THE UNIVERSITY OF
SYDNEY

Communication Efficiency in Information Gathering through Dynamic Information Flow

Abdallah Kassir, BE (Hons 1)

A thesis submitted in fulfillment
of the requirements for the degree of
Doctor of Philosophy

THE UNIVERSITY OF
SYDNEY

Australian Centre for Field Robotics
School of Aerospace, Mechanical and Mechatronic Engineering
The University of Sydney

October 2014

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the University or other institute of higher learning, except where due acknowledgement has been made in the text.

Abdallah Kassir, BE (Hons 1)

15 October 2014

Abstract

Abdallah Kassir, BE (Hons 1)
The University of Sydney

Doctor of Philosophy
October 2014

Communication Efficiency in Information Gathering through Dynamic Information Flow

This thesis addresses the problem of how to improve the performance of multi-robot information gathering by actively controlling the rate of communication between robots. Examples of multi-robot information gathering applications include cooperative tracking using mobile robots, cooperative search and environmental monitoring. Unlike single robot systems, multi-robot systems can provide complementary computation capabilities, a diversity of sensors and sensor view-points, modularity and robustness against failures. Communication is essential in such systems for decentralised data fusion and decision making, but wireless networks impose capacity constraints that are frequently overlooked. While existing research has focussed on improving available communication throughput, the aim in this thesis is to develop algorithms that make more efficient use of the communication capacity that is available. One challenge is that information may be shared at various levels of abstraction, raising the question of where information should be processed in the network. This decision in turn is dependent on limits of the computational resources available. Therefore, the flow of information needs to be controlled based on its value with respect to the task at hand given the communication constraints and the computation constraints. It is thus necessary to consider a fundamental trade-off between communication limits, computation limits and information value.

In this thesis, we approach this trade-off by posing the problem of deciding when, where and in what form to communicate in terms of decentralised constrained optimisation. We formalise this notion by introducing the *dynamic information flow (DIF)*

problem. Since decentralised information gathering requires communication for both data fusion and decision making, we suggest variants of DIF that either consider data fusion communication independently or both data fusion and decision making communication simultaneously. For the data fusion case, we propose efficient decentralised solutions that dynamically adjust the flow of information to improve information gain while obeying communication constraints. For the decision making case, we present an algorithm for communication efficiency targeted to linear-quadratic (LQ) systems and then extend the algorithm to information gathering tasks through local LQ approximations. The algorithm is then integrated with our solution for the data fusion case to produce a complete communication efficiency solution for information gathering. We analyse our suggested algorithms, present important performance guarantees and validate the algorithms in a custom-designed decentralised simulation framework with real-world scenarios. We also validate the algorithms through field-robotic experimental demonstrations involving two outdoor mobile robots, a ground station and a stationary camera. Experimental results demonstrate that our solutions achieved higher information gathering performance for the majority of test cases in comparison to naive down-sampling of information rates that utilise the same amount of communication bandwidth.

Our work has both theoretical and practical significance. The DIF problem formulation represents a new theoretical framework for studying communication efficiency and developing novel algorithms. Practically, our solutions to DIF enable applications of rich heterogeneous information gathering systems with many different types of sensors and computational resources without requiring manual design of the network topology.

Acknowledgements

I would like to begin by expressing my sincere gratitude and appreciation for my supervisors Dr. Robert Fitch and Prof. Salah Sukkarieh. Without your support and guidance, I would not have accomplished the great achievement of completing my thesis.

I could not find the right words to thank my family and closest friends for their patience and for being the pillar I lean against in difficulty. I have reserved a special thanks for my parents and my sisters. I thank you from the bottom of my heart, especially my sister Shereen. As for my close friends, I would like to say: you are my extended family. Thank you for being there anytime everytime. I have to thank Samir, Hussein, Kaveh, Fadel, Jihad, Dawood, Ali, Ragheed, Hassan and everyone else ...

To my friends at the ACFR, you are just wonderful! I do not know how I would have crossed this bridge without your help. I would like to thank Jason and Nick for their discussions, Zhe for his help with experiments and Prasad and Tariq for their jokes. I also thank Suchet, Marcos, Victor, Joe, and Ken for their chats and help with experiments. Finally, I would like to say that I really enjoyed playing soccer with the ACFR group.

I am grateful to everyone that helped me at the ACFR. If I have to name, then I would thank Dr. Tim Bailey for the continuous help, Dr. Ian Manchester for the deep insight, Dr. James Underwood for help with setting up experiments and Dr. Thierry Peynot for the small talks. I am also grateful to the technical staff at the ACFR especially Esa, Jereme and Mark.

I would also like to thank Professor Stephen Boyd for his helpful advice on ADMM.

I hope I have not missed anyone. If I have, then please forgive me.

Most importantly, I thank Allah, my lord, creator and cherisher, for his continuous and everlasting support, guidance and help.

To Al-Mahdi

Contents

List of Figures

List of Tables

List of Algorithms

List of Theorems

Nomenclature

Abbreviations

ACFR	Australian Centre for Field Robotics
UAV	unmanned aerial vehicle
IMU	inertial measurement unit
EKF	extended Kalman filter
RRT	rapidly-exploring random tree
DDF	decentralised data fusion
POMDP	partially observable Markov decision process
Dec-POMDP	decentralised partially observable Markov decision process
KL	Kullback–Leibler
ADMM	the alternating direction method of multipliers
DAG	directed acyclic graph
SPIN	sensor protocols for information via negotiation
DADMM	the distributed alternating direction method of multipliers
DIF	dynamic information flow
pdf	probability density function
LQ	linear-quadratic
LMI	linear matrix inequality
LQISO	linear-quadratic information structure optimisation
SDP	semi-definite programming
DDM	decentralised decision making
BFGS	Broyden-Fletcher-Golfarb-Shanno
ASyMTRe	Automated Synthesis of Multi-Robot Task Solutions through Software Reconfiguration

Chapter 1

Introduction

The objective of this thesis is to develop a unified and principled formulation and solution to the problem of improving the efficiency of communication resource usage by robots in decentralised information gathering. Decentralised information gathering involves multiple robots actively cooperating to maximise information about their environment. While existing approaches address various elements of communication efficiency in decentralised information gathering, the aim is to unify the problem of communication efficiency under the arch of one formulation that is amenable to practical algorithms with performance guarantees.

Decentralised information gathering systems typically rely on communication infrastructure, such as wireless networks, with limited resources. Communication resource constraints limit the size, applicability and versatility of decentralised information gathering systems. While existing research has focussed on improving available communication throughput, work here is targeted to making the use of communication more efficient for the task of decentralised information gathering. One challenge is that information may be shared at various levels of abstraction, raising the question of where information should be processed in the network. This decision in turn is dependent on limits of the computational resources available. Therefore, both communication constraints and computation constraints need to be considered.

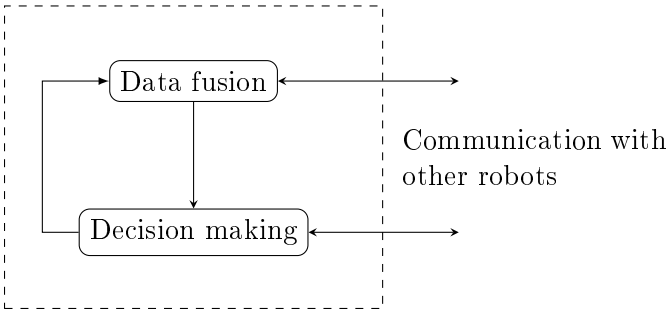


Figure 1.1 – Abstract representation of information gathering for one robot.

Decentralised information gathering, as a distributed form of active perception, typically requires communication at two layers, decentralised data fusion (DDF) and decentralised decision making (DDM), as shown in Figure 1.1. This two-layer division is common for most implementations due to the complexity of the general information gathering problem.

We introduce the novel *dynamic information flow (DIF)* problem formulation with three variants that either address communication efficiency at the DDF layer independently or at both layers. For each variant, we provide computationally efficient solutions. The solutions are analysed with performance guarantees provided for the solutions at the DDF layer. The solutions are also validated through field-robotic experimental demonstrations and extended simulations on a custom-designed decentralised simulation framework.

From a practical perspective, the outcomes of this thesis can be applied to a wide range of decentralised information gathering systems such as cooperative information gathering mobile robots, wireless sensor networks and large-scale spatio-temporal mapping systems. The suggested approaches simplify the development of such systems and advance many system designs towards practical implementations.

1.1 Decentralised Information Gathering

A decentralised information gathering task involves a team of robots that actively cooperates to maximise information about a given phenomenon. This task forms the

basis of many applications such as cooperative search [9, 28, 30, 46], target tracking [17, 18, 47, 106] and environmental monitoring [20, 39, 93]. Robot teams are useful for information gathering because they can exploit diverse sensing and motion capabilities, access multiple simultaneous view-points, are more robust against failures and cover large areas more rapidly than single-robot systems.

Communication is fundamental to the task because robots must cooperatively perform data fusion and decision making. Although communication may take place over wired or wireless networks, wireless networks are usually required for most mobile robotic platforms. Decentralised information gathering applications differ in their type and rate of communication needs.

1.1.1 Cooperative Tracking

In multi-robot cooperative tracking, each robot attempts to follow the most informative path to reduce the uncertainty of the estimate about the tacked targets. Reduction in the targets' state uncertainty naturally leads to tracking. Raw sensor data or processed observations may be shared between robots or sensor nodes. For sensor nodes without processing ability, raw sensor data has to be sent to be processed off-board. However, data are usually exchanged over a wireless medium with limited and shared bandwidth.

1.1.2 Cooperative Search

Another application of decentralised information gathering is cooperative search. Unlike tracking, the belief of the target position cannot be approximated by a single Gaussian probability density function (pdf). Instead, more complex representations are required. One example is occupancy grids [29]. DDF using occupancy grids results in large amounts of data being exchanged. DDM may also involve exchange of predicted observations resulting in large amounts of data transmission over the common network.

1.1.3 Environmental Monitoring

Environmental monitoring is a another example of a decentralised information gathering application. Environmental monitoring usually involves the spatio-temporal mapping of a phenomenon such as traffic [15], temperature, wind fields or water contamination. The representation of these fields using grids leads to high communication and processing requirements.

1.2 Resource Constraints

Communication is not an infinite resource. However, research in multi-robot systems often makes two invalid assumptions that fail to respect the physical limits of real communication networks. The first such assumption is that simultaneous communication between multiple pairs of robots is independent. In most existing wireless networks, bandwidth resources are shared *globally* and link capacity decreases rapidly as the number of robots increases [42, 107]. As illustrated in Figure 1.2, the number of connections required is quadratic in the number of nodes. The second invalid assumption, sometimes called the *r-disc* model, is that constant bandwidth is available within a given radius about a robot and that zero bandwidth is available otherwise. Real communication links are far more variable [63]. The implications of failing to consider communication limitations are significant and hence communication in realistic environments is currently a topic of considerable research interest [85].

One possible approach to address the issue of communication limits is to simply increase total network bandwidth by using more powerful and sophisticated radio hardware. However, it is always possible to generate a problem instance that exceeds any given resource limit. Sensors such as 3D laser range-finders generate data at a high rate, typically 1.3 million points per second. High-resolution cameras can produce data at even higher rates. The typical data rates for sensors commonly used in robotics are shown in Table 1.1 while the *maximum* available throughput of the

Table 1.1 – Typical data rates of commonly-used high-data-rate sensors.

Sensor	Data Rate
2D Laser	576 kbps
3D Laser	41.6 Mbps
Camera	104 Mbps

latest wireless standards are shown in Table 1.2. The actual available throughput is typically significantly less in most real-world environments.

In decentralised information gathering systems, communication is also used for DDM. Data rates required for DDM using different probability distribution representations are shown in Table 1.3. The values shown are for a two-agent system planning twice per second with five possible actions each using 64-bit floating-point precision. The values in the table clearly show that communication immediately becomes a problem as the planning horizon increases. Controlling the communication of these large amounts of data is essential to real-world application of decentralised information gathering systems.

We believe that a better approach is to develop algorithms that make efficient use of the communication resources at hand. We refer to this approach as improving *communication efficiency* in information gathering. The idea is to choose when and how a given pair of robots should communicate based on the information value of the communication and given resource limits.

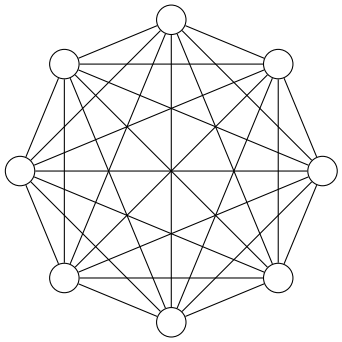


Figure 1.2 – The number of connections in a mesh network is quadratic in the number of nodes. For the network of eight nodes shown in the figure, twenty-eight connections are required to fully connect all the nodes.

Table 1.2 – Typical data throughput rates of latest wireless 802.11 standards.

Wireless Technology	Maximum Data Rate
802.11g	54 Mbps
802.11n	600 Mbps
802.15.4 (ZigBee)	200 kbps
Bluetooth V2.0 + EDR	3 Mbps

Table 1.3 – Typical data rates required by exhaustive-search decision making using grid-based pdf representations. The values are for a two-agent system planning twice per second with five possible actions each using 64-bit floating-point precision.

Planning Horizon	100×100 Grid pdf	8-Dimensional Gaussian pdf
1	3.2 Mbps	23.04 kbps
2	16 Mbps	115.200 kbps
4	400 Mbps	2.88 Mbps

1.3 Communication Efficiency in Data Fusion

The first main demand on communication in decentralised information gathering teams arises from data fusion. Robots need to share sensor observations to exploit the diversity of views provided by sensors on different robots. The importance of data sharing depends on the quality of data shared with respect to each of the robots. We would like to investigate whether robots can share observations selectively so that they can increase the efficiency of communication. The main challenge is that information may be represented at multiple levels of abstraction ranging from raw sensor data to highly compressed forms such as target state observations. Therefore, we must choose not only how to route data but also in what form. This decision must consider computation costs, since data may be processed at various possible locations within a system with varying resource capacity. A given robot may process its sensor data on-board, transmit this data to a powerful off-board processing station or rely on the computation resource of another robot. Manual design of a communication policy in this context is difficult and can result in poor communication efficiency. For example, down-sampling the rate of sensor data transmission may obey bandwidth constraints but can lead to unnecessary degradation in the performance of state estimation al-

gorithms. The design task increases in difficulty for large heterogeneous systems as we have shown in our previous work [78]. Moreover, if the task or hardware properties of an information gathering team are expected to change throughout operation, then a fixed policy would fail to maintain its intended performance. Therefore, the information flow must be adjusted dynamically and autonomously.

An example of a decentralised information gathering task that can benefit from communication efficiency in data fusion is that of multiple information gathering robots with limited inter-robot communication bandwidth. In such tasks, broadcasting observations from each robot to all robots is unjustified. When distributed spatially over a large area, the priority of robots should be gathering information from their proximities. For example, when tracking multiple dispersed targets [13], robots only need to receive observations of nearby targets. As a consequence of communication efficiency, we expect the robots to select when and with whom to share observations based on the impact of the observations on the robots' estimates and based on the cost of communication computed according to separation distances.

Small agile robots such as quadrotors or small mobile robots provide a relatively low-cost option to multi-robot systems. However, such robots usually lack the necessary computational resources for processing data from sensors with high data rates such as high-resolution cameras. With access to an off-board processing station, these robots can send raw data wirelessly to the processing station and receive processed observations in return. Upholding communication and computation efficiency would dictate that when robots move away from the processing station and available throughput decreases, robots should process down-sampled data on-board instead of sending data with the full rate to be processed off-board.

Another case of interest is when surveillance robots are equipped with sensors having a limited field of view. Such situations justify the use of stationary cameras to obtain the required coverage. Images from the camera can then be sent to the robots for object detection. The extra view-point provided by the camera can greatly aid the robots. Due to wireless communication limitations, the robot will lose the ability to receive the images wirelessly if its distance from the camera increases. Moreover,

Figure 1.3 – Autonomous ground vehicle with many sensors.

the robot might not require image observations if it already has an accurate estimate of the target. As an example of desirable behaviour, the robots should only receive raw camera data when they are close to the camera and when the camera covers a view-point not covered by the robots.

Finally, the problem of sensor selection also appears in the case of a single robot with many sensors such as that shown in Figure 1.3. In such cases, processing the entirety of the sensor data may be prohibitive. Pre-selection of the sensors before the experiment introduces unnecessary rigidity in the system design. Instead, smart selection of sensors according to different robot missions, locations and environments is desirable.

1.4 Communication-Efficient Information Gathering

Communication efficiency in information gathering necessitates efficient communication for both the data fusion and decision making layers. Therefore, communication-

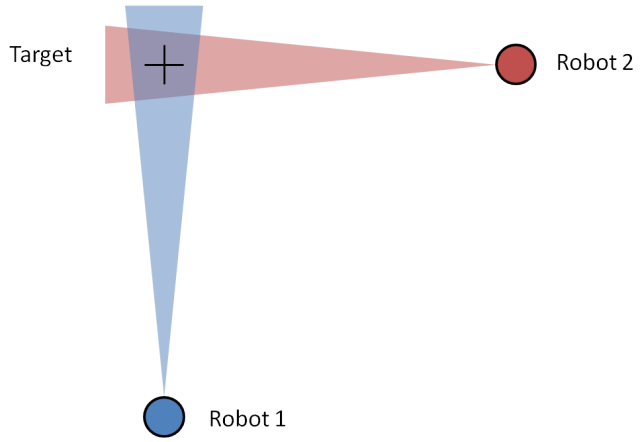


Figure 1.4 – Robots tracking a target with bearing-only sensors.

efficient data fusion needs to be coupled with communication-efficient decision making to achieve communication-efficient information gathering.

Communication is needed during cooperative decision making since robots in multi-robot systems are usually coupled in utility or dynamics [49, 50, 68]. The degree of cooperation required is related to the degree of coupling between robots.

In a similar manner to the data fusion case, to achieve communication efficiency in decision making, we would like to investigate whether robots can selectively choose with which of the other robots in a team they should negotiate decisions. This research question has traditionally appeared in team decision theory. Team decision theory deals with the problem of information structures, or more specifically, what a robot in a team needs to know in order to choose its optimal decision. From a different perspective, it also deals with how the amount or type of information to which a robot has access affects its ability to make decisions.

Consider two robots tracking a target with bearing-only sensors as shown in Figure 1.4. Bearing-only sensors achieve optimal performance when they are situated at a 90° angle relative to the target. When robots are close to each other, they need to negotiate their movement direction so they can increase their relative angle. When they are at the optimal angle, negotiation is only required once the target begins to move.

As another example, consider a team of robots with the task of mapping an environment [23]. The mission can be formulated as a decentralised information gathering problem. To achieve acceptable performance, the robots need to minimise their exploration overlap during the mission through continuous inter-robot negotiation. However, when the robots are far apart, and it is highly unlikely that their future observations will overlap, then the need for negotiation is reduced.

The term *negotiation*, in this thesis, refers to the process of cooperative decision making. In other words, it refers to the process by which robots in a team jointly consider the utility or effect of their actions. This is opposed to *local* decision making by which robots only consider the effect of their own actions.

1.5 Dynamic Information Flow

In this thesis, we formalise the notion of communication efficiency in information gathering by introducing a novel problem formulation which we call the *dynamic information flow* problem. Given a graph-based representation of a decentralised information gathering system, the objective is to maximise the information value of communication by minimising a cost-based metric subject to constraints. The graph representation models an information gathering team as a system where data flows along a typical pipeline comprising sensors, perception algorithms, estimation algorithms and control algorithms. These logical elements are connected by communication links with associated costs, and a system may contain many such elements. For example, a single laser sensor may be connected to many other elements implemented on multiple robot platforms.

The DIF problem structure is designed to model trade-offs between information value, communication cost and computation cost. The information value of sensor observations is not defined globally but instead is defined relative to the belief state of each estimator element. Link costs are abstract costs that model both communication and computation. For example, a given sensor observation may be of high value to an

estimator, but obtaining this information may incur a high cost due to the computational demands of a perception algorithm or due to large communication bandwidth requirements. Formulating the problem in this way provides a mechanism to balance these diverse costs against information value in a principled manner. Because link costs are abstract and dynamic, the problem admits any realistic communication link model and is not limited to the r-disc assumption. The threshold-DIF variant can model the global bandwidth constraint imposed by common shared-channel communication systems. Modelling system elements logically as a graph where flow rates are dynamically optimised avoids the need to manually pre-determine the information architecture of the system. This property is particularly useful for heterogeneous systems with many types of robots that have a range of sensing and computational resources.

We define the DIF problem in Chapter 3 through a family of optimisation problems with two concrete variants for the data fusion case, *min-cost-DIF* and *threshold-DIF*. We also introduce a third variant, *negotiation-DIF*, that includes decision making.

1.5.1 Min-Cost-DIF and Threshold-DIF

A solution to the min-cost-DIF and theshold-DIF problems is in the form of a set of multicast flow rates that determine which pairs of robots communicate at any given time. In min-cost-DIF, the objective is to minimise the sum of link costs, assuming the relative scale of these costs is known. In threshold-DIF, the relative scale of costs is not assumed to be known and the objective is to find a solution that satisfies a given flow threshold.

We present algorithms and analysis for both problem variants in Chapter 4. Our solution to min-cost-DIF is based on an adaptation of multicast routing. We prove that min-cost-DIF can be transformed such that existing multicast routing algorithms may be applied, and we present one such algorithm. Our solution to threshold-DIF is based on an optimisation method known as the alternating direction method of multipliers (ADMM) [10]. We derive a decentralised version of this algorithm which

we call *the distributed alternating direction method of multipliers (DADMM)* and show how it can be applied to solve threshold-DIF. We analyse convergence and running time for all algorithms and validate these results through simulations including up to 28 nodes.

We also present experimental results that illustrate the behaviour of our algorithms and compare information gain performance with simple bandwidth-limiting methods. The task we consider is to track a moving target using multiple types of sensors. For the case of min-cost-DIF, the experimental system consists of one mobile robot equipped with a camera and one auxiliary static ground station. We also present simulation results for two mobile ground robots. For threshold-DIF, the experimental system consists of two outdoor mobile robots, with and without an auxiliary static camera. One robot is equipped with a 2D laser sensor and the other is equipped with a 3D laser. To further evaluate the performance of our algorithms, we present results from Monte Carlo simulations that demonstrate statistical significance.

Our results demonstrate that the algorithms efficiently use available communication bandwidth to increase information gain. We observe that sensor data are either processed on-board or transmitted and processed at the ground station appropriately. We also observe that information from multiple sensor sources is communicated selectively based on sensor utility, available bandwidth and route overlap.

1.5.2 Sensor Utility

An essential requirement for success in improving communication efficiency is the accurate estimation of sensor utility. Computing the exact sensor utility computation through a decentralised partially observable Markov decision process (Dec-POMDP) framework results in an NEXP-complete problem [8]. Instead of exact sensor utility computation, we require approximations that are computationally efficient. In Section 4.1.3, we present a myopic approximation to sensor utility that was used in our experiments.

Efficiently computable theoretical bounds for sensor utility exist for some cases; however, they can be too conservative. In Section 4.4, we show results of a multi-robot field mapping example comparing different sensor utility estimates. Results empirically show that existing theoretical bounds are too conservative to be of any practical use. They also show that the myopic approximations are usually acceptable approximations of the exact utility when computed over a fixed time horizon.

In Section 5.4, we show how the sensor utility estimate can be improved when coupled with the decision making layer. The coupling allows for the estimation of an observation's utility based on its impact on control decisions and not just on its uncertainty reduction.

1.5.3 Negotiation-DIF

Negotiation-DIF is a problem formulation of communication efficiency in information gathering. Negotiation-DIF extends DIF to include communication-efficient decision making. Due to the complexity of information structures, negotiation-DIF only models the case of soft communication constraints. In other words, communication is modelled as link costs instead of explicit constraints.

Negotiation-DIF is of particular importance to applications that involve large amounts of data exchange during data fusion and cooperative decision making. Negotiation-DIF is not limited to a particular decision-making algorithm. It can be applied to a wide range of decentralised decision making algorithms.

A key assumption of negotiation-DIF is the decoupling between the DDF and DDM layers. This assumption is necessary since, otherwise, robots need to jointly determine communication and control actions. Although this can be done through a Dec-POMDP framework, the resulting problem would be intractable.

The solution to negotiation-DIF requires a combination of our solution to min-cost-DIF with an extended version of our solution to the comms-LQ problem introduced in Section 1.5.4. As we show in Chapter 5, the amalgamation occurs quite naturally.

The same link costs can be assigned to links sharing a common resource for both algorithms.

We present simulation results of information gathering experiments using the negotiation-DIF formulation to attain communication efficiency in both data fusion and decision making. The simulation evaluate negotiation-DIF qualitatively through a heterogeneous multiple-node experiment and quantitatively through a simple two-robot simulation. The advantage of our communication efficiency solution is statistically verified against naive down-sampling through a Monte-Carlo simulation.

1.5.4 Comms-LQ

We introduce comms-LQ as a communication-efficient decision-making problem formulation for the special case of a linear-quadratic (LQ) team. Comms-LQ is adapted by negotiation-DIF extending DIF to the decision making layer. The aim of comms-LQ is to obtain an optimal feedback control policy while minimising communication link costs. An LQ team is defined as a team of robots with decoupled linear dynamics and a coupled quadratic cost function. Each robot's control decision requires knowledge of the state of other robots where the state information is transferred via communication links with associated costs.

Communication link costs are abstract costs that are suitable for representing cases where multiple decentralised algorithms are utilising the same communication infrastructure. If only a single decentralised algorithm were running on the communication network, an alternative approach which optimises decisions given a constraint on communication capacity could be used [90]. A solution to comms-LQ is a control policy that minimises the team quadratic cost function and communication link costs simultaneously.

Although comms-LQ is targeted to an LQ team problem, it can be modified to a more general set of problems by acting as an auxiliary layer added to existing DDM algorithms and by taking local LQ approximations. In particular, we extend the

approach to adjust the frequency of negotiation in decentralised information gathering applications in negotiation-DIF.

Our solution to comms-LQ is based on the linear matrix inequality (LMI) formulation of the LQ problem [12, 81, 98] and is called linear-quadratic information structure optimisation (LQISO). The LMI formulation allows for extra flexibility in the design of a feedback control law for an LQ system. We exploit this flexibility to add communication costs to links between robots.

We show results of applying LQISO to an LQ problem having demonstrated a reduction in communication that is consistent with chosen communication costs. The consequent performance demonstrates an efficient use of the reduced communication capacity available. The extension of LQISO is also demonstrated for a sample decentralised information gathering task with results showing a reduction in communication and no significant impact on information gathering performance.

1.6 Scope and Assumptions

Work presented in this thesis introduces communication efficiency to an existing decentralised information gathering method. Rather than introducing a new information gathering solution, the communication efficiency solutions presented add an auxiliary layer that regulates communication between nodes of an information gathering team.

A key assumption made in this thesis and which is typical for large-scale systems is the decoupling between data fusion and decision making. This assumption is closely related to the assumption of decoupling between estimation and control for centralised systems.

As presented in this thesis, the DIF problem formulation assumes existing communication infrastructure that can provide continuous information flow between, albeit at a limited rate which may also be zero. It also assumes that the infrastructure can instantaneously switch between routes.

Finally, we note that the solutions presented are aimed at systems involving high-data-rate communication in such a way that the communication overhead introduced by these solutions is negligible in comparison to the bandwidth of information being transferred.

1.7 Contributions

The main contribution of this thesis is to improve the performance and applicability of heterogeneous high-data-rate decentralised information gathering systems by introducing communication efficiency. Our algorithms address the trade-off between information utility and communication limits and are computationally efficient. These algorithms enable the implementation of rich heterogeneous systems with diverse sensing, computation and mobility capabilities. A list detailing the specific contributions is shown below. These contributions have so far been published in [51] and [52].

- Introduction of DIF as a novel and principled problem formulation of communication efficiency in information gathering. The DIF formulation represents the trade-off between communication, computation and information gain as a distributed optimisation. It is a general and flexible formulation that is amenable to efficient algorithms. DIF is introduced concretely through three variants: min-cost-DIF, threshold-DIF and negotiation-DIF. Min-cost-DIF and threshold-DIF consider communication for data fusion only while negotiation-DIF considers communication for both data fusion and decision making.
- A solution to min-cost-DIF based on multicast routing achieving link-cost communication-efficient data fusion.
- A solution to threshold-DIF based on a distributed version of ADMM achieving communication-efficient data fusion with explicit global resource constraints.
- A solution to negotiation-DIF achieved by integrating the multicast routing algorithm used for min-cost-DIF with the extended version of LQISO. LQISO is

a solution algorithm to the communication-efficient decision-making problem for LQ systems based on the LMI formulation of the LQ optimal control problem.

- Analysis for all algorithms and performance guarantees for the solution algorithms of min-cost-DIF and threshold-DIF.
- Experimental validation using two mobile robots, a processing ground station and a stationary camera and extended simulation results including large systems and Monte Carlo analysis.

DIF is a novel formulation that permits decentralised, efficient and practically implementable solutions. The novelty of DIF lies in the representation of communication efficiency in information gathering as a multicast graph-based decentralised optimisation which is the first general formulation of this type for a broad class of systems and tasks. The suggested solutions to min-cost-DIF and threshold-DIF are of particular benefit to heterogeneous decentralised information gathering systems with vast amounts of sensor data. The solutions allow for larger systems and/or improved mission performance in comparison to naive down-sampling methods.

The LQISO and extended-LQISO algorithms are applicable to multi-robot systems that exchange significant amounts of information during cooperative decision making. The algorithms guide the use of communication resources by the decision-making process. This resource usage reduction reserves more of the available bandwidth for data fusion and can potentially improve system performance. In general, similar to the data fusion case, the algorithms also allow for larger systems and/or improved mission performance in comparison to naive down-sampling methods.

Negotiation-DIF combines the benefits of min-cost-DIF and extended-LQISO. It is targeted to systems with high communication usage for both data fusion and decentralised decision making.

1.8 Outline

The remainder of this thesis is organised as follows:

Chapter 2 surveys related work in the fields of communication efficiency in data fusion and communication efficiency in decision making.

Chapter 3 defines the DIF problem and LQISO problem in addition to the overall problem of communication efficiency in information gathering which is the combination of the first two problems.

Chapter 4 presents our solution approach to the DIF problem with detailed analysis.

Chapter 5 presents our solution approaches to comms-LQ and finally to negotiation-DIF.

Chapter 6 presents the results of the experiments conducted using our solution approach to DIF, LQISO and the general communication efficiency in information gathering problem.

Chapter 7 summarises the thesis and suggests possible future directions.

Appendix A proves the non-submodularity of linear-Gaussian information gathering.

Chapter 2

Related Work

Several approaches to the problem of communication constraints in multi-robot systems have been suggested in existing work in the field. We have divided existing approaches into three groups. Approaches that address communication constraints in general are surveyed in Section 2.1. Approaches that improve the communication efficiency of data fusion are surveyed in Section 2.2, while approaches that improve communication efficiency in cooperative decision making are surveyed in Section 2.3. Finally, in Section 2.4, we provide a brief survey on methods that actively plan for connectivity. In Section 2.5, we position our work in relation to network flow optimisation problems and in Section 2.6, we list existing sensor utility approximation methods.

We summarise the presented related work, identify possible shortcomings and then delineate our direction in Section 2.7. While existing work in the field has managed to address the problem of communication constraints for specific multi-robot systems, our aim is to present a unified and principled approach to communication efficiency in decentralised information gathering systems. We are unaware of other approaches with these properties.

2.1 Addressing Communication Resource Limits

Available communication hardware in multi-robot systems typically imposes resource limits that hinder the implementation of distributed algorithms having a consequent effect on the ability to achieve the team goal. For a multi-robot system that relies on wireless communication, the main resource limit that faces multi-robot algorithms is the available *data throughput* between robots. Limits on available throughput in turn affect the performance of decentralised algorithms as experimentally demonstrated by Fitch and Lal [27] for the case of decentralised planning. This fundamental issue has been addressed differently by different research communities.

Early work addressing communication limits in multi-robot systems, such as that of Yoshida et al. [107], resorted to enforcing *local* communication to minimise interference between links. Ohkawa, Shibata and Tanie [76] analysed the size of the communication neighbourhood necessary to achieve the team task. Recently, the problem of communication efficiency has gained increasing prominence [85].

One group of approaches addresses the problem of communication efficiency by intelligent selection of the multi-robot network topology. Bayram and Bozma [7] propose optimising communication efficiency by modelling the network topology formation as a pairwise game. The authors use a centralised coordinator to adjust the network topology to optimise a function that includes communication cost and task utility. An alternative set of approaches, surveyed by Zhang et al. in [108], allows communication resources to be regulated using auction methods where nodes bid for communication resources.

The exploitation of heterogeneous capabilities of multi-robot teams was investigated by Donald [21] who introduced the idea of information variants, investigating conditions under which communication may be replaced with computation or prior knowledge for example. Inspired by the idea, Tang and Parker [96] later introduced Automated Synthesis of Multi-Robot Task Solutions through Software Reconfiguration (ASyMTRe) with the aim to automatically determine connections, in a multi-robot team, between modules with different sensing, perceptual and motor capabil-

ities. ASyMTRe is based on a definition of the set of available sensors, perceptual schemas, communication schemas and motor schemas. DIF adopts a similar formulation; however, the specific scope of DIF, information gathering applications with communication limits or costs, allows DIF to explicitly define a graph structure on which the communication efficiency can be posed as a decentralised optimisation problem.

The limitations of wireless communication have also led to novel approaches that aim to boost the capacity of wireless networks. Multi-radio multi-channel networks [102, 104] can significantly increase network capacity by using multiple communication channels in parallel. Recent work by Kuo and Fitch [57] has shown that a single channel may be reused in a neighbour-to-neighbour architecture while avoiding mutual interference. The authors demonstrate the ability of their approach to maintain constant throughput with an increasing number of nodes.

Approaches that deal with communication efficiency for general multi-robot systems do not typically consider the content of the data exchanged. While the aim of these approaches is to maximise the throughput available from source to destination, this thesis takes a complementary view. Instead of transmitting as much data as possible, we attempt to transmit only the most valuable data. Thus, data with little information value do not consume communication resources and available bandwidth is used efficiently.

From an optimal control point of view, the entire communication-efficient information gathering problem can be modelled as a Dec-POMDP [14, 35, 37, 99] which is a powerful and general approach. Communication decisions can be designated as possible actions that are selected using existing optimal control algorithms [37]. However, Dec-POMDPs are computationally intractable for large problems due to the "curse of dimensionality" [8]. We are interested in large problems with many robots and sensors, and we focus on computationally efficient solutions to the more specialised DIF problem.

2.2 Communication Efficiency in Data Fusion

The issue of high communication demand appeared during pioneering work in the field of data fusion in robotics [40]. Several studies of possible efficient network topologies were conducted by Grocholsky and Nettleton [41, 74]. Nettleton also investigated methods that avoid overlap in information for different network topologies. More recently, Gupta et al. [43] have proposed a sensor scheduling strategy for multiple sensors with bounds on the estimation error covariance.

Current approaches to communication efficiency in data fusion are typically specialised in their applications. One group of approaches is specific to wireless sensor networks while another group is targeted at networks with severe communication limits. A third group of approaches is specialised to target tracking. We discuss research work that has been conducted on each of these three groups.

2.2.1 Wireless Sensor Networks

Wireless sensor networks typically consist of a large number of small sensor nodes with limited energy and processing capability. Therefore, the need for localised communication in wireless sensor networks to avoid the inflation in the number of communication links was realised during early work in this area [24]. Kulik, Heinzelman and Balakrishnan [56] introduce the SPIN routing protocol as a routing mechanism for sensor networks. Sensor nodes send an advertising message that contains metadata about the sensor information available and potential recipients send requests as required. However, the semantics of the metadata are not specified and are considered application-dependent. The strategy determines the order of sensor selection out of a set of known sensor models. Bagula et al. [5] present an efficient multipath routing algorithm for wireless sensor networks. The suggested routing model incorporates delay and reliability quality-of-service constraints. Data from different sources is considered to be independent. Instead of bandwidth limits, Schurgers and Srivastava [87] consider the case where communication is limited by available energy

at network nodes. Consequently, intelligent routing methods are employed to reduce and equally distribute energy usage. A gradient-based routing technique is proposed where nodes are assigned heights after a user transmits an "interest" message and information is then sent along the steepest descent path. An extensive survey on routing techniques in wireless sensor networks can be found in [3].

Communication protocols in wireless sensor networks typically consider homogeneous nodes. Heterogeneous nodes with different capabilities which may include sensing, processing or both introduce extra challenges. One of those challenges is the problem of dynamically selecting the processing platform for the produced sensor data. Heterogeneity also introduces the possibility of multicast routing which is typically overlooked in research on wireless sensor networks.

2.2.2 Strict Communication Limits

Another body of work in the field of communication efficiency investigates the issue of information sharing with severely limited communication throughput. This area of research is motivated by military applications or applications with miniature sensor nodes. Data is regulated at the level of bits. Nerurkar and Roumeliotis [73] present a cooperative localisation framework in which robots rely on the transmission of quantised sensor observations due to strict communication limitations. The robots choose the quantisation rule based on the available bandwidth. The authors present hybrid estimators through which each agent processes its own analog observations and quantised observations from other agents. The suggested approach produces a communication-efficient framework for cooperative localisation. Ribeiro and Giannakis [83] introduce distributed estimators for binary observations with non-Gaussian noise probabilities.

Field-scale robotics do not suffer from the severe limitations that necessitate communication regulation at the bit level. Instead, the main cause of limitation in field robotics is the abundance of sensor data.

2.2.3 Target Tracking

Target tracking with high-data-rate sensors introduces extra complexity in deciding when and where to process raw sensor data. However, existing communication-efficient target tracking solutions have traditionally only considered point observations. Chen et al. [16] propose an algorithm for sensor networks that uses minimal communication by only transmitting relative changes. The algorithm relies on binary sensors that detect the presence of a sensor inside a sector of a detection disc around each sensor. Nodes only need to transmit information to their neighbours if the target changes its sector location. Zheng et al. [109] propose an auction-based adaptive sensor activation algorithm for the purpose of target tracking. The algorithm relies on predicting a target location and using auction methods to assign a new cluster whose mission is to track the target. Since nodes outside the cluster are not activated for tracking, they do not consume any computational or communication resources. Hence, the energy efficiency is improved.

Approaches aimed at target tracking offer promising results for the application of target tracking but do not readily generalise to other information gathering applications. They do not address the challenges of heterogeneous systems. More specifically, these approaches do not address the problem of selecting where to process raw data of high-data-rate sensors.

2.3 Communication Efficiency in Decision Making

Communication efficiency in decision making branches from *team decision theory* which can be traced back to the pioneering work of Radner and Marschack [64, 79]. The work presented by these authors stems from an economical background and its aim is to analyse the performance of teams in organisations. Team decision theory differs from game theory [72] by assuming cooperative teams instead of selfish agents. Therefore, team decision theory is more relevant to multi-robot systems. Results from team decision theory are usually limited to decentralised LQ problems.

An important research focus in team decision theory is that of information structures [45]. Information structure design attempts to answer the question of which agent needs to know what piece of information before deciding its next action. Thus, the information structure design problem is closely related to communication-efficient decision making. However, the difficulty of the problem has limited theoretical results to the simple case of LQ systems [6, 32, 45, 84], while existing results for non-LQ systems are limited to special applications.

Notable research work in communication-efficient decision making from a different point of view is that of Klavins [54]. The author introduces the notion of communication complexity that attempts to capture the need for communication in multi-robot systems based on the coordination requirements. The author analyses the complexity of several communication schemes.

2.3.1 LQ Systems

LQ systems have linear dynamics and quadratic cost and are readily solvable for the case of unlimited communication bandwidth. Interesting research problems appear when communication constraints are imposed. Rotkowitz and Lall [84] identify a class of convex problems in decentralised LQ control. The authors show that decentralised LQ problems with information structure constraints remain convex if the condition of *quadratic invariance* is satisfied. Schwager et al. [88] give a condition on the stability of a second-order decentralised control system in terms of the network update time. Matveev and Savkin [67] analyse the problem of a centralised controller receiving observations over communication channels from distributed sensors. The authors provide tight lower bounds on the channel capacities for which stabilization by the controller is possible.

A directly related problem to information structure analysis is *distributed LQ control*. In distributed LQ control, communication costs are assigned to elements of the state vector or the control decisions are explicitly subject to information structure constraints. Speyer, Seok and Michelin [94] formulate an LQ team optimal control

problem with stochastic dynamics and observations where communication costs are appended to the Lagrangian. Although this is an intuitive approach, it is difficult to solve. Molin and Hirche [69] suggest a solution to the LQ problem with communication costs for the discrete-time finite horizon case. The authors formulate the scheduling decisions in a dynamic programming framework with control optimality retained with respect to the chosen scheduling. Semsar-Kazerooni and Khorasani [89] suggest the decomposition of a control to local and global components allowing for a decentralised consensus algorithm with guaranteed convergence. Finally, Nguyen et al. [75] use decentralised linear functional observers to reduce the communication requirements of robotic formation feedback control.

