbf{E}[G \mid \Theta_n = \theta_n^*] + \nu_G, \qquad \nu_G \sim \mathcal{N}(0, \sigma^2(\theta_n^*)), \tag{6.4}$$

where $\sigma^2(\theta_n^*) = \mathbf{V}\left(G \mid \Theta_n = \theta_n^*\right)$.

We obtain the expected gain pattern $\mathbf{E}[\boldsymbol{G} \mid \Theta_n]$ by linear regression. Specifically, we fit the expected gain pattern to a Jth-order Fourier series $\varphi : \mathbb{R} \to \mathbb{R}$, i.e., given the true bearing θ,

$$\begin{aligned}\mathbf{E}\left[G_k \mid \Theta_n = \theta\right] &= \varphi(\theta + \phi_k) \\ &= a_0 + \sum_{j=1}^{J} a_j \cos(j(\phi_k + \theta)) + \sum_{j=1}^{J} b_j \sin(j(\phi_k + \theta)).\end{aligned} \tag{6.5}$$

From this Fourier model, we obtain the expected gain pattern $\boldsymbol{\varphi}(\theta) = \mathbf{E}\left[\boldsymbol{G} \mid \Theta_n = \theta\right]$, where $\boldsymbol{\varphi} : \mathbb{R} \to \mathbb{R}^K$ is generated by sampling the Fourier series (Eq. (6.5)) with a phase offset θ at K regular intervals, i.e., $\boldsymbol{\varphi}(\theta) = (\varphi(\theta), \varphi(\theta + 2\pi/K), \ldots, \varphi(\theta + 2\pi))$.

Given the expected and observed sensor output, $\boldsymbol{\varphi}$ and $\boldsymbol{g}$, the main goal of Bayesian sensor data fusion is to compute probability density functions (PDFs) of the bearing and range to a target from the robot. Given that the likelihoods are assumed to be Gaussian, the measurement tuple $\boldsymbol{Z}_n = \{\mu(\boldsymbol{G}_n), \sigma^2(\boldsymbol{G}_n)\}$. To learn the mapping from $\boldsymbol{G}_n$ to $\boldsymbol{Z}_n$, we use a data-driven approach based on training experiments, described below.

6.4 LIKELIHOOD FUNCTIONS FOR OBSERVATIONS

The most critical component of the system is the sensor model, which allows us to convert the signal received from the radio tag to an instantaneous estimate of

the target's location. An inaccurate or overconfident observation can lead to poor decision making and imprecise final location estimates.

We are interested in learning the *likelihood function* $L(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{z}_n)$, i.e., the probability of receiving the measurement $\boldsymbol{z}_n$ at location $\boldsymbol{x}_n$, given the target is at a particular location $\boldsymbol{y}_n$:

$$L(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{z}_n) = p\left(\boldsymbol{X}_n = \boldsymbol{x}_n, \boldsymbol{Z}_n = \boldsymbol{z}_n \mid \boldsymbol{Y}_n = \boldsymbol{y}_n\right) \text{ for } \boldsymbol{y}_n \in \mathcal{S}. \tag{6.6}$$

We could include uncertainty about the vehicle location $\boldsymbol{x}_n$ by including it in the measurement $\boldsymbol{z}_n$ (i.e., formulating the problem as a POMDP), however, we assume full knowledge of vehicle state in this chapter.

To construct our sensor models, we must determine what we are measuring and the uncertainty over these measurements. As mentioned above, our measurements $\boldsymbol{Z}_n$ are, broadly speaking, functions of the random vector $\boldsymbol{G}$.

In this work we take both range and azimuth readings of the target, where both observations are assumed to be normally distributed. This results in each measurement comprising the mean and variance $\boldsymbol{z}_n = \{\mu, \sigma^2\}$. Given a *measurement function* $h : (\mathcal{S} \times \mathcal{S}) \to \mathcal{H}$ that maps the vehicle $\boldsymbol{x}_n$ and target state $\boldsymbol{y}_n$ to the measurement space $\mathcal{H}$, the Gaussian likelihood function is:

$$L(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{z}_n) = f(h(\boldsymbol{x}_n, \boldsymbol{y}_n); \mu, \sigma^2), \tag{6.7}$$

where f is the PDF of the normal distribution.

6.4.1 *Azimuth likelihood function*

We model the likelihood of each azimuth measurement with a Gaussian bearing-error model [224] where $\boldsymbol{Z}_{\Theta_n} = \{\mu_\Theta(\boldsymbol{G}), \sigma^2_\Theta(\boldsymbol{G})\}$. That is, the difference between the true bearing to the target θ^*_n and the estimated bearing $\hat{\theta}_n$ (i.e., the bearing-error) is Gaussian distributed. Importantly, the bearing estimate $\hat{\theta}_n = \mu_\Theta(\boldsymbol{g})$ and its variance $\sigma^2_\Theta(\boldsymbol{g})$ are not measured directly but instead are given as functions of observation quality (i.e., the correlation coefficient, discussed below). As a result, when $\boldsymbol{G} = \boldsymbol{g}$, the bearing-error likelihood function L_Θ is given by

$$L_\Theta(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{z}_{\Theta_n}) = f\left(b(\boldsymbol{x}_n, \boldsymbol{y}_n); \mu_\Theta(\boldsymbol{g}), \sigma^2_\Theta(\boldsymbol{g})\right). \tag{6.8}$$

Now, given our model $\boldsymbol{\varphi}$ of the gain pattern, our problem becomes that of inverse regression to find the expected bearing and uncertainty. The Gaussian bearing-error assumption states

$$\hat{\theta}_n = \theta^*_n + \nu_{\Theta_n}, \qquad \nu_{\Theta_n} \sim \mathcal{N}(0, \sigma^2_\Theta(\boldsymbol{g})), \tag{6.9}$$

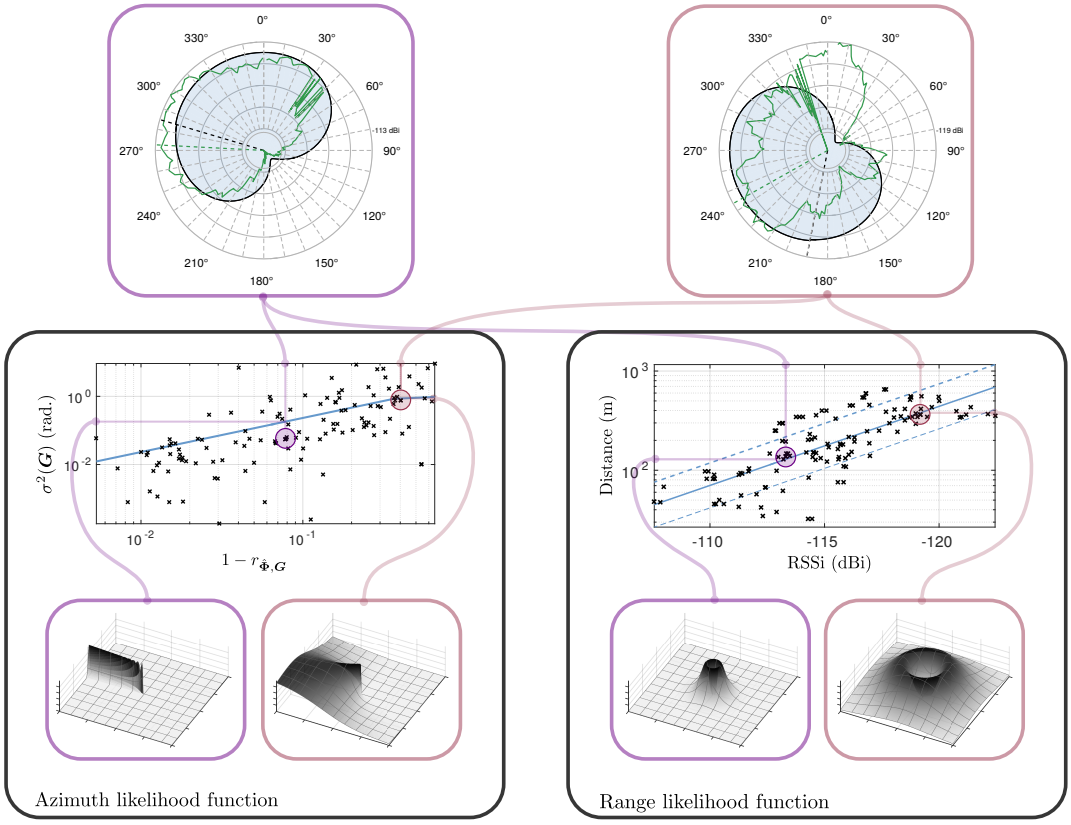
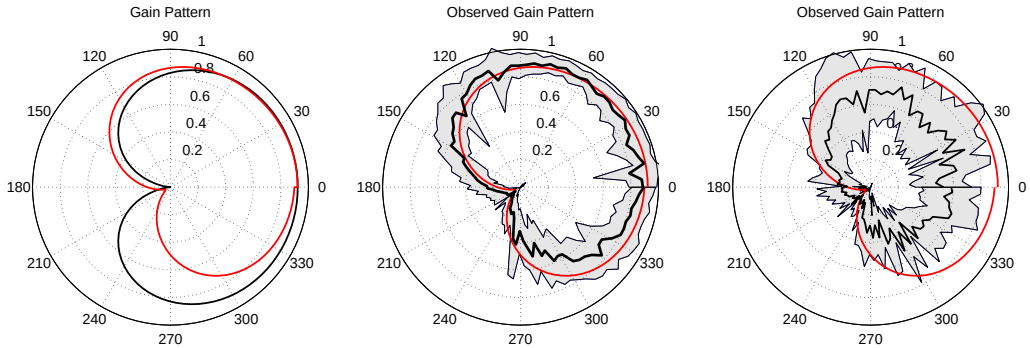


Figure 6.3: Obtaining range-azimuth likelihood functions from observations. The top row illustrates two example observations taken online with a stationary target. The radial plots illustrate real RSSI readings (green line) $\boldsymbol{g}$ and a third-order Fourier series model $\boldsymbol{\varphi}(\theta)$ of the radiation pattern (black line). The model is offset (rotated) such that it is oriented towards the true bearing to the target θ_n^*, and the RSSI values are offset by the maximum correlation $\mu_\Theta(\boldsymbol{g}) = \arg\max_\theta r_{\boldsymbol{\varphi}(\theta),\boldsymbol{g}}$. These offsets are illustrated with dotted green and black radial lines. On the left subfigure, the maximum value correlation coefficient $r_{\hat{\boldsymbol{\varphi}},\boldsymbol{g}}$ maps to a bearing-error $\sigma^2_\Theta(\boldsymbol{g})$, which is illustrated in the grid plots below. On the right subfigure, the maximum RSSI value $g_{\max}$ maps to an expected range $\mu_R(\boldsymbol{g})$ with a fixed range-error $\hat{\sigma}^2_R$ giving the associated grid plots below.



(a) Theoretical gain (b) Stationary tag observations (c) Real bird observations

Figure 6.4: Third-order fast Fourier transform gain pattern model $\mathbf{E}[\boldsymbol{G} \mid \theta]$ (red) plotted against: 6.4a the theoretical two-point phased array model $\tilde{G}(\theta, \lambda, \psi)$ (without on-board filter distortion), 6.4b the observations relative to a static tag in the canopy $\boldsymbol{G}_{stationary}$, and 6.4c the observations relative to a real bird (the noisy miner), moving during observations $\boldsymbol{G}_{bird}$. In 6.4b and 6.4c, the mean (solid black line) and standard deviation (shaded gray) normalised gain pattern are shown.

when $\hat{\theta}_n = \mathbf{E}[\Theta \mid \boldsymbol{g}]$ and $\sigma^2_\Theta(\boldsymbol{G}) = \mathbf{V}(\Theta_n \mid \boldsymbol{g})$. We find the expected azimuth by minimising the sum of squares of the residuals, i.e.,

$$\hat{\theta}_n = \mu_\Theta(\boldsymbol{g}) = \underset{\theta \in [0, 2\pi)}{\arg\min} \|\boldsymbol{g} - \boldsymbol{\varphi}(\theta)\|^2. \tag{6.10}$$

To infer the variance $\mathbf{V}(\Theta_n \mid \boldsymbol{g})$ for a given signal $\boldsymbol{g}$, we note that the collection of $\{\boldsymbol{G}\}$ is *heteroscedastic*, i.e., the conditional variance can change with each observation. This is shown in the scattergram in Fig. 6.3 where the bearing error is plotted against observation quality (correlation). We assume this unexplained variance is due to hidden causes of observation noise, such as the target animal moving during a measurement, or spurious recordings due to multipath interference. In typical regression, heteroscedasticity is considered undesirable and is reduced by introducing more regressors or non-linear transformations of the existing variables. In our case, given that this knowledge is hidden, we cannot introduce more variables and instead marginalise out this quantity to infer the conditional variance from data. Below, we show how the *coefficient of determination* expresses the proportion of variability in our model (i.e., the heteroscedasticty is attributed to bearing error).

In the context of regression, we can obtain the *fraction of variance unexplained (FVU)* for a response variable through the coefficient of determination. In linear regres-

sion, where we have the sample variance s_g^2 as an estimate of the population variance $\mathbf{V}(G)$, the FVU is given by the sample correlation coefficient r^2:

$$\frac{\mathbf{V}(G \mid \Theta_n = \hat{\theta})}{\mathbf{V}(G)} \simeq \frac{s^2_{g|\hat{\theta}}}{s_g^2} = 1 - r^2_{g,\boldsymbol{\varphi}(\hat{\theta})}. \tag{6.11}$$

However, we are interested in the bearing variance $\mathbf{V}(\Theta_n \mid \boldsymbol{G})$, which we can approximate from the model variance $\mathbf{V}(\boldsymbol{\varphi}(\Theta_n) \mid \boldsymbol{G})$ by Taylor expansion. Recall that our estimate $\hat{\theta}_n = \mu_\Theta(g_1, g_2, \ldots, g_K)$ is a function of the random vector $\boldsymbol{G}$. We can approximate the variance of this mapping via a first-order Taylor expansion [8],

$$\mathbf{V}(\Theta \mid \boldsymbol{G}) \simeq \sum_{i=1}^{N}\sum_{j=1}^{N} \boldsymbol{\Sigma}_{ij} \frac{\partial \mu_\Theta(\boldsymbol{\varphi}(\Theta))}{\partial G_i} \frac{\partial \mu_\Theta(\boldsymbol{\varphi}(\Theta))}{\partial G_j}. \tag{6.12}$$

Now, because the measurement $\boldsymbol{G}$ comprises i.i.d. variables G_n, the covariance matrix is given by $\boldsymbol{\Sigma} = \mathbf{V}(G \mid \Theta)\boldsymbol{I}_N$ where $\boldsymbol{I}_N$ is the identify matrix. This gives the conditional variance in Eq. (6.12) as

$$\mathbf{V}(\Theta_n \mid \boldsymbol{G}) \simeq \mathbf{V}(G \mid \Theta_n) K \sum_{n=1}^{K} \left(\frac{\partial \mu_\Theta(\boldsymbol{G})}{\partial G_k} \right)^2. \tag{6.13}$$

Since small changes in each realisation of G will introduce small changes in μ_Θ, the variance in Eq. (6.13) is approximately linear for low noise ν_G; however, the approximation becomes worse as ν_G becomes large. By using the coefficient of determination (Eq. (6.11)), we can express the variance of a given sensor reading $\boldsymbol{g}$ in Eq. (6.13) as

$$\sigma^2(\boldsymbol{g}) = \mathbf{V}(\Theta_n \mid \boldsymbol{G} = \boldsymbol{g}) \simeq s_g^2 \left(1 - r^2_{\boldsymbol{\varphi}(\hat{\theta}),\boldsymbol{g}}\right) N \sum_{n=1}^{N} \left(\frac{\partial \mu_\Theta(\boldsymbol{g})}{\partial g_n} \right)^2. \tag{6.14}$$

Thus, $\sigma^2(\boldsymbol{g})$ can be expressed as a function of $s_g^2 \left(1 - r^2_{\boldsymbol{\varphi}(\hat{\theta}),\boldsymbol{g}}\right)$.

In practice, we regress only on $\sigma^2(\boldsymbol{g})$, assuming the variable is a piecewise continuous function of the explanatory variable $(1 - r_{\boldsymbol{\varphi}(\hat{\theta}),\boldsymbol{g}})$. We can also determine azimuth $\hat{\theta} = \mu_\Theta(\boldsymbol{g})$ for each measurement $\boldsymbol{g}$ by the correlation coefficient $r_{\boldsymbol{\varphi}(\hat{\theta}),\boldsymbol{g}}$. That is, following each observation, the recorded gain pattern is correlated against the model $\boldsymbol{\varphi}(\theta)$ with regular phase offsets θ and the lag that corresponds to the maximum correlation then gives the estimated angle of arrival, i.e., Eq. (6.10) becomes $\mu_\Theta(\boldsymbol{g}) = \arg\max_{\theta \in [0,2\pi)} r_{\boldsymbol{\varphi}(\theta),\boldsymbol{g}}$. This process of obtaining an azimuth observation is illustrated on the left of Fig. 6.3 and example likelihood functions from one trial can be seen in Fig. 6.5.

6.4.2 *Range likelihood function*

Next, we estimate the distance to the target using a Gaussian range-error model where the set $\boldsymbol{Z}_R = \{\mu_R(\boldsymbol{G}), \hat{\sigma}_R^2\}$. The range errors are assumed to be logarithmic,

as discussed below. Furthermore, unlike the bearing observations, the scattergram in Fig. 6.3 does not indicate the noise is heteroscedastic, i.e., the variance is constant for each observation. This yields the likelihood function

$$L_R(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{z}_{R_n}) = f\left(\log\left(d(\boldsymbol{x}_n, \boldsymbol{y}_n)\right); \mu_R(\boldsymbol{g}), \hat{\sigma}_R^2\right). \tag{6.15}$$

In general, range measurements in cluttered environments can be highly imprecise due to multipath interference. We anticipate the vehicle to be deployed in similar environments and estimate the variance under these conditions. Although the error in range measurements can be significant, including such observations is still useful. Because the noise is homoscedastic, we can rely on range measurements to provide an approximate location. The ability to focus on an approximate location is particularly beneficial when the search area would otherwise be expansive, such as in tracking scenarios where there is little prior knowledge of the target's location, and when bearing uncertainty is high.

We are interested in mapping the sensor output $\boldsymbol{g}$ to the distance between transmitter and receiver. Due to atmospheric interactions, the signal amplitude will decrease with range. Denote $d(\boldsymbol{x}_n, \boldsymbol{y}_n)$ as the Euclidean distance between our receiver $\boldsymbol{x}_n$ and the transmitter $\boldsymbol{y}$. Then, the received power w_n is a function of the transmitted power v_n and the attenuation per meter α [37]:

$$w_n = v_n e^{\alpha d(\boldsymbol{x}_n, \boldsymbol{y}_n)}. \tag{6.16}$$

In Eq. (6.16) we have assumed that w_n and v_n take into account the link budget, which characterises all gains and losses in the telecommunication system. Most of these components are fixed for a given system (e.g., transmitter and receiver losses), however, for a directional antenna, the gain relative to the average radiation intensity (the *isotropic directivity*) depends on the immediate angle of arrival ϕ_k. As a result, the RSSI values g_k are a function of the received power w_n and angle of arrival ϕ_k. The isotropic directivity is approximately constant if we take the maximum RSSI value $g_{\max} = \max_k g_k$. Thus, we use the value $g_{\max}$ to estimate distance.

Now, let the true distance to the target be $r_n^* = d(\boldsymbol{x}_n, \boldsymbol{y}_n^*)$ and its estimate be a function of $\boldsymbol{g}$, i.e., $\hat{r}_n = \mu_R(\boldsymbol{g})$. From the above discussion and Eq. (6.16), $w_n = w(g_{\max})$ for some linear function $w : \mathbb{R} \to \mathbb{R}$. Moreover, r_n^* is a function of $\log w(g_{\max})$ and the Gaussian range-error assumption may be expressed as

$$\log \hat{r}_n = \log r_n^* + \nu_R, \qquad \nu_R \sim \mathcal{N}(0, \sigma_R^2), \tag{6.17}$$

where $\sigma_R^2 = \mathbf{V}(\log R_n)$. We thus obtain the estimated range $\hat{r}_n$ as

$$\log \hat{r}_n = \mu_R(\boldsymbol{g}) = \alpha^{-1}(\log w(g_{\max}) - \log v_n). \tag{6.18}$$

The function $\mu_R(\boldsymbol{G})$ can be fitted to a first degree polynomial function of $\log g_{\max}$. The variance σ_R^2 is estimated by the sample variance $\hat{\sigma}_R^2 = s_R^2$. The procedure for obtaining a Gaussian range-error observation is illustrated on the right of Fig. 6.3 and example range likelihood functions can be seen in Fig. 6.5.

6.4.3 *Combined likelihood function*

The individual likelihood functions may be combined to obtain a range-azimuth likelihood function $L(\boldsymbol{y}_n; \boldsymbol{u}_n)$, where $\boldsymbol{Z}_n = \{\mu_\Theta(\boldsymbol{G}), \sigma_\Theta^2(\boldsymbol{G}), \mu_R(\boldsymbol{G}), \hat{\sigma}_R^2\}$. That is, assuming independent errors ν_{Θ_n} and ν_{R_n}, the likelihood functions are multiplied pointwise [224], i.e.,

$$L(\boldsymbol{y}_n; \boldsymbol{u}_n) = L_\Theta(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{z}_{\Theta_n}) \circ L_R(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{z}_{R_n}). \tag{6.19}$$

We tested the null hypothesis that these errors are independent by computing the sample correlation coefficient. Since the errors are assumed to be normal, the hypothesis was tested via a Student's t-distribution with 95% confidence and 150 observations. The results showed a correlation of $r_{\nu_{\Theta_n}, \nu_{R_n}} = -0.08 \pm 0.136$, giving a confidence of less than 66% that the errors are correlated. This result further supports the heteroscedasticity assertion, i.e., that poor quality observations are not significantly correlated with distance.

6.5 BAYESIAN DATA FUSION

Given the likelihood function in Eq. (6.19), we can combine numerous observations to determine the most likely position of the target animal. To achieve this, we use Bayesian data fusion, assuming independent observations; this process is illustrated in Fig. 6.5.

We are ultimately interested in knowing the probability of the target's state after all N observations, i.e., the *posterior belief* [224, 235],

$$bel(t_n, \hat{\boldsymbol{y}}_n) = p\left(\hat{\boldsymbol{Y}}_n = \hat{\boldsymbol{y}}_n \mid \boldsymbol{U}_n^- = \boldsymbol{u}_n^-\right) . \tag{6.20}$$

Further, we assume that the target can transition between observations such that $\boldsymbol{y}_n = \boldsymbol{y}_{n-1} + \boldsymbol{\nu}_Y$ with $\boldsymbol{\nu}_Y \sim \mathcal{N}(0, \boldsymbol{\Sigma}_Y)$ for some covariance $\boldsymbol{\Sigma}_Y$. This leads to the *transition density*

$$q(\boldsymbol{y}_n \mid \hat{\boldsymbol{y}}_{n-1}) = p\left(\hat{\boldsymbol{Y}}_n = \hat{\boldsymbol{y}}_n \mid \hat{\boldsymbol{Y}}_{n-1} = \hat{\boldsymbol{y}}_{n-1}\right) . \tag{6.21}$$

Computing the posterior belief (Eq. (6.20)) becomes simpler if the process $(\boldsymbol{Y}(t))_{t \geq 0}$ is assumed to be Markovian and each observation $\boldsymbol{Z}_n$ only depends on $\boldsymbol{Y}_n$, i.e.,

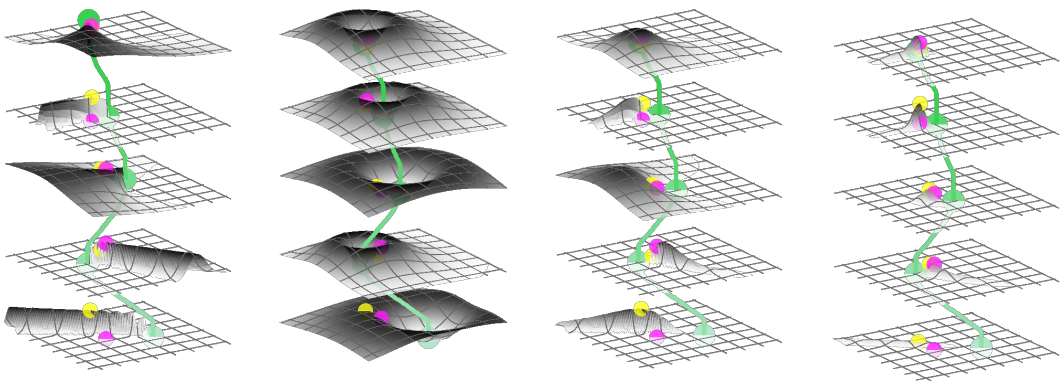


Figure 6.5: Bayesian data fusion to obtain target estimates. The distributions shown are spatially discrete grids over a 750 m squared area (with grid-lines every 100 m for illustrative purposes only). In each subfigure, time (observation number n) increases up the page and higher probability mass is represented as darker, raised regions. The UAV location $\boldsymbol{x}_n$ is illustrated by a green dot, the target location y_n in purple and the maximum likelihood estimate $\hat{\boldsymbol{y}}_n$ in yellow. From left to right: the bearing-only likelihood function L_Θ; the range-only likelihood function L_R; the combined likelihood function L_n; and the posterior belief $b(t_n, \boldsymbol{y}_n)$.

