KTH Signals
Sensors and Systems

On Control under Communication Constraints in Autonomous Multi-robot Systems

ALBERTO SPERANZON

Licentiate Thesis
Stockholm, Sweden 2004

TRITA-S3-REG-0403
ISSN 1404-2150
ISBN 91-7283-885-X

KTH Signals, Sensors and Systems
SE-100 44 Stockholm
SWEDEN

Academic thesis, which with the approval of Kungliga Tekniska Högskolan, will be presented for public review in fulfillment of the requirements for a Licentiate of Engineering in Automatic Control. The public review will be held on 2004/11/12 at Royal Institute of Technology, Stockholm, Room V2, Teknikringen 76.

Tryck: Universitetsservice US AB

Abstract

Multi-robot systems have important applications, such as space explorations, underwater missions, and surveillance operations. In most of these cases robots need to exchange data through communication. Limitations in the communication system however impose constraints on the design of coordination strategies. In this thesis we present three papers on cooperative control problems in which different communication constraints are considered.

The first paper describes a rendezvous problem for a team of robots that exchanges position information through communication. A local control law for each robot should steer the team to a common meeting point when communicated data are quantized. The robots are not equipped with any sensors so the positions of other teammates are not measured. Two different types of quantized communication are considered: uniform and logarithmic. Logarithmic quantization is often preferable since it requires that fewer bits are communicated compared to when uniform quantization is used. For a class of feasible communication topologies, control laws that solve the rendezvous problem are derived.

A hierarchical control structure is proposed in the second paper, for modelling autonomous underwater vehicles employed in finding a minimum of a scalar field. The controller is composed of two layers. The upper layer is the team controller, which is modeled as discrete-event system. It generates waypoints based on the simplex search optimization algorithm. The waypoints are used as target points by the lower control layer, which continuously steers each vehicle from the current to the next waypoint. It is shown that the communication of measurements is needed at each step for the team controller to generate unique waypoints. A protocol is proposed to reduce the amount of data to be exchanged, motivated by that underwater communication is costly in terms of energy.

In the third paper, a probabilistic pursuit–evasion game is considered as an example to study constrained communication in multi-robot systems. This system can be used to model search-and-rescue operations and multi-robot exploration. Communication protocols based on time-triggered and event-triggered synchronization schemes are considered. It is shown that by limiting the communication to events when the probabilistic map updated by the individual pursuer contains new information, as measured by a map entropy, the utilization of the communication link can be considerably improved compared to conventional time-triggered communication.

Acknowledgements

First of all I would like to thanks my supervisor Karl Henrik Johansson for his guidance, inspiring discussions and support throughout my research. Without his never ending patience this thesis would probably not exists today.

I owe my gratitude to Bo Wahlberg who accepted me as a graduate student at the Automatic Control Group. He also gave valuable comments on the thesis manuscript. I am also particularly grateful to Mikael Johansson, who has read and commented several parts.

The main part of the thesis consists in three papers. I am indebted to the coauthors of these. They are stated on the first page of each paper and their affiliation is included in the first chapter of the thesis. The coauthors are Fabio Fagnani, Karl Henrik Johansson, Jorge Silva, João Borges de Sousa and Sandro Zampieri.

I would like to thanks João Pedro Hespanha for introducing me to the problem on probabilistic pursuit–evasion games during his short visit at KTH.

I am grateful to Shankar Sastry and René Vidal for providing the photo of an experiment on pursuit–evasion, João de Sousa for the photo of the underwater vehicle, and Paul S. Schenker for the photo of the collaborative platforms of JPL.

I would like to take the opportunity to thank Henning Schmidt and Jochen Giese for all the interesting discussions we had, not only about research.

To all the friends I met here in Stockholm, who have given me many great moments far away from robots and control systems: thank you.

Special thanks go to my family in Italy whose love and support made the distance between us appear much less than it really is.

Most of all I would like to thank my wife Jovita for her continuous presence, support and encouragement. I must express my adoration of my daughter, Alice, who is the greatest joy of our life.

The research described in this thesis has been sponsored by the European Commission through the RECSYS project IST-2001-32515 and the Swedish Research Council. The support is gratefully acknowledged.

Contents

List of Figures

Chapter 1

Introduction

Strategies for multi-robot systems to perform particular tasks very often rely on the presence of a communication network, which links the different components of the system. Limitations of such communication links impose constraints that must be taken into account in development of the strategies. The papers collected in this thesis deal with the design of controllers when the task to be performed and the communication limitations are specifications of the problem.

The purpose of this introduction is to motivate the problems considered in the papers. In the following we discuss the general problem of controlling a team of robots with communication limitations and three motivating examples are presented. Summaries of the three papers and their main contributions are included at the end of this chapter.

1.1 Multi-robot systems under communication constraints

Over the past decade, a significant shift of focus has occurred in the field of mobile robotics as researchers have begun to investigate problems involving many, rather than single, robots. Several new robotics application areas, such as underwater and space exploration, service robotics in public and private domains can benefit from the use of multi-robot systems. Multi-robot systems can often deal with tasks that are difficult, or even impossible, to be accomplished by an individual robot. A team of robots may also provide redundancy, efficiency, cheaper deployment beyond what is possible with single robots. Let us consider two typical applications of multi-robot systems.

The first (see Figure 1.1) is a search-and-rescue operation where a team of robots is deployed in an unknown or partially known hazardous environment, where a human being or an object should be found. As shown in Figure 1.1, a team of ground mobile robots equipped with sensors are coordinated, together with aerial

Figure 1.1: Multi-robot application: search-and-rescue scenario. (Photograph provided by courtesy of the Robotics and Intelligent Machines Laboratory, Berkeley, USA).[VSK+02].

autonomous robots in order to cooperatively search a large area. Applications similar to search-and-rescue operations are surveillance, where a team of robots has the task of patrolling partially known environments, or exploration where the robots coordinate in order to build a map of an unknown environment. Figure 1.2 shows an example where two robots are needed in order to accomplish a task. In particular the robots cooperate to transport a metal bar, which none of the two robots could transport alone. The robots are equipped with navigation systems and force sensors which measure the torques and forces applied to the bar. One of the robot is the leader and coordination is achieved measuring the forces applied to the bar. These two applications exemplify the effectiveness of using multi-robot systems in solving rather complex tasks.

The new challenge that needs to be addressed in order to make a multi-robot system efficient is the design of coordination strategies that are distributed. This means that the strategies should be designed so that they take advantage of the large number of robots available, they are robust to failures of single individuals and they use the information communicated, when available.

In many applications communication is of fundamental importance in order to improve performances. If we consider the search-and-rescue example, it is easy to imagine that the communication of measurements of the environment would allow the robots to better plan the searching for the human being or the object they should find. Also in the bar-lifting problem communication is needed in order to synchronize the grasp or the release of the bar. However, there are applications in robotics where direct communication is not needed in order to accomplish a task.

Figure 1.2: Multi-robot application: cooperative bar lifting. (Photograph provided by courtesy of Jet Propulsion Laboratory, Pasadena, USA).

The transportation of the bar, when no grasping or releasing actions are needed, is an example.

In this thesis we will consider the design and analysis of distributed strategies when robots can exchange data. The presence of a communication network connecting robots, however, imposes limitations since data needs to be quantized to be sent over the network, it can be received with large delays due to bandwidth limitations of the channels or it can be lost due to noise in the environment or because the network is congested, etc. Thus a new design problem arises: how do we design control strategies that coordinate a team of robots under communication constraints? In this thesis we presents three papers in which three different coordination problems are considered under various communication constraints: quantization (Paper A), noisy channel (Paper B), bandwidth limitation (Paper C).

1.2 Motivating examples

We consider here three motivating examples which are related to the coordination problems addressed in the papers.

Example 1 - Formation control

Formation control problems arise in those applications where a team of robots have to maintain specific geometries. The need of keeping specific formations could be motivated by sensor fusion constraints, as for example in a team of spacecrafts employed to create a large interferometer [Lin03], energy consumption limitations,

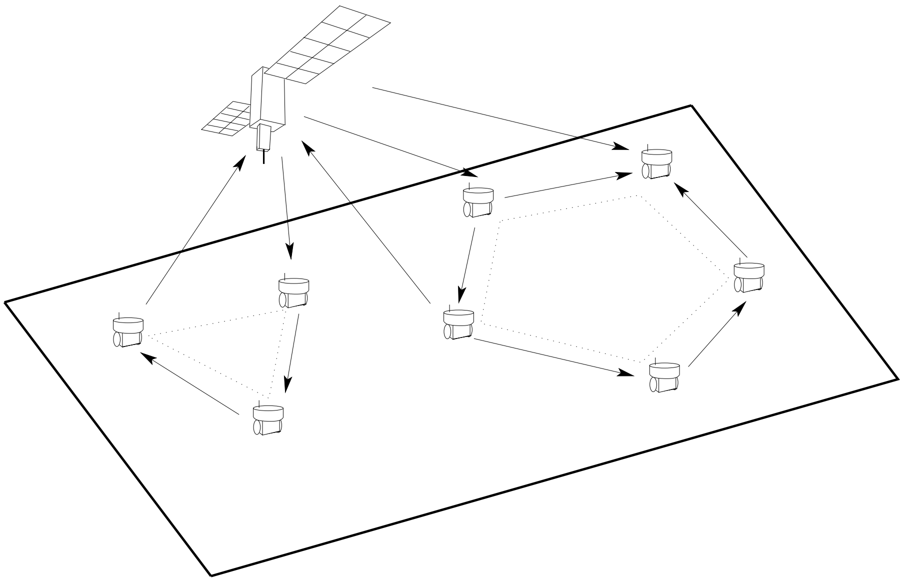


Figure 1.3: Example 1. Formation control.

which can be attained by robots flying in formation [BLH01], and nearest-neighbor communication constraints [SM03]. Figure 1.3 shows a scenario where a team of ground vehicles together with a satellite are coordinated to keep geometric formations (dotted lines). Vehicles can communicate to other robots of the same formation or with the satellite (arrows). Thus there is a large amount of data that has to be communicated if the number of vehicles is large. In Paper A we will discuss the rendezvous problem, which is a particular problem of formation control, for a team of robots when the communication takes place over quantized channels.

Example 2 - Collaborative tracking

Collaborative tracking of moving object consists in aggregate a multitude of sensor data to improve accuracy of the position and velocity estimates of the moving object and to use such information for tracking [MSJH04, ZSR02]. Figure 1.4 shows a simple example. In this case the robots are assume to have limited sensor capabilities so they need to fuse their information in order to determine the position of the moving object. Communication limitations can degrade the global estimate. In Paper B we consider the problem of finding a local minima of a scalar field using two underwater vehicles. In underwater applications communication becomes a critical issue since underwater channels are noisy and a large amount of energy needs to be used for the transmission.

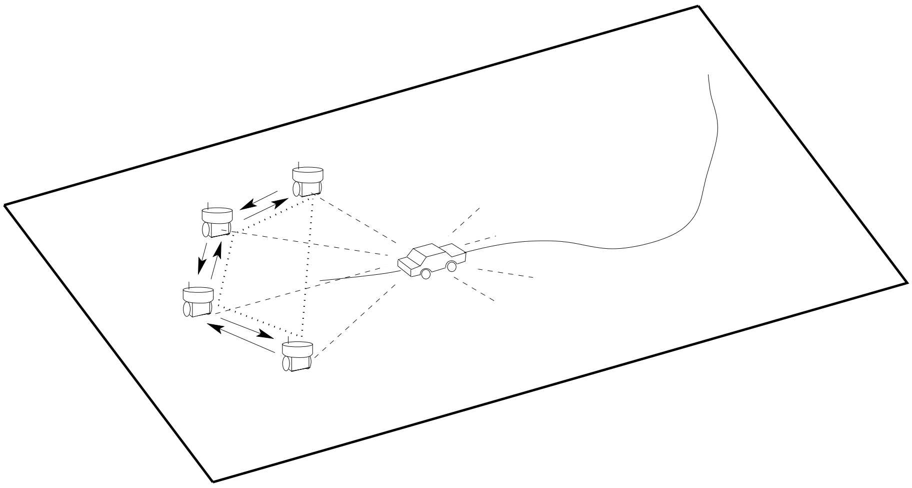


Figure 1.4: Example 2. Collaborative tracking.

Example 3 - Collaborative map building
Collaborative map building is another common problem for multi-robot systems, in which robots are employed in building a map of an unknown environment [FBKT00]. In Figure 1.5 robots are deployed in a structured environment such as a house or a factory, and they exchange information in order to build a complete map. In this case it is crucial to consider the quantity of data transmitted over the network, because of bandwidth constraints and traffic congestion. In Paper C we discuss problems related to collaborative map building.

1.3 Main contributions of the thesis

This thesis contains the three papers listed below.

Paper A: *On multi-vehicle rendezvous under quantized communication.*

Multi-vehicle rendezvous can be considered as a formation control problem where all members of the group eventually meet at a single unspecified location. In this paper we assume the vehicles do not perform any active sensing of the neighborhood, but they can exchange information through quantized communication channels. We also assume the vehicles can communicate with any other vehicle. In order to have an efficient utilization of the bandwidth some data is transmitted through logarithmically quantized channels. We derive some communication topologies i.e., a graph associated to the system that gives the type of quantized channel utilized between pairs of vehicles. We prove that for such communication topologies the relative dis-

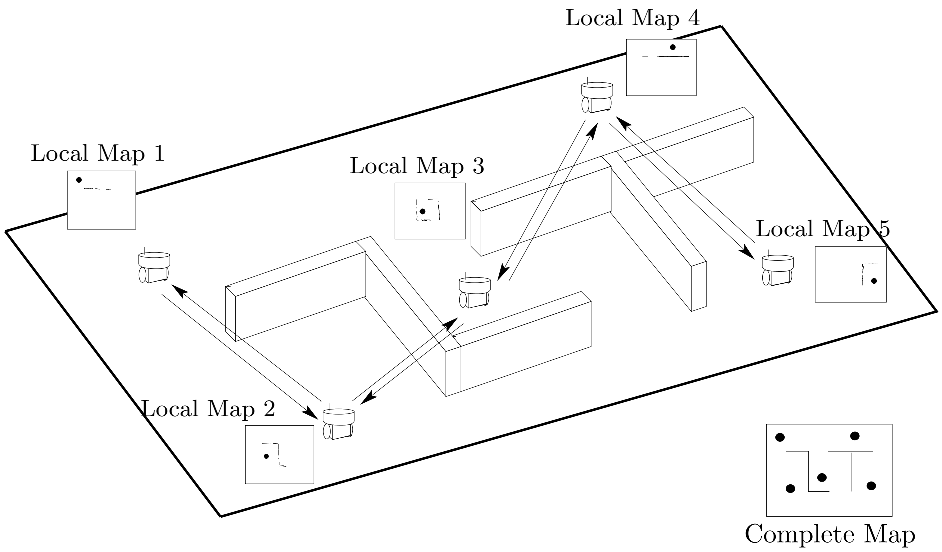


Figure 1.5: Example 3. Collaborative map building.

tance between vehicles converges to a neighborhood of the origin.

Paper A is based on the following publications:

> K. H. Johansson, A. Speranzon, and S. Zampieri: "On quantization and communication topologies in multi-vehicle rendezvous". *Submitted to* 16th IFAC World Congress, Prague, Czech Republic, 2005

> F. Fagnani, K. H. Johansson, A. Speranzon, and S. Zampieri: "On multi-vehicle rendezvous under quantized communication". International Symposium on Mathematical Theory of Networks and Systems, Leuven, Belgium, 2004

Paper B: *On collaborative optimization and communication for a team of autonomous underwater vehicle.*

A hierarchical control structure is proposed in the second paper, for modelling autonomous underwater vehicles employed in finding a minimum of a scalar field. The controller is composed of two layers. The upper layer is the team controller which is modeled as discrete-event system. It generates waypoints based on the simplex search optimization algorithm. The waypoints are used as target points by the lower control layer, which continuously steer each vehicle from the current to the next waypoint. It is shown that the communication of measurements is needed at each step for the team controller to generate unique waypoints. A protocol is proposed to reduces the amount of data to be exchanged, motivated by that underwater communication is costly in terms of energy.

Paper B is based on the following publications:

> J.B. de Sousa, K.H. Johansson, A. Speranzon, and J. Silva: “A control architecture for multiple submarines in coordinated search missions”. *Submitted to* 16th IFAC World Congress, Prague, Czech Republic, 2005

> J. Silva, A. Speranzon, J.B. de Sousa, and K. H. Johansson: “Hierarchical search strategy for a team of autonomous vehicles”, International Symposium on Intelligent Autonomous Vehicles, Lisbon, 2004

> A. Speranzon, J. Silva, J.B. de Sousa, and K. H. Johansson: “On collaborative optimization and communication for a team of autonomous underwater vehicles”, Reglermöte, Gothenburg, 2004.

Paper C: *On some communication schemes for distributed pursuit–evasion games.*

Pursuit–evasion game is a typical problem in game theory that can be used as model for multi-robot search-rescue problems, exploration, etc. In this paper we consider the probabilistic set-up introduced by Hespanha et al. [HKS99] where all pursuers contributes to build a global probabilistic map, which is the probability of the evader to be in one of the possible cells the environment has been divided in, given all the measurements taken by all pursuers. Such model is extended and made distributed. Each pursuer builds a local probabilistic map which is communicated with other pursuers (synchronized) only when enough information has been collected by each pursuers. We compare two possible communication strategies: periodic communication and event-based communication. The event that triggers the communication is the entropy of the probabilistic map i.e., the information content of the map. This idea can be related to the difference of periodic and event-based sampling. Communication of probabilistic maps with limited bandwidth is discussed in the end of the paper.

Paper C is based on the following publications:

> A. Speranzon, and K. H. Johansson: “On some communication schemes for distributed pursuit-evasion games”. IEEE Conference on Decision and Control, Maui, HI, 2003

> A. Speranzon, and K. H. Johansson: “Distributed pursuit-evasion game: evaluation of some communication schemes”. Symposium on Autonomous Intelligent Networks and Systems. Menlo Park, CA, 2003

A. Speranzon, and K. H. Johansson: “On localization and communication issues in pursuit-evasion game”. IROS, Workshop on Cooperative Robotics. Lausanne, Switzerland, 2002

1.4 Remark on notation

The thesis consists of three separated papers, therefore the notation throughout the thesis in not consistent, but is introduced separately in each paper.

1.5 Other publications

The author of the thesis has been co-author of other publications in the field of robotics and automatic control and some have influenced the contents of this thesis. They include the following:

M. Mazo, A. Speranzon, K.H. Johansson, X. Hu: “Multi-robot tracking of a moving object using directional sensors”. IEEE International Conference on Robotics and Automation, 2004.

E. Pagello, A. D’Angelo, C. Ferrari, R. Polesel, R. Rosati, A. Speranzon: “Emergent behaviors of a robot team performing cooperative tasks”. *Advanced Robotics*, Vol. 15, No. 1, 3-20, 2003.

C. Altafini, A. Speranzon, K.H. Johansson: “Hybrid Control of a Truck and Trailer Vehicle”. In Hybrid Systems: Computation and Control, C.J. Tomlin and M.R. Greenstreet, Ed. - Lecture Notes in Computer Science, Springer-Verlag. 2002.

P. de Pascalis, M. Ferraresso, M. Lorenzetti, A. Modolo, M. Peluso, R. Polesel, R. Rosati, N. Scattolin, A. Speranzon, W. Zanette: “Golem Team in Middle-Sized Robots League”. In RoboCup-2000: Robot Soccer World Cup IV, P. Stone, T. Balch, and G. Kraetszchmar, Ed. - Springer-Verlag, Berlin, 2001.