The use of linear matrix inequalities (LMIs) in linear control problems allows additional design criteria to be added to the LQ problem [51, 86, 90, 91]. Lu, Xie and Fu [62] consider the problem of choosing a "communication sequence" of observations for a $\mathcal{H}_\infty$ control problem. The communication sequence is assumed to be periodic and an LMI optimisation problem is devised to determine the optimal sequence. Scherer, Gahinet and Chilali [86] introduce an LMI approach to linear control for multi-objective control. As an example, the approach shows how a mixture of $\mathcal{H}_2$ and $\mathcal{H}_\infty$ objectives can be specified using LMIs. This mixture is not achievable using classical Riccati equation solutions. Semsar-Kazerooni and Khorasani [90] provide a solution for the LQ team problem with a restricted information structure. In Section 5.2, we build on this solution by introducing a novel distributed LQ control approach that allows the designation of communication link costs instead of a fixed information structure.

2.3.2 Non-LQ Systems

As an example of an approach to communication-efficient decision making for a non-LQ system, Jennings's group [25] introduces a decentralised information gathering system where agents first communicate to find if their decisions are coupled in utility. If they are coupled, then they ensure that each others' decisions are considered in

their optimisation. The team utility is to maximise information over a time horizon. Agents exchange predicted likelihoods over the time horizon and evaluate overlap. The likelihoods have to be transmitted for all possible actions and for all steps of the horizon. This amounts to a large amount of data being communicated. Work by Xu, Fitch and Sukkarieh [105] allows robots to incrementally learn the prediction of other robots' observation utility and adjust inter-robot negotiation accordingly. The work proposes a method by which robots switch between negotiation and local decision making based on the learnt utility. This approach is suited to homogeneous systems with sufficient capacity for the extra computational overhead necessitated by the learning process. Addressing the problem of spatial redeployment for a multi-robot team due to the introduction of new agent-task pairs, Liu and Shell [61] investigate the effect of inter-robot communication range on paths produced by their algorithm. Simulations results show that little change is observed in resulting paths when robots increase their direct neighbourhood size beyond seven. This demonstrates relevance to applications with limited communication ranges. Other relevant pieces of work include the work of Rekleitis et al. [82] that presents a multi-robot coverage algorithm with robots only assuming line-of-sight communication and that of Otte and Correll [77] that introduces the Any-Com method which seeks efficient use of communication for the purpose of path planning.

Task allocation is occasionally employed to simplify multi-robot coordination [55, 105]. Liu and Shell [60] introduce an algorithm for multi-robot task-allocation with distributed variants. The distributed variants rely on message passing and do not assume global knowledge of the problem specification. In our work, we rely on algorithms that reach task allocation implicitly rather than explicitly.

In the context of decentralised optimisation, we note that algorithms introduced by Mathews [66] allow for dynamic communication rates in decentralised optimisation. The author employs decentralised optimisation to decide on control actions over a receding time horizon for decentralised information gathering. The algorithm dynamically determines the required communication rates for the optimisation by the

estimated coupling in utility between sensor nodes. The algorithm however does not allow for different communication costs to be assigned to different links.

2.4 Communication-Aware Motion Planning

Communication-aware motion planning has recently become a highly active research area in robotics. Communication-aware motion planning is an approach to the issue of communication in multi-robot systems through which robots actively seek to maintain some communication quality metric while performing their tasks. Hsieh et al. [48] use a radio signal strength map with a reactive controller to maintain communication links between robots. Mostofi [71] represents communication quality degradation as noise to allow for an information-theoretic trade-off between sensing and communication. Fink, Ribeiro and Kumar [26] use a rapidly-exploring random tree (RRT) motion planner with a convex optimisation that is run at each tree extension to determine if the communication quality constraints are satisfied. Many other approaches have also been suggested by various authors for different scenarios. For instance, Stachura and Frew [95] target motion planning for multi-hop communication scenarios while others consider connections to a fixed ground station [34, 97]. As another example, the work by Mather and Hsieh [65] is targeted for task allocation tasks. Finally, Lindhé and Johansson [58] specifically address motion planning for tracking while considering multipath fading and Goerner, Chakraborty and Sycara [36] address the case of mobile robots collecting data from spatially distributed sources.

Although connectivity maintenance is imperative for multi-robot systems, the problem we define in this thesis is largely orthogonal to the problem of connectivity and with a different objective. The problems we define assume a connected network with focus on using this connected network efficiently by choosing when data should be transmitted instead.

2.5 Network Flow Optimisation

A classic problem in network flow optimisation is the *minimum cost flow* problem [2]. The minimum cost flow problem has known efficient decentralised solutions; however, the DIF problem is more closely related to the *multicast network routing* problem. This problem is equivalent to the Steiner tree problem on directed graphs which is NP-complete [80]. In a special case using *network coding*, multicast routing can be solved in polynomial time and in a decentralised manner [19, 103]. Our algorithms exploit this special case. However, in our implementations we use an approximation that approaches the performance provided by network coding in relatively small networks.

2.6 Sensor Utility

Sensor utility can be computed exactly using the partially observable Markov decision process (POMDP) formulation of the information gathering problem; however, this problem formulation is intractable since Dec-POMDPs are NEXP-complete [8]. Due to the difficulty of the problem of sensor utility estimation, existing approaches either rely on myopic approximations or theoretical bounds that can be obtained for simple problems.

An approximate myopic reward for sensor utility is used in the work by Williamson, Gerding and Jennings [100]. The utility is based on the Kullback–Leibler divergence obtained by incorporating an observation. Sensor utility is approximated by computing the entropy reduction caused by the last observation received. This approach is advantageous due to the simplicity of implementation since entropy reduction can be computed without additional data storage and with a computational time independent of the planning horizon.

Some theoretical bounds related to sensor utility have been devised for linear-Gaussian systems. Work by Sastry's group [92] provides lower and upper bounds for the communication rate required to maintain a bounded error covariance for a Kalman filter. Pappas's group [4] provides an upper bound on the deviation in performance after a

specific time horizon due to a current deviation in the estimate. This upper bound can be used as an upper bound of non-myopic utility of a sensor observation. However, such a bound is usually conservative and may not be of much benefit in some cases.

Sensor utility estimation can be made more efficient by learning the utility of observations as a function of the robot and target states. This approach suggested by Xu, Fitch and Sukkarieh [105] is suited to homogeneous systems with sufficient capacity for the extra computational overhead necessitated by the learning process. To avoid this additional overhead, in this thesis, we employ a simple myopic approximation similar to that in [100].

2.7 Summary

Although existing approaches to communication efficiency have addressed a large range of outstanding problems, they unnecessarily assume the rigidity of various elements in a multi-robot system. A dynamic and unified strategy to the problem of communication for information gathering tasks is achievable by modelling the trade-off between the resources of available hardware components and the team mission as a distributed optimisation. While the use of multi-radio multi-channel networks has boosted available bandwidth for a general multi-robot network, the common assumption is that the bandwidth requirements cannot be adjusted throughout system operation. Another unnecessary assumption which is of main importance to heterogeneous systems is the assumption of a fixed sensor-data processing pipeline. Finally, communication-efficient decision making approaches have typically ignored the need to simultaneously consider communication efficiency in data fusion.

In this thesis, we seek a principled, dynamic, unified and practical approach to communication efficiency in information gathering. The flow of information between various nodes of the system is expected to change throughout system operation. The dynamics of the flow should be dictated by an optimisation of the trade-off between

communication resource limits and information utility. The approach should be flexible enough to consider heterogeneous systems and to consider communication at both the data fusion and decision making layers. Finally, for practical implementation, the approach should be decentralised and have minimal computation and communication overhead.

Chapter 3

The Dynamic Information Flow Problem

In this chapter, we formally define the dynamic information flow (DIF) problem. We introduce two variants, min-cost-DIF and threshold-DIF, both of which correspond to communication efficiency in data fusion. We then introduce a third variant, negotiation-DIF, as a problem formulation for communication efficiency in information gathering by extending min-cost-DIF to include communication-efficient decision making.

3.1 The DIF Problem

The goal in the DIF problem is to maximise information gain by controlling the flow of information within a decentralised information gathering system subject to communication and processing constraints. We define the DIF problem in general form.

Before providing the formal definition of DIF, we briefly state the key assumptions that define its scope. The DIF formulation targets decentralised information gathering systems with the following properties. The formulation assumes that decision

making is decoupled from estimation or data fusion. The controller, or the decision making module, supposes the existence of an estimator that provides an estimate of the world with a measure of uncertainty. The separation between estimation and control [33] is utilised due to the simplification it provides. It also allows us to improve communication efficiency without the complexity of having to consider control decisions concurrently. The second assumption is that sensors, which are the input sources of information, continuously produce data that are consumed by other elements of the system. Moreover, raw sensor data may need to be processed before being used in estimation.

In DIF, the flow of data between elements is modelled through a graph structure. A decentralised information gathering system is a configuration of several elemental components. *Sensors* are elements that generate sensor data measured by physical sensing devices, such as laser scanners and cameras. Data from such sensors are transformed into *observations* by applying algorithms such as object detection and classification. *Processors* are computational elements that perform these processing tasks. Processors may be cascaded if necessary. The observations generated by processors act as input into *estimator* elements that maintain belief states. For example, an estimator could be an extended Kalman filter (EKF) in the tracking case or an occupancy grid in the mapping case. *Controller* elements use the estimate from the estimators to make decisions and take actions. For instance, a controller may be the path planner of a robot. For simplicity of presentation, we defer discussing the role of controller elements to Section 3.4.

Data flows via a communication system from sensors to processors, from processors to other processors and from processors to estimators. The topology of the resulting network is a directed acyclic graph, where information value, communication and computation demands induce costs or constraints on the links in the graph. The induced link costs may vary according to the properties of the underlying communication mechanism, which may not be the same for all links. Elements of the system generally are physically distributed among multiple robots or ground stations and therefore communicate using an inter-robot communication system such as a wireless

network. It is also possible for multiple elements to reside within a single physical platform and communicate using an intra-robot communication system such as a wired network or in-memory communication.

These system elements can also be viewed in terms of the well-known network flow problem [2] as follows. The commodity that flows through the network in this case is information in the form of sensor data or processed observations. Sensors correspond to supply nodes, estimators correspond to demand nodes, and processors correspond to intermediate, or transshipment, nodes. Communication links between nodes correspond to arcs or links between nodes of the network.

An example diagram of a decentralised information gathering system is shown in Figure 3.1a. This system topology is represented by the directed acyclic graph shown in Figure 3.1b. These diagrams could correspond, for example, to the case of two robots tracking a target using different types of sensors and with access to an off-board processing station. For target detection, each robot either processes its raw sensor data on-board or transmits the data to be processed off-board. Moreover, the robots can choose to either share raw sensor data or processed point observations instead.

Formally, the data fusion layer of a decentralised information gathering team is represented by a directed acyclic graph (DAG) $G = \{V, E\}$ where V is the set of vertices (or equivalently, nodes) and E is the set of edges or links. In the graph G, for every $i, k \in V$, if $(i, k) \in E$ then we say that k is a child node of i and i is a parent node of k. The set $\mathcal{C}(i) = \{k \in V : (i, k) \in E\}$ is the set of children of node i. Similarly, the set $\mathcal{P}(k) = \{i \in V : (i, k) \in E\}$ is defined as the set of parents of node k. We define $\mathcal{N}(i) = \mathcal{P}(i) \cup \mathcal{C}(i)$ as the neighbourhood of node i. A node with no parents is called a head node. A node with no children is called a tail node. We denote the depth of the graph G as $\kappa(G)$ defined as the number of nodes in the longest path from a head node to a tail node. The set $\bar{\mathcal{C}}(i) = \{k \in V : \text{there exists a directed path from } i \text{ to } k\}$ is referred to as the set of successors of node i. The set $\bar{\mathcal{P}}(k) = \{i \in V : \text{there exists a directed path from } i \text{ to } k\}$ is referred to as the set of ancestors of node i.

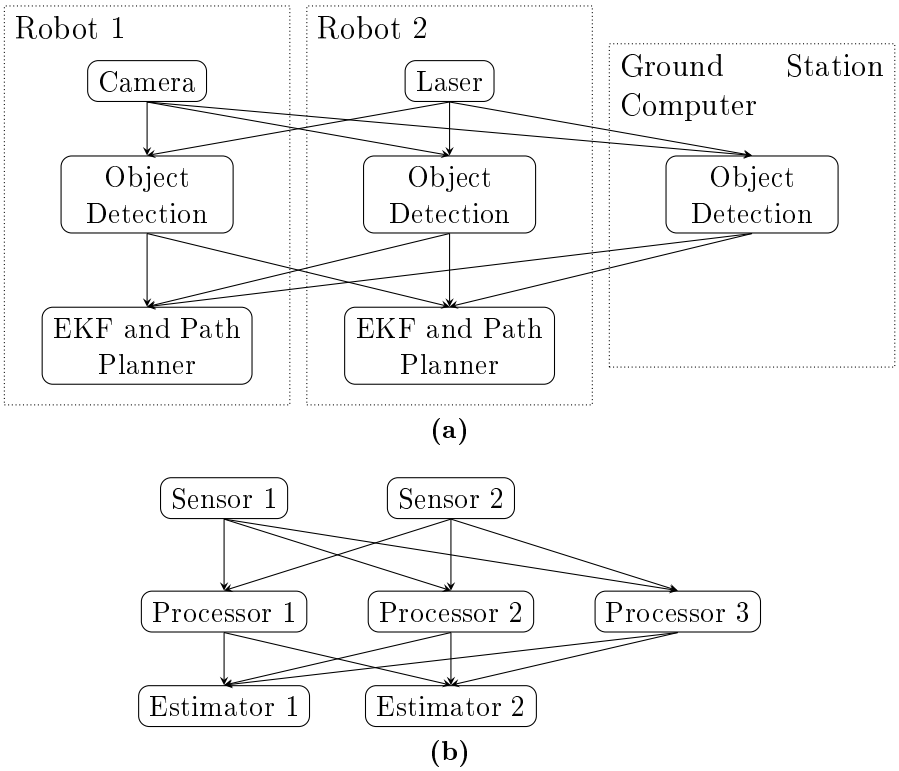


Figure 3.1 – (a) An example of a decentralised information gathering system with two robots and one off-board processor. (b) The corresponding network topology in our dynamic information flow formulation.

The set of nodes is partitioned into three mutually exclusive subsets: the set of sensors V_s which act as sources, the set of processor nodes V_p which act as intermediate nodes and the set of estimator nodes V_e which act as destination nodes. Links connect nodes in V_s to nodes in V_p, within V_p and nodes in V_p to nodes in V_e.

Sensor data is multicast from each sensor node $m \in V_s$ to all connected estimator nodes $j \in V_e$. Sensor m produces data at a fixed rate and this data is consumed by connected estimators at the same rate. To represent this production/consumption rate we introduce the variable $r_i^m(j)$ at node i for each sensor m and destination j. Variable $r_i^m(j)$ is called the inward flow and is set to sensor m's data rate if $i = m$ or else the negative of sensor m's data rate if $i = j$ or 0 otherwise. The time-averaged data rate of the flow passing through link (i, k) originating from source m and destined to j is defined as $x_{ik}^m(j)$. As an example, Figure 3.2 shows a graph of an acyclic network with a single source. The inward flow variables are indicated for the source

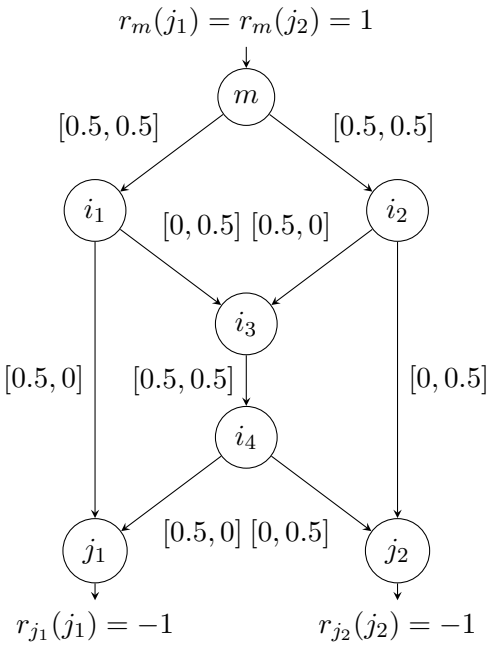


Figure 3.2 – An example routing configuration for the butterfly network. Numbers within brackets are the flow values for each link. The first value corresponds to destination j_1 and the second corresponds to destination j_2.

and destination nodes. A possible flow variable configuration is also shown for each link inside square brackets. The left entry is for j_1 and the right entry is for j_2.

A set of flow variables $\{x_{ik}^m(j) : j \in V_e\}$ will lead to an average total flow of h_{ik}^m on link (i, k). The relation between the total flow and the destination-specific flow variables will also depend on the underlying multicast implementation. Network coding allows data received at a node to be encoded or decoded. It has been shown that, with a proper choice of encoding/decoding functions, the total flow is simply the maximum flow over all destinations as defined in Equation 3.1 [1]. This relation will be assumed for the current problem formulation. The general validity of this assumption is discussed further in Section 4.1.3.

$$h_{ik}^m = \max_{j \in V_e} x_{ik}^m(j) \tag{3.1}$$

Communication load, computation load and sensor observation utility induce a net link cost of c_{ik}^m per unit of data flow from source m passing through link (i, k). The

link cost is multiplied by the total flow h_{ik}^m to obtain the total link cost arising from source m. Summing over all sources, link (i,k) has a total cost of $\sum_m c_{ik}^m h_{ik}^m$.

Sensor observations induce a reward when reaching an estimator. In order to represent this reward, the information value of sensor m to estimator j is subtracted from the cost of each link incident to j.

Because a sensor observation may have little value for a given estimator, the system requires a mechanism by which a sensor can decide not to send any data to a certain destination. We model this option by adding a virtual zero-cost link directly from each sensor to all connected estimators.

We now define the general dynamic information flow problem as follows. Given link costs $\{c_{ik}^m\}$ and inward flow rates $\{r_i^m(j)\}$, choose the set of flow variables $\{x_{ik}^m(j)\}$ such that the total cost summed over all links in the network is minimised subject to constraints. Link costs and constraints may vary over time.

3.2 Min-Cost-DIF

We define the first concrete form of the general problem, *min-cost-DIF*, according to the constrained optimisation defined in (3.2-3.5). Information value, communication and computation resource demand are represented using link costs. This formulation is appropriate for situations where the relative costs between the items are known *a priori*.

$$\text{minimise} \qquad \sum_{(i,k)\in E, m\in V_s} c_{ik}^m h_{ik}^m \tag{3.2}$$

$$\text{subject to} \qquad x_{ik}^m(j) \geq 0 \tag{3.3}$$

$$h_{ik}^m = \max_{j\in V_e} x_{ik}^m(j) \tag{3.4}$$

$$\sum_{l\in\mathcal{P}(i)} x_{li}^m(j) - \sum_{k\in\mathcal{C}(i)} x_{ik}^m(j) + r_i^m(j) = 0 \tag{3.5}$$

The first constraint given by Inequality 3.3 ensures that flow is always positive. The second constraint given by Equation 3.4 represents the multicast condition and the third constraint given by Equation 3.5 ensures that the sum of all inward and outward flow at a node is zero.

In min-cost-DIF, link costs may change over time due to changes in communication and processing costs as well as changes in sensor utility. For example, robots may move closer or further away from each other, resulting in a change in communication costs. Sensor viewpoint may also change, leading to a change in the value of on-board sensor observations.

3.3 Threshold-DIF

We introduce a second problem variant, *threshold-DIF*, to represent the case where the correct scale between communication costs, computation costs and information value is *not* known *a priori*. In this case, communication bandwidth and processing power are viewed as limited resources. The goal of threshold-DIF is thus to maximise information gain subject to communication bandwidth and processing power constraints.

We augment the optimisation problem (3.2-3.5) to include the two additional constraints (3.6-3.7) and define three additional input parameters, ν_{ik}^m, C_{ik} and K_s, to represent resource capacity limits. Constraint 3.6 bounds the weighted sum of flows originating from different sensors to respect a fixed capacity C_{ik}. The summation over all sensors in V_s is required since a link may carry messages originating from different sensors. Weights $\{\nu_{ik}^m\}$ are used to scale flow values h_{ik}^m on a per-link basis. For example, variations in required communication bandwidth due to link quality can be modelled by assigning appropriate values to $\{\nu_{ik}^m\}$. Similar to link costs, these variables may change over time.

$$\sum_{m \in V_s} \nu_{ik}^m h_{ik}^m \leq C_{ik} \tag{3.6}$$

$$\sum_{(i,k)\in\mathcal{S}_s} \sum_{m} \nu_{ik}^{m} h_{ik}^{m} \leq K_s \tag{3.7}$$

To motivate the constraints introduced by threshold-DIF, consider the network diagram in Figure 3.3 which is a subset of the diagram in Figure 3.1. The limit C_{ik} in Constraint 3.6 for instance might represent the limit on the image-processing frame-rate applied to the link from the camera to the object detection module in Robot 1. Now, if we consider the link from the object detection module in Robot 1 to Robot 2, it may hold processed observations originating from both the camera and laser. To account for the possible discrepancy in the data rates from these sensors, the weights ν_{ik}^{m} are chosen to accordingly. To motivate Constraint 3.7, we need to consider links that share a common resource.

Referring once more to Figure 3.3, Constraint 3.7 may be used to represent the following resource constraints. If the two robots in this example exclusively use wireless communications to share observations, then all four links that cross the robot boundaries share a common resource (the wireless communication medium). This constraint is indicated in the figure by the dashed link. Moreover, if each robot only uses one computer for all processing requirements then the links from both sensors to each of the object detection modules share another common resource, the on-board processing computer. These constraints are indicated in the figure by dotted links. This class of constraints, which we call *inter-link* constraints, is represented by Equation 3.7, where K_s is a fixed upper bound on resource s and $\mathcal{S}_s$ is the set of links sharing resource s. Again, the flow rates are weighted because inter-link constraints impose bounds on the total flow across different links with data from different sources. For example, the link holding raw images from the camera will typically hold higher data rates than that from the laser, yet the difference in this data rate might decrease after processing. Links involved in such constraints could either be emanating from different nodes, as shown in the example in Figure 3.3, or from the same node, when a node sends to many nodes using the same medium.

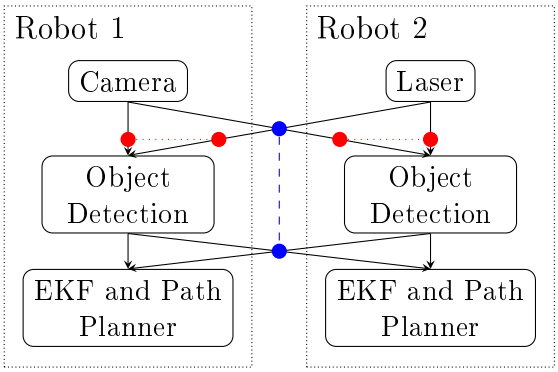


Figure 3.3 – An example of a system with inter-link constraints. Links tagged with the dashed line share a wireless communication medium, while those tagged with a dotted line share a common processing resource.

3.4 Negotiation-DIF

The third problem addressed in this thesis is negotiation-DIF. Negotiation-DIF extends DIF to include communication efficiency in decision making. The objective of negotiation-DIF is to minimise communication costs for data fusion and decision making simultaneously.

In negotiation-DIF, a decision making layer is added to the DIF formulation. The decision making layer includes controller nodes that take as input the state of an estimator and produce as output an action decision. The controller nodes may only require a subset of the estimator state. The controllers also negotiate collaterally over the available communication medium to achieve a cooperative team decision. Figure 3.4 is an example diagram of a decentralised information gathering system with the decision making layer shown.

The decision making layer does not retain the directed acyclic property of the DIF network. Therefore, the approaches to min-cost-DIF and threshold-DIF are not applicable. The objective of negotiation-DIF can be achieved through a partially observable Markov decision process (POMDP) formulation; however, this formulation leads to an intractable problem. Instead, we aim to solve a simpler problem by making the following assumptions. We assume that the information gathering problem layers decentralised data fusion (DDF) and decentralised decision making (DDM) are de-

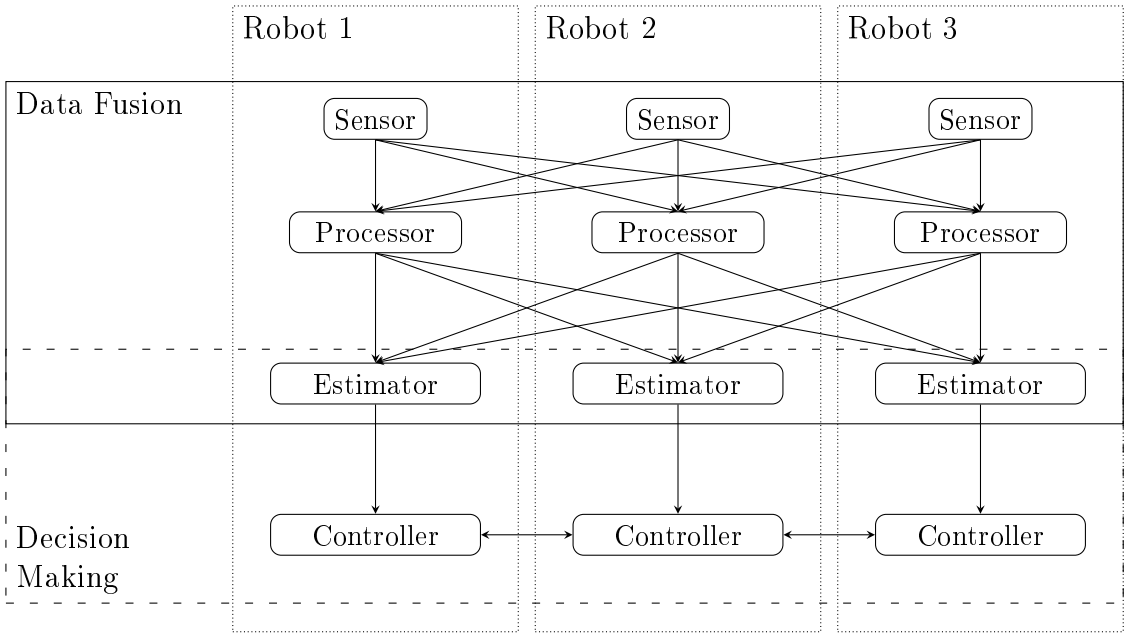


Figure 3.4 – Communication layers of a decentralised information gathering system.

coupled. We further assume that the DDF layer has a DIF representation. The final assumption is that the DDM layer includes negotiation that occurs at an adjustable rate.

Since the two layers share a common resource, the wireless communication medium, the usage of this resource should be regulated collectively. For simplicity, we resort to link costs and adopt the min-cost-DIF formulation for the data fusion layer. The two layers use the same values for link costs but communication efficiency is optimised separately.

The separation between DDF and DDM is closely related to the issue of separation between estimation and control studied by stochastic-control theorists [101]. This separation retains equivalence for some special cases. Nevertheless, it is a common assumption for most practical information gathering implementations. The separation means that communication for each layer can be adjusted separately, avoiding the difficulty of balancing the utility of communication between the two layers. However, we attempt to retain some coupling between the layers by assigning the same link

costs. Furthermore, as delineated later in Section 5.4, the solution at the DDM layer also attempts to improve the sensor utility estimate for the DDF.

Negotiation-DIF is defined abstractly as follows. Given the two sets of equivalent link costs for DDF and DDM communication, determine the communication rates for DDM and the routing variables $\{x_{ik}^m(j)\}$ of the min-cost-DIF network based on the sensor utility. The comms-linear-quadratic (LQ) problem defined in Section 3.4.1 permits a concrete problem definition for communication efficiency at the decision making layer.

3.4.1 Comms-LQ

Comms-LQ is a communication-efficient decision making problem formulation for LQ systems. The objective of comms-LQ is to obtain a communication-efficient feedback control policy of an *LQ team* based on the communication costs between robots. Although comms-LQ is targeted to LQ systems, it can be extended to non-LQ systems through local LQ approximations.

An LQ team is a decentralised team of robots with the following properties. The robots have decoupled linear dynamics. The dynamics of robot i are given by Equation 3.8, where x_i is the state vector, and u_i is the control vector. The team has a global quadratic cost defined in Equation 3.9, which is known to all robots. The team state vector is denoted by x, and u is the team control vector. The team control cost R is assumed to be block diagonal. Since the robot dynamics are decoupled, the team dynamics matrices (A, B) will also be block diagonal.

$$\dot{x}_i = f_i(x_i, u_i) = A_i x_i + B_i u_i \tag{3.8}$$

$$J = \frac{1}{2} \int x^T Q x + u^T R u \, \mathrm{d}t \tag{3.9}$$

We assume that the control policy has the form given by Equation 3.10 where K is a feedback gain matrix which is always positive semi-definite. Since R and B are block

diagonal, the control policy for robot i is given by Equation 3.11 where K_i is the i-th row of the matrix K and R_i is the i-th block of R.

$$u = -R^{-1}B^T Kx \tag{3.10}$$

$$u_i = -R_i^{-1}B_i^T K_i x \tag{3.11}$$

In comms-LQ, we assume that the required communication rate between agents can be directly determined from the values of the feedback gain matrix K. This assumption is not strictly correct; however, it serves as a useful approximation. Furthermore, the interpretation of this relation is not unique. One interpretation of this relation is achieved by taking the absolute value of the terms in K as proportionality weights for the communication rates required between robots.

In general, the obtained matrix K is dense. This means that the control vector for each robot depends on the entire team state. For a decentralised system, this means that each robot must continuously receive state information from all robots. To improve communication efficiency, we would like to reduce the required communication based on a given set of communication costs.

To this end, define the symmetric communication cost matrix $U \in \mathcal{S}^n$ with the same dimensions as K. Each element in U is a positive communication cost of the corresponding element in K. This cost corresponds to the communication cost of the link represented by the element in K. Further define the diagonal matrix $\bar{U}$ which contains the lower-triangular elements of U placed along its diagonal. Finally, define the vector $\text{vec}(K)$ as the vector which consists of the lower-triangular elements of K put in vector form.

The comms-LQ problem is formally defined by Problem 3.13. The objective is given by Equation 3.12. Given a set of communication link costs U, state cost Q, control cost R, the solution of comms-LQ is to find a feedback matrix K that minimises Objective 3.12. By assumption, this matrix K provides an appropriate communication policy. Problem 3.13 is a difficult problem to solve with no tractable solution to the best of the author's knowledge. In Section 5.2, we present an approximate solution

based on the linear matrix inequality (LMI) formulation of the LQ optimal control problem.

$$J_c = \frac{1}{2}\left\{\int x^T Q x + u^T R u \,\mathrm{d}t + vec(K)^T \bar{U} vec(K)\right\} \tag{3.12}$$

$$\begin{aligned} &\text{minimise} && (3.12) \\ &\text{subject to} && (3.8) \text{ and } (3.10) \end{aligned} \tag{3.13}$$

3.4.2 Problem Formulation

With the definition of comms-LQ, we can concretely define the negotiation-DIF problem. Since the comms-LQ problem formulation is limited to LQ teams, we assume a local LQ approximation of the information gathering problem. This assumption entails both the attainability of an LQ approximation as well as its representativeness. Another key assumption, which follows from the decoupling assumption mentioned in the negotiation-DIF graph representation, is that the decision making layer assumes unconstrained information flow at the data fusion layer. Negotiation-DIF is now defined as follows. Given the sensor utility, common communication costs for both DDF and DDM and a local LQ approximation, solve the problem (3.2-3.5) and Problem 3.13. Figure 3.5 schematically shows how the same link cost is assumed for both layers.

3.5 Sensor Utility

Sensor utility is one of the main inputs into the DIF problem because the goal in DIF is to *maximise* information gathering performance under resource constraints. Sensor utility is a measure that includes the relative importance of sensor data with respect to a specific estimator. From an information gathering perspective, this importance corresponds to the predicted entropy reduction or mutual information that is commonly used as a decision metric in that domain.

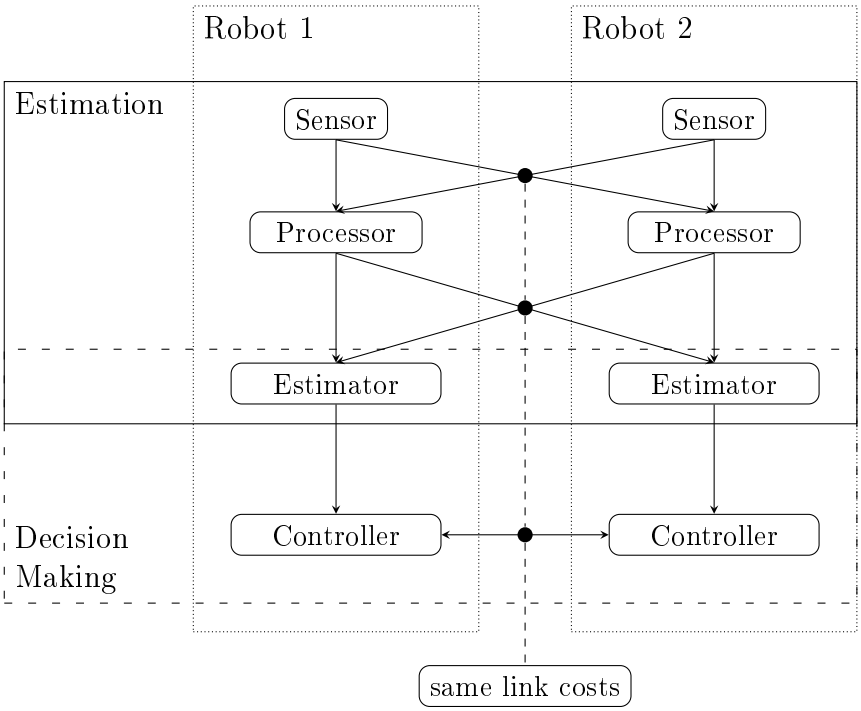


Figure 3.5 – Layout of negotiation-DIF.

Computing the exact sensor utility is possible through the decentralised partially observable Markov decision process (Dec-POMDP) formulation. However, given that Dec-POMDPs are NEXP-complete, for the demonstrations shown in this thesis, we employed a simple myopic approximation. This approximation is presented in Section 4.1.3 with a discussion of the validity of this approximation provided in Section 4.4. This discussion shows that efficiently computable theoretical bounds for sensor utility can be too conservative to be of practical importance. It also shows that the myopic approximation, on the other hand, can be relatively accurate when planning occurs over a fixed finite horizon.

In DIF, sensor utility is represented by a value that is subtracted from the cost of links incident to estimators. The DIF problem formulation assumes an objective function that is linear in the flow-rate variables. The resultant cost contribution of source m on each link (i, k) is equal to $c_{ik}^m h_{ik}^m$. This assumption implies that the reward at an estimator should be proportional to the flow rate. This assumption is not valid in

general; however, with the sensor utility being continuously updated, the error caused by this assumption remains practically acceptable.

An important concept is the evaluation of sensor utility based on the change sensor observations induce in action decisions. Interestingly, this concept intersects with the idea of non-myopic sensor utility estimation since it considers the future effect of an observation. In Section 5.4, we show how the sensor utility estimate can be improved when coupled with the decision making layer through the negotiation-DIF formulation.

3.6 Resource Costs and Limits

DIF relies on the specification of communication and/or computation costs or limits. The intent is that costs represent contention for resources while limits represent resource constraints. Obtaining a model that infers these costs or limits from the underlying hardware configuration has been studied by various researchers [36, 85], yet this issue is outside the scope of this thesis. The crude model employed by the demonstrations shown in this thesis is specified in Section 4.1.3.

3.7 Summary

This chapter presented the three DIF problems for which we aim to introduce a solution in this thesis. Solutions to the first two DIF variants, min-cost-DIF and threshold-DIF are presented in Chapter 4. A solution to the third variant is presented in Chapter 5.

Chapter 4

Min-Cost-DIF and Threshold-DIF

In this chapter, we present algorithms that solve the min-cost-DIF and threshold-DIF problems defined in Section 3.1. The solution to min-cost-DIF is presented in Section 4.1. Section 4.2 then presents a distributed optimisation method required by our solution to threshold-DIF which appears in Section 4.3. Section 4.4 provides a brief empirical analysis on the myopic approximation of sensor utility used by our implementation of the algorithms used in this chapter.

4.1 Min-Cost-DIF

In this section, we present a message-passing algorithm that solves the min-cost-DIF problem. We introduce a mapping that transforms an instance of min-cost-DIF into an instance of multicast network routing, prove equivalence and show that an algorithm that was originally developed for multicast network routing also finds an optimal solution to min-cost-DIF. We then describe our implementation of this algorithm in the context of min-cost-DIF.

4.1.1 Min-Cost-DIF Using Multicast Network Routing

An instance of min-cost-DIF can be transformed into an instance of multicast network routing [19, 103] as follows. The flow variable $x_{ik}(j)$ is replaced with $t_i(j)\phi_{ik}(j)$, where $t_i(j)$ is the total flow passing through i and destined to j while $\phi_{ik}(j)$ is the routing variable for link (i,k); more specifically, it is the fraction of $t_i(j)$ that is routed to k.

Following this change of variables, the resulting formulation is given by the optimisation problem (4.1-4.5). Constraint 4.3 states that the sum of the routing variables for each node is equal to one, while Equation 4.4 and Equation 4.5 are equivalent to Equation 3.4 and Equation 3.5 respectively.

$$\text{minimise} \quad \sum_{(i,k)\in E} c_{ik} h_{ik} \tag{4.1}$$

$$\text{subject to} \quad \phi_{ik}(j) \geq 0 \tag{4.2}$$

$$\sum_{k\in\mathcal{C}(i)} \phi_{ik}(j) = 1 \tag{4.3}$$

$$h_{ik} = \max_j t_i(j)\phi_{ik}(j) \tag{4.4}$$

$$t_i(j) = r_i(j) + \sum_{l\in\mathcal{P}(i)} t_l(j)\phi_{li}(j) \tag{4.5}$$

Given this mapping, existing algorithms for multicast network routing can be applied. Here we summarise one such algorithm, originally presented in [19]. The algorithm is based on message passing and relies on obtaining the *marginal cost* $\delta_{ik}(j)$ for each link. The marginal cost is the rate at which the total cost increases due to a unit increase in flow along that link and is given by Equation 4.6.

$$\delta_{ik}(j) = \begin{cases} c_{ik}/n + \sum_{l \in \mathcal{C}(k)} \phi_{kl}(j)\delta_{kl}(j) & \text{if } t_i(j)\phi_{ik}(j) \text{ and } n-1 \text{ other flows on link } (i,k) \text{ are the maximum} \\ \sum_{l \in \mathcal{C}(k)} \phi_{kl}(j)\delta_{kl}(j) & \text{otherwise} \end{cases} \tag{4.6}$$

Recall from Chapter 3 that in the context of information gathering sources correspond to sensors, processors to intermediate nodes and estimators to destination nodes. Min-cost-DIF can be solved for each source m independently and in parallel. The full problem can be decomposed into independent sub-problems, one for each source, since the objective is additive and there are no inter-source constraints. This is evident from the problem formulation (3.2-3.5). Therefore, for simplicity of notation the subscript m is dropped from all variables in this section.

At the start of the algorithm, the routing variables $\{\phi_{ik}(j)\}$ are initialised arbitrarily such that they obey Constraints 4.2 and 4.3. The routing variables are then repeatedly updated such that after iteration t the routing variables are set as $\phi_{ik}^{t+1}(j) = \phi_{ik}^{t}(j) + \Delta\phi_{ik}(j)^t$. The update direction $\Delta\phi_{ik}(j)^t$ is defined in Equation 4.7. The set E_j is the set of edges belonging to the subgraph containing the ancestors of destination j and $\delta_{i,\min}(j) = \min_k \delta_{ik}(j)$.

$$\Delta\phi_{ik}(j)^t = \begin{cases} 0 & \text{if } (i,k) \in E_j \\ -\min\left\{\phi_{ik}^t(j), \dfrac{\alpha(\delta_{ik}(j) - \delta_{i,\min}(j))}{t_i(j)}\right\} & \text{if } \delta_{ik}(j) \neq \delta_{i,\min}(j) \\ \sum_{\substack{\delta_{ip}(j) \neq \\ \delta_{i,\min}(j)}} \Delta\phi_{ip}(j)^t & \text{if } \delta_{ik}(j) = \delta_{i,\min}(j) \end{cases} \tag{4.7}$$

The algorithm runs synchronously. First, the head nodes send messages with their flow contributions to their children. Once a node receives messages from all of its parents, it passes the message to its own children and so forth. The purpose of

this downward sweep is to allow nodes to compute the flow of the current routing configuration. The flow values are necessary to compute the marginal costs required in the upward sweep. The downward sweep is followed by an upward sweep during which the marginal costs are computed according to Equation 4.6 and the routing variables are updated according to Equation 4.7. The downward and upward sweeps are decentralised, synchronous and are guaranteed to visit every node. Their sequence is dictated by Algorithm 4.1. The synchronicity property of Algorithm 4.1 is proved in Lemma 4.1.