$L(\hat{\boldsymbol{y}}; \boldsymbol{u}) = \prod_{n=1}^{N} L(\hat{\boldsymbol{y}}_n; \boldsymbol{u}_n)$. As the likelihood function in Eq. (6.19) is defined this way, recursive Bayesian filtering [224] can be used to update the belief. That is, the posterior belief is computed as

$$\overline{bel}(t_n, \hat{\boldsymbol{y}}_n) = \int q(\hat{\boldsymbol{y}}_n \mid \hat{\boldsymbol{y}}_{n-1}) bel(t_{n-1}, \hat{\boldsymbol{y}}_{n-1}) \mathrm{d}\hat{\boldsymbol{y}}_{n-1} \tag{6.22}$$

$$bel(t_n, \hat{\boldsymbol{y}}_n) = \eta L(\hat{\boldsymbol{y}}_n; \boldsymbol{u}_n) \overline{bel}\,(t_n, \hat{\boldsymbol{y}}_n)\,, \tag{6.23}$$

where η is a normalisation constant such that $\int \mathrm{d}\hat{\boldsymbol{y}}_n bel(t_n, \hat{\boldsymbol{y}}_n) = 1$ and $L(\hat{\boldsymbol{y}}_n; \boldsymbol{u}_n)$ is the likelihood function (6.19). The first step (6.22) gives a *motion update*, and the second step (6.23) gives the *information update* to obtain a new belief of the target location [224].

Early approaches to recursive Bayesian filtering focused on Gaussian implementations due to convenient analytical solutions to computing the posterior belief in Eq. (6.20), e.g., Kalman filters and extensions such as the unscented and extended Kalman filters. However, these methods are approximations to the nonlinear, non-Gaussian Bayesian filter (shown in Eqs. (6.22)-(6.23)). In contrast, grid-based filtering allows for resolution-complete recursive estimation [9, 224] and can be computed in reasonable time over our workspace. Thus, we represent our workspace $\mathcal{S}$ as an $I \times J$ grid in $\mathbb{R}^2$.

The evolution model, Eq. (6.22), is functionally equivalent to Gaussian convolution. Further, given our grid-based workspace $\mathcal{S}$, this convolution is simply a Gaussian blur, a spatial (low-pass) filter commonly used in image processing. To

efficiently implement this model, we leverage results from computer vision for convolution and use a separable Gaussian kernel of width $3|\boldsymbol{\Sigma}_Y|$.

Finally, we require an estimate of the location of the target $\hat{\boldsymbol{y}}_n$ given the posterior $bel(t_n, \boldsymbol{y}_n)$. Two obvious choices for this estimate are the expected value of the posterior $\mathbf{E}[\boldsymbol{Y}_n] = \int \mathrm{d}\boldsymbol{y}_n \; \boldsymbol{y}_n \; bel(t_n, \boldsymbol{y}_n)$, or the maximum *a posteriori* (MAP) estimate $\arg\max_{\boldsymbol{y}_n \in \mathcal{S}} bel(t_n, \boldsymbol{y}_n)$. The MAP estimate performed marginally better in preliminary trials; however, in practice the target does not remain stationary and so we instead maximise recursively over all posteriors:

$$\hat{\boldsymbol{y}}_n = \underset{o \in [1,n), \boldsymbol{y}_o \in \mathcal{S}}{\arg\max} \; bel(t_o, \boldsymbol{y}_o) \; . \tag{6.24}$$

In this way, the location estimate likelihood is strictly increasing.

6.6 DECISION MAKING BY INFORMATION GAIN

The overall objective is to minimise entropy over the target location. However, it is more convenient to consider the equivalent problem of maximising the information gain of each observation.

Let $\hat{\boldsymbol{Y}}_{n^-}$ be distributed according to the target belief after the motion update step, i.e., Eq. (6.22)). The information gained in taking the action $\boldsymbol{U}_n = \boldsymbol{u}_n$ is quantified by the mutual information $I(\boldsymbol{Y}_n; \boldsymbol{Y}_{n^-})$ between the posterior and the prior belief:

$$I(\boldsymbol{Y}_n; \boldsymbol{Y}_{n^-}) = H(\boldsymbol{Y}_n) - H(\boldsymbol{Y}_n \mid \boldsymbol{Y}_{n^-}). \tag{6.25}$$

Decomposing Eq. (6.25) using the chain rule, the entropy minimisation problem defined in Eq. (6.1) can be expressed as

$$\boldsymbol{U}^* = \underset{\boldsymbol{U} \subseteq \mathcal{U}}{\arg\max} \, \mathbf{E}_{\boldsymbol{U}} \left[H(\boldsymbol{Y}_1) + \sum_{n=2}^{N} I\left(\boldsymbol{Y}_n; \boldsymbol{Y}_{n^-}\right) \right] . \tag{6.26}$$

The objective of Eq. (6.26) is equivalent to entropy minimisation and is, in general, non-convex and analytically intractable. However, the mutual information given in Eq. (6.25) is monotone submodular and thus the quality of the solution provided by a greedy algorithm is at least 63% of optimal [132]. That is, given a deterministic greedy algorithm that selects the action

$$\boldsymbol{U}_n = \underset{\boldsymbol{U}_n \in \mathcal{U}}{\arg\max} \, \mathbf{E}\left[I(\boldsymbol{Y}_n; \boldsymbol{Y}_{n^-}) \right] \tag{6.27}$$

at each decision step, the resulting path $\hat{\boldsymbol{u}}$ is within a constant factor of optimal of the objective shown in Eq. (6.26), i.e.,

$$\hat{\boldsymbol{u}} \geq \left(1 - \frac{1}{e}\right) \boldsymbol{u}^* . \tag{6.28}$$

Furthermore, this is the most efficient algorithm to obtain such a bound unless P = NP [132].

Optimising each observation $\boldsymbol{U}_n$ is constrained in that only the vehicle locations $\boldsymbol{x}_n \subset \boldsymbol{u}_n$ can be selected, and consequently only the expected information gain at each sample $\boldsymbol{s}$ can be computed, i.e., we choose future waypoints $\boldsymbol{x}_n$ such that

$$\boldsymbol{x}_n = \arg\max_{\boldsymbol{s}\in\mathcal{S}} \mathbf{E}\left[I(\boldsymbol{Y}_n;\boldsymbol{Y}_{n^-}) \mid \boldsymbol{X}_n = \boldsymbol{s}\right]. \tag{6.29}$$

As mentioned above, we assume independent errors in the likelihood functions shown in Eq. (6.19), giving $p(\boldsymbol{Z}_n) = p(\boldsymbol{Z}_{\Theta_n})p(\boldsymbol{Z}_{R_n})$. However, even solving for independent priors requires inverting all possible distributions at all sample locations $\boldsymbol{s} \in \mathcal{S}$; this is generally intractable.

As an efficient alternative, we assume that the target location for the next observation is the maximum likelihood position after the motion update, i.e., $\boldsymbol{Y}_n = \hat{\boldsymbol{y}}_{n^-}$. As a result, for a fixed viewpoint $\boldsymbol{s}$, the expected range measurement $\mathbf{E}[R_n \mid \boldsymbol{X}_n = \boldsymbol{s}, \boldsymbol{Y}_n = \hat{\boldsymbol{y}}_{n^-}] = d(\boldsymbol{s}, \hat{\boldsymbol{y}}_{n^-})$ and expected bearing measurement $\mathbf{E}[\Theta_n \mid \boldsymbol{X}_n = \boldsymbol{s}, \boldsymbol{Y}_n = \hat{\boldsymbol{y}}_{n^-}] = b(\boldsymbol{s}, \hat{\boldsymbol{y}}_{n^-})$ to the target are given. Moreover, the expected variance $\hat{\sigma}^2_\Theta$ is given by marginalising out $\boldsymbol{G}$ such that $\hat{\sigma}^2_\Theta = \mathbf{E}[\sigma^2_\Theta(\boldsymbol{G})] \simeq 0.2$ radians. In this case, the expected observation is a function of the viewpoint $\boldsymbol{s}$:

$$\hat{\boldsymbol{z}}_n(\boldsymbol{s}) = \{d(\boldsymbol{s}, \hat{\boldsymbol{y}}_{n^-}), \hat{\sigma}^2_\Theta, b(\boldsymbol{s}, \hat{\boldsymbol{y}}_{n^-}), \hat{\sigma}^2_R\}, \tag{6.30}$$

and the optimisation over potential viewpoints $\boldsymbol{s}$ from Eq. (6.29) becomes

$$\boldsymbol{x}_n = \arg\max_{\boldsymbol{s}\in\mathcal{S}} I\left(\boldsymbol{Z}_n = \hat{\boldsymbol{z}}_n(\boldsymbol{s}); \boldsymbol{Y}_n = \hat{\boldsymbol{y}}_{n^-}\right). \tag{6.31}$$

To reduce computation time, instead of sampling every location in the workspace $\boldsymbol{s} \in \mathcal{S}$ as indicated in Eq. (6.31), we simply sample a uniformly distributed subset. Given the stochastic nature of observations this does not appear to affect the quality of the planner.

6.7 EXPERIMENTAL SYSTEM

Our experimental system comprises a commercial UAV platform, a custom antenna array and sensor payload. Algorithmic components from the previous sections are implemented in the robotic operating system (ROS) [190] and executed on a ground-based laptop computer. This section describes the UAV platform and sensor payload components. An overview is shown in Fig. 6.6.

The UAV used in our system is the *Falcon 8*, a commercial eight-rotor platform manufactured by Ascending Technologies with proprietary high-quality flight control and autonomous GPS waypoint-following systems. It is structured around two

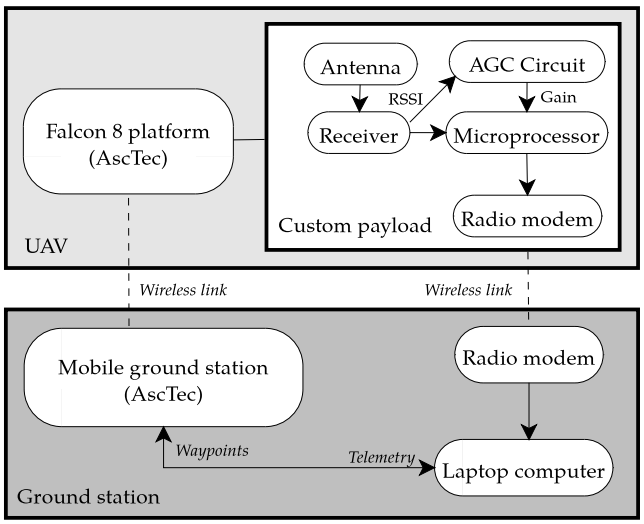
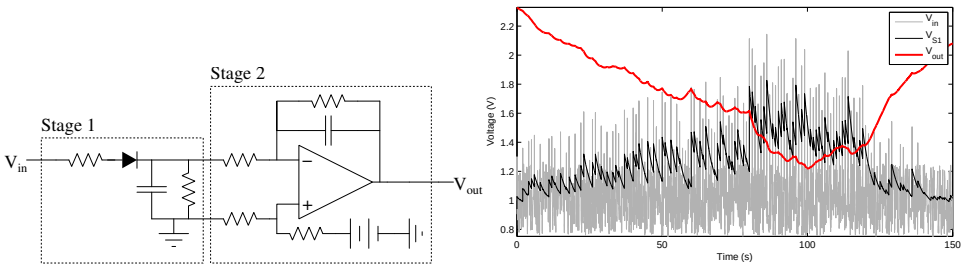


Figure 6.6: Diagram of the wildlife telemetry tracking system.



(a) Diagram of the AGC circuit (b) Simulation results for the AGC circuit

Figure 6.7: The AGC circuit. The diagram in 6.7a illustrates both stages of the circuit: catch and hold (Stage 1), and integrating amplifier (Stage 2). The plots in Fig. 6.7b show a simulated input pulse of 10ms with a period of 1.05s (V_{in}), the output from Stage 1 (V_{S1}), and output from Stage 2 (V_{out}).

colinear sets of four rotors with a maximum take-off weight of 2200g and payload capacity of 750g. The platform is connected by wireless communication to a ground station, which can relay telemetry data and accept control commands via USB.

The sensor array is fed into a custom transceiver subsystem that consists of a Radiometrix LMR1 receiver, an ARM 32-bit Cortex-M3 microprocessor mounted to a custom miniaturised printed circuit board, an analog filtering circuit, and a Digi XTend radio modem. These components were chosen such that the total mass of the sensor payload does not exceed the payload capacity of the Falcon 8. The complete system is shown earlier in Fig. 6.1.

The BioTrack Pip Ag393 radio tags regularly transmit an unmodulated on-off-keyed signal with a pulse width of 10 ms and period of 1.05 s; the receiver RSSI output is an equivalent waveform with an amplitude corresponding to the signal

strength at the RF input channel (see V_{in} in Fig. 6.7b). To avoid using high-powered signal processing components while at the same time receiving high-fidelity RSSI measurements, we designed a simple analog circuit to handle signal processing. The circuit implemented is a typical AGC circuit as seen in Fig. 6.7a. Stage 1 is a peak-hold rectifier circuit with a low leakage rate to hold the amplitude of each pulse (V_{S_1} in Fig. 6.7b); and Stage 2 acts as an inverting amplifier and integrator circuit to smooth the peak-hold signal and upsample the voltage to the analog-to-digital converter input range of 0 - 3.3 VDC (V_{out} in Fig. 6.7b). The filter output V_{out} is sampled at 5 Hz by an analog-to-digital converter within the microprocessor subsystem and transmits this packet to the ground station through a radio modem.

6.8 SYSTEM VALIDATION

In preliminary experiments, we validate that the bearing-only sensor model (Sec. 6.4.1) is suitable for *localising* radio-tagged wildlife and stationary targets. The uncertainty model for these experiments was learned in a data-driven approach similar to that described in Sec. 6.4.1. However, the number of pieces and breakpoints were chosen in a less rigorous manner (see [53] for details).

Here we provide results from two experiments with autonomous flight: (1) algorithm validation and (2) live bird trials. The aim of the algorithm validation is to localise a stationary tag mounted in the canopy of a tree. Live bird trials were performed with radio-tagged noisy miners (*Manorina melanocephala*), a small territorial bird species.

In these experiments, observations are taken at 50 m altitude from the launch elevation and take approximately 45 s to complete. Rotation rate for an observation was hand-tuned using data from five preliminary flights. For all autonomous flights, observation positions were computed online using the planner from Sec. 6.6.

We use three separate trial sites (Sites A, B and C) known to be within the territory of a target bird.

6.8.1 *Validation: Stationary tag*

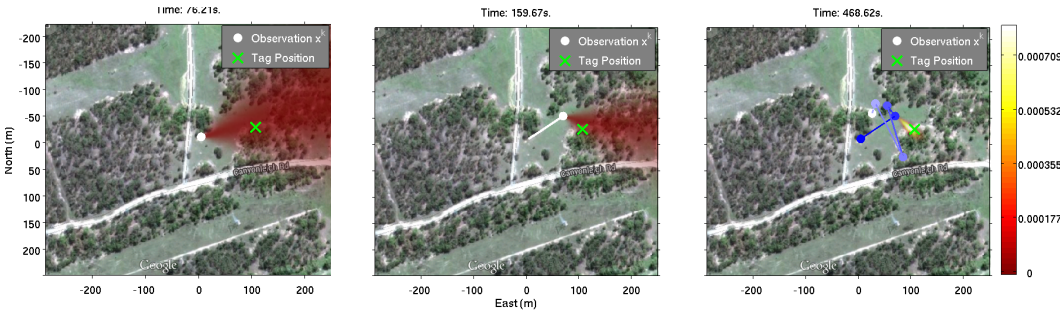
Six flights were performed to evaluate localisation performance on a stationary tag in a 1000x1000 m grid with 1 m-resolution cell edges. Table 6.1 presents the mean number of observations per trial, mean estimation error and mean entropy of the *a posteriori* belief of each trial, mean and variance of each observation range to tag, mean and variance of observation bearing error, and the mean and variance of the maximum likelihood correlation of each observation. The former measures

Table 6.1: Localisation results for a stationary tag. Each tag was placed in the canopy of a site within a target bird's territory.

Site	Trials	Total Observations	Mean Error* (m)	Mean Entropy* (bits)	Mean Distance (m)		Correlation $r^{\dagger}$		Observation Error ν_{Θ_n} (rad.)†	
					μ	(σ)	μ	(σ)	μ	(σ)
A	2	11	16.43	10.56	73.3	(22.0)	0.951	(0.0489)	0.1173	(0.0703)
B	2	11	25.2	15.7	203	(95.4)	0.941	(0.0355)	0.0691	(0.0537)
C	2	8	29.9	15.86	230	(118.9)	0.848	(0.175)	0.215	(0.349)
	6	30	23.8	14.04	168.8	(78.77)	0.913	(0.0865)	0.1338	(0.1577)

* Based on ground truth coordinate at the end of each trial

† Each observation is independent of parent trial



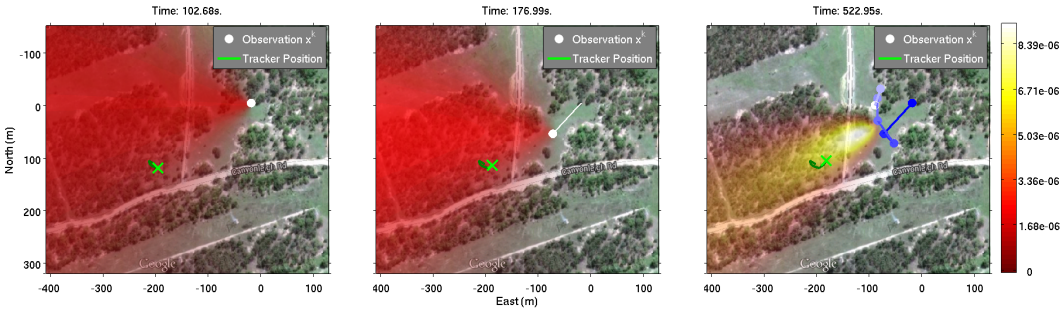
(a) $n = 1$ (b) $n = 2$ (c) $n = 6$

Figure 6.8: Localisation of a static radio tag in a tree canopy. Figures 6.8a, 6.8b and 6.8c illustrate the convergence of the *a posteriori* belief $bel(t_n, \hat{\boldsymbol{y}}_n)$ of the tag location after the first, second and last observation for this trial, respectively. The belief is represented as a grid with 1m resolution.

report on the final observation of each trial; the latter are calculated over all observations at each site (e.g., 11 observations at Site A). Lower values for all statistical measures imply higher accuracy, precision and certainty of the tag's position. The observation range is dependent only on environmental factors.

Site A yields better results across all measures, even though many observations are closer than the ideal system range, most likely suggesting the actual system range is shorter than that reported in [53]. Further, the site's landscape was relatively flat with sparser vegetation than other sites and thus yields less dominant multipath propagation. In Sites B and C the algorithm performs sufficiently and consistently localises the stationary tag to within 30 m.

A typical trial for the stationary tag at Site A is depicted in Fig. 6.8 showing the *a posteriori* belief, vehicle trajectory and tag location for observations 1, 2 and 6. Ideally, the planner would be able to circle around the centroid, however due to safety precautions the planner considers waypoints within a 90 m radius of the UAV



(a) $n = 1$ (b) $n = 2$ (c) $n = 7$

Figure 6.9: Localisation of the noisy miners tagged with a low-power radio transmitter. Figures 6.9a, 6.9b and 6.9c illustrate the convergence of the *a posteriori* belief of the bird location after the first, second and last observation for this trial, respectively. As an indication of the actual bird position, the trajectory of the trackers during an observation (solid light green), and past trajectory (solid dark green) are shown.

home position (the map origin). To account for the aforementioned ideal range, the planning set excludes waypoints within 60 m of the *a priori* belief centroid.

6.8.2 *Validation: Noisy miners*

Following the algorithm validation on stationary targets, we performed three flights from different launch sites to localise small, live birds. Each bird was fitted with a radio transmitter equivalent to the stationary case above with the same grid parameters as above.

For these trials, several birds were captured and, subsequently, the radio tag was taped to their back feathers. Each tag transmitted on a unique frequency that was preprogrammed into both the manual and robotic receiver systems. The tags emitted an on-off key modulated signal as described above. Moreover, they were lightweight (approximately 2 g) and, subsequently, low-power (less than 1 mW) transmitters due to the small size of the species.

Manual tracking was undertaken using a Titley Australis 26k very high frequency (VHF) radio receiver system and a Yagi three element hand-held directional antenna (shown earlier in Fig. 6.1). The approximate location of a bird was identified by driving in an offroad-capable vehicle to different sites until a radio signal was audible from the receiver. Once a signal was detected, the tracker continued to point the antenna towards the strongest (loudest) signal while walking through the landscape. This procedure involved constant adjustment of the volume and the gain of the receiver and continued until the bird was sighted. The GPS trajec-

tory followed by the manual tracker was recorded. Both these experiments and those detailed below were performed in an open, grassy, Box Ironbark woodland, and thus relatively easy to traverse on foot. However, locating the birds was often complicated by logistical issues such as limited road accessibility, fence lines and different land tenures (including private property).

After achieving visual confirmation of a bird's location, the UAV was launched approximately from the manual tracker's starting position. The UAV trajectory and raw sensor data were recorded in real-time and later replayed to generate the figures reported. Each flight was performed at a constant altitude of 75 m (such that the canopy was cleared) and each observation took approximately 45 s to complete. For planning viewpoints, the UAV was constrained to choose locations within 300 m of the ground control station (GCS) (i.e., the starting position) for the pilot to maintain visual line-of-sight.

We present an illustration of a trial at Site A in Fig. 6.9. Figure 6.9a shows the launch position at approximately 300 m from the bird in a clearing to allow direct line of sight to the UAV system when flying toward the bird. Power lines in the area restrict the planning distance to 90 m from the home (launch) position, indicated by the 90 m radius arc in Fig. 6.9c. Finally, Fig. 6.9 shows the belief converging to a MAP estimate within 50 m of the bird – the entropy and accuracy of this trial should improve if the UAV could plan a larger radius around the bird (see stationary tag results in Tab. 6.1).

Results for the second and third trials are similar. These results are less exhaustive and quantitative than those with stationary tags due to two reasons: *1)* the inaccuracy of the manual tracking and unpredictable movement of the bird (e.g., if a bird moves between the fourth and fifth observation the manual tracks will be lagged) and *2)* if the bird moves during an observation, the observation gives the incorrect gain pattern. Further, the planning must take into account the environment (e.g., power lines) and allow an unobstructed line of sight to the UAV system for safety precautions, causing difficulty in observing the tag from optimal bearings.

6.9 FIELD TRIALS: CRITICALLY ENDANGERED SWIFT PARROT

The full system was used to validate tracking in a real-world scenario tracking radio-tagged swift parrots. In this section we report on the performance of the system when compared against skilled human trackers. Moreover, we provide detailed heatmaps of posterior locations of the bird and discuss the ecological significance of these trials.

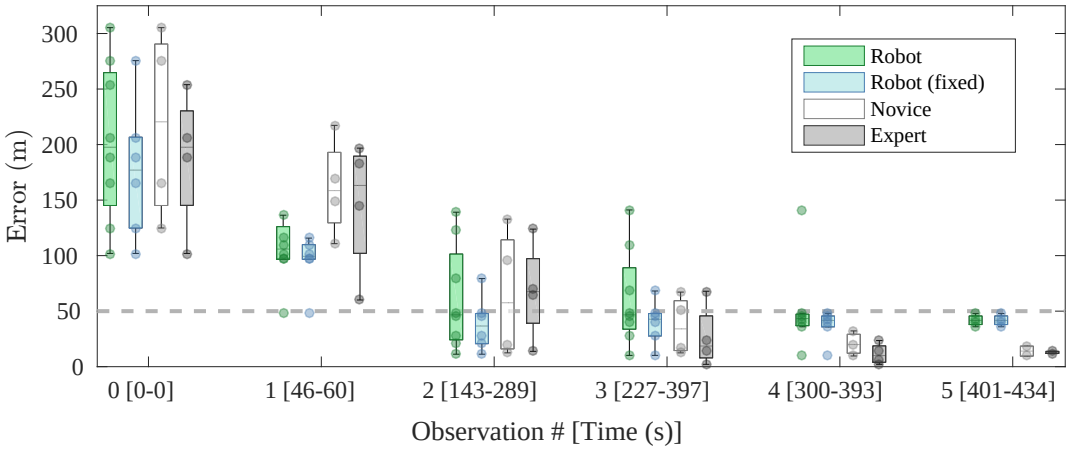


Figure 6.10: Evaluating the performance of the robotic system against human trackers for eight flights at four different sites (two flights per trial). For the robot (green and blue), The box plot shows the error in the system's target estimate $\hat{\boldsymbol{y}}_n$ for each observation n. For the human trackers, we plot the Euclidean distance between the tracker location and the final estimate at the observation time t_n. The blue bar has two flights removed as we were not certain of the final location of the bird (i.e., the target moved during the flight and visual confirmation was lost).