C. Altafini, A. Speranzon, B. Wahlberg: “A Feedback Control Scheme for Reversing a Truck and Trailer Vehicle”. *IEEE Transactions on Robotics and Automation*, Dec. 2001.

R. Polesel, R. Rosati, A. Speranzon, C. Ferrari, E. Pagello: “Using Collision Avoidance Algorithms for Designing Multi-robot Emergent Behaviors”. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2000.

1.6 Affiliation of the coauthors of the papers in the thesis

Fabio Fagnani
Dipartimento di Matematica,
Politecnico di Torino,
Corso Duca degli Abruzzi 24,
10129 Torino, Italy
fagnani@calvino.polito.it

Karl Henrik Johansson
Royal Institute of Technology
Dept. of Signals, Sensors and Systems
Qsquldas väg 10
100-44 Stockholm, Sweden
kallej@s3.kth.se

Jorge Silva
Instituto Superior
de Engenharia do Porto,
Rua Dr. António Bernardino
de Almeida 431,
4200-072 Porto, Portugal
jmes@dee.isep.ipp.pt

Joao B. De Sousa
Faculdade de Engenharia
da Universidade do Porto,
Rua Dr. Roberto Frias,
44200-465 Porto, Portugal
jtasso@fe.up.pt

Sandro Zampieri
Dipartimento di Ingegneria
dell'Informazione,
Universitá di Padova,
via Gradenigo 6/A,
35131 Padova, Italy
zampi@dei.unipd.it

References

[BLH01] R.W. Beard, J. Lawton, and F.Y. Hadaegh. A coordination architecture for spacecraft formation control. *IEEE Transaction on Control Systems Technology*, 9:777–790, 2001.

[FBKT00] D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A probabilistic approach to collaborative multi-robot localization. *Special issue of Autonomous Robots on Heterogeneous Multi-Robot Systems*, 3, 2000.

[HKS99] J.P. Hespanha, H. J. Kim, and S. Sastry. Multiple-agent probabilistic pursuit–evasion games. In *IEEE Conference on Decision and Control*, volume 3, pages 2432–2437, 1999.

[Lin03] C.A. Lindensmith. Technology plan for the Terrestrial Planet Finder. Technical report, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, 2003.

[MSJH04] M. Mazo, A. Speranzon, K.H. Johansson, and X. Hu. Multi-robot tracking of a moving object using directional sensors. In *IEEE International Conference on Robotics and Automation*, 2004.

[SM03] R. O. Saber and R. M. Murray. Flocking with obstacle avoidance: cooperation with limited communication in mobile networks. In *Proc. of the 42nd IEEE Conference on Decision and Control, HI, USA*, 2003.

[VSK+02] R. Vidal, O. Shakernia, J. Kim, D. Shim, and S. Sastry. Probabilistic pursuit-evasion games: Theory, implementation and experimental evaluation. *IEEE Transactions on Robotics and Automation*, 2002.

[ZSR02] F. Zhao, J. Shin, and J. Reich. Information-driven dynamic sensor collaboration for tracking applications. *IEEE Signal Processing Magazine*, 2002.

Part I

Background

Chapter 2

Control under communication constraints

> "*In control and communication we are always fighting nature's tendency to degrade the organized and to destroy the meaningful*"
> - N. Wiener

In recent years a certain interest has been developed on control problems in which the communication is an essential component [Mit01]. A block diagram of such a system is shown in Figure 2.1. For these systems, control design objectives such as regulation, tracking, etc. need to be addressed considering the limitations imposed by communication channels linking the different components (cf. Figure 2.1).

In the following we describe in some detail the digital communication channel highlighting the constraints that it imposes. We then discuss how such constraints can be modeled from a control perspective in order to incorporate them in the design of controllers.

2.1 Digital communication system

A block diagram of a digital communication system is shown in Figure 2.2. The problem of communication is how to design the transmitter and receiver (shown in dashed line in Figure 2.2), so that symbols selected at the source can be reproduced at the destination either exactly or approximately. The reason why it is a difficult problem is due to the presence of the noise η in the channel and bandwidth constraints. The noise degrades the information sent over the channel and specific design of the transmitter and receiver are needed in order to reliably communicate symbols. The noise comes from many different sources, such as thermal noise in the components, interference or degradation due to signal attenuation, amplitude

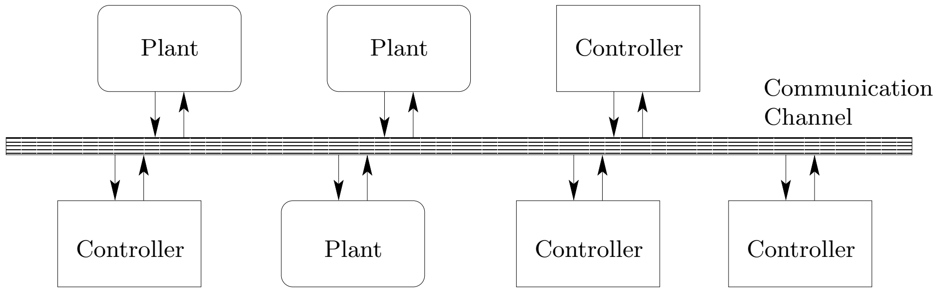


Figure 2.1: Block diagram of control systems interconnected via a communication channel.

and phase distortion or multi-path distortion, which is caused by the existence of more than one propagation path between the transmitter and the receiver. Bandwidth constraints are generally caused by physical limitations of the medium and electronic components used to implement the different parts of the transmitter and receiver.

We will review in the following the main properties, and limitation of a digital communication

The source is modeled as a discrete-time stochastic process, $\{X_n\}$ with alphabet $\mathcal{X}$ and probability mass function $p(x_n) = \Pr[X_n = x_n]$ for $x_n \in \mathcal{X}$. We also assume that the random variables $\{X_n\}$ are independent and identical distributed (i.i.d.)[1]. We assume that the source alphabet $\mathcal{X}$ has cardinality M.

The transmitter operates on the symbol in order to produce a signal compatible with the channel, adding some redundancy used at the receiver to correct possible errors. The transmitter is composed of a source and channel encoder and a modulator.

For a digital memoryless communication system, a source encoder is defined as the mapping[2]

$$c : \mathcal{X} \to \mathcal{D}^* : x \mapsto c(x) \tag{2.1}$$

where x is a realization of X The source codeword $c(x)$ is a string of length m of symbols chosen from a discrete alphabet $\mathcal{D}$ with $\mathrm{card}(\mathcal{D}) = d$. The mapping c is usually chosen so that the symbol x is converted into a codeword $c(x)$ that has little or no redundancy. Roughly speaking the M symbols produced by the source are mapped into M source codewords, which represent a compressed version of the original symbols.

The channel encoders add redundancy in a controlled fashion, so that errors caused by channel noise η can be detected and possibly corrected at the receiver

[1] In the following we neglect the time dependence of X_n since we have assumed the random variables being i.i.d.

[2] For a set $\mathcal{A}$ we defined $\mathcal{A}^n = \{(a_1, \ldots, a_n) |\, a_i \in \mathcal{A}, i = 1, \ldots, n\}$ and $\mathcal{A}^* = \cup_{n=0}^{\infty} \mathcal{A}^n$.

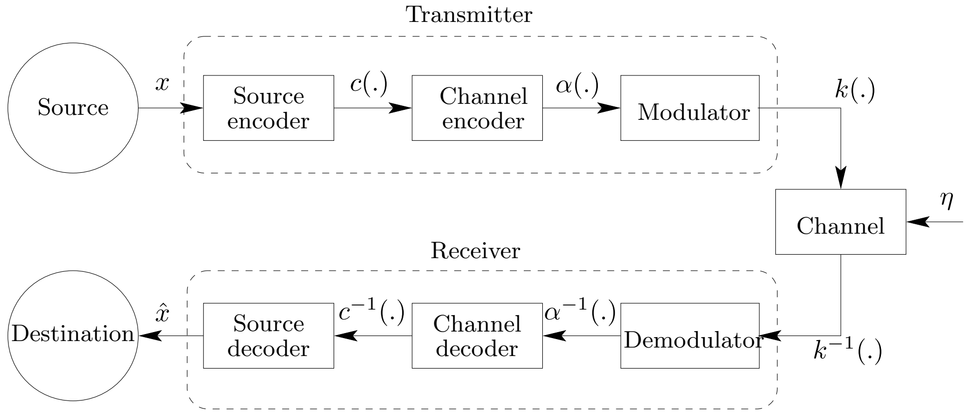


Figure 2.2: General block diagram of a digital communication system.

side. Let $\mathcal{I}_M = \{0, \ldots, M-1\}$ be an index set representing the M source codewords. A channel encoder can be defined as the mapping

$$\alpha : \mathcal{I}_M \to \mathcal{Z}^N.$$

where $\mathcal{Z}$ is the channel alphabet. The encoder α maps M different source codewords into M channel codewords, which are all represented by N bits. This map defines a (M, N) block channel code[3].

The modulator converts digital data into a signal waveform transmitted over the channel. Formally, it is defined by the mapping

$$k : J_M \to \mathcal{S} : j \mapsto s_j(t)$$

where $J_M = \{0, \ldots, M-1\}$ is a representation of the channel codewords and $s_j(t) \in \mathcal{S} = \{s_0(t), \ldots, s_{M-1}(t)\}$ is a continuous-time representation of the j-th channel symbol.

The channel is merely the medium used to transmit the signal $s_j(t)$. Generally it is air, water, wire, etc.

The receiver shown in Figure 2.2 performs the inverse operation of the transmitter. In particular the demodulator maps a received signal waveform $\hat{s}_j(t)$ to an index j, which corresponds to the channel codeword $\widehat{\alpha^{-1}}(j)$. Such codeword is corrected by the channel decoder and mapped back to the source codeword $\hat{x}$. A transmission error occurs when $\hat{x} \neq x$.

The design of the transmitter and receiver, so that reliable communication is achieved, can be split into the design of source encoder/decoder, channel encoder/decoder and modulator/demodulator. The design objectives of each component are

[3]Notice that here we have restricted ourselves to block codes. It is possible to generalize to other codes, see [CT91].

Source encoder/decoder: find a representation of the input symbols $\{x_0, \ldots, x_{M-1}\}$ with minimal redundancy from which the original symbols can be reconstructed,

Channel encoder/decoder: add redundancy so that large number of errors can be detected and corrected,

Modulator/demodulator: choose a set of continuous-time waveforms $\{s_0(t), \ldots, s_{M-1}(t)\}$ such that is possible to discriminate which signal has been sent.

Next we present two fundamental theorems of digital communication, namely the Shannon's Source Coding Theorem and Shannon's Channel Coding Theorem [Sha48]. First, however, let us introduce some notation, which is also used in the second part of the thesis.

The entropy $H(X)$ of a discrete random variable X is

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x).$$

The entropy is as a measure (in bits) of the uncertainty of a random variable. Let c represents a source encoder as in (2.1). Then the expected length of the source code (the ensemble of all source codewords) is defined as

$$L = \sum_{x \in \mathcal{X}} p(x) \ell(x)$$

where $\ell(x)$ is the length of the codeword $c(x)$ in symbols. Let us define the (average) rate R of the source code as

$$R_s = L \log d.$$

where d is the cardinality of the source code alphabet $\mathcal{D}$. The rate R_s is measured in bits per source symbol. We can then state the following theorem.

Theorem 2.1 (Shannon's Source Coding Theorem) *For a discrete memoryless source with entropy H a lossless source code of rate R_s exists if $R_s > H$. A lossless code does not exists for any $R_s < H$.*

This means that for a binary source coding the average source code length is bounded below by the entropy of the source.

Let us assume the source code is lossless and let $\mathbf{X} = \{X_1, \ldots, X_N\}$ be a sequence of N random variables and $\mathbf{x} = (x_1, \ldots, x_N)$ a sequence of symbols[4].

The rate R_c for the (M, N) block channel code is defined as

$$R_c = \frac{\log M}{N}$$

[4] In the following we will denote with capital boldface letters finite sequences of random variables and with small boldface letters finite sequences of symbols.

measured in bits per channel use. A rate R_c is said to be achievable if there exists a sequence of (M, N) block codes such that the average probability of error

$$P_e = \sum_{\mathbf{x} \in \mathcal{X}^N} \Pr\big[\hat{\mathbf{X}} \neq \mathbf{x} | \mathbf{X} = \mathbf{x}\big]\, p(\mathbf{x}) \to 0 \text{ as } N \to \infty.$$

Let the digital channel be the system composed by modulator, channel and demodulator. If we denote with Z_n the input variable to the digital channel, namely channel codewords, and with Y_n the out variable with $Z_n \in \mathcal{Z}$ and $Y_n \in \mathcal{Y}$, then a discrete channel is described by the following conditional probability

$$p(\mathbf{y}|\mathbf{z}) = \Pr[\mathbf{Y} = \mathbf{y}|\mathbf{Z} = \mathbf{z}].$$

In particular for a discrete memoryless channel we can write

$$p(\mathbf{y}|\mathbf{z}) = \prod_{i=1}^{N} p(y_i|z_i).$$

The capacity of the the digital memoryless channel is defined as

$$\begin{aligned} C &= \max_{p(z)} I(Y;Z) \\ &= \max_{p(z)} \left\{ \sum_{z \in \mathcal{Z}} \sum_{y \in \mathcal{Y}} p(y|z)p(z) \log \frac{p(y|z)}{\sum_{z \in \mathcal{Z}} p(y|z)p(z)} \right\} \end{aligned}$$

where $p(z)$ is the probability mass function for $Z \in \mathcal{Z}$. We can state the following theorem.

Theorem 2.2 (Shannon's Channel Coding Theorem) *Let C be the capacity of a discrete memoryless channel the all rates $R_c < C$ are achievable. No rate $R_c > C$ is achievable.*

The capacity C of a channel can be considered as the maximum of all achievable rates. The capacity can be computed for various channel models and depends on the noise affecting the channel.

The important consequence of this theorem is that there is an upper-bound on the maximum rate at which we can transmit reliably and such rate depends on the channel.

2.2 Communication limitations in the control problem

The two theorems we have reviewed in the previous section state what are the limitations, in a digital communication system, that need to be considered in order to reliably communicate a message from the source to the destination. In practice,

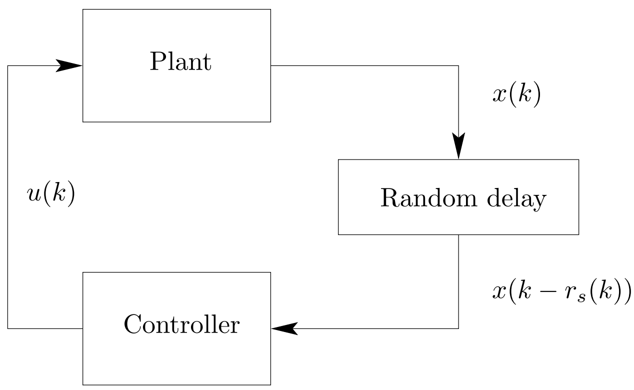


Figure 2.3: Control system with communication channel modeled as random delay.

in order to achieve low probability of error, coding strategies require to generate long channel codewords. This directly influences the complexity of the channel encoder/decoder which can determine the message to be delayed. Bandwidth limitations impose a maximum rate at which the messages can be sent and large messages need to be split into smaller sub-messages. If the communication system is shared, message dropping or delays can occur.

These limitations on delays, loss, and quantization, originated by the digital communication systems, impose constraint in the design of distributed controllers. All these constraints are not typically encounter in the standard design of, only recently researchers have addressed these problems in the contest of control under communication constraints. In the following we review some models of simple control systems when the communication system imposes random time delays, loss of data and quantization.

2.2.1 Random delays

Delays in the control loop are undesirable since, in general, they reduce the stability of the system. When the delays are bounded, the control system can be modeled as a finite dimensional discrete-time jump-linear system with jumps modeled as a finite state Markov chain. In particular, for a linear, as shown in Figure 2.3, we have

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ u(k) &= K_{r_s(k)} x(k - r_s(k)) \end{aligned}$$

where $r_s(k)$ is modeled as a finite state Markov process [XHH00, Nil98].

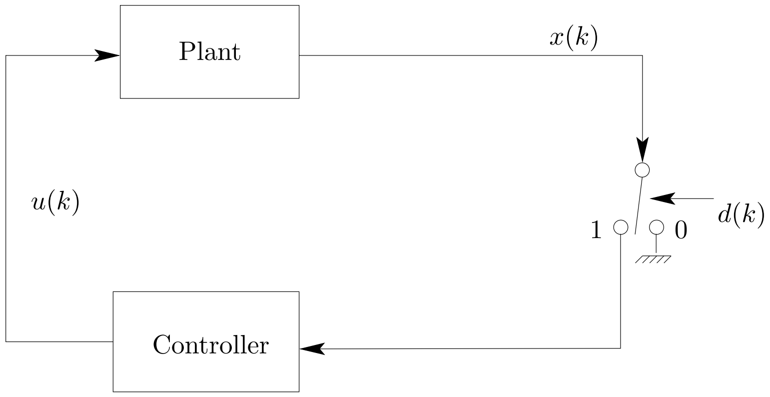


Figure 2.4: Control system with communication channel modeled as data drop (solid line). Data are dropped when the switch is in position 0.

2.2.2 Loss of data

Loss of data in a control system with communication channels can be depicted as shown in Figure 2.4. The communication channel is modeled as a switch controlled by a stochastic process $d(k)$, called dropout process, assumed i.i.d. with probability distribution $\Pr[d(k) = 0] = \epsilon$ and $\Pr[d(k) = 1] = 1 - \epsilon$. When $d(k) = 0$ then the data is dropped (position 0 of the switch) and when $d(k) = 1$ the state $x(k)$ is sent successfully (position 1 of the switch). When a drop occurs, data available in the controller can be used to predict the dropped data. Note, however, that persistent packet losses are very hard to handle since they effectively break the feedback loop.

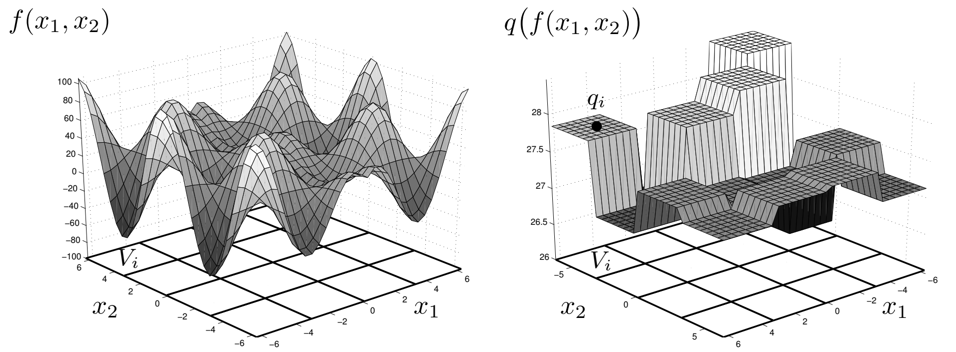
2.2.3 Quantization

Let $\mathcal{I}$ be an index set and $\mathcal{Q} = \{q_i\}_{i\in\mathcal{I}}$ a subset of $\mathbb{R}^n$. A quantizer is mathematically described by a piecewise constant function

$$q : \mathbb{R}^n \to \mathcal{Q}.$$

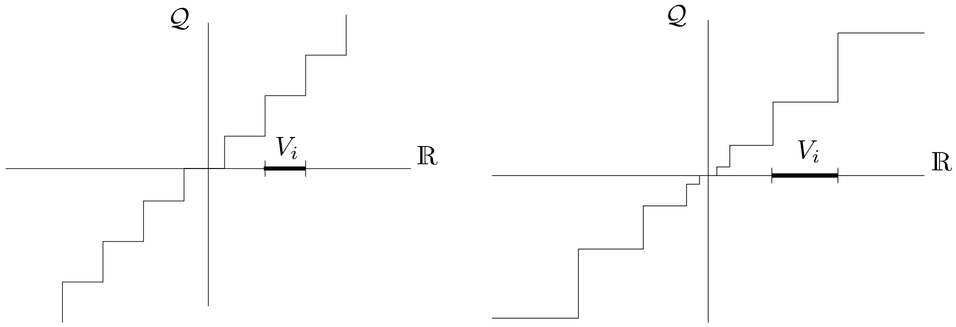
To each point $q_i \in Q$ we can associate a quantization region $V_i = \mathrm{cl}\{x \in \mathbb{R}^n | q(x) = q_i\}$, where cl is the closure of a set. In Figure 2.5(a) is shown a two dimensional function $f(x_1, x_2)$ and in Figure 2.5(b) the quantized version of it, $q\big(f(x, y)\big)$. As shown in the figures, the function q maps V_i to a single value q_i.