Due to possible changes in link costs, this message passing optimisation runs continuously throughout system operation. As the system configuration changes, link costs are updated with new values. To ensure convergence, the interval between updates is set to an adequate time period. Further details on the appropriate length of the interval between updates can be found in Section 4.1.3.

Algorithm 4.1: Synchronous message passing on DAGs

```
1: For node i
2: if i is a head node then
3:    Perform a downward update and send downward message to children
4: end if
5: loop
6:    if a downward message is received from all parents then
7:       Perform a downward update and send downward message to children
8:       if i is a tail node then
9:          Perform an upward update and send upward message to parents
10:      end if
11:   end if
12:   if an upward message is received from all children then
13:      Perform an upward update and send upward message to parents
14:      if i is a head node then
15:         Perform a downward update and downward message to children
16:      end if
17:   end if
18: end loop
```

Lemma 4.1 (Graph traversal synchronicity)**.** *In Algorithm 4.1, after node i has performed its t-th downward update and before forwarding this update:*

1. *Node i and all of its successors would have performed exactly $t-1$ upward updates*
2. *All of its ancestors would have performed exactly t downward updates*

Proof. Suppose one of node i's successors has performed $t' > t-1$ upward updates. This means that at least one tail node in the successors has performed t' downward updates. This is impossible because node i has only yet forwarded $t-1$ downward messages. Since node i has performed t updates then its ancestors have performed at least t updates. Now, suppose one of node i's ancestors has performed t'' updates where $t'' > t$. Then, the head nodes in the ancestry have performed at least t'' updates. This in turn means that they have performed $t''-1$ upward updates which means the tail nodes have performed $t''-1 > t-1$ updates, which is impossible as just shown. This means that node i has forwarded at least $t'-1 \geq t$ downward updates, which leads to a contradiction. □

Due to observation rewards, negative costs may be assigned to links incident to an estimator $j \in V_e$. These negative costs are handled within the framework of the multicast network routing algorithm by solving an equivalent problem. A large enough constant $\bar{c}$ is added to all $\{c_{ij} : j \in V_e\}$ to obtain a set of non-negative cost variables $\{c'_{ik}\}$ defined in (4.8).

The equivalence of solving the optimisation problem (4.1-4.5) to solving the problem with cost variable c'_{ik} instead of c_{ik} is proved in Theorem 4.1.

$$c'_{ik} = \begin{cases} c_{ik} + \bar{c} & \text{if } k \in V_e \\ c_{ik} & \text{otherwise} \end{cases} \tag{4.8}$$

Theorem 4.1 (Multicast routing with negative terminal links)**.** *Replacing link cost c_{ik} in problem (4.1-4.5) with c'_{ik} defined in Equation 4.8 results in another problem instance equivalent to the original problem.*

Proof. For every link (i, j) such that $j \in V_e$, Equation 4.4 turns into Equation 4.9 instead since the only flow that should run along that link is the flow destined to estimator j.

$$h_{ij} = t_i(j)\phi_{ij}(j),\ \forall j \in V_e,\ \forall (i,j) \in E \tag{4.9}$$

After adding $\bar{c}$ to the cost of these links to obtain $\{c'_{ij}\}$, a total of $\bar{c}\sum_{(i,j)} h_{ij}$ is added to the problem objective. Since $\bar{c}$ is constant, we now proceed to prove that $\sum_{(i,j)} h_{ij}$ is constant. After substituting h_{ij} from Equation 4.9, we obtain Equation 4.10.

$$\sum_{(i,j)} h_{ij} = \sum_{j \in V_e} \sum_{i \in \mathcal{P}(j)} t_i(j)\phi_{ij}(j) \tag{4.10}$$

The right hand side of the equation is equal to the total flow arriving at a destination node summed over all destinations. By definition, this flow is equal to the source flow multiplied by the number of destinations and hence is constant. □

4.1.2 Analysis

Subject to the choice of step size parameter α, the multicast routing algorithm is guaranteed to converge to the global optimum [19]. Since a DAG contains no loops by definition, no contingencies are required to avoid routing loops. By Theorem 4.1, a min-cost-DIF instance with negative costs on links to an estimator can still be solved using the multicast routing algorithm by adding a sufficiently large positive constant to all such links.

The running time of multicast network routing with network coding is not explicitly provided in [19, 103] but is implied to be polynomial in the size of the network. In practice we have observed a polynomial rate of increase as a function of network size, as shown in Section 4.1.3 below.

4.1.3 Implementation and Scalability

Implementing multicast routing for min-cost-DIF involves three main challenges. First, we need to allow all nodes to find the available sources and destinations in a decentralised manner. Second, we need to ensure that the nodes have a suitable mechanism to compute any changing input parameters. Finally, we must choose a suitable multicast policy to implement the chosen flow rates $\{x_{ik}(j)\}$.

Each node must find the set of sources and destinations to which it is connected in the network. Initially, each node is aware of its direct neighbours only. By performing only one downward sweep and one upward sweep of message passing described in Algorithm 4.1, each node can obtain the list of sources and destinations to which it is connected. In our implementation, the downward messages contain the set of source identities received so far and the upward messages contain the set of destination identities received so far.

Link Costs

Link costs are continuously computed due to changes in the team configuration throughout the progress of its mission. In our implementation, communication costs are simply set proportional to inter-robot distance. Processing costs are assumed to be constant throughout the system operation. The DIF formulation does not specify a particular model for communication costs. Many complicated models for robotics applications have been suggested by various researchers [71, 85]. These models take into consideration various phenomena that affect communication quality including noise, attenuation and multipath fading. The investigation of different models is outside the scope of this thesis. Therefore, we assume a simple distance-proportional model while noting that the DIF formulation is not restricted to any particular model. The distance-proportional model simply sets the communication cost of a link between robots proportional to the inter-robot distance. The constant of proportionality is set to a value fixed by the operator.

Sensor Utility

In our implementation, the utility of a sensor is evaluated based on the improvement it induces in the estimate of each estimator. Sensor utility depends both on the sensor and on the current state of an estimator. Specifically, an estimator approximates a sensor utility by evaluating the most recent sensor observation received. The value of the observation is computed as the reduction in entropy realised by fusing the observation into the estimator. For a Gaussian representation, this value is proportional to the reduction in the log-determinant of the covariance matrix after an observation.

Sensor utility can be difficult to compute since each estimator must receive observations from a sensor in order to evaluate this utility. We maintain sensor utility values dynamically though an exploration-exploitation model. Each node obeys the chosen flow rate $x_{ik}(j)$ with probability $(1-\epsilon)$ (exploitation) and switches to another randomly selected flow rate with probability ϵ (exploration). The value of ϵ is set to a small positive number less than one.

Flow Rates

The chosen flow rates $\{x_{ik}(j)\}$ are implemented using a multicast policy that determines how the inward flow of messages at a node is distributed amongst its children. In our problem formulation, we assume that network coding is used. For small-sized networks, network coding can introduce unnecessary complication with little performance advantage [53, 59]. As an alternative, multicast routing can be implemented without network coding by using randomisation. The probability of sending a given inward message along a given outward edge is set proportional to flow variable $x_{ik}(j)$. In this case, the average total flow h_{ik} through the link for a source flow rate of r is given by Equation 4.11 instead of the network coding relation given in Equation 3.1.

$$\bar{h}_{ik} = r - \prod_{j}(r - x_{ik}(j)) \tag{4.11}$$

For a given set of flow variables, randomisation will result in higher total flow through a link. The extra capacity required in comparison to network coding is given by the relation in Equation 4.12. From the relation, we deduce that the gap is zero if the maximum flow variable over a link is equal to either zero or the source flow and that the gap is less significant when there are fewer destinations. Therefore, for ease of implementation we use randomisation to implement the multicast policy. However, the total flow is still approximated by Equation 3.1 since Equation 4.11 otherwise leads to a non-convex problem. We found this approximation to be valid in practice.

$$\begin{aligned} \bar{h}_{ik} - h_{ik} &= r - h_{ik} - \prod_j (r - x_{ik}(j)) \\ &\leq r - h_{ik} - (r - \max_{j \in V_e} x_{ik}(j))^{N_j} \\ &= r - \max_{j \in V_e} x_{ik}(j) - (r - \max_{j \in V_e} x_{ik}(j))^{N_j} \end{aligned} \tag{4.12}$$

Scalability

To demonstrate the scalability of the algorithm, we performed an empirical study of the convergence time for a given set of link costs. This study, which gives a convergence time estimate, also gives further insight into the time required between link cost updates.

We evaluated the convergence time in min-cost-DIF through a simulated network using randomised but fixed link costs. Our simulated network includes one sensor and a variable number of processors and estimators where the number of processors is always one more than the number of estimators. Results of the simulation for an increasing number of nodes are shown in Figure 4.1. The convergence condition is satisfied when the change in the solution variables is below a certain threshold.

Convergence time depends both on the number of iterations required until convergence and the time expended in each iteration. The number of iterations to convergence is hardware independent and the results shown in Figure 4.1 indicate that the number of iterations to convergence is a sub-linear function of the network size. The time

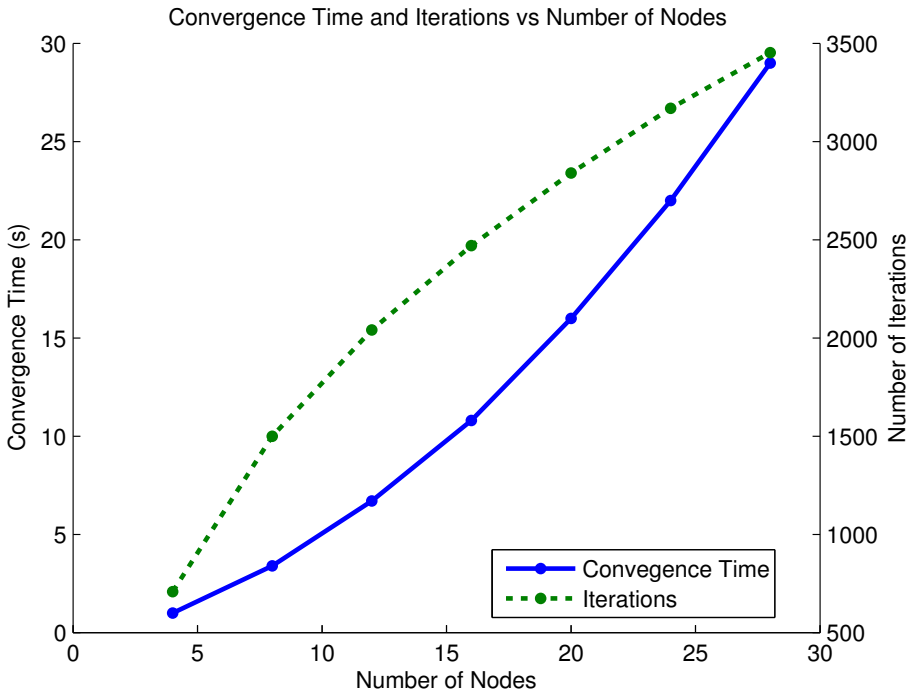


Figure 4.1 – Execution time and convergence time of the message passing optimisation as a function of the number of nodes in a simulated network. The simulated network includes one sensor and an increasing number of processors and estimators where the number of processors is always one more than the number of estimators.

required for each iteration involves computing routing updates and transmitting the updated values. The time complexity of the routing updates in Equation 4.6 and Equation 4.7 is polynomial in the size of the network. Transmission time, on the other hand, is typically linear in the size of the transmitted message which in turn is proportional to the size of the network.

The convergence times shown do not account for communication delay. To estimate such delay in practice, we observed from experiment data (experiment shown in Section 6.4.1 specifically) that the time for one iteration over wireless networks is typically less than 100 milliseconds. This value corresponds to networks concurrently being utilised for transmission of sensor data and processed observations. Based on this estimate and the number of iterations at convergence, we can estimate typical values for convergence time for networks from 5 to 10 nodes to be between 50 and 150 seconds.

This empirical analysis indicates that the running time of our algorithm is polynomial in the size of the network for typical problem instances of interest. We observed that the number of iterations required until convergence is sub-linear in the number of nodes, and the time complexity of each iteration is polynomial in the number of nodes. In practice, we found iteration times to be dominated by communication delay resulting in the convergence time indicated above. We envisage that this period can be reduced by employing various quality of service protocols, yet this solution is deferred to future work.

Based on the above analysis, we can now specify nominal values for the interval between link cost updates. The frequency of link cost updates is set such that the multicast routing algorithm has sufficient time to converge. In our implementation, we chose fixed values between 50 and 150 seconds. However, we note that while the optimisation algorithm is iterating, a valid routing is available and information can continuously flow through the network. The frequency of link cost updates determines how reactive the algorithm is to changes in estimated sensor utility, and thus the importance of a higher update frequency would be to handle situations where sensor utility changes rapidly. We leave consideration of this case to future work.

4.2 Distributed Optimisation for Threshold-DIF

In this section, we introduce a distributed optimisation method that will form the basis of our solution to threshold-DIF. The method is a distributed version of the alternating direction method of multipliers (ADMM) which we call *the distributed alternating direction method of multipliers (DADMM)*. DADMM solves distributed non-smooth constrained convex optimisation problems with a DAG structure. Thresold-DIF is a distributed optimisation problem since its objective is a sum of local objectives and it only has neighbour-to-neighbour constraints. It is non-smooth due to the linear objective, and it has the structure of a DAG by definition.

First, a brief introduction to ADMM is provided for convenience. Then, DADMM is presented by first defining the set of problems it solves, then describing the straight-

forward extension to inequality constraints and finally providing the algorithm followed by an analysis of complexity.

4.2.1 ADMM

ADMM is a method that allows for the decomposition of the optimisation of non-smooth convex problems. For convenience, this section provides a brief summary of ADMM based on [10].

ADMM solves optimisation problems of the form given by Problem 4.13. The objective is assumed to be a sum of two proper convex functions f_1 and f_2 where the first is a function of the vector z_1 and the other is a function of the vector z_2. We refer to z_1 as the *primary* vector variable and to z_2 as the *secondary* vector variable. The augmented Lagrangian of the problem is given by Equation 4.14.

$$\begin{aligned} &\text{minimise} && f_1(z_1) + f_2(z_2) \\ &\text{subject to} && A_1 z_1 + A_2 z_2 = b \end{aligned} \tag{4.13}$$

$$\begin{aligned} L(z_1, z_2, y) =& f_1(z_1) + f_2(z_2) + y^T(A_1 z_1 + A_2 z_2 - b) \\ &+ (\rho/2)\|A_1 z_1 + A_2 z_2 - b\|_2^2 \end{aligned} \tag{4.14}$$

ADMM is summarised by the updates shown in Equations 4.15. Each iteration involves three updates. The primary update minimises the Lagrangian about z_1, the secondary update minimises the Lagrangian about z_2 and the third updates the Lagrangian variable y. Proof of the convergence of the updates is shown in [10].

$$\begin{aligned} z_1^{k+1} &:= \arg\min_{z_1} L(z_1, z_2^k, y^k) \\ z_2^{k+1} &:= \arg\min_{z_2} L(z_1^{k+1}, z_2, y^k) \\ y^{k+1} &:= y^k + \rho(A_1 z_1^{k+1} + A_2 z_2^{k+1} - b) \end{aligned} \tag{4.15}$$

4.2.2 Problem Formulation

The general form of optimisation problems that are solved by DADMM is described as follows. Consider the DAG $G = \{V, E\}$ defined in Section 3.1. Attach to each node $i \in V$ a vector variable x_i and a proper convex function $f_i(x_i)$, that is not necessarily smooth. Node i can have constraints with its parents as per Equation 4.17 where g_i is an affine function. The notation x_U where $U = \{i_1, ..., i_n\} \subset V$ is defined as the concatenation of all vectors x_i such that $i \in U$, i.e. $x_U = (x_{i_1}, ..., x_{i_n})$. Function g_i is interpreted as a vector valued function with its dimension indicating the number of constraints n_g^i. The goal of DADMM is to solve the optimisation problem (4.16-4.17).

$$\text{minimise} \qquad \sum_{i \in V} f_i(x_i) \tag{4.16}$$

$$\text{subject to} \qquad g_i(x_i, x_{\mathcal{P}(i)})) = 0, \ \forall i \in V \tag{4.17}$$

4.2.3 Inequality Constraints

The standard form of ADMM does not include inequality constraints. Thus, we have only included equality constraints g_i in the problem definition. This is a non-restrictive assumption since by adding extra variables, inequality constraints can be transformed into equality constraints as we will show. Suppose that instead of g_i we have a function $\bar{g}_i$ that is required to satisfy Inequality 4.18.

$$\bar{g}_i \leq 0 \tag{4.18}$$

By adding a slack variable p_i, this inequality constraint becomes an equality constraint plus a non-negative constraint on p_i as shown in Constraints 4.19. The slack variable p_i can be viewed as a variable belonging to a virtual parent node whose objective is an indicator function that is zero when p_i is non-negative and infinity otherwise.

$$\begin{aligned} g_i = \bar{g}_i + p_i = 0 \\ p_i \geq 0 \end{aligned} \tag{4.19}$$

The distributed nature of DADMM allows p_i to be optimised independently. The solution of the optimisation over p_i is given by Equation 4.20. Although Equation 4.18 may also be substituted with an indicator function, it cannot be optimised independently through a simple projection since it involves variables that are included in other objectives.

$$p_i := \max\{-\bar{g}_i, 0\} \tag{4.20}$$

4.2.4 DADMM

DADMM consists of a preliminary decentralisation step followed by the main optimisation process. The decentralisation step is only performed once during which the optimisation problem is modified, through the addition of variables and constraints, such that it only requires neighbour-to-neighbour communication. In the optimisation process, message passing and optimisation updates run in a sequence that enforces decentralisation while retaining equivalence to centralised ADMM.

The decentralisation step modifies Constraint 4.17. For every vector x_i where $i \in V$, a mirror vector $\bar{x}_i$ is introduced. The vector $\bar{x}_i$ acts as an interface for all other nodes. Any child node k that has a constraint including x_i replaces x_i with a local copy $\tilde{x}_i^k$ and an equality constraint between $\tilde{x}_i^k$ and $\bar{x}_i$ is added. Symmetrically, from node i's perspective $x_{\mathcal{P}(i)}$ is replaced with $\tilde{x}_{\mathcal{P}(i)}$. Therefore, from node i's perspective, Constraint 4.17 is replaced with the set of constraints (4.21-4.23).

$$g_i(x_i, \tilde{x}^i_{\mathcal{P}(i)}) = 0 \tag{4.21}$$

$$x_i - \bar{x}_i = 0 \tag{4.22}$$

$$\tilde{x}^i_{\mathcal{P}(i)} - \bar{x}_{\mathcal{P}(i)} = 0 \tag{4.23}$$

The new constraints are assigned the following Lagrangian multiplier vectors. The Lagrangian multiplier λ_i is associated with Constraint 4.21, μ_i is associated with Constraint 4.22 and $\eta^i_{\mathcal{P}(i)}$ is associated with Constraint 4.23.

All the above constraints and variables except $\bar{x}_{\mathcal{P}(i)}$ are attached to node i. This means that $\bar{x}_{\mathcal{P}(i)}$ is node i's only dependency on its parent nodes and $\bar{x}_i$ is the interface variable that is shared with node i's children. We note that Constraint 4.21 is now an internal constraint. This decentralisation has decoupled the parents of node i. The decoupling is evident by noting that Equation 4.23 is a decoupled set of equality constraints: $\tilde{x}^i_l = \bar{x}_l,\ \forall l \in \mathcal{P}(i)$. The decentralisation step is shown schematically in Figure 4.2.

From the ADMM perspective, the sets of vector variables x_i and $\tilde{x}^i_{\mathcal{P}(i)}$ are mapped to z_1 in Problem 4.13 and the sets of variables $\bar{x}_i$ are mapped to z_2. The sets of constraints (4.21-4.23) are collectively mapped to the equality constraint in Problem 4.13. Define $\hat{x}_i = \left(x_i, \tilde{x}^i_{\mathcal{P}(i)}\right)$. Based on the mapping to ADMM, $\hat{x}_i$ is the primary vector variable while $\bar{x}_i$ is the secondary vector variable.

The main optimisation process consists of message passing and optimisation updates defined in a sequential and decentralised manner that is equivalent to the centralised version in Equation 4.15. The process begins with the head nodes and then proceeds to traverse the graph according to Algorithm 4.1. The algorithm refers to two types of updates and two types of messages, upward and downward. We will now proceed to define what takes place during each update and what each message contains.

At the outset, each node $i \in V$ is initialised with ${}^1x_i$, ${}^1\bar{x}_i$, ${}^1\tilde{x}^i_{\mathcal{P}(i)}$ ${}^0\lambda_i$, ${}^0\mu_i$ and ${}^0\eta^i_{\mathcal{P}(i)}$. In the t-th downward update, node i updates its Lagrangian multipliers to obtain ${}^t\lambda_i$, ${}^t\mu_i$ and ${}^t\eta^i_{\mathcal{P}(i)}$. It then updates the primary variables to obtain ${}^{t+1}x_i$ and ${}^{t+1}\tilde{x}^i_{\mathcal{P}(i)}$. The node's downward message contains ${}^t\bar{x}_i$ that is required by its children nodes to update their primary variables. In the t-th upward update node, i updates its

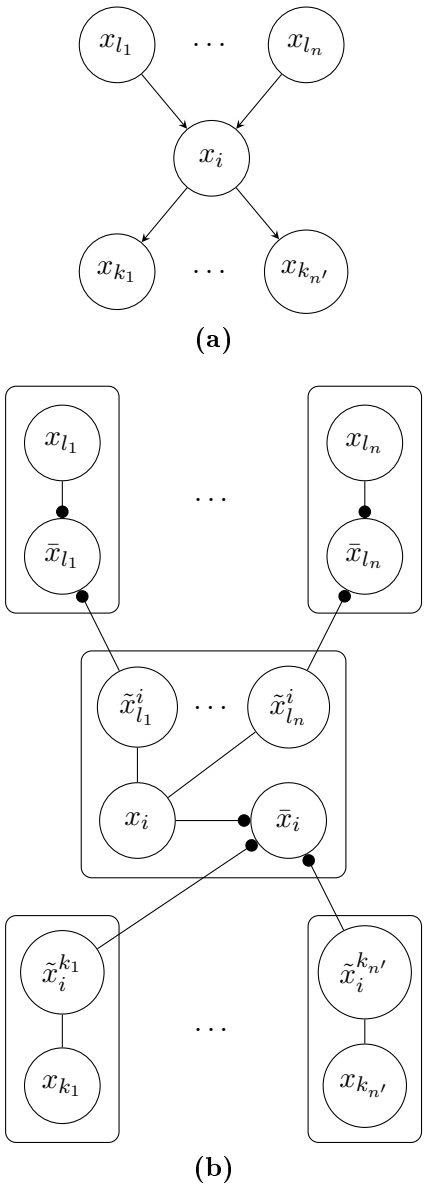


Figure 4.2 – Decentralisation step of DADMM. In Figure 4.2a, node i is shown with its parents and children in a DAG. It is assumed that node i has constraints that include all of its parents. By transforming the network into that of Figure 4.2b, each of node i's parents only needs to communicate with node i given that they are not coupled elsewhere.

secondary variables to obtain ${}^{t+1}\bar{x}_i$. It then sends its upward message containing variables ${}^{t+1}\tilde{x}^i_{\mathcal{P}(i)}$ and Lagrangian multipliers ${}^{t}\eta^i_{\mathcal{P}(i)}$.

The decentralised nature of the process is evident from the definition of the updates at the node level. We need to show that the process is in fact equivalent to performing the centralised version of ADMM on the entire system. A proof of this equivalence is provided in Theorem 4.2 following Lemma 4.2.

Lemma 4.2 (DADMM sequence following)**.** *In Algorithm 4.1, every downward update t of node i is followed by an upward update t. Moreover, every upward update t of node i is followed by a downward update $t+1$.*

Proof. From Lemma 4.1, we know that if node i has just performed the t-th update then it has performed $t-1$ upward updates. Now, suppose that the next update is the $t+1$-th downward update. This is impossible because according to Lemma 4.1, node i would have performed t upward updates. The second part of the statement can be proved through a similar argument. □

Theorem 4.2 (DADMM sequence correctness)**.** *For each node i, after the t-th update of the Lagrangian multipliers in the downward update, the variables owned by the node are equal to the ADMM update t of those variables.*

Proof. The proof is by induction. Before the start of the algorithm, the variables of node i are set to ${}^{1}x_i$, ${}^{1}\bar{x}_i$, ${}^{1}\tilde{x}^i_{\mathcal{P}(i)}$ ${}^{0}\lambda_i$, ${}^{0}\mu_i$ and ${}^{0}\eta^i_{\mathcal{P}(i)}$. During the first downward update, node i's Lagrangian variables are updated according to Equations 4.24. The node would have received ${}^{1}\bar{x}_{\mathcal{P}(i)}$ from its parents' downward messages. At this stage, all variables belong to the ADMM update at $t'=1$.

$$\begin{aligned} {}^{1}\lambda_i &:= {}^{0}\lambda_i + g({}^{1}x_i, {}^{1}\tilde{x}^i_{\mathcal{P}(i)}) \\ {}^{1}\mu_i &:= {}^{0}\mu_i + ({}^{1}x_i - {}^{1}\bar{x}_i) \\ {}^{1}\eta^i_{\mathcal{P}(i)} &:= {}^{0}\eta^i_{\mathcal{P}(i)} + ({}^{1}\tilde{x}^i_{\mathcal{P}(i)} - {}^{1}\bar{x}_{\mathcal{P}(i)}) \end{aligned} \tag{4.24}$$

Assume that after node i's $t-1$-th Lagrangian update, all variables belong to the ADMM update at $t'=t-1$. We now prove the statement for $t'=t$. After the $t-1$-th

Lagrangian variable update, node i directly updates its primary variables according to Equation 4.25.

$$\begin{aligned}&({}^{t+1}x_i, {}^{t+1}\tilde{x}^i_{\mathcal{P}(i)}) := \\ &\underset{x_i, \tilde{x}^i_{\mathcal{P}(i)}}{\arg\min}\, \mathcal{L}^1_i(x_i, \tilde{x}^i_{\mathcal{P}(i)}, {}^{t}\bar{x}_i, {}^{t}\bar{x}_{\mathcal{P}(i)}, {}^{t}\lambda_i, {}^{t}\mu_i, {}^{t}\eta^i_{\mathcal{P}(i)})\end{aligned} \tag{4.25}$$

According to Lemma 4.2, the downward update is followed by an upward update. In the upward update, node i would have received ${}^{t+1}\tilde{x}_i^{\mathcal{C}(i)}$ from its children as well as the corresponding Lagrangian variables ${}^{t}\eta_i^{\mathcal{C}(i)}$. After receiving these variables, node i updates the secondary variables according to Equation 4.26.

$${}^{t+1}\bar{x}_i := \underset{\bar{x}_i}{\arg\min}\, \mathcal{L}^2_i({}^{t+1}x_i, {}^{t+1}\tilde{x}_i^{\mathcal{C}(i)}, \bar{x}_i, {}^{t-1}\mu_i, {}^{t}\eta_i^{\mathcal{C}(i)}) \tag{4.26}$$

The minimisation over $\bar{x}_i$ can be written explicitly as shown in Equation 4.27.

$$\begin{aligned}&{}^{t+1}\bar{x}_i := \\ &\frac{1}{|\mathcal{C}(i)|+1}\left[{}^{t+1}x_i + {}^{t}\mu_i + \sum_{k\in\mathcal{C}(i)} \left[{}^{t+1}\tilde{x}_i^k + {}^{t}\eta_i^k\right]\right]\end{aligned} \tag{4.27}$$

Finally, from Lemma 4.2, we have given that the upward update is followed by downward update $t+1$ during which the Lagrangian variables are updated in an analogous manner to Equation 4.24 to obtain ${}^{t+1}\lambda_i$, ${}^{t+1}\mu_i$ and ${}^{t+1}\eta^i_{\mathcal{P}(i)}$. Hence, the t+1-th ADMM update is complete. □

4.2.5 Analysis

In this section, we analyse the computational complexity of one iteration of DADMM. Analysis of the full problem depends on its convergence rate, which we consider for the special case of threshold-DIF in Section 4.3.

First, we define the following terms:

$$\begin{aligned} n_g^i &: \text{ number of constraints for node } i \\ n_g^{\max} &:= \max_{i \in V} n_g^i \\ n_i &:= |x_i| = |\bar{x}_i| \\ n_i' &:= |\tilde{x}^i_{\mathcal{P}(i)}| \\ n_{\max} &:= \max_{i \in V} n_i \end{aligned} \quad . \tag{4.28}$$

The worst-case complexity of one iteration of DADMM is given by Theorem 4.3.

Theorem 4.3 (DADMM complexity)**.** *Each iteration of DADMM over a DAG* $G = \{V, E\}$ *runs in*

$$\mathcal{O}(\kappa(G)(n_{\max}|V|)^2(n_g^{\max} + n_{\max}|V|))$$

time.

Proof. Each ADMM iteration consists of three updates: the primary update, secondary update and the Lagrangian update. Since the algorithm is decentralised and synchronous, its running time is dominated by the time to update a single node multiplied by the depth of G.

We now develop an upper bound on the computation performed by an arbitrary node i. The primary update involves solving $\nabla_{\hat{x}_i} L = 0$. This equation is a linear equation, since the Lagrangian is quadratic and has the form $A\hat{x}_i = d$, where $A \in \mathbb{R}^{(n_i+n_i')^2}$ and $d \in \mathbb{R}^{n_i+n_i'}$ since the primary vector $\hat{x}_i$ has dimension equal to $n_i + n_i'$. Each element in A potentially contains a term from the objective and a term from each constraint in which the corresponding element in $\hat{x}_i$ is involved. Each element of $\hat{x}_i$ is involved in an equality constraint (with the secondary variables) and may appear in the constraints of g_i. Therefore, the elements of A can be computed in $\mathcal{O}((n_i + n_i')^2(2 + n_g^i)) = \mathcal{O}((n_i + n_i')^2 n_g^i)$ time. Each element in d can include a secondary variable and a Lagrangian multiplier from each constraint. The time complexity of computing the elements of d is $\mathcal{O}((n_i + n_i')(1 + n_g^i))$. Solving for $\hat{x}_i$, assuming a dense matrix A,

takes $\mathcal{O}((n_i + n'_i)^3)$ time. Therefore, the time complexity of the primary update is $\mathcal{O}((n_i + n'_i)^2(n_i + n'_i + n^i_g))$.

The secondary update computes an average over x_i and $\tilde{x}_i^{\mathcal{C}(i)}$ to obtain $\bar{x}_i$. Thus, its time complexity is $\mathcal{O}(n_i(1 + |\mathcal{C}(i)|)) = \mathcal{O}(n_i|\mathcal{C}(i)|)$.

The Lagrangian update involves an evaluation of all constraints given by Equations 4.21, 4.22 and 4.23. The evaluation of each element of g_i involves at most $n_i + n'_i$ variables. The evaluation of the other constraints involves two variables each. Therefore, the time complexity of the Lagrangian update is $\mathcal{O}(n^i_g(n_i+n'_i)+n_i+n'_i) = \mathcal{O}(n^i_g(n_i + n'_i))$.

The total time complexity is the sum of these three updates. Thus, we have $\mathcal{O}((n_i + n'_i)^2(n_i+n'_i+n^i_g))+\mathcal{O}(n_i|\mathcal{C}(i)|)+\mathcal{O}(n^i_g(n_i+n'_i)) = \mathcal{O}((n_i+n'_i)^2(n_i+n'_i+n^i_g)+n_i|\mathcal{C}(i)|)$.

We now restate the bound as a function of $n_{\max}$, n^i_g and $|V|$. Since $n_{\max}$ is an upper bound for n_i, $n_{\max}|\mathcal{P}(i)|$ is an upper bound on n'_i. An upper bound for the number of children or the number of parents is simply the number of nodes $|V|$. Hence, a more conservative upper bound for n'_i is $n_{\max}|V|$. Therefore, the overall time complexity for the work performed by node i can be rewritten as $\mathcal{O}((n_{\max}|V|)^2(n^{\max}_g + n_{\max}|V|))$. The total time complexity for one iteration of the algorithm is thus $\mathcal{O}(\kappa(G)(n_{\max}|V|)^2(n^{\max}_g + n_{\max}|V|))$. □

4.3 Threshold-DIF

In this section, we show how DADMM can be applied to the threshold-DIF problem. The mapping from threshold-DIF to the general problem formulation of DADMM is presented in detail. A detailed complexity analysis of DADMM in terms of the threshold-DIF problem size is provided.

4.3.1 Threshold-DIF Using DADMM

The threshold-DIF problem can be solved using DADMM. However, we first must reformulate Constraints 3.4 and 3.7 such that they are compatible with the DADMM framework.

The *maximum* function in Equation 3.4 is replaced by Inequality 4.32. The maximum function is non-smooth and cannot be optimised in one step. With the Inequalities 4.32, the objective and all equality and inequality constraints of the optimisation become linear as required by DADMM.

The set of inequalities in Equation 4.32 is equivalent to the maximum relation in Equation 3.4 as long as one of the constraints is active. One constraint will always be active for h_{ik}^m if the link cost c_{ik}^m is positive. From the problem formulation, we know that the link cost c_{ij}^m can only be negative if $j \in V_e$, i.e. if the link is incident to an estimator j. For these links, we replace the set of inequalities in Equation 4.32 with the set of equalities given in Equation 4.29.

$$\begin{aligned} x_{ik}^m(j) &= 0, \quad \text{if } k \neq j \\ x_{ik}^m &= h_{ik}^m, \quad \text{if } k = j \end{aligned} \tag{4.29}$$

The inter-link constraint given in Equation 3.7 is replaced by Constraint 4.35 where $\mathcal{S}_{is}$ is the set of links emanating from node i involved in the inter-link constraint s. The DADMM format only permits constraints between a node and its parents. Therefore, Constraint 4.35 only applies between links from node i and links from node i's parents. This condition is not restrictive since an extra link can be added between non-neighbouring nodes with inter-link constraints while retaining the directed acyclic property of the graph. The graph remains acyclic by preserving any ordering between the two nodes between which the extra link is added. Since the network is a connected DAG, then, by definition, for any two nodes i and k either k is a successor of node i ($k \in \bar{\mathcal{C}}(i)$) or k is an ancestor of i ($k \in \bar{\mathcal{P}}(i)$) or neither. If $k \in \bar{\mathcal{C}}(i)$, the link should extend from i to k. If $k \in \bar{\mathcal{P}}(i)$, the link should extend from k to i. If there is no directed path between the nodes, then either direction retains the acyclic property.

After transforming the constraints, we obtain the problem (4.30-4.35) shown below.

$$\text{minimise} \quad \sum_{(i,k)\in E} c_{ik}^m h_{ik}^m \tag{4.30}$$

$$\text{subject to} \quad x_{ik}^m(j) \geq 0 \tag{4.31}$$

$$x_{ik}^m(j) \leq h_{ik}^m \tag{4.32}$$

$$\sum_{l\in\mathcal{P}(i)} x_{li}^m(j) - \sum_{k\in\mathcal{C}(i)} x_{ik}^m(j) + r_i^m(j) = 0 \tag{4.33}$$

$$\sum_m \nu_{ik}^m h_{ik}^m \leq C_{ik} \tag{4.34}$$

$$\sum_{k\in\mathcal{S}_{is}} \sum_m \nu_{ik}^m h_{ik}^m + \sum_{l\in\mathcal{P}(i)} \sum_{k'\in\mathcal{S}_{ls}} \sum_m \nu_{lk'}^m h_{lk'}^m \leq K_s \tag{4.35}$$

To solve threshold-DIF, all that is required at this stage is that the threshold-DIF variables and constraints be mapped to the variables of the distributed optimisation problem (4.16-4.17). This can be done as follows. The sets of variables $\{h_{ik}^m : m \in V_s, k \in \mathcal{C}(i)\}$ and $\{x_{ik}^m(j) : m \in V_s, j \in V_e, k \in \mathcal{C}(i)\}$ are mapped to x_i. The objective function f_i in Equation 4.16 is represented by $\sum_{k\in\mathcal{C}(i)} \sum_m c_{ik}^m h_{ik}^m$ and the indicator functions resulting from the inequality constraints. The constraint g_i in Equation 4.17 is represented by the equality constraints and the equality versions of the inequality constraints involving node i in the problem (4.30-4.35).

DADMM runs continuously throughout system operation. As the system configuration changes, link costs and weights are updated with new values. To ensure convergence, the interval between updates is set to an adequate time period. In practice, this interval was found to be of similar length to that determined for min-cost-DIF in Section 4.1.3.

In our implementation, to account for link quality degradation, a value proportional to the inter-robot distance is added to the weights $\{\nu_{ik}^m\}$. This signals a need for

re-transmission due to a decline in link quality. The constant of proportionality is a fixed value set by the operator. More complicated communication models can be employed but this investigation is outside the scope of this thesis.

4.3.2 Analysis

In this section, we provide time complexity analysis of DADMM when applied to threshold-DIF. The complexity is expressed in terms of the size of the threshold-DIF input parameters.

Complexity analysis is provided for the entire optimisation process including the number of iterations required for convergence. We first determine the complexity of one iteration following directly from Theorem 4.3. We then find a bound on the number of iterations based on the algorithm's convergence rate.

The complexity of a DADMM iteration was determined in Section 4.2. The complexity of one iteration in terms of threshold-DIF problem specification can be determined by substituting the appropriate values for $n_{\max}$ and $n_g^{\max}$. To begin, we denote the number of sources, destinations and inter-link constraints in the network as follows:

- N_m: number of sources in the network.
- N_j: number of destinations in the network.
- N_s: number of inter-link constraints.

The maximum number of primary variables $n_{\max}$ is proportional to the maximum number of routing variables which, in turn, is proportional to the number of sources multiplied by the number of destinations multiplied by the number of children. The number of children is bounded from above by the number of nodes. Therefore, $n_{\max}$ is bounded such that $n_{\max} \leq N_m N_j |V|$.

The maximum number of constraints $n_g^{\max}$ is bounded by the maximum number of flow consistency constraints in Equation 4.33 and the maximum number of inter-link

constraints in Inequality 4.35. The number of consistency constraints is proportional to the number of sources multiplied by the number of destinations. Therefore, $n_g^{\max}$ is bounded such that $n_g^{\max} \leq N_m N_j + N_s$.

Substituting the obtained bounds into the result of Theorem 4.3, the complexity of one iteration of DADMM for threshold-DIF becomes $\mathcal{O}(\kappa(G)(|V|^2 N_m N_j)^2(|V|^2 N_m N_j + Ns))$. In threshold-DIF, the depth of the underlying graph is a function of processor cascading which is independent of the number of robots. Therefore, the depth is assumed to be constant. Hence, the time complexity of one iteration can be restated as $\mathcal{O}((|V|^2 N_m N_j)^2(|V|^2 N_m N_j + Ns))$

To determine the complexity of the whole optimisation process, the convergence rate is required. A convergence rate in an ergodic sense is established in [44] with relatively mild assumptions.

The result is restated here after establishing the appropriate notation. Define the primal vector of the k-th iteration as $z^k = (z_1^k, z_2^k)$ where z_1^k and z_2^k are the ADMM primary and secondary vectors defined in Section 4.2.1. Define the ergodic average $\tilde{z}^k = \sum_{k'=1}^{k+1} z^{k'}$ and define z^* and y^* as the optimal primal and dual vectors. Then, if we assume that $z^0 = 0$ and $y^0 = 0$, the convergence result is given by Inequality 4.36. The positive constants α and β are independent of the dimension and value of both the primal and dual variables.

$$L(\tilde{z}^k, y^*) - L(z^*, y^*) \leq \frac{\alpha\|z^*\|^2 + \beta\|y^*\|^2}{(k+1)} \tag{4.36}$$

From Equation 4.36, it is clear that in order to obtain a bound on the convergence rate, we need to find an upper bound on the norm of the primal and dual optimal vectors. The absolute value of the elements in the primal vector have an upper bound u_z which follows from the problem definition in Section 4.3.1. Hence, an upper bound on the squared norm of the primal vector is given by Equation 4.37.

$$\|z^*\|^2 \leq n_z u_z^2 \tag{4.37}$$

We now seek a bound for the squared norm of the dual vector $\|y^*\|^2$. To simplify the analysis, we note that the centralised *ADMM* version of the threshold-DIF problem has the form of the optimisation problem defined in Equation 4.38 where the indicator function $I_{\geq 0}$ is defined in Equation 4.39 below. The set $\mathcal{I}$ contains the indices of the variables added to convert any inequality constraint into an equality constraint as described in Section 4.2. These variables need to satisfy the inequality constraint $z_{(i)} \geq 0$ and they only appear in one row of the set of equality constraints $Az = b$.

$$\begin{aligned} &\text{minimise} \quad c^T z + \frac{\rho}{2}\|Az - b\|^2 + \sum_{i \in \mathcal{I}} I_{\geq 0}(z_{(i)}) \\ &\text{subject to} \quad Az = b \\ &\qquad\qquad A \in \mathbb{R}^{n_g \times n_z},\ b \in \mathbb{R}^{n_g} \end{aligned} \tag{4.38}$$

$$I_{\geq 0}(z_{(i)}) = \begin{cases} 0 & \text{if } z_{(i)} \geq 0 \\ \infty & \text{otherwise} \end{cases} \tag{4.39}$$

An upper bound on $\|y^*\|^2$ can be obtained from the following lemma.