6.9.1 *Experimental Setup*

The trials were performed on 3–7 July 2017 in Temora, New South Wales, Australia. Prior to these trials, six birds were detected by an experienced volunteer undertaking targeted surveys in the surrounding Riverina region, where swift parrots are known to migrate on a regular basis. Follow-up surveys were conducted by the authors in late June, confirming that the survey location was suitable for this trial by detecting at least 30 birds. By the end of August, at least 200 swift parrots (10% of the global population) were detected in the area.

For Bayesian data fusion, the workspace $\mathcal{S} \subset \mathbb{R}^2$ was discretised into a square, 300×300 grid, i.e., $I = 300$ and $J = 300$. Each cell represented a 5×5 m area and thus the workspace extended 750 m in all cardinal directions from the GCS. We assumed a uniform prior on the target location and evolution model covariance $\boldsymbol{\Sigma}_Y = \sigma_Y \boldsymbol{I}_2$, where $\sigma_Y = 20$ m.

6.9.2 *Evaluating the performance of the system*

To validate our approach, we compared the performance of the robotic system against human tracker performance in locating swift parrots in the wild. The results indicate that the robot is able to approximate the location of the target species

in less time than human trackers. Moreover, the reported position estimates are obtained in less than 5 observations (approximately 10 minutes) and lie within 50 m of the true bird location.

The boxplot in Fig. 6.10 collates the tracking performance from eight flights at four different sites. At each site, we obtained the GPS trajectory of a novice and an expert tracker performing manual wildlife telemetry (as described above). Once a human tracker had established the true location $\boldsymbol{y}$ for the target, the UAV began its flight trial. A flight was performed for each type of tracker (novice, expert). Thus, we obtained two tests of the robot system at each site with known true bird location.

For fair comparison, both the robot and the humans began trials from the same initial coordinates, with the target animal location unknown. This starting location was chosen such that the radio signal was strong enough to be measured by the onboard payload. In order to quantify performance, we compare the robotic tracker estimate $\hat{\boldsymbol{y}}_k$ with the Euclidean distance between the human tracker and the final ground truth $\boldsymbol{y}$ at that time t_k. These locations are plotted in Fig. 6.10, where the robot, on average, takes less time to locate the bird to within 50 m (around Observation 2, between 143–289 s).

6.9.3 *Ecological significance of trials*

The quantitative data from our trials provide significant insight into the movement and habitat of swift parrots. The Temora region was chosen because, based on a small number of sightings, it was assumed that numerous swift parrots had migrated to the area in the weeks leading up to the trial (see Sec. 6.9.1). The results in this paper were obtained over a seven-day trial in the region and the posterior estimates from all flights were aggregated to yield the heatmap shown in Fig. 6.11. The figure shows that the flocks used two distinct areas for foraging and roosting, including sites where the species had not previously been recorded.

Swift parrots are small, critically endangered migratory birds that are dependant on highly variable winter nectar resources. As a result, the small population (less than 2000 birds) spreads across vast areas of South-Eastern Australia each year in search of suitable food. Given their small body and hence tag size, as well as their capacity for highly variable and large movements, this species has never before been successfully radio-tracked. Figure 6.11 provides an example of the distribution and abundance of swift parrots across their winter range in a single season, together with the location of our study site.

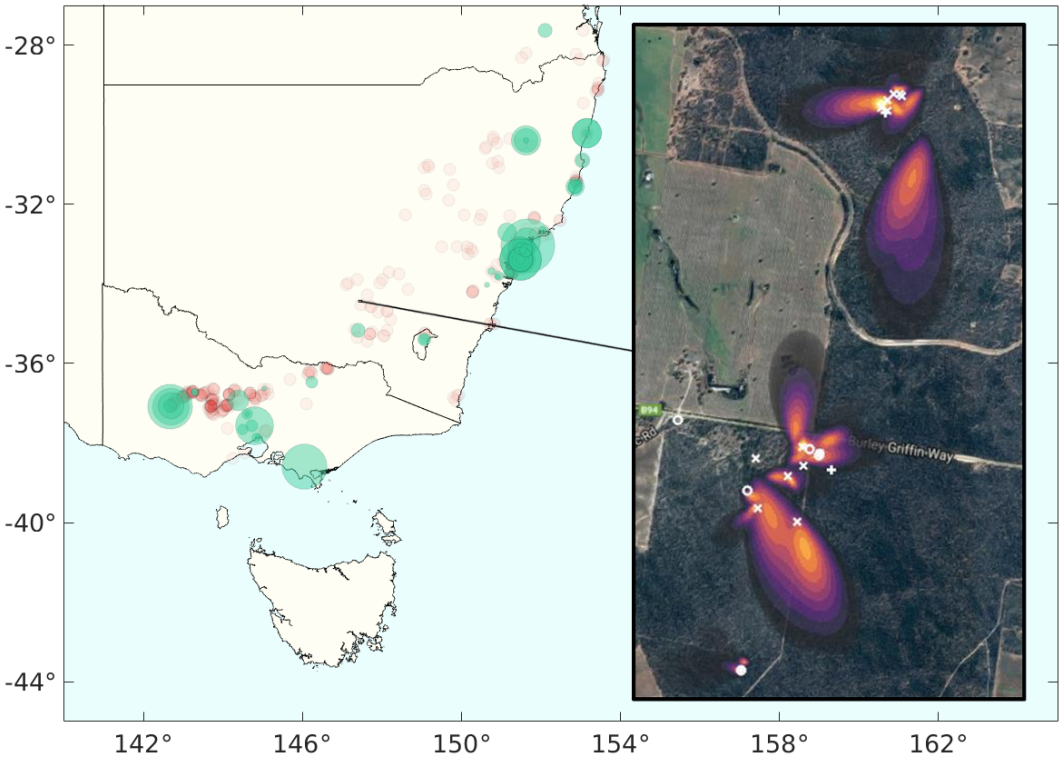


Figure 6.11: Recorded spatial distribution of swift parrots. Main figure: swift parrot sightings in South-East Australia for May 2007. Green circles denote sites where flocks were confirmed, red circles denote failed surveys where no birds were found. The size of the green circles indicate the size of the flock, ranging from 1 to 100 birds. Inset: heatmap illustrating aggregated posterior distributions from our trial in July 2017. The posterior distributions of all trials were normalised and aggregated to give an indication of the most likely communal roosting grounds of the flock. When ground truth data were available for a tag, we plot a white symbol on the map; each unique tag frequency has a unique symbol: 'o', '×', '+', or '∗'.

6.10 SUMMARY

Here we presented and experimentally validated an aerial robot system for tracking small radio-tagged wildlife. The system comprised a novel antenna and custom electronics for achieving high-fidelity recording of the radio signal. The algorithmic contributions included rigorous derivation of a range-azimuth sensor model as well as complete tracking algorithms from first principles. We validated the system in several ways, via: stationary targets, a test bird species (the noisy miner), and the critically endangered swift parrot. The field experiments showed that the system is capable of locating and tracking small dynamic animals in comparable, and occasionally faster, time than experienced human trackers.

Here we considered information-theoretic reasoning from the perspective of an autonomous system, where the main challenge is to optimally gather data rather than model the communication and storage (as was done in the previous two chapters). Due to the nature of the problem, this chapter focused more on sensor modelling and employed a simple greedy informative path planning algorithm. In the following chapter, however, we explore information gathering tasks with a team of robots, where myopic planning algorithms are not suitable.

7 INFORMATION GATHERING WITH TEAMS OF ROBOTS

In the previous chapter, information gathering was used to solve the problem of automating wildlife telemetry tracking. In that context, we were able to assume the travel costs were negligible and, by leveraging the submodularity of entropy, focus on the challenges of sensor modelling. Here, we extend our formulation to consider the more general problem of active perception with a team of robots where the expected travel cost can not be ignored. Although the approach considered here is motivated by multi-robot information gathering tasks it is generally applicable to any utility function.

7.1 OVERVIEW

In this chapter we present Dec-MCTS, a powerful new method of decentralised coordination for any objective function defined over action sequences of a team of robots. Our approach provides convergence guarantees but does not require submodularity assumptions, and is essentially a novel decentralised variant of the MCTS algorithm [38].

At a high level, our method alternates between exploring each robot's individual action space and optimising a probability distribution over the joint-action space. In any particular round, we first use MCTS to find locally favourable sequences of actions for each robot, given a probabilistic estimate of other robots' actions. Then, robots periodically attempt to communicate a highly compressed version of their local search trees which, together, correspond to a product distribution approximation. These communicated distributions are used to estimate the underlying joint distribution. Our method thus inherits important properties from MCTS, such as the ability to compute anytime solutions and to incorporate prior knowledge about the environment. Moreover, it is suitable for online replanning to adapt to changes in the objective function or team behaviour.

This chapter specifically focuses on the extensive theoretical analysis of the algorithm, leveraging results from probability theory and game theory. Practical considerations of the algorithm, as well as experimental validation on multi-robot information gathering tasks is provided in [24]. Our primary analytical result is

to show convergence rates for the expected payoff at the root of the search tree towards the optimal payoff sequence. Thus, the proposed MCTS tree expansion policy balances exploration and exploitation while the reward distributions are changing. This result is proven by extending the MCTS analysis of [126] for the context of switching bandit problems [88]. Our second result leverages the analysis of importance sampling in probability collectives [257] to show that the product distribution optimisation phase locally minimises the KL divergence to the optimal joint probability distribution. While, given the difficulty of the problem, these results do not directly yield guarantees for global optimality, the analysis provides strong motivation for the use of these components in our algorithm for decentralised, long-horizon planning with general objective function definitions.

7.2 PROBLEM STATEMENT

We consider a team of R robots $\{1, 2, \ldots, R\}$, where each robot r plans its own sequence of future actions $\boldsymbol{x}^r = (x_1^r, \ldots)$. Each action x_j^r has an associated cost c_j^r and each robot has a cost budget B^r such that the sum of the costs must be less than the budget, i.e., $\sum_{x_j^r \in \boldsymbol{x}^r} c_j^r \leq B^r$. This cost budget may be an energy or time constraint defined by the application, or it may be used to enforce a planning horizon. The feasible set of actions and associated costs at each step j are a function of the previous actions $(x_1^r, \ldots, x_{j-1}^r)$. Thus, there is a predefined set $\mathcal{X}^r$ of feasible action sequences $\boldsymbol{x}^r$ for each robot r. We denote $\boldsymbol{x}$ as the set of action sequences for all robots $\boldsymbol{x} := \{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^R\}$ and $\boldsymbol{x}^{(r)}$ as the set of action sequences for all robots except robot r, i.e., $\boldsymbol{x}^{(r)} := \boldsymbol{x} \setminus \boldsymbol{x}^r$. We denote $\mathcal{X}$ as the set of all feasible $\boldsymbol{x}$ and $\mathcal{X}^{(r)}$ as the set of all feasible $\boldsymbol{x}^{(r)}$.

The aim is to maximise a global objective function $g(\boldsymbol{x})$ that is a function of the action sequences of all robots. We assume each robot r knows the global objective function g, but does not know the action sequences $\boldsymbol{x}^{(r)}$ selected by the other robots. For most of our proposed approach, we assume g is deterministic given a known set of action sequences $\boldsymbol{x}$; in [24] we discuss extensions for probabilistic objective functions. Moreover, the main applications we consider in [24] are information gathering tasks, however the problem generalises to any form of active perception.

The problem must be solved in a decentralised and online setting. We assume that robots can communicate during planning-time to improve coordination. The communication channel may be unpredictable and intermittent, and all communication is asynchronous. Therefore, each robot will plan based on the information it has available locally. Bandwidth may be constrained and therefore message sizes should remain small, even as the plans grow. Although we do not consider explic-

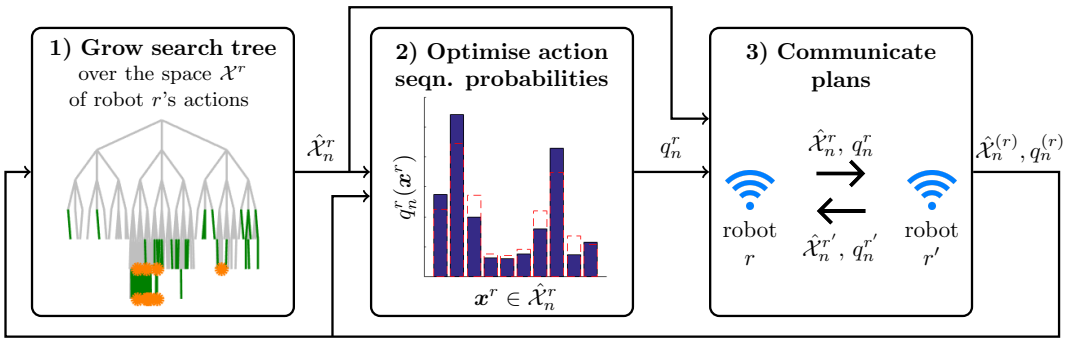


Figure 7.1: Overview of the algorithm running on-board robot r. 1) The search tree is expanded by adding new actions (green). Periodically, the set of best nodes (orange) is selected as the domain $\hat{\mathcal{X}}_n^r$. 2) The probability distribution q_n^r is optimised (from dotted red to solid blue). 3) If possible, the domains and distributions are communicated between robots.

itly planning to maintain communication connectivity, this may be encoded in the objective function $g(\boldsymbol{x})$ if a reliable communication model is available.

7.3 DEC-MCTS

In this section, we give an overview of the Dec-MCTS algorithm as a decentralised solution to the general multi-robot planning problem. We first provide an overview of the algorithm followed by a detailed explanation of all components. For more detail such as pseudocode and practical considerations of the algorithm, refer to [24].

Dec-MCTS runs simultaneously and asynchronously on all robots; we present the algorithm from the perspective of robot r. The algorithm cycles between the three phases illustrated in Fig. 7.1: 1) incrementally grow a search tree using MCTS while taking into account information about the other robots' plans, 2) update the probability distribution over possible action sequences, and 3) communicate probability distributions with the other robots. These three phases continue regardless of whether or not the communication was successful, until a computation budget is met.

A key idea of Dec-MCTS is to represent and reason over plans in a probabilistic manner. In particular, robot r's current plan is represented by a probability distribution over action sequences. We define a probability mass function q_n^r, such that $q_n^r(\boldsymbol{x}^r)$ defines the probability that robot r will select the action sequence $\boldsymbol{x}^r$. In general, the domain of the distribution q_n^r is the set of all possible action sequences $\mathcal{X}^r$. However, to enable tractable computation and realistic communication, we restrict the domain of q_n^r to a dynamically selected subset $\hat{\mathcal{X}}_n^r \subset \mathcal{X}^r$, i.e., $q_n^r(\boldsymbol{x}^r) = 0, \forall \boldsymbol{x}^r \notin \hat{\mathcal{X}}_n^r$. As the Dec-MCTS algorithm progresses, both the domain $\hat{\mathcal{X}}_n^r$

and the probability distribution q_n^r are optimised. Note the subscript n for q_n^r and $\hat{\mathcal{X}}_n^r$ is used to denote the nth iteration of the main loop of our algorithm.

An illustration of the main loop is shown in Fig. 7.1 and pseudocode for the algorithm is provided in [24]. During the MCTS phase, a search tree $\mathcal{T}^r$ is grown over the space $\mathcal{X}^r$ of robot r's action sequences using a new variant of the UCT algorithm. This tree growth is performed while considering the probability distributions over the other robots' plans, denoted $\hat{\mathcal{X}}_n^{(r)}$, $q_n^{(r)}$. Periodically, the domain $\hat{\mathcal{X}}_n^r$ for robot r's distribution is updated by selecting the most promising action sequences identified by the tree search. In the probability distribution optimisation phase, the probabilities assigned to action sequences $q_n^r(\boldsymbol{x}^r)$ are optimised using a decentralised gradient descent algorithm while considering the distributions of the other robots. In the communication phase, robot r communicates its domain $\hat{\mathcal{X}}_n^r$ and probability distribution q_n^r to the other robots. If robot r receives a new distribution from any of the other robots, then in the next iteration $\hat{\mathcal{X}}_n^r$ and q_n^r are optimised while considering this new information. During this optimisation process, it is possible that $q_n^{(r)}$ will change such that a previously optimal leaf of the tree $\mathcal{T}^r$ becomes suboptimal; we refer to the times at which this happens as *breakpoints*.

When the computation budget is met, the algorithm returns the action sequence $\boldsymbol{x}^r$ that has the highest probability $q_n^r(\boldsymbol{x}^r)$. In online settings, the robot would then typically execute the first action x_1^r in the action sequence, and then perform replanning to take into account new information received by observations. If the changes to the objective function are minor, then replanning may be performed more efficiently by adapting the previous search tree.

The global objective function g is optimised by each robot r using a local utility function f^r. We define f^r as the difference in global utility between robot r performing action sequence $\boldsymbol{x}^r$ and a default "no reward" sequence $\boldsymbol{x}_0^r$, assuming fixed action sequences $\boldsymbol{x}^{(r)}$ for the other robots, i.e.,

$$f^r(\boldsymbol{x}) := g(\boldsymbol{x}^r \cup \boldsymbol{x}^{(r)}) - g(\boldsymbol{x}_0^r \cup \boldsymbol{x}^{(r)}). \tag{7.1}$$

The default sequence $\boldsymbol{x}_0^r$ is chosen to be suitable for the application and would typically be an empty action sequence. In practice, optimising with respect to f^r rather than g improves the performance since f^r is more sensitive to robot r's plan and the variance of f^r is less affected by the uncertainty of the other robots' plans [256]. We chose this local utility function since it is generally applicable, although further performance improvements could be achieved with problem-specific heuristics [192].

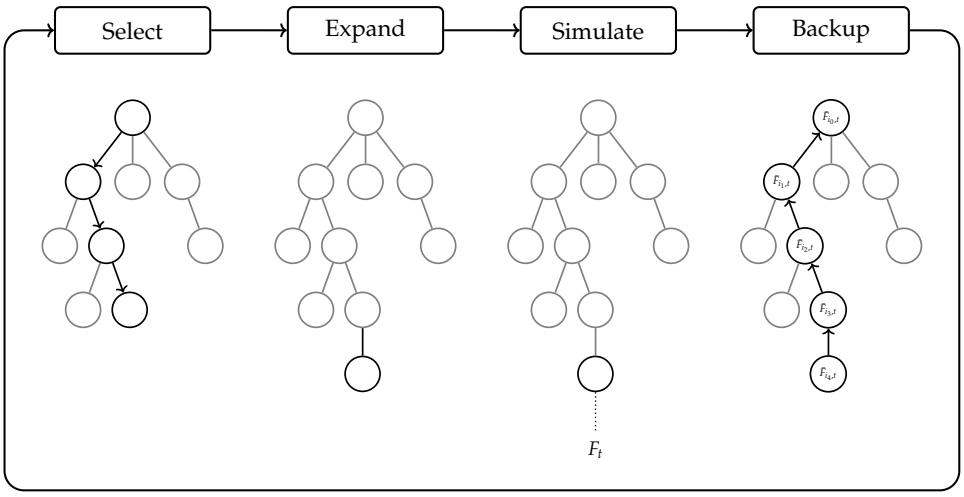


Figure 7.2: The four main stages in the standard MCTS algorithm: node selection, expansion, simulation, and backup.

7.3.1 *Monte Carlo tree search with discounted-UCB*

The first phase of the algorithm is the MCTS update. A single search tree $\mathcal{T}^r$ is maintained by robot r which only contains the actions of robot r. The tree $\mathcal{T}^r$ is defined such that each edge in the tree represents an action by robot r, and a path from the root node i_0 to another node i_d at depth d represents a valid sequence of actions by robot r. The MCTS algorithm incrementally grows $\mathcal{T}^r$ from the root node using a best-first expansion policy. During the MCTS phase, coordination with other robots occurs implicitly by considering the plans of the other robots when performing the rollout policy and evaluation of the global objective function. This information about the other robots' plans comes from the second phase of the algorithm, detailed later in Sec. 7.3.2. In this subsection, we detail our proposed MCTS algorithm which features a novel bandit-based node selection policy designed for our planning scenario.

Standard MCTS incrementally grows a tree by iterating through four phases: *selection*, *expansion*, *simulation* and *backprogation* [38]; this process is illustrated in Fig. 7.2. During each iteration t, a new leaf node is added, where each node represents a sequence of actions and contains statistics about the expected reward of all action sequences that begin with this sequence.

The selection phase selects an expandable node in the tree, where an expandable node is defined as a node that has at least one child that has not yet been visited during the search. In order to find an expandable node, the algorithm begins at the root node i_0 of the tree and recursively selects child nodes until an expandable node i_{d-1} is reached. For selecting the next child at each level of the tree, we propose an extension of the UCT policy [125], detailed in Sec. 7.3.1.1, to balance

exploration and exploitation. In the expansion phase, a new child node i_d is added to the selected expandable node i_{d-1}, which extends the parent's action sequence with an additional action.

In the simulation phase, the expected utility $\mathbf{E}[g(\boldsymbol{X})]$ of the expanded node i_d is estimated by performing and evaluating a rollout policy that extends the action sequence represented by the node until a terminal state is reached. This rollout policy could be a random policy or a heuristic for the problem [110]. The objective is evaluated for this sequence of actions and this result is saved.

For our problem, the objective is a function of the action sequence $\boldsymbol{x}^r$ as well as the unknown plans of the other robots $\boldsymbol{x}^{(r)}$, and thus we require an extension of the standard simulation procedure. To compute the rollout score, we first sample $\boldsymbol{x}^{(r)}$ from a probability distribution $q_n^{(r)}$ over the plans of the other robots. A heuristic rollout policy extended from i_d defines $\boldsymbol{x}^r$, which should be a function of $\boldsymbol{x}^{(r)}$ to simulate coordination between the robots. Additionally, we optimise $\boldsymbol{x}^r$ using the local utility f^r (as defined in (7.1)) rather than g. The rollout score is computed as the utility of this joint sample $f^r(\boldsymbol{x}^r \cup \boldsymbol{x}^{(r)})$, which is an estimate for $\mathbf{E}[f^r(\boldsymbol{X})]$ given the current belief q_n. We denote F_t as the rollout evaluation at sample round t.

In the backpropagation phase, the rollout evaluation F_t is added to the statistics of all nodes along the path from the expanded node back to the root of the tree. Typically, these statistics are unbiased estimators of the rollout evaluations; however, as we discuss in the following section, it is more suitable to use a weighted average in the context of this algorithm.

7.3.1.1 *D-UCB node selection policy*

The node selection policy dictates the order in which the tree $\mathcal{T}^r$ is expanded. Consider an arbitrary node i_d at depth d in the tree which has an associated set of child nodes $\mathrm{Ch}(i_d)$. For every sample round t where node i_d is visited, the problem is to select a child $I_{i_d,t} \in \mathrm{Ch}(i_d)$ that balances both visiting promising subtrees and exploring uncertain ones.

An established approach for node selection is based on maintaining an *upper confidence bound (UCB)* on the value of each node. Under this paradigm, at each sample round t, a UCB U_{j,t_{i_d},t_j} is computed for all children $j \in \mathrm{Ch}(i_d)$ of the parent node i_d. Here, t_{i_d} is the number of times the parent node i_d has been visited and t_j is the number of times child node j has been visited. The algorithm then selects the node that maximises this quantity, i.e.,

$$I_{i_d,t} = \underset{j \in \mathrm{Ch}(i_d)}{\arg\max}\, U_{j,t_{i_d},t_j}. \tag{7.2}$$

This continues recursively until an expandable node is reached.

The *de facto* UCB U_{j,t_{i_d},t_j} is a combination of the empirical mean of rewards received at node j and a confidence interval derived from the Chernoff-Hoeffding inequality [38]. This bound was originally used in the context of the MAB problem and called UCB1 [11]; when used for tree search, it is labelled UCT [125]. UCT was shown to yield polynomial regret when the reward distributions at the leaf nodes are stationary [125]. However, Dec-MCTS alternates between growing the tree for a number of rollouts τ_n and updating the probability distributions for other robots. As mentioned above, this introduces breakpoints as instants where the reward distribution and optimal action can change abruptly. We denote the number of breakpoints up until time t as Υ_t. Due to these breakpoints, the most recent rollouts are more relevant since they are obtained by sampling the most recent distributions. It was shown by Garivier and Moulines [88] that UCB1 is inefficient in the bandit setting when breakpoints are expected. In this scenario a discounted variant (termed D-UCB) yields tighter bounds on regret. Due to the expected breakpoints caused by updating the distributions, we extend the approach of [88] for tree search, and propose a discounted variant of UCT for node selection, which we term D-UCT, described as follows.