Depending on the quantization map q, we can divide the quantizers in two classes: uniform quantizers and nonuniform quantizers. The first class is characterized by the fact that the quantization regions are of equal size, see in Figure 2.6(a). The nonuniform quantizers have quantization regions that need not to be equal. Figure 2.6(b) shows an example of scalar logarithmic. In Paper A we will discuss the application of uniform and logarithmic quantizers. We recall here how these



(a) Function $f(x_1, x_2)$. (b) Quantized function, $q(f(x, y))$.

Figure 2.5: A 2D quantizer. V_i represents a quantization region and q_i is the quantized value associated to the quantization region V_i.



(a) Uniform quantizer. The scalar case. (b) Logarithmic quantizer. The scalar case.

Figure 2.6: Two different types of quantizers: uniform and logarithmic.

maps are defined. Let $\delta > 0$ be the quantization step. A scalar uniform quantizer is a map $q_u : \mathbb{R} \to Q$ such that

$$q_{\mathrm{u}}(x) = \delta \left\lfloor \frac{x}{\delta} \right\rfloor .$$

The quantization regions for a scalar uniform quantizer are the intervals $V_i = [-\delta/2 + i\delta, \delta/2 + i\delta]$, $i = \mathbb{Z}$.

A logarithmic quantizer is a map $q_\ell : \mathbb{R} \to Q$ such that

$$q_\ell(x) = \exp(q_u(\ln x)).$$

The quantization regions for a scalar logarithmic quantizer are the intervals $V_i = [\exp(-\delta/2 + i\delta), \exp(\delta/2 + i\delta)]$, $i = \mathbb{Z}$.

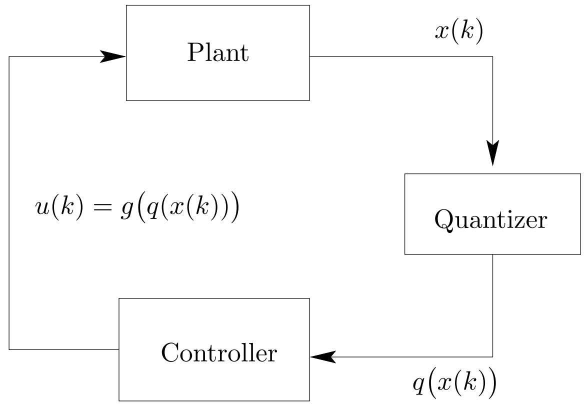


Figure 2.7: Control system with communication modeled as quantization.

Feedback control problems where the communication is modeled as quantization have been considered quite extensively in the literature, see for example [PB03, FZ03, EM01, BL00, Del90]. Figure 2.7 depicts such system where the feedback control action is $u(k) = g(q(x(k)))$. Depending if the quantization step δ is time independent or not, two different quantizers have been studied: static and dynamic quantizers. The basic idea of a dynamic quantizer is that changing the size of the quantization intervals, one can extract more information in a specific area of the domain on which q is defined[5]. An intuitive way of thinking of a dynamic quantizer is to consider a digital camera with zooming capability and finite number of pixels. The possibility to zoom-in a specific part of the image allows the system to see details that otherwise would have been lost.

[5]Notice that this discussion is not restricted to scalar quantizers. In the general case δ is a vector defining the quantization step along each dimension $\mathbb{R}^n$.

Chapter 3

Autonomous multi-robot systems

"If you want to be incrementally better: Be competitive. If you want to be exponentially better: Be cooperative."
- Anonymous

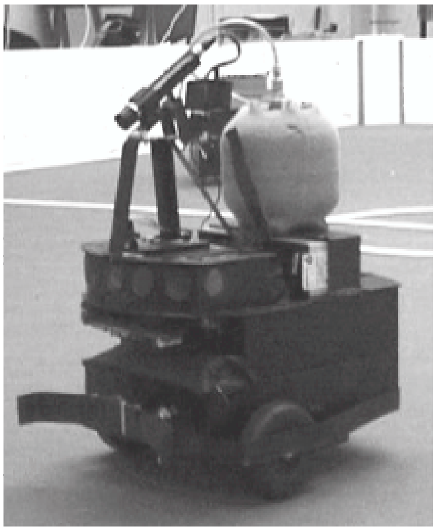
A multi-robot system is a collection of robots which cooperate to solve a common task. Such systems exhibit many advantages compared to single-robot solutions, such as flexibility, robustness, feasibility, efficiency [DGR00, Par00, KYC$^+$96]. The possibility of splitting the robots in groups [SVS04] performing tasks at different locations, the ability of completing a task even if some robot breaks down [GC01], the capability of fusing sensor data in order to obtain better information about the environment [MSJH04], are few examples of the potential of multi-robot system compared to single-robot solutions. However, in order to design cooperative control strategies for multi-robot systems with such properties, a mathematical model of the system and the task is needed. In the following we present mathematical models of common mobile robot platform and then review some mathematical models for multi-robot systems that have some relationships with the models used in three papers included in the thesis.

3.1 Single-robot models

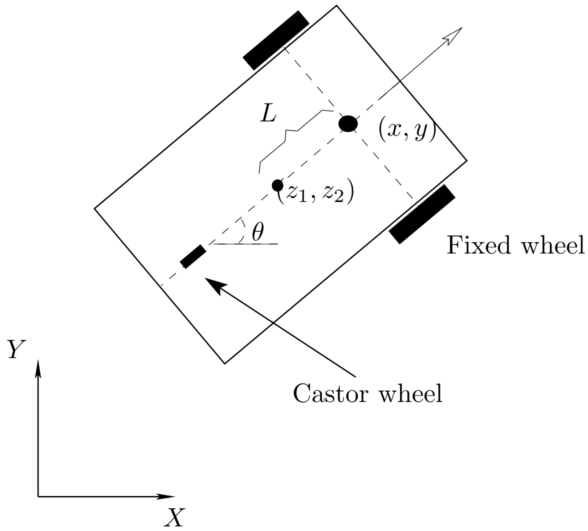
We review here some common mathematical models of single robots.

3.1.1 Unicycle robot

A unicycle robot is composed of two independently actuated wheels and a small passive castor wheel used to keep the balance. The simplest kinematic model for



(a) Unicycle robot used by the Padova team during the Robocup world cup championship in 1999. The platform is a modified version of a Pioneer robot of ActivMedia. [PDF+03].



(b) The unicycle model.

Figure 3.1: Unicycle robot.

the unicycle is given by

$$\begin{aligned}\dot{x} &= v\cos\theta \\ \dot{y} &= v\sin\theta \\ \dot{\theta} &= \omega\end{aligned}$$

where (x, y) is the center point of the front wheel axis and θ is the orientation of the unicycle, see Figure 3.1(b). All quantities are respect a global coordinate frame. The input signals v and ω are the translational and angular velocities, respectively. The unicycle model is used to describe many indoor robots, such as the robot shown in Figure 3.1(a) as well as outdoor mobile robots with skid-steer capabilities (caterpillar-like robots).

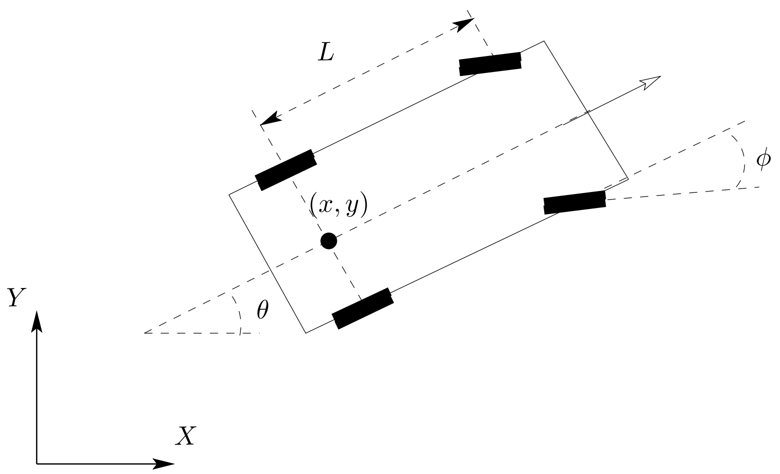


Figure 3.2: Car-like robot.

3.1.2 Car-like robot

A car-like robot is shown in Figure 3.2. The mathematical model is

$$\dot{x} = v \cos\theta$$
$$\dot{y} = v \sin\theta$$
$$\dot{\theta} = \frac{v \tan\phi}{L}.$$

The control inputs are the velocity v and the steering angle ϕ. The model is very similar to that of a unicycle. The main difference is that $\dot{\theta}$ depends on v. Thus, for a car-like robot it is not possible to turn on place.

3.1.3 Underwater robot

Autonomous underwater vehicles (AUV's) are typically small unmanned submarines. In Figure 3.3 is shown a photo of the AUV Isurus of University of Porto, Portugal.

Referring to Figure 3.4, we have

- $(x, y, z)^T$ is the position of the vehicle with respect to a global coordinate system,
- $(\phi, \theta, \psi)^T$ is the attitude of the vehicle with respect to a global coordinate system,
- $(u, v, w)^T$ are the linear velocities with respect to a body-fixed coordinate frame,

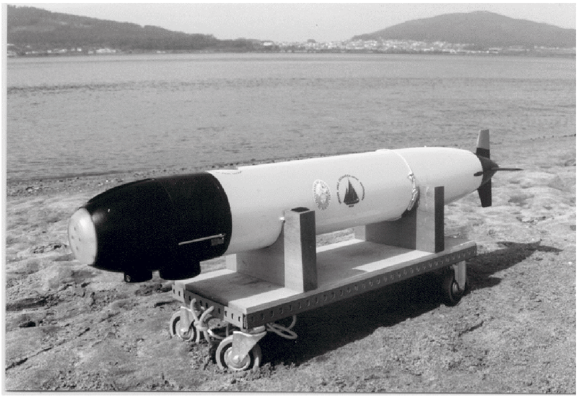


Figure 3.3: Underwater robot ISURUS. (Photograph provided by courtesy of the Underwater Systems and Technology Laboratory, University of Porto, Portugal).

- $(p, q, r)^T$ are the angular velocities with respect to a body-fixed coordinate frame.

The general kinematic equations of an underwater vehicle are

$$
\begin{aligned}
\dot{x} &= u\cos\psi\cos\theta + v(\cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi) \\
&\quad + w(\sin\psi\sin\phi + \cos\psi\cos\phi\sin\theta) \\
\dot{y} &= u\sin\psi\cos\theta + v(\cos\psi\cos\phi + \sin\phi\sin\theta\sin\psi) \\
&\quad + w(\sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi) \\
\dot{z} &= -u\sin\theta + v\cos\theta\sin\phi + w\cos\theta\cos\phi \\
\dot{\theta} &= p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \\
\dot{\phi} &= q\cos\phi - r\sin\phi \\
\dot{\psi} &= q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta}, \qquad \theta \neq \pm 90^\circ
\end{aligned}
$$

In Paper B we have considered a simplified model of an underwater vehicle constraining to move in a plane. In this case the kinematic model of the vehicle is

$$
\begin{aligned}
\dot{x} &= u\cos\psi - v\sin\psi \\
\dot{y} &= u\sin\psi + v\cos\psi \\
\dot{\psi} &= r
\end{aligned}
$$

where x and y are the cartesian coordinates of its center of mass with respect to a global coordinate frame, ψ defines the vehicle's orientation and r its angular speed. The input u (surge speed) and v (sway speed) are the body-fixed frame components of the vehicle's speed (see Figure 3.4).

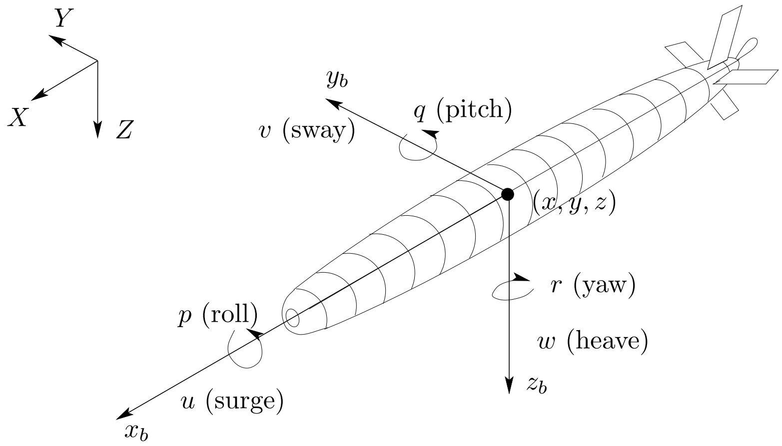


Figure 3.4: Underwater robot.

3.2 Multi-robots mathematical model

In the previous sections, we have described typical mathematical models for single-robot systems. For multi-robot systems several mathematical models have been proposed in the literature. We will review here some of these models that have important connections with those proposed in the papers included in this thesis.

3.2.1 Formation graph

The idea of associating a graph to a team of robots to model the interactions among robots is quite natural. In such model, each robot of the system can be viewed as a vertex of a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. An edge exists between two vertices if the robots associated to those vertexes interact in some way. Algebraic properties of the matrices associated to the graph $\mathcal{G}$ can be used to analyze the properties of the overall multi-robot system. In particular, in Paper A, we associated to a multi-robot system a graph which represents the local interaction between the robots, and specifically an exchange of information. Examples, in the literature, include deployment and coverage tasks as in [CMB04], or the problem of organizing the robots in formations as in [OSM02].

3.2.2 Hierarchical structure

A hierarchical control structure can be used to model multi-robot systems [DLS95]. Each robot is described by three different layers: an upper layer which is modeled by a discrete-event system, a lower layer which is represented by a continuous-time system and an interface which interconnects the two layers, as shown in Figure 3.5. The upper layer, which we call here team controller, is responsible for generating

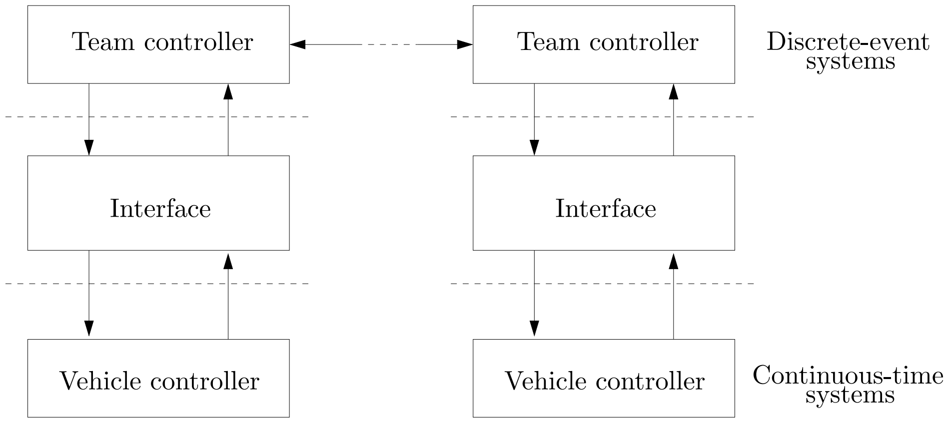


Figure 3.5: Hierarchical controller

waypoints, namely points in the configuration space the robot needs to reach. The waypoints are generated accordingly to a coordination algorithm which is mapped on a discrete-event system and depends on the particular assigned task. The lower layer, or vehicle controller, is responsible to drive the robot from one waypoint to the next producing feasible trajectories. The continuous-time system modeling the dynamics of the robot (for example it could represent a unicycle or a car-like robot) is used to design a suitable controller for trajectory generation and tracking. The interaction between the two layers is done through an interface. When a new waypoint is generated it is passed by the interface to the vehicle controller which computes a trajectory from the current position the desired waypoint. Once the robot reaches such waypoint, or more realistically it is in a neighborhood of it, an event or set of events are triggered by the vehicle controller and the discrete-event system can compute a new waypoint.

The interaction among robots is modeled as interaction among the team controller of each individual. In particular, changes in the structure of the multi-robot system, such as division of the team in subgroups or failures in some of the robots can be modeled as external events that influence the behavior of the team controllers. In Paper B a hierarchical control structure is proposed for a team of underwater vehicles.

3.2.3 Artificial potential functions

Artificial potentials were introduced to robotics for obstacle avoidance and navigation [Kha86, RK92]. They have recently been exploited to derive control laws for autonomous multi-robot systems where convergence proofs to desired configurations are explicitly provided (see for example [McI96, LF01]). The basic idea (cf. [ÖFL04, LF01]) is that each robot is subject to a system of central forces generated by artificial potential fields. In particular there are inter-robot forces that are negative when the relative distance between two robots is less than a fixed threshold and vanishe when the robots are far from each other. Robots are also subject to forces directed towards some other robots which represent virtual leaders of the group. A controlled dissipative force is also applied to each robot and is designed such that it is zero when the robot is moving at the desired speed. Thus a team of robot embedded in such artificial potential fields can be analyzed as a system of virtual forces and a controller is designed based on the inter-robot forces.

In Paper C a probabilistic pursuit–evasion game is considered. Each pursuers create probabilistic maps, which can be considered as an artificial potential field, in order to move in an unknown environment.

Chapter 4

Multi-robot systems with communication constraints

> *"The most important thing in communication is to hear what isn't being said."*
> - P.F. Drucker

A multi-robot system with communication constraints is a special case of distributed control system with subsystems interconnected with digital communication channels. A block diagram showing three robots connected via communication links is shown Figure 4.1. A particular restriction, compared with the general distributed system discussed in previous chapters (see Figure 2.1) is that only controllers exchange data.

The design of controllers for multi-robot systems has been mostly focused on the development of cooperative strategies, where a team of robots accomplish a given task [LF01, FBKT00, HKS99, BA98]. Communication, when present, has been considered as a system property and limitations have seldom been accounted for in the design. As the applications grow in complexity the limitations become more important; for example, when the number of robots is large or when the robots are deployed in particular environments, such as underwater or in space. A general theory for control design of multi-robot systems under communication constraints is not existing and few results are available in the literature. We will present here two interesting frameworks for modeling these type of problems when topological communication constraints are considered, namely constraints on "who can communicate with who".