Lemma 4.3 (ADMM Lagrangian multiplier boundedness)**.** *Assume that the equality constraints and the inequality constraints $z_i \geq 0$ active at x^* are all linearly independent. Then, there exists a positive constant γ independent of z, n_z and n_g such that $\|y^*\|^2 \leq \gamma \|c\|^2$.*

Proof. At optimality, we have $Az - b = 0$ and zero belongs to the subdifferential of the Lagrangian as shown in Equation 4.40. The i-th element of the vector $b_I \in \mathbb{R}^{n_z}$ is defined in Equation 4.41 where $\partial I_{\geq 0}$ is the subgradient of the non-smooth indicator function.

$$0 \in c + A^T y^* + b_I \tag{4.40}$$

$$b_{I(i)} = \begin{cases} \partial I_{\geq 0} & \text{if } i \in \mathcal{I} \text{ and the constraint } z_i \geq 0 \text{ is active} \\ 0 & \text{otherwise} \end{cases} \tag{4.41}$$

The subgradient $\partial I_{\geq 0}$ evaluated at 0 is an unbounded set and therefore cannot be used to directly bound y^*. However, the optimality condition in Equation 4.40 has n_z rows while y^* has dimension n_g. Furthermore, the maximum number of active inequality constraints, i.e. constraints where $b_{I(i)} \neq 0$, is equal to $(n_z - n_g)$ since otherwise, x^* would be over-defined by the constraints due to the linear independence assumption. Consequently, if all rows in Equation 4.40 such that $b_{I(i)} \neq 0$ are removed, there will remain at least n_g rows.

To this end, we need to make sure that the matrix $\bar{A}^T$ obtained after removing the rows from A^T remains full rank. When active, the inequality constraint $z_{(i)} \geq 0$ becomes $z_{(i)} = 0$. If this equality is augmented as a row vector to the matrix A, due to the linear independence assumption, the rank of A becomes $n_z + 1$. Through elementary row operations, any non-zero element on the column corresponding to $z_{(i)}$ can be changed to zero with no change in the rank of the matrix. At this stage, the row $z_{(i)} = 0$ can then be removed with the rank of the matrix dropping back to n_g. Once the row is removed, the column corresponding $z_{(i)}$ column is now all zeros and can hence be removed with no change in rank. This proves that $\bar{A}^T$ has full rank n_g.

Therefore, the optimality condition in Equation 4.40 can be restated as Equation 4.42 where $\bar{c}$ is the vector obtained after removing all the corresponding rows from c. The vector b_I becomes a zero vector after removing these rows.

$$\bar{A}^T y^* = -\bar{c} \tag{4.42}$$

From Equation 4.42 we obtain Equation 4.43 where $\sigma_{\min}(\bar{A}\bar{A}^T)$ is the minimum eigenvalue of $\bar{A}\bar{A}^T$ and is greater than zero since $\bar{A}$ is full rank.

$$\frac{\|\bar{c}\|^2}{\|y^*\|^2} = \frac{y^{*T}\bar{A}\bar{A}^T y^*}{\|y^*\|^2} \geq \sigma_{\min}(\bar{A}\bar{A}^T) \tag{4.43}$$

The proof is established by setting $\gamma = 1/\sigma_{\min}(\bar{A}\bar{A}^T)$ and noting that $\|\bar{c}\|^2 \leq \|c\|^2$ since $\bar{c}$ is obtained by removing elements from c. □

We assume that all elements in c are upper-bounded by a constant value u_c. This is a reasonable assumption since c represents the link cost vector and only the relative cost is of importance. Consequently, from Lemma 4.3, we have the upper bound given in Equation 4.44 for the squared norm of the dual vector.

$$\|y^*\|^2 \leq \gamma u_c^2 n_g \tag{4.44}$$

We can now state the main complexity result given by Theorem 4.4. The complexity is polynomial as expected since the problem is convex and the number of variables is polynomial in the number of nodes.

Theorem 4.4 (Threshold-DIF complexity)**.** *Obtaining an ϵ-optimal solution for the threshold-DIF problem using DADMM has a computational complexity of*

$$\mathcal{O}((|V|^2 N_m N_j)^2 (|V|^2 N_m N_j + Ns)(|V|^3 N_m N_j + N_s)/\epsilon) \tag{4.45}$$

Proof. Bounds in Equation 4.44 and Equation 4.37 mean that the left hand side of Equation 4.36 is bounded by a constant weighted sum of n_z and n_g. Therefore, an upper bound on the number of iterations k required to produce an error ϵ is given as $\mathcal{O}((n_z + n_g)/\epsilon)$.

From the proof of Theorem 4.3, we note that the number of primary variables n_z is bounded by $\mathcal{O}(|V|^2 N_m N_j)$ multiplied by the number of nodes. Therefore, we have $n_z \leq \mathcal{O}(|V|^3 N_m N_j)$. The number of constraints n_g, on the other hand, can be bounded such that $n_g \leq \mathcal{O}(|V|^3 N_m N_j + N_s)$. Thus, the resulting number of iterations of the optimisation process is given by Equation 4.46.

$$k \leq \mathcal{O}(|V|^3 N_m N_j + N_s)/\epsilon \tag{4.46}$$

The complexity of the whole optimisation process is obtained by multiplying the number of iterations by the complexity of each iteration. Thus, for an ϵ-optimal solution, DADMM for threshold-DIF runs in $\mathcal{O}((|V|^2 N_m N_j)^2 (|V|^2 N_m N_j + Ns)(|V|^3 N_m N_j + N_s)/\epsilon)$ time. □

4.4 Myopic Sensor Utility Approximation

In this section, we provide a brief empirical analysis of the validity of the myopic approximation of sensor utility used in our DIF implementations. In our DIF implementations, an estimator approximates sensor utility by evaluating the most recent sensor observation received. The value of the observation is computed as the reduction in entropy realised by fusing the observation into the estimator. This approach is advantageous due to the simplicity of implementation. Entropy reduction can be computed efficiently without additional data storage. The disadvantage of this approximation is that it is myopic. Myopic approximations only reflect the instantaneous effect of a sensor observation on the information gathering performance of a single robot. Myopic approximations are commonly used since the long-term value of a sensor observation can be difficult to compute in the general case [100].

Inspired by the recent success in exploiting the submodularity property of mutual information for the sensor selection problem [38], one may assume that such property would prove beneficial for providing a sensor utility estimate. However, submodularity does not readily extend to information gathering tasks with dynamic environments which is the case of interest. For further details, Appendix A provides an analysis of the submodularity of linear-Gaussian systems and gives a simple counterexample.

Further insight into the error introduced by a myopic approximation for linear-Gaussian systems can be obtained by the comparison with the upper bound provided in [4]. Define P_k as the covariance matrix of the estimate at time k and define ϕ_k as the propagation function of the covariance from time 0 to time k such that $P_k = \phi_k(P_0)$. Then, due to the concavity of the discrete Riccati equation [92] we have the upper bound given by Equation 4.47 for the error introduced by an ϵ deviation of P_0 in the direction of $Q \in \mathcal{S}_+$. If we assume that $\alpha I \preceq P_k \preceq \beta I$, then we have the bound given by Equation 4.48 and $\lambda_{\min}(P_k) \geq \alpha$ and the final bound on the error is given

by Equation 4.49.

$$\begin{aligned}\log|\phi_k(P_0+\epsilon Q)| - \log|\phi_k(P_0)| &\leq \frac{d}{d\epsilon}\log|\phi_k(P_0+\epsilon Q)| \\ &= \operatorname{tr}\left((\phi_k(P_0))^{-1}\frac{d}{d\epsilon}\phi_k(P+\epsilon Q)\right) \\ &\leq \frac{1}{\lambda_{\min}(\phi_k(P_0))}\operatorname{tr}\left(\frac{d}{d\epsilon}\phi_k(P_0+\epsilon Q)\right)\end{aligned} \tag{4.47}$$

$$\operatorname{tr}\left(\frac{d}{d\epsilon}\phi_k(P_0+\epsilon Q)\right) \leq \beta\left(\frac{\beta}{\beta+\alpha}\right)^k \operatorname{tr}(P_0^{-1}Q) \tag{4.48}$$

$$\log|\phi_k(P_0+\epsilon Q)| - \log|\phi_k(P_0)| \leq \frac{\beta}{\alpha}\left(\frac{\beta}{\beta+\alpha}\right)^k \operatorname{tr}(P_0^{-1}Q) \tag{4.49}$$

To evaluate the myopic approximation for fixed receding-horizon planning relative to the upper bound, a multi-robot mapping simulation was performed. The simulation scenario included two mobile robots mapping a spatio-temporal varying field. To permit the use of the performance bound, open-loop control was assumed. At the end of each time horizon, the first robot computed the expected information content of its estimate at the end of the second horizon with and without fusing the observation from the second robot. The resulting information content during the simulation is shown alongside the utility bound in Figure 4.3. As shown in the figure, the myopic approximation is much closer to the multiple time-step utility when compared with the conservative upper-bound. Although the upper bound is easy to compute, for many practical applications, the myopic approximation provides a better estimate.

4.5 Summary

In this chapter, we proposed an efficient decentralised solution for both the min-cost-DIF and threshold-DIF problems defined in Sections 3.2 and 3.3. Our solution to min-cost-DIF was adapted from recent results in multicast routing, which we extended to allow for negative link costs that represent sensor utility. In threshold-DIF, flow

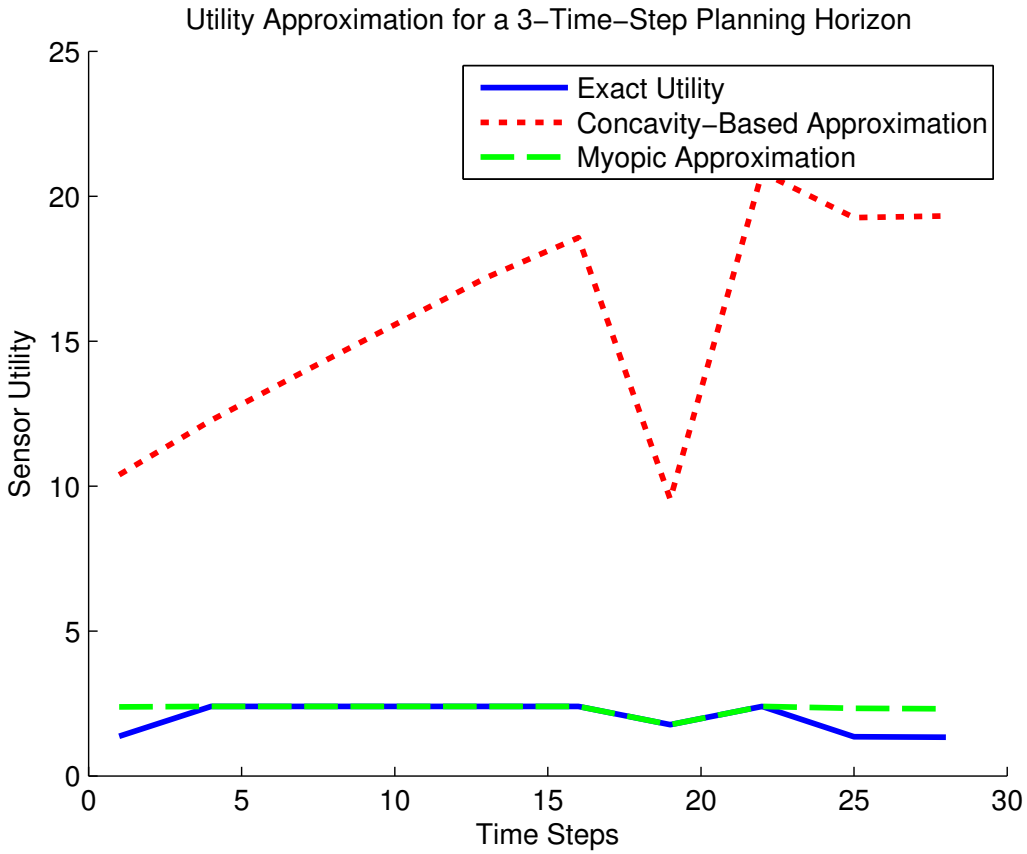


Figure 4.3 – Comparison of fixed-horizon sensor utility approximations. The solid line shows the exact sensor utility computed over a fixed time horizon. The dashed line shows the myopic approximation while the dotted line shows the concavity-based approximation.

rates are optimised based on the value of information while obeying local computation limits and global communication limits. Our solution to threshold-DIF is based on a distributed version of ADMM that requires neighbour-to-neighbour communication only. Finally, we proved that the convergence time of our solution is polynomial in the size of the network. In the following chapter, we present a solution to the third problem considered in this thesis, negotiation-DIF.

Chapter 5

Negotiation-DIF

In this chapter, we present a solution to the negotiation-DIF problem. Negotiation-DIF addresses communication efficiency at both the data fusion and decision making layers concurrently.

First, we begin with a brief introduction of background material in Section 5.1. This introduction will aid our presentation of linear-quadratic information structure optimisation (LQISO), a solution algorithm for the comms-LQ problem. Presented in Section 5.2, LQISO provides a communication-efficient decision making solution for LQ problems. Then, in Section 5.3, we extend LQISO to provide a communication-efficiency solution for decision making in decentralised information gathering. Finally, in Section 5.4, we combine the solution of min-cost-DIF with the extended version of LQISO to present a solution to negotiation-DIF.

5.1 LMIs in LQ Optimal Control

In this section, we introduce background material on the use of linear matrix inequalities (LMIs) in LQ optimal control. This background information is necessary for the presentation of the algorithm described in Section 5.2. A detailed discussion on LMIs and optimal control is outside the scope of this thesis and can be found in

[12, 81]. LMI formulations of the LQ optimal control problem allow the addition of extra criteria such as communication constraints.

To this end, consider a system with linear dynamics given by Equation 5.1 and quadratic cost function given by Equation 5.2. Assume that the control vector is set to the feedback law given by Equation 5.3. When the matrix K is the solution of the algebraic Riccati equation, then Equation 5.3 is the optimal feedback control policy.

$$\dot{x} = f(x,u) = Ax + Bu \tag{5.1}$$

$$g(x,u) = \frac{1}{2}\left(x^T Q x + u^T R u\right) \tag{5.2}$$

$$u = -R^{-1}B^T K x \tag{5.3}$$

If we consider the quadratic function given by Equation 5.4 where K positive definite, then the dissipation inequality [98] is given by Equation 5.5 and in differential form in Equation 5.6.

$$V(x) = \frac{1}{2}x^T K x \tag{5.4}$$

$$\int_{t_0}^{t_1} g(x,u)dt + V(x_1) \geq V(x_0) \tag{5.5}$$

$$\frac{\partial V}{\partial x}^T f(x,u) \geq -g(x,u) \tag{5.6}$$

If V satisfies the dissipation inequality for the choice of control action, then it is a lower bound on the value function of the system. If we substitute the definitions of V, f and g into the dissipation inequality we obtain Inequality 5.7. Consequently, we obtain the inequalities (5.8-5.11). Inequality 5.9 is obtained since a scalar is equal to its transpose. Inequality 5.10 results from the arbitrary choice of x and Inequality 5.11 is a result of the Schur complement lemma.

$$x^T KAx - x^T KBR^{-1}B^T Kx + \frac{1}{2}x^T Qx + \frac{1}{2}x^T KBR^{-1}B^T Kx \qquad \geq 0 \tag{5.7}$$

$$x^T KAx - \frac{1}{2}x^T KBR^{-1}B^T Kx + \frac{1}{2}x^T Qx \geq 0 \quad (5.8)$$

$$\frac{1}{2}x^T KAx + \frac{1}{2}x^T A^T Kx - \frac{1}{2}x^T KBR^{-1}B^T Kx + \frac{1}{2}x^T Qx \geq 0 \quad (5.9)$$

$$KA + A^T K - KBR^{-1}B^T K + Q \succeq 0 \quad (5.10)$$

$$\begin{bmatrix} KA + A^T K + Q & KB \\ B^T K & R \end{bmatrix} \succeq 0 \quad (5.11)$$

The set defined by Inequality 5.11 contains a maximal element which corresponds to the solution of the algebraic Riccati equation when it exists [81]. In Section 5.2, we search this set for other solutions that take into account communication costs.

5.2 LQISO

This section introduces LQISO as a novel solution approach to comms-LQ. Although the solution is an approximation, it has proven to be useful for the purpose of communication efficiency. We present the algorithm, analyse its complexity and provide some examples.

5.2.1 Algorithm

Due to the difficulty of solving comms-LQ as defined in Probem 3.13, LQISO solves a surrogate problem instead. The LQISO algorithm relies on the LMI formulation of the LQ optimal control problem.

The optimisation problem defined in Problem 5.12 is equivalent to solving the algebraic Riccati equation, obtaining the steady-state optimal feedback control of the LQ problem. The optimisation problem is convex and can be solved through semi-definite programming (SDP) methods. Communication costs are added to the objective of the LMI formulation as a convex quadratic penalty function. As defined in Section 3.4.1, U is the communication cost matrix. The matrix U has the same dimensions as the team feedback gain matrix K. Each element in U is a positive communication cost of

the corresponding element in K. This cost corresponds to the communication cost of the link represented by the element in K. The resulting optimisation problem with communication costs added is given by Problem 5.13. The matrix $\bar{U}$ contains the lower-triangular elements of U placed along its diagonal. The vector $\text{vec}(K)$ consists of the lower-triangular elements of K put in vector form.

$$\begin{aligned} \text{min.} \quad & -\text{tr}(K) \\ \text{s.t.} \quad & \begin{bmatrix} KA + A^T K + Q & KB & \mathbf{0} \\ B^T K & R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & K \end{bmatrix} \succeq 0 \end{aligned} \tag{5.12}$$

$$\begin{aligned} \text{min.} \quad & -\text{tr}(K) + \frac{1}{2}\text{vec}(K)^T \bar{U} \text{vec}(K) \\ \text{s.t.} \quad & \begin{bmatrix} KA + A^T K + Q & KB & \mathbf{0} \\ B^T K & R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & K \end{bmatrix} \succeq 0 \end{aligned} \tag{5.13}$$

The modified formulation with communication costs is another LMI convex optimisation problem that is readily solvable using interior-point methods. We note that the original LMI formulation is a convex optimisation problem. The added communication cost is also convex since $\bar{U}$, being diagonal with positive terms, is always positive semi-definite. Thus, the modified formulation with communication costs is convex.

The outcome of the optimisation problem defined by Problem 5.13 is a modified gain matrix K which takes into account communication costs. According to the assumptions of the comms-LQ problem, the matrix K can be used to determine the communication rates subject to different interpretations. We present an interpretation below through an intuitive motivation. A slightly different interpretation is adopted in Section 5.3.

Based on robot i's feedback policy defined by Equation 3.11, and repeated in Equation 5.14 for convenience, the structure of K will determine how an element of the control vector belonging to a certain robot will depend on an element of the state

vector belonging to another robot. Hence, the elements of K_i give an indication of the coupling between robots. When multiplied by $-R_i^{-1}B_i^T$ they indicate the degree that the control action of robot i depends on the state of robot i and other robots. Therefore, we interpret the absolute value of the elements of K as proportionality weights for the communication rates required between robots. For example, if the magnitude of an element of the matrix pertaining to one robot is twice that of the element pertaining to another robot then the first robot is set to communicate at twice the rate of the second. Approximating weights as communication rates is accurate for short time steps when the state vector path is continuous.

$$u_i = -R_i^{-1}B_i^T K_i x \tag{5.14}$$

Standard SDP Form

The optimisation problem can be transformed to the standard SDP form with a linear objective form through Schur's lemma [12] and a simple change of variables. The quadratic term in the objective is replaced with the single variable f and Inequality 5.15. Using Schur's lemma, Inequality 5.15 is transformed to the semi-definite form in Inequality 5.16. Adding this inequality to the original inequality we obtain an optimisation problem in standard SDP form shown in Problem 5.17.

$$f - \text{vec}(K)^T \bar{U} \text{vec}(K) \geq 0 \tag{5.15}$$

$$\begin{bmatrix} \bar{U} & \text{vec}(K) \\ \text{vec}(K)^T & f \end{bmatrix} \succeq 0 \tag{5.16}$$

$$\begin{aligned} \text{min.} \quad & -\text{tr}(K) + f \\ \text{s.t.} \quad & \begin{bmatrix} KA + A^TK + Q & KB & 0 & & \\ B^TK & R & & & \\ & & K & & \\ 0 & & & \bar{U} & \text{vec}(K) \\ & & & \text{vec}(K)^T & f \end{bmatrix} \succeq 0 \end{aligned} \tag{5.17}$$

5.2.2 Analysis

We analyse the complexity of the optimisation in SDP form based on self-concordance of the objective and the barrier function. The objective function f is self-concordant since the log-determinant function and the quadratic term $\text{vec}(K)^T U \text{vec}(K)$ are self-concordant. The constraint is a semi-definite inequality which is also self-concordant.

$$f = t(-\text{tr}(K) + \text{vec}(K)^T U \text{vec}(K)) - \log\det(F) \tag{5.18}$$

Based on self-concordance, the number of Newton iterations required until convergence is given by Equation 5.19 where $\bar{\theta}$ is the degree of the positive-definite cone [11]. In our case $\bar{\theta} = n_k$ where $n_k \times n_k$ is the dimension of K.

$$\mathcal{O}\left(\sqrt{\bar{\theta}}\log(\bar{\theta})\right) \tag{5.19}$$

Each Newton update requires the computation of the Jacobian and the Hessian of f. The Jacobian of f with respect to the ij^{th} element of K is given by Equation 5.20. The Hessian of f is given by Equation 5.21. The complexity of computing the Jacobian and Hessian is $\mathcal{O}(n_k^6)$.

$$\frac{\partial f}{\partial K_{ij}} = t\left(U_{ij}K_{ij} - \delta_{ij}\right) - \text{tr}\left(F^{-1}\frac{\partial F}{\partial K_{ij}}\right) \tag{5.20}$$

$$\frac{\partial^2 f}{\partial K_{ij} \partial K_{kl}} = t\delta_{ik}\delta_{jl}U_{ij} + \mathrm{tr}\left(F^{-1}\frac{\partial F}{\partial K_{kl}}F^{-1}\frac{\partial F}{\partial K_{ij}}\right) \tag{5.21}$$

5.2.3 Examples

The LQISO algorithm was tested in simulation for LQ teams with different combinations of communication costs and utility couplings. The LQ team consists of two vehicles moving in a 2D plane with linear dynamics and such that the acceleration in each dimension is separately and directly controllable. The individual state vector and dynamics matrices are shown in Equation 5.22 for $i \in \{1, 2\}$. The hat $(\hat{\cdot})$ notation is included to avoid confusion with the state vector symbol x_i.

$$x_i = \begin{bmatrix} \hat{x}_i \\ \dot{\hat{x}}_i \\ \hat{y}_i \\ \dot{\hat{y}}_i \end{bmatrix} \quad A_i = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad B_i = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.22}$$

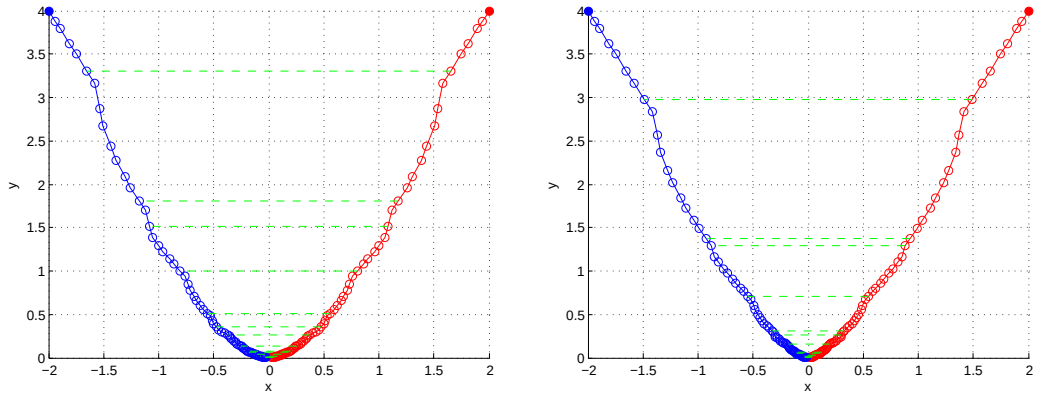
Table 5.1 displays the results for the different combinations for one dimension only, $\hat{x}$ for instance. The cost matrices Q shown in the table can be chosen to represent robots seeking to reach the origin but from opposite directions. The non-zero off-diagonal terms penalise the product $\hat{x}_1\hat{x}_2$ while the diagonal terms force the two robots to the origin. The communication cost matrices U penalise inter-robot terms to reduce communication.

From case 1 to case 2 and from 3 to 4, the communication cost increases from 1 to 10. This results in a decrease in the inter-controller gains of K to cater for the increase in communication costs. Also evident is the decrease in the controller's self gains; this allows each robot to take smaller steps due to the reduction in communication. Comparing case 1 with case 3, it is noticed that the reduction in cost coupling reduces the inter-controller gains, signalling a reduced need for communication. These results clearly relate with intuition.

Table 5.1 – Results of running LQISO on different combinations of team cost functions and communication costs for one dimension of the problem in Section 5.2.3

	Q	U	K
Case 1	$\begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 2.90 & 1.31 & 0.90 & 0.31 \\ 1.31 & 2.14 & 0.31 & 0.14 \\ 0.90 & 0.31 & 2.90 & 1.31 \\ 0.31 & 0.14 & 1.31 & 2.14 \end{bmatrix}$
Case 2	$\begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 10 & 10 \\ 0 & 0 & 10 & 10 \\ 10 & 10 & 0 & 0 \\ 10 & 10 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 2.09 & 1.02 & 0.09 & 0.02 \\ 1.02 & 2.01 & 0.02 & 0.01 \\ 0.09 & 0.02 & 2.09 & 1.02 \\ 0.02 & 0.01 & 1.02 & 2.01 \end{bmatrix}$
Case 3	$\begin{bmatrix} 2 & 0 & 0.5 & 0 \\ 0 & 2 & 0 & 0 \\ 0.5 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 3.08 & 1.40 & 0.50 & 0.17 \\ 1.40 & 2.19 & 0.17 & 0.08 \\ 0.50 & 0.17 & 3.08 & 1.40 \\ 0.17 & 0.08 & 1.40 & 2.19 \end{bmatrix}$
Case 4	$\begin{bmatrix} 2 & 0 & 0.5 & 0 \\ 0 & 2 & 0 & 0 \\ 0.5 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 10 & 10 \\ 0 & 0 & 10 & 10 \\ 10 & 10 & 0 & 0 \\ 10 & 10 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 2.68 & 1.24 & 0.09 & 0.02 \\ 1.24 & 2.12 & 0.02 & 0.01 \\ 0.09 & 0.02 & 2.68 & 1.24 \\ 0.02 & 0.01 & 1.24 & 2.12 \end{bmatrix}$

Figure 5.1a shows the paths chosen by robots with free communication and the paths chosen by robots with communication costs as in case 1 of Table 5.1 in Figure 5.1b. The cost function is also that of case 1. The robots start from the upper corners at the bullseye symbols and approach the origin from opposite directions. Instead of communicating their state vector at every time step, the robots send the required state component, at a rate proportional to the element of the gain matrix. The abrupt changes in path directions correspond to a time step when communication occurred and from the paths, it is seen that robots in the case of Figure 5.1b waited longer to communicate. Robots in the case of Figure 5.1b transfer 20% less data than those



(a) Assuming free communication **(b)** With communication costs

Figure 5.1 – Paths taken by two vehicles with a coupled quadratic cost. Robots start from the upper corners of the figure and head towards the origin approaching from opposite directions. Dashed lines indicate communication. The communication cost chosen are as per case 1 of Table 5.1

in the case of Figure 5.1a, yet they still trace similar paths and approach the origin from opposite directions. The results presented demonstrate that the LQISO can in fact be used to reduce communication while upholding performance.

5.3 Extended-LQISO

This section describes extended-LQISO which extends LQISO to non-LQ problems. The aim of extended-LQISO is to introduce communication efficiency into decision making for decentralised information gathering. Extended-LQISO acts as an auxiliary layer added to existing decentralised decision making algorithms. It does not serve as a decision making method on its own.

5.3.1 Algorithm

Suppose a decentralised information gathering team employs a particular DDM algorithm. A typically valid assumption is that the DDM algorithm involves commu-

nication at a certain rate. LQISO can be adapted to regulate this communication demand by adjusting communication rates between robots.

The steps of the algorithm are described in Algorithm 5.1. Before robots choose their next team decision using the DDM algorithm, they produce a local LQ approximation. The state dynamics are linearised and a quadratic approximation of the cost is also produced. Once the LQ approximation is obtained, the robots run LQISO which outputs the feedback gain matrix K.

In Section 5.2, communication rates were set directly proportional to the value of the off-diagonal elements. However, in the case of extended-LQISO, communication decisions are reached in a different manner, mainly for simplicity. Robots sum the absolute value of the terms (L^1 norm) of the matrix which are exclusive to themselves. These elements lie in square matrices along the diagonal. Then, the robots obtain the L^1 norm of the terms which couple them with other robots. For each pair of robots, if the ratio of the coupling value to the sum of the robots' local values is below a certain threshold, the robots do not cooperate through DDM. Currently, the threshold is a parameter chosen by the algorithm designer.

An essential requirement of the algorithm is the ability to produce a Hessian of the cost as a function of the state vector at the predicted team state. Since calculating the Hessian is cumbersome in most applications, the Hessian can be recursively estimated through Equation 5.23 which is a modified version of the Hessian approximation formula used in the Broyden-Fletcher-Golfarb-Shanno (BFGS) quasi-Newton optimisation method. To ensure the update is positive semi-definite, the absolute value of $\delta J^T \delta x$ is used.

$$Q_{k+1} = Q_k + \frac{\delta J \delta J^T}{|\delta J^T \delta x|} - \frac{Q_k \delta x (Q_k \delta x)^T}{\delta x^T Q_k \delta x} \tag{5.23}$$

For information gathering tasks, the team state vector includes the robots' state vector, the targets' position estimate and covariance. In addition to the robots' dynamics, both the target estimate and its covariance also have their own dynamics equations. Based on Shannon information theory, a suitable objective for an infor-

Algorithm 5.1: Extended-LQISO

```
1: loop
2:    Perform DDF
3:    Linearise dynamics and obtain quadratic cost approximation
4:    Exchange LQ dynamics and cost
5:    Run LQISO to determine if cooperation is required
6:    if Cooperation is required then
7:       Run DDM
8:    else
9:       Run local decision making
10:   end if
11: end loop
```

mation gathering task is to maximise information or minimise entropy. Alternatively, the reward function, which is the derivative of the value function along an optimal path, is mutual information and the cost function is the negative of the reward function. In the next section, the results of a simulated sample problem are demonstrated showing the effect this algorithm has on reducing communication while maintaining good performance for information gathering.

5.3.2 Sample Problem

Extended-LQISO was applied to the following decentralised information gathering example. Two robots with dynamics as in Equation 5.22 are equipped with range-only sensors. The robots' task is to minimise the uncertainty in the estimate of two moving targets. An EKF is used for estimation. The sensor model function for the range only sensor is given by Equation 5.24. Abusing notation, x_i is the position of robot i, x_t is the targets' state, and z_i is robot i's observation of the targets.

$$z_i = h(x_i, x_t) = \|x_i - x_t\|_2 + v_i,\ v_i \sim \mathcal{N}(0, V) \tag{5.24}$$

The EKF approximation results in the first order Taylor approximation of the sensor model about the targets estimate given by Equation 5.25.

$$\Delta z_i = H_i(x_i, x_t)\Delta x_t = \frac{(x_t - x_i)^T}{\|x_i - x_t\|_2}\Delta x_t \tag{5.25}$$

A constant velocity model is used for the target dynamics. The target dynamics given by Equation 5.26 include additive Gaussian noise represented by w with a fixed covariance matrix W. The covariance of the targets estimate is denoted by P. Since the cost is also a function of the covariance, then the covariance dynamics need to be considered. The covariance dynamics of the EKF approximation is the differential Riccati equation which is nonlinear. The corresponding dynamics of the covariance matrix can be derived as in Equation 5.27. By treating P as a vector and the equation on the right hand side as a multivariate function, the equation can be linearised.

$$\dot{x}_t = A_t x_t + w,\ w \sim \mathcal{N}(0, W) \tag{5.26}$$

$$\dot{P} = A_t P + P A_t^T - \sum_{i=1}^{2} P H_i^T V^{-1} H_i P + W \tag{5.27}$$

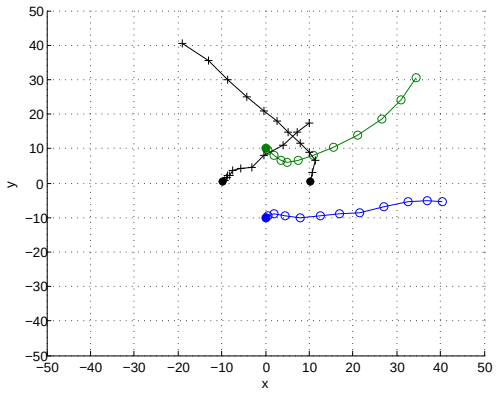
Quadratic Approximation of Utility

The team cost function can be derived to obtain Equation 5.28. This equation is the negative of mutual information. For the approximation of the Hessian, the Jacobian of the cost function needs to be calculated at the expected state and then the difference in state and Jacobian is used to recursively approximate the Hessian through Equation 5.23. The Jacobian needs to be computed relative to the estimate of x_t, the x_i's and P. The Jacobian of the cost function can be derived through straight-forward arithmetic.

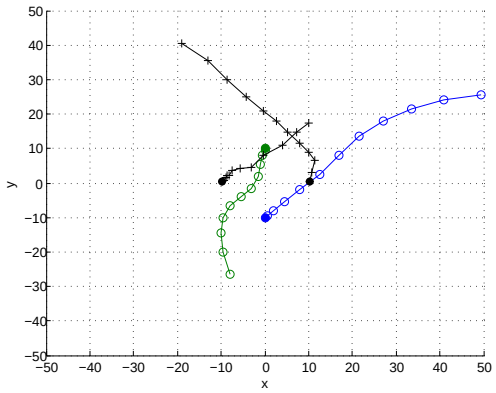
$$g = \text{tr}\left(A_t + A_t^T - \sum_{i=1}^{2} H_i^T V^{-1} H_i P + P^{-1} W \right) \tag{5.28}$$

Results

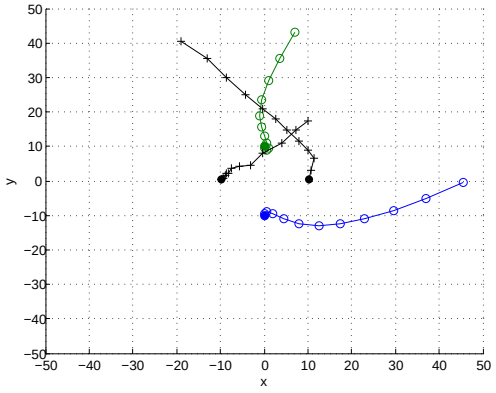
To evaluate the benefit of extended-LQISO, the information gathering task was simulated using three different communication strategies. Figure 5.2 displays the paths taken by the robots for each of the three different cases. In the case of Figure 5.2a, the strategy was set so that robots cooperate at every third time step. In the case of Figure 5.2b, extended-LQISO was used to control when the robots cooperated. In the case of Figure 5.2c, robots cooperated at every time step. Figure 5.3a and Figure 5.3b display a plot of the corresponding entropy of the targets estimate and the total data transferred over the network respectively, for all three cases. By observing Figure 5.3b, it is clear that the extended-LQISO case (crosses) used less than 60% of the total communication bandwidth required by the full communication case (circles). Meanwhile, the entropy reduction performance was only affected during a few time steps, as shown in Figure 5.3a. On the other hand, unlike the subsampling case



(a) Resulting paths where robots cooperated at every third time step



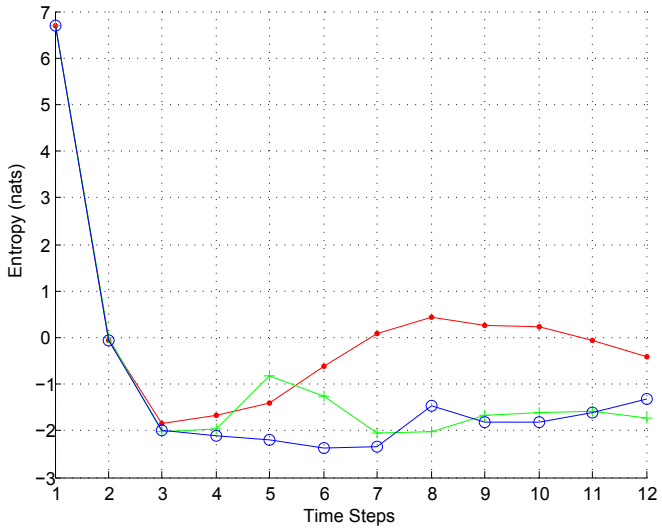
(b) Resulting paths where robots used the extended-LQISO algorithm



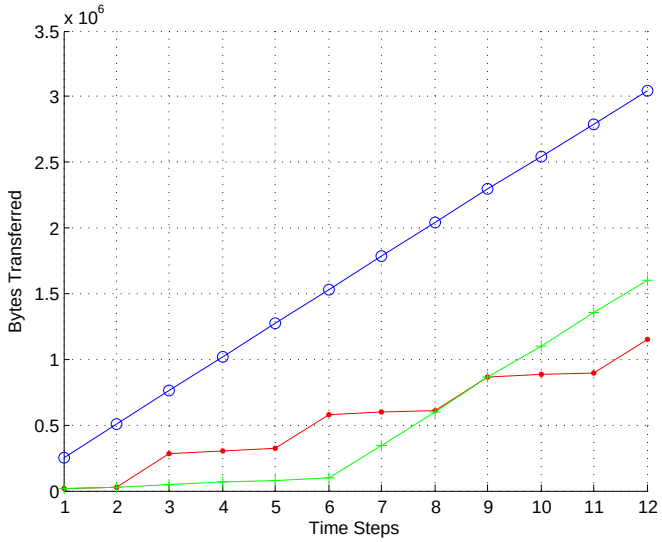
(c) Resulting paths where robots cooperated at every time step

Figure 5.2 – Resulting paths of robots using three different communication strategies. The corresponding entropy reduction and communication loads for the three cases are shown in Figures 5.3a and 5.3b Robot paths are represented by the lines with circle markers. Target paths are represented by the lines with cross markers. The starting positions are indicated by the filled circles.

(dots), extended-LQISO detected an increased need for cooperation after time step 6 and hence managed to maintain its information gathering performance until the conclusion of the simulation. The fact that extended-LQISO is observed to outperform the case where robots consistently cooperate is due to the randomness induced into the observations used in the simulation to mimic a real-world sensor.



(a) Resulting target estimate entropy for the three different strategies. The dot-marked line represents the case where robots cooperated every third time step. The cross-marked line represents the case where robots used extended-LQISO. The circle-marked line represents the case where the robots cooperated at each time step.



(b) Communication required for the three different strategies. The line styles correspond to the same cases as described in Figure 5.3a. The slope of the lines indicate whether the robots cooperated at that time step. A small amount of data is still transmitted when robot do not cooperate. This is due to other tasks requiring communication such as data fusion.

Figure 5.3 – Entropy and communication using different communication strategies.

5.4 Solution to Negotiation-DIF

In this section, we introduce our solution to negotiation-DIF attained by integrating extended-LQISO with our solution to min-cost-DIF. In addition to the composition of the two algorithms, we show how extended-LQISO provides the DIF framework with a sensor utility based on the impact on control actions and at the individual target granularity level.

Our solution to negotiation-DIF consists of our solution to min-cost-DIF and extended-LQISO running concurrently using the same communication costs for links utilising the same communication resource. Extended-LQISO runs as described in Section 5.3 and our solution to min-cost-DIF is used to adjust data fusion in a manner identical to Section 4.1. However, in this case, the sensor utilities are estimated with the aid of extended-LQISO.