Given some discount factor $\gamma \in (1/2, 1)$ and exploration constant $C_p > 1/\sqrt{8}$, the D-UCT bound is defined as:

$$U_{j,t_{i_d},t_j}(\gamma) := \bar{F}_{j,t_j}(\gamma) + c_{t_{i_d},t_j}(\gamma), \tag{7.3}$$

where $\bar{F}_{j,t_j}(\gamma)$ is the discounted empirical reward, and $c_{t_{i_d},t_j}(\gamma)$ is a discounted exploration bonus. A lower discount factor γ enforces only the most recent rollouts to contribute towards the UCB, whereas at the upper limit $\gamma \to 1$ D-UCT becomes equivalent to UCT. These quantities are computed as follows. First, recall that the indicator function $\mathbf{1}_{\{I_{i_d,t}=j\}}$ returns 1 if node j was selected at round t, and 0 otherwise. Then, denote the discounted number of times the child node j has been visited as:

$$t_j(\gamma) := \sum_{u=1}^{t} \gamma^{t-u} \mathbf{1}_{\{I_{i_d,u}=j\}}, \tag{7.4}$$

and the discounted number of times the parent node has been visited as:

$$t_{i_d}(\gamma) := \sum_{j \in \mathrm{Ch}(i_d)} t_j(\gamma). \tag{7.5}$$

Recall that F_t is the rollout score received at sample t. Then, the discounted empirical average is given by:

$$\bar{F}_{j,t_j}(\gamma) := \frac{1}{t_j(\gamma)} \sum_{u=1}^{t} \gamma^{t-u} F_u \mathbf{1}_{\{I_{i_d,u}=j\}}, \tag{7.6}$$

and the discounted exploration bonus is defined as:

$$c_{t_{i_d},t_j}(\gamma) := 2C_p\sqrt{\frac{\log t_{i_d}(\gamma)}{t_j(\gamma)}}. \tag{7.7}$$

The aim of an online planner, such as Dec-MCTS, is to find the best first action, execute this action, and then replan. Thus, we are interested in the convergence of the root node towards selection of the optimal action. Given the expected upper bound on the number of breakpoints occuring in the subtree rooted at node j, i.e., $\mathbf{E}[\mathrm{Y}_{t_j}]$, selecting the discounted factor as

$$\gamma_{t_j} = 1 - \sqrt{\frac{\mathbf{E}[\mathrm{Y}_{t_j}]}{16t_j}} \tag{7.8}$$

allows us to minimise the time for this convergence, as shown in Sec. 7.4. Having γ change dynamically, such as in (7.8), makes it difficult to efficiently recompute $\bar{F}_{j,t_j}$ and $c_{t_{i_d},t_j}$ as t grows large. Therefore, in practice, typically it is best to set γ to a fixed constant.

7.3.2 *Decentralised product distribution optimisation*

The second phase of the algorithm updates a probability distribution q^r over the set of possible action sequences for robot r. These distributions are communicated between robots and used for performing rollouts during MCTS. To optimise these distributions in a decentralised manner for improving global utility, we adapt a type of variational method known as probability collectives [255]. This method can be viewed as a game between independent robots, where each robot selects their action sequence by sampling from a distribution.

One challenge is that the set of possible action sequences is typically of exponential size. We obtain a sparse representation by selecting the sample space $\hat{\mathcal{X}}_n^r \subset \mathcal{X}^r$ as the most promising action sequences $\{\boldsymbol{x}_1^r, \boldsymbol{x}_2^r, ...\}$ found by MCTS. We select a fixed number of nodes with the highest $\mathbf{E}[f^r(\boldsymbol{X})]$ obtained so far. $\hat{\mathcal{X}}_n^r$ is the action sequences used during the initial rollouts when the selected nodes were first expanded.

The set $\hat{\mathcal{X}}_n^r$ has an associated probability distribution q^r such that $q^r(\boldsymbol{x}^i)$ defines the probability that robot r will select $\boldsymbol{x}^r \in \hat{\mathcal{X}}_n^r$. The distributions for different robots are independent and therefore define a product distribution, such that the probability of a joint action sequence selection $\boldsymbol{x}$ is

$$q(\boldsymbol{X} = \boldsymbol{x}) := \prod_r q^r(\boldsymbol{X}^r = \boldsymbol{x}^r). \tag{7.9}$$

The advantage of defining q as a product distribution is so that each robot selects its action sequence independently, and therefore allows decentralised execution.

Consider the general class of joint probability distributions p that are not restricted to product distributions. Define the expected global objective function for a joint distribution p as $\mathbf{E}[G] := \mathbf{E}[g(\boldsymbol{X})]$, and let Γ be a desired value for $\mathbf{E}[G]$. By the principle of maximum entropy, the most likely p that satisfies $\mathbf{E}[G] = \Gamma$ is the p that maximises entropy. This most likely p can be found by minimising the maxent Lagrangian:

$$L(G) := \lambda\left(\Gamma - \mathbf{E}[G]\right) - H(G), \tag{7.10}$$

where $H(G)$ is the Shannon entropy and λ is a Lagrange multiplier. The intuition is to iteratively increase Γ and optimise p. A descent scheme for p can be formulated with Newton's method.

For decentralised planning and execution, we are interested in optimising the product distribution q rather than a more general joint distribution p. We can approximate q by finding the q with the minimum pq KL-divergence $D_{\mathrm{KL}}\left[p \parallel q\right]$. This formulates a descent scheme with the update policy for q^r where we use f^r rather than g. Intuitively, this update rule increases the probability that robot r selects $\boldsymbol{x}^r$ if this results in an improved global utility, while also ensuring the entropy of q^r does not decrease too rapidly. That is, at each iteration, the probability distributions are updated according to:

$$q_n^r(\boldsymbol{x}^r) = q_n^r(\boldsymbol{x}^r) - \alpha q_n^r(\boldsymbol{x}^r)\left[\frac{\mathbb{E}_{q_n}[f^r] - \mathbb{E}_{q_n}[f^r \mid \boldsymbol{x}^r]}{\beta} + \mathrm{H}(q_n^r) + \ln\left(q_n^r(\boldsymbol{x}^r)\right)\right] \tag{7.11}$$

where β is a free parameter that controls the convergence rate of the distributions. Parameter β should slowly decrease and α remain fixed. For efficiency purposes, in our implementation q^r is set to a uniform distribution when $\hat{\mathcal{X}}_n^r$ changes.

7.4 ANALYSIS

In this section, we provide a detailed theoretical analysis of Dec-MCTS. The algorithm is an anytime and decentralised approach to multi-robot coordination with two key algorithmic components: 1) the tree search (Sec. 7.3.1) is designed to perform long-horizon planning for single-robot action sequences while considering the changing plans of the other robots, and 2) the product distribution optimisation (Sec. 7.3.2) is designed to directly optimise the *joint* multi-robot plan while being restricted to a small subset of possible action sequences. While it is difficult to make any strong claims of global optimality in the context of decentralised, long-horizon planning with general objective functions, we focus our analysis on characterising the convergence properties of these two algorithmic components, then discuss the implications of these results.

In Sec. 7.4.1 we begin by presenting and analysing a special type of MAB problem that is related to tree search. Section 7.4.2 then presents our main analytical result that the D-UCT algorithm maintains an exploration-exploitation trade-off for child selection while the distributions q_n^r are changing (and converging). In Sec. 7.4.3 we characterise the convergence of distributing the optimisation process (Sec. 7.3.2) given a contracted sample space of distributions $\hat{\mathcal{X}}_n^r \subset \mathcal{X}^r$. Finally, in Sec. 7.4.4 we remark on the implications of these results in the context of the overall Dec-MCTS algorithm.

7.4.1 *D-UCB applied to bandits*

We begin our analysis by studying D-UCB [88] for a specific type of non-stationary, switching bandit problem. The classic MAB problem is that of a gambler deciding which arm to play from a row of slot machines with stationary but unknown reward distributions. As a result, bandits are the canonical model for studying the trade off of acquiring knowledge ("exploration") and maximising reward ("exploitation"), or the *exploration-exploitation dilemma*. In the context of MCTS, the "arm" is analogous to a node selected to expand for a given MCTS rollout. We can therefore leverage the analysis of the MAB for the tree search problem (later in Sec. 7.4.2). To achieve this, we modify the assumptions on the type of reward distributions for each arm to those expected at internal nodes of the tree while performing the proposed D-UCT algorithm.

Our analysis follows that of [125, 126] who analyse the use of UCB1 as the MCTS node selection policy. We mainly reference the technical report [126] where the proofs for their theorems are given. Specifically, in this section we analyse D-UCB applied to a special type of bandit problem, then in Sec. 7.4.2 we exploit this analysis in applying D-UCT to the root node of a tree.

In the remainder of this section, we will first provide an upper bound on the number of pulls of any arm that is suboptimal in Lemma 7.1. Then, Lemma 7.2 will bound the difference between the optimal payoff and expected total payoff up to some arbitrary time. Lemma 7.3 then gives concentration bounds of the actual mean about this expected value. We then give the asymptotic probability of the algorithm failing in Lemma 7.4.

7.4.1.1 *Technical preliminaries*

We consider D-UCB applied to a particular type of switching bandit problem. Let $I_t(\gamma) \in \{1, \ldots, K\}$ denote the arm pulled at round t, with K the number of possible arms. After selecting node $I_t(\gamma) = i$, the gambler receives a stochastic payoff $F_{i,t} \in$

$[0,1]$. The sequence of payoffs generate the stochastic process $\{F_{i,t}\}_t$, $i = 1,\ldots,K$, $t \geq 1$.

The D-UCB arm selection policy uses the same bound (7.3) as in Sec. 7.3.1.1. Specifically, given a discount factor $\gamma \in (1/2,1)$, the D-UCB algorithm chooses the arm with the best discounted UCB:

$$I_t(\gamma) = \arg\max_{i\in\{1,\ldots,K\}}\{\bar{F}_{i,t}(\gamma) + c_{t,t_i}(\gamma)\}, \tag{7.12}$$

where $\bar{F}_{i,t}(\gamma)$ denotes the discounted average reward (7.6) and $c_{t,t_i}(\gamma)$ is the bias sequence (7.7) for arm i at round t. Similar to Sec. 7.3.1.1, $t_i(\gamma)$ denotes the discounted number of times arm i is pulled (7.4) and $t(\gamma)$ denotes the discounted total number of pulls (7.5).

As in [125], we allow the mean value of the payoffs to drift as a function of time; however, these values can also change dramatically at *breakpoints*. These breakpoints are defined as epochs when a previously suboptimal arm becomes optimal. We denote by Υ_t the number of breakpoints *before* time t. When referring to quantities that are not discounted (i.e., $\gamma = 1$), we remove the γ argument (e.g., $t = t(1)$, $\bar{F}_{i,t} = \bar{F}_{i,t}(1)$, etc.). Further, the filtration $\mathcal{F}$ referred to in this paper is the natural filtration.

Recall we require a number of assumptions on the reward distributions of each arm so that our analysis for this bandit problem can later be exploited in our analysis for D-UCT. Our first assumption relates to the payoff sequence of each arm.

Assumption 7.1. *Fix $1 \leq i \leq K$. Let $\{\mathcal{F}_{i,t}\}_t$ be a filtration such that $\{F_{i,t}\}_t$ is $\{\mathcal{F}_{i,t}\}$-adapted and $F_{i,t}$ is conditionally independent of $\mathcal{F}_{i,t+1}, \mathcal{F}_{i,t+2}, \ldots$ given $\mathcal{F}_{i,t-1}$. Further, there exists an integer T_p such that for $t_i \geq T_p$ and $t < t_i$, $F_{i,t}$ is independent from $\mathcal{F}_{i,t}$.*

As mentioned, we allow the expected value for each arm $\mu_{i,t}$ to drift over time, and change abruptly at a breakpoint. We assume the number of breakpoints are upper bounded as follows.

Assumption 7.2. *The monotone sequence giving the maximum number of breakpoints up to time t $\{\Upsilon_t\}_t$ is known and bounded, s.t., $\lim_{t\to\infty}\Upsilon_t = \sup_t \Upsilon_t < \infty$ and (by definition) $\Upsilon_{t+1} \geq \Upsilon_t$.*

The number of abrupt changes to $\mu_{i,t}$ are thus bounded by $\sup_t \Upsilon_t$. As the following assumption states, we also assume that the expected payoff converges.

Assumption 7.3. *The limit $\mu_i = \lim_{t\to\infty}\mu_{i,t}$ exists for all $i \in \{1,\ldots,K\}$.*

The difference between the expected reward at time t and the limit is termed the drift $\delta_{i,t} = \mu_{i,t} - \mu_i$. For any arbitrary time t, denote the optimal arm as i_t^*, and

define the optimal expected payoff by $\mu_{i_t^*,t} = \max_{i\in\{1,\dots,K\}} \mu_{i,t}$. Thus, we obtain the optimal expected payoff *up to time t* as

$$\mu_t^* = \frac{1}{t}\sum_{u=1}^{t} \mu_{i_u^*,u}. \tag{7.13}$$

Finally, the minimum difference between the expected reward for an optimal arm i_u^* and expected reward for arm i at all times is obtained as

$$\Delta_{i,t} = \min_{u\in\{1,\dots,t\}} \left\{\mu_{i_u^*,u} - \mu_{i,u} : i \neq i_u^*\right\}. \tag{7.14}$$

Our last assumption is that we require an index $T_0(\epsilon)$ above which the drift $\delta_{i,t}$ becomes proportional to $\Delta_{i,t}$. Let $M_i(t)$ denote the number of pulls of arm i following the most recent breakpoint.

Assumption 7.4. *There exists an index $T_0(\epsilon)$ such that, for any arbitrary $\epsilon > 0$ and $M_i(t) \geq T_0(\epsilon)$, $|\delta_{i,t}| \leq \epsilon\Delta_{i,t}/2$ and $|\delta_t^*| \leq \epsilon\Delta_{i,t}/2$ for all i.*

7.4.1.2 *Theoretical analysis*

Given these assumptions, we now begin our analysis of D-UCB for this bandit problem. First, we bound the number of times each suboptimal arm is pulled. Note that in this context, $\mathbf{E}$ is the expectation under the policy D-UCB using the discount factor γ.

Let $\tilde{T}_i(t) = \sum_{u=1}^{t} \mathbf{1}_{\{I_u(\gamma)=i\neq i_u^*\}}$ be the number of times arm i was played when it was not the best arm in the first t rounds.

Lemma 7.1 (Number of Suboptimal Pulls)**.** *Consider D-UCB applied to a non-stationary, switching bandit problem where Assumptions 7.1-7.4 are satisfied and where the bias sequence $c_{t,t_i}(\gamma)$ used by D-UCB is given by (7.7). Let $C_p > 1/\sqrt{8}$ and $\gamma_t = 1 - \sqrt{\mathbf{E}[Y_t]/16t}$. For any arm $i \in \{1,\dots,K\}$ and $t > 1$:*

$$\mathbf{E}[\tilde{T}_i(t)] \leq O\left(\sqrt{\mathbf{E}[Y_t]t}(C_p^2 \log t + T_0(\epsilon) + T_p)\right). \tag{7.15}$$

Proof. We follow the proof of Theorem 1 of [88], with minor modifications to account for the transitory periods T_0, T_p discussed in Theorem 2 of [126]. Note that in order to simplify notation, we substitute temporal functions (e.g., $u_i(\gamma)$ for $t_i(\gamma)$) when the index u is used instead of t.

Fix the index i of a suboptimal arm. Let

$$A_0(t,\epsilon,\gamma) = \min\{t_i(\gamma) \mid c_{t,t_i}(\gamma) \leq (1-\epsilon)\Delta_{i,t}/2\}. \tag{7.16}$$

. Thus, by the definition of $c_{t,t_i}(\gamma)$,

$$A_0(t,\epsilon,\gamma) = \frac{16C_p^2 \log t(\gamma)}{(1-\epsilon)^2\Delta_{i,t}^2}. \tag{7.17}$$

We let $A(t,\epsilon,\gamma) = \max(A_0(t,\epsilon,\gamma), T_0(\epsilon), T_p)$. Then the number of times a suboptimal arm i is played is:

$$\tilde{T}_i(t,\gamma) = 1 + \sum_{u=K+1}^{t} \mathbb{1}_{\{u:(I_u(\gamma)=i\neq i_u^*)\wedge(u_i(\gamma)<A(u,\gamma))\}} + \sum_{u=K+1}^{t} \mathbb{1}_{\{u:(I_u(\gamma)=i\neq i_u^*)\wedge(u_i(\gamma)\geq A(u,\gamma))\}}. \tag{7.18}$$

Further, let

$$D(\gamma) = \frac{\log\Big((1-\gamma)C_p^2 \log\left(K(\gamma)\right)\Big)}{\log(\gamma)}. \tag{7.19}$$

From [88], we have

$$\begin{aligned}\tilde{T}_i(t,\gamma) \leq 1 + \lceil(1-\gamma)t\rceil A(t,\epsilon,\gamma)\gamma^{-1/(1-\gamma)} + \Upsilon_t D(\gamma) \\ + \sum_{u=K+1}^{t} \mathbb{1}_{\{u:(I_u(\gamma)=i\neq i_u^*)\wedge(u_i(\gamma)\geq A(u,\epsilon,\gamma))\}},\end{aligned} \tag{7.20}$$

for any positive $A(t,\epsilon,\gamma)$ and $D(\gamma)$. As in [88], there are three conditions under which a suboptimal arm will be played when $t_i(\gamma) \geq A(t,\epsilon,\gamma)$ (following a breakpoint):

$$\{t : (I_t(\gamma) = i \neq i_t^*) \wedge (t_i(\gamma) \geq A(t,\epsilon,\gamma))\} \subseteq \begin{cases} \{t : (\mu_t^* - \mu_{i,t} < 2c_{t,t_i}(\gamma)) \wedge (t_i(\gamma) \geq A(t,\epsilon,\gamma))\} \\ \cup \left\{t : \bar{F}_t^*(\gamma) \leq \mu_t^* - c_{t,t_{i^*}}(\gamma)\right\} \\ \cup \left\{t : \bar{F}_{i,t}(\gamma) \geq \mu_{i,t} + c_{t,t_i}(\gamma)\right\}. \end{cases}$$

We will start with the first case, following the logic of Theorem 2 in [126]. Since $c_{t,t_i}(\gamma)$ decreases in t_i and $t_i(\gamma) \geq A(t,\epsilon,\gamma) \geq A_0(t,\epsilon,\gamma)$, we have that $c_{t,t_i}(\gamma) \leq c_{t,A_0(t,\epsilon,\gamma)}(\gamma)$ and thus for the choice of $A_0(t,\epsilon,\gamma)$,

$$c_{t,t_i}(\gamma) \leq 2\sqrt{C_p^2 \log t(\gamma)/A_0(t,\epsilon,\gamma)} \leq \Delta_{i,t}/2. \tag{7.21}$$

Thus, the first case can not occur when $t(\gamma) \geq A_0(t,\epsilon,\gamma)$. Now, when $t(\gamma) \geq T_0(\epsilon)$, we have that $|\delta_{i,t}| \leq \epsilon\Delta_{i,t}/2$. Since $\mu^* - \mu_i \geq \Delta_{i,t}, t = 1,2,\ldots$, we have

$$\begin{aligned}\mu_t^* - \mu_{i,t} - 2c_{t,t_i}(\gamma) &\geq \Delta_{i,t} - |\delta_t^*| - \delta_{i,t} - 2c_{t,t_i}(\gamma) \\ &\geq \Delta_{i,t} - \epsilon\Delta_{i,t} - (1-\epsilon)\Delta_{i,t} \\ &= 0.\end{aligned}$$

Thus, the set is empty (i.e., event $(\mu_t^* - \mu_{i,t} < 2c_{t,t_i}(\gamma)) \wedge (t_i(\gamma) \geq A(t,\epsilon,\gamma))$ never occurs).

We now examine the probability of the second and third cases occurring. Recall that $M_i(t)$ denotes the number of pulls of arm i after the most recent breakpoint.

Then, under Assumption 7.1, when $M_i(t) \geq T_p \leq A(t, \epsilon, \gamma)$ we can exploit Theorem 18 of [88] to complete the proof. The probability of poorly estimating the mean payoffs is now upper bounded as [88]

$$p\left(\bar{F}_{i,t}(\gamma) \geq \mu_{i,t} + c_{t,t_i}(\gamma)\right) \leq (1-\gamma)^{-1} - K + \left\lceil \frac{\log \frac{1}{1-\gamma}}{\log(1+\eta)} \right\rceil \frac{(1-\gamma)t}{1-\gamma^{1/(1-\gamma)}} \tag{7.22}$$

for all positive η. Substituting this result into (7.20) and taking expectations of both sides [88]

$$\mathbf{E}[\tilde{T}_i(t,\gamma)] \leq C_1(1-\gamma)t + C_2 \frac{\mathbf{E}[Y_t]}{1-\gamma} \log \frac{1}{1-\gamma}, \tag{7.23}$$

where

$$C_1 = \frac{32\sqrt{2}C_p^2 \log \frac{1}{1-\gamma}}{(1-\epsilon)^2 \Delta_{i,t}^2 \gamma^{1/(1-\gamma)}} + \frac{T_0(\epsilon)}{2\sqrt{2}} + \frac{4}{(1-\frac{1}{e}) \log\left(1 + 4\sqrt{1 - 1/2C_p^2}\right)} \tag{7.24}$$

and

$$C_2 = \frac{\gamma - 1}{\log(1-\gamma)\log\gamma} \times \log(1-\gamma) C_p^2 \log K(\gamma). \tag{7.25}$$

When γ goes to 1, $C_2 \to 1$ and

$$C_1 \to \frac{16eC_p^2 \log \frac{1}{1-\gamma}}{(1-\epsilon)^2 \Delta_{i,t}^2} + T_0(\epsilon) + T_p + \frac{2}{(1-\frac{1}{e}) \log\left(1 + 4\sqrt{1 - 1/2C_p^2}\right)}. \tag{7.26}$$

Finally, we can minimise the expected number of times a suboptimal action is taken by setting the discount factor to $\gamma_t = 1 - \sqrt{\mathbf{E}[Y_t]/16t}$. Selecting this discount factor gives $\mathbf{E}[\tilde{T}_i(t) \mid \gamma] = O\left(\sqrt{\mathbf{E}[Y_t]t}(C_p^2 \log t + T_0(\epsilon) + T_p)\right)$ and thus we obtain the bound (7.15) for $t > 1$. □

Remark 7.1. *A common misconception is that the parameter C_p should be set to $1/\sqrt{2}$ in order to satisfy the Chernoff-Hoeffditg bound [38, 125]. However, in the analysis by [11] and [126], setting C_p to $1/\sqrt{2}$ simply allows the tail inequality to be bounded by t^{-4} and thus converge [11]. Alternatively, we can select any positive C_p to ensure that the tail inequality is bounded by a negative exponent on t. As a result, we leave the value $C_p > 1/\sqrt{8}$.* △

From now on, we assume γ is set as per Lemma 7.1, e.g., $\mathbf{E}[F_t] = \mathbf{E}[F_t(\gamma) \mid \gamma_t = 1 - \sqrt{\mathbf{E}[Y_t]/16t}]$.

The following lemma gives convergence of the expected undiscounted payoff $\mathbf{E}[\bar{F}_t]$ received up to time t towards the optimal payoff μ^*. The proof is a simplified version of Theorem 2 of [126] that allows for changing "best arms". The proof uses the expected number of suboptimal pulls (Lemma 7.1) and the definition of drift δ_t^* to bound the payoff.

Lemma 7.2 (Expected Payoff Convergence Towards Optimal Payoff)**.** *Let*

$$\bar{F}_t := \sum_{i=1}^{K} \frac{T_i(t)}{t} \bar{F}_{i,t}. \tag{7.27}$$

Under the assumptions of Lemma 7.1,

$$|\mathbf{E}[\bar{F}_t] - \mu^*| \leq |\delta_t^*| + O\left(K\sqrt{\mathbf{E}[\mathrm{Y}_t]/t}\left(C_p^2 \log t + T_0 + T_p\right)\right), \tag{7.28}$$

where $T_0 = T_0(1/2)$.

Proof. The proof is a slightly generalised version of Theorem 3 of [126] to allow for switching optimal arms.

Without loss of generality we assume that there is a unique "best arm" at any given time t. We denote the index of this arm by i_t^*. By the triangle inequality, $|\mu^* - \mathbf{E}[\bar{F}_t]| \leq |\mu^* - \mu_t^*| + |\mu_t^* - \mathbf{E}[\bar{F}_t]| = |\delta_t^*| + |\bar{\mu}_t^* - \mathbf{E}[\bar{F}_t]|$. We bound the last term as follows:

$$\begin{aligned} t\,|\mu_t^* - \mathbf{E}[\bar{F}_t]| &= \left|\sum_{u=1}^{t} \mathbf{E}[F_u^*] - \mathbf{E}\left[\sum_{i=1}^{K} T_i(t)\bar{F}_{i,t}\right]\right| \\ &= \left|\sum_{u=1}^{t} \mathbf{E}[F_u^*] - \mathbf{E}\left[T^*(t)\bar{F}_t^*\right]\right| + \mathbf{E}\left[\sum_{i=1}^{K} \tilde{T}_i(t)\bar{F}_{i,t}\right], \end{aligned}$$

since $0 \leq \bar{F}_{i,t} \leq 1$, the last term is bounded by $O(K\sqrt{\mathbf{E}[\mathrm{Y}_t]/t}(C_p^2 \log t + T_0 + T_p))$. Again, since $F_{i,t} \leq 1$, we can deduce that $\mathbf{E}\left[T^*(t)\bar{F}_t^*\right] \leq \mathbf{E}\left[T^*(t)\right] \leq \sum_{u=1}^{t} \mathbf{E}[F_u^*] \leq t$ and upper bound the first term by

$$\begin{aligned} \mathbf{E}[t - T^*(t)] &= \sum_{i=1}^{K} \mathbf{E}\left[\tilde{T}_i(t)\right] \\ &= O(K\sqrt{\mathbf{E}[\mathrm{Y}_t]/t}(C_p^2 \log t + T_0 + T_p)). \end{aligned}$$

Collecting terms yields the bound in (7.28). □

From Lemma 7.2, we have the convergence of the *expected payoff* $\mathbf{E}[\bar{F}_t]$ about the optimal payoff; however, we are yet to obtain results about the concentration of the actual payoff $\bar{F}_t$ about this quantity. To bound this concentration, we leverage the results of [126], which has a non-trivial assumption related to the number of suboptimal pulls. Denote Z_t the indicator variable that a suboptimal arm was pulled. As with [126], from Assumption 7.1, we have that, for $t \geq T_p$, the indicator Z_t is independent of $Z_{t+1}, Z_{t+2}, \ldots$, given $Z_1, \ldots, Z_{t-1}$. Thus, after T_p and T_0, the non-stationary bandit problem becomes equivalent to a stationary problem with high probability. This allows us to establish the concentration of $\mathbf{E}[\bar{F}_t]$ about $\bar{F}_t$ in the following lemma.