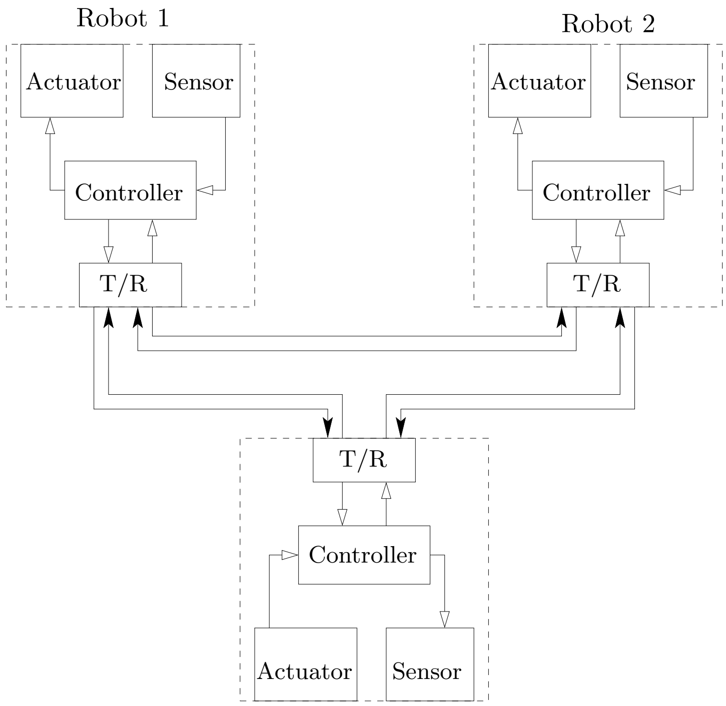


Figure 4.1: Multi-robot system with communication channels depicted. The solid arrows represent wireless communication channels and empty arrows represent the sensor and actuation signals. The T/R blocks represent the transmitters and receivers of the digital communication systems.

4.1 Team decision theory

The problem of designing control strategies for a team of robots when communication constraints are topological can be related to the problem studied in team decision theory [Chu72, HC72, Wit68]. We summarize here the model and some important results developed.

Let us consider a team of n robots. There are five basic ingredients of decision theory.

1. The state of the world $\xi = (\xi_1, \dots, \xi_n) \in \Omega$ is capture by a vector of random variables defined on a suitable probability space. The vector ξ represents all the uncertainties in the problem under consideration, e.g. unknown initial conditions, measurement noise, uncertain parameters, etc.

2. A set of decision variables $u = (u_1, \dots, u_n) \in U$, each representing the decision

of one robot.

3. A measurable function $J(\xi, u)$, called payoff function.

4. A set of information functions $z = \eta(\xi, u) \in Z$, so that $z = (\eta_1(\xi, u), \ldots, \eta_n(\xi, u)$. In other words z_i represents the information known to the robot i. This includes information communicated by other robots. The set $\{\eta_1, \ldots, \eta_n\}$ is known as the information structure.

5. A set of strategies $\gamma = (\gamma_1, \ldots, \gamma_n) \in \Gamma$, where γ_i is a mapping from the z_i-space to the u_i-space. Robot i must choose actions $u_i = \gamma_i(z_i)$ based on his local information, hence the problem is decentralized.

The dynamic team decision problem is

$$\min_{\gamma \in \Gamma} E\big[J(\xi, \gamma(\eta(\xi, u)))\big].$$

The reason why the problem is called dynamic is because $z = \eta(\xi, u)$. If $z = \eta(\xi)$ the problem is said to be static. In the simplest case the information functions are linear in ξ and in the control actions other member have taken, namely

$$z_i = H_i \xi + \sum_j D_{i,j} u_j, \qquad \forall i \in \mathcal{I}.$$

where H_i and $D_{i,j}$ are matrices of appropriate dimensions and are known to each robot. The information structure in this case is defined as the matrices H_i and $D_{i,j}$, with $i, j \in \mathcal{I}$. Roughly speaking the information structure is a formal notion of "who knows what". Notice that no dynamics in the topology is allowed i.e., H_i and $D_{i,j}$ are not time dependent.

Since we interested in modeling causal systems, we assume

$$D_{i,j} \neq 0 \Rightarrow D_{j,i} = 0 \quad \forall i, j \in \mathcal{I}, i \neq j.$$

Thus if the control action of j affects the information of i, then the control action of i cannot affect the information of j i.e., we have in mind here a discrete-time dynamic situation in which the current actions can affect, at most, information in the succeeding, but not current, stage. This means that we can graphically represent the information structure by a precedence diagram. Let us consider an example.

Example 4.1
Consider the multi-robot system of Figure 4.2a. The information structure can be represented by the precedence diagram of Figure 4.2b, showing how the information of each robot influences the information of the others. The information is then given

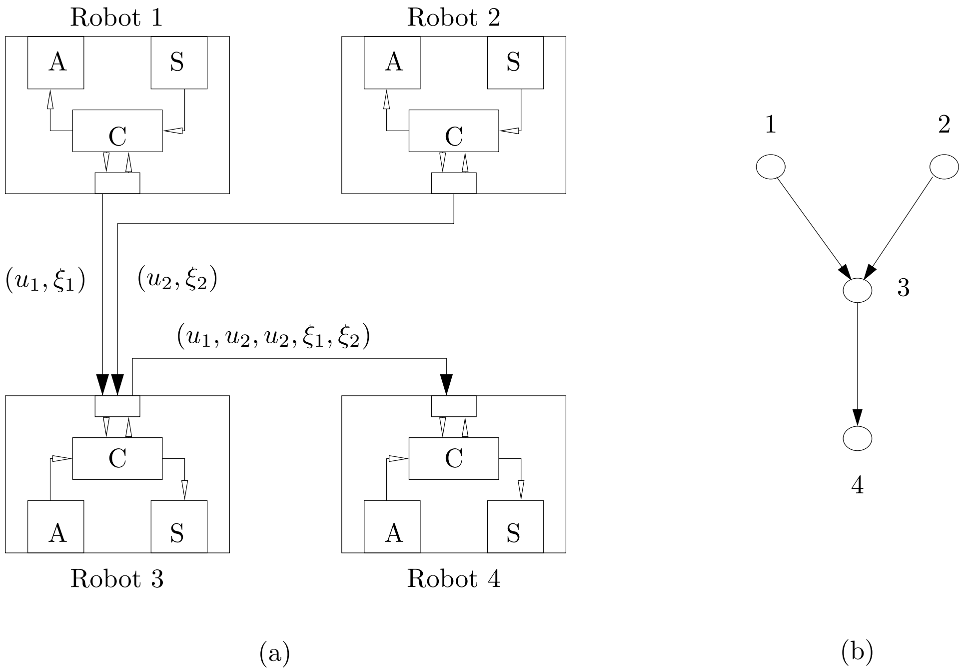


Figure 4.2: Example of a partially nested information structure.

by

$$
\begin{aligned}
z_1 &= H_1 \xi \\
z_2 &= H_2 \xi \\
z_3 &= H_3 \xi + D_{3,1} u_1 + D_{3,2} u_2 \\
z_4 &= \underbrace{\begin{pmatrix} H_4' \\ H_3 \end{pmatrix}}_{H_4} \xi + \underbrace{\begin{pmatrix} 0 \\ D_{3,1} \end{pmatrix}}_{D_{4,1}} u_1 + \underbrace{\begin{pmatrix} 0 \\ D_{3,2} \end{pmatrix}}_{D_{4,2}} u_2 + \underbrace{\begin{pmatrix} D_{4,3}' \\ 0 \end{pmatrix}}_{D_{4,3}} u_3.
\end{aligned}
$$

where $\xi = (\xi_1, \xi_2, \xi_3)$.

The information structure of this example is called partially nested, in which the follower, in the precedence diagram, can always deduce the action of its precedents. An important result in team decision theory is the following: in a dynamic multi-robot system with partially nested information structure, the optimal control for each robot exists, is unique and is linear in z_i.

It is important to point out that for some information structure it is possible to construct a nonlinear controller that achieves lower payoff than the best linear

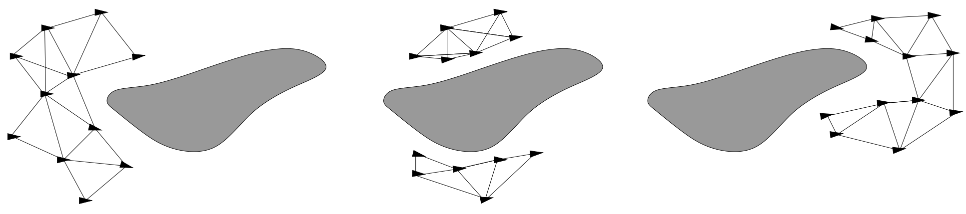


Figure 4.3: Example of $n = 11$ robots in a formation control problem. The heading of the formation is the average of the single robot heading.

one (see [Wit68]). Thus only for specific communication topologies it is possible to compute the control strategies that optimize the payoff.

4.2 Consensus problems

Multi-robot systems where each robot is constrained to communicate only with nearest neighbor robots have been studying since the seminal work of Vicsek et al. [VCBJ$^+$95]. The paper addresses the problem of modeling the behavior of autonomous agents described as points or particles moving on a plane. Each agent's heading is updated considering an average of its own and its neighbor's headings. More complex scenarios and tool for analysis have been developed, see [OSM03, JLM03, TJP03b] and references therein. We will review here some results that can be found in the literature on these so called consensus problems.

Let $n > 1$ be the number of robots in the system, and let $x_i(t)$ be the state information of the ith robot. A continuous-time consensus protocol can be summarized as

$$\dot{x}_i(t) = -\sum_{j \in \mathcal{N}_i(t)} \alpha_{i,j}(t)\big(x_i(t) - x_j(t)\big) \qquad \forall i = 1, \ldots, n,$$

where $\mathcal{N}_i(t)$ represents the set of robots in the neighborhood of the robot i. A design problem for this system corresponds to find the weights $\alpha_{i,j}(t)$ so that the system has particular properties; for example in finding conditions that guarantee convergence to a common steady state x_{ss}, e.g. [OSM03, JLM03]. It is important to notice that in general the set of neighbor robots can vary with time, thus the topology is time varying. Graphs are usually used to model such systems and the algebraic structure of the matrices associated to these graphs allows to analyze the properties of the consensus protocol. Figure 4.3 shows an example where a team of 11 robots is represented as a graph. Edges of the graph represent inter-robot communication. The consensus protocol involves the heading of the robots, namely $x_i(t) = \theta_i(t)$ where $\theta_i(t)$ is the heading of robot i. In Figure 4.3 are shown three different shots of the robots' motion. In particular we can notice how the topology changes over time, the most evident moment is shown in Figure 4.3b when the team splits in two independent groups to avoid an obstacle (shown in gray). Figure 4.3c shows the robots forming again a single group and heading in the same direction.

Let $\{\mathcal{G}_p\}_{p\in\mathcal{P}}$ be all the possible graphs with n vertices, parameterized by the index $p \in \mathcal{P}$ with $\mathcal{P}$ a suitable index set. Let $\sigma : \{1, 2, \dots\} \to \mathcal{P}$ denote a switching signal. Then it is possible to prove that the consensus is achieved, namely

$$\lim_{t\to\infty} \theta_i(t) = \theta_{ss} \quad \forall i = 1, \dots, n,$$

if the graphs $\{\mathcal{G}_{\sigma(t)}\}$ are connected most of the time [JLM03]. Moreover in [Jad03] it is proved that a necessary and sufficient condition for the robots to converge to a steady state is that the switching of topologies converges in finite time to a connected topology, and no more switches occur. Namely $\exists T > 0$ such that

$$\mathcal{G}_{\sigma(T)} = \mathcal{G}_M, \qquad M \in \mathcal{P}$$

with $\mathcal{G}_M$ connected and $\sigma(t) = M$ for all $t \geqslant T$. The main limitation of these results, at this point of the development, is the lack of general existence conditions i.e., given a consensus problem we do not know if there exists a time $T > 0$ such that the topologies will be connected or if the topologies are connected most of the time.

Consensus problem have been used in literature to solve different types of problems, e.g. formation [GHM03, FM02], rendezvous [LMA03] and flocking [TJP03a, TJP03b] problems.

4.3 Towards a theory for multi-robot systems

The development of systematic methods for the analysis and design of controllers for multi-robot systems under communication constraint has become a very important research area since applications have become more demanding, both from the complexity of the tasks that have to be solve and the amount of data exchanged by the robots. Team decision theory and consensus problems are two possible approaches for studying multi-robot systems under communication constraints. However, the limitations imposed by the communication system, in both approaches, are restricted to topological constraints. In Chapter 2 we have listed other important limitations the communication channel imposes such as time delays, loss of data and quantization. Towards the development of general tools for multi-robot systems, in the papers included in the thesis we have studied the effects of some of these limitations to specific multi-robot tasks.

References

[BA98] T. Balch and R C. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Transaction on Robotics and Automation*, 14:926–939, 1998.

[BL00] R.W. Brockett and D. Liberzon. Quantized feedback stabilization of linear systems. *IEEE Transaction on Automatic Control*, 45(7):1279–1289, 2000.

[Chu72] K.C. Chu. Team decision theory and information structures in optimal control problems - part 2. *IEEE Transaction on Automatic Control*, 17(17):22–28, February 1972.

[CMB04] J. Cortes, S. Martinez, and F. Bullo. Spatially-distributed coverage optimization and control with limited-range interactions. *ESAIM: Control, Optimisation, and Calculus of Variations*, 2004. Submitted.

[CT91] T.M. Cover and J.A. Thomas. *Elements of information theory.* Wiley Interscience, 1991.

[Del90] D.F. Delchamps. Stabilizing a liner system with quantized state feedback. *IEEE Transaction on Automatic Control*, 35:916–924, 1990.

[DGR00] B.R. Donald, L. Gariepy, and D. Rus. Distributed manipulation of multiple objects using ropes. In *IEEE International Conference on Robotics and Automation*, pages 450–457, 2000.

[DLS95] N.G. Datta, J. Lygeros, and S. Sastry. *Hierarchical hybrid control: a case study"*, pages 166–190. Lecture Notes in Computer Science, Springer, 1995.

[EM01] N. Elia and S. J. Mitter. Stabilization of linear systems with limited information. *IEEE Transaction on Automatic and Control*, 46(9):1384–1400, 2001.

[FBKT00] D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A probabilistic approach to collaborative multi-robot localization. *Special issue of Autonomous Robots on Heterogeneous Multi-Robot Systems*, 3, 2000.

[FM02] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicles formations. In *IFAC World Congress*, 2002.

[FZ03] F. Fagnani and S. Zampieri. Stability analysis and synthesis for scalar linear systems with quantized feedback. *IEEE Transaction on Automatic and Control*, 48(9):1569–1584, 2003.

[GC01] K. Goldberg and B. Chen. Collaborative control of robot motion: Robustness to error. In *IEEE/RSJ International Conference on Robots and Systems*, 2001.

[GHM03] V. Gupta, B. Hassibi, and R. Murray. Stability analysis of stochastic varying formation of dynamic agents. In *IEEE Conference on Decision and Control*, 2003.

[HC72] Y. Ho and K.C. Chu. Team decision theory and information structures in optimal control problems - part 1. *IEEE Transaction on Automatic Control*, 17(1):15–22, February 1972.

[HKS99] J.P. Hespanha, H. J. Kim, and S. Sastry. Multiple-agent probabilistic pursuit–evasion games. In *IEEE Conference on Decision and Control*, volume 3, pages 2432–2437, 1999.

[Jad03] A. Jadbabaie. On coordination strategies for mobile agents with changing nearest neighbor sets. In *IEEE Mediterranian Conference on Control and Automation*, 2003.

[JLM03] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[Kha86] O. Khatib. Real time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5:90–99, 1986.

[KYC+96] O. Khatib, K. Yokoi, K. Chang, R. Holmberg, and A. Casal. Vehicle/arm coordination and mobile manipulator decentralized cooperation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 546–553, 1996.

[LF01] N.E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. In *IEEE Conference on Decision and Control*, pages 2968–2973, 2001.

[LMA03] J. Lin, A. S. Morse, and B. D. O. Anderson. The multi-agent rendezvous problem. In *Proc. of the 42nd IEEE Conference on Decision and Control, HI, USA*, 2003.

[McI96] C.R. McInnes. Potential function methods for autonomous spacecraft guidance and control. *Advances in Astronautical Sciences*, pages 2093–2109, 1996.

[Mit01] S.J. Mitter. Control with limited information. *European Journal of Control*, 7(1), 2001.

[MSJH04] M. Mazo, A. Speranzon, K.H. Johansson, and X. Hu. Multi-robot tracking of a moving object using directional sensors. In *IEEE International Conference on Robotics and Automation*, 2004.

[Nil98] J. Nilsson. *Real-time control systems with delays*. PhD thesis, Lund Institute of Technology, 1998.

[ÖFL04] P. Ögren, E. Fiorelli, and N.E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transaction on Automatic and Control*, 2004.

[OSM02] R. Olfati-Saber and R.M. Murray. Graph rigidity and distributed formation stabilization of multi-vehicle systems. In *IEEE Conference on Decision and Control*, 2002.

[OSM03] R. Olfati Saber and M. Murray. Flocking with obstacle avoidance: cooperation with limited communication in mobile networks. In *IEEE Conference on Decision and Control*, 2003.

[Par00] L. Parker. Current state of the art in distributed robot systems. In Berhen J. Parker L., Bekey G., editor, *Distributed Autonomous Robotics Systems 4*. Springer, 2000.

[PB03] B. Picasso and A. Bicchi. Stabilization of LTI systems with quantized state-quantized input static feedback. In A. Pnueli and O. Maler, editors, *Hybrid Systems: Computation and Control*, volume LNCS 2623 of *Lecture Notes in Computer Science*. Springer-Verlag, 2003.

[PDF+03] E. Pagello, A. D'Angelo, C. Ferrari, R. Polesel, R. Rosati, and A. Speranzon. Emergent behaviors of a robot team performing cooperative tasks. *Advanced Robotics*, 15(1):3–20, 2003.

[RK92] E. Rimon and D.E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8:501–508, 1992.

[Sha48] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423,623–656, 1948.

[SVS04] J. Sousa, P. Varaiya, and T. Simsek. Distributed control of teams of unmanned air vehicles. In *Mathematical Theory of Networks and Systems*, 2004.

[TJP03a] H.G. Tanner, A. Jadbabaie, and G.J. Pappas. Stable flocking of mobile agents, part i: fixed topology. In *IEEE Conference on decision and control*, 2003.

[TJP03b] H.G. Tanner, A. Jadbabaie, and G.J. Pappas. Stable flocking of mobile agents, part ii: dynamic topology. In *IEEE Conference on decision and control*, 2003.

[VCBJ+95] T Vicsek, A. Czirók, Eshel Ben-Jaco, I. Cohen, and O. Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75:1226–1229, 1995.

[Wit68] H.S. Witsenhausen. A counterexample in stochastic optimum control. *SIAM Journal of Control*, 6(1):131–147, 1968.

[XHH00] L. Xiao, A. Hassibi, and J.P. How. Control with random communication delays via a discrete-time jump system approach. In *American Control Conference*, 2000.

Part II

Papers

Paper A

On Multi-Vehicle Rendezvous Under Quantized Communication

F. Fagnani, K.H. Johansson, A. Speranzon, S. Zampieri

Abstract

A rendezvous problem for a team of autonomous vehicles, which communicate over quantized channels, is analyzed. The paper illustrates how communication topologies based on uniform and logarithmic quantization influence the performance. Since a logarithmic quantizer in general imposes fewer bits to be communicated compared to a uniform quantizer, the results indicate estimates of lower limits on the amount of information that needs to be exchanged in order for the vehicles to meet. Simulation examples illustrate the results.

***Keyworkds*:** Quantized communication, Rendezvous, Multi-vehicle system.

5.1 Introduction

Interplay between coordination and communication is important in many multi-vehicle systems, e.g., car platoons on automated highways [Var93], formations of autonomous underwater vehicles [dSP02], and multi-robot search-and-rescue missions
[SJ03]. Constrained communication between vehicles suggest the deployment of distributed (local) control strategies [LMA03, SM03]. In many cases not only the communication topology is important, however, but also the amount of data being transmitted. Therefore, in this paper we study multi-vehicle control under quantized communication. The problem is related to the stabilization of linear plants with quantized control, which has recently been extensively studied, see [FZ03] and references therein.