The outcome of extended-LQISO in Section 5.3 is a positive definite matrix K obtained for each time step. The row block K_i of the matrix provides an approximation of the coupling between the control action of robot i and the current estimate and the state of other robots. This can be seen clearly for the case of a two-state environment and three robots with the aid of Equation 5.29 where $\{j_1, j_2\}$ correspond to the environment estimate column blocks and $\{k_1, k_2, k_3\}$ correspond to the robot blocks. These coupling values are in one-to-one correspondence with the links shown in Figure 5.4.

While the coupling terms between the control action and the state of other robots are used to determine the frequency of negotiation required between robots as outlined in Section 5.3, in this case, the terms between robot states and the current estimate are also used for the purpose of estimating sensor utility. Since a separate term is obtained for each element of the estimate state vector, the sensor utility can be estimated down to the individual element level. For the case of target tracking, this granularity means that robots can determine which target is of higher priority to its decisions at the current system state.

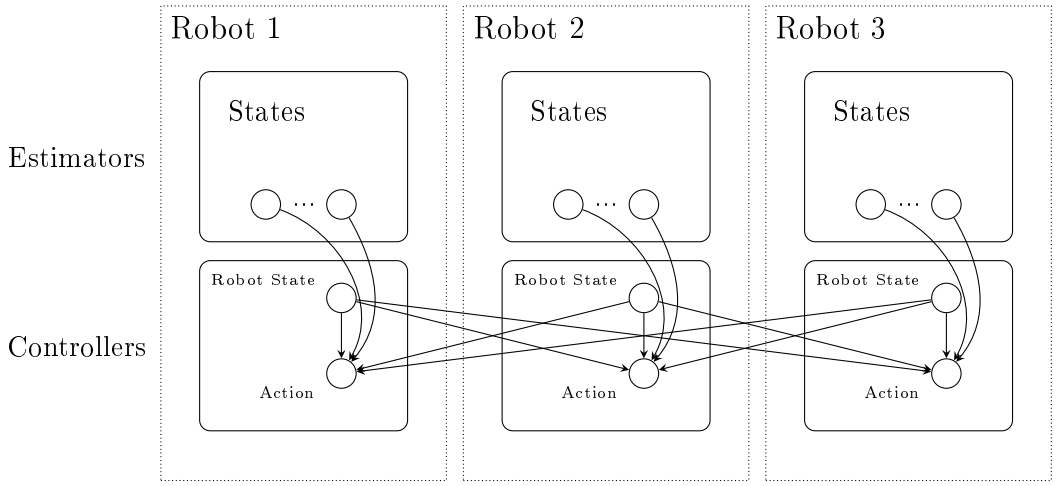


Figure 5.4 – Information structure topology of a decentralised information gathering system.

The sensor utility is estimated as follows. The entropy reduction is calculated for each target. Then, the obtained value for each target is multiplied by the L^2 norm of the block in the coupling matrix relating the robot state to that target's estimate. The obtained values are then summed to compute the sensor utility.

$$u_i = -R_i^{-1} B_i^T \begin{bmatrix} K_{i,j_1} & K_{i,j_2} & K_{i,k_1} & K_{i,k_2} & K_{i,k_3} \end{bmatrix} \begin{bmatrix} x_{j_1} \\ x_{j_2} \\ x_{k_1} \\ x_{k_2} \\ x_{k_3} \end{bmatrix} \tag{5.29}$$

5.5 Summary

In this chapter, we presented a solution to the negotiation-DIF problem defined in Section 3.4. The solution was obtained by combining our solution to min-cost-DIF with extended-LQISO presented in Section 5.3. The result of our negotiation-DIF solution is a complete communication efficiency solution for information gathering.

Chapter 6

Experiments

This chapter contains the results of experiments validating our algorithms presented in Chapter 4 and Chapter 5. The main aim of the experiments is to demonstrate that the suggested algorithms enhance the communication efficiency of decentralised information gathering systems. The experiments also aim to demonstrate the scalability and flexibility of the DIF formulations and their corresponding solutions. Not only do the experiments aim to validate technical correctness, but they also aim to reveal interesting qualitative behaviour that can be intuitively related to communication efficiency.

We first describe our indoor and outdoor experimental systems in Section 6.1 and Section 6.2 respectively. Then, the results for min-cost-DIF are shown in Section 6.3 followed by the results for threshold-DIF in Section 6.4. The results for negotiation-DIF are shown in Section 6.5. Finally, Section 6.6 provides a discussion of the significance of the results presented and highlights the lessons learnt.

6.1 Indoor Experimental System

The indoor test site is a bounded and unobstructed ground space in the field lab at the Australian Centre for Field Robotics (ACFR) with approximate dimensions of

Figure 6.1 – The Pioneer P3-DX robot used in the indoor experiment.

5m×5m. For our indoor mobile robot experiment, we use a modified ActiveMedia Robotics Pioneer P3-DX robot. An image of the robot is shown in Figure 6.1. The robot is retrofitted with an on-board computer with an Intel Atom N270 1.6 GHz processor. The robot is equipped with a SICK LMS291 2D lidar used exclusively for localisation. The robot has an on-board Logitech webcam used as a 2D bearing-only sensor. Object detection is implemented using a crude colour-based object extraction method under controlled background and lighting conditions. The ground station for the experimental system is a standard laptop that is connected wirelessly to the robot and is used to monitor the experiment status. The laptop also acts as the off-board processing station. Software is written in the ROS framework. For simulation, Gazebo was used to simulate the platforms as well as the environment.

Figure 6.2 – The Segway RMP 400 robots used in the experiments.

6.2 Outdoor Experimental System

The main components of the outdoor experimental system are the mobile robots, the experimental site, the ground station, the communication system and the software. Each of these components is described in the sections that follow.

6.2.1 Mobile Robots

Our outdoor mobile robot experiments use two modified Segway RMP 400 robots. An image of the robots is shown in Figure 6.2. The first robot is equipped with a Velodyne 3D Lidar with a 360° field of view. The second robot is equipped with a 2D SICK LMS291 horizontally mounted laser scanner with a 180° field of view. Each robot is also equipped with a server-class computer with an eight-core processor. For localisation, the two robots rely on high-accuracy Novatel inertial measurement unit (IMU) and differential global positioning system (DGPS) modules.

Figure 6.3 – The outdoor experimental site.

6.2.2 Experiment Site

The outdoor experiments were conducted in a semi-urban environment at The University of Sydney. The site is a rectangular-shaped lawn outside the ACFR with approximate dimensions of 12m×30m shown in Figure 6.3.

6.2.3 Ground Station

The ground station used for the outdoor experiments comprises two standard laptop computers, one for each robot, functioning as control stations and monitors and a third laptop functioning as the main control terminal. Communication with the robots takes place over a 5GHz 802.11a WiFi network. The network is formed of two Netgear wireless access points connected using 802.3ab gigabit Ethernet at the ground station and a Netgear wireless network card in each of the robots. The off-board stationary camera used as a third sensor node in the experiments is a Prosilica GC2450 camera acting as a bearing-only sensor located at the ground station.

6.2.4 Communication System

The communication system is depicted schematically in Figure 6.4. Each of the two robots connects directly to its own access point over 5GHz WiFi. These two wireless links are set to a separate channels to prevent interference. Experimental testing conducted on the wireless hardware resulted in a maximum data-rate of 20Mbps for each link. The two access points are connected to a gigabit Ethernet switch to which the camera and control station computers are connected. A third access point also connected to the switch, denoted as WAP3 in the figure, provides internet access to the system.

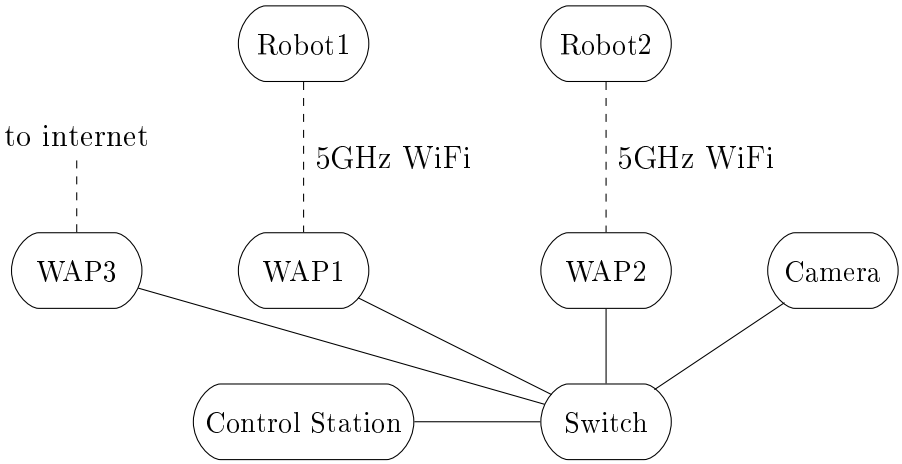


Figure 6.4 – Communication system used for outdoor experiments. Solid lines represent gigabit Ethernet connections while dashed lines represent wireless connections.

6.2.5 Software

All DIF algorithms are implemented as a distributed multi-node ROS system. An EKF implementation is used for estimation and data fusion. For the experiments, the DIF algorithms are added as an auxiliary layer to an existing LIDAR object detection system [22, 70]. Object detection for camera images is achieved using fiducial markers generated and detected by ArUco library [31]. The min-cost-DIF and threshold-DIF experiments rely on the path planner described in [106]. On the other hand, negotiation-DIF experiments use a discrete-action fixed-horizon planner

that selects actions based on an exhaustive forward search. For simulations, the DIF algorithms and path planning software are identical to the case of hardware experiments. Sensor observations are simulated at the point observation level. The platform is also simulated through a simplified dynamics model.

6.3 Min-Cost-DIF

This section presents the results of our solution to min-cost-DIF demonstrated on scenarios where the link costs are readily attainable. We present results for two scenarios. The first scenario was tested in simulation and involves two robots. The purpose of this scenario is to validate the advantage of our solution to min-cost-DIF over simple down-sampling of information. It also highlights the benefits of assuming the ability for multicast. The other scenario was tested in both simulation and hardware and involves a robot aided by a processing ground station. The purpose of this scenario is to demonstrate the flexibility of min-cost-DIF in allowing for the dynamic selection of a sensor-data processing location.

6.3.1 Two-Robot Simulation

We evaluated our solution to min-cost-DIF for a scenario consisting of two mobile robots tracking a moving target. The aim of this simulation is to demonstrate the multicast behaviour of dynamic information flow and to show the improvement in information gain realised in the min-cost-DIF setting in comparison to uniform down-sampling of data rates.

The experimental setup is shown in Figure 6.5 and is based on the outdoor experimental system described in Section 6.2. The two mobile robots are assigned to separate workspaces with approximate dimensions of 5m$\times$5m each. The first robot has a 360° field-of-view sensor and the second robot has a 180° field-of-view sensor. The sensors are bearing-only sensors, forcing cooperation between the robots. A moving target

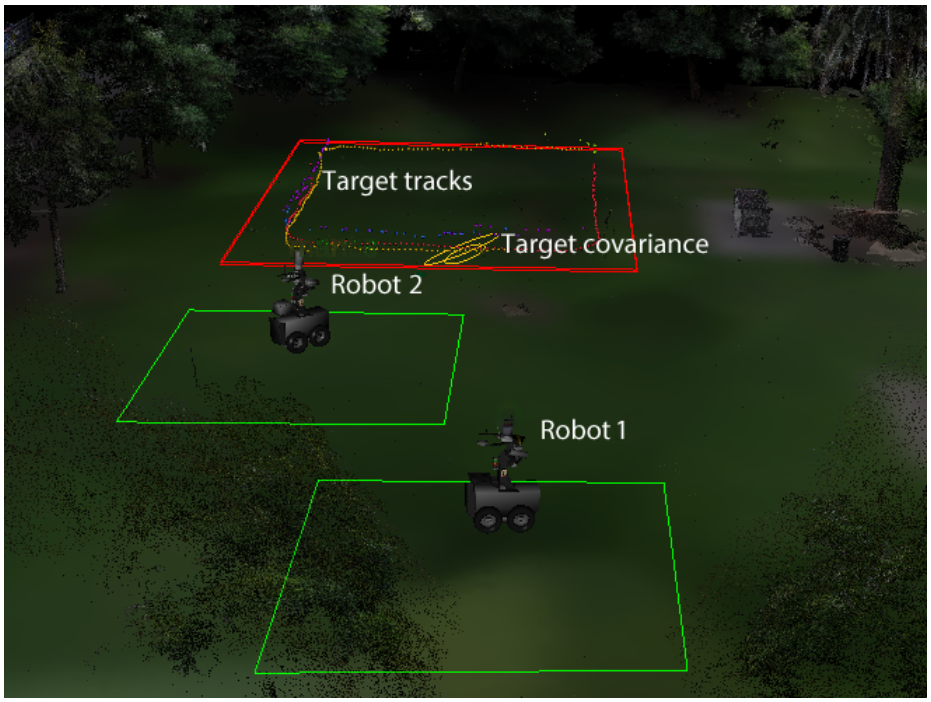


Figure 6.5 – Demonstration setting of the two-robot simulation and experiment. Robots 1 and 2 are shown with their boundaries. Sample target tracks are shown making a square pattern inside the region of interest.

tracked by the robots moves in a circular pattern within a square region of interest outside the robots' workspaces. The region is approximately 10m×10m in size.

The network diagram of the system is shown in Figure 6.6. The object detection routine on each robot imposes a processing cost. Due to the multicast property, the processing cost of each processing module is distributed among the receiving estimators. Therefore, if the estimators collectively evaluate an observation utility greater than the processing cost, then the sensor raw data should be processed and sent forward. Virtual links, not shown in the figure, allow for the no-send policy. The cost of the links incident onto the estimators is adjusted throughout the simulation by the robots' sensor utility evaluations.

To validate the performance of the algorithm, a control test was also run. In the control test, the sensor rates were reduced to the same average rate used in the dynamic case. The control test is also referred to as *down-sampled* communication.

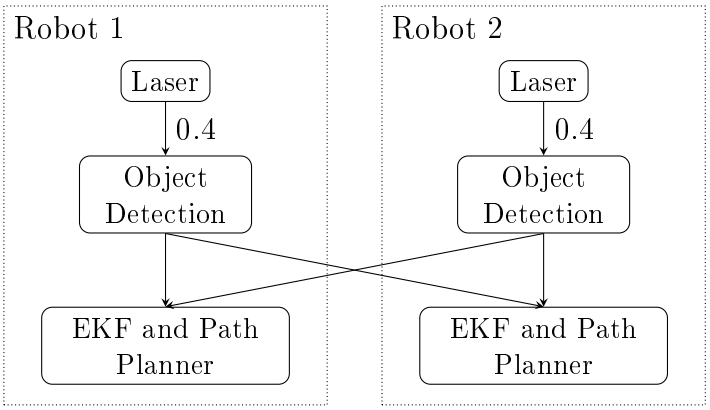
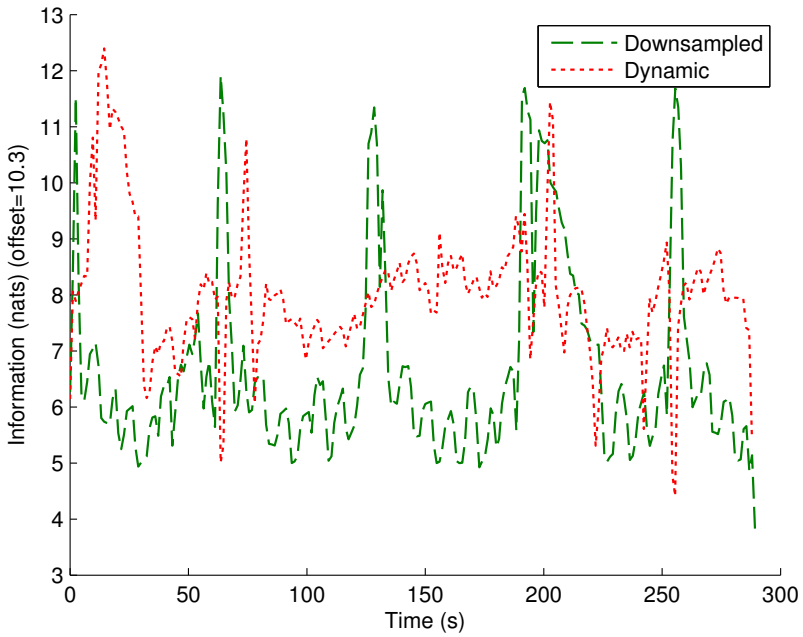


Figure 6.6 – The min-cost-DIF network diagram for the two-robot simulation. Virtual links are omitted for clarity.

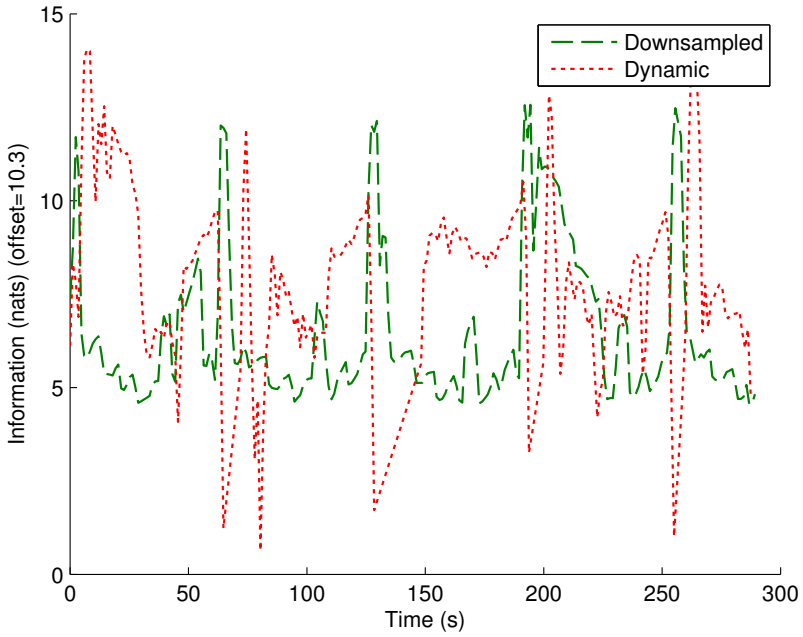
The information for each of the robots' estimate over time is shown in Figure 6.7 and the corresponding average bars are shown in Figure 6.8. In Figure 6.7a, the information in Robot 1's estimate is higher on average for the dynamic flow case. By using a fixed rate, the performance of the down-sampled displayed long periods of poor information gathering performance for particular system configurations; for example, when the target was outside the field of view of Robot 2. This behaviour was prevented in the case of dynamic flow since inter-robot communication was boosted when required. The advantage of the dynamic flow case can also be seen more distinctly in Figure 6.8a. The dynamic case for Robot 2 observed some lag in information gathering performance initially. This may be attributed to the delay in sensor utility estimation. The dynamic case eventually outperformed the down-sampled case as shown in Figure 6.7b and its overall advantage is confirmed in Figure 6.8b.

The flow rates and estimated sensor utilities for the dynamic flow case are plotted against time in Figure 6.9. The flow rates shown correspond to the percentage of observations transmitted while the sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation. The plots show consistency between sensor utility and flow. In particular, the oblique parts of the flow curve correspond to time periods where a robot would receive sensor observations freely due to raw data already being processed for the other robot. This free reception of data is a feature of multicast routing.

As noted in Section 4.1.3, discrete flow decisions minimise the error from our maximum flow approximation of the total flow in each link. As seen in Figure 6.9, with the exception of transient periods, the flow variables were mainly either 0 or 100.

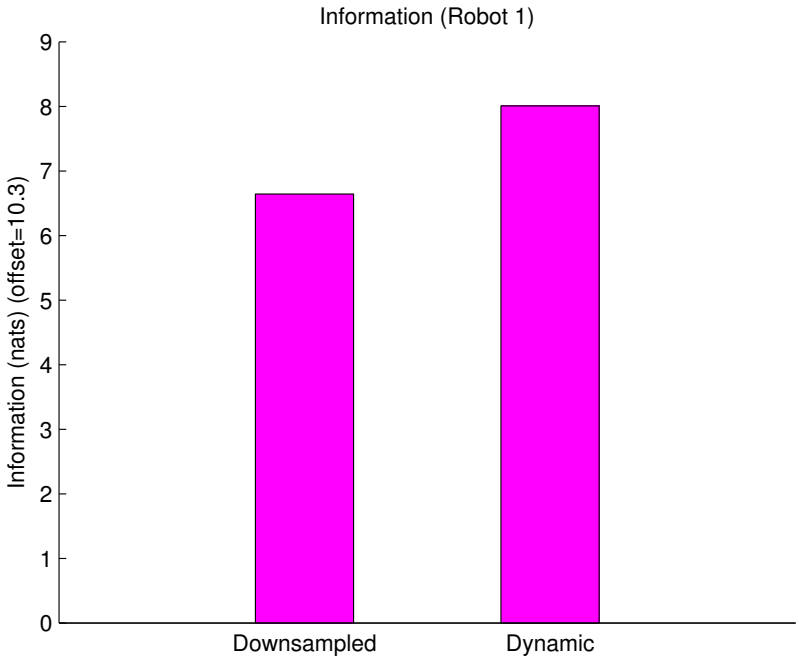


(a) Robot 1

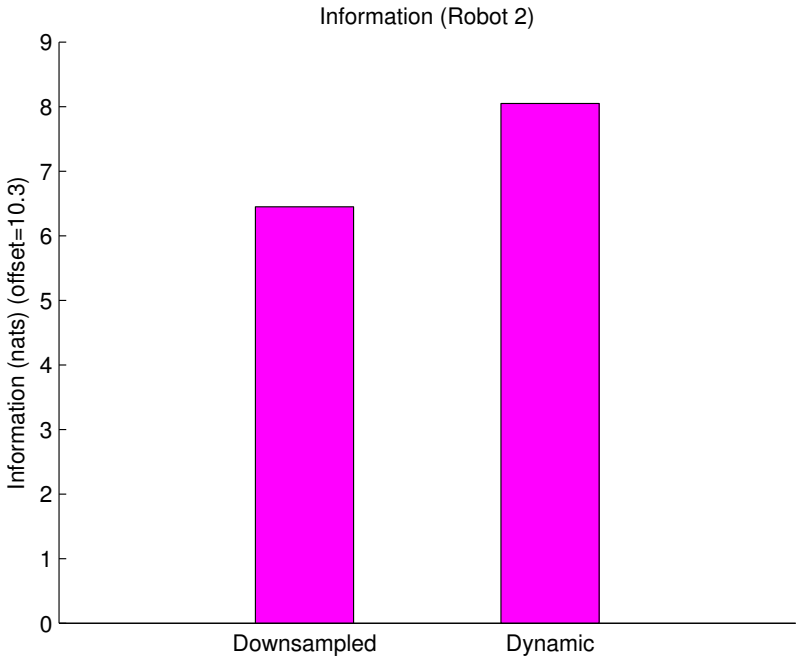


(b) Robot 2

Figure 6.7 – The information value (negative entropy) of the robots' target estimate for the two-robot simulation. The plots are shown for two communication methods, sub-sampled and dynamic, with both requiring the same amount of computation on average. The values correspond to the negative entropy with the addition of an offset value.



(a) Robot 1



(b) Robot 2

Figure 6.8 – Time averages of the plots of Figure 6.7 shown in bar format. The improvement for the dynamic case over sub-sampling is clearly observed.

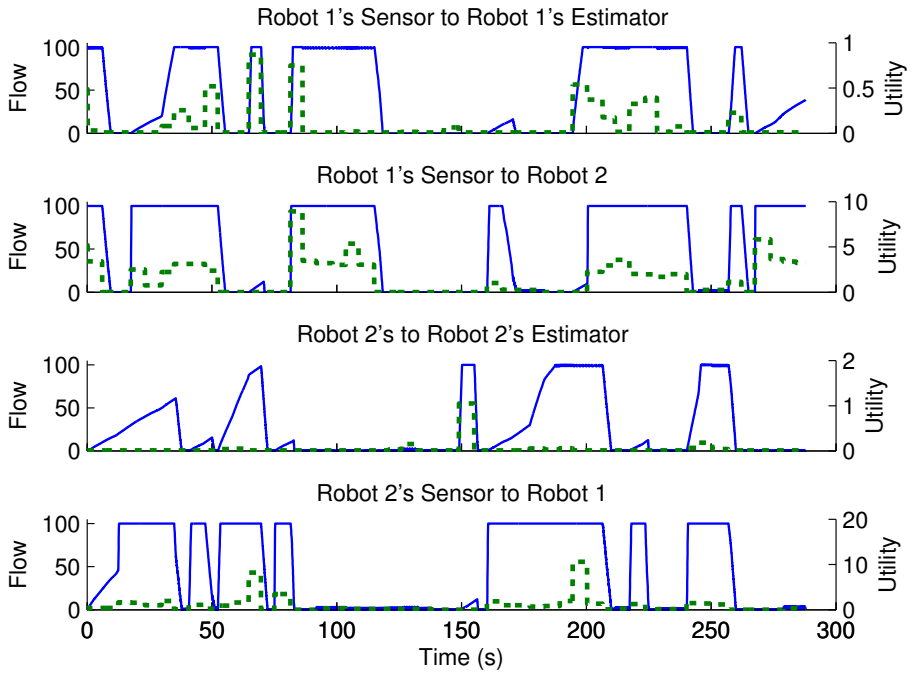


Figure 6.9 – Flow rates and sensor utility for the two-robot simulation. Flow rates are shown in solid lines while the evaluated sensor utility is shown in dashed lines. The flow rates shown correspond to the percentage of observations transmitted while the sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation.

6.3.2 Robot and Ground Station Simulation

We also evaluated our solution for a scenario including one mobile robot and one ground station. The main purpose of this simulation is to show the generality of the min-cost-DIF approach.

Transmitting raw data to be processed at a ground station with increased computational resources can reduce the time delay observed in processing. This reduction, in turn, can have a positive effect on information gathering performance. However, this reduction is only possible if the delay caused by transmission does not outweigh the performance advantage of the extra computational resources. In addition, on-board processing can also cause reduced battery life for mobile robots. However, relaying processing off-board is only advantageous if transmitting raw data does not require similar amounts of energy. This simulation and the hardware experiment that follows demonstrate the ability of min-cost-DIF to represent such scenarios.

The experimental system corresponds to the simulation case of that described in Section 6.1. It includes one mobile robot equipped with a camera and a ground station. The robot's on-board processing is computationally expensive; however, it has wireless access to the off-board processing station. The demonstration begins with the robot near the processing station. The robot then proceeds to gain information about a moving target. As the robot distances from the processing station, the communication cost increases.

The network diagram of the demonstration is depicted in Figure 6.10. Subject to communication cost, which is set proportional to distance, the robot's decision is expected to vary between on-board and off-board processing. The maximum flow assumption is valid for this scenario since there is only one destination node.

Figure 6.11 shows the flow between the robot and the ground station, the communication cost and the robot's processing cost over time. The processing cost is assumed to be a fixed value of 0.6 as shown in the figure. The communication cost is set to the distance between the robot and the ground station and the flow multiplied by 0.1. In Figure 6.11, at around time step 50, communication cost outweighs the cost

of on-board processing. This leads to a switch from off-board processing to on-board processing. It should be noted that the communication cost displayed in the plot is only for transmission from the sensor to the processor. When compared with on-board processing cost, the total communication cost is doubled. The results of this simulation show the generality of the dynamic flow formulation. Even though the expected behaviour here is quite obvious, the aim is to show that this behaviour was achieved within the min-cost-DIF framework without modification.

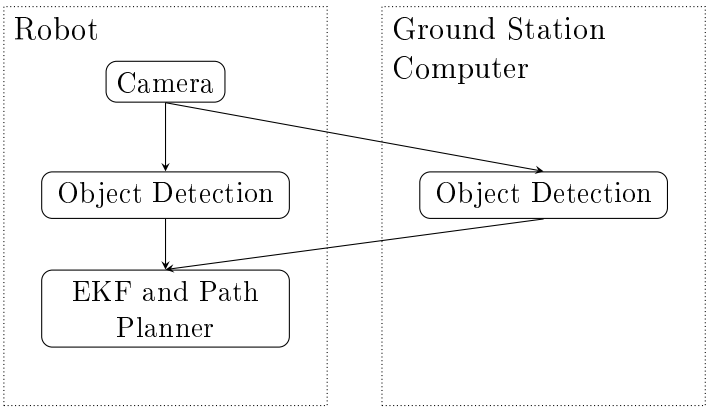


Figure 6.10 – The min-cost-DIF diagram of the robot and ground station demonstration scenario.

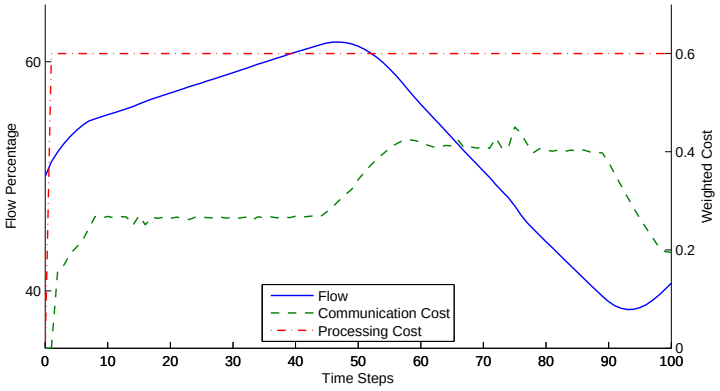


Figure 6.11 – Flow, communication cost and processing cost of the link from the sensor to the ground station processor for the one robot/one ground station simulation.

6.3.3 Robot and Ground Station Experiment

We also validated the demonstration scenario of Section 6.3.2 in a hardware experiment using the experimental system described in Section 6.1. The mobile robot used the on-board webcam as a 2D bearing-only sensor to track a moving target. With the exception of the off-board processing ground station, all processes were executed on-board the robot computers including localisation, image processing (when required), estimation, decision making and the information flow control algorithm.

The change in flow, communication cost and processing cost over time is shown in Figure 6.13. These results are consistent with the simulation results for the analogous case. We observe that the robot initially chooses to send images to be processed off-

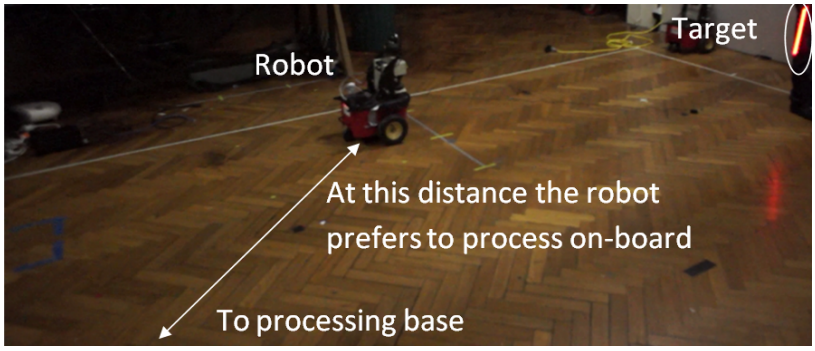


Figure 6.12 – Snapshot from the one robot/one ground station experiment. The image shows the robot having moved away to follow the target. At this distance, the robot prefers to perform processing on-board.

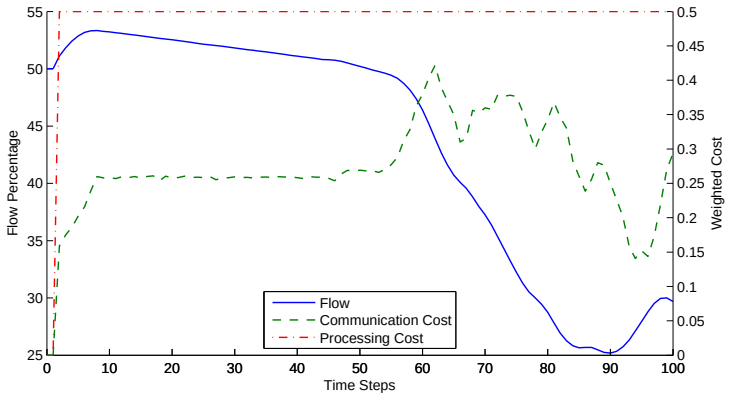


Figure 6.13 – Flow, communication cost and processing cost of the link from the sensor to the ground station processor for the one robot/one ground station experiment.

board by the ground station. The ground station performs object detection and sends point observations back to the robot. As the robot moved away to track the target, the communication cost increased and thus the robot chose to perform processing on-board. A snapshot of the moment when the robot decided to switch to on-board processing is shown in Figure 6.12.

6.3.4 Monte Carlo Simulation

We performed a Monte Carlo simulation for the experimental setting of Section 6.3.1, comparing the dynamic communication with down-sampling. The aim of the simulation is to analyse the statistical significance of the performance advantage introduced by our solution to min-cost-DIF.

The dynamic communication case was compared against two down-sampling rates. The first rate of 50% is chosen to be approximately equal to the average usage of 51.5% for the dynamic case while the second rate is 60%. Each method was tested in twenty randomised trials running for three minutes each.

The results of the Monte Carlo simulation are shown in Figure 6.14 in box-plot format. The results shown assume each trial as one sample. Dynamic communication outperforms the down-sampling rate of 50% with a Welch's t-test for statistical significance resulting in a p-value of 0.0143. The p-value for the comparison against the down-sampling rate of 60% is 0.5086 which shows that the performance of the dynamic case is comparable to that of a 60% down-sampling rate.

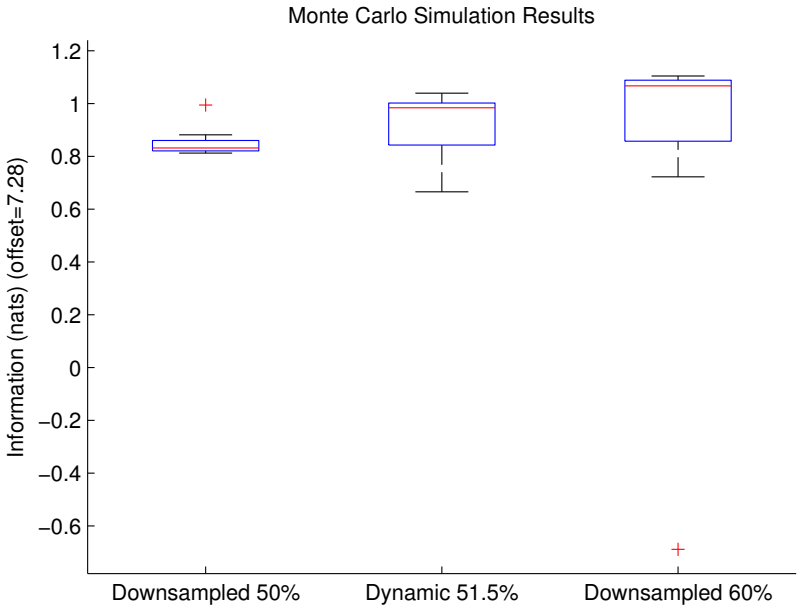


Figure 6.14 – Monte Carlo simulation results comparing down-sampling with dynamic information flow. Each method was tested on the two-robot scenario for twenty trials running for one minute each. In the results depicted, a trial acts as one sample. The box extents represent the first and third quartile, while the whiskers represent the extrema. The median is represented by the horizontal line inside the box. The values correspond to the negative entropy with the addition of an offset value.

6.4 Threshold-DIF

This section presents results of demonstrations illustrating different features of our solution to threshold-DIF. These features include the ability to explicitly assign global communication limits, improved information gathering performance and the ability to switch between sensors on-line. Two information gathering demonstration scenarios were tested in addition to a multiple-node simulation. The first scenario, tested in a hardware experiment, aims to demonstrate how two robots can intelligently share limited communication bandwidth to improve information gathering performance. The second scenario introduces an auxiliary stationary camera that transmits raw images wirelessly. The camera needs to interrupt any data fusion communication between the robots to send images. We show in both a simulation and hardware experiment how this is achieved within the threshold-DIF framework. Finally, we show the results of a multi-node network simulation with the purpose of demonstrating the scalability of our solution. Information rates are assumed by the simulation and sensor utilities are randomised.

6.4.1 Two-Robot Experiment

We tested our solution to threshold-DIF in a demonstration scenario of two mobile robots tracking a moving target. The aim of this experiment is to show the information gain advantage of dynamic information flow in the case of limited inter-robot communication bandwidth.

Experimental Setup

The experimental system used for this experiment is the outdoor system described in Section 6.2, while the experimental setting is the same as that of Section 6.3.1 depicted in Figure 6.5. The two robots are placed in two separate areas with virtually bounded geographical regions to avoid collision. Target tracking is limited to a geographically

bounded region of interest and for the purpose of the demonstration, tracking was limited to one target performing circular patterns.

The network diagram for this demonstration is shown in Figure 6.16. It is assumed that maximum communication bandwidth is limited and does not allow both robots to send sensor data at the full rate. We also assume that communication throughput decreases with inter-robot distance. Therefore, the robots are required to share the available bandwidth. The bandwidth sharing constraint is indicated by the dashed line drawn between the two inter-robot links. Virtual links, not shown in the figure, allow for the no-send decision. It is expected that through dynamic information flow

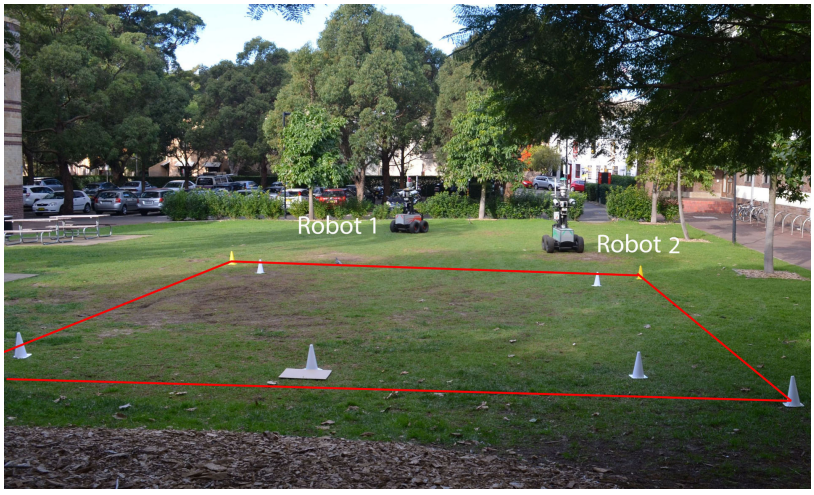


Figure 6.15 – The outdoor experimental setup with robots visible outside the tracking region of interest. The border of the tracking region is designated by solid lines.

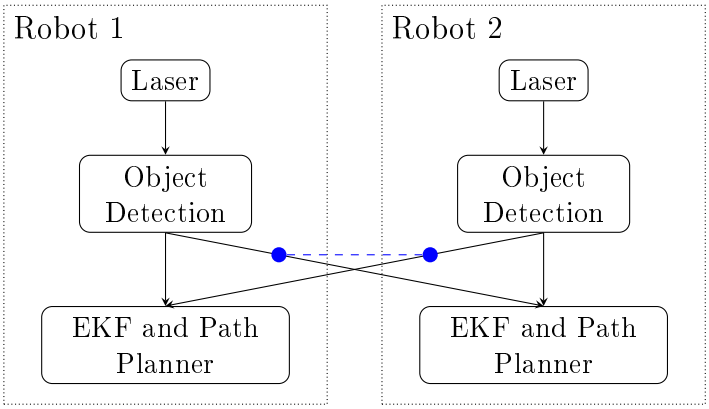


Figure 6.16 – The theshold-DIF diagram for the two-robot scenario. Virtual links are omitted for clarity.

the bandwidth will be shared efficiently with respect to sensor utility. The maximum flow assumption is valid in this scenario since no processing costs are assigned.

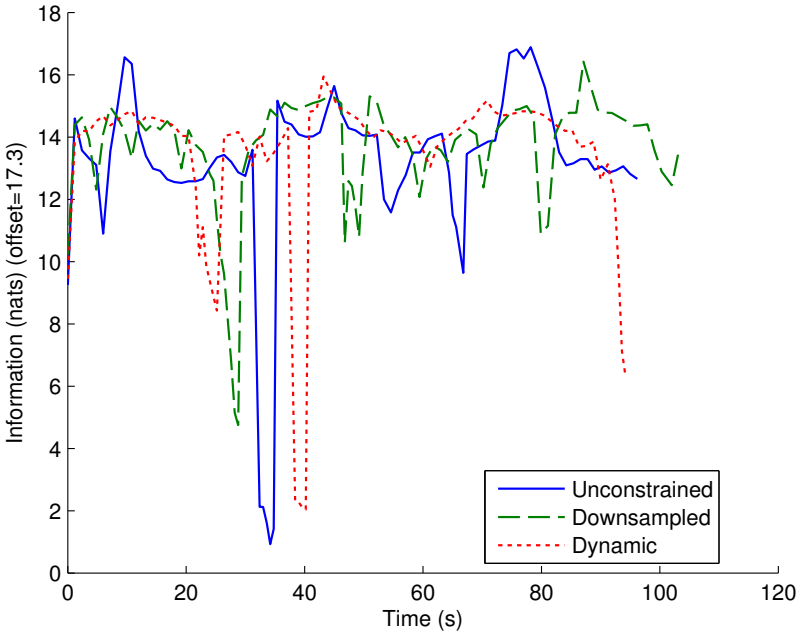
To validate the performance of dynamic information flow, two control tests were run for the purpose of comparison. The first control test allows unconstrained communication between all nodes and shall be referred to as the *unconstrained* case. This test mainly acts as a benchmark since it violates bandwidth constraints. The second control test involves a reduced communication rate that obeys bandwidth bounds. This test shall be referred to as the *down-sampled* case.