Lemma 7.3 (Payoff Convergence Towards Expected Payoff)**.** *Fix an arbitrary* $0 < \varepsilon \leq 1$ *and let* $\Gamma_t = 9C_p \mathbf{E}[Y_t] t \sqrt{2 \log(2/\varepsilon)}$*. Then, under the assumptions of Lemma 7.1, for*

$$t \geq O\left(K\sqrt{\mathbf{E}[Y_t]t}(C_p^2 \log t + T_0 + T_p)\right), \tag{7.29}$$

the following bound holds:

$$p\left(t|\bar{F}_t - \mathbf{E}[\bar{F}_t]| \geq \Gamma_t\right) \leq \varepsilon. \tag{7.30}$$

Proof. We modify the proof of Theorem 5 of [126] slightly by using Lemma A.1 in Appendix A.2 to account for the bound on $\mathbf{E}[\tilde{T}_i(t)]$.

Using the same notation as [126], Z_t is the indicator variable that a suboptimal arm was pulled at time t. It is important to note that this result follows by letting $t_i \geq T_p$, i.e., we are concerned with the process $\{Z_t\}_{t \geq T_p}$. At this stage, Z_t is independent of $Z_{t+1}, \ldots, Z_t$, given $Z_1, \ldots, Z_{t-1}$ and thus Lemma 11 of [126] holds. Then, following Theorem 5 of [126], it will suffice to prove that there exists a t such that $a_t \leq (2/9)\Gamma_t$ and $R_t \leq (4/9)\Gamma_t$.

By Lemma 7.1,

$$\mathbf{E}[\sum_{u=1}^{t} Z_u] \leq O(K\sqrt{\mathbf{E}[Y_t]/t}(\log t + T_0 + T_p)), \tag{7.31}$$

hence $a_t, R_t = O(K\sqrt{\mathbf{E}[Y_t]/t}(\log t + T_0 + T_p))$. Thus, since $\Gamma_t = O(\mathbf{E}[Y_t]t)$ and $a_t, R_t = O(\sqrt{\mathbf{E}[Y_t]t} \log t)$, the index t exists. □

Next, we are interested in the probability of the algorithm failing. The proof relies on our assumption that the breakpoint sequence is known monotone and bounded, resulting in D-UCB becoming equivalent to UCB1 for large t.

Lemma 7.4 (Convergence of Failure Probability)**.** *Under the assumptions of Lemma 7.1 it holds that*

$$\lim_{t \to \infty} p(I_t(\gamma) \neq i_t^* = i^*) = 0. \tag{7.32}$$

Proof. Since $\lim_{t\to\infty} Y_t = \sup_t Y_t < \infty$ and $\lim_{t\to\infty} \gamma_t = 1$, for large t w.l.o.g. we have a unique "best arm" $i^* = i_t^*$ and the algorithm becomes UCB1 applied to a non-stationary bandit problem. Theorems 4 and 6 of [126] yields the result. □

7.4.2 *D-UCB applied to trees*

We now discuss the application of D-UCB as the node selection policy of MCTS, termed D-UCT. The assumptions we made about the bandit problem in the above

section allows us to analyse convergence of the actual to the optimal payoff sequence at the root node after some transitory period.

Recall that the node selection problem at each node in the tree is equivalent to the bandit problem, however with different assumptions on the payoff received. From the perspective of node i_d, after selecting node $I_{i_d,t} = j$, the tree search further down the tree (e.g., $I_{j,t}$) and subsequent MCTS rollout yield a stochastic payoff $F_{j,t} = F_t \in [0,1]$ which is adapted to $\mathcal{F}_{j,t}$ (Assumption 7.1). As nodes are slowly expanded in the tree search, the expected reward at any node higher up the tree slowly drifts until all nodes are explored in the subtree (Assumption 7.3).

The sequence of payoffs generate the stochastic process $\{F_{j,t}\}_t$, $\forall j \in \mathrm{Ch}(i_d)$ and $t \geq 1$. We simplify the analysis by assuming a constant branching factor K, i.e., $\mathrm{Ch}(i_d) = \{1, \ldots, K\}, \forall i_d$.

Applying the above lemmas to the tree $\mathcal{T}^r$, we require some extra notation. Recall that $\bar{F}_{i_d,t_{i_d}}$ is the empirical mean; it follows that $\bar{F}_{i_0,t_{i_0}}$ is the mean at the root node. Further, let $\mu^*_{i_0}$ denote the optimal expected payoff at the root node and note that $t_{i_0} = t$.

Theorem 7.1. *Consider algorithm D-UCT running on a tree $\mathcal{T}$ of depth D and branching factor K. The payoff distributions of the leaf nodes are independently distributed and can change at breakpoints. The sequence that gives the expected bound of breakpoints $\{\mathbf{E}[\mathrm{Y}_{t_j}]\}$ follows Assumption 7.2 and $\gamma_{t_j} = 1 - \sqrt{\mathbf{E}[\mathrm{Y}_{t_j}]/16t_j}$ for all nodes j. Then, when $M_{i_0}(t) \geq T_p$ and $M_{i_0}(t) \geq T_0$, the bias of the payoff at the root node,*

$$|\bar{F}_{i_0,t_{i_0}} - \mu^*_{i_0}| = O\left(KD\log(t)\sqrt{\mathbf{E}[\mathrm{Y}_t]/t}\right). \tag{7.33}$$

Further, the probability of failure at the root node becomes zero as t grows large.

Proof. The proof is done by induction on D.

First, for $D = 1$, running the D-UCB algorithm as the node selection policy is equivalent to running D-UCB on a bandit problem. Thus, the payoffs $F_{i_0,t}$ are i.i.d. ($T_p = 0$) and, comparing Lemma 7.1 to Theorem 1 of [88], we deduce that $T_0 = 0$. The expected bias thus follows from Lemma 7.2, and the concentration of the actual payoff about the expected value follows from Lemma 7.3. By Assumption 7.2, the asymptotic probability of failure follows from Lemma 7.4.

Now, consider a tree of depth D and assume the statement holds up to a tree of depth $D-1$. First, note that, due to our assumptions, Lemmas 7.1-7.4 hold for any internal node of the tree. Regarding Assumption 7.1, before T_p, the payoffs $F_{i_0,t}$ are not independently distributed. Instead, since D-UCB node selection is also used further down the tree ($d > 0$), the payoff is $\{\mathcal{F}_{i_0,t}\}$-adapted. However, there is a point T_p where the payoffs become independent as the tree search problem

becomes equivalent to an MAB problem. When $M_{i_0}(t) \geq T_p$ and $M_{i_0}(t) \geq T_0$, it follows by Lemma 7.2 that the bias at the root converges at the rate of

$$|\bar{F}_{i_0,t_{i_0}} - \mu^*| = |\delta^*_{t_{i_1}}| + O\left(K \log(t) \sqrt{\mathbf{E}[\mathrm{Y}_t]/t}\right), \tag{7.34}$$

where $\delta^*_{t_{i_1}}$ is the rate of convergence of the bias for the best move. By the induction hypothesis

$$|\delta^*_{t_{i_1}}| = O\left(K(D-1) \log(t) \sqrt{\mathbf{E}[\mathrm{Y}_t]/t}\right), \qquad i_1 = 1, \ldots, K. \tag{7.35}$$

Substituting this result into (7.34) gives (7.33). By Assumption 7.2, the expected number of breakpoints $\lim_{t\to\infty} \mathbf{E}[\mathrm{Y}_t]$ is bounded, and hence by Lemma 7.4 the probability of failure becomes zero. □ □

Remark 7.2. *The results presented here are mainly concerned with the convergence of the bias* after some transitory period. *For the standard UCT case, [126] assumed the T_p term was 0 and suggested $T_0 = O(K^D)$. However, it was recently shown that this transitory period using the UCT algorithm on a binary tree ($K = 2$) of depth D can be $\Omega(\exp(\exp(\ldots \exp(1) \ldots)))$ ($D-1$ nested exponentials) in a worst-case instance [60]. [89] suggest instead that the UCT (and thus D-UCT) strategy will be most successful when the leaves of large subtrees share similar rewards, i.e., a "smoothness" assumption on the reward distributions. Active perception scenarios typically exhibit some degree of "smoothness", such that similar sequences of actions yield similar rewards and thus there is a correlation amongst subtree leaves.* △

Remark 7.3. *Assumption 7.2 states several conditions for the breakpoint sequence $\{\mathbf{E}[\mathrm{Y}_t]\}_t$. We can ensure these assumptions are satisfied by selecting appropriate values for the sample period τ_n (i.e., the number of MCTS rollouts between each probability distribution update). Here, we provide a concrete example definition for τ_n. Recall that n is the number of times $\hat{\mathcal{X}}_n$ is changed and τ_n is the number of calls to the MCTS sample step with sample space $\hat{\mathcal{X}}_n$. Let $c > 0$ denote the fixed number of iterations of optimising β. For this example, we let $\tau_n = 1/\lfloor at^{-2} \rfloor$, and therefore $n = \lceil a(1 - t^{-1}) \rceil$ where $a > c$. Therefore, the expected upper bound on breakpoints $\mathbf{E}[\mathrm{Y}_t] = n/c$ and thus $\lim_{t\to\infty} \mathbf{E}[\mathrm{Y}_t] = a/c$. This ensures Lemma 7.4 holds and the bias at the root node (7.33) is*

$$|\bar{F}_{i_0,t_{i_0}} - \mu^*_{i_0}| = O\left(KD \log(t)/\sqrt{t}\right). \tag{7.36}$$

Therefore, D-UCT achieves a polynomial convergence rate, even in problems where the reward distributions are changing abruptly, such as in Dec-MCTS. △

7.4.3 *Variational methods by importance sampling*

We now consider the effect of contracting the sample space $\hat{\mathcal{X}}_n \subset \mathcal{X}$ on the convergence of the distributed optimisation (Sec. 7.3.2). Recall that the pq KL divergence

is the divergence from a product distribution q_n to the optimal joint distribution p_n. We then have the following proposition:

Proposition 7.1. *The joint distributions asymptotically converge to a local minimum, in terms of pq* KL *divergence, given an appropriate subset* $\hat{\mathcal{X}}_n \subset \mathcal{X}$.

We justify Proposition 7.1 as follows. Consider the situation where, at each iteration n, we randomly choose a subset $\hat{\mathcal{X}}_n^r \subset \mathcal{X}^r$ for each robot. This approach is equivalent to Monte Carlo sampling of the expected utility and thus the biased estimator is consistent (asymptotically converges to $\mathbf{E}[f^r]$). For tractable computation and faster convergence, in our algorithm we modify the random selection by choosing a sparse set of strategies $\hat{\mathcal{X}}_n$ with the highest expected utility (Sec. 7.3.2). Although this does not ensure we sample the entire domain $\mathcal{X}$ asymptotically, in practice $q_n(\hat{\mathcal{X}}_n)$ is a reasonably accurate representation of $q_n(\mathcal{X})$, and therefore this gives us an approximation to importance sampling [257]. Variants of probability collectives have been shown to converge to a distribution that locally minimises the *pq* KL divergence under reasonable assumptions, such as an appropriate cooling schedule for β [255].

7.4.4 *Analysis of Dec-MCTS*

The analyses above show separately that the tree search component of Dec-MCTS balances exploration and exploitation and that, under reasonable assumptions, the joint distributions converge to the product distribution that best optimises the joint action sequence. These results provide strong motivation for the use of these components in the algorithm. However, they do not immediately yield a characterisation of optimality for the complete Dec-MCTS algorithm. To prove convergence rates and global optimality, we would need to characterise the co-dependence between the evolution of the reward distributions $\mathbf{E}_{q_n}[f^r \mid \boldsymbol{x}^r]$ used when sampling the tree and the contraction of the sample space $\hat{\mathcal{X}}_n$ used for optimising q_n. This co-dependence is complex due to the cyclic nature of the algorithm and communication of information between robots, and thus it is unlikely that any strong claims for global optimality can be made. However, this is generally not achievable in the context of decentralised, long-horizon planning with general objective functions, as addressed in this paper. Despite this, the experiments presented in [24] show that the Dec-MCTS algorithm converges rapidly to high-quality solutions in multi-robot active perception scenarios.

7.5 SUMMARY

This chapter presented Dec-MCTS, a decentralised planning algorithm designed for multi-robot active perception tasks. Dec-MCTS uses a discounted tree search algorithm (to explore the potential trajectories of each robot) combined with a distributed optimisation technique (to optimise the joint trajectory of the whole team). The tree search algorithm employs a discounted variant of the well-known UCB policy for node selection in order to account for each robot's objective changing due to communication with the rest of the team. The main contributions here include a proof that the discounted tree search yields logarithmic regret even while these reward distributions are changing. Furthermore, we present a proposition that the distributed optimisation converges when including a nonmyopic planner, suggesting convergence guarantees of the algorithm as a whole. The Dec-MCTS path planning algorithm could be further enhanced by studying the communication between agents (see Chapters 4 and 5) and problem-specific subroutines (see Chapter 6). In the next chapter, we elaborate on this concept and conclude the thesis by summarising its contents and discussing more in-depth connections between each chapter.

8 CONCLUSION

8.1 SUMMARY

This thesis focused on the problem of information-theoretic reasoning in complex adaptive systems. The measures in this framework provide general approaches to studying key primatives of computation such as communication and memory. Thus, information theory is suitable in a wide variety of tasks that require analysing or controlling these component operations. We applied this reasoning in addressing two fundamental decision rules: model selection for nonlinear time series, and planning under uncertainty in robotic information gathering tasks.

The thesis began by studying the information storage and transfer of distributed systems, given multivariate time series data. In particular, we investigated the information processing of multi-agent systems through interaction networks and coherence diagrams using transfer entropy and active information storage. The time series were assumed to comprise univariate components that can be modelled as coupled Markov chains. The approach was exemplified through studies involving simulated RoboCup games, revealing correlations to the team's objective (the scoreline). We then extended our study to include systems that admit partially observable dynamics, i.e., where the system state is hidden and observed only through a filter. Using attractor reconstruction, we showed that transfer entropy can be used in the case of both full and partial observability to infer coupling between components. These algorithms were experimentally verified through inferring the coupling of networks of chaotic attractors.

Following this, we proposed methods for selecting (robotic) sensor locations such that knowledge about the environment is improved. This general problem is known as active perception in robotics, where the interplay between planning and modelling introduces complex feedback loops. Our studies exploited several approaches to improving sensor placement such as robustly handling measurement uncertainty, long horizon (nonmyopic) planning, and teams of robots. First, we considered the problem of wildlife telemetry tracking with an autonomous aerial vehicle. In this case, we were able to track the animal in a small number of observations and so the focus is on sensor modelling, rather than nonmyopic planning or scaling up to multiple robots. For this problem, we presented a system is capable

of localising stationary targets to high precision and tracking radio-tagged animals in the wild. Following this, we studied the more general problem of informative path planning for a team of robots with an arbitrary objective function whereby the cost of the path must be within a given budget. For this case, we presented a novel algorithm that yields convergence guarantees whilst employing efficient distributed optimisation algorithms.

8.2 CONTRIBUTIONS

The thesis made a number of theoretical and practical contributions to the study of autonomous agents, multi-agent systems, and (more generally) distributed computation in complex systems.

8.2.1 *Processing in multi-agent dynamics*

In Chapter 4 we identified interaction networks that link together autonomous agents, using only the observational data and without reconstructing the agents' control logic and internal behaviour. The methodology is not aimed at explicit interactions within a team, but rather at implicit interactions, across teams, that may be spatially long-ranged. The approach for constructing interaction networks used a novel application of information dynamics analysing pair-wise interactions and role-based tactics, exemplified by RoboCup 2D Simulation League games.

The interaction networks were demonstrated with two network sub-types: information-sink and information-source diagrams. In an information-sink diagram every node (every player) has an incoming edge, while in an information-source diagram every node has an outgoing edge. These diagrams represent simplifications to full effective network diagrams, and while they do not reveal the full interaction structure they are significantly more efficient to compute, and highlight the strongest of the interactions. Information-sink and -source diagrams were computed for two experimental set-ups that matched the RoboCup-2014 vice-champion team Gliders [186] against two top-five teams, Cyrus [119] and HELIOS [5]. These results showed, for the first time, a number of asymmetries in the tactical schemes used by the teams. These quantified asymmetries were then used in allocating suitable defensive resources by team Gliders, resulting in a statistically significant performance gain.

The follow-up analysis involved computation of information transfer and storage in order to quantify (relative) responsiveness and rigidity respectively. These notions can be applied to individual agents, tactical roles of agents, and the team

overall. Both measures, relative responsiveness and rigidity, were correlated with the game results, pointing out important couplings in particularly intense interactions across teams, and highlighting the tactical roles and field areas where the game outcomes were mostly decided.

We then examined average role-based multi-agent dynamics across games via novel state-space coherence diagrams which clustered the dynamic processes in an abstract state-space. In our examples, the state-space plots identified several salient features of competing tactical formations, providing a crucial step in classifying the games in terms of tactical behaviour. In general, these diagrams are useful when there is a need to cluster dynamic, rather than static, processes.

8.2.2 *Communication in distributed nonlinear systems*

In Chapter 5, we presented a principled method to learn the structure of nonlinear dynamical networks where the components are coupled via a DAG. We modelled multivariate time series as a general type of distributed processing system, termed a POSGDS, where each subsystem comprises a latent state observed only through a read-out function. We depicted the time evolution of this system as a DBN. Using this model, we drew on methods from Bayesian network structure learning and differential topology literature in order to recover the underlying network structure.

Specifically, we derived scoring functions based on the KL divergence of the candidate DAG from the complete network. By using attractor reconstruction, we illustrated how to compute the AIC and BIC scoring functions for the DBN and that the log-likelihood of the POSGDS can be interpreted in the context of information dynamics. Our main result was that the KL divergence can be decomposed as the difference between stochastic interaction and transfer entropy, two measures that are typically used to quantify the divergence from independence and information transfer between subsystems. As we discussed, however, under certain circumstances it suffices to simply use the transfer entropy to quantify the quality of the candidate graph structure.

We then provided the means for turning transfer entropy into a scoring function by penalising it through significance testing. We showed how, using these scores, structure learning can be performed on networks where the observations are assumed to be generated from a parametric distribution (from the exponential family) or some arbitrary nonlinear function. To demonstrate this process, we performed experiments on coupled Lorenz and Rössler systems. The results indicated that this approach is suitable for recovering POSGDSs from data. Further, KL

divergence is related to model encoding, which is a fundamental measure used in complex systems analysis. Our theoretical results, therefore, have potential implications for other areas of research. For example, the notion of equivalence classes in BN structure learning [2] should lend insight into the area of effective network analysis [174, 218] (see Sec. 8.3.1).

8.2.3 *Autonomous wildlife tracking*

In Chapter 6, we presented and validated a customised aerial robot that can be used to perform autonomous wildlife tracking. We demonstrated that only a small number of high quality observations were required to track the animal in the plane. Because of this, the focus of our research was on sensor modelling, rather than path planning.

In order to yield robust measurements, we designed a small lightweight antenna for use on-board the UAV. For measurements and associated uncertainty, we obtained models of the expected RSSI for both range and bearing observations trained from data collected in a representative outdoor environment. We then developed an experimental system that comprised software architecture and custom electronics in order to control the robot autonomously and online.

First, a preliminary system that employed bearing-only observations online was experimentally shown to localise stationary targets to within 30 m and the capable of localising noisy miners. Following this, we used the full range-azimuth sensor model for tracking the critically endangered swift parrot (*Lathamus discolor*) species. In eight field trials, the robot performed comparably and often better than skilled human trackers.

8.2.4 *Decentralised informative path planning*

In Chapter 7, we presented a novel algorithm for decentralised coordination, Dec-MCTS, that is suitable for a general class of problems. The problem formulation was more generally applicable than the wildlife tracking scenario above, where the algorithm admits any general objective function and allows for multiple robots to be coordinated concurrently. In that chapter, our emphasis was on the theoretical aspects of the algorithm, i.e., convergence proofs, rather than experimental validation or problem-specific sensor modelling (which is given in detail in [24]).

A key conceptual feature of this approach is its generality in representing joint action sequences probabilistically, rather than deterministically. Dec-MCTS has the ability to efficiently plan over long planning horizons, computes anytime solutions,

allows incorporating prior knowledge, and provides convergence rate guarantees. Our main contribution to this work was a proof that the nonmyopic planning algorithm, a discounted variant of MCTS, converges in the event of changing reward distributions. These distributions are updated through a distributed optimisation technique that we showed converges to a local minimum even in the event of the MCTS algorithm.

8.3 DISCUSSION AND FUTURE WORK

The results we present in this thesis have far-reaching implications in numerous fields that involve nonlinear interactions between information processing systems. Most notably, we illustrate how applications such as decentralised coordination in information gathering tasks face similar challenges to that of studying the information dynamics in distributed computation. In doing so, we aim to have further coalesced the fields of complex systems science and robotic planning under uncertainty.

8.3.1 *Information dynamics in distributed computation*

The information dynamics tools introduced in this thesis are applicable towards analysing several artificial life and biological systems, where an accurate estimation of the information-processing channels can reveal a computational structure underlying the emergence of collective behaviours.

8.3.1.1 *Application to engineered systems*

We believe that the model selection algorithms proposed in Chapters 4 and 5 would be useful not only in studying multi-agent team sports and coupled ODEs, but also in various analyses of distributed dynamics, e.g., decentralised coordination [86, 260]; swarm engineering [160, 247]; and evolutionary robotics [78, 79, 183, 184].

A related direction of future research is to investigate how each tactical role in multi-agent scenarios could correspond to a different relation between rigidity and responsiveness, and relate these to components of the information-theoretic measure of autonomy [22]. Ultimately, the analysis can be extended to include comprehensive tactical planning and decision-making.

More specifically, the interaction diagrams in Chapter 4 were described as efficient approximations to full effective networks (see Sec. 4.5.3). In this thesis, the diagrams were used qualitatively (by exploiting the asymmetry of the Cyrus tactics, see Sec. 4.7.1); however, a precise network of interactions would further allow

for inference over the future movements of the players. In theory, the RoboCup simulations could be executed such that certain "ground truth" networks could be obtained in order to understand the quality of these diagrams. Similarly, in Sec. 4.7.2 the correlations to the scoreline imply that the information dynamics of a team are predictive of the game outcome. This predictability would be of potential importance in designing objective functions for distributed optimisation (such as the problem statement described in Chapter 7).

For swarm engineering, one often relies on understanding the behaviour of biological swarms, related to our study of multi-agent dynamics. It is worth pointing out several related simulation-based studies which have previously used information dynamics to identify leadership within a swarm, e.g. leaderships in pairs of zebrafish [39], and covert leadership in a swarm of robots distinguished by transfer entropy [229]. While a leader is defined as a swarm member that acts upon specific information in addition to what is provided by local interactions [61, 228], a covert leader is treated no differently than others in the swarm, so that leaders and followers interact identically [197]. By contrasting transfer entropies across individuals, the study [229] was able to distinguish the covert leaders from the followers by characterising the covert leaders with a lesser amount of transfer entropy than the followers. Furthermore, perhaps counter-intuitively, the leaders do not share more information with the swarm than the followers. In the context of Chapter 4, the followers may be seen as larger "information sinks" than the leaders, highlighting another potential use of the information sink diagrams. Similar information dynamics measures have also been very recently used to measure pairwise correlations in a biological swarm of soldier crabs [239], finding that in smaller swarms the crabs tend to make decisions based on their own past behaviour, whereas in larger swarms they make decisions based on behaviour of their neighbours rather than their own.

8.3.1.2 *Theoretical extensions*

The approach proposed in Chapter 5 complements explicit Bayesian identification and comparison of state space models. In DCM, and more generally in approximate Bayesian inference, models are identified in terms of their parameters via an optimisation of an approximate posterior density over model parameters with respect to a variational (free energy) bound on log evidence [83]. After these parameters have been identified, this bound can be used directly for model comparison and selection. Interestingly, free energy is derived from the KL divergence between the approximate and true posterior and thus automatically penalises more complex models; however, in Eq. (3.37), these distributions are inverted. It appears

this method is equivalent to the expected log-likelihood approach we presented in the same chapter however this needs to be further investigated. In future work, it would be interesting to explore the relationship between transfer entropy and the variational free energy bound. Specifically, computing an evidence bound directly from the transfer entropy may allow us to avoid the computationally expensive significance testing described in Sec. 5.5 and instead use an approximation to evidence for structure learning.