The main contribution of this paper is to illustrates how communication topologies based on uniformly and logarithmically quantized communication influence the solution to a multi-vehicle rendezvous problem. A team of autonomous vehicles with only local position information is to meet under minimum communication capabilities. We prove the existence of several classes of solutions to this rendezvous problem. In particular, we emphasize that uniform quantizers can sometimes be replaced by logarithmic quantizers and thus reduce the need for communication bandwidth.

The outline of the paper is as follows. The rendezvous problem is defined in Section 5.2 together with feasible communication topologies. A few illustrative two-vehicle cases are studied in detail in Section 5.3. Teams of three and more vehicles are then considered in Section 5.4. Convergence properties are investigated through simulations in Section 5.5. Some conclusions are given in Section 5.6.

5.2 Problem formulation

Consider $n \geqslant 2$ vehicles moving in a plane, with dynamics described by the discrete-time system

$$\mathbf{x}^+ = \mathbf{x} + \mathbf{u} \tag{5.1}$$

$$\mathbf{y}^+ = \mathbf{y} + \mathbf{v} \tag{5.2}$$

where $\mathbf{x} = (x_1, \ldots, x_n)^T \in \mathcal{X} \subset \mathbb{R}^n$ and $\mathbf{y} = (y_1, \ldots, y_n)^T \in \mathcal{Y} \subset \mathbb{R}^n$, so that (x_i, y_i) denotes the position of vehicle i with respect to a fixed coordinate system. Let $\mathcal{U} \subset \mathbb{R}^n$ and $\mathcal{V} \subset \mathbb{R}^n$ denote the set of control values. The controls $\mathbf{u} = (u_1, \ldots, u_n)$ and $\mathbf{v} = (v_1, \ldots, v_n)$ we are considering are feedback maps from the corresponding state-space $\mathcal{X}$ and $\mathcal{Y}$, respectively. Since the control of the x- and y-coordinates are independent, we only consider x in the sequel.

5.2.1 Control and communication topology

The control of each vehicle depends on its own state and the state information communicated from other vehicles. Hence,

$$u_i = g_i(x_i, \mathbf{c}_i), \quad i = 1, \ldots, n,$$

where $g_i : \mathcal{X}_i \times C_i \to U_i$ is the control map and C_i the value set of the communication variable $\mathbf{c}_i$. A communication topology describes what information is transmitted to which vehicle.

Definition 5.1 *A communication topology is a map*

$$\Psi : \mathcal{X} \times C \to C$$

such that $\Psi = (\psi_1, \ldots, \psi_n)^T$ *with*

$$\psi_i : \prod_{j \neq i} \mathcal{X}_j \times \prod_{j \neq i} C_j \to C_i$$

describing how states and communication variables from the other vehicles are transmitted to vehicle i*.*

The feedback map of vehicle i is thus based on x_i and $\mathbf{c}_i = \psi_i(\{x_j\}_{j \neq i}, \{\mathbf{c}_j\}_{j \neq i})$. The transmission is supposed to be instantaneous. A consequence of this assumption is that the vehicle model together with the communication topology can be ill-posed, as illustrated with the following example.

Example 5.2
Consider a two-vehicle system where the vehicles communicate their absolute position, i.e.,

$$\Psi(\mathbf{x}, \mathbf{c}) = \begin{pmatrix} \psi_1(x_2, c_2) \\ \psi_2(x_1, c_1) \end{pmatrix} = \begin{pmatrix} x_2 \\ x_1 \end{pmatrix}. \tag{5.3}$$

At time k*, the vecicles hence communicate*

$$\Psi(\mathbf{x}(k), \mathbf{c}(k)) = \begin{pmatrix} x_2(k) \\ x_1(k) \end{pmatrix},$$

which simply corresponds to their position data.

Consider now instead a two-vehicle system where the vehicles communicate their relative positions, i.e.,

$$\Psi(\mathbf{x}, \mathbf{c}) = \begin{pmatrix} \psi_1(x_2, c_2) \\ \psi_2(x_1, c_1) \end{pmatrix} = \begin{pmatrix} x_2 - x_1 \\ x_1 - x_2 \end{pmatrix}. \tag{5.4}$$

At time k, the vehicles should then communicate

$$\Psi(\mathbf{x}(k), \mathbf{c}(k)) = \begin{pmatrix} x_2(k) - x_1(k) \\ x_1(k) - x_2(k) \end{pmatrix},$$

which is not possible. This fact can be easily seen from that at time $k = 0$, each vehicle know only their initial position $x_1(0) = x_{10}$ and $x_2(0) = x_{20}$, respectively, so its impossible to communicate the differences

$$\Psi(\mathbf{x}(0), \mathbf{c}(0)) = \begin{pmatrix} x_2(0) - x_1(0) \\ x_1(0) - x_2(0) \end{pmatrix}.$$

Hence, the communication topology with relative positions is not feasible in the sense that it cannot be realized.

We consider only topologies that are physically realizable, i.e., feasible communication topologies according to the following definition.

Definition 5.2 *A communication topology is feasible if there exists a surjective map $f : \mathcal{X} \to C$ such that*

$$\mathbf{c} = \Psi(\mathbf{x}, \mathbf{c}) \quad \Rightarrow \quad \mathbf{c} = f(\mathbf{x}).$$

The notion of feasibility for a communication topology given here is related to the concept of deadlock-freeness introduced in [AT92] for decentralized stochastic systems.

Example 2 (cont'd). It easy to see that (5.3) yields a feasible communication topology, while (5.4) is unfeasible.

5.2.2 Quantized communication

We further restrict the communication by imposing that communicated data are quantized. In particular, uniform and logarithmic quantizations are considered. Recall the following definitions of scalar uniform and logarithmic quantizers.

Definition 5.3 *Let δ be a positive parameter. A uniform quantizer is a map $q_u : \mathbb{R} \to \mathbb{R}$ such that*

$$q_u(x) = \delta \left\lfloor \frac{x}{\delta} \right\rfloor.$$

Notice that the error due to the quantization of a variable x is bounded by δ i.e.,

$$|q_u(x) - x| \leqslant \delta. \tag{5.5}$$

Definition 5.4 *Let δ be a positive parameter. A logarithmic quantizer is a map $q_\ell : \mathbb{R} \to \mathbb{R}$ such that*

$$q_\ell(x) = \exp(q_u(\ln x)).$$

The quantization error for a logarithmic quantizer is bounded as

$$|q_l(x) - x| \leqslant \delta |x|. \tag{5.6}$$

We are interested in how communication topology and quantization of transmitted data influence the performance of the multi-vehicle system. Therefore, we consider classes of maps Ψ_δ composed of various configurations of uniform and logarithmic quantizers. For simplicity, we suppose that all quantizers are parameterized in a single quantization parameter δ.

Example 5.3
For a two-vehicle system, a uniformly quantized communication topology is given by

$$\Psi_\delta(\mathbf{x}, \mathbf{c}) = \begin{pmatrix} \psi_1(x_2, c_2) \\ \psi_2(x_1, c_1) \end{pmatrix} = \begin{pmatrix} q_u(x_2) \\ q_u(x_1) \end{pmatrix}.$$

5.2.3 Rendezvous

We are interested in the convergence to a multi-vehicle formation under quantized communication topology. Especially, we pose the following rendezvous problem.

Definition 5.5 *A feedback map $g = (g_1, \ldots, g_n)$ and a communication topology Ψ_δ solve the rendezvous problem if for all initial states $\mathbf{x}_0 \in \mathcal{X}$, and for all $\epsilon > 0$, there exists $\delta = \delta(\epsilon)$ such that*

$$x_i - x_j \to \mathcal{B}_\epsilon$$

with $i < j$ and $i, j = 1, \ldots, n$.

A solution to the rendezvous problem is able to make the vehicles converge arbitrarily close to a meeting point. Note that the meeting point is not pre-specified, but is only restricted to be close to the hyperplane $x_1 = \cdots = x_n$.

If the communication is not quantized, then the rendezvous problem is readily solved also in the case with no communication, i.e., when Ψ is empty. In this case, a decentralized deadbeat controller provides a solution. We consider such a solution trivial, since it enforces the vehicles to meet in the origin. It is instead desirable to have the meeting point close to the initial position of the vehicles. In next section, we present a linear quadratic control problem for the two-vehicle system that address this problem and that also suggest a natural extension to the quantized case.

5.3 Two-vehicles rendezvous

Let the communication topology of a two-vehicle system be

$$\Psi_0(\mathbf{x}, \mathbf{c}) = \begin{pmatrix} \psi_1(x_1, c_1) \\ \psi_1(x_2, c_2) \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

Let $z = x_1 - x_2$ be the output of the system (5.1) when $n = 2$. In order to avoid aggressive solution such as the dead-beat control we need to penalize the control input. Thus we consider an LQ problem with cost

$$\begin{aligned} J(u_1, u_2) &= \sum_{t=0}^{\infty} z^2 + u_1^2 + u_2^2 \\ &= \sum_{t=0}^{\infty} (x_1 + x_2)^2 + u_1^2 + u_2^2. \end{aligned}$$

The resulting state feedback $\mathbf{u} = \mathbf{K}\mathbf{x}$ gives the closed-loop system

$$\begin{aligned} x_1^+ &= x_1 - k(x_1 - x_2) \\ x_2^+ &= x_2 - k(x_2 - x_1), \end{aligned} \tag{5.7}$$

with $k = 1/(1 + \sqrt{3})$. It corresponds to

$$z^+ = (1 - 2k)z,$$

which is asymptotically stable, meaning that the difference $x_1 - x_2$ tends to zero asymptotically. Note that (5.7) it is not asymptotically stabile. The two vehicles, in general, will thus rendezvous at a point different from the origin.

The linear feedback $\mathbf{K}$ computed above is used to design controllers when the communication topology is composed of various configurations of uniform and logarithmic quantizers.

5.3.1 Uniform-uniform quantization

Proposition 5.1 *The feasible communication topology*

$$\Psi_\delta = \begin{pmatrix} \psi_1(x_2, c_2) \\ \psi_2(x_1, c_1) \end{pmatrix} = \begin{pmatrix} q_u(x_2) \\ q_u(x_1) \end{pmatrix} \tag{5.8}$$

and the feedback

$$\begin{aligned} u_1 &= g_1(x_1, c_1) = -k(x_1 - c_1) \\ u_2 &= g_2(x_2, c_2) = -k(x_2 - c_2) \end{aligned} \tag{5.9}$$

solve the rendezvous problem

Proof. Let us consider the difference $z = x_1 - x_2$. We introduce the following Lyapunov function $\mathcal{V}(z) = |z|$. Thus the increment $\Delta\mathcal{V}(z) = \mathcal{V}(z^+) - \mathcal{V}(z) = |z^+| - |z|$ is such that

$$\begin{aligned} \Delta\mathcal{V}(z) &= |z - 2kz - c_1 + c_2| - |z| \\ &\leqslant -2k|z| + 2k\delta \end{aligned}$$

where we have substituted $c_1 = \psi_1$ and $c_2 = \psi_2$ given in (5.8) and used (5.5). Hence $\Delta\mathcal{V}(z) < 0$ if $|z| > \delta$. If we choose $\delta = \epsilon$, then z will asymptotically converge to $\mathcal{B}_\epsilon$. Thus the communication topology (5.8) and feedback (5.9) solves the rendezvous problem. $\square$

Notice that instead of posing a deterministic LQ problem, we can consider a stochastic LQ problem where the quantization error of the uniform quantizer is approximated by additive white noise, namely

$$q_u(x) = x + e$$

and where e is white noise uniformly distributed in $[-\delta, \delta]$ and $Ee^2 = \delta^2/12$. This approach yields the same feedback control as above.

5.3.2 Uniform-logarithmic quantization

Proposition 5.2 *The feasible communication topology*

$$\Psi_\delta = \begin{pmatrix} \psi_1(x_2, c_2) \\ \psi_2(x_1, c_1) \end{pmatrix} = \begin{pmatrix} q_\ell(q_u(x_1) - x_2) \\ q_u(x_1) \end{pmatrix} \tag{5.10}$$

and the feedback

$$\begin{aligned} u_1 &= g_1(x_1, c_1) = -k\, c_1 \\ u_2 &= g_2(x_2, c_2) = -k(x_2 - c_2) \end{aligned} \tag{5.11}$$

solve the rendezvous problem

Proof. Let us consider the difference $z = x_1 - x_2$. As in the proof of Proposition 5.1,

$$\begin{aligned} \Delta\mathcal{V}(z) &= |z - k(c_1 - x_2 + c_2)| - |z| \\ &\leqslant -2k|z| + k|z|\delta + 2k\delta + k\delta^2 \end{aligned}$$

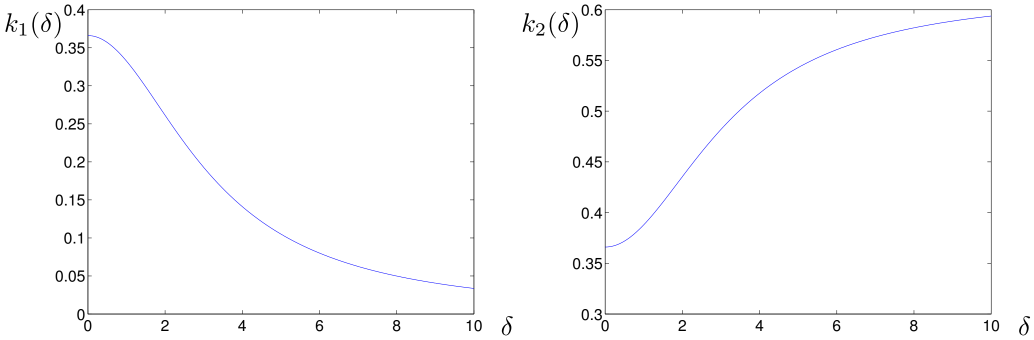
where c_1 and c_2 are given by the communication topology map Ψ_δ defined in (5.10) and where we used (5.6). $\Delta\mathcal{V}(z) < 0$ if

$$|z| > \frac{2\delta(1+\delta)}{2-\delta}.$$

If we choose $\delta = -1/2 - \epsilon/4 - \sqrt{4 + 20\epsilon + \epsilon^2}/4$, z converges asymptotically to $\mathcal{B}_\epsilon$. Thus the communication topology (5.10) and feedback (5.11) solves the rendezvous problem. $\square$

The effect of the logarithmic quantization can be approximate as multiplicative noise acting on the system. If, for simplicity, we neglect the presence of uniform quantization, then we have

$$\begin{aligned} x_1^+ &= x_1 + k_{11}(x_1 - x_2)(1 + e) \\ x_2^+ &= x_2 + k_{22}(x_2 - x_1) \end{aligned}$$



(a) $k_1(\delta)$ as function of δ

(b) $k_2(\delta)$ as function of δ

Figure 5.4: For large quantization step δ the value of $k_1(\delta)$ tends to zero.

where e is white noise uniformly distributed in $[-\delta, \delta]$ and variance $Ee^2(t) = \delta^2/12$. Let $z = x_1 - x_2$, we consider an optimal control problem with cost

$$\begin{aligned} J(u_1, u_2) &= E \sum_{t=0}^{\infty} z^2 + u_1^2(1+e) + u_2^2 \\ &= E \sum_{t=0}^{\infty} z^2 + (u_1, u_2) \underbrace{\begin{pmatrix} 1 + \frac{\delta^2}{12} & 0 \\ 0 & 1 \end{pmatrix}}_{R} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \end{aligned}$$

and dynamics

$$z^+ = z + \underbrace{(11)}_{B} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + u_1 e.$$

The feedback law is linear and given by

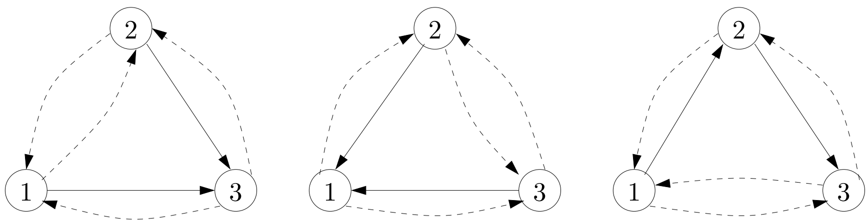
$$K = (R + B^T P B + \Omega(P))^{-1} B^T P$$

where P is the solution of a generalized Riccati difference equation and $\Omega(P)$ is the following matrix

$$\Omega(P) = \begin{pmatrix} 1 + \frac{\delta^2}{12} & 0 \\ 0 & 0 \end{pmatrix},$$

see [BD98]. The optimal feedback is

$$\begin{aligned} k_1(\delta) &= k + a\delta^2 + O(\delta^4) \\ k_2(\delta) &= k + b\delta^2 + O(\delta^4) \end{aligned}$$



Topology 1 Topology 2 Topology 3

Figure 5.5: Three different communication topologies for $n = 3$. Solid lines denote uniformly quantized communication channels, and dashed lines logarithmically quantized communication channels.

where k is the same value computed solving the deterministic optimal control problem and a, b are two real values an order of magnitude less than k. If we plot $k_1(\delta)$ and $k_2(\delta)$ we observe that for large value of δ the gain $k_1(\delta)$ tends to zero, see Figure 5.4(a), meaning that the vehicle first vehicle, whose input depends on the logarithmically quantized value of the relative distance $x_1 - x2$, does not move since $x_1 - x_2$ is known with very large error. On the other hand the second vehicle, which rely on a perfect knowledge of the position of the first, applies a large input in order to move towards it, as shown in Figure 5.4(b).

5.4 n-vehicles rendezvous

For any pair of vehicles (i, j), $i < j$ we define an output variable $w_{i,j} = x_i - x_j$. Let $\mathbf{w}$ be a vector collecting all subset of such output variables. Similar to the two vehicles case we consider an optimal control problem with cost function

$$\begin{aligned} J(\mathbf{u}) &= \sum_{t=1}^{\infty} \mathbf{w}^T \mathbf{w} + \mathbf{u}^T R \mathbf{u} \\ &= \sum_{t=1}^{\infty} \mathbf{x}^T W^T W \mathbf{x} + \mathbf{u}^T R \mathbf{u}. \end{aligned} \tag{5.12}$$

Note that the matrix $W^T W$ is singular, thus the Riccati equation associated with the optimal control problem does not admit a unique solution. It is anyway possible to regularize the problem; consider any subset $\mathbf{z} = \{z_{i,j}\}$ of $\mathbf{w}$. Let $Z \in \mathbb{R}^{(n-1)\times n}$ such that $\mathbf{z} = Z\mathbf{x}$. Then there exists a matrix L such that

$$\mathbf{w} = \begin{pmatrix} I \\ L \end{pmatrix} \mathbf{z}.$$

Minimizing the cost function (5.12) subject to the dynamics (5.1) is equivalent to minimizing the cost

$$J(\mathbf{u}) = \sum_{t=1}^{\infty} \mathbf{z}^T (I + L^T L)\mathbf{z} + \mathbf{u}^T R\mathbf{u} \tag{5.13}$$

subject to dynamics $\mathbf{z}^+ = \mathbf{z} + Z\mathbf{u}$.