Results

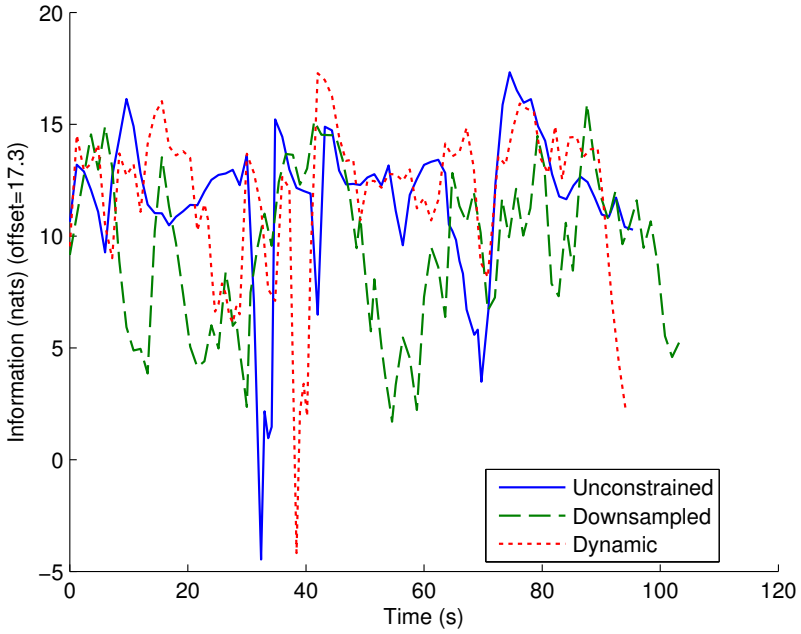
The information value for the robots' target estimates over time is shown in Figure 6.17 with the corresponding average bars shown in Figure 6.18. In both these figures, we observe minimal difference in information value across the three communication methods for Robot 1. Because Robot 1 has a 360° field-of-view sensor, the target is always visible, and therefore tracking does not depend on observations received from Robot 2. However, we do observe a difference in information value for Robot 2. Figure 6.18b shows that the down-sampled method results in reduced information gathering performance when compared to our method. This effect is also evident in Figure 6.17b where there is a clear decline in information for the down-sampled case at times 20 and 60. This decline occurs because the target drops outside the robot's sensor field of view. Dynamic flow ensured that information was directed from Robot 1 to Robot 2, but down-sampling naively shared the communication medium.

The advantage of dynamic information flow is further confirmed in Figure 6.19. The bottom two plots in the figure show the approximate sensor observation utilities and data flow rates between robots over time for the dynamic flow case. The flow rates shown correspond to the percentage of observations transmitted while the sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation. The flow from Robot 1 to Robot 2 dominates

bandwidth usage when the target is not in Robot 2's sensor field of view. However, at times 40 and 80, the flow was directed from Robot 2 to Robot 1 because the target was closer to Robot 2. The top plot shows the distance between the robots over time. Modelling loss of link quality, the weights $\{\nu_{ik}^m\}$ were increased by adding 0.1 of the separation distance. At the average inter-robot distance, the available bandwidth is limited to approximately half of the maximum bandwidth. As expected, the sum of the information flows obeys this reduced capacity.

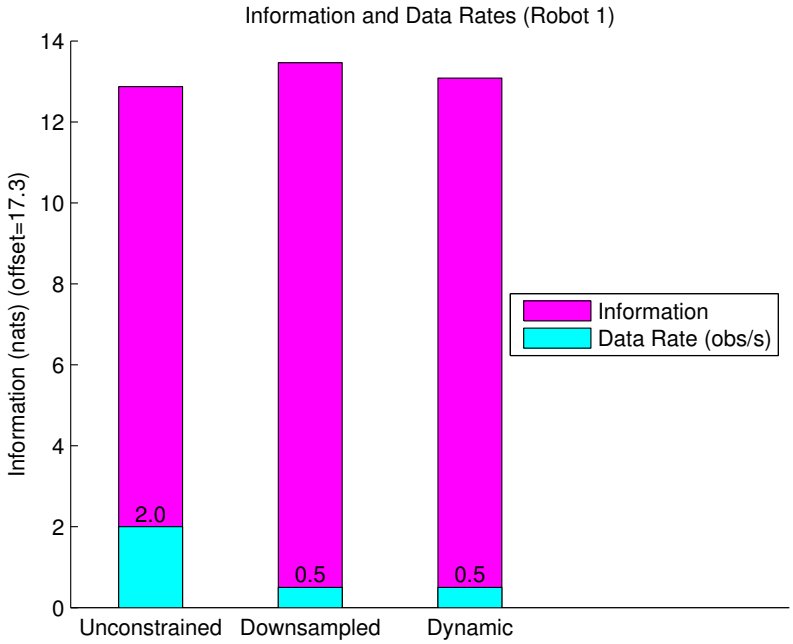


(a) Robot 1

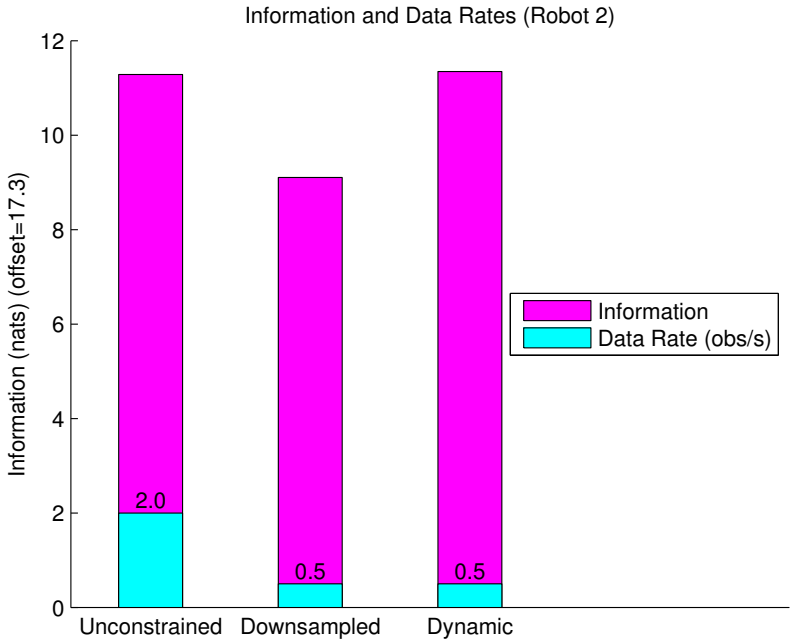


(b) Robot 2

Figure 6.17 – The information value (negative entropy) of the robots' target estimate for the two-robot hardware experiment. Plots shown are for all three communication methods: unconstrained, down-sampled and dynamic. The sudden drops in information are due to target loss. The values correspond to the negative entropy with the addition of an offset value.



(a) Robot 1



(b) Robot 2

Figure 6.18 – Time averages of the plots in Figure 6.17 shown in bar format. Data rates are shown superimposed. The dynamic case shows a clear improvement in information gain in comparison to down-sampling for Robot 2.

Figure 6.19 – Inter-robot flow rates and sensor utility over time for the dynamic flow case of the two-robot experiment. In the first plot, the inter-robot distance is shown. In the lower plots, flow rate is shown as solid lines and sensor utility is shown as dotted lines. Flow rate varies with utility and available bandwidth varies with inter-robot distance. The flow rates shown correspond to the percentage of observations transmitted while the sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation.

6.4.2 Three-Sensor-Node Experiment

We also evaluated our approach in a more complex scenario. This scenario involves the two robots from the previous experiment with the addition of a stationary camera. The aim of this three-sensor-node scenario is to show the generality of our method and to emphasise the multicast behaviour of dynamic information flow.

Experimental Setup

In this scenario, the two robots are aided in tracking by the off-board stationary camera acting as a bearing-only sensor. The camera is placed outside the tracking region of interest opposite the robots. The camera sends raw images to Robot 2 which processes the images and shares the observations with Robot 1. Hence, wireless communication is required for three links: 1) the link from the camera to Robot 2, 2) the link from Robot 1 to Robot 2 and 3) the link from Robot 2 to Robot 1. In the experiment, the robots remained stationary. This does not affect the results of the demonstration since separation distance is ignored in this scenario. Simulation results with moving robots in a similar setup are shown later in Section 6.4.3.

The network diagram is shown in Figure 6.20. The experiment assumes that available bandwidth is sufficient for the two mobile robots to share observations. However, if images are to be transmitted wirelessly from the static camera, then the wireless

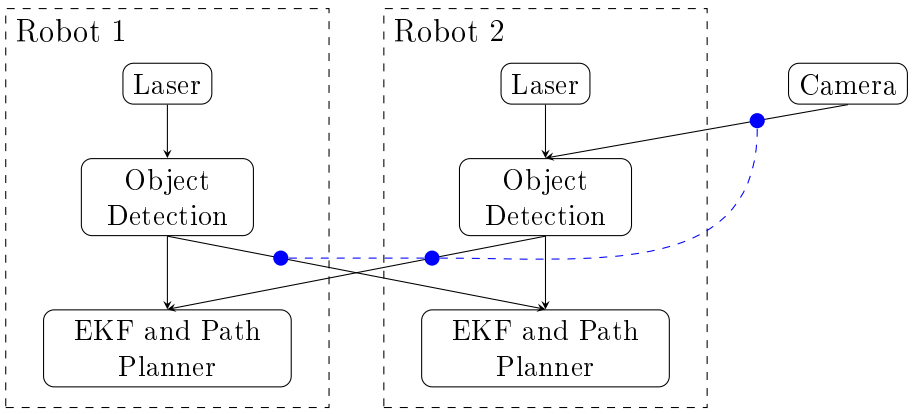


Figure 6.20 – The threshold-DIF diagram for the three-sensor-node scenario. Virtual links are omitted for clarity.

network reaches capacity at only half the full rate of data from the camera. The camera provides accurate tracking when the target is in its proximity and inside its field of view. Dynamic flow is expected to allow images to be sent from the camera in such a situation with the flow from the camera being accompanied by a simultaneous reduction of flow in the other links sharing the medium.

All three communication methods were tested - unconstrained, down-sampling and dynamic. However, for this experiment, images sent from the camera to Robot 2 caused congestion in the wireless network and effectively reduced the unconstrained transfer rate.

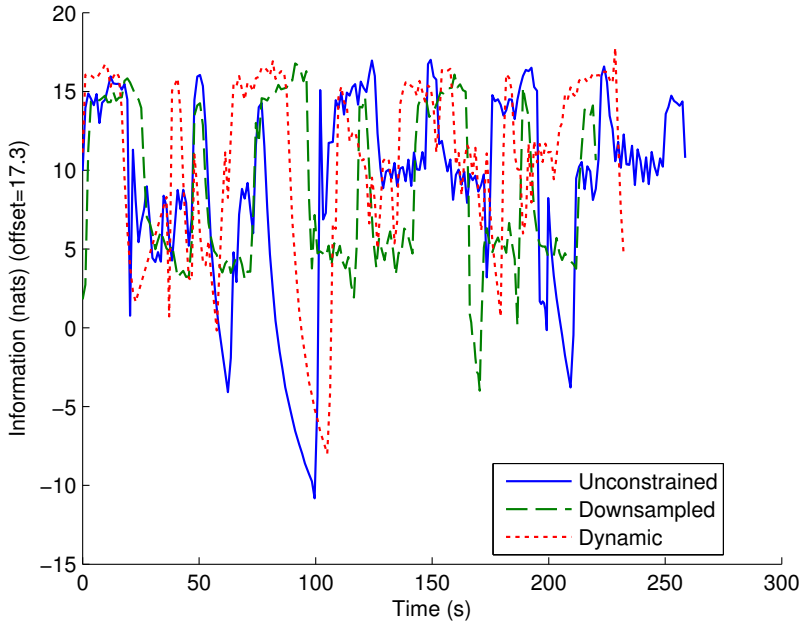
Results

The information value for each of the robots' estimates is plotted over time in Figure 6.21 with the corresponding average bars shown in Figure 6.22. The figures show that the dynamic case outperforms down-sampling for both robots. They also show better performance for the dynamic case in comparison to the unconstrained case for Robot 1. This may be attributed to the fact that the unconstrained case would have failed to produce the desired communication rates due to infrastructure bandwidth limitations. The results for Robot 2 show similar performance between the dynamic case and the unconstrained case.

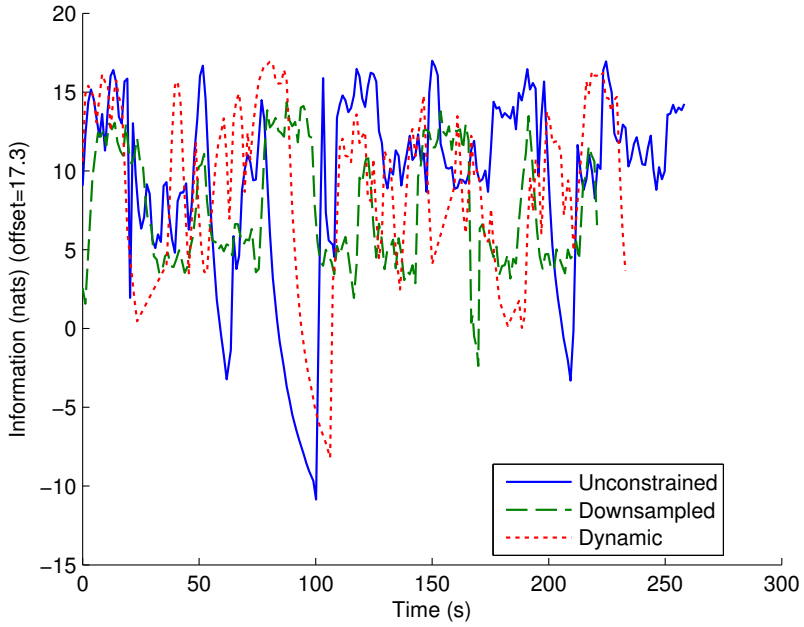
One of the main objectives of this experiment is to highlight the multicast behaviour. Multicast behaviour can be observed by analysing the bottom two plots of Figure 6.23. Figure 6.23 shows the flow rates and approximate sensor utilities for relevant links. The flow rates shown correspond to the percentage of observations transmitted while the sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation. At times 80, 150 and 190, the flow from the camera to Robot 2 retains a high value even though the camera utility for Robot 2 is low during those times. Robot 2 receives these observations without inducing additional cost since observations destined to Robot 1 are processed on-board Robot 2. Robot 1 receives these observations due to their high utility for

Robot 1 but these observations must be processed on-board Robot 2 according to the system architecture. Multicast routing ensures that such flow does not get tallied twice. The top two plots confirm that the system obeys the bandwidth limits as the drop in flow takes place concurrently with the rise in flow from the camera. It should be noted that the maximum possible flow from the off-board stationary camera is only 50% according to bandwidth bounds.

These results also validate the maximum flow approximation of total flow. The flow in each of the inter-robot links shown in the top two plots in Figure 6.23 holds data to only one destination. Therefore, there is no error arising from the maximum flow approximation for those links. The link from the camera holds data destined to both robots. At instances when the camera link is at zero or at maximum flow, there is no loss due to the maximum flow approximation. Also, the number of destinations in this case is two and hence the loss calculated from Equation 4.12 is at most a reduction of 25% of maximum flow for all other instances.

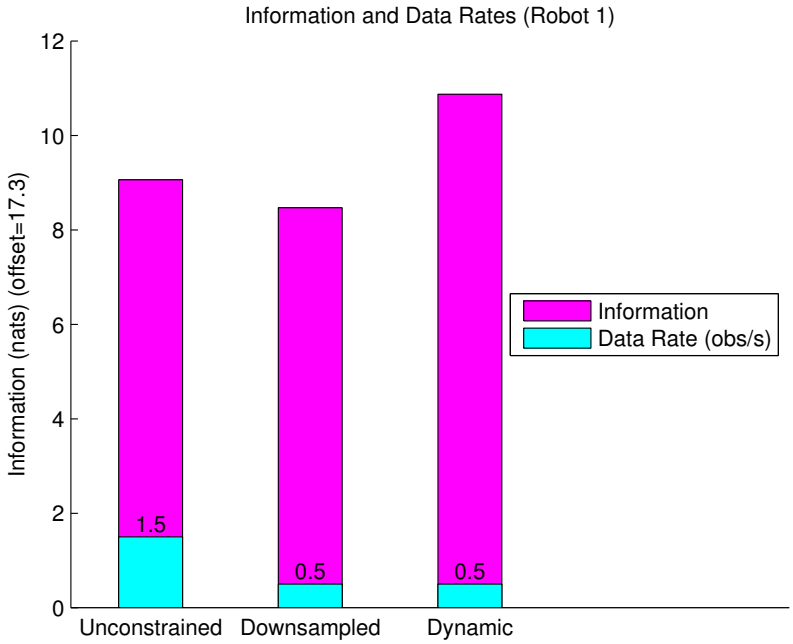


(a)

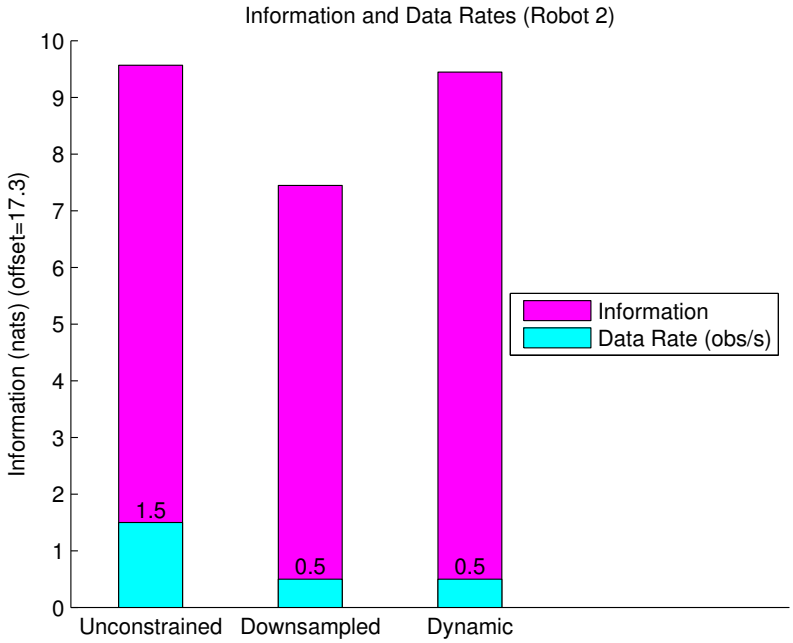


(b)

Figure 6.21 – The information value (negative entropy) of the robots' target estimate for the three-sensor-node experiment. Plots are shown for all three communication methods: unconstrained, down-sampled and dynamic. The values correspond to the negative entropy with the addition of an offset value.



(a)



(b)

Figure 6.22 – Time averages of the plots of Figure 6.21 shown in bar format. The communication data rate is shown to the right of each average bar. The performance advantage of dynamic information flow is clearly observed.

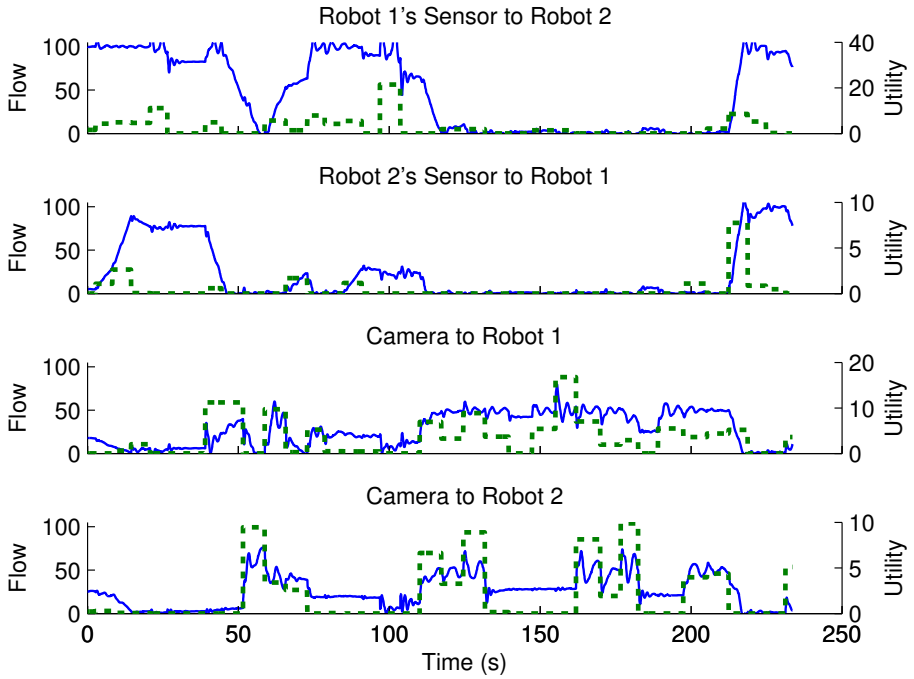


Figure 6.23 – Inter-robot information flow and sensor utility over time for the dynamic flow case of the three-sensor-node experiment. Flow rates are shown in solid lines while sensor utility is shown in dashed lines. Based on the bandwidth constraints and the camera's data rate, 50 is the maximum flow available for the data sourced from the camera. At approximately 120 seconds, communication from the camera interrupts communication between robots due to the increased utility of camera observations. The flow rates shown correspond to the percentage of observations transmitted while the sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation.

6.4.3 Three-Sensor-Node Simulation

We repeated the experiment presented in Section 6.4.3 in simulation. Here, we allow robots to move in order to improve tracking performance.

Experimental Setup

The experimental setting is similar to that of Section 6.4.2. Sensor output was simulated through point observations. Therefore, no raw images were involved and the raw-data communication rate was chosen arbitrarily. In a similar manner to the hardware case, communication is required for the links between the robots as well as the link from the camera.

The network diagram for this experiment is the same as shown earlier in Figure 6.20. Through dynamic information flow, the camera is expected to selectively interrupt communication between the two robots in order to send its images based on the benefit of its observations as evaluated by the robots.

The simulation was run for all three communication methods. The unconstrained communication method is naturally expected to produce higher information gain since the bandwidth bounds are not enforced due to the fictitious data rates.

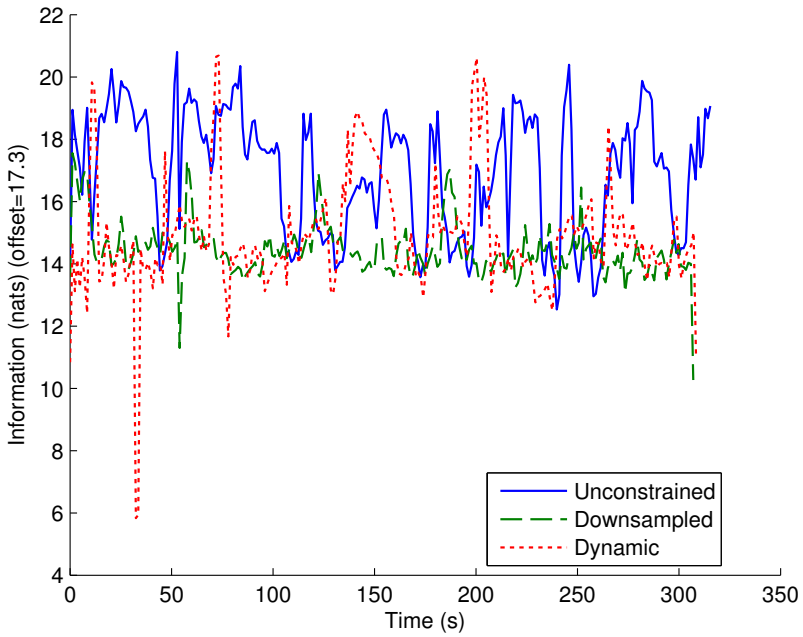
Results

The information value of the robots' target estimates over time is shown in Figure 6.24 with the corresponding average bars shown in Figure 6.25. As expected, unconstrained communication results in higher information on average. However, this communication setting violates the bandwidth bounds. Nevertheless, Figure 6.8 shows that the dynamic case outperforms the down-sampled case for both robots. Figure 6.24 shows that the dynamic case dominates the down-sampled case at time 70, between times 150 and 200 and between times 250 and 300. Observations were received from the camera at these times for the dynamic flow case as shown in Fig-

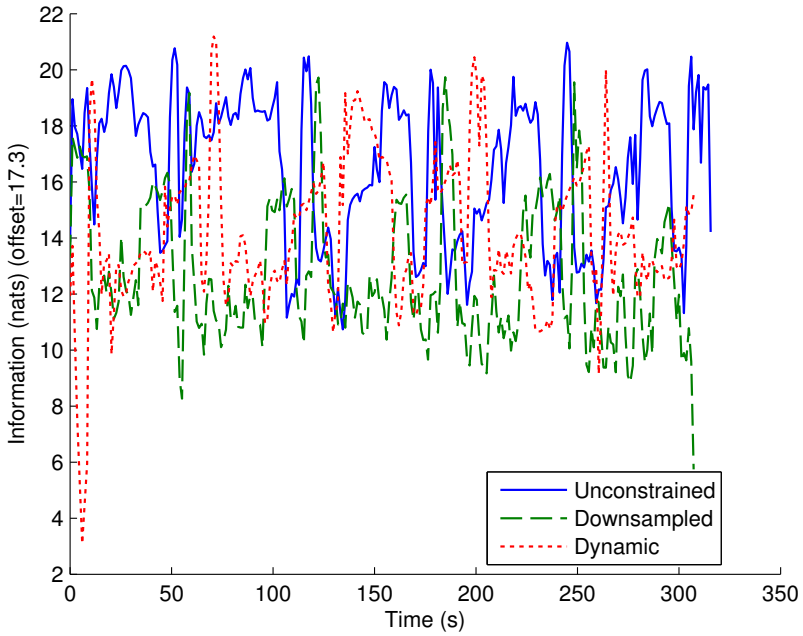
ure 6.26. These times correspond to the configuration where the target enters the camera's field of view and becomes closer to the camera than the mobile robots.

The bottom two plots shown in Figure 6.26 highlight an aspect of multicast behaviour different to that highlighted by the hardware analogue of this simulation. When only one robot evaluates a higher utility for camera observations, such as at times 120, 150 and 220, no significant change in flow is observed. However, when both robots evaluate an improvement in the utility, the increase in the flow from the camera can be clearly seen.

Similarly to the hardware case, we note there is no error arising from the maximum flow assumption in the links between the two robots. The loss occurring in the link from the camera is negligible.

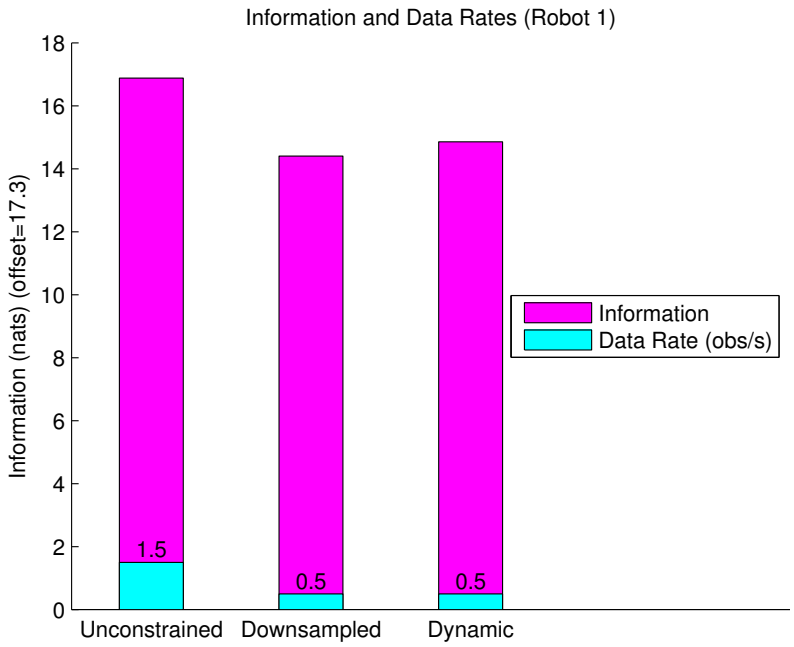


(a) Robot 1

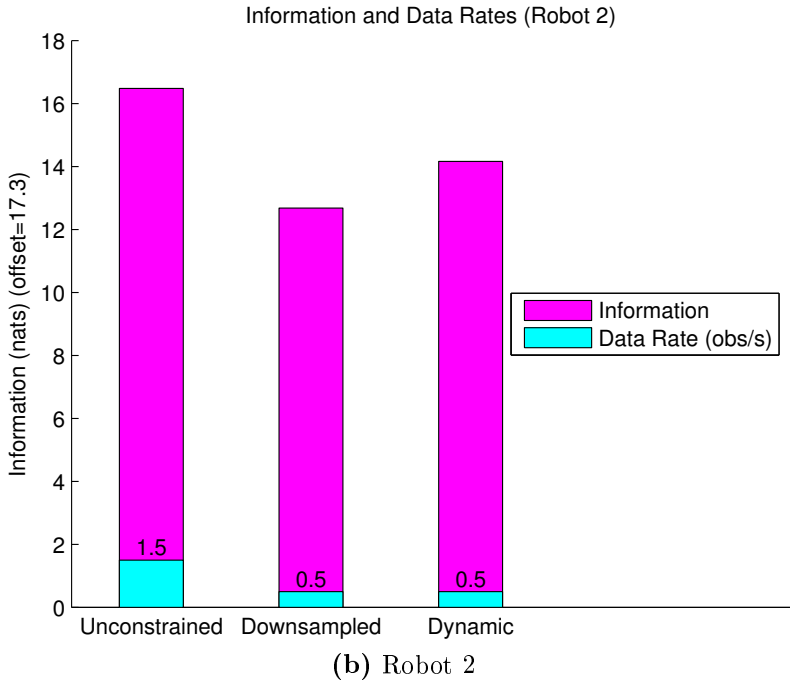


(b) Robot 2

Figure 6.24 – The information value (negative entropy) of the robots' target estimate for the three-sensor-node simulation. Plots are shown for all three communication methods: unconstrained, down-sampled and dynamic. The values correspond to the negative entropy with the addition of an offset value.



(a) Robot 1



(b) Robot 2

Figure 6.25 – Time averages of the plots in Figure 6.24 shown in bar format. The data rates are shown superimposed. The improvement for Robot 2 achieved by dynamic communication over down-sampling using the same communication rates is clearly observed. The unconstrained method violates the bandwidth constraints and is only included as a benchmark.

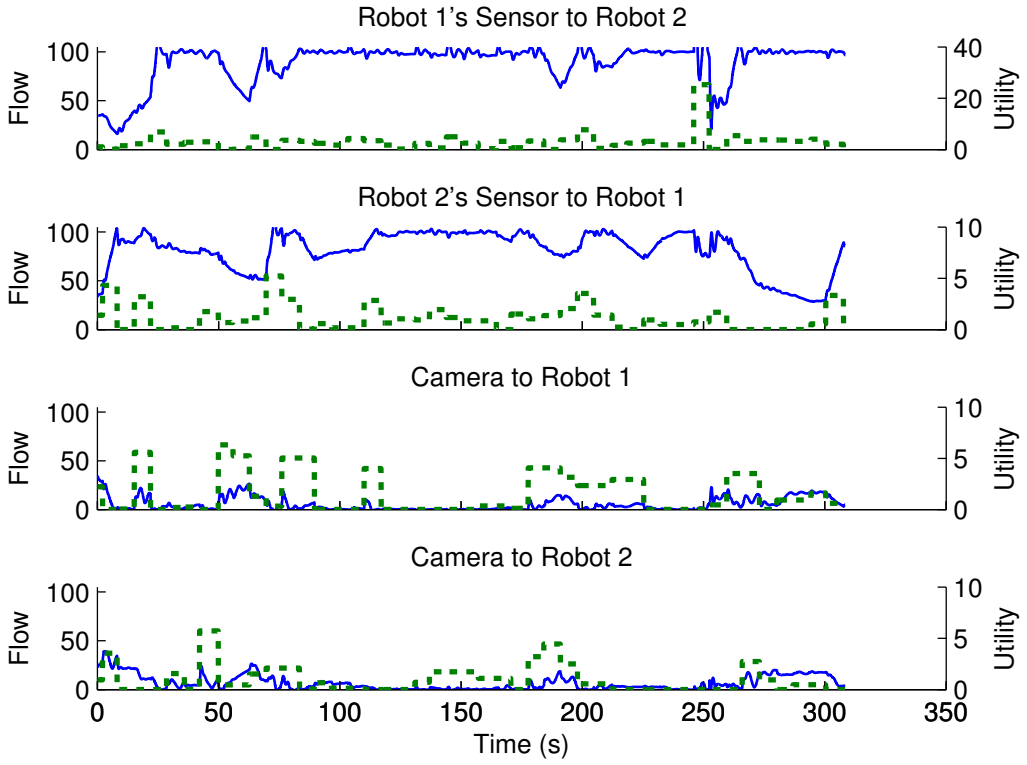


Figure 6.26 – Information flow and sensor utility for the three-sensor-node simulation. Flow rates and sensor utility are represented as in Figure 6.23. The maximum flow rate available for the data sourced from the camera is assumed to be 50. The flow rates shown correspond to the percentage of observations transmitted while the sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation.

6.4.4 Multiple-Node Simulation

We demonstrated our decentralised algorithm on a simulated fifteen-node threshold-DIF network. The purpose of the simulation is to demonstrate our approach for a problem that is not amenable to manually designed communication protocols. The simulation was limited to fifteen nodes to maintain near-real-time performance.

The simulated network comprises five agents each equipped with a sensor, a processor and an estimator. The network has a fully connected topology such that each sensor is connected to all processors and each processor is connected to all estimators. The agents are spatially distributed evenly in a linear manner. Each sensor is assumed to produce data at a rate of 100 units. A global communication constraint of 500 units was applied. In addition, a per-link capacity constraint of 200 units was applied to each processor-estimator link. Sensor utilities were externally randomised and provided to the estimators.

Results

Figure 6.27 displays, in chronological order, the routing state of the network at various time instances throughout the simulation. Each column represents one agent equipped with a sensor, processor and estimator. This configuration is a matter of choice rather than a restriction of the algorithm. The links shown in the figure represent those that carried more than 10 units of data during the simulation. The figure demonstrates the shift of flow from one part of the network to another as the sensor utilities change. More importantly, the routing states shown cannot be determined based merely on intuition. From an inter-agent perspective, the active links for the corresponding time instances are also shown in Figure 6.28. The average number of active inter-agent links at each time instance is less than half the fully-connected network maximum of ten links. Figure 6.29 and Figure 6.30 show the flow rates of sensor observations from processors 1 and 2 to estimators 1 and 2. The flow rates shown correspond to the percentage of observations transmitted.

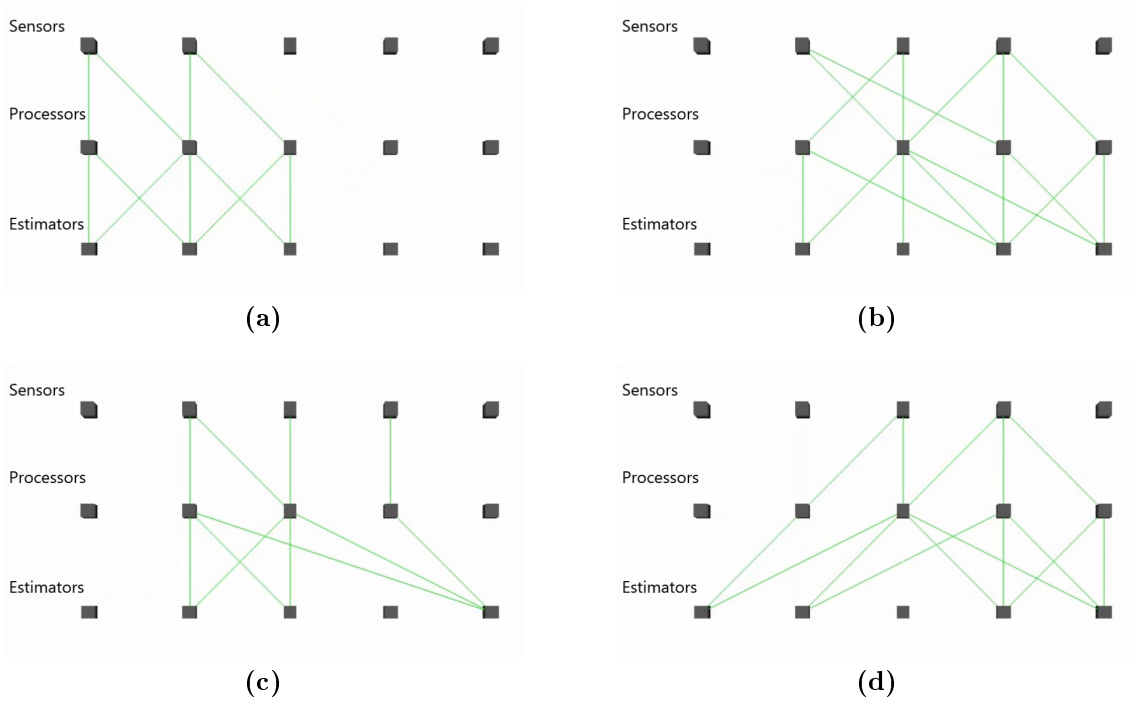


Figure 6.27 – Routing state of multiple-node simulation at various times depicting active wireless links. Each column represents one agent equipped with a sensor, processor and estimator. Active links are represented by green lines. A link is considered active if it holds more than 10 units of data flow.

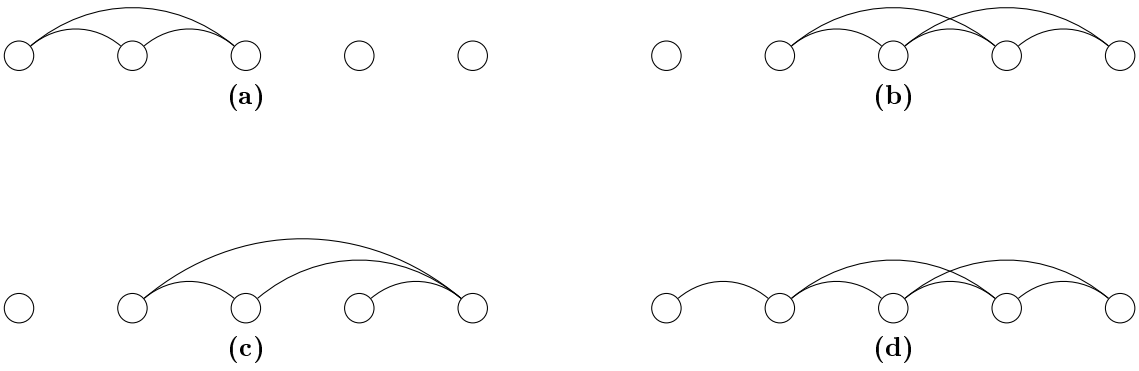


Figure 6.28 – Active inter-agent communication links sharing the same communication medium for the configurations in Figure 6.27. Each agent is composed of a sensor, processor and an estimator. In our simulated environment, the agents are assumed to be spatially distributed in a linear manner as shown in the figure.