Multivariate extensions to transfer entropy are known to eliminate redundant pairwise relationships and take into account the influence of confounding relationships in a network (i.e., synergistic effects) [241, 254]. In Chapter 5 we have shown that this intuition holds for distributed dynamical systems when confined to a DAG topology. We conjecture that these methods are also applicable when cyclic dependencies exist within a graph, given any generic observation can be used in reconstructing the dynamics [68]; however, the methods presented are more likely to reveal *one* source in the cycle, rather than *all* information sources due to redundancy.

The theoretical results of Chapter 5 supplements understanding in fields where transfer entropy is commonly employed to study physical processes. Point processes are being increasingly viewed as models for a variety of information processing systems, e.g., as spiking neural trains [217] and adversaries in robotic patrolling models [101]. It was recently shown how transfer entropy can be computed for continuous time point processes [217], allowing for efficient use of the analytical scoring functions (g_{TEA}, g_{AIC}, and g_{BIC}) in a number of contexts. Another intriguing line of research is the physical and thermodynamic interpretation of transfer entropy [185], particularly its relationship to the arrow of time [216]. The notion of endomorphisms as discussed here and time asymmetry of thermodynamics is thus a promising connection that we will explore in the future.

8.3.1.3 *Practical considerations*

There are a number of extensions that should be considered for further practical implementation of the model selection approaches used in Chapters 4 and 5. The most important future directions involve performing more experiments and reducing the computational complexity of applying these frameworks to large networks.

Currently, we assume that we can bound the history κ we take into account for model selection, i.e., that the dimensionality or Markov order of each subsystem is known. However, this is generally infeasible in practice and more general algorithms are desirable to infer the embedding dimension and time delay for

unknown systems. Fortunately, there are numerous techniques to recover these parameters [191, 213].

Any structure learning algorithm has significant computational complexity, as we discuss in Chapter 5. To avoid these issues, in Chapter 4, we presented an efficient approximation to functional network analysis and, in Chapter 5, we studied networks with a small number of nodes. Moreover, the empirical significance testing in these approaches further degrades the algorithmic performance. It would be interesting to investigate analytical methods for significance testing when using the non-parametric density estimators (similar to the analytical regularisation of AIC, BIC, and χ^2-distribution approaches). Moreover, recent results in submodular set function optimisation suggests that greedy optimisation of information-theoretic scoring functions may yield approximately optimal structure [268]. This would ensures that the network is within a bound of optimal whilst reducing the computation burden significantly.

In Chapter 5, we presented the theory on computing scoring functions for exponential families (i.e., the TEA, AIC and BIC scores). In the future, we aim to perform empirical studies using discrete models such as cellular automata, where some rules exhibit complex and chaotic behaviour. Moreover, the information dynamics of these models are already widely studied through measures such as transfer entropy [147].

An important concept to consider in stochastic systems is the convergence of the shadow (reconstructed) manifold to the true manifold [227]. We have implicitly accounted for this phenomena by using CPDs in Chapter 5, however it is important to investigate the property of convergence with different density estimation techniques. In addition, we are interested in the effect of synchrony in these networks and the relationship to previous results for dynamical systems coupled by spanning trees [258]. We conjecture that the approach used here will allow us to derive scoring functions without the assumption of multinomial observations, and thus afford the use of non-parametric density estimators. Parametric techniques, such as learning the parameters of dynamical systems [92, 102], could be considered in place of the posterior approximations.

Finally, the reconstruction theorems Chapter 5 typically make the assumption that the map (or flow) is a diffeomorphism (invertible in time). Thus, given any state, the past and future are uniquely determined and the time delay can be taken positive or negative. In certain cases, however, the time-reversed system is acausal, giving a map that is not time-invertible (an endomorphism). Ideally, we would aim to have methods to infer coupling for both endomorphisms and diffeomorphisms. As discussed in Chapter 3, Takens [233] showed that if the map is an endomorphism, taking the delay vector of temporally *previous* observations forms

an embedding. The generalised theorems in [68, 221, 222], however, were established for diffeomorphisms, rather than endomorphisms; we can only conjecture that taking a delay of past observations (as we have done throughout the chapter) follows for these results.

8.3.2 *Generalising information gathering tasks*

There are many avenues of inquiry for improving the system hardware and decision making algorithms presented in Chapter 6 and 7.

8.3.2.1 *Algorithmic extensions*

In Chapter 6, we consider the problem of wildlife tracking with a single robot, where the travel cost was ignored. Using the Dec-MCTS algorithm, presented in Chapter 7, enables long time-horizon planning with travel costs, which would allow for efficient search and tracking of numerous animals simultaneously. This problem has already been partially addressed by through optimal information gathering algorithms without considering travel cost [243]. However, these are yet to be used in real tracking experiments.

Other extensions, however, should be considered for highly dynamic animals, where we are interested in more fine-scale movement patterns and thus aim to maintain real-time information on the animal's trajectory. In robotics, this general problem is known as *persistent monitoring*, where the problem is to maintain information on an entire, continuous environment, rather than monitoring a small number of discrete features (e.g., birds) [87]. Recent approaches abstract the environment into discrete spatial locations and model the likelihood of observing birds by Poisson processes [264]. Extensions to this model have been made where it is assumed that presence of a robot causes the animals to change their behaviour [101].

In addition to addressing the multi-robot wildlife tracking problem considered above, it would be interesting to apply the general framework of Chapter 7 where standard algorithms already exist for associated single-agent scenarios. Problem-specific single-agent planning algorithms could replace the MCTS component of Dec-MCTS, while still performing the distributed product distribution optimisation phase, in order to provide stronger theoretical guarantees or algorithmic efficiency for special cases. Scenarios where this could be applicable include multi-robot mission monitoring [28], persistent monitoring [6], travelling salesman problem variants [25], collision avoidance [172], and dynamic coverage problems [105]. It would also be worth investigating other MCTS variants, e.g., BRUE [76], as an alternative to the tree search approach.

The problem formulation considered in Chapter 7 is general in that we are interested in planning sequences of actions for each robot to optimise a joint objective function, without requiring assumptions such as submodularity (as is done in Chapter 6). A straightforward extension to our approach would be to adapt the algorithm to address the Dec-POMDP formulation. This could be achieved by generalising the MCTS component of our algorithm to POMCP [210] while still using our proposed D-UCT tree expansion policy. A difficulty would be to efficiently find good-quality solutions while also considering probabilistic transition models and having the search tree branch for both actions and observations.

8.3.2.2 *Practical considerations*

In the tracking experiments, we assumed that the radio tag is initially observable. In future work it is important to consider the case where no tag is initially observable, which introduces a search and detect component to the problem, and the issue of when the operator should move [27, 28], in addition to localisation.

Moreover, we designed a bespoke aerial vehicle suitable in order to meet weight and time constraints. For widespread use, it is important to develop low-cost, off-the-shelf components for radio telemetry with aerial systems [69, 240]. So far, however, systems designed for this purpose are yet to demonstrate autonomous tracking and are often focused on developing radio signal detection algorithms.

8.3.3 *Information dynamics for team coordination*

The approaches we use to study the communication and storage of multi-agent (and, more generally, distributed) systems can be used for optimisation of team coordination. In particular, they provide efficient, non-parametric methods for important open problems in robotics.

A major result of Chapter 4 was that information dynamics correlate to the scoreline; this is exemplified by the state-space coherence diagrams. These results suggest that certain measures could allow for general purpose objective functions to optimise in multi-robot tasks. This concept is beginning to be explored through genetic algorithms [183] and reinforcement learning [161]. However, these approaches consider embodied systems (rather than fully distributed, autonomous systems) and do not jointly optimise information transfer and storage, as is suggested by the coherence diagrams.

In Chapter 7, our approach is lacking a communication-planning algorithm that selects when to communicate and who to communicate to while running Dec-MCTS in scenarios with limited communication bandwidth. A key difficulty here is to

develop a measure of information value of a communication message in the context of improving planning performance. Along this line of inquiry, our measures presented in Chapter 4 allow for quantifying the dynamics of inter-agent dependencies in a team that is optimising a collective goal. This can be supplemented with communication planning algorithms. For instance, dynamic programming formulations similar to how [170] plans the use of navigation hardware to maintain localisation accuracy. Other communication-planning formulations that may be useful here include [115, 141].

Another interesting line of inquiry is to incorporate coalition forming into Dec-MCTS. As formulated, static coalitions of agents can be formed by generalising the product distributions in our framework to be partial joint distributions. The product distribution described in Sec. 7.3.2 would be defined over *groups* of robots rather than individuals. Each group acts jointly, with a single distribution modelling the joint actions of its members, and coordination between groups is conducted as in our algorithm. Just as our current approach corresponds to mean-field methods, this approach maps nicely to generalised mean field inference [259] or region-based variational methods [261], and guarantees from these approaches may be applicable. It would also be interesting to study *dynamic* coalition forming, where the mapping between agents and robots is allowed to change, and to develop convergence guarantees for this case. A key challenge would be to determine which robots' plans are more tightly coupled and therefore would benefit from planning within a coalition. In this case we would aim to employ the techniques presented in Chapter 4.

8.4 concluding remarks

This thesis demonstrated the usefulness of information theory in various areas of science and engineering. We studied topics that, at first glance, appear disconnected, i.e., quantifying distributed computation and robotic information gathering. Through presenting these concepts under the same (decision-theoretic) framework, we showed how information theory can be used as a general tool to handle a variety of challenges. Using this approach, we addressed a number of fundamental problems that occur when analysing and improving artificial and biological systems. This thesis thus supports the long-standing notion that information-theoretic reasoning is a broadly applicable and invaluable tool in scientific discovery.

APPENDIX

A.1 DISTRIBUTED NONLINEAR SYSTEMS

Here, we present the extended results of Table 5.1, 5.2. That is, we give the precision, recall, fallout, and F_1-scores for the eight networks of Lorenz attractors shown in Figure 5.5. These results are given for a number of different sample sizes to illustrate the sample complexity of this problem: $N = 5000$ (Table A.1, Table A.2), $N = 10,000$ (Table A.3, Table A.4), $N = 25,000$ (Table A.5, Table A.6), $N = 50,000$ (Table A.7, Table A.8), and $N = 100,000$ (Table A.9, Table A.10). Each table has results for various p-values (with a p-value of ∞ denoting the maximum likelihood score (5.29)), as well as two different observation noise variances, $\sigma_\psi = 1$ and $\sigma_\psi = 10$.

Table A.1: Classification results for three-node ($M = 3$) networks with $N = 5000$ samples. We present the precision (P), recall (R), fallout (F), and F_1-score for the eight arbitrary topologies of coupled Lorenz systems represented by Figure 5.5.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^1	R	-	-	-	-	-	-	-	-
	F	0.33	0.22	0.33	0.22	0.22	0.33	0.33	0.22
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^2	R	1	0.5	1	0.5	1	0.5	1	0.5
	F	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14
	P	0.67	0.5	0.67	0.5	0.67	0.5	0.67	0.5
	F_1	0.8	0.5	0.8	0.5	0.8	0.5	0.8	0.5
G^3	R	1	0.5	1	1	1	1	1	0.5
	F	0	0	0	0	0	0	0	0
	P	1	1	1	1	1	1	1	1
	F_1	1	0.67	1	1	1	1	1	0.67
G^4	R	1	0	1	1	1	0.5	1	0
	F	0.14	0.43	0.14	0.14	0.14	0.14	0.14	0.43
	P	0.67	0	0.67	0.67	0.67	0.5	0.67	0
	F_1	0.8	-	0.8	0.8	0.8	0.5	0.8	-

Table A.2: Classification results for four-node ($M = 4$) networks with $N = 5000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^5	R	-	-	-	-	-	-	-	-
	F	0.31	0.25	0.31	0.19	0.31	0.25	0.31	0.19
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^6	R	0.67	0.67	0.67	0.33	0.67	0.33	0.67	0
	F	0.15	0.23	0.15	0.23	0.15	0.23	0.15	0.31
	P	0.5	0.4	0.5	0.25	0.5	0.25	0.5	0
	F_1	0.57	0.5	0.57	0.29	0.57	0.29	0.57	-
G^7	R	1	0.25	1	0.25	0.75	0.25	0.75	0.5
	F	0	0.25	0	0.17	0.083	0.25	0.083	0.083
	P	1	0.25	1	0.33	0.75	0.25	0.75	0.67
	F_1	1	0.25	1	0.29	0.75	0.25	0.75	0.57
G^8	R	1	0.25	1	0.5	1	0.75	1	0.25
	F	0	0.25	0	0.083	0	0.083	0	0.25
	P	1	0.25	1	0.67	1	0.75	1	0.25
	F_1	1	0.25	1	0.57	1	0.75	1	0.25

Table A.3: Classification results for three-node ($M = 3$) networks with $N = 10,000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^1	R	-	-	-	-	-	-	-	-
	F	0.22	0.11	0.22	0.11	0.22	0.22	0.22	0.11
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^2	R	1	0.5	1	0.5	1	0.5	1	0.5
	F	0	0.14	0	0.14	0	0.14	0	0.14
	P	1	0.5	1	0.5	1	0.5	1	0.5
	F_1	1	0.5	1	0.5	1	0.5	1	0.5
G^3	R	1	0.5	1	1	1	0	1	0.5
	F	0	0.14	0	0	0	0.29	0	0.14
	P	1	0.5	1	1	1	0	1	0.5
	F_1	1	0.5	1	1	1	-	1	0.5
G^4	R	1	1	1	0.5	1	0.5	1	1
	F	0.14	0.14	0	0	0.14	0.14	0.14	0.14
	P	0.67	0.67	1	1	0.67	0.5	0.67	0.67
	F_1	0.8	0.8	1	0.67	0.8	0.5	0.8	0.8

Table A.4: Classification results for four-node ($M = 4$) networks with $N = 10,000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^5	R	-	-	-	-	-	-	-	-
	F	0.31	0.25	0.31	0.19	0.31	0.19	0.31	0.25
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^6	R	0.67	0.33	0.67	0	1	1	0.67	0.33
	F	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
	P	0.5	0.33	0.5	0	0.6	0.6	0.5	0.33
	F_1	0.57	0.33	0.57	-	0.75	0.75	0.57	0.33
G^7	R	0.75	0.5	1	0.5	1	0.25	0.75	0.5
	F	0.083	0.083	0	0.083	0	0.17	0.083	0.083
	P	0.75	0.67	1	0.67	1	0.33	0.75	0.67
	F_1	0.75	0.57	1	0.57	1	0.29	0.75	0.57
G^8	R	1	0.25	1	0.25	1	0	1	0.25
	F	0	0.17	0	0.17	0	0.25	0	0.17
	P	1	0.33	1	0.33	1	0	1	0.33
	F_1	1	0.29	1	0.29	1	-	1	0.29

Table A.5: Classification results for three-node ($M = 3$) networks with $N = 25,000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^1	R	-	-	-	-	-	-	-	-
	F	0.22	0.11	0.22	0.11	0.22	0.22	0.22	0.11
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^2	R	1	1	1	0.5	1	0.5	1	1
	F	0	0.14	0	0.14	0	0.14	0	0.14
	P	1	0.67	1	0.5	1	0.5	1	0.67
	F_1	1	0.8	1	0.5	1	0.5	1	0.8
G^3	R	1	1	1	0.5	1	1	1	1
	F	0	0	0	0.14	0	0	0	0
	P	1	1	1	0.5	1	1	1	1
	F_1	1	1	1	0.5	1	1	1	1
G^4	R	1	1	1	1	1	0.5	1	1
	F	0	0	0	0	0	0.14	0	0
	P	1	1	1	1	1	0.5	1	1
	F_1	1	1	1	1	1	0.5	1	1

Table A.6: Classification results for four-node ($M = 4$) networks with $N = 25,000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^5	R	-	-	-	-	-	-	-	-
	F	0.31	0.19	0.31	0.19	0.31	0.19	0.31	0.19
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^6	R	1	0.33	1	0.33	1	0.33	1	0.33
	F	0.15	0.15	0.15	0.15	0.15	0.23	0.15	0.15
	P	0.6	0.33	0.6	0.33	0.6	0.25	0.6	0.33
	F_1	0.75	0.33	0.75	0.33	0.75	0.29	0.75	0.33
G^7	R	1	0.5	1	0.75	1	0.75	1	0.5
	F	0	0.17	0	0	0	0	0	0.17
	P	1	0.5	1	1	1	1	1	0.5
	F_1	1	0.5	1	0.86	1	0.86	1	0.5
G^8	R	1	0.75	1	0.75	1	0.75	1	0.75
	F	0	0	0	0	0	0	0	0
	P	1	1	1	1	1	1	1	1
	F_1	1	0.86	1	0.86	1	0.86	1	0.86

Table A.7: Classification results for three-node ($M = 3$) networks with $N = 50,000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^1	R	-	-	-	-	-	-	-	-
	F	0	0.11	0	0	0	0.11	0	0.22
	P	-	0	-	-	-	0	-	0
	F_1	-	-	-	-	-	-	-	-
G^2	R	1	0.5	1	0.5	1	0.5	1	0.5
	F	0	0.14	0	0.14	0	0.14	0	0.14
	P	1	0.5	1	0.5	1	0.5	1	0.5
	F_1	1	0.5	1	0.5	1	0.5	1	0.5
G^3	R	1	1	1	0.5	1	1	1	1
	F	0	0.14	0	0.14	0	0.14	0	0
	P	1	0.67	1	0.5	1	0.67	1	1
	F_1	1	0.8	1	0.5	1	0.8	1	1
G^4	R	1	0.5	1	1	1	0.5	1	1
	F	0	0.14	0	0	0	0.14	0	0
	P	1	0.5	1	1	1	0.5	1	1
	F_1	1	0.5	1	1	1	0.5	1	1

Table A.8: Classification results for four-node ($M = 4$) networks with $N = 50,000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^5	R	-	-	-	-	-	-	-	-
	F	0.19	0.062	0.19	0.19	0.19	0.12	0.19	0.12
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^6	R	1	0.33	1	0	1	0.33	1	0.33
	F	0	0.15	0	0	0	0.23	0.15	0.15
	P	1	0.33	1	-	1	0.25	0.6	0.33
	F_1	1	0.33	1	-	1	0.29	0.75	0.33
G^7	R	1	0.75	1	0.5	1	0.5	1	0.75
	F	0	0	0	0.17	0	0.083	0	0
	P	1	1	1	0.5	1	0.67	1	1
	F_1	1	0.86	1	0.5	1	0.57	1	0.86
G^8	R	1	0.75	1	0.75	1	0.75	1	0.75
	F	0	0	0	0	0	0	0	0
	P	1	1	1	1	1	1	1	1
	F_1	1	0.86	1	0.86	1	0.86	1	0.86

Table A.9: Classification results for three-node ($M = 3$) networks with $N = 100,000$ samples.

		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
Graph		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^1	R	-	-	-	-	-	-	-	-
	F	0	0.22	0	0.11	0	0.22	0	0.11
	P	-	0	-	0	-	0	-	0
	F_1	-	-	-	-	-	-	-	-
G^2	R	1	0.5	1	1	1	1	1	1
	F	0	0.14	0	0	0	0	0	0.14
	P	1	0.5	1	1	1	1	1	0.67
	F_1	1	0.5	1	1	1	1	1	0.8
G^3	R	1	1	1	1	1	1	1	1
	F	0	0	0	0	0	0	0	0
	P	1	1	1	1	1	1	1	1
	F_1	1	1	1	1	1	1	1	1
G^4	R	1	1	1	1	1	1	1	1
	F	0	0	0	0	0	0	0	0
	P	1	1	1	1	1	1	1	1
	F_1	1	1	1	1	1	1	1	1

Table A.10: Classification results for four-node ($M = 4$) networks with $N = 100,000$ samples.

Graph		$p = \infty$		$p = 0.01$		$p = 0.001$		$p = 0.0001$	
		$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$	$\sigma_\psi = 1$	$\sigma_\psi = 10$
G^5	R	-	-	-	-	-	-	-	-
	F	0.19	0.062	0.19	0.062	0.19	0.19	0.19	0.12
	P	0	0	0	0	0	0	0	0
	F_1	-	-	-	-	-	-	-	-
G^6	R	1	0.33	1	0.67	1	0.33	1	0.33
	F	0	0.15	0	0.15	0	0.077	0	0.15
	P	1	0.33	1	0.5	1	0.5	1	0.33
	F_1	1	0.33	1	0.57	1	0.4	1	0.33
G^7	R	1	-	1	-	1	-	1	-
	F	0	-	0	-	0	-	0	-
	P	1	-	1	-	1	-	1	-
	F_1	1	-	1	-	1	-	1	-
G^8	R	1	0.75	1	0.75	1	0.5	1	0.75
	F	0	0	0	0	0	0.083	0	0
	P	1	1	1	1	1	0.67	1	1
	F_1	1	0.86	1	0.86	1	0.57	1	0.86

A.2 DEC-MCTS LEMMAS

Lemma A.1. *Let $Z_i, \mathcal{F}_i, a_i$ be as it Lemma 13 of [126]. Let $\{F_i\}$ be an i.i.d. sequence with mean μ, and $\{Y_i\}$ an $\mathcal{F}_i$-adapted process. We assume that both F_i and Y_i lie it the $[0,1]$ interval. Consider the partial sums*

$$S_t = \sum_{u=1}^{t} (1 - Z_u) F_u + Z_u Y_u.$$

Fix an arbitrary $0 < \varepsilon \leq 1$, let $\Gamma_t = 9\mathbf{E}[Y_t] t \sqrt{2\log(2/\varepsilon)}$ and let

$$R_t = \mathbf{E}\left[\sum_u F_u\right] - \mathbf{E}[S_t].$$

Then for t such that $a_t \leq (2/9)\Gamma_t$ and $|R_t| \leq (4/9)\Gamma_t$,

$$p(|S_t - \mathbf{E}[S_t]| \geq \Gamma_t) \leq \varepsilon \tag{A.1}$$

Proof. The proof follows Lemma 14 of [126].

We will show that $p(S_t - \mathbf{E}[S_t] \geq \Gamma_t) \leq \varepsilon$ as $p(S_t - \mathbf{E}[S_t] \leq \Gamma_t) \leq \varepsilon$ is proved analogously. Let $p = p(S_t \geq \mathbf{E}[S_t] + \Gamma_t)$. We have $S_t = \sum_{u=1}^{t} F_u + \sum_{u=1}^{t} Z_u(Y_u - F_u) \leq \sum_{u=1}^{t} F_u + 2\sum_{u=1}^{t} Z_u$. Therefore

$$p \leq p\left(\sum_{u=1}^{t} F_u + 2\sum_{u=1}^{t} Z_u \geq \mathbf{E}\left[\sum_{u=1}^{t} F_u\right] - R_t + \Gamma_t\right).$$

Using the inequality $\mathbb{I}(A + B \geq \Gamma) \leq \mathbb{I}(A \geq \alpha\Gamma) + \mathbb{I}(B \geq (1-\alpha)\Gamma)$ that holds for any $A, B \geq 0, 0 \leq \alpha \leq 1$ we get

$$p \leq p\left(\sum_{u=1}^{t} F_u \geq \mathbf{E}\left[\sum_{u=1}^{t} F_u\right] + (1/9)\Gamma_t\right) + p\left(2\sum_{u=1}^{t} Z_u \geq (8/9)\Gamma - R_t\right).$$

Using the Hoeffding-Azuma inequality, the first term can be bounded by

$$\begin{aligned} p\left(\sum_{u=1}^{t} F_u \geq \mathbf{E}\left[\sum_{u=1}^{t} F_u\right] + (1/9)\Gamma_t t\right) &\leq \exp\left(-\frac{2(\Gamma_t/9)^2}{t}\right) \\ &= (\varepsilon/2)^{4t} \\ &\leq \varepsilon/2, \end{aligned}$$

for $n \geq 1$ and $0 < \varepsilon < 1$. Since by assumption $|R_t| \leq (4/9)\Gamma$, the second term can be upper bounded by

$$p\left(2\sum_{u=1}^{t} Z_u \geq (4/9)\Gamma_t\right) = p\left(\sum_{u=1}^{t} Z_u \geq (2/9)\Gamma_t\right).$$

By Lemma 13 of [126], this term is bounded by $(\varepsilon/2)^n \leq \varepsilon/2$ for $t \geq 1$ and $0 < \varepsilon < 1$. Collecting terms yields the first inequality (A.1). □

BIBLIOGRAPHY

[1] P. H. Abreu, J. Moura, D. C. Silva, L. P. Reis, and J. Garganta, "Performance analysis in soccer: A cartesian coordinates based approach using RoboCup data," *Soft Comput.*, vol. 16, no. 1, pp. 47–61, 2012.

[2] S. Acid and L. M. de Campos, "Searching for Bayesian network structures in the space of restricted acyclic partially directed graphs," *J. Artif. Intell. Res.*, vol. 18, no. 1, pp. 445–490, 2003.

[3] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Proc. of IEEE ISIT*, 1973, pp. 267–281.

[4] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Autom. Control*, vol. 19, no. 6, pp. 716–723, 1974.

[5] H. Akiyama, T. Nakashima, K. Yamashita, and S. Mifune, "HELIOS2014 team description paper," Tech. Rep., 2014.

[6] S. Alamdari, E. Fata, and S. L. Smith, "Persistent monitoring in discrete environments: Minimizing the maximum weighted latency between observations," *Int. J. Robot. Res.*, vol. 33, no. 1, pp. 138–154, 2014.

[7] C. Amato, "Decision making under uncertainty: Theory and application," in, M. J. Kochenderfer, Ed. MIT Press, Cambridge and London, 2015, ch. Cooperative Decision Making.