Proposition 5.3 *The state feedback that minimizes the cost* (5.13) *when* $R = rI$ *is* $\mathbf{u} = K\mathbf{z}$ *with*

$$K = -k(nI - \mathbb{1})Z^T(ZZ^T)^{-1}, \qquad k = \frac{n + \sqrt{n^2 + 4nr}}{n(2r + n + \sqrt{n^2 + 4nr})}. \tag{5.14}$$

where $\mathbb{1}$ *is the* $n \times n$ *unit matrix.*

Proof. See appendix. □

The matrix W can be interpreted as the incident matrix of a complete digraph $\mathcal{G} = (V, E)$, with $\mathrm{card}(V) = n$, where edges between vertices represent communication between vehicles. The matrix Z is the incident matrix of a directed tree in the graph $\mathcal{G}$. Each pair of edges (i, j) and (j, i), $i < j$ represents a quantized communication channel between vehicle i and vehicle j.

5.4.1 Uniform quantization topology

Proposition 5.4 *The communication topology*

$$c_i = \psi_1\big(\{x_j\}_{j \neq i}, \{\mathbf{c}_j\}_{j \neq i}\big) = \big(\{q_u(x_j)\}_{j \neq i}\big).$$

and feedback

$$u_i = -k(n-1)x_i + k \sum_{j=1\, j \neq i}^{n} q_u(x_j),$$

with k as in (5.14), solves the rendezvous problem.

Proof. Similar to the proof of Proposition 8. □

5.4.2 Uniform-logarithm quantization topologies

Since the logarithmic quantized channels are more efficient, because less bits need to be transmitted compared with uniform quantized channels, we would like to have a communication topology with as many link as possible. Let $n = 3$, we consider topologies where the digraph $\mathcal{G}$ has a tree representing uniformly quantized channels and the remaining edges representing logarithmically quantized channels. Since we have tree vehicles the number of possible directed trees, up to a re-labeling of the vertices are three as shown in Figure 5.5.

Proposition 5.5 *The communication topologies*

$$
\begin{aligned}
c_1 &= \big(q_\ell(c_{3,1}-x_3), q_\ell(c_{3,2}-x_3)\big)\\
c_2 &= \big(\big(q_\ell(c_{3,1}-x_3), q_\ell(c_{3,2}-x_3)\big)\\
c_3 &= \big(q_u(x_1), q_u(x_2)\big)
\end{aligned}
\tag{5.15}
$$

$$
\begin{aligned}
c_1 &= \big(q_\ell(c_{3,1}-x_3), q_\ell(c_{3,2}-x_3)\big)\\
c_2 &= \big(\big(q_\ell(c_{3,1}-x_3), q_\ell(c_{3,2}-x_3)\big)\\
c_3 &= \big(q_u(x_1), q_u(x_2)\big)
\end{aligned}
$$

$$
\begin{aligned}
c_1 &= \big(q_\ell(c_{2,1}-x_2), q_\ell(c_{3,2}-x_3)\big)\\
c_2 &= \big(\big(q_u(x_1), q_\ell(c_{3,2}-x_3)\big)\\
c_3 &= \big(q_\ell(c_{2,1}-x_2), q_u(x_2)\big).
\end{aligned}
$$

and the feedback control laws

$$
u_i(t) = -k_{i,1}c_{i,1} - k_{i,2}c_{i,2} \qquad \forall i = 1,\dots,3 \tag{5.16}
$$

with $k_{i,j}$*,* $i = 1,2,3$ *and* $j = 1,2$ *elements of the matrix* K *defined in* (5.14) *with* $n = 3$*, solve the rendezvous problem.*

Proof. Assume the communication topology is the first one (see Figure 5.5). We prove the statement for such case, the other can be proved in a similar way. Let $R = I$ then for this case

$$
K = k\begin{pmatrix} 2 & -1\\ -1 & 2\\ -1 & -1\end{pmatrix}
$$

We introduce the Lyapunov function $\mathcal{V}(z_{i,j}) = |z_{i,j}|$, with $z_{i,j} \in \{z_{1,3}, z_{2,3}\}$ Let $\Delta\mathcal{V}(z_{i,j}) = \mathcal{V}(z_{i,j}^+) - \mathcal{V}(z_{i,j})$, then we have

$$
\begin{aligned}
\Delta\mathcal{V}(z_{1,3}) &\leqslant -3k|z_{1,3}| + 2k\delta^2 + 2k\delta|z_{1,3}| + k\delta^2\\
&\quad + k\delta|z_{2,3}| + 3k\delta\\
\Delta\mathcal{V}(z_{2,3}) &\leqslant -3k|z_{2,3}| + 2k\delta^2 + 2k\delta|z_{2,3}| + k\delta^2\\
&\quad + k\delta|z_{1,3}| + 3k\delta.
\end{aligned}
$$

Thus we have $\Delta\mathcal{V}(z_{1,3}) < 0$ and $\Delta\mathcal{V}(z_{2,3}) <$ if

$$
\begin{aligned}
|z_{1,3}| &> \frac{3\delta(1+\delta) + \delta|z_{2,3}|}{3-2\delta}\\
|z_{2,3}| &> \frac{3\delta(1+\delta) + \delta|z_{1,3}|}{3-2\delta}.
\end{aligned}
$$

For any $\epsilon > 0$ there exists $\delta = -1/2 - \epsilon/2 + \sqrt{1 + 6\epsilon + \epsilon^2}/2$ such that $z_{1,3}$ and $z_{2,3}$ asymptotically converges to the ball $\mathcal{B}_\epsilon$. Thus the communication topology (5.15) and the feedback (5.16) solve the rendezvous problem. $\square$

5.5 Simulation results

A simulation study have been done for the three-vehicle case. In Figures 5.6(a)-5.6(c) with solid lines are shown the trajectories of the three vehicles for the three different communication topologies of Figure 5.5. In dashed line are shown the trajectories when the communication is without quantization (perfect). As we can notice the three vehicles rendezvous, but trajectories are very different depending on the topology. If we consider the time evolution of $z_{1,2} = x_1 - x_2$ and $z_{2,3} = x_2 - x_3$ and shown in Figure (5.7(a)) (in Figure (5.7(b)) are shown similar time evolutions for the relative distances in the y-coordinate), we can notice that vehicles communicating using the topology 1 (cf., Figure 5.5) rendezvous slowly than when communicating using the other two topologies. This behavior can be explained considering that the third vehicle knows with higher accuracy the position of the two vehicles V_1, V_2 while these two last have a very rough information of their relative distance to the vehicle V_3, due to the logarithmically quantized communication. This results in slower performance, compare to the remaining topologies.

5.6 Conclusions

In this paper we have considered the "multi-vehicle" rendezvous problem under quantized communication topologies. In particular results have been derived for two and three vehicles systems for different topologies and various configurations of uniform and logarithmic quantizers. Some simulation results showing the behavior of the different topologies have been studied in order to verify the results. The trajectories followed by the vehicles seem to depend upon the communication topology used.

Appendix

Proof Proposition 5.3

Let L be the matrix such that

$$W = \begin{pmatrix} I \\ L \end{pmatrix} Z$$

The matrix W^T is the incidence matrix of the complete oriented graph $\mathcal{G}$. Since the graph is complete, there is an edge between any vertex i and j with $i \neq j$ and the degree[1] of each vertex is $n-1$ where n is number of vertices. The matrix $W^T W$ is the Laplacian matrix of the complete graph $\mathcal{G}$, as defined for example in [GR01]. The Laplacian matrix $W^T W$, since the graph is complete, is such that

$$(W^T W)_{ij} = \begin{cases} n-1 = \deg(v_i) & \text{if } i = j \\ -1 & \text{otherwise.} \end{cases}$$

[1]The degree of a vertex of an oriented graph is the total number of edges arriving and departing from that vertex

Thus we can write the Laplacian as $W^T W = nI - \mathbb{1}$. Notice that

$$Z^T(I + L^T L)Z = W^T W = nI - \mathbb{1}.$$

We now prove show that we can find $\alpha > 0$ such that $S = \alpha(I + L^T L)$, is a solution of the Riccati equation associated to the optimal control problem with cost (5.13) and dynamics $\mathbf{z}^+ = \mathbf{z} + Z\mathbf{u}$. The Riccati equation is

$$SZ(R + Z^T SZ)^{-1} Z^T S = I + L^T L$$

If we substitute we get that

$$\alpha n(I + L^T L)Z(R + \alpha Z^T n(I + L^T L)Z)^{-1} Z^T \alpha n(I + L^T L) = I + L^T L.$$

Multiplying on the right and the left the previous expression with Z^T and Z respectively and since $Z^T(I + L^T L)Z = nI - \mathbb{1}$, we get

$$\alpha(nI - \mathbb{1}) \underbrace{\big(rI + \alpha(nI - \mathbb{1})\big)^{-1}}_{\Gamma} \alpha(nI - \mathbb{1}) = nI - \mathbb{1}. \tag{5.17}$$

Notice that Γ can be written as

$$\Gamma = \big((r + \alpha\, n)I - \alpha\mathbb{1}\big)^{-1} = \frac{1}{r + \alpha\, n}\left(I + \sum_{i=1}^{\infty}\left(\frac{\alpha\, n}{r + \alpha\, n}\right)^i \mathbb{1}^i\right).$$

Since $\mathbb{1}^i = n^{i-1}\mathbb{1}$, computing the sum of the geometric series we have that

$$\Gamma = \gamma(\alpha)\big(I + \beta(\alpha)\mathbb{1}\big)$$

with

$$\gamma(\alpha) = \frac{1}{r + \alpha\, n}, \quad \beta(\alpha) = \frac{\alpha\, n}{(-n^2 + n)\,\alpha + r}.$$

The Riccati equation (5.17) becomes

$$\alpha^2\gamma(\alpha)n\big(nI - \mathbb{1}\big) = \big(nI - \mathbb{1}\big)$$

from which we can compute

$$\alpha = \frac{1}{2} + \frac{1}{2n}\sqrt{n^2 + 4nr}. \tag{5.18}$$

The feedback K is then given by

$$K = -(rI + Z^T SZ)^{-1} Z^T S.$$

If we consider KZ, using the fact that $Z^T SZ = \alpha(nI - \mathbb{1})$ and (5.18) then

$$KZ = \gamma(\alpha)\big(I + \beta(\alpha)\mathbb{1}\big)\alpha\big(nI - \mathbb{1}\big) = -k\big(nI - \mathbb{1}\big)$$

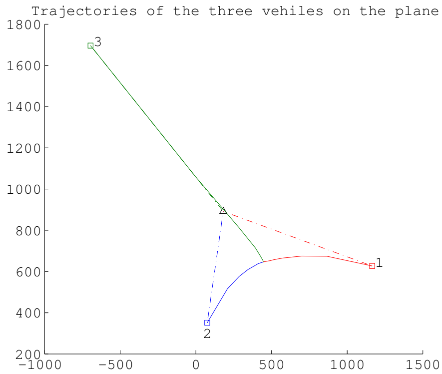
with

$$k = \alpha\gamma(\alpha) = \frac{n + \sqrt{n^2 + 4nr}}{n(2r + n + \sqrt{n^2 + 4nr})}.$$

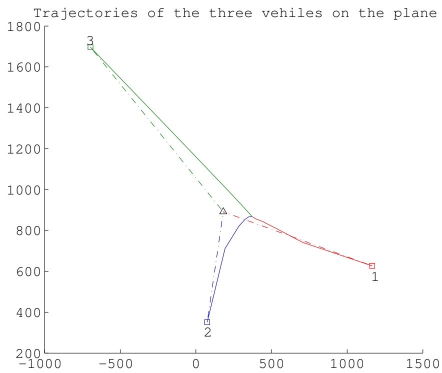
The feedback K is then

$$K = -k(nI - \mathbb{1})Z^T(ZZ^T)^{-1}$$

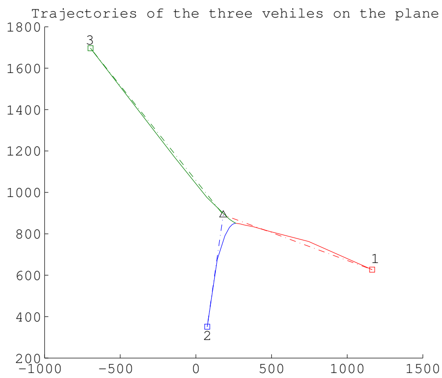
$\square$



(a) Vehicles' trajectories with communication topology 1 (see Figure 5.5).

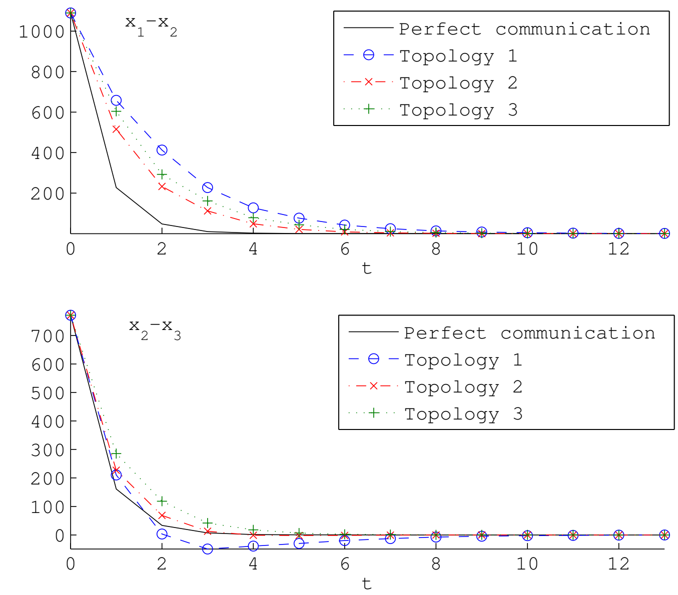


(b) Vehicles' trajectories with communication topology 2 (see Figure 5.5).

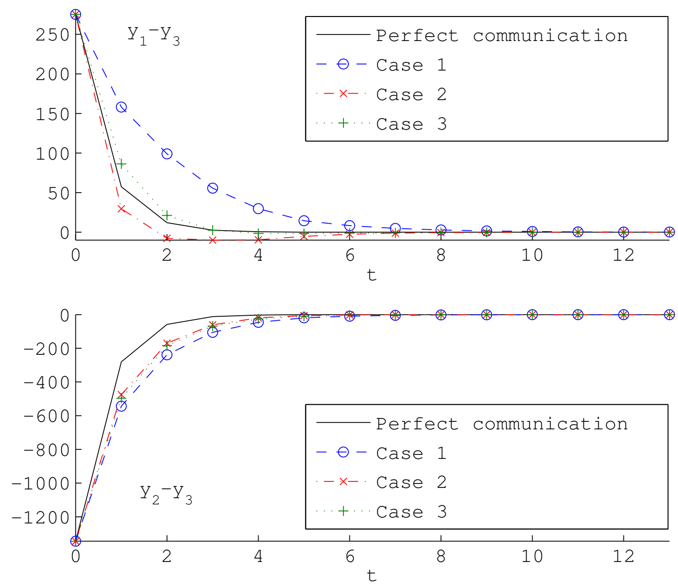


(c) Vehicles' trajectories with communication topology 3 (see Figure 5.5).

Figure 5.6: Trajectories for three vehicles for the three different topologies of figure 5.5. In the simulations we assumed the uniform quantization error equal to zero.



(a) Differences $x_1(t) - x_2(t)$ and $x_2(t) - x_3(t)$ for the three vehicles corresponding to the trajectories of Figures 5.6(a)– 5.6(c).



(b) Differences $y_1(t) - y_2(t)$ and $y_2(t) - y_3(t)$ for the three vehicles corresponding to the trajectories of Figures 5.6(a)– 5.6(c).

Figure 5.7: Performance comparison of the difference communication topologies.

References

[AT92] M.S. Andersland and D. Teneketsis. Information structures, causality and nonsequential stochastic control I: design-independent properties. *SIAM Journal of Control and Optimization*, 30(6):1447–1475, 1992.

[BD98] A. Beghi and D. D'Alessandro. Discrete-time optimal control with control-dependent noise and generalized Riccati difference equation. *Automatica*, 34(8):1031–1034, 1998.

[dSP02] J. B. de Sousa and F. L. Pereira. On coordinated control strategies for networked dynamic control systems: an application to AUV's. In *Proceedings of International Symposium of Mathematical Theory of Networks and Systems*, 2002.

[FZ03] F. Fagnani and S. Zampieri. Stability analysis and synthesis for scalar linear systems with quantized feedback. *IEEE Transaction on Automatic and Control*, 48(9):1569–1584, 2003.

[GR01] C. Godisl and G. Royle. *Algebraic Graph Theory*. Springer, 2001.

[LMA03] J. Lin, A. S. Morse, and B. D. O. Anderson. The multi-agent rendezvous problem. In *Proc. of the 42nd IEEE Conference on Decision and Control, HI, USA*, 2003.

[SJ03] A. Speranzon and K. H. Johansson. On some communication schemes for distributed pursuit–evasion games. In *Proc. of the 42nd IEEE Conference on Decision and Control, HI, USA*, 2003.

[SM03] R. O. Saber and R. M. Murray. Flocking with obstacle avoidance: cooperation with limited communication in mobile networks. In *Proc. of the 42nd IEEE Conference on Decision and Control, HI, USA*, 2003.

[Var93] P. Varaiya. Smart cars on smart roads: Problems of control. *IEEE Transaction on Automatic and Control*, 38(2):195–207, 1993.

Paper B

Hierarchical Control Architecture for a Team of Underwater Vehicles in Search Missions

A. Speranzon, J. Silva, J. B. de Sousa, K.H. Johansson

Abstract

A multi-vehicle search strategy for finding an optimum of a scalar based on the simplex search algorithm is proposed. The strategy is described as hierarchical scheme with two layers. A team controller (upper layer) is described by a discrete-event system. The output of this layer is a set of waypoints for the vehicles and it is used by the vehicle controller (lower layer) to drive each vehicle to the next waypoint. The vehicles can communicate sensor data. Since underwater communication is costly in terms of energy, a protocol that reduces the average communication load is considered. Simulations are carried out in order to evaluate the performance of the search strategy for varying fields and levels of measurement noise.

***Keyworkds*:** Discrete-event systems, Simplex optimization, Multi-robot system, Minimum communication.

6.1 Introduction

The problem of coordination and control of multiple heterogeneous vehicles has recently attracted the attention of researchers in control engineering and computer science. There are several aspects to this problem such as sensing capabilities, layered control strategies, stability of geometric formations and control under communication constraints, just to name a few. In this paper we are concerned with a specific multi-vehicle control problem: given a scalar field, coordinate the motions of a set of vehicles with sampling capabilities to find its minimum (maximum) in a given region. Here we report our investigations concerning the implementation of the fixed-size simplex search algorithm [SHH62]. The simplex algorithm is a direct search method used in many practical optimization problems. Its simplicity and robustness properties [NM65, LRWW98] makes it an interesting algorithm for minimum search applications with multiple vehicles. The simplex method is usually applied in situations where the cost of gradient estimation is high. The method behaves as a gradient descent method even if no explicit gradient calculation is needed. In spite of its wide application, there are few cases for which it possible to prove convergence of the simplex algorithm, for a scalar field with dimension two or higher. It was, actually, shown that the original Nelder-Mead algorithm [NM65] (in which the size of the simplex can change over time) can converge to a non-stationary point even for quite smooth and strictly convex functions [McK99]. The simplex algorithm is useful to improve an initial estimate of the solution in few iterations without explicit estimation of the gradient and with few function evaluations. An informal description of a multi-vehicle search strategy based on the simplex algorithm is proposed in [dSP02]. Here we formalize the search strategy and discuss some of its properties. Our objective is to use the simplex algorithm to progress towards a minimum and to get as close as possible to it. In order to improve the estimate of the solution obtained with the simplex algorithm we can also resort to other kinds of search strategies combined with dynamic estimation. It is not the objective of this paper to analyze the final phase of the search procedure, but we plan to use the proposed algorithm as part of a global search strategy. Optimization algorithms have been used as the inspiration for other multi-vehicle search strategies. Bachmayer et al. [BL02] use a pure gradient-based method for scenarios where a vehicle platoon searches the minimum of a convex and smooth scalar fields. Burian et al. [BYBS96] report results with mixed strategies and present illustrative examples using real data, such as depth profiles of a lake. However, these results are drawn for single vehicle operation.