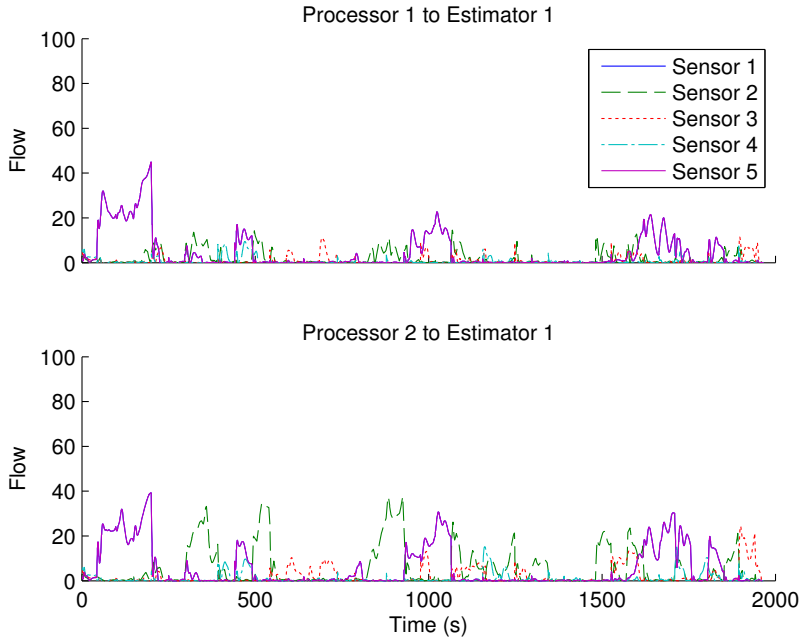


Figure 6.29 – Data flow for links from processors 1 and 2 arriving at estimator 1 for the multiple-node simulation. The flow rates shown correspond to the percentage of observations transmitted

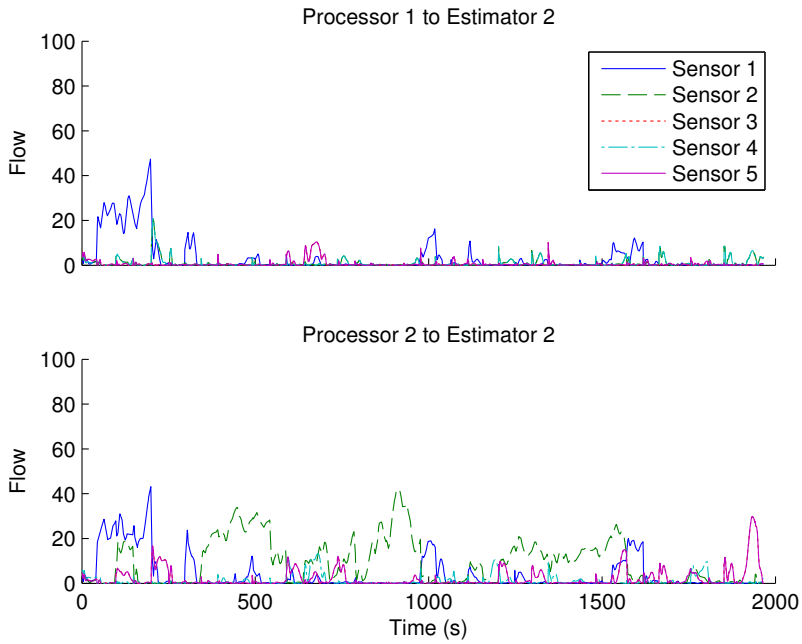


Figure 6.30 – Data flow for links from processors 1 and 2 arriving at estimator 2 for the multiple-node simulation. The flow rates shown correspond to the percentage of observations transmitted

6.4.5 Monte Carlo Simulation

To validate the performance of dynamic information in comparison to down-sampling, we conducted a Monte Carlo simulation for the experimental setting of Section 6.4.1. The aim of the simulation is to analyse the statistical significance of performance improvement due to our solution to threshold-DIF.

The results of the Monte Carlo simulation are shown in Figure 6.31 in box-plot format. For each communication method, we ran twenty randomly initialised trials running for one minute each. The results shown assume each trial as one sample. Dynamic communication outperforms down-sampling with p-value less than 0.03 based on the Welch's t-test.

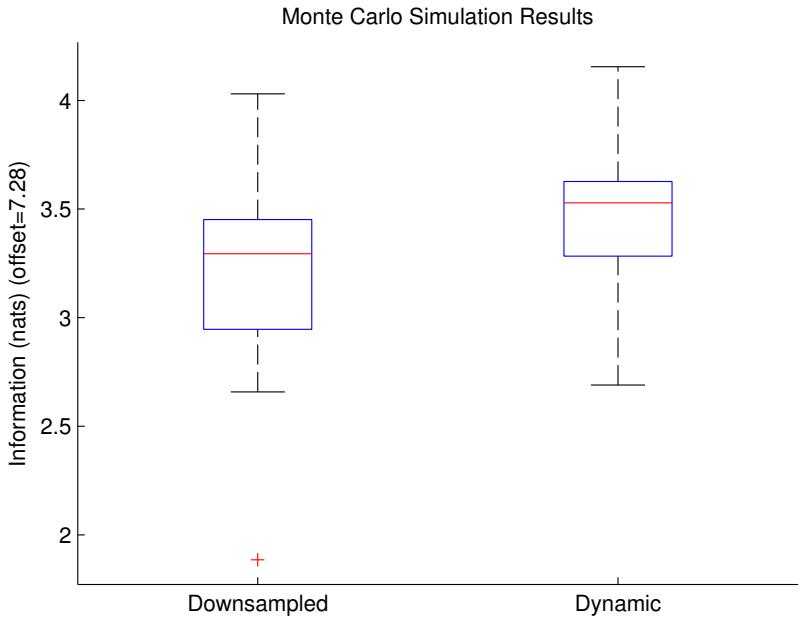


Figure 6.31 – Monte Carlo simulation results comparing down-sampling with dynamic information flow. Each method was tested on the two-robot scenario for twenty trials running for one minute each. In the results depicted, a trial acts as one sample. The box extents represent the first and third quartile, while the whiskers represent the extrema. The median is represented by the horizontal line inside the box. The values correspond to the negative entropy with the addition of an offset value.

6.5 Negotiation-DIF

In this section, we show the results of a complete and integrated solution of communication efficiency in information gathering applied to a simulated real-world example. The solution addresses communication efficiency for both data fusion and decision making. The example is a multiple-node real-world scenario that uses the negotiation-DIF formulation. It is implemented in our decentralised simulation framework. This scenario is representative of many applications in agriculture, surveillance and mining.

The complexity of the scenario precludes the possibility of a manually designed communication strategy. The flow of information throughout the network needs to continuously adjust to changes in the system's state. In addition to data fusion, communication is required for decision making so the robots can coordinate their decisions. Our solution to negotiation-DIF enhances communication efficiency at both the data fusion and decision making layers concurrently.

The scenario has been designed to highlight performance on large systems as well as the flexibility features of negotiation-DIF and its corresponding solution. In addition, it demonstrates interesting behaviour such as switching processing locations, selective cooperation and interjection of images from the stationary camera.

The section also includes a simple two-robot simulation that aims to verify the performance of negotiation-DIF in comparison to other communication methods. The advantage of our communication efficiency solution is also verified against naive downsampling through a Monte Carlo simulation test.

6.5.1 Multiple-Node Simulation

The multiple-node scenario involves two mobile robots, two processing ground stations and one stationary camera. The mission of the robots is to track two moving targets.

The aim of this simulation is to demonstrate the applicability of negotiation-DIF to real-world applications. The experimental scenario of this simulation mimics scaled

versions of applications in agriculture and surveillance. In agriculture, the ability to track and herd cattle using multi-robot systems is desirable. In surveillance applications, using a combination of mobile agents and static sensors for tracking is beneficial for real-time security requirements.

Manual design of a communication strategy is not possible due to the complexity of this scenario caused by several factors. First, there are three types of sensors, a fixed camera and two sensors with different fields of view and each mounted on a mobile robot. These sensors' utility will vary throughout the demonstration. Second, the sensors on board the cameras are bearing-only sensors that require coordination between the robots in order to maximise their joint utility. Another cause of complexity is the multiplicity of routes between sensors and estimators.

Qualitatively, this scenario induces interesting behaviour that corresponds to communication efficiency. For instance, it is expected that the robots will relay raw data to be processed off-board only when they are close to one of the processing stations. Furthermore, it is expected that the robots will only receive image observations from the stationary camera when they are sufficiently close to the camera and when their current estimates have high uncertainty. In addition, the robots are not expected to cooperate when they are distanced from each other and when each robot is tracking a different target.

Experimental Setup

The experimental setting is depicted in Figure 6.32 and is based on the outdoor experimental system. The setting includes two mobile robots, two processing ground stations and one stationary camera. The two robots are equipped with bearing-only sensors, one with a 360° field of view and the other with a 180° field of view. The robots share a large operating region and track two targets circulating the outside border of the region. The robots have access to an off-board camera and two off-board processing stations.

Figure 6.32 – Demonstration setting of the multiple-node simulation. The two mobile robots are shown within their movement region. Targets tracks are shown circulating outside the robot region. The ground stations and camera are also depicted.

The network diagram for this demonstration is shown in Figure 6.33. A common wireless medium is used for all communication between the robots, the camera and processing stations. The link cost increases with increasing communication distance through a crude distance-proportional model. Virtual links, not shown in the figure, allow for the no-send decision.

Results

The flow rates for wireless links to the estimator of Robot 1 are shown in Figure 6.34, while the flow rates for wireless links to the estimator of Robot 2 are shown in Figure 6.35. The flow rate correspond to the percentage of observations transmitted. For

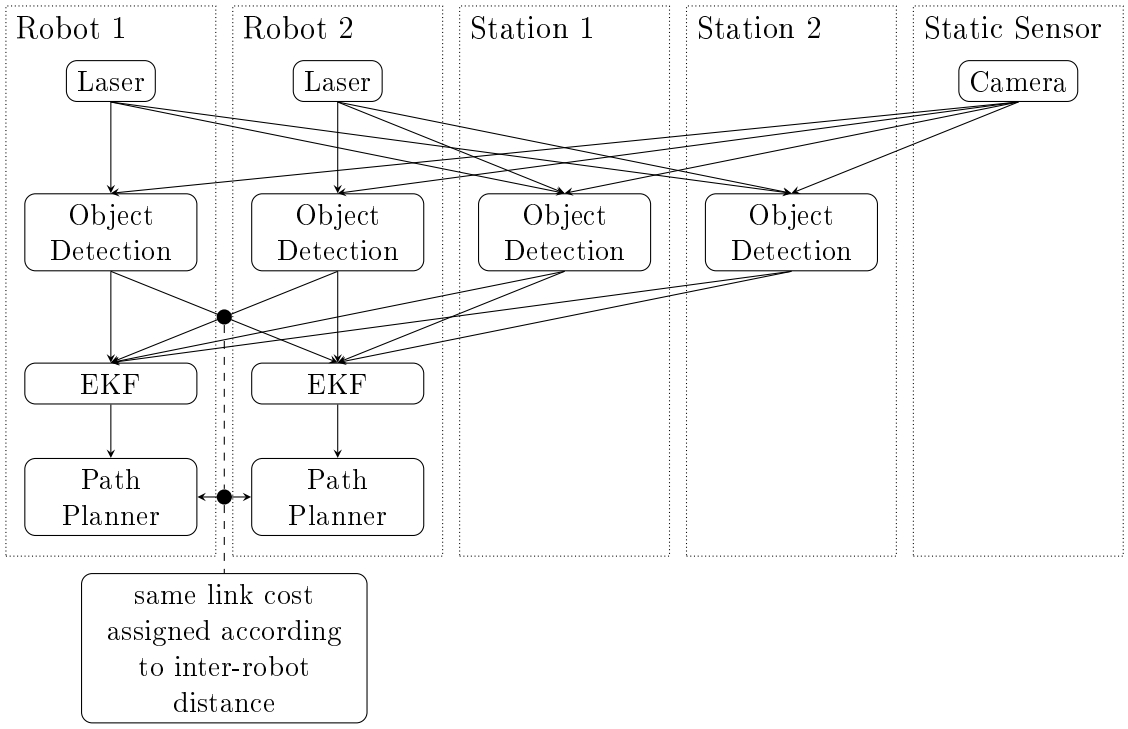


Figure 6.33 – The negotiation-DIF diagram for the multiple-node simulation. Virtual links are omitted for clarity.

a visually-oriented display of the results, snapshots of the system state with active wireless links are shown in Figure 6.36. In Figure 6.36a, both robots are close to the camera. Therefore, they both receive aiding observations from the camera. This is evident from the values at time 50 shown in the flow plots. At time 125, Robot 1 receives observations from the camera while Robot 2 is completely disconnected from other nodes. At time 170, the robots exchange observations. In the same time, Robot 2 is close to Station 2 and therefore, it relays sensor data processing to the stations instead of processing the data on-board. At time 175, Robot 1 receives observations from the camera. However, the camera data is first processed at Station 1 and then relayed to Robot 1. The use of a processing station as a relay is particularly interesting since it was not directly anticipated by the author.

The results of this simulation show that real-time performance is maintained for a network involving eleven nodes. Real-time performance is evident from the time-scale

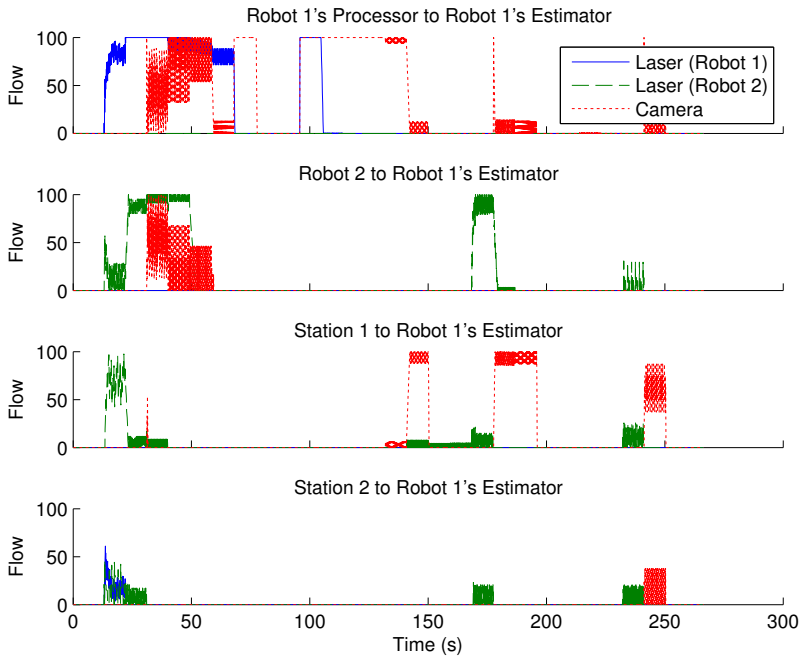


Figure 6.34 – Data flow for links arriving at Robot 1's estimator for the multiple-node simulation. The flow rates shown correspond to the percentage of observations transmitted.

of the flow switching observed both in Figure 6.36 and in the plots of Figure 6.34 and Figure 6.35.

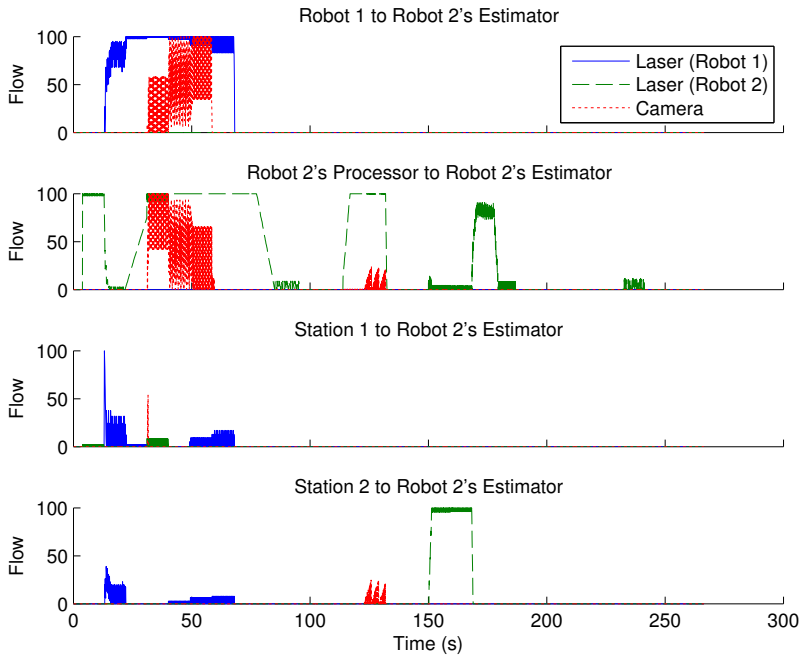
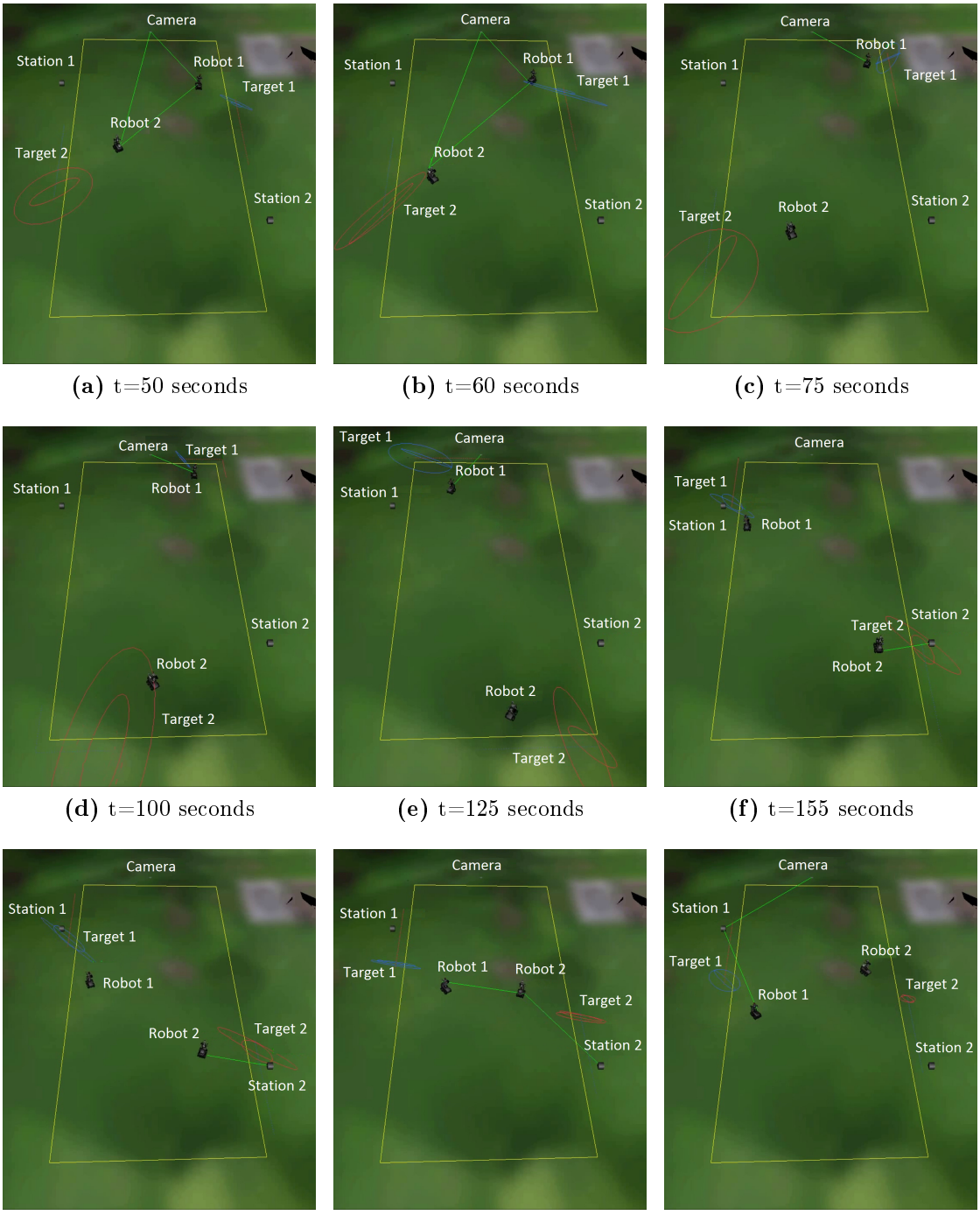


Figure 6.35 – Data flow for links arriving at Robot 2's estimator for the multiple-node simulation. The flow rates shown correspond to the percentage of observations transmitted



(a) t=50 seconds **(b)** t=60 seconds **(c)** t=75 seconds

(d) t=100 seconds **(e)** t=125 seconds **(f)** t=155 seconds

(g) t=160 seconds **(h)** t=170 seconds **(i)** t=175 seconds

Figure 6.36 – System state of multiple-node simulation at various times t. Active wireless links are represented by green lines.

6.5.2 Two-Robot Simulation

To validate the performance of our solution over other communication strategies, we tested and compared different communication strategies in a simulated environment involving two mobile robots cooperatively tracking a moving target.

Experimental Setup

The experiment setting is shown in Figure 6.37 and is based on the outdoor system of Section 6.2. The two robots are equipped with bearing-only sensors, one with a 360° field of view and the other with a 180° field of view. In this case, the two mobile robots are assigned to the same rectangular workspace. This experimental setting also involved two targets that circulated around the robots' workspace. The fact

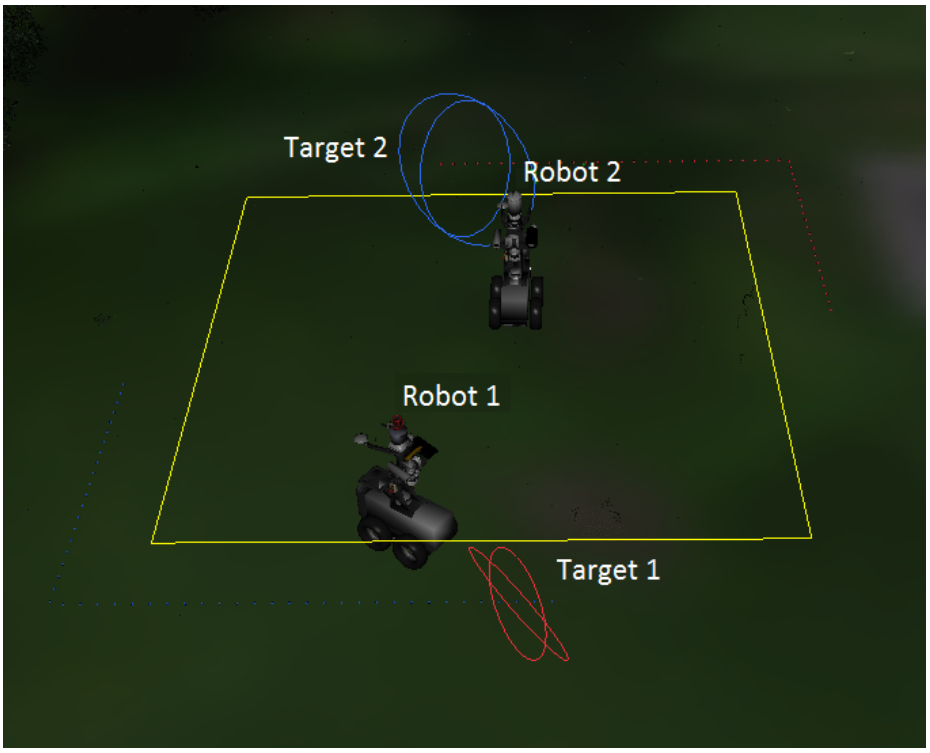


Figure 6.37 – Demonstration setting of the two-robot simulation. Robots 1 and 2 are shown with their boundary. Sample target tracks are shown making a square pattern around the robots' workspace.

that the robots now share the same workspace and are required to track two targets increases the need for cooperative decision making between robots.

The network diagram for this demonstration is shown in Figure 6.38. Data fusion and decision making share a common wireless communication medium. The links representing both processes are assigned the same link cost. Without communication constraints, data fusion requires bandwidth of approximately 2KB/s while cooperative decision making requires approximately 8KB/s. The link cost increases with inter-robot distance. We employ a crude model that sets the link cost proportional to distance. Virtual links, not shown in the figure, allow for the no-send decision.

To validate the advantage of the dynamic information flow formulation, three control tests were run for the purpose of comparison. The first control test allows unconstrained data fusion and negotiation between all nodes. This test is referred to as the *unconstrained* case. In the second control test, communication is reduced to the same rate used by the dynamic information flow case for both data fusion and decision making. This test is referred to as the *down-sampled* case. The third control test allows unconstrained communication for data fusion but involves local decision making only. The purpose of the third control test, referred to as the *no negotiation* case, is to validate the benefit of negotiation.

Results

The information value for the robots' target estimates over time is shown in Figure 6.39 with the corresponding average bars shown in Figure 6.40. The dynamic case achieves an information gathering performance comparable to the unconstrained case while requiring approximately 50 % less communication bandwidth for data fusion and decision making. The percentage of bandwidth saved for decision making is particularly important since decision making consumes more bandwidth in this experiment.

Figure 6.41 shows the inter-robot distance and the chosen data flow and negotiation rates over time as selected by our dynamic flow method. The flow rates shown

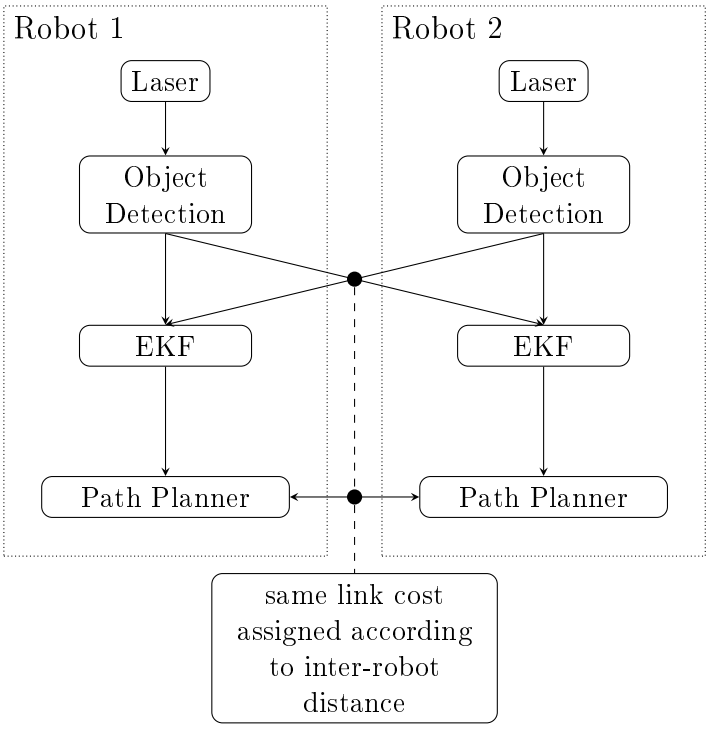
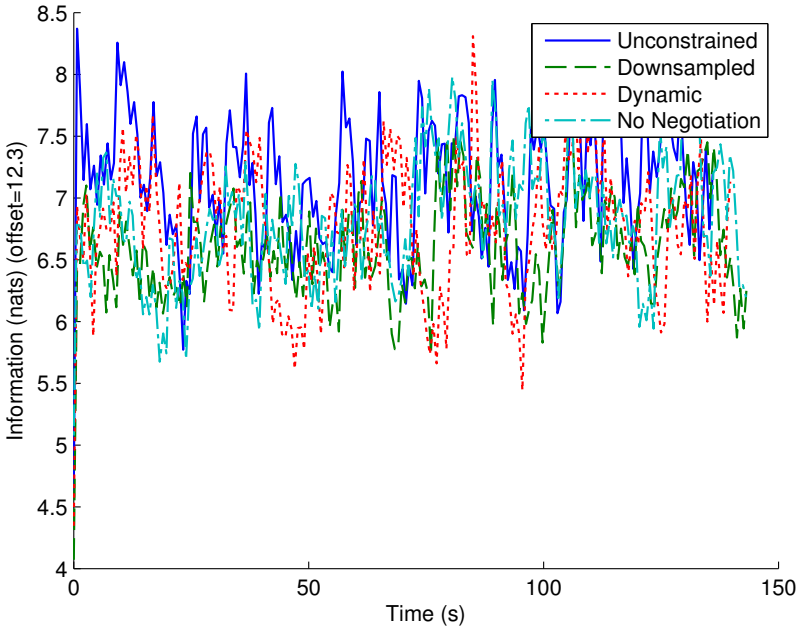
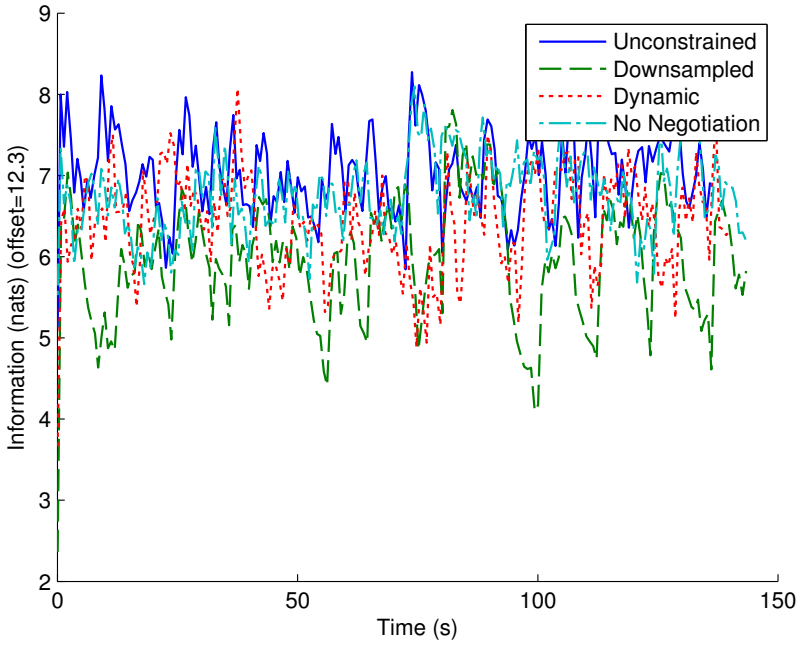


Figure 6.38 – The negotiation-DIF diagram for the two-robot simulation. Virtual links are omitted for clarity.

correspond to the percentage of observations transmitted while the negotiation rate corresponds to the percentage of time steps for which robots cooperatively made decisions. The sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation. The inter-robot distance was used to determine the communication cost with a constant of proportionality of 1. The second and third plots show many instances of complementarity in transmission indicating an intelligent usage of available bandwidth. The last plot shows how bandwidth usage due to negotiation was saved when inter-robot coupling was determined to be of a reduced impact on performance with the threshold set at 0.2.

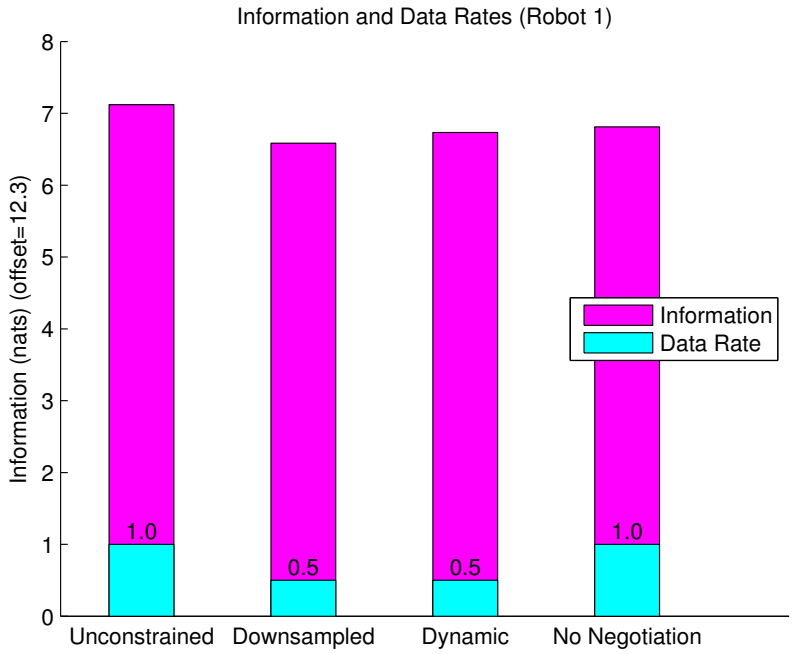


(a) Robot 1

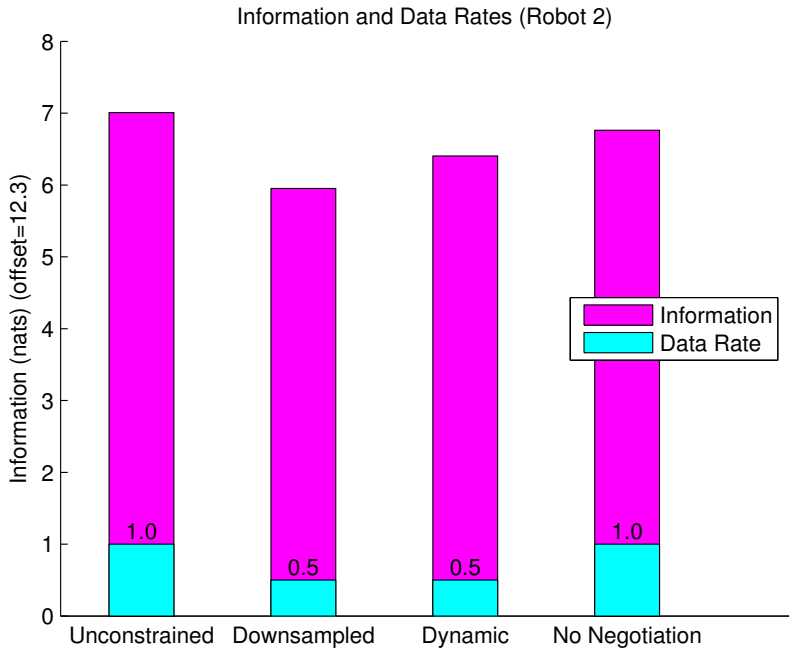


(b) Robot 2

Figure 6.39 – The information value (negative entropy) of the robots' target estimate for the two-robot simulation. Plots shown are for all four communication methods: unconstrained, down-sampled, dynamic and no-negotiation. The values correspond to the negative entropy with the addition of an offset value.



(a) Robot 1



(b) Robot 2

Figure 6.40 – Time averages of the plots of Figure 6.39 shown in bar format. The data rates are shown superimposed. The data rate values are ratios to the unconstrained rate for both data fusion and negotiation. The dynamic case shows a clear improvement in information gain in comparison to down-sampling for Robot 2.

Figure 6.41 – Inter-robot data flow and negotiation rates for the dynamic case of the two-robot simulation. Sensor utility and coupling are also shown in dashed lines. The flow rates shown correspond to the percentage of observations transmitted while the negotiation rate corresponds to the percentage of time steps for which robots cooperatively made decisions. The sensor utilities correspond to the resulting difference in the log determinant of the estimate's covariance matrix after an observation.

6.5.3 Monte Carlo Simulation

We performed a Monte Carlo simulation for the experimental setting of Section 6.5.2, comparing the dynamic communication with down-sampling. The aim of the simulation is to provide statistically significant results that verify the performance of our solution to negotiation-DIF.

The dynamic communication case was compared against two sets of down-sampling rates. The first set of 50% for DDF and 70% for DDM is chosen to be approximately equal to the average usage of 51.2% and 67% of the dynamic case. The second set is chosen to be 60% and 70%. Each method was tested in twenty randomised trials running for two minutes each.

The results of the Monte Carlo simulation are shown in Figure 6.42 in box-plot format. The results shown assume each trial as one sample. Dynamic communication outperforms the down-sampling rate of 50% and 70% with a Welch's t-test for statistical significance resulting in a p-value less than 0.001.

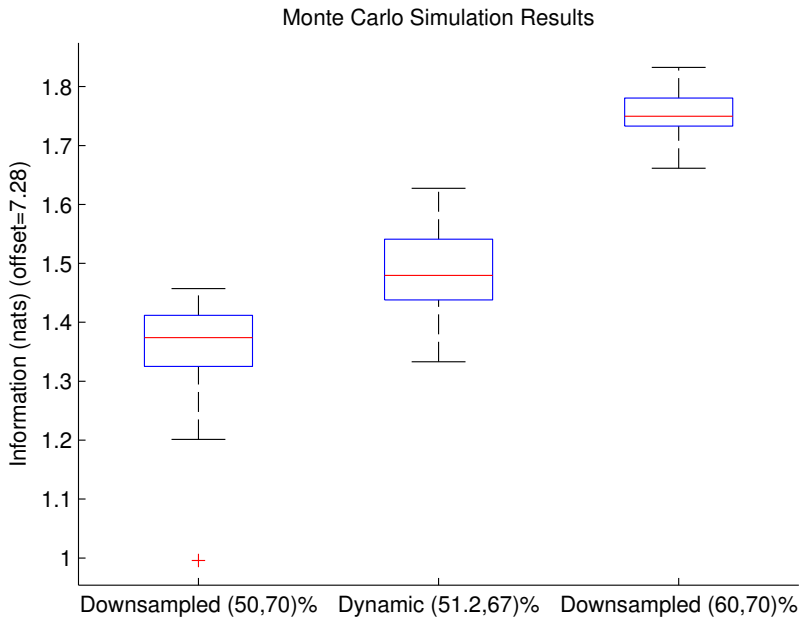


Figure 6.42 – Monte Carlo simulation results comparing down-sampling with dynamic information flow. Each method was tested on the scenario of Section 6.5.2 for twenty trials running for one minute each. In the results depicted, a trial acts as one sample. The box extents represent the first and third quartile, while the whiskers represent the extrema. The data median is represented by the horizontal line inside the box. The values correspond to the negative entropy with the addition of an offset value.

6.6 Discussion and Lessons Learnt

The results shown demonstrate the performance advantage of the solutions in Chapters 4 and 5 and their applicability to heterogeneous information gathering systems. The performance advantage of the solutions over naive down-sampling of data rates was validated experimentally through Monte Carlo simulations. The experimental settings shown in this chapter involved heterogeneous systems with nodes possessing different sensor types and levels of computation. The range of sensors included cameras, 2D lasers with a 180° field of view and a 3D laser with a 360° field of view. The computational capabilities of nodes in the experiments ranged from nil for the case of the fixed camera to eight-core computers.

The real-time performance of the algorithms was demonstrated for networks up to eleven nodes as shown in the multiple-node simulation of Section 6.5.1. Nevertheless, the scalability of the algorithms is not without limit since the complexity remains superlinear. In practice, we have found that networks up to fifteen nodes, such as that of Section 6.4.4, maintain real-time performance.

The myopic approximation of sensor utility did not cause critical reduction in performance as testified by the advantage of our algorithms over down-sampling. However, an observable effect was the lag in any performance boost caused by the algorithms. This effect was observed in the results of Section 6.3.1. We hope to address this issue in future work.

The hardware experiments confirmed our claim that any algorithm introduced to deal with communication efficiency needs to be itself communication-efficient. During the experiments, we realised a slight delay in the performance of our algorithms over loaded wireless networks. For example, the maximum time between updates for the simulation in Section 6.4.3 was approximately 0.14 seconds while this time was approximately 0.31 seconds for the corresponding hardware experiment. This observation supports the hypothesis that other methods with extra communication and computation overhead will fail to produce the required communication efficiency.

From an implementation perspective, several remarks about the algorithms need to be noted. First, the data fusion framework needs to be able to deal with intermittent reception of observations in addition to the reception of observations from several different sources. Although the solution to threshold-DIF obeys the specified resource limits, the algorithm is not an any-time algorithm and thus approximations are required until the algorithm has converged.

Chapter 7

Conclusions and Future Work

This thesis has presented algorithms that improve communication efficiency in decentralised information gathering in a global and principled manner. Improved communication efficiency, in turn, eases current limits on the size, heterogeneity, applicability and versatility of decentralised information systems. This chapter provides a summary of the thesis, a summary of contributions and possible future research directions.

7.1 Thesis Summary

This thesis introduced a comprehensive and principled approach to improving the efficiency of communication required by the data fusion and decision making layers in decentralised information gathering. The approach is based on the novel dynamic information flow problem formulation. We introduced three variants of the DIF problem that model different types of communication resource limitations. The problem formulations of the three variants were presented in Chapter 3.

The first variant, min-cost-DIF, permits assigning communication costs for links used at the data fusion layer. In Section 4.1, we proposed a solution to min-cost-DIF based on optimal multicast routing algorithms. We proved equivalence of our problem formulation to the standard multicast routing problem. The proposed solution allows

the system to decide on-line when, to whom and at what level of abstraction should information flow between robots while taking into account the multicast nature of data sharing. In Section 6.3, we presented simulation and experimental results of decentralised information gathering systems involving mobile robots tracking a moving target. In one experiment, a robot was shown to switch from off-board to on-board processing as its distance from the station increased. In another, robots selectively processed raw sensor data and shared observations whenever the combined sensor utility computed by both robots justified the processing cost. The resulting information gathering performance surpassed that of simple down-sampling methods. This performance advantage was further validated through the results of a Monte Carlo simulation.

The second variant, threshold-DIF, improves on min-cost-DIF by allowing explicit bandwidth constraints to be assigned to communication links. Thus, threshold-DIF avoids the difficulty required in assigning scaling factors between communication, computation and sensor utility. In Section 4.3, we proposed a solution to threshold-DIF based on a distributed version of ADMM and provided a detailed complexity analysis in terms of the problem size. In Section 6.4, we presented experimental results for two scenarios. The first involved two mobile robots tracking a moving target. The inter-robot communication bandwidth was assumed to be limited and inadequate for simultaneous two-way communication. The communication direction alternated during the experiment according to sensor utility. The second scenario involved the two robots tracking a moving target with the aid of a stationary camera. When the camera's observation utility was valuable, inter-robot communication halted so the camera could send raw images to one of the robots for processing. In both scenarios, our solution outperformed simple down-sampling methods using the same amount of communication bandwidth. For the first scenario, this performance advantage was further validated through a Monte Carlo simulation. We also presented a simulation of threshold-DIF implemented on a network with fifteen nodes to demonstrate the performance of our solution on networks for which manual design is challenging.