[8] A. H.-S. Ang and W. H. Tang, *Probability Concepts in Engineering Planning and Design*. John Wiley & Songs, Inc., 1984, vol. I.

[9] M. S. Arulampalam, S. Maskell, and N. Gordon, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.

[10] M. Asada, H. Kitano, I. Noda, and M. Veloso, "RoboCup: Today and tomorrow – What we have have learned," *Artif. Intell.*, vol. 110, pp. 193–214, 1999.

[11] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, no. 2, pp. 235–256, 2002.

[12] D. Auger, "Multiple tree for partially observable Monte-Carlo tree search," in *Proc. of EvoApplications*, 2011, pp. 53–62.

[13] N. Ay, "Information geometry on complexity and stochastic interaction," *Entropy*, vol. 17, no. 4, pp. 2432–2458, 2015.

[14] N. Ay and T. Wennekers, "Dynamical properties of strongly interacting Markov chains," *Neural Net.*, vol. 16, no. 10, pp. 1483–1497, 2003.

[15] ——, "Temporal infomax leads to almost deterministic dynamical systems," *Neurocomput.*, vol. 52, pp. 461–466, 2003.

[16] R. Bajcsy, "Active perception," *Proc. IEEE*, vol. 76, no. 8, pp. 966–1005, 1988.

[17] R. Bajcsy, Y. Aloimonos, and J. K. Tsotsos, "Revisiting active perception," *Auton. Robots*, vol. 42, no. 2, pp. 177–196, 2017.

[18] L. Barnett, A. B. Barrett, and A. K. Seth, "Granger causality and transfer entropy are equivalent for Gaussian variables," *Phys. Rev. Lett.*, vol. 103, e238701, 2009.

[19] L. Barnett and T. Bossomaier, "Transfer entropy as a log-likelihood ratio," *Phys. Rev. Lett.*, vol. 109, no. 13, p. 138 105, 2012.

[20] H. Bayram, J. Vander Hook, and V. Isler, "Gathering bearing data for target localization," *IEEE Robot. Autom, Lett.*, vol. 1, no. 1, pp. 369–374, 2016.

[21] P. Berthold, M. Kaatz, and U. Querner, "Long-term satellite tracking of white stork (*Ciconia ciconia*) migration: Constancy versus variability," *J. Ornithol.*, vol. 145, no. 4, pp. 356–359, 2004.

[22] N. Bertschinger, E. Olbrich, N. Ay, and J. Jost, "Autonomy: An information-theoretic perspective," *Biosystems*, vol. 91, no. 2, pp. 331–345, 2008.

[23] G. Best, O. M. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Decentralised Monte Carlo tree search for active perception," in *Proc. of WAFR*, 2016.

[24] G. Best, O. M. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Dec-MCTS: Decentralized planning for multi-robot active perception," *Int. J. Robot. Res.*, vol. 38, pp. 316–337, 2–3 2018.

[25] G. Best, J. Faigl, and R. Fitch, "Online planning for multi-robot active perception with self-organising maps," *Auton. Robot.*, vol. 42, no. 4, pp. 715–738, 2018.

[26] G. Best, M. Forrai, R. R. Mettu, and R. Fitch, "Planning-aware communication for decentralised multi-robot coordination," in *Proc. of IEEE ICRA*, 2018.

[27] G. Best, W. Martens, and R. Fitch, "A spatiotemporal optimal stopping problem for mission monitoring with stationary viewpoints," in *Proc. of RSS*, 2015.

[28] ——, "Path planning with spatiotemporal optimal stopping for stochastic mission monitoring," *IEEE Trans. Robot.*, vol. 33, no. 3, pp. 629–646, 2017.

[29] W. Bialek, A. Cavagna, I. Giardina, T. Mora, E. Silvestri, M. Viale, and A. M. Walczak, "Statistical mechanics for natural flocks of birds," *Proc. of Natl. Acad. Sci.*, vol. 109, no. 13, pp. 4786–4791, 2012.

[30] P. Billingsley, *Ergodic Theory and Information*. John Wiley & Sons, Inc., 1965.

[31] B. A. Block, I. D. Jonsen, S. J. Jorgensen, A. J. Winship, S. A. Shaffer, S. J. Bograd, E. L. Hazen, D. G. Foley, G. Breed, A.-L. Harrison, *et al.*, "Tracking apex marine predator movements in a dynamic ocean," *Nature*, vol. 475, no. 7354, pp. 86–90, 2011.

[32] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Phys. Rep.*, vol. 424, no. 4, pp. 175–308, 2006.

[33] R. R. Bouckaert, "Properties of Bayesian belief network learning algorithms," in *Proc. of AUAI UAI*, 1994, pp. 102–109, ISBN: 1-55860-332-8.

[34] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*. John Wiley & Sons, 2015.

[35] S. L. Bressler and A. K. Seth, "Wiener–granger causality: A well established methodology," *Neuroimage*, vol. 58, no. 2, pp. 323–329, 2011.

[36] E. S. Bridge, K. Thorup, M. S. Bowlin, *et al.*, "Technology on the move: Recent and forthcoming innovations for tracking migratory birds," *Biosci.*, vol. 61, no. 9, pp. 689–698, 2011.

[37] G. Brooker, *Introduction to Sensors for Ranging and Imaging*. SciTech Publishing, Inc., 2009.

[38] C. Browne, E. Powley, D. Whitehouse, S. Lucas, P. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of Monte Carlo tree search methods," *IEEE Trans. Comput. Intell. AI in Games*, vol. 4, no. 1, pp. 1–43, 2012.

[39] S. Butail, V. Mwaffo, and M. Porfiri, "Model-free information-theoretic approach to infer leadership in pairs of zebrafish," *Phys. Rev. E*, vol. 93, p. 042 411, 4 2016.

[40] M. Butler, M. Prokopenko, and T. Howard, "Flexible synchronisation within RoboCup environment: A comparative analysis," in *RoboCup 2000: Robot Soccer World Cup IV*, ser. Lecture Notes in Computer Science, P. Stone, T. Balch, and G. Kraetzschmar, Eds., vol. 2019, Springer Berlin Heidelberg, 2001, pp. 119–128.

[41] F. Caballero, L. Merino, I. Maza, and A. Ollero, "A particle filtering method for wireless sensor network localization with an aerial robot beacon," in *Proc. of IEEE ICRA*, 2008, pp. 596–601.

[42] M. Campbell, A. J. Hoane Jr, and F.-h. Hsu, "Deep blue," *Art. Intell.*, vol. 134, no. 1-2, pp. 57–83, 2002.

[43] L. M. de Campos, "A scoring function for learning Bayesian networks based on mutual information and conditional independence tests," *J. Mach. Learn. Res.*, vol. 7, pp. 2149–2187, 2006.

[44] M. Casdagli, S. Eubank, J. D. Farmer, and J. Gibson, "State space reconstruction in the presence of noise," *Physica D.*, vol. 51, no. 1, pp. 52–98, 1991.

[45] D. Chabot and D. M. Bird, "Wildlife research and management methods in the 21st century: Where do unmanned aircraft fit in?" *J. Unmanned Veh. Syst.*, vol. 3, no. 4, pp. 137–155, 2015.

[46] B. Chai, D. B. Walther, D. M. Beck, and L. Fei-Fei, "Exploring functional connectivity of the human brain using multivariate information analysis," in *Advances in Neural Information Processing Systems*, Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, Eds., vol. 22, NIPS Foundation, 2009, pp. 270–278.

[47] B. Charrow, "Information-theoretic active perception for multi-robot teams," PhD thesis, University of Pennsylvania, 2015.

[48] G. M. J. B. Chaslot, M. H. M. Winands, and H. J. van den Herik, "Parallel Monte-Carlo tree search," in *Proc of CG*, 2008, pp. 60–71.

[49] N. Chentanez, A. G. Barto, and S. P. Singh, "Intrinsically motivated reinforcement learning," in *Advances in Neural Information Processing Systems 17*, L. Saul, Y. Weiss, and L. Bottou, Eds., MIT Press, 2004, pp. 1281–1288.

[50] D. M. Chickering, "Learning equivalence classes of Bayesian-network structures," *J. Mach. Learn. Res.*, vol. 2, pp. 445–498, 2002.

[51] D. M. Chickering, "Learning bayesian networks is np-complete," in *Learning from data*, Springer, 1996, pp. 121–130.

[52] O. M. Cliff, M. Prokopenko, and R. Fitch, "An information criterion for inferring coupling in distributed dynamical systems," *Front. Robot. AI*, vol. 3, no. 71, 2016.

[53] O. M. Cliff, R. Fitch, S. Sukkarieh, D. L. Saunders, and R. Heinsohn, "Online localization of radio-tagged wildlife with an autonomous aerial robot system," in *Proc. of RSS*, 2015.

[54] O. M. Cliff, N. Harding, M. Piraveenan, E. Y. Erten, M. Gambhir, and M. Prokopenko, "Investigating spatiotemporal dynamics and synchrony of influenza epidemics in Australia: An agent-based modelling approach," *Simul. Model. Pract. Th.*, vol. 87, pp. 412–431, 2018.

[55] O. M. Cliff, J. T. Lizier, X. R. Wang, P. Wang, O. Obst, and M. Prokopenko, "Towards quantifying interaction networks in a football match," English, in *RoboCup 2013: Robot World Cup XVII*, ser. Lecture Notes in Computer Science, S. Behnke, M. Veloso, A. Visser, and R. Xiong, Eds., vol. 8371, Springer Berlin Heidelberg, 2014, pp. 1–12.

[56] O. M. Cliff, J. T. Lizier, X. R. Wang, P. Wang, O. Obst, and M. Prokopenko, "Quantifying long-range interactions and coherent structure in multi-agent dynamics," *Art. Life*, vol. 23, no. 1, pp. 34–57, 2017.

[57] O. M. Cliff, M. Prokopenko, and R. Fitch, "Minimising the Kullback-Leibler divergence for model selection in distributed nonlinear systems," *Entropy*, vol. 20, no. 2, p. 51, 2018.

[58] O. M. Cliff, D. Saunders, S. Sukkarieh, and R. Fitch, "Robotic ecology: Tracking small animals with an autonomous aerial vehicle," *Science Robot.*, vol. 3, eaat8409, 23 2018.

[59] O. M. Cliff, V. Sintchenko, T. C. Sorrell, K. Vadlamudi, N. McLean, and M. Prokopenko, "Network properties of salmonella epidemics," *Sci. Rep.*, vol. 9, no. 1, p. 6159, 2019.

[60] P. Coquelin and R. Munos, "Bandit algorithms for tree search," in *Proc. of UAI*, 2007, pp. 67–74.

[61] I. D. Couzin, J. Krause, N. R. Franks, and S. A. Levin, "Effective leadership and decision-making in animal groups on the move.," *Nature*, vol. 433, no. 7025, pp. 513–6, 2005.

[62] E. Crosato, L. Jiang, V. Lecheval, J. T. Lizier, X. R. Wang, P. Tichit, G. Theraulaz, and M. Prokopenko, "Informative and misinformative interactions in a school of fish," *Swarm Intell.*, 2017. DOI: 10.1007/s11721-018-0157-x.

[63] J. P. Crutchfield and D. P. Feldman, "Regularities unseen, randomness observed: Levels of entropy convergence," *Chaos*, vol. 13, no. 1, pp. 25–54, 2003.

[64] J. P. Crutchfield and K. Young, "Inferring statistical complexity," *Phys. Rev. Lett.*, vol. 63, no. 2, p. 105, 1989.

[65] K. M. Cuomo and A. V. Oppenheim, "Circuit implementation of synchronized chaos with applications to communications," *Phys. Rev. Lett.*, vol. 71, no. 1, p. 65, 1993.

[66] R. Daly, Q. Shen, and J. S. Aitken, "Learning Bayesian networks: Approaches and issues," *Knowl. Eng. Rev.*, vol. 26, no. 2, pp. 99–157, 2011.

[67] S. Dasgupta, F. Wörgötter, and P. Manoonpong, "Information dynamics based self-adaptive reservoir for delay temporal memory tasks," *Evolving Systems*, vol. 4, no. 4, pp. 235–249, 2013.

[68] E. R. Deyle and G. Sugihara, "Generalized theorems for nonlinear state space reconstruction," *PLOS ONE*, vol. 6, no. 3, e18295, 2011.

[69] G. A. M. Dos Santos, Z. Barnes, E. Lo, B. Ritoper, L. Nishizaki, X. Tejeda, A. Ke, H. Lin, C. Schurgers, A. Lin, *et al.*, "Small unmanned aerial vehicle system for wildlife radio collar tracking," in *Proc. of IEEE MASS*, 2014, pp. 761–766.

[70] J. Duch, J. S. Waitzman, and L. A. N. Amaral, "Quantifying the performance of individual players in a team activity," *PLOS ONE*, vol. 5, no. 6, e10937, 2010.

[71] J. A. Edlund, N. Chaumont, A. Hintze, C. Koch, G. Tononi, and C. Adami, "Integrated information increases with fitness in the evolution of animats," *PLOS Comp. Bio.*, vol. 7, no. 10, e1002236, 2011.

[72] L. Faes, G. Nollo, and A. Porta, "Information-based detection of nonlinear Granger causality in multivariate processes via a nonuniform embedding technique," *Phys. Rev. E*, vol. 83, no. 5, p. 051 112, 2011.

[73] ——, "Non-uniform multivariate embedding to assess the information transfer in cardiovascular and cardiorespiratory variability series," *Comput. Biol. Med.*, vol. 42, no. 3, pp. 290–297, 2012.

[74] L. Faes, A. Porta, G. Rossato, A. Adami, D. Tonon, A. Corica, and G. Nollo, "Investigating the mechanisms of cardiovascular and cerebrovascular regulation in orthostatic syncope through an information decomposition strategy," *Autonomic Neuroscience*, vol. 178, no. 1-2, pp. 76–82, 2013.

[75] D. P. Feldman, C. S. McTague, and J. P. Crutchfield, "The organization of intrinsic computation: Complexity-entropy diagrams and the diversity of natural information processing," *Chaos*, vol. 18, no. 4, p. 043 106, 2008.

[76] Z. Feldman and C. Domshlak, "Simple regret optimization in online planning for Markov decision processes," *J. Artif. Intell. Res.*, vol. 51, pp. 165–205, 2014.

[77] J. Fewell, D. Armbruster, J. Ingraham, A. Petersen, and J. Waters, "Basketball teams as strategic networks," *PLOS ONE*, vol. 7, no. 11, e47445, 2012.

[78] R. Fitch, Stoy, S. Kernbach, R. Nagpal, and W. Shen, "Special issue: Reconfigurable modular robotics," *Robot. Auton. Syst.*, vol. 62, no. 7, pp. 943–1084, 2014.

[79] R. Fitch and Z. J. Butler, "Million module march: Scalable locomotion for large self-reconfiguring robots," *Int. J. Robot. Res.*, vol. 27, no. 3-4, pp. 331–343, 2008.

[80] E. W. Frew, "Observer trajectory generation for target-motion estimation using monocular vision," PhD Thesis, Stanford University, 2003.

[81] N. Friedman, K. Murphy, and S. Russell, "Learning the structure of dynamic probabilistic networks," in *Proc. of AUAI UAI*, 1998, pp. 139–147.

[82] K. J. Friston, "Functional and effective connectivity in neuroimaging: A synthesis," *Hum. Brain Mapp.*, vol. 2, no. 1-2, pp. 56–78, 1994.

[83] K. Friston, J. Kilner, and L. Harrison, "A free energy principle for the brain," *J. Physiol. Paris.*, vol. 100, no. 1, pp. 70–87, 2006.

[84] K. Friston, R. Moran, and A. K. Seth, "Analysing connectivity with Granger causality and dynamic causal modelling," *Curr. Opin. Neurobiol.*, vol. 23, no. 2, pp. 172–178, 2013.

[85] H. Fujisaka and T. Yamada, "Stability theory of synchronized motion in coupled-oscillator systems," *Prog. Theor. Phys.*, vol. 69, no. 1, pp. 32–47, 1983.

[86] S. K. Gan, R. Fitch, and S. Sukkarieh, "Online decentralized information gathering with spatial–temporal constraints," *Auton. Robot.*, vol. 37, no. 1, pp. 1–25, 2014.

[87] S. Garg and N. Ayanian, "Persistent monitoring of stochastic spatio-temporal phenomena with a small team of robots," in *Proc. of RSS*, 2014.

[88] A. Garivier and E. Moulines, "On upper-confidence bound policies for switching bandit problems," in *Proc. of ALT*, 2011, pp. 174–188.

[89] S. Gelly, L. Kocsis, M. Schoenauer, M. Sebag, D. Silver, C. Szepesvári, and O. Teytaud, "The grand challenge of computer go: Monte Carlo tree search and extensions," *Commun. ACM*, vol. 55, no. 3, pp. 106–113, 2012.

[90] A. Gelman and C. R. Shalizi, "Philosophy and the practice of Bayesian statistics," *Br. J. Math. Stat. Psychol.*, vol. 66, no. 1, pp. 8–38, 2013.

[91] Z. Ghahramani, "Learning dynamic Bayesian networks," in *Adaptive Processing of Sequences and Data Structures*, ser. Lecture Notes in Comp. Sci. Vol. 1387, 1998, pp. 168–197.

[92] Z. Ghahramani and S. T. Roweis, "Learning nonlinear dynamical systems using an EM algorithm," in *Advances in Neural Information Processing Systems 11*, MIT Press, 1999, pp. 431–437.

[93] G. Gómez-Herrero, W. Wu, K. Rutanen, M. Soriano, G. Pipa, and R. Vicente, "Assessing coupling dynamics from an ensemble of time series," *Entropy*, vol. 17, no. 4, pp. 1958–1970, 2015.

[94] L. F. Gonzalez, G. A. Montes, E. Puig, S. Johnson, K. Mengersen, and K. J. Gaston, "Unmanned aerial vehicles (UAVs) and artificial intelligence revolutionizing wildlife monitoring and conservation," *Sensors*, vol. 16, no. 1, p. 97, 2016.

[95] C. W. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica*, pp. 424–438, 1969.

[96] A. Gretton, P. Spirtes, and R. E. Tillman, "Nonlinear directed acyclic structure learning with weakly additive noise models," in *Advances in Neural Information Processing Systems 22*, Curran Associates, Inc., 2009, pp. 1847–1855.

[97] B. Grocholsky, "Information-theoretic control of multiple sensor platforms," PhD thesis, The University of Sydney, 2002.

[98] H. Haken, "Analogy between higher instabilities in fluids and lasers," *Phys. Lett. A*, vol. 53, no. 1, pp. 77–78, 1975.

[99] R. He and P. G. Vaidya, "Analysis and synthesis of synchronous periodic and chaotic systems," *Phys. Rev. A*, vol. 46, no. 12, p. 7387, 1992.

[100] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Mach. Learn.*, vol. 20, no. 3, pp. 20–197, 1995.

[101] B. Hefferan, O. M. Cliff, and R. Fitch, "Adversarial patrolling with reactive point processes," in *Proc. of ARAA ACRA*, 2016.

[102] A. Hefny, C. Downey, and G. J. Gordon, "Supervised learning for dynamical system learning," in *Advances in Neural Information Processing Systems 28*, Curran Associates, Inc., 2015, pp. 1963–1971.

[103] G. M. Hoffmann and C. J. Tomlin, "Mobile sensor network control using mutual information methods and particle filters," in *Proc. of IEEE AC*, 2010, pp. 32–47.

[104] G. Hollinger and G. Sukhatme, "Sampling-based robotic information gathering algorithms," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1271–1287, 2014.

[105] W. Hönig and N. Ayanian, "Dynamic multi-target coverage with robotic cameras," in *Proc. of IEEE/RSJ IROS*, 2016, pp. 1871–1878.

[106] J. V. Hook, P. Tokekar, and V. Isler, "Cautious greedy strategy for bearing-only active localization: Analysis and field experiments," *J. Field Robot.*, vol. 31, no. 2, pp. 296–318, 2014.

[107] P. O. Hoyer, D. Janzing, J. M. Mooij, J. Peters, and B. Schölkopf, "Nonlinear causal discovery with additive noise models," in *Advances in Neural Information Processing Systems 21*, Curran Associates, Inc., 2009, pp. 689–696.

[108] N. E. Hussey, S. T. Kessel, K. Aarestrup, S. J. Cooke, P. D. Cowley, A. T. Fisk, R. G. Harcourt, K. N. Holland, S. J. Iverson, J. F. Kocik, *et al.*, "Aquatic animal telemetry: A panoramic window into the underwater world," *Science*, vol. 348, no. 6240, p. 1 255 642, 2015.

[109] R. G. James, N. Barnett, and J. P. Crutchfield, "Information flows? A critique of transfer entropies," *Phys. Rev. Lett.*, vol. 116, no. 23, p. 238 701, 2016.

[110] S. James, G. Konidaris, and B. Rosman, "An analysis of Monte Carlo tree search," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2017.

[111] A. M. Jensen, D. K. Geller, and Y. Chen, "Monte carlo simulation analysis of tagged fish radio tracking performance by swarming unmanned aerial vehicles in fractional order potential fields," *J. Intell. Robotic Syst.*, vol. 74, no. 1-2, pp. 287–307, 2014.

[112] A. Jensen and Y. Chen, "Tracking tagged fish with swarming unmanned aerial vehicles using fractional order potential fields and kalman filtering," in *Proc. of IEEE ICUAS*, 2013, pp. 1144–1149.

[113] K. Kaneko, "Overview of coupled map lattices," *Chaos*, vol. 2, no. 3, pp. 279–282, 1992.

[114] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis*. Cambridge University Press, 2004.

[115] A. Kassir, R. Fitch, and S. Sukkarieh, "Communication-aware information gathering with dynamic information flow," *Int. J. Robot. Res.*, vol. 34, no. 2, pp. 173–200, 2015.

[116] R. Kays, M. C. Crofoot, W. Jetz, and M. Wikelski, "Terrestrial animal tracking as an eye on life and planet," *Science*, vol. 348, no. 6240, aaa2478, 2015.

[117] R. Kays, S. Tilak, M. Crofoot, *et al.*, "Tracking animal location and activity with an automated radio telemetry system in a tropical rainforest," *Comput. J.*, vol. 54, no. 12, pp. 1931–1948, 2011.

[118] A. Kessler, N Batbayar, T Natsagdorj, D Batsuur, and A. Smith, "Satellite telemetry reveals long-distance migration in the Asian great bustard *Otis tarda dybowskii*," *J. Avian Biol.*, vol. 44, no. 4, pp. 311–320, 2013.

[119] R. Khayami, N. Zare, M. Karimi, P. Mahor, A. Afshar, M. S. Najafi, M. Asadi, F. Tekrar, E. Asali, and A. Keshavarzi, "CYRUS 2D simulation team description paper 2014," in *RoboCup 2014 Symposium and Competitions: Team description papers*, 2014.

[120] A. Kijima, K. Yokoyama, H. Shima, and Y. Yamamoto, "Emergence of self-similarity in football dynamics," *Eur. Phys. J. B.*, vol. 87, no. 2, 41, pp. 1–6, 2014.

[121] H. Kitano and M. Asada, "The RoboCup humanoid challenge as the millennium challenge for advanced robotics," *Adv. Robot.*, vol. 13, no. 8, pp. 723–736, 1998.

[122] H. Kitano, M. Tambe, P. Stone, M. Veloso, S. Coradeschi, E. Osawa, H. Matsubara, I. Noda, and M. Asada, "The RoboCup synthetic agent challenge 97," in *RoboCup 1997: Robot Soccer World Cup I*, ser. Lecture Notes in Computer Science, H. Kitano, Ed., vol. 1395, Springer Berlin Heidelberg, 1998, pp. 62–73, ISBN: 3-540-64473-3.

[123] A. T. Klesh, P. T. Kabamba, and A. R. Girard, "Path planning for cooperative time-optimal information collection," in *Proc. of IEEE AC*, 2008, pp. 1991–1996.

[124] L. Kocarev and U Parlitz, "Generalized synchronization, predictability, and equivalence of unidirectionally coupled dynamical systems," *Phys. Rev. Lett.*, vol. 76, no. 11, p. 1816, 1996.

[125] L. Kocsis and C. Szepesvári, "Bandit based Monte-Carlo planning," in *Proc. of ECML*, 2006, pp. 282–293.

[126] L. Kocsis, C. Szepesvári, and J. Willemson, "Improved Monte-Carlo search," University of Tartu, Tech. Rep., 2006.

[127] J. R. Kok, M. T. J. Spaan, and N. A. Vlassis, "Non-communicative multi-robot coordination in dynamic environments," *Robot. Auton. Syst.*, vol. 50, no. 2-3, pp. 99–114, 2005.

[128] D. Koller and N. Friedman, *Probabilistic graphical models: Principles and techniques*. MIT press, 2009.

[129] F. Korner, R. Speck, A. H. Goktogan, and S. Sukkarieh, "Autonomous airborne wildlife tracking using radio signal strength," in *Proc. of IEEE/RSJ IROS*, 2010, pp. 107–112.

[130] L. Kozachenko, L. F. Friston, and N. N Leonenko, "Sample estimate of the entropy of a random vector," *Probl. Peredachi Inf.*, vol. 23, no. 2, pp. 9–16, 1987.

[131] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Phys. Rev. E*, vol. 69, no. 6, p. 066 138, 2004.