The main contributions of this paper are the definition of a new motion coordination strategy for the vehicles performing the search operation and a communications protocol for the implementation of the simplex algorithm.

Our work is mainly motivated by the PISCIS project [CMdS+03] at the Underwater Systems and Technology Laboratory (USTL), Porto University, but the results can be applied also to other vehicles and scenarios. The PISCIS project offers an experimental testbed consisting of two small size autonomous underwater vehi-

Figure 6.8: Autonomous Underwater Vehicle used in the PISCIS project at Porto University.

cles (Figure 6.8) with environmental sensors and acoustic modems for underwater communications.

The paper is organized as follows. A general hierarchical model for the multi-vehicle control system is presented in Section 6.2. It consists of two parts: a team controller and a vehicle controller. In Section 6.3 we describe how to map the simplex algorithm to such hierarchical control structure. We first construct a team controller that implements the simplex search algorithm. Starting from the construction of such team controller we then extend it to the case of two independent team controllers which synchronize thought a communication channel. Since communication underwater requires a large amount of power we discuss some communication issues in Section 6.4. In Section 6.5 we describe the dynamic model of the underwater vehicles. Simulation results are reported in Section 6.6 and some conclusions are drawn in Section 6.7.

6.2 Hierarchical multi-vehicle model

Consider a compact convex set $\Omega \subset \mathbb{R}^2$ containing the origin. Define a field through a scalar-valued map $V : \Omega \to \mathbb{R}$ with a global minimum at the origin. Let $n \geqslant 2$ vehicles be positioned in $p_i \in \Omega$, $i = 1, \ldots, n$. Each vehicle can take measurements and communicate them to the other vehicles. Based on the measurements the vehicles are supposed to find the minimum of V. In practice there are limitations on how often measurements can be taken and how accurate the communication is. We propose a hierarchical control strategy [Var00, DLS95, Var72] with two layers: an upper layer, called team controller, modeled by a discrete-event system and a lower layer modeled by a continuous-time control system called vehicle controller, as shown in Figure 6.9. The discrete layer generates waypoints for the autonomous underwater vehicles (AUV's) according to an optimization algorithm. The continuous layer uses waypoints as target points to be reached by the AUV's. The continuous-layer therefore generates feasible trajectories for the AUV's which connects waypoints.

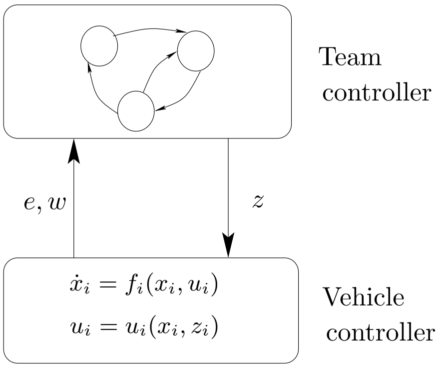


Figure 6.9: Hierarchical control structure for n vehicles.

We assume that the vehicles are able to exchange information through a communication channel.

In the next two subsections we review some definitions and properties of discrete-event systems and discuss how their interaction with continuous-time systems create a hierarchical control structure.

6.2.1 Discrete layer

The discrete layer is modeled by a discrete-event system [CL99].

Definition 6.6 *A discrete-event system is a quintuple*

$$D = (Z, E, W, \xi, z_0) \tag{6.19}$$

where E is the alphabet of events $E = \{e_0, e_1, e_2, \dots\}$, Z is the discrete state space, W the set of inputs, $\xi : E \times Z \times W \to Z$ the transition function, and $z_0 \in Z$ the initial state.

The transition function ξ defines the evolution of the discrete-event system, i.e., it maps the current state to next state once an event happens. Note that in our definition of a discrete-event system, the transition function depends also on the input set W. In the following we recall two important concepts for discrete-event systems that we will use later in the paper.

Definition 6.7 *The* language generated *by $D = (Z, E, W, \xi, z_0)$ is defined as*

$$\mathcal{L}(D) = \{(e, w) \in (E \times W)^* : \xi(e, z_0, w) \textit{ is feasible}\}$$

where $w = (w_1, w_2, w_2) \in W$ is a vector of three real values.

Definition 6.8 *Given two discrete-event systems D_1 and D_2, they are* equivalent *if $\mathcal{L}(D_1) = \mathcal{L}(D_2)$.*

6.2.2 Continuous layer

The continuous layer represents the dynamics of the vehicles and the continuous-time control algorithms. Each vehicle $i = 1, \ldots, n$ is described by a control system

$$\dot{x}_i = f_i(x_i, u_i), \qquad u_i \in U_i$$

where $f_i : X_i \times U_i \to \Omega$ defines the dynamics of the individual vehicles with continuous state x_i and admissible continuous controls in U_i. The control u_i is a state feedback that depends on both the continuous state and the state of the discrete-event system in the discrete layer z_i i.e.,

$$u_i = u_i(x_i, z_i).$$

The interactions between the two layers are described in the following.

6.3 Team controller

The search strategy described in this paper is based on the simplex algorithm which we describe in some detail in the next subsection. We then show how it is possible to design a team controller, modeled as a discrete-event system, that executes the simplex algorithm. An equivalent implementation with communicating team controllers is then designed.

6.3.1 Simplex search

The simplex algorithm is a direct search method used in many practical optimization problems. It is usually applied in situations where the cost of function evaluation is high and gradient calculation is difficult, as happens in scalar field corrupted by noise or time-varying. The algorithm is useful to improve an initial estimate of the solution with few function evaluations. Its simplicity and robustness properties [LRWW98, NM65], make it an interesting algorithm for minimum search applications with multiple vehicles. Notice that the widely used gradient based methods cannot cope with the existence of noise in the field. Moreover, the main objective in this kind of application is not an algorithm which converges in few iterations but one which enhances the synergy between the vehicles given the problem constraints.

Let us define a triangular grid $\mathcal{G} \in \Omega$ as depicted in Figure 6.10, with aperture $d > 0$. Introduce an arbitrary point $p_0 \in \Omega^\circ$ and a base of vectors given by b_1, b_2 such that $b_1^T b_1 = b_2^T b_2 = d^2$ and $b_1^T b_2 = d^2 \cos \pi/3$. The grid is then equal to

$$\mathcal{G} = \{p \in \Omega |\ p = p_0 + kb_1 + \ell b_2,\ k, \ell \in \mathbb{Z}\}.$$

A simplex $z = (z_1, z_2, z_3) \in \mathcal{G}^3$ is defined by three neighboring vertices of $\mathcal{G}$, which belong to a triangle.

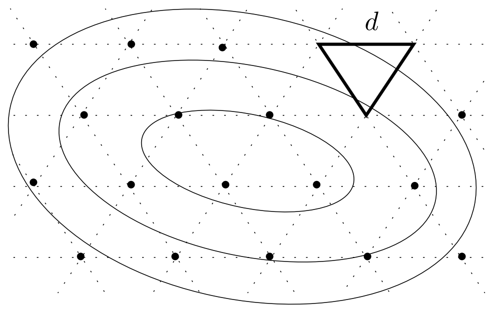


Figure 6.10: A triangular grid with aperture d over a two-dimensional scalar field depicted by its level curves. The solid line triangle illustrates the state z of the discrete-event system evolving on the grid.

We suppose, without loss of generality, that $V(z_3) \geqslant V(z_i)$, $i = 1, 2$. Given a simplex $z = (z_1, z_2, z_3)$ the next simplex, z', is generated from z by reflecting z_3 with respect to the other vertices, namely

$$z \mapsto z' = \mathcal{S}(z_1, z_2, z_3') \qquad z_3' = z_1 + z_2 - z_3 \tag{6.20}$$

where $\mathcal{S}(.)$ defines the following simplex algorithm:

$z(0) \leftarrow (z_1(0), z_2(0), z_3(0))$
$k \leftarrow 0$
while true **do**
  $i \leftarrow \arg\max_i V(z_i(k))$
  $z_i' \leftarrow z_j + z_h - z_i$ with $j, h \in \{1, 2, 3\}$
      and $j \neq h, j \neq i, h \neq i$
  $z_j' \leftarrow z_j$
  $z_h' \leftarrow z_h$
  $z(k+1) \leftarrow (z_1', z_2', z_3')$
  $k \leftarrow k + 1$
  if $k \geqslant 2 \wedge z(k) = z(k-2)$ **then**
    stop
  end if
end while

Notice that the algorithm terminates at time instance $k = N$ with $N \geqslant 2$, if $z(N) = z(N-2)$. Since the algorithm is deterministic, it follows that a continuation after step N would lead to an oscillation between the two discrete states $z(N)$ and $z(N-1)$.

Definition 6.9 *A fixed point χ for the simplex algorithm is a pair of two simplexes that makes the algorithm to terminate. Thus*

$$\chi = \{\big(z(k-1), z(k)\big) \in \mathcal{G}^3 \times \mathcal{G}^3 : z(k-2) = z(k), k \geqslant 2\}.$$

6.3.2 Simplex search as team controller

Let $D = (Z, E, w, \xi, z_0)$ be a discrete-event system modeling the team controller. We will design D so that it implements the simplex search algorithm.

Let the state space of the discrete-event system be $Z \subset \mathcal{G}^3$ such that $z = (p_1, p_2, p_3) \in Z$ is a simplex. The event alphabet E consists of a single enabling event, $\{e\}$, triggered by the underlying continuous-time layer as shown in Figure 6.9. Such event is generated when the submarines reach the waypoints (z_1, z_2), vertices of the simplex z. The activation of such event is discussed in detail in section 6.5.

When an event is triggered the input w is also generated. It consists of the value of the scalar field V at the AUV's current position, which we indicate with p_1, p_2 and p_3. Thus $w = (V(p_1), V(p_2), V(p_3)) \in W$ is the ordered triple of measurements.

Remark 6.1
We assume that the value of the third vertex of the simplex is know to both vehicles. We will discuss later in this section how this value is available to the AUV's.

The transition function $\xi(e, z, w)$ of the discrete-event system is defined as follows

$$\xi(e, z, w) = \begin{cases} (p'_3, p_2, p_1), & \text{if } w_3 \geqslant \max\{w_1, w_2\} \\ (p_1, p'_2, p_3), & \text{if } w_2 \geqslant \max\{w_1, w_3\} \\ & \text{and } w_2 \neq w_3 \\ (p'_1, p_2, p_3), & \text{if } w_1 > \max\{w_2, w_3\} \end{cases} \tag{6.21}$$

where p'_k with $k \in \{1, 2, 3\}$ is the reflected vertex as defined in (6.20). The following proposition follows form construction.

Proposition 6.6 *Let $D = (Z, E, W, \xi, z_0)$ with $z(0) = z_0$ be the team controller previously defined. Let $\mathcal{Z} = \{z(0), \ldots, z(N)\}$ be a sequence of simplexes generated by the simplex algorithm, such that $(z(N-1), z(N))$ is a fixed point. Then the language generated by D is $\mathcal{L}(D) = \mathcal{Z} \cup \{z(N-1), z(N)\}^*$.*

Thus the team controller implements the simplex and in particular z is the set of waypoints where the AUV's need to move in order to find the minimizer of the field V. Such waypoints are used by the continuous-time layer to compute feasible trajectories for the AUV's to reach the vertices of the new simplex.

6.3.3 Communicating team controllers

Consider now the case with two communicating AUV's, see Figure 6.11. Let $\ell_i \in L_i \subset \mathbb{R}^2$ be the data locally available to the ith AUV, in particular $\ell_i = (V(p_i), V(p_3)$, $i = 1, 2$. At each step the ith AUV needs to have some information about $V(p_j)$, $j \in 1, 2$, $j \neq i$ in order to compute the next simplex. We assume that such information is available through a communication channel. Let $c_i \in C_i \subset \mathbb{R}$ be the measurement received by the ith AUV, namely $c_i = V(p_j)$, with $j \neq i$. Notice that

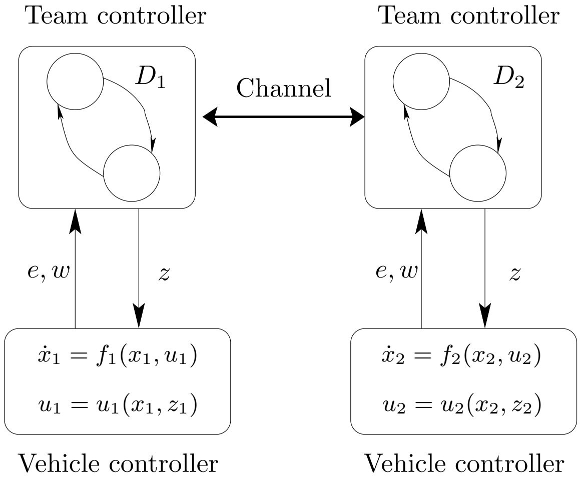


Figure 6.11: Decentralized hierarchical control structure for $n = 2$ vehicles with a communication channel.

when the two AUV's have exchanged measurements they have available the data (l_i, c_i).

We, then,define the following two team controllers that generate waypoints for the AUV's

$$D_i = \big(Z, E_i, (L_i, C_i), \xi_i, z_0\big), \qquad\qquad i = 1, 2 \tag{6.22}$$

The event alphabet contains a common enabling event, triggered by the continuous-time layer, thus $E_i = \{e\}$. In other words, the two AUV's observe the same enabling event e which is triggered when both vehicles have reached their previously computed waypoint. The transition function ξ_i is defined as follows

$$\xi_1(e, z_1, (\ell_1, c_1)) = \begin{cases} (p_3', p_2, p_1), & \text{if } V(p_3) \geqslant \max\{V(p_1), c_1\} \\ (p_1, p_2', p_3), & \text{if } c_1 \geqslant \max\{V(p_1), V(p_3)\} \\ & \text{and } c_1 \neq V(p_3) \\ (p_1', p_2, p_3), & \text{if } V(p_1) > \max\{V(p_3), c_1\} \end{cases} \tag{6.23}$$

and,

$$\xi_2(e, z_2, (\ell_2, c_2)) = \begin{cases} (p_3', p_2, p_1), & \text{if } V(p_3) \geqslant \max\{V(p_2), c_2\} \\ (p_1, p_2', p_3), & \text{if } V(p_2) \geqslant \max\{V(p_3), c_2\} \\ & \text{and } V(p_2) \neq V(p_3) \\ (p_1', p_2, p_3), & \text{if } c_2 > \max\{V(p_2), V(p_3)\} \end{cases} \tag{6.24}$$

Proposition 6.7 *The team controller D and the team controller obtained by composing D_1 and D_2 are language equivalent: $\mathcal{L}(D) = \mathcal{L}(D_1 \| D_2)$.*

The result follows by construction. The parallel composition of D_1 and D_2 is the discrete-event system

$$D_p = D_1 \| D_2 = \big(Z \times Z, \{e\}, (L_1 \cup C_1) \cup (L_2 \cup C_2), \xi_p, (z_0, z_0)\big).$$

where the initial state z_0 is the same simplex for both AUV's. Note that $(L_1 \cup C_1) \cup (L_2 \cup C_2) = W$, defined as for the centralized scheme. Since the enabling event e is observed by both discrete-event system then the transition function ξ_p is defined as follows

$$\xi_p(e, (z_1, z_2), w) = \big(\xi_1(e, z_1, w), \xi_2(e, z_2, w)\big)$$

with $w \in W$ and where $(z_1, z_2) \in Z \times Z$. Since the AUV's know the position of each other and $\xi_1 = \xi_2$ it follows that if the initial condition z_0 is the same for D_1 and D_2 then their future states are the same, i.e. $z_1(k) = z_2(k)$ for all $k \geqslant 0$. This implies that $\xi(e, z_1, w) = \xi(e, z_2, w)$, thus $\xi_p(e, (z_1, z_2), w)$ represents a single state of Z. Then we conclude that $\mathcal{L}(D_p) = \mathcal{L}(D)$.

Remark 6.2
The team controllers specify how the discrete-event systems compute a new simplex. However, they do not define which AUV moves to the reflected point. Such decision is implicitly defined by the transition functions ξ_i (cf. equations (6.23) *and* (6.24)*) if we interpret the next state $\xi_i(e, z_i, (\ell_i, c_i))$ as an ordered tuple, namely the next waypoint for the ith AUV is the ith element of $\xi_i(e, z_i, (\ell_i, c_i))$, with $i = 1, 2$.*

6.4 Communication issues

Underwater communication is very costly in terms of energy since the SNR is generally very low [PSRS01]. From the transition functions (6.23) and (6.24) we notice that both AUV's need to know the value of the field V at all the three vertices of the current simplex in order to generate the next simplex. This is achieved transmitting measurement using underwater acoustic modems. We propose a communication protocols in which measurements and very short (in term of bits/symbols) synchronization codes are used. In doing this the average data rate is decreased compare to a protocol where the raw measurements are communicated.

If the communicated data are not available then the discrete event systems D_1 and D_2 are non-deterministic. This fact follows from that there are three possible states $z_i' = \xi_i(e, z_i, (\ell_i, \emptyset))$, that can be reached.

We consider the following communication strategy. At each step the first AUV sends its measurement, $V(p_1)$ to the second. This AUV then replies using a message γ_k depending on the comparison of the field's value at the vertices of the current simplex. Summarizing we have

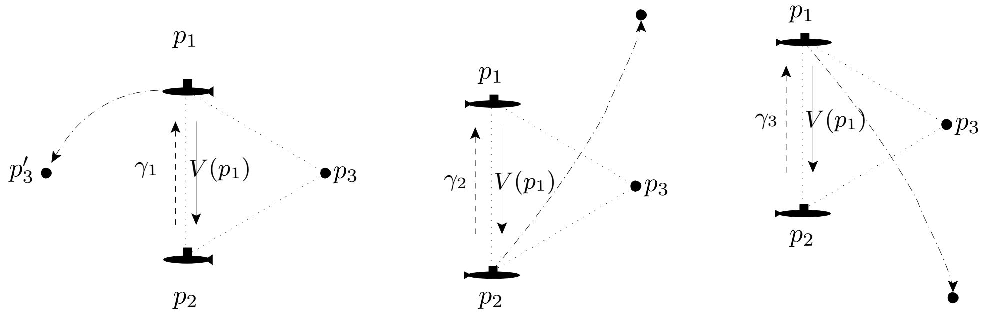


Figure 6.12: Motion of the AUV's and communicated date for two different scenarios. In dotted line is shown the current simplex. With the arrowed solid line is represented the communication of raw measurement, while with a dashed line is represented the transmission of a message. In deash-dotted line is shown the trajectory of the AUV's toward the destination vertexes.

message	condition
γ_1	$V(p_3)$ is the largest
γ_2	$V(p_2)$ is the largest
γ_3	$V(p_1)$ is the largest

The messages γ_k need tp be designed very small, in term of bits/symbols, compared with a measurement.

Observe that the second AUV, after receiving $V(p_1)$, knows the field's value at all vertices of the current simplex, thus it can determine uniquely the message γ_k, $k = 1, \dots, 3$.