The third variant, negotiation-DIF, extends min-cost-DIF to allow for the simultaneous improvement of communication efficiency at the data fusion and decision making layers. In negotiation-DIF, the data fusion and decision making layers share a common communication link cost. The proposed solution to negotiation-DIF was presented in Chapter 5. The solution is based on the solution to min-cost-DIF combined with the extended-LQISO algorithm proposed in Section 5.3. The output of extended-LQISO provides the min-cost-DIF layer with sensor utility scaling factors that relate feedback control actions to sensor observations. These scaling factors are also used to enhance sensor utility estimation. Negotiation-DIF is beneficial for systems that communicate large amounts of data during decision making negotiations. In Section 6.5, we verified our solution to negotiation-DIF in simulation for a two mobile robot scenario. The results of the simulation showed the robots concurrently deciding on communication at the data fusion and decision making layers. The robots simultaneously decided whether to share observations and whether to negotiate or take decisions independently instead. Monte Carlo simulation results demonstrated a statistically significant improvement caused by negotiation-DIF in comparison with simple subsampling. We also presented simulation results for a complex decentralised information gathering system that mimics real applications. The system involved one camera, two processing ground stations and two mobile robots actively tracking two moving targets.

7.2 Summary of Contributions

This section provides a discussion of the contributions towards communication efficiency in information gathering that were introduced by this thesis. The discussion is divided into individual sections for each contribution.

7.2.1 DIF Formulation

This thesis introduced the novel DIF formulation of communication efficiency in information gathering. The benefit of the DIF formulation is that it is general since it is applicable to any system that is composed of sensors, processors, estimators and controllers. It is also general because, unlike existing methods, it is not specific to a particular application. The DIF formulation permits decentralised, efficient and practically implementable algorithms due to its graph structure representation and its capacity to include local and global costs, utilities and constraints.

Three problem variants of dynamic information flow were defined. The first two target applications where communication is dominated by data fusion while the third targets applications where communication is dominated by both data fusion and negotiation. These options are useful in practice because most information gathering systems involve data fusion and some also require high bandwidth for decision making.

7.2.2 Min-Cost-DIF Solution

This thesis introduced an efficient decentralised solution to min-cost-DIF. The solution was adapted from recent results in multicast routing, which we extended to allow for negative link costs that represent sensor utility. The benefits of the solution to min-cost-DIF are mainly applicable to heterogeneous systems with large amounts of sensor data. Min-cost-DIF is targeted to applications where communication limits are more suitably represented as link costs. This is the case in applications where link costs are dynamic or difficult to specify due to links being shared with other algorithms.

7.2.3 Threshold-DIF Solution

This thesis introduced an efficient decentralised solution to threshold-DIF. The solution is based on a distributed version of ADMM that requires neighbour-to-neighbour

communication only. Similar to the min-cost-DIF case, the solution is of importance to heterogeneous systems with large amounts of data. However, threshold-DIF is targeted to applications where communication limits are known and need to be both globally and explicitly specified, such as standard mesh networks or WiFi networks.

7.2.4 Sensor Utility

This thesis provided a concise empirical analysis of sensor utility approximations. For a simple multi-robot mapping application, we showed that the myopic approximation is much closer to the exact sensor utility than a conservative theoretical upper bound.

7.2.5 LQISO

This thesis introduced the LQISO algorithm. LQISO is based on the LMI formulation of the LQ team problem that allows the addition of communication costs onto the optimal control problem. The result is a convex optimisation solvable in polynomial time. The thesis also demonstrated how this can be extended to reduce communication required by a DDM algorithm for information gathering while simultaneously maintaining acceptable team performance.

7.2.6 Negotiation-DIF Solution

This thesis introduced a solution to negotiation-DIF. The solution was obtained by combining our solution to min-cost-DIF with extended-LQISO. The result of our negotiation-DIF solution is a complete solution to communication efficiency in information gathering. The solution to negotiation-DIF extends the benefits of dynamic communication to information gathering applications with high communication demand for both data fusion and decision making.

7.3 Future Work

Many future research directions exist for both communication-efficient data fusion independently as well as for communication-efficient information gathering. These future directions either aim to theoretically extend or address shortcomings of the suggested algorithms or improve the algorithms towards a simpler implementation with the goal of wide usage within the domain of decentralised information gathering.

7.3.1 Communication Efficiency in Data Fusion

For communication efficiency in data fusion, we project the following future directions:

- **Non-myopic sensor utility:** An important future advancement of our solutions to DIF is an accurate non-myopic sensor utility estimate. A possible direction might include machine learning techniques that learn the sensor utility as a function of the platform state. The main challenge for this direction is computation and communication efficiency of the learning algorithms. A non-myopic estimate will reduce rapid variations in sensor utility.
- **Any-time feasible solution:** A future goal is to transform our current distributed solution to threshold-DIF into an any-time feasible solution. This will greatly simplify implementation avoiding any need for approximations. Our prediction is that any-time feasibility might be achieved through feasibility projection methods from optimisation theory.
- **Learn communication policies:** Communication patterns observed during conducted DIF experiments suggest that machine learning techniques can be employed to learn communication policies directly. Machine learning techniques add an extra layer of intelligence and can further improve communication efficiency particularly over larger time horizons.
- **Plug-and-play capability:** A future ambition is to provide our algorithms with a plug-and-play capability. In our current implementation, the identities

of the nodes in the network need to be globally known. We envisage that this is not a necessary requirement and can be replaced by an automated discovery process. Nodes can advertise their types and then run a handshaking procedure with appropriate nodes before establishing connections.

- **Select sensor operational modes:** Another future direction worth investigating is the adaptation of DIF to the case of selecting operational modes of a sensor. By modelling each mode as a separate sensor, the DIF formulation could potentially lead to decisions on which mode should operate for different time periods.

7.3.2 Communication Efficiency in Information Gathering

For the case of communication efficiency in information gathering, we project the following future directions:

- **Assign explicit resource constraints:** A desirable future research outcome is the ability to assign explicit resource constraints instead of link costs as is currently required by negotiation-DIF. By assigning explicit constraints, a system designer can confirm that the limit on communication will not be violated and hence the system will not fail. Since negotiation-DIF is an amalgamation of min-cost-DIF and comms-LQ, we aim to fuse threshold-DIF with comms-LQ to allow for explicit communication constraints.

- **Decentralise LQISO:** The LQISO algorithm introduced in Section 5.2 is amenable to several future improvements. One desirable improvement is decentralising the algorithm. Decentralising the algorithm may expand its applicability to large teams of small unmanned aerial vehicles (UAVs) performing coordination tasks in applications such as aerial surveillance or construction.

- **Reduce the complexity of LQISO:** Another possible improvement to LQISO is the reduction in the algorithm's time complexity. The time complexity of the current version of the algorithm is determined by the complexity of SDP.

- **Add stability criteria to LQISO:** We desire to add stability criteria to LQISO. This means that LQISO can then be used directly for distributed LQ control. This can have many implications for industries that rely on distributed control such as the chemical engineering industry.

- **Use multi-radio multi-channel networks:** Finally, we hypothesise that employing recent advances in multi-radio multi-channel networks [57] in the implementation of our algorithms will help further exploit the benefits of these algorithms.

7.3.3 Applications

Since the solution methods in this thesis are applicable to general information gathering applications, in the future, we predict that our algorithms will be of benefit for applications such in agriculture, cooperative mapping, urban surveillance, chemical plants and mining:

- **Agriculture:** In agriculture, automated herding of cattle over large areas through mobile robots requires cooperative tracking and coordination by the robots. The large area to be covered and the necessity of using bandwidth-limited wireless communication naturally lead to the DIF formulation of the problem. Hence, we predict that such systems can be enabled through the solutions we have provided or enhancements thereof.

- **Mapping:** A large team of mobile robots exploring a wide area does not need to continuously share the entire discovered portion of the map. We hypothesise that our algorithms can provide an automated and dynamic communication-efficient solution.

- **Urban surveillance:** A traditional problem in urban surveillance is the over-abundance of information available. Once more, we predict that our algorithms can provide a solution for this problem.

- **Tracking personnel and equipment:** We believe that the DIF problem formulation applies to the problem of tracking personnel and equipment in industries with geographically large areas such as ports and mines. The adaptation of DIF into such commercial applications is definitely a future ambition we desire to achieve.

Bibliography

[1] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, 2000.

[2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice hall, 1993.

[3] J. Al-Karaki and A. Kamal. Routing techniques in wireless sensor networks: a survey. *IEEE Wireless Communications*, 11(6):6–28, 2004.

[4] N. Atanasov, J. L. Ny, K. Daniilidis, and G. J. Pappas. Information acquisition with sensing robots: Algorithms and error bounds. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[5] A. Bagula and K. Mazandu. Energy constrained multipath routing in wireless sensor networks. In F. Sandnes, Y. Zhang, C. Rong, L. T. Yang, and J. Ma, editors, *Ubiquitous Intelligence and Computing*, volume 5061 of *Lecture Notes in Computer Science*, pages 453–467. Springer Berlin Heidelberg, 2008.

[6] L. Bakule and M. Papík. Decentralized control and communication. *Annual Reviews in Control*, 36(1):1 – 10, 2012.

[7] H. Bayram and H. Bozma. Multirobot communication network topology via centralized pairwise games. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[8] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein. The complexity of decentralized control of markov decision processes. *Journal Mathematics of Operations Research*, 27(4):819–840, 2002.

[9] F. Bourgault, T. Furukawa, and H. Durrant-Whyte. Decentralized bayesian negotiation for cooperative search. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004.

[10] S. Boyd, N. Parikh, and E. Chu. *Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers*. Now Publishers Incorporated, 2011.

[11] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[12] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System & Control Theory*, volume 15. Society for Industrial and Applied Mathematics (SIAM), 1994.

[13] G. M. Brooker, J. A. Randle, M. Attia, Z. Xu, T. Abuhashim, A. Kassir, J. Chung, S. Sukkarieh, N. Tahir, and J. Dickens. First airborne trial of a UAV based optical locust tracker. In *Proceedings of the ARAA Australasian Conference on Robotics and Automation (ACRA)*, 2011.

[14] A. Carlin and S. Zilberstein. Value-based observation compression for DEC-POMDPs. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, May 2008.

[15] J. Chen, K. H. Low, C. K.-Y. Tan, A. Oran, P. Jaillet, J. Dolan, and G. Sukhatme. Decentralized data fusion and active sensing with mobile sensors for modeling and predicting spatiotemporal traffic phenomena. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2012.

[16] L. Chen, Z. Wang, B. Szymanski, J. W. Branch, D. Verma, R. Damarla, and J. Ibbotson. Dynamic service execution in sensor networks. *The Computer Journal*, 53(5):513–527, 2010.

[17] T. Chung, V. Gupta, J. Burdick, and R. Murray. On a decentralized active sensing strategy using mobile sensor platforms in a network. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 2004.

[18] T. H. Chung, J. W. Burdick, and R. M. Murray. A decentralized motion coordination strategy for dynamic target tracking. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2006.

[19] Y. Cui, Y. Xue, and K. Nahrstedt. Optimal distributed multicast routing using network coding. In *Proceedings of the IEEE International Conference on Communications (ICC)*, 2007.

[20] F. Delle Fave, A. Rogers, Z. Xu, S. Sukkarieh, and N. R. Jennings. Deploying the max-sum algorithm for coordination and task allocation of unmanned aerial vehicles for live aerial imagery collection. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[21] B. R. Donald. On information invariants in robotics. *Artificial Intelligence*, 72 (1–2):217 – 304, 1995.

[22] B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, and A. Frenkel. On the segmentation of 3D LIDAR point clouds. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[23] H. F. Durrant-Whyte, G. Dissanayake, and P. W. Gibbens. Toward deployment of large–scale simultaneous localisation and map building (SLAM) systems. In J. Hollerbach and D. Koditschek, editors, *Robotics Research*, Springer Tracts in Advanced Robotics, pages 161–168. Springer-Verlag, 2000.

[24] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 1999.

[25] F. M. D. Fave, Z. Xu, A. Rogers, and N. R. Jennings. Decentralised coordination of unmanned aerial vehicles for target search using the max-sum algorithm. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS), Workshop on Agents in Real Time and Environment*, May 2010.

[26] J. Fink, A. Ribeiro, and V. Kumar. Motion planning for robust wireless networking. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[27] R. Fitch and R. Lal. Experiments with a ZigBee wireless communication system for self-reconfiguring modular robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2009.

[28] S. K. Gan and S. Sukkarieh. Multi-UAV target search using explicit decentralized gradient-based negotiation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[29] S. K. Gan, R. Fitch, and S. Sukkarieh. Real-time decentralized search with inter-agent collision avoidance. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[30] S. K. Gan, R. Fitch, and S. Sukkarieh. Online decentralized information gathering with spatial–temporal constraints. *Autonomous Robots*, 37(1):1–25, 2014.

[31] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

[32] A. Gattami. Distributed stochastic control: A team theoretic approach. In *Proceedings of the International Symposium on Mathematical Theory of Networks and Systems*, 2006.

[33] T. Georgiou and A. Lindquist. The separation principle in stochastic control, redux. *IEEE Transactions on Automatic Control*, 58(10):2481–2494, 2013.

[34] A. Ghaffarkhah and Y. Mostofi. Communication-aware motion planning in mobile networks. *IEEE Transactions on Automatic Control*, 56(10): 2478–2485, 2011.

[35] P. J. Gmytrasiewicz and E. H. Durfee. Rational communication in multi-agent environments. *Autonomous Agents and Multi-Agent Systems*, 4(3):233–272, 2001.

[36] J. Goerner, C. N., and K. Sycara. Energy efficient data collection with mobile robots in heterogeneous sensor networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[37] C. V. Goldman and S. Zilberstein. Optimizing information exchange in cooperative multi-agent systems. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, July 2003.

[38] D. Golovin and A. Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.

[39] D. Golovin, M. Faulkner, and A. Krause. Online distributed sensor selection. In *Proceedings of the ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN)*, 2010.

[40] S. Grime and H. Durrant-Whyte. Data fusion in decentralized sensor networks. *Control Engineering Practice*, 2(5):849 – 863, 1994.

[41] B. Grocholsky. *Information-Theoretic Control of Multiple Sensor Platforms*. PhD thesis, The University of Sydney, 2002.

[42] P. Gupta and P. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, 2000.

[43] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray. On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42(2):251–260, 2006.

[44] B. He and X. Yuan. On the $O(1/n)$ convergence rate of the Douglas–Rachford alternating direction method. *SIAM Journal on Numerical Analysis*, 50(2): 700–709, 2012.

[45] Y.-C. Ho. Team decision theory and information structures. *Proceedings of the IEEE*, 68(6):644 – 654, 1980.

[46] G. Hollinger and S. Singh. Multirobot coordination with periodic connectivity: Theory and experiments. *IEEE Transactions on Robotics*, 28(4): 967–973, 2012.

[47] G. Hollinger, S. Singh, J. Djugash, and A. Kehagias. Efficient multi-robot search for a moving target. *The International Journal of Robotics Research*, 28(2):201–219, 2009.

[48] M. A. Hsieh, A. Cowley, V. Kumar, and C. J. Taylor. Maintaining network connectivity and performance in robot teams. *Journal of Field Robotics*, 25 (1-2):111–131, 2008.

[49] G. Inalhan, D. Stipanovic, and C. Tomlin. Decentralized optimization, with application to multiple aircraft coordination. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 2002.

[50] A. Jadbabaie, J. Lin, and A. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988 – 1001, 2003.

[51] A. Kassir, R. Fitch, and S. Sukkarieh. Decentralised information gathering with communication costs. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[52] A. Kassir, R. Fitch, and S. Sukkarieh. Communication-aware information gathering with dynamic information flow (in press). *The International Journal of Robotics Research*, 0(0):–, 2014.

[53] A. Keshavarz-Haddadt and R. Riedi. Bounds on the benefit of network coding: Throughput and energy saving in wireless networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, 2008.

[54] E. Klavins. Communication complexity of multi-robot systems. In J.-D. Boissonnat, J. Burdick, K. Goldberg, and S. Hutchinson, editors, *Algorithmic Foundations of Robotics V*, volume 7 of *Springer Tracts in Advanced Robotics*, pages 275–292. Springer Berlin Heidelberg, 2004.

[55] A. K. Kulatunga, B. T. Skinner, D. K. Liu, and H. T. Nguyen. Distributed simultaneous task allocation and motion coordination of autonomous vehicles using a parallel computing cluster. In T.-J. Tarn, S.-B. Chen, and C. Zhou, editors, *Robotic Welding, Intelligence and Automation*, volume 362 of *Lecture Notes in Control and Information Sciences*, pages 409–420. Springer-Verlag, 2007.

[56] J. Kulik, W. Heinzelman, and H. Balakrishnan. Negotiation-based protocols for disseminating information in wireless sensor networks. *Wireless Networks*, 8(2/3):169–185, 2002.

[57] V. Kuo and R. Fitch. A multi-radio architecture for neighbor-to-neighbor communication in modular robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[58] M. Lindhé and K. H. Johansson. Exploiting multipath fading with a mobile robot. *The International Journal of Robotics Research*, 32(12):1363–1380, 2013.

[59] J. Liu, D. Goeckel, and D. Towsley. Bounds on the gain of network coding and broadcasting in wireless networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, 2007.

[60] L. Liu and D. A. Shell. A distributable and computation-flexible assignment algorithm: From local task swapping to global optimality. In *Proceedings of Robotics: Science and Systems (RSS)*, 2012.

[61] L. Liu and D. A. Shell. Physically routing robots in a multi-robot network: Flexibility through a three-dimensional matching graph. *The International Journal of Robotics Research*, 32(12):1475–1494, 2013.

[62] L. Lu, L. Xie, and M. Fu. Optimal control of networked systems with limited communication: a combined heuristic and convex optimization approach. In *Proceedings of th IEEE Conference on Decision and Control (CDC)*, 2003.

[63] M. Malmirchegini and Y. Mostofi. On the spatial predictability of communication channels. *IEEE Transactions on Wireless Communications*, 11 (3):964–978, 2012.

[64] J. Marschak and R. Radner. *Economic Theory of Teams*. Yale University Press, 1972.

[65] T. Mather and M. Hsieh. Ensemble synthesis of distributed control and communication strategies. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[66] G. M. Mathews. *Asynchronous Decision Making for Decentralised Autonomous Systems*. PhD thesis, The University of Sydney, 2008.

[67] A. Matveev and A. Savkin. Multirate stabilization of linear multiple sensor systems via limited capacity communication channels. *SIAM Journal on Control and Optimization*, 44(2):584–617, 2005.

[68] P. J. Modi. *Distributed constraint optimization for multiagent systems.* PhD thesis, University of Southern California, 2003.

[69] A. Molin and S. Hirche. On LQG joint optimal scheduling and control under communication constraints. In *Proceedings of the IEEE Conference on Decision and Control held jointly with the Chinese Control Conference*, 2009.

[70] P. Morton, B. Douillard, and J. Underwood. An evaluation of dynamic object tracking with 3D LIDAR. In *Proceedings of the ARAA Australasian Conference on Robotics and Automation (ACRA)*, 2011.

[71] Y. Mostofi. Decentralized communication-aware motion planning in mobile networks: An information-gain approach. *Journal of Intelligent and Robotic Systems*, 56(1-2):233–256, 2009.

[72] J. Nash. Non-cooperative games. *The Annals of Mathematics*, 54(2):286–295, 1951.

[73] E. D. Nerurkar and S. I. Roumeliotis. A communication-bandwidth-aware hybrid estimation framework for multi-robot cooperative localization. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[74] E. Nettleton. *Decentralised Architectures for Tracking and Navigation with Multiple Flight Vehicles.* PhD thesis, The University of Sydney, 2003.

[75] A. D. Nguyen, V. T. Ngo, Q. P. Ha, and G. Dissanayake. Robotic formation: initialisation, trajectory planning and decentralised control. *International Journal of Automation and Control*, 2(1):22–45, 2008.

[76] K. Ohkawa, T. Shibata, and K. Tanie. Method for generating of global cooperation based on local communication. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1998.

[77] M. Otte and N. Correll. Any-Com multi-robot path-planning with dynamic teams: Multi-robot coordination under communication constraints. In *Proceedings of the International Symposium on Experimental Robotics (ISER)*, 2010.

[78] T. Peynot and A. Kassir. Laser-camera data discrepancies and reliable perception in outdoor robotics. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[79] R. Radner. Team decision problems. *The Annals of Mathematical Statistics*, 33(3):857–881, 1962.

[80] S. Ramanathan. Multicast tree generation in networks with asymmetric links. *IEEE/ACM Transactions on Networking*, 4(4):558–568, 1996.

[81] M. Rami and X. Y. Zhou. Linear matrix inequalities, Riccati equations, and indefinite stochastic linear quadratic controls. *IEEE Transactions on Automatic Control*, 45(6):1131 –1143, 2000.

[82] I. Rekleitis, V. Lee-Shue, A. P. New, and H. Choset. Limited communication, multi-robot team based coverage. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2004.

[83] A. Ribeiro and G. B. Giannakis. Bandwidth-constrained distributed estimation for wireless sesnor networks-part II: Unkown probability desity function. *IEEE Transactions on Signal Processing*, 54(7):2784–2796, 2006.

[84] M. Rotkowitz and S. Lall. A characterization of convex problems in decentralized control. *IEEE Transactions on Automatic Control*, 51(2): 274–286, 2006.

[85] B. M. Sadler, D. Rus, and G. S. Sukhatme. *The International Journal of Robotics Research, Special Issue on Robotic Communications and Collaboration in Complex Environments*, 32(12), 2013.

[86] C. Scherer, P. Gahinet, and M. Chilali. Multiobjective output-feedback control via LMI optimization. *IEEE Transactions on Automatic Control*, 42 (7):896–911, 1997.

[87] C. Schurgers and M. Srivastava. Energy efficient routing in wireless sensor networks. In *Proceedings of the IEEE Military Communications Conference*, 2001.

[88] M. Schwager, N. Michael, V. Kumar, and D. Rus. Time scales and stability in networked multi-robot systems. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[89] E. Semsar-Kazerooni and K. Khorasani. Optimal consensus algorithms for cooperative team of agents subject to partial information. *Automatica*, 44(11): 2766 – 2777, 2008.

[90] E. Semsar-Kazerooni and K. Khorasani. Multi-agent team cooperation: A game theory approach. *Automatica*, 45(10):2205 – 2213, 2009.

[91] P. Shah and P. Parrilo. H2-optimal decentralized control over posets: A state space solution for state-feedback. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 2010.

[92] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry. Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, 49(9):1453–1464, 2004.

[93] R. N. Smith, M. Schwager, S. L. Smith, B. H. Jones, D. Rus, and G. S. Sukhatme. Persistent ocean monitoring with underwater gliders: Adapting sampling resolution. *Journal of Field Robotics*, 28(5):714–741, 2011.

[94] J. Speyer, I. Seok, and A. Michelin. Decentralized control based on the value of information in large vehicle arrays. In *Proceedings of the American Control Conference (ACC)*, 2008.

[95] M. Stachura and E. W. Frew. Cooperative target localization with a communication-aware unmanned aircraft system. *AIAA Journal of Guidance, Control, and Dynamics*, 34(5):1352–1362, 2011.

[96] F. Tang and L. E. Parker. ASyMTRe: Automated synthesis of multi-robot task solutions through software reconfiguration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

[97] J. N. Twigg, J. R. Fink, P. L. Yu, and B. M. Sadler. Efficient base station connectivity area discovery. *The International Journal of Robotics Research*, 32(12):1398–1410, 2013.

[98] J. Willems. Least squares stationary optimal control and the algebraic Riccati equation. *IEEE Transactions on Automatic Control*, 16(6):621 – 634, 1971.

[99] S. Williamson, E. Gerding, and N. Jennings. A principled information valuation for communications during multi-agent coordination. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS), Workshop on Multi-Agent Sequential Decision Making in Uncertain Domains*, 2008.

[100] S. A. Williamson, E. H. Gerding, and N. R. Jennings. Reward shaping for valuing communications during multi-agent coordination. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2009.

[101] H. S. Witsenhausen. Separation of estimation and control for discrete time systems. *Proceedings of the IEEE*, 59(11):1557–1566, 1971.

[102] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu. A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks. In *Proceedings of the International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN)*, 2000.

[103] Y. Xi and E. Yeh. Distributed algorithms for minimum cost multicast with network coding. *IEEE/ACM Transactions on Networking*, 18(2):379 –392, 2010.

[104] K. Xing, X. Cheng, L. Ma, and Q. Liang. Superimposed code based channel assignment in multi-radio multi-channel wireless mesh networks. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2007.

[105] Z. Xu, R. Fitch, and S. Sukkarieh. Decentralised coordination of mobile robots for target tracking with learnt utility models. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[106] Z. Xu, R. Fitch, J. Underwood, and S. Sukkarieh. Decentralized coordinated tracking with mixed discrete–continuous decisions. *Journal of Field Robotics*, 30(5):717–740, 2013.

[107] E. Yoshida, M. Yamamoto, T. Arai, J. Ota, and D. Kurabayashi. A design method of local communication area in multiple mobile robot system. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1995.

[108] Y. Zhang, C. Lee, D. Niyato, and P. Wang. Auction approaches for resource allocation in wireless systems: A survey. *IEEE Communications Surveys Tutorials*, 15(3):1020–1041, 2013.

[109] J. Zheng, M. Z. A. Bhuiyan, S. Liang, X. Xing, and G. Wang. Auction-based adaptive sensor activation algorithm for target tracking in wireless sensor networks. *Future Generation Computer Systems*, 39(0):88 – 99, 2014.

Appendix A

Non-Submodularity of Linear-Gaussian Systems

Submodularity does not extend to information gathering tasks with environment dynamics. We show this by analysing the required conditions for the submodularity of linear-Gaussian systems and by giving a simple counterexample.

Definition A.1 (T-submodularity)**.** *The information structure of an information gathering problem is said to be T-submodular if the sensor observation utility is submodular after the estimate is propagated T timesteps.*

We analyse the 1-submodularity of the sensor selection problem for linear-Gaussian systems and we give a simple counterexample to 1-submodularity.

A.1 Properties of the log-determinant function

To analyse the mutual-information-based sensor selection problem for linear Gaussian systems, we analyse the sign of the derivatives of the log-determinant function.

Define a log-determinant function g such that $g(r, s, t) = \log |A + rB + sC + tD|$ where A, B, C and D are any positive semi-definite matrices. The first and second derivatives of g have the properties given by Lemma A.1.

Lemma A.1 (Sign of log-determinant derivatives)**.** *The first and second derivatives of* $g(r,s,t) = \log|A + rB + sC + tD|$ *where* A*,* B*,* C *and* D *are positive semi-definite have the following properties.*

1. *The derivatives along* r*,* s *and* t *are non-negative.*

$$\frac{\partial f(r,s,t)}{\partial r} \geq 0\,,\ \frac{\partial f(r,s,t)}{\partial s} \geq 0\,,\ \frac{\partial f(r,s,t)}{\partial t} \geq 0 \tag{A.1}$$

2. *The second derivatives of* g *are non-positive.*

$$\frac{\partial^2 f(r,s,t)}{\partial s^2} \leq 0\,,\ \frac{\partial^2 f(r,s,t)}{\partial r^2} \leq 0\,,\ \frac{\partial^2 f(r,s,t)}{\partial t^2} \leq 0 \tag{A.2}$$

$$\frac{\partial^2 f(r,s,t)}{\partial s \partial r} \leq 0\,,\ \frac{\partial^2 f(r,s,t)}{\partial t \partial r} \leq 0\,,\ \frac{\partial^2 f(r,s,t)}{\partial t \partial s} \leq 0 \tag{A.3}$$

Proof.

1.

$$\begin{aligned}\frac{\partial f(r,s,t)}{\partial r} &= \operatorname{tr}\left[(A + rB + sC + tD)^{-1}B\right] \\ &= \operatorname{tr}\left[B^{\frac{1}{2}}(A + rB + sC + tD)^{-1}B^{\frac{1}{2}}\right] \geq 0\end{aligned} \tag{A.4}$$

$$\frac{\partial^3 f(r,s,t)}{\partial t \partial s \partial r} \geq 0 \tag{A.5}$$

By symmetry, the same follows for s and t.

2.

$$\begin{aligned}\frac{\partial^2 f(r,s,t)}{\partial s \partial r} &= \frac{\partial(\frac{\partial f(r,s,t)}{\partial r})}{\partial s} \\ &= \operatorname{tr}\left[-(A + rB + sC + tD)^{-1}C(A + rB + sC + tD)^{-1}B\right] \\ &= -\operatorname{tr}\left[B^{\frac{1}{2}}(A + rB + sC + tD)^{-1}C(A + rB + sC + tD)^{-1}B^{\frac{1}{2}}\right] \\ &\leq 0\end{aligned} \tag{A.6}$$

By symmetry, the same follows for the other variable combinations.

□

The sign semi-definiteness of the first and second derivatives of the log-determinant function allow us to prove the monotonicity and submodularity of the log-determinant function and subsequently the monotonicity and submodularity of the sensor selection problem without environment dynamics.

Theorem A.1 (Log-determinant submodularity)**.**

1. *The* $\log\det$ *function is monotonic over the positive definite cone.*

$$\log|A+B| \geq \log|A| \tag{A.7}$$

2. *The* $\log\det$ *function is submodular over the positive definite cone.*

$$\log|A+B+C| - \log|A+C| \leq \log|A+B| - \log|A| \tag{A.8}$$

Proof.

1.

$$\log|A+B| - \log|A| = \int_0^1 \frac{\partial f}{\partial r}(r,0,0)\,\mathrm{d}r \geq 0 \tag{A.9}$$

Since $\frac{\partial f}{\partial r} \geq 0$ for all $r \geq 0$ Therefore,

$$\begin{aligned} \log|A+B| - \log|A| &\geq 0 \\ \log|A+B| &\geq \log|A| \end{aligned} \tag{A.10}$$

2.

$$\log|A+B| - \log|A| = \int_0^1 \frac{\partial f}{\partial r}(r,0,0)\,\mathrm{d}r \tag{A.11}$$

$$\log|A+B+C| - \log|A+C| = \int_0^1 \frac{\partial f}{\partial r}(r,1,0)\,\mathrm{d}r \tag{A.12}$$

$$\begin{aligned}\int_0^1 \frac{\partial f}{\partial r}(r,1,0)\,\mathrm{d}r - \int_0^1 \frac{\partial f}{\partial r}(r,0,0)\,\mathrm{d}r &= \int_0^1 \frac{\partial}{\partial s}\int_0^1 \frac{\partial f}{\partial r}(r,s,0)\,\mathrm{d}r\,\mathrm{d}s \\ &= \int_0^1\int_0^1 \frac{\partial^2 f}{\partial s \partial r}(r,s,0)\,\mathrm{d}r\,\mathrm{d}s \leq 0\end{aligned} \tag{A.13}$$

The result follows immediately.

□

The submodularity of the sensor selection problem with environment dynamics is related to the third derivative of the log-determinant function. More specifically, if the third derivatives are always positive for a set of sensors and a particular dynamics model, then the sensor selection problem is submodular. The general third derivative is given by Equation A.14. For simplicity of notation, define $E_{r,s,t} = A + rB + sC + tD$.

$$\begin{aligned}\frac{\partial^3 f(r,s,t)}{\partial t \partial s \partial r} &= \frac{\partial\left(\frac{\partial^2 f(r,s,t)}{\partial s \partial r}\right)}{\partial t} \\ &= \operatorname{tr}\left[E_{r,s,t}^{-1} D E_{r,s,t}^{-1} C E_{r,s,t}^{-1} B\right] + \operatorname{tr}\left[E_{r,s,t}^{-1} C E_{r,s,t}^{-1} D E_{r,s,t}^{-1} B\right]\end{aligned} \tag{A.14}$$

The positive-definiteness of the third derivatives leads to the monotonicity of the submodularity gap defined in Definition A.2.

Definition A.2 (Submodularity gap)**.** *The submodularity gap of the* $\log\det$ *function is defined as:*

$$\log|A+B| - \log|A| - \log|A+B+C| + \log|A+C| \tag{A.15}$$

If the third derivatives are positive semi-definite then we can show that the submodularity gap of the log-determinant function is monotonically decreasing over the space of positive semi-definite matrices using the same approach as in Theorem A.1. The

submodularity gap at $t = 0$ is given by Equation A.16 while the submodularity gap at $t = 1$ is given by Equation A.17. The difference given in Equation A.18 is positive if all third derivatives are positive and in that case Inequality A.19 holds.

$$\begin{aligned}&\log|A+B| - \log|A| - \log|A+B+C| + \log|A+C| \\ &= -\int_0^1 \int_0^1 \frac{\partial^2 f}{\partial s \partial r}(r,s,0)\,\mathrm{d}r\,\mathrm{d}s\end{aligned} \tag{A.16}$$

$$\begin{aligned}&\log|A+B+D| - \log|A+D| - \log|A+B+C+D| + \log|A+C+D| \\ &= -\int_0^1 \int_0^1 \frac{\partial^2 f}{\partial s \partial r}(r,s,1)\,\mathrm{d}r\,\mathrm{d}s\end{aligned} \tag{A.17}$$

$$\begin{aligned}&\int_0^1 \int_0^1 \frac{\partial^2 f}{\partial s \partial r}(r,s,1)\,\mathrm{d}r\,\mathrm{d}s - \int_0^1 \int_0^1 \frac{\partial^2 f}{\partial s \partial r}(r,s,0)\,\mathrm{d}r\,\mathrm{d}s \\ &= \int_0^1 \frac{\partial}{\partial t} \int_0^1 \int_0^1 \frac{\partial^2 f}{\partial s \partial r}(r,s,t)\,\mathrm{d}r\,\mathrm{d}s\,\mathrm{d}s \\ &= \int_0^1 \int_0^1 \int_0^1 \frac{\partial^3 f}{\partial t \partial s \partial r}(r,s,t)\,\mathrm{d}r\,\mathrm{d}s\,\mathrm{d}s\end{aligned} \tag{A.18}$$

$$\begin{aligned}&\log|A+B| - \log|A| - \log|A+B+C| + \log|A+C| \\ &\geq \log|A+B+D| - \log|A+D| - \log|A+B+C+D| + \log|A+C+D|\end{aligned} \tag{A.19}$$

We now relate the monotonicity of the submodularity gap of the log-determinant function to the submodularity of the mutual-information-based sensor selection problem for a linear-Gaussian system. To this end, consider the linear Gaussian system given by Equation A.20.

$$x_{k+1} = Ax_k + w_k, \text{ where } w_k \sim \mathcal{N}(0, W) \tag{A.20}$$

Based on the equations of the discrete Kalman filter, the entropy of the estimate at timestep 2 conditioned on the observation set $Z_1 = \{z_1, ..., z_N\}$ at timestep 1 is given by Equation A.21.

$$\begin{aligned} H(x_2|Z_1) = &\quad H(x_1|Z_1) + H(x_2|x_1) - H(x_1|x_2, Z_1) \\ = &\quad \frac{1}{2}\left[\log\left((2\pi e)^n|P_{x_1|Z_1}|\right) + \log\left((2\pi e)^n|W|\right) - \log\left((2\pi e)^n|P_{x_1|x_2,Z_1}|\right)\right] \\ = &\quad \frac{n}{2}\log(2\pi e) + \frac{1}{2}\left[-\log|P^{-1}_{x_1|Z_1}| + \log|W| + \log|P^{-1}_{x_1|x_2,Z_1}|\right] \end{aligned} \tag{A.21}$$

The posterior covariance at time step 1 after fusing the set of observations Z_1 is given by Equation A.22. On the other hand, $P_{x_1|x_2,Z_1}$ can be computed by treating x_2 as an observation with the information matrix $F^T W^{-1} F$ as shown in Equation A.23.

$$P_{x_1|Z_1} = P^{-1}_{x_1} + I_{z_1} + \ldots + I_{z_N} \tag{A.22}$$

$$P^{-1}_{x_1|x_2,Z_1} = P^{-1}_{x_1|Z_1} + F^T W^{-1} F \tag{A.23}$$

By substituting Equation A.22 and Equation A.23 into Equation A.21, we obtain Equation A.24.

$$\begin{aligned} H(x_2|Z_1) = &\quad \frac{n}{2}\log(2\pi e) + \frac{1}{2}\left[-\log|P^{-1}_{x_1} + I_{z_1 1} + \ldots + I_{z_1 N}| + \log|Q|\right. \\ &\quad \left. + \log|P^{-1}_{x_1} + I_{z_1 1} + \ldots + I_{z_1 N} + F^T W^{-1} F|\right] \end{aligned} \tag{A.24}$$

The information gain after is the change in entropy as is given by Equation A.25.

$$I(x_2; Z_1) = H(x_1) - H(x_2|Z_1) \tag{A.25}$$

For any two sets $X_1, Y_1 \subset Z_1$, the conditional information gain is given by Equation A.26.

$$I(x_2; X_1|Y_1) = I(x_2; X_1, Y_1) - I(x_2; Y_1) = H(x_2|X_1) - H(x_2|X_1, Y_1) \tag{A.26}$$

We can now show that 1-submodularity of the sensor selection problem corresponds to the positive-definiteness of the submodularity gap. Consider three sensor observations $z_a, z_b, z_c \in Z_1$. Then the corresponding sensor selection problem is 1-submodular if the value in Equation A.27 is positive.

$$\begin{aligned}
& I(x_2; z_a, z_b) - I(x_2; z_a) - I(x_2; \{z_a, z_b, z_c\}) + I(x_2; \{z_a, z_c\}) \\
= \; & H(x_2|z_a) - H(x_2|\{z_a, z_b\}) - H(x_2|\{z_a, z_c\}) + H(x_2|\{z_a, z_b, z_c\}) \\
= \; & \frac{1}{2}\left[\log|P_{x_1}^{-1} + I_{z_a} + F^T W^{-1} F| - \log|P_{x_1}^{-1} + I_{z_a}|\right. \\
& - \log|P_{x_1}^{-1} + I_{z_a} + I_{z_b} + F^T W^{-1} F| + \log|P_{x_1}^{-1} + I_{z_a} + I_{z_b}| \\
& - \log|P_{x_1}^{-1} + I_{z_a} + I_{z_c} + F^T W^{-1} F| + \log|P_{x_1}^{-1} + I_{z_a} + I_{z_c}| \\
& \left. + \log|P_{x_1}^{-1} + I_{z_a} + I_{z_b} + I_{z_c} + F^T W^{-1} F| - \log|P_{x_1}^{-1} + I_{z_a} + I_{z_b} + I_{z_c}|\right]
\end{aligned} \tag{A.27}$$

By setting the variables of Equation A.19 such that $A := P_{x_1}^{-1} + I_{z_a}$, $B := I_{z_b}$, $C := I_{z_c}$ and $D := F^T W^{-1} F$, it is clear that the positive-definiteness of the submodularity gap corresponds to the submodularity of the sensor selection problem.

A.2 Counterexample

We give a simple linear-Gaussian counterexample that shows that the sensor selection problem is not 1-submodular. Suppose the dynamics of the estimated state are given by Equations A.28-A.30.

$$x_{k+1} = Ax_k + w_k, \;\; w_k \sim \mathcal{N}(0, W) \tag{A.28}$$

$$A = \begin{bmatrix} 1 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.29}$$

$$W = \sigma_w^2 \begin{bmatrix} dt^3/3 & dt^2/2 & 0 & 0 \\ dt^2/2 & dt & 0 & 0 \\ 0 & 0 & dt^3/3 & dt^2/2 \\ 0 & 0 & dt^2/2 & dt \end{bmatrix} \tag{A.30}$$

Further suppose we have three sensors with the linear sensor models given according to Equation A.31 and suppose a prior covariance given by Equation A.32.

$$\begin{aligned} H_1 &= \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} \\ H_2 &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \\ H_3 &= \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \end{aligned} \tag{A.31}$$

$$P_{x_1} = \begin{bmatrix} 4.9760 & 0.3760 & -5.2032 & 1.1435 \\ 0.3760 & 9.2592 & 2.8160 & 6.0227 \\ -5.2032 & 2.8160 & 6.9280 & 2.1174 \\ 1.1435 & 6.0227 & 2.1174 & 8.5429 \end{bmatrix} \tag{A.32}$$

By setting $dt = 10$, the submodularity gap is negative as computed in Equation A.33.

$$-\log|P_{x_1|z_1,z_2}| + \log|P_{x_1|z_1,z_2,z_3}| + \log|P_{x_1|z_1}| - \log|P_{x_1|z_1,z_3}| = -0.1011 \tag{A.33}$$