[132] A. Krause and C. Guestrin, "Submodularity and its applications in optimized information gathering," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 4, 32:1–32:20, 2011.

[133] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learn. Res.*, vol. 9, pp. 235–284, 2008.

[134] A. J. Kulkarni and K. Tai, "Probability collectives: A multi-agent approach for solving combinatorial optimization problems," *Appl. Soft Comput.*, vol. 10, no. 3, pp. 759–771, 2010.

[135] W. Lam and F. Bacchus, "Learning Bayesian belief networks: An approach based on the MDL principle," *Comp. Intell.*, vol. 10, no. 3, pp. 269–293, 1994.

[136] M. Leonardo, A. M. Jensen, C. Coopmans, M. McKee, and Y. Chen, "A miniature wildlife tracking uav payload system using acoustic biotelemetry," in *Proc. of ASME IDETC/CIE*, 2013, V004T08A056.

[137] R. X. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part III: Measurement models," in *Proc. of SPIE SDPST*, 2001, pp. 423–446.

[138] ——, "Survey of maneuvering target tracking. part I: Dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333–1364, 2003.

[139] ——, "Survey of maneuvering target tracking. part V: Multiple-model methods," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 4, pp. 1255–1321, 2005.

[140] X. S. Liang, "Information flow and causality as rigorous notions *ab initio*," *Phys. Rev. E*, vol. 94, no. 5, p. 052 201, 2016.

[141] M. Lindhé and K. H. Johansson, "Exploiting multipath fading with a mobile robot," *Int. J. Robot. Res.*, vol. 32, no. 12, pp. 1363–1380, 2013.

[142] W. Liu and A. Winfield, "Modelling and optimisation of adaptive foraging in swarm robotic systems," *Int. J. Robot. Res.*, vol. 29, no. 14, pp. 1741–1760, 2010.

[143] Y. Liu and G. Nejat, "Robotic urban search and rescue: A survey from the control perspective," *J. Intell. Robotic Syst.*, vol. 72, no. 2, pp. 147–165, 2013.

[144] J. T. Lizier and M. Prokopenko, "Differentiating information transfer and causal effect," *Eur. Phys. J. B.*, vol. 73, no. 4, pp. 605–615, 2010.

[145] J. T. Lizier, M. Prokopenko, and A. Y. Zomaya, "Information modification and particle collisions in distributed computation," *Chaos*, vol. 20, no. 3, pp. 037 109–13, 2010.

[146] J. T. Lizier and M. Rubinov, "Multivariate construction of effective computational networks from observational data," Max Planck Institute for Mathematics in the Sciences, Preprint 25/2012, 2012.

[147] J. T. Lizier, "The local information dynamics of distributed computation in complex systems," PhD thesis, Univ. Sydney, 2010.

[148] J. T. Lizier, "Jidt: An information-theoretic toolkit for studying the dynamics of complex systems," *Front. Robot. AI*, vol. 1, no. 11, 2014.

[149] J. T. Lizier, "Measuring the dynamics of information processing on a local scale in time and space," in *Directed Information Measures in Neuroscience*, ser. Understanding Complex Systems, Springer, 2014, pp. 161–193.

[150] J. T. Lizier, J. Heinzle, A. Horstmann, J.-D. Haynes, and M. Prokopenko, "Multivariate information-theoretic measures reveal directed information structure and task relevant changes in fMRI connectivity," *J. Comp. Neurosci.*, vol. 30, no. 1, pp. 85–107, 2011.

[151] J. T. Lizier, M. Prokopenko, and A. Y. Zomaya, "Local information transfer as a spatiotemporal filter for complex systems," *Phys. Rev. E*, vol. 77, no. 2, p. 026 110, 2008.

[152] ——, "Coherent information structure in complex computation," *Theory Biosc.*, vol. 131, pp. 193–203, 2012.

[153] ——, "Local measures of information storage in complex distributed computation," *Inf. Sci.*, vol. 208, pp. 39–54, 2012.

[154] A. L. Lloyd, "The coupled logistic map: A simple model for the effects of spatial heterogeneity on population dynamics," *J. Theor. Biol.*, vol. 173, no. 3, pp. 217–230, 1995.

[155] R. D. Lord, F. C. Bellrose, and W. W. Cochran, "Radiotelemetry of the respiration of a flying duck," *Science*, vol. 137, no. 3523, pp. 39–40, 1962.

[156] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, 1963.

[157] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

[158] D. Marinazzo, M. Pellicoro, and S. Stramaglia, "Causal information approach to partial conditioning in multivariate data sets," *Comput. Math. Methods. Med.*, vol. 2012, pp. 303601+, 2012.

[159] D. G. Mayo, *Error and the Growth of Experimental Knowledge*. University of Chicago Press, 1996.

[160] J. M. Miller, X. R. Wang, J. T. Lizier, M. Prokopenko, and L. F. Rossi, "Measuring information dynamics in swarms," in *Guided Self-Organization: Inception*, ser. Emergence, Complexity and Computation, vol. 9, 2014, pp. 343–364.

[161] G. Montúfar, K. Ghazi-Zahedi, and N. Ay, "Information theoretically aided reinforcement learning for embodied agents," *ArXiv preprint arXiv:1605.09735*, 2016.

[162] H. S. Mortveit and C. M. Reidys, "Discrete, sequential dynamical systems," *Discrete Math.*, vol. 226, no. 1, pp. 281–295, 2001.

[163] L. Mota, L. P. Reis, and N. Lau, "Multi-robot coordination using setplays in the middle-size and simulation leagues," *Mechatronics*, vol. 21, no. 2, pp. 434–444, 2011.

[164] K. Murphy, "Dynamic Bayesian Networks: Representation, Inference and Learning," PhD thesis, UC Berkeley, 2002.

[165] R. Nathan, W. M. Getz, E. Revilla, M. Holyoak, R. Kadmon, D. Saltz, and P. E. Smouse, "A movement ecology paradigm for unifying organismal movement research," *Proc. Natl. Acad. Sci. U.S.A*, vol. 105, no. 49, pp. 19 052–19 059, 2008.

[166] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions–I," *Math. Program*, vol. 14, no. 1, pp. 265–294, 1978.

[167] J. Nguyen, N. Lawrance, R. Fitch, and S. Sukkarieh, "Energy-constrained motion planning for information gathering with autonomous aerial soaring," in *Proc. of IEEE ICRA*, 2013, pp. 3825–3831.

[168] I. Noda and P. Stone, "The RoboCup soccer server and CMUnited clients: Implemented infrastructure for MAS research," *Auton. Agents Multi Agent Syst.*, vol. 7, no. 1–2, pp. 101–120, 2003.

[169] N. Noori, A. Renzaglia, J. Vander Hook, and V. Isler, "Constrained probabilistic search for a one-dimensional random walker," *IEEE Trans. Robot.*, vol. 32, no. 2, pp. 261–274, 2016.

[170] P. Ondruska, C. Gurau, L. Marchegiani, C. H. Tong, and I. Posner, "Scheduled perception for energy-efficient path following," in *Proc. of IEEE ICRA*, 2015, pp. 4799–4806.

[171] S. Oppel, A. N. Powell, and D. L. Dickson, "Timing and distance of king eider migration and winter movements," *Condor*, vol. 110, no. 2, pp. 296–305, 2008.

[172] M. Otte and N. Correll, "Any-com multi-robot path-planning: Maximizing collaboration for variable bandwidth," in *Distributed Autonomous Robotic Systems: The 10th International Symposium*, Springer Berlin Heidelberg, 2013, pp. 161–173.

[173] S. Panigada, G. P. Donovan, J.-N. Druon, G. Lauriano, N. Pierantonio, E. Pirotta, M. Zanardelli, A. N. Zerbini, and G. N. di Sciara, "Satellite tagging of mediterranean fin whales: Working towards the identification of critical habitats and the focussing of mitigation measures," *Sci. Rep.*, vol. 7, 2017.

[174] H.-J. Park and K. Friston, "Structural and functional brain networks: From connections to cognition," *Science*, vol. 342, no. 6158, p. 1 238 411, 2013.

[175] T. Patten, M. Zillich, R. Fitch, M. Vincze, and S. Sukkarieh, "Viewpoint evaluation for online 3-D active object classification," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 73–81, 2016.

[176] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 2014.

[177] J. L. Peña and H. Touchette, "A network theory analysis of football strategies," in *Proc. of EPSC*, 2013, pp. 517–528.

[178] J. Peters, D. Janzing, and B. Schölkopf, "Causal inference on discrete data using additive noise models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2436–2450, 2011.

[179] L. Pitonakova, R. Crowder, and S. Bullock, "Information flow principles for plasticity in foraging robot swarms," *Swarm Intell.*, vol. 10, no. 1, pp. 33–63, 2016, ISSN: 1935-3820.

[180] A. Posch and S. Sukkarieh, "UAV based search for a radio tagged animal using particle filters," in *Proc. of ARAA ACRA*, 2009, pp. 107–112.

[181] I. Priede, "A basking shark (cetorhinus maximus) tracked by satellite together with simultaneous remote sensing," *Fish. Res.*, vol. 2, no. 3, pp. 201–216, 1984.

[182] M. Prokopenko, F. Boschetti, and A. J. Ryan, "An information-theoretic primer on complexity, self-organization, and emergence," *Complexity*, vol. 15, no. 1, pp. 11–28, 2009.

[183] M. Prokopenko, V. Gerasimov, and I. Tanev, "Evolving spatiotemporal coordination in a modular robotic system," in *From Animals to Animats*, ser. Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2006, pp. 558–569.

[184] ——, "Measuring spatiotemporal coordination in a modular robotic system," in *Proc. of Alife X*, 2006, pp. 185–191.

[185] M. Prokopenko and I. Einav, "Information thermodynamics of near-equilibrium computation," *Phys. Rev. E*, vol. 91, no. 6, p. 062 143, 2015.

[186] M. Prokopenko and J. T. Lizier, "Transfer entropy and transient limits of computation," *Sci. Rep.*, vol. 4, p. 5394, 2014.

[187] M. Prokopenko, O. Obst, P. Wang, and J. Held, "Gliders2012: Tactics with action-dependent evaluation functions," in *ROBOCUP 2012 Symposium and Competitions: Team Description Papers*, 2012.

[188] M. Prokopenko and P. Wang, "Evaluating team performance at the edge of chaos," in *RoboCup 2003: Robot Soccer World Cup VII*, 2003, pp. 89–101.

[189] ——, "Relating the entropy of joint beliefs to multi-agent coordination," in *RoboCup 2002: Robot Soccer World Cup VI*, 2003, pp. 367–374, ISBN: 3-540-40666-2.

[190] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. of IEEE ICRA, Workshop on Open Source Software*, 2009.

[191] M. Ragwitz and H. Kantz, "Markov models from data by simple nonlinear time series predictors in delay embedding spaces," *Phys. Rev. E*, vol. 65, no. 5, p. 056 201, 2002.

[192] A. Rahmattalabi, J. J. Chung, M. Colby, and K. Tumer, "D++: Structural credit assignment in tightly coupled multiagent domains," in *Proc. of IEEE/RSJ IROS*, 2016, pp. 4424–4429.

[193] I. Rezek, D. S. Leslie, S. Reece, S. J. Roberts, A. Rogers, R. K. Dash, and N. R. Jennings, "On similarities between inference in game theory and machine learning," *J. Artif. Intell. Res.*, vol. 33, pp. 259–283, 2008.

[194] M. Riedmiller, T. Gabel, R. Hafner, and S. Lange, "Reinforcement learning for robot soccer," *Auton. Robot.*, vol. 27, no. 1, pp. 55–73, 2009.

[195] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, 1978.

[196] J. Rodewald, J. Colombi, K. Oyama, and A. Johnson, "Using information-theoretic principles to analyze and evaluate complex adaptive supply network architectures," *Procedia Computer Sci.*, vol. 61, pp. 147–152, 2015.

[197] L. Rossi, X. Han, and C.-C. Shen, "Autonomous navigation of wireless robot swarms with covert leaders," in *Proc. of ICST RoboComm*, 2007.

[198] O. E. Rössler, "An equation for continuous chaos," *Phys. Lett. A*, vol. 57, no. 5, pp. 397–398, 1976.

[199] N. F. Rulkov, M. M. Sushchik, L. S. Tsimring, and H. D. Abarbanel, "Generalized synchronization of chaos in directionally coupled chaotic systems," *Phys. Rev. E*, vol. 51, no. 2, p. 980, 1995.

[200] A. Ryan and J. K. Hedrick, "Particle filter based information-theoretic active sensing," *Robot. Auton. Syst.*, vol. 58, no. 5, pp. 574–584, 2010.

[201] L. Sandoval, "Structure of a global network of financial companies based on transfer entropy," *Entropy*, vol. 16, no. 8, pp. 4443–4482, 2014.

[202] T. Schreiber, "Measuring information transfer," *Phys. Rev. Lett.*, vol. 85, no. 2, pp. 461–464, 2000.

[203] J. Schumacher, T. Wunderle, P. Fries, F. Jäkel, and G. Pipa, "A statistical framework to infer delay and direction of information flow from measurements of complex systems," *Neural Comput.*, vol. 27, no. 8, pp. 1555–1608, 2015.

[204] G. Schwarz, "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, 1978.

[205] C. R. Shalizi, "Methods and techniques of complex systems science: An overview," in *Complex Systems Science in Biomedicine*, Springer, 2006, pp. 33–114.

[206] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, 1948.

[207] ——, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 4, pp. 623–666, 1948.

[208] T. Silander and P. Myllymaki, "A simple approach for finding the globally optimal Bayesian network structure," in *Proc. of AUAI UAI*, 2006, pp. 445–452.

[209] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[210] D. Silver and J. Veness, "Monte-Carlo planning in large POMDPs," in *Advances in Neural Information Processing Systems 23*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, Eds., Curran Associates, Inc., 2010, pp. 2164–2172.

[211] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser, "Efficient informative sensing using multiple robots," *J. Artif. Intell. Res.*, vol. 34, no. 1, pp. 707–755, 2009.

[212] P. Skoglar, J. Nygards, and M. Ulvklo, "Concurrent path and sensor planning for a UAV – towards an information based approach incorporating models of environment and sensor," in *Proc. of IEEE/RSJ IROS*, 2006, pp. 2436–2442.

[213] M. Small and C. K. Tse, "Optimal embedding parameters: A modelling paradigm," *Physica D*, vol. 194, no. 3, pp. 283–296, 2004.

[214] D. A. Smirnov, "Spurious causalities with transfer entropy," *Phys. Rev. E*, vol. 87, no. 4, p. 042 917, 2013.

[215] A. Somani, N. Ye, D. Hsu, and W. S. Lee, "DESPOT: online POMDP planning with regularization," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2013, pp. 1772–1780.

[216] R. E. Spinney, J. T. Lizier, and M. Prokopenko, "Transfer entropy in physical systems and the arrow of time," *Phys. Rev. E*, vol. 94, no. 2, p. 022 135, 2016.

[217] R. E. Spinney, M. Prokopenko, and J. T. Lizier, "Transfer entropy in continuous time, with applications to jump and neural spiking processes," *Phys. Rev. E*, vol. 95, no. 3, p. 032 319, 2017.

[218] O. Sporns, D. R. Chialvo, M. Kaiser, and C. C. Hilgetag, "Organization, development and function of complex brain networks," *Trends Cogn. Sci.*, vol. 8, no. 9, pp. 418–425, 2004.

[219] O. Sporns, *Networks of the brain*. Cambridge, Massachusetts, USA: MIT Press, 2011.

[220] J. Stark, D. S. Broomhead, M. E. Davies, and J Huke, "Takens embedding theorems for forced and stochastic systems," *Nonlinear Anal. Theory Methods Appl.*, vol. 30, no. 9, pp. 5303–5314, 1997.

[221] J. Stark, "Delay embeddings for forced systems. I. Deterministic forcing," *J. Nonlinear Sci.*, vol. 9, no. 3, pp. 255–332, 1999.

[222] J. Stark, D. S. Broomhead, M. E. Davies, and J Huke, "Delay embeddings for forced systems. II. Stochastic forcing," *J. Nonlinear Sci.*, vol. 13, no. 6, pp. 519–577, 2003.

[223] O. Stetter, D. Battaglia, J. Soriano, and T. Geisel, "Model-free reconstruction of excitatory neuronal connectivity from calcium imaging signals," *PLOS Comput. Biol.*, vol. 8, no. 8, e1002653, 2012.

[224] L. D. Stone, R. L. Streit, T. L. Corwin, and K. L. Bell, *Bayesian Multiple Target Tracking*. Artech House, 2013.

[225] P. Stone and M. Veloso, "Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork," *Artif. Intell.*, vol. 110, no. 2, pp. 241–273, 1999.

[226] S. Stramaglia, G.-R. Wu, M. Pellicoro, and D. Marinazzo, "Expanding the transfer entropy to identify information circuits in complex systems," *Phys. Rev. E*, vol. 86, no. 6, e066211, 2012.

[227] G. Sugihara, R. May, H. Ye, H. Hsieh C, E. Deyle, M. Fogarty, and S. Munch, "Detecting causality in complex ecosystems," *Science*, vol. 338, no. 6106, pp. 496–500, 2012.

[228] Y. Sun, L. F. Rossi, H. Luan, and C.-C. Shen, "Modeling and analyzing large swarms with covert leaders," in *Proc. of IEEE SASO*, 2013, pp. 169–178.

[229] Y. Sun, L. F. Rossi, C.-C. Shen, J. Miller, X. R. Wang, J. T. Lizier, M. Prokopenko, and U. Senanayake, "Information transfer in swarms with leaders," in *Proc. of CI*, 2014.

[230] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT Press Cambridge, 1998.

[231] J. Suzuki, "Learning Bayesian belief networks based on the minimum description length principle: Basic properties," *IEICE Trans. Fundamentals*, vol. 82, no. 10, pp. 2237–2245, 1999.

[232] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical Systems and Turbulence*, ser. Lecture Notes in Math. Vol. 898, 1981, pp. 366–381.

[233] ——, "The reconstruction theorem for endomorphisms," *Bull. Braz. Math. Soc.*, vol. 33, no. 2, pp. 231–262, 2002.

[234] M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *J. Mach. Learn. Res.*, vol. 10, no. 1, pp. 1633–1685, 2009.

[235] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. The MIT Press, 2005.

[236] J. Tisdale, Z. Kim, and J. Hedrick, "Autonomous UAV path planning and estimation," *IEEE Robot. Autom. Mag.*, vol. 16, no. 2, pp. 35–42, 2009.

[237] P. Tokekar, D. Bhadauria, A. Studenski, and V. Isler, "A robotic system for monitoring carp in Minnesota lakes," *J. Field Robot.*, vol. 27, no. 6, pp. 779–789, 2010.

[238] P. Tokekar, E. Branson, J. Vander Hook, and V. Isler, "Tracking aquatic invaders: Autonomous robots for monitoring invasive fish," *IEEE Robot. Autom. Mag.*, vol. 20, no. 3, pp. 33–41, 2013.

[239] T. Tomaru, H. Murakami, T. Niizato, Y. Nishiyama, K. Sonoda, T. Moriyama, and Y.-P. Gunji, "Information transfer in a swarm of soldier crabs," *Artif. Life Robot.*, vol. 21, no. 2, pp. 177–180, 2016.

[240] J. A. Tremblay, A. Desrochers, Y. Aubry, P. Pace, and D. M. Bird, "A low-cost technique for radio-tracking wildlife using a small standard unmanned aerial vehicle," *J. Unmanned Veh. Syst.*, no. 0, pp. 1–7, 2017.

[241] V. A. Vakorin, O. A. Krakovska, and A. R. McIntosh, "Confounding effects of indirect connections on causality estimation," *J. Neurosci. Methods*, vol. 184, no. 1, pp. 152–160, 2009.

[242] P. A. Valdes-Sosa, A. Roebroeck, J. Daunizeau, and K. Friston, "Effective connectivity: Influence, causality and biophysical modeling," *Neuroimage*, vol. 58, no. 2, pp. 339–361, 2011.

[243] J. Vander Hook, P. Tokekar, and V. Isler, "Algorithms for cooperative active localization of static targets with mobile bearing sensors under communication constraints," *IEEE Trans. Robot.*, vol. 31, no. 4, pp. 864–876, 2015.

[244] R. Vicente and M. Wibral, "Efficient estimation of information transfer," in *Directed Information Measures in Neuroscience*, ser. Understanding Complex Systems, Springer, 2014, pp. 37–58.

[245] R. Vicente, M. Wibral, M. Lindner, and G. Pipa, "Transfer entropy – a model-free measure of effective connectivity for the neurosciences," *J. Comp. Neurosci.*, vol. 30, no. 1, pp. 45–67, 2011.

[246] L. Vilar, D. Araujo, K. Davids, and Y. Bar-Yam, "Science of winning soccer: Emergent pattern-forming dynamics in association football," *J. Syst. Sci. Complex.*, vol. 26, pp. 73–84, 2013.

[247] X. R. Wang, J. M. Miller, J. T. Lizier, M. Prokopenko, and L. F. Rossi, "Quantifying and tracing information cascades in swarms," *PLOS ONE*, vol. 7, no. 7, e40084, 2012.

[248] S. Whiteson, N. Kohl, R. Miikkulainen, and P. Stone, "Evolving keepaway soccer players through task decomposition," *Mach. Learn.*, vol. 59, no. 1, pp. 5–30, 2005.

[249] M. Wibral, J. Lizier, S. Vögler, V. Priesemann, and R. Galuske, "Local active information storage as a tool to understand distributed neural information processing," *Frontiers in neuroinformatics*, vol. 8, p. 1, 2014.

[250] M. Wibral, B. Rahm, M. Rieder, M. Lindner, R. Vicente, and J. Kaiser, "Transfer entropy in magnetoencephalographic data: Quantifying information flow in cortical and cerebellar networks," *Proc. Biophys. Mol. Bio.*, vol. 105, no. 1-2, pp. 80–97, 2011.

[251] M. Wibral, R. Vicente, and M. Lindner, "Transfer entropy in neuroscience," in *Directed Information Measures in Neuroscience*, M. Wibral, R. Vicente, and J. T. Lizier, Eds., Springer, 2014, pp. 3–36.

[252] M. Wibral, R. Vicente, and J. T. Lizier, *Directed Information Measures in Neuroscience*. Springer, 2014.

[253] N. Wiener and P. Masani, "The prediction theory of multivariate stochastic processes," *Acta Mathematica*, vol. 98, no. 1, pp. 111–150, 1957.

[254] P. L. Williams and R. D. Beer, "Generalized measures of information transfer," *ArXiv preprint arXiv:1102.1507*, 2011.

[255] D. H. Wolpert and S. Bieniawski, "Distributed control by Lagrangian steepest descent," in *Proc. of IEEE CDC*, 2004, pp. 1562–1567.

[256] D. H. Wolpert, S. R. Bieniawski, and D. G. Rajnarayan, "Handbook of statistics 31: Machine learning: Theory and applications," in. Elsevier, 2013, ch. Probability Collectives in Optimization, pp. 61–99.

[257] D. H. Wolpert, C. E. M. Strauss, and D. Rajnarayan, "Advances in distributed optimization using probability collectives," *Adv. Complex Syst.*, vol. 09, no. 04, pp. 383–436, 2006.

[258] C. W. Wu, "Synchronization in networks of nonlinear dynamical systems coupled via a directed graph," *Nonlinearity*, vol. 18, no. 3, p. 1057, 2005.

[259] E. P. Xing, M. I. Jordan, and S. Russell, "Graph partition strategies for generalized mean field inference," in *Proc. of AUAI UAI*, 2004, pp. 602–610.

[260] Z. Xu, R. Fitch, J. P. Underwood, and S. Sukkarieh, "Decentralized coordinated tracking with mixed discrete-continuous decisions," *J. Field Robot.*, vol. 30, no. 5, pp. 717–740, 2013.

[261] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Constructing free-energy approximations and generalized belief propagation algorithms," *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2282–2312, 2005.

[262] C. Yoo, R. Fitch, and S. Sukkarieh, "Probabilistic temporal logic for motion planning with resource threshold constraints," in *Proc. of RSS*, 2012.

[263] ——, "Online task planning and control for aerial robots with fuel constraints in winds," in *Proc. of WAFR*, 2014.

[264] J. Yu, S. Karaman, and D. Rus, "Persistent monitoring of events with stochastic arrivals at multiple stations," *IEEE Trans. Robot.*, vol. 31, no. 3, pp. 521–535, 2015.

[265] C. Zachreson, K. M. Fair, O. M. Cliff, N. Harding, M. Piraveenan, and M. Prokopenko, "Urbanization affects peak timing, prevalence, and bimodality of influenza pandemics in Australia: Results of a census-calibrated model," *Sci. Adv.*, vol. 4, no. 12, eaau5294, 2018.

[266] K. Zahedi, G. Martius, and N. Ay, "Linear combination of one-step predictive information with an external reward in an episodic policy gradient setting: A critical analysis," *Front. Psychol.*, vol. 131, no. 3, 2013.

[267] K. Zhao and R. Jurdak, "Understanding the spatiotemporal pattern of grazing cattle movement," *Sci. Rep.*, vol. 6, p. 31 967, 2016.

[268] Y. Zhou and C. J. Spanos, "Causal meets submodular: Subset selection with directed information," in *Advances in Neural Information Processing Systems 29*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., Curran Associates, Inc., 2016, pp. 2649–2657.