The team controllers require the AUV's to know the value of the field V at the common point p_3 at each step. The communication strategy suggested, implicitly assigns the role of master to one of the two AUV's. Thus this constraint can be relaxed and it is necessary that only the master always knows the field's value at that point. We consider as example a possible case that could occur. Suppose $V(p_3)$ is the largest value. This case is shown in Figure 6.12(a). The field assumes largest value at p_3. Thus the next simplex is $\{p_1, p_2, p_3'\}$. With the communication protocol proposed, the vehicle in p_2 receives the measurement taken by the vehicle in p_1, namely $V(p_1)$, shown with a solid arrows line. At this point the vehicle in p_2 has all the measurements available and computes which vertex of the current simplex should be reflected. The message γ_3 is then communicate the the vehicle in p_1. The transition functions ξ_i give

$$\xi_2(e, p_2, \{V(p_1), V(p_2), V(p_3)\}) = (p_3', p_2, p_1)$$

and,

$$\xi_1(e, p_1, \{V(p_1), \gamma_3, V(p_3)\}) = (p_3', p_2, p_1).$$

Thus the next simplex is the ordered triple (p_3', p_2, p_1). This means that vehicle in p_1 will need to move to p_3' and vehicle p_2 should remain standing still.

At the next step the value of $V(p_1)$ is known only to the vehicle in p_2, but after the communication of $V(p_3')$ it can compute which of the vertexes should be reflected and transmit back the corresponding message.

The other two cases, shown in Figure 6.12(b) and Figure 6.12(c), follow similarly.

6.5 Vehicle controller

A complete nonlinear model of a Autonomous Underwater Vehicles can be found in [Fos94]. In this work we consider the problem of controlling the AUV's on a plane. The nonlinear model of the system is

$$\begin{aligned} \dot{x}_i &= V \cos \psi_i \\ \dot{y}_i &= V \sin \psi_i \\ \dot{\psi}_i &= r_i \qquad\qquad\qquad\qquad i = 1, 2 \end{aligned}$$

where $(x_i, y_i)^T$ is the position of the ith AUV with respect to a global coordinate frame, ψ_i is the yaw angle and r_i is the yaw rate. The velocity V is assumed constant. The guidance in the horizontal plane is achieved using a "line of sight" control law: at each time step the vehicle is commanded to head towards the reference waypoint z_i. The continuous-time controller for the yaw is a PID

$$r_i(x_i, z_i) = k_p \varepsilon_i + k_i \int_0^t \varepsilon_i(\tau) d\tau - k_d \frac{d\psi_i}{dt}$$

where

$$\varepsilon_i = \angle(z_i - (x_i, y_i)^T) - \psi_i$$

The event e is generated when the vehicles reach the assigned waypoints. Due to the vehicles' control limitations, in practice, it is not possible to assure that the vehicles will reach the exact waypoint. To overcome this difficulty, when a vehicle reaches a neighborhood of radius δ of the assigned vertex of the simplex, a new measurement is taken and the event e is triggered. Therefore, the values are not always sampled at the exact grid intersections. Additionally, position estimation errors may lead to sampling being done even further from the desired point. Monte Carlo simulations, reported in the following section, have been performed in order to test the robustness properties of the algorithm performing.

Table 6.1: Simulation Results (Average/Standard Deviation)

	Noise level (Std. Deviation)			
	0	1000	2500	5000
Completion time (s)	627	609/48	615/69	585/83
Total travelled distance (m)	1632	1585/124	1600/180	1525/218
Distance to minimizer (m)	22	32/16	40/22	75/41

6.6 Simulation

In this section, we present simulation results to illustrate the implementation of the simplex based search strategy, namely the interaction between team and vehicle controllers. We consider the two autonomous underwater vehicles identical.

In what concerns the size of the simplex (or grid aperture) we are interested in setting it as small as possible because the smaller the simplex the closer we can get to the minimizer. However, the grid aperture is limited below by the dynamic behavior of the vehicle, namely the vehicles' turning radius. The considered simplex size is $d = 40$ m. Positioning errors were modeled by considering a worst case estimation error and by enlarging the acceptance neighborhood in order to encompass this error. In practice, each the vehicle performs a local filtering of the acquired samples along its trajectory. This was also considered in our simulations allowing higher levels of sensor and field noise.

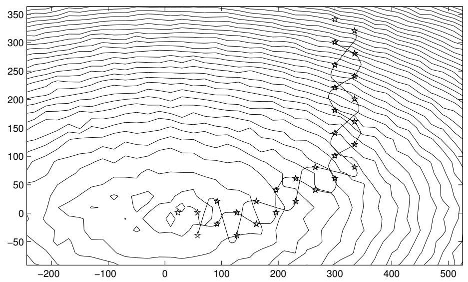
Figure 6.13 shows a simulation run for the scalar field $V(p_1, p_2) = p_1^2 + 4p_2^2$ with gaussian noise superimposed, modeling measurement noise, and vehicles departing from the vertexes of the simplex with centroid at (320,320) (m). Simulations were stopped when the simplex reached a fixed point. The vertexes of the successive simplexes are marked with stars. The same scenario, with different noise realizations, was simulated 100 times. We also considered three different noise levels Results are collected in Table 6.1. As we can notice the vehicles are able to arrive very close to the minimizer. On the other hand, a fixed point is reached much earlier when the noise level is high as shown in the "Completion Time" row.

We have also simulated the behavior of the proposed control structure when the scalar field is time-varying. In Figure 6.14, we have simulated a scalar field that drifts at constant speed (a rough simulation of the stream's effect on a temperature field, for example). Figure 6.14 shows four snapshots of the evolution of the AUV's. As illustrated in the figure, the vehicles are able to move very close to the minimizer of the field.

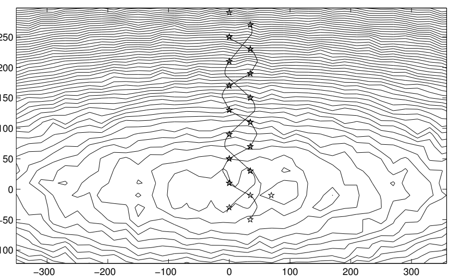
6.7 Conclusions

In this paper we have considered the problem of finding the minimizer of a scalar field using a team of autonomous underwater vehicles. We proposed a hierarchical control strategy in which the high level consists of a team controller, modeled as a discrete-event system, which generates waypoints according to the simplex algorithm. The low level consists of a vehicle controller which generates continuous-time control commands that move the AUV's between waypoints.

A communication protocol have been designed in order to reduce the average bit-rate. Simulation results have been carried out to show how the hierarchical control system moves the AUV's towards the minimizer of a time-invariant and a time-varying vector field.

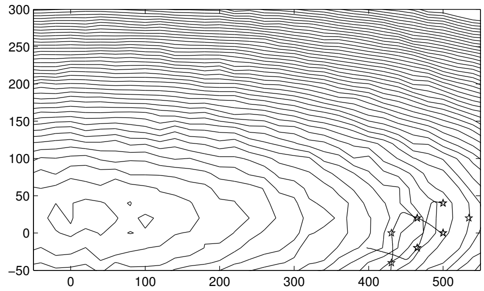


(a)

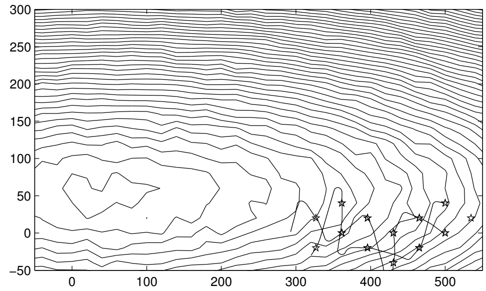


(b)

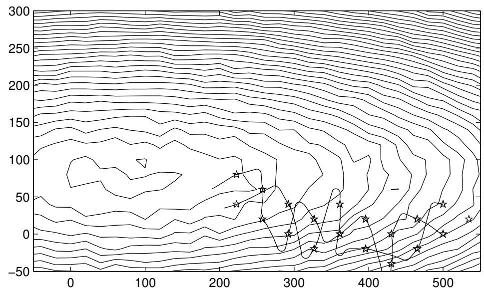
Figure 6.13: Trajectories of two AUV's controlled by the hierarchical control algorithm proposed in the paper.



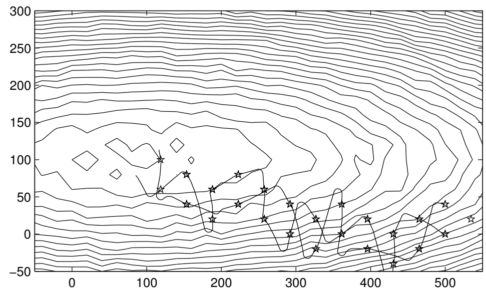
(a) Situation after 6 time steps.



(b) Situation after 12 time steps.



(c) Situation after 18 time steps.



(d) Situation after 24 time steps.

Figure 6.14: The figures show the simplex, the trajectories of the AUV's and a moving scalar field at different time steps. The scalar field, shown through level curves, is a quadratic function with superimposed white noise.

References

[BL02] Ralf Bachmayer and Naomi Ehrich Leonard. Vehicle networks for gradient descent in a sampled environment. In *IEEE Conference on Decision and Control*, 2002.

[BYBS96] E. Burian, D. Yoerger, A. Bradley, and H. Singh. Gradient search with autonomous underwater vehicles using scalar measurements. In *IEEE Symp. Autonomous Underwater Vehicle Technology*, pages 86–89, 1996.

[CL99] C.G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems*. Kluwer Academic, 1999.

[CMdS+03] N. Cruz, A. Matos, J.B. de Sousa, F.L. Pereira, J. Silva, E.P. Silva, J. Coimbra, and E.B. Dias. Operations with multiple autonomous underwater vehicles: the piscis project. In *Proceedings of the Second Annual Symposium on Autonomous Intelligent Networks and Systems*, 2003.

[DLS95] N.G. Datta, J. Lygeros, and S. Sastry. *Hierarchical hybrid control: a case study"*, pages 166–190. Lecture Notes in Computer Science, Springer, 1995.

[dSP02] J. Borges de Sousa and Fernando Lobo Pereira. On coordinated control strategies for networked dynamic control systems - an application to auvs. In *Proceedings of International Symposium of Mathematical Theory of Networks and Systems (MTNS)*, 2002.

[Fos94] T. I. Fossen. *Guidance and Control of Ocean Vehicles*. John Wiley and Sons Ltd., 1994.

[LRWW98] Jeffrey C. Lagarias, James A. Reeds, Margaret H. Wright, and Paul E. Wright. Convergence properties of the nedler-mead simplex in low dimensions. *SIAM Journal of Optimization*, 9(1):112–147, 1998.

[McK99] K.I.M. McKinnon. Convergence of the nelder mead simplex method to a nonstationary point. *SIAM Journal of Optimization*, 148-158, 1999.

[NM65] J.A. Nelder and R. Mead. A simplex method for function minimization. *Computing Journal*, 7:308–313, 1965.

[PSRS01] J. Proakis, E. Sozer, J. Rice, and M. Stojanovic. Shallow water acoustic networks. *IEEE Communications Magazine*, 39(11):114–119, 2001.

[SHH62] W. Spendley, G.R. Hext, and F.R. Himsworth. Sequential applications of simplex designs in optimization and evolutionary operation. *Technometrics*, 4:441–461, 1962.

[Var72] P. Varaiya. Theory of hierarchical, multilevel systems. *IEEE Transaction on Automatic Control*, 17(2):280–281, 1972.

[Var00] P. Varaiya. A question about hierarchical systems. In In T. Djaferis and I. Schick, editors, *System theory: modeling, analysis and control*. Kluwer, 2000.

Paper C

On Some Communication Schemes for Distributed Pursuit–Evasion Games

A. Speranzon, K.H. Johansson

Abstract

A probabilistic pursuit-evasion game from the literature is used as an example to study constrained communication in multi-robot systems. Communication protocols based on time-triggered and event-triggered synchronization schemes are considered. It is shown that by limiting the communication to events when the probabilistic map updated by the individual pursuer contains new information, as measured through a map entropy, the utilization of the communication link can be considerably improved compared to conventional time-triggered communication.

***Keyworkds*:** Probabilistic map, Relative entropy, Time- and Event-triggered communication.

7.1 Introduction

Multi-robot systems have many advantages compared to single-robot systems, including improved flexibility, sensing, and reliability. For most mobile robot systems, one needs to address challenges related to sensor noise, self-localization, and partial knowledge of the environment. For a multi-robot system, the inter-robot communication adds to this list. In practice, every communication channel has a limited bandwidth, which is due to the physical laws on the achievable data rate and to that channels might be shared with other users. The performance of the multi-robot system is often highly dependent on the utilization of the communication network. Still, integrated design of communication protocols has so far gained little attention in the literature of multi-robot systems, cf., [BA94, TBF98].

The main contribution of this paper is to illustrate how information theory [Sha48, CT91] can be used in the design of a multi-robot system, in order to optimize the communication utilization with respect to a control performance. We let a pursuit–evasion game [Isa67, BO95, HKS99, HP02] with several pursuers serve as a prototype system, since it is a good representative for many multi-robot tasks. In particular we consider a probabilistic approach for pursuit–evasion where each pursuer build a probabilistic map of the environment [HKS99, HKA00]. Map entropy is used as an information measure of the probabilistic map. It establishes an event-triggered communication scheme for the pursuers, which is compared with a time-triggered scheme with periodic communications. The considered multi-robot problem can be viewed as a realistic benchmark problem for the design of integrated control and communication systems. Recent work in that area has focused mainly on stabilizability under limited communication, e.g., [WB99, Mit00]. For a stochastic control system, the advantage of event-triggered control compared to time-triggered was discussed by Åström and Bernhardsson [ÅB02]. For distributed real-time systems, design specifications sometimes lead to that a time-triggered scheme is instead preferable, as advocated by Kopetz [Kop93].

The paper is organized as follow. In Section 7.2 we extend the pursuit–evasion model of Hespanha et al. [HKS99] by introducing an explicit broadcasting communication protocol for the pursuer, in which they synchronize their probabilistic maps by broadcasting the current map to each other. Two particular communication schemes are discussed in Section 7.3: time-triggered and event-triggered. The synchronization events are in the latter based on the probabilistic map entropy. In Section 7.4 quantization is utilized to cope with bandwidth limitations. It is shown that the map entropy can be used to quantize the probabilistic map in an efficient way. Simulation results are presented in Section 7.5.

7.2 Pursuit–Evasion with Communication

We consider a pursuit–evasion game with $n_p > 1$ pursuers, $P_1, \dots, P_{n_p}$, and one randomly moving evader. Following Hespanha et al. [HKS99], we suppose that the game is played in a finite-dimensional square space, uniformly partitioned in $n_c^2 < \infty$ cells denoted

$$\mathcal{X} = \{1, 2, \dots, n_c\} \times \{1, 2, \dots, n_c\}$$

Each cell can be occupied by the evader, a pursuer or an obstacle. Neither the evader nor the pursuers can occupy a cell with an obstacle, although the evader and a pursuer can share a cell. The latter corresponds to a capture of the evader. We assume discrete time $t \in \mathcal{T} = \{0, 1, 2, \dots\}$. The motions of the pursuers and the evader are modeled as a controlled Markov chain, see [HP02] for details. Pursuer P_i, $i = 1, \dots, n_p$, senses at each time instance $t \in \mathcal{T}$ the triple

$$\mathbf{z}_i(t) = \{\mathbf{s}_i(t), \mathbf{o}_i(t), \mathbf{e}_i(t)\}$$

where $\mathbf{s}_i(t) \in \mathcal{X}$ is the position of the pursuer, $\mathbf{o}_i(t) \subset \mathcal{X}$ is a measurement of the obstacle locations sensed by the pursuer, and $\mathbf{e}_i(t) \in \mathcal{X}$ is the corresponding measurement of the evader[2]. We assume that all sensors (detecting position, obstacles, and evader) are ideal, and thus are not affected by measurement noise etc. The measurement space is denoted $\mathcal{Z} = \mathcal{X} \times 2^{\mathcal{X}} \times \mathcal{X}$, where $2^{\mathcal{X}}$ is the power set of $\mathcal{X}$.

7.2.1 Synchronization

We extend the pursuit–evasion model of Hespanha et al. by introducing limited pursuer communication. Pursuers gather individual sensor information, make local decisions and communicate at synchronization time instances $\tau \in \mathcal{T}$.

Definition 7.10 *A synchronization is a complete broadcasting communication in which all pursuers exchange information with each other. Denote the data received by pursuer P_i*

$$\mathbf{Y}^i(\tau) = \left\{\mathbf{y}^j(\tau)\right\}_{j \neq i}$$

where $\mathbf{y}^j(\tau)$ is the data transmitted by P_i.

Transmitted data Y^i is in this paper a probabilistic map, as introduced in next section. A synchronization is depicted in Figure 7.15: the network is simultaneously accessed by all pursuers when a synchronization is performed. In the paper we consider two types of synchronization: time-triggered and event-triggered.

Definition 7.11 *A time-triggered synchronization is a synchronization that occurs at a time $\tau_t \in \{\Delta, 2\Delta, \dots\}$, where $\Delta \in \mathcal{T}$ is the synchronization period. An event-triggered synchronization occurs at a time $\tau_e \in \mathcal{T}$, at which a pre-specified event takes place.*

[2] Boldface indicates a random variable and the normal typeface its realization.

7.2.2 Probabilistic Map

The probabilistic map for a pursuer is the probability mass function for the position of the evader conditioned on the available data up to time t.

Definition 7.12 *An element of the probabilistic map of pursuer P_i is given by*

$$\tilde{p}^i_{t+1}(x_e, Z_t) = P(\mathbf{x}_e(t+1) = x_e | \mathbf{Z}^i_t = Z_t) \tag{7.25}$$

where $\mathbf{Z}^i_t \in \{\mathbf{z}_i(0), \ldots, \mathbf{z}_i(t)\}$, is the sequence of measurements taken by pursuer P_i up to time t and $\mathbf{x}_e(t) \in \mathcal{X}$ is the position of the evader at time t.

The probabilistic map is a square matrix, where each element is given by equation (7.25). The map is updated through a two-step algorithm: a measurement step in which $P(\mathbf{x}_e(t) = x_e | \mathbf{Z}^i_t = Z_t)$ is computed using the current measurements, and a prediction step in which $\tilde{p}^i_{t+1}(x_e, Z_t)$ is computed using an evader motion model, see [HKS99] for details. The game starts with an a-priori probabilistic map $\tilde{p}^i_{0|-1}(x_e, \emptyset)$ that we assume to be the uniform distribution.

7.2.3 Control Policy

Let $\mathbf{u}_i(t)$ denote the control action of P_i, which gives the position of P_i at time $t+1$. We consider greedy control policies with constrained motion for the pursuer, i.e.,

$$\mathbf{u}_i(t) = \arg \max_{v \in \mathcal{N}(s_i)} p^i_{t+1}(v, Z_t, Y_t) \tag{7.26}$$

where $\mathcal{N}(s_i)$ are all neighboring cells of the current position s_i of P_i. Thus, at t the control policy moves P_i to a neighboring cell v, which maximizes the conditional probability of finding the evader at $t+1$. The greedy policy does not maximize the local probabilistic map $\tilde{p}^i_{t+1}$, but the probabilistic map p^i_{t+1} which we let depend also on the probabilistic maps received through the network; hence the greedy policy (7.26) depends on the local measurement Z_t and communicated data Y_t. The fusion of probabilistic maps at a synchronization time is computed as a normalized product of all maps, i.e., as an independent opinion pool [Ber85]. This is based on an assumption that the pursuers are far from each other between consecutive synchronization instances. Measurements are thus considered to be approximately independent, cf., [HK02].

7.3 Entropy-triggered synchronization

In this section we introduce an event-triggered synchronization scheme based on the map entropy. We use the notation

$$m^i_x(t) = p^i_{t+1}(x, Z_t, Y_t)$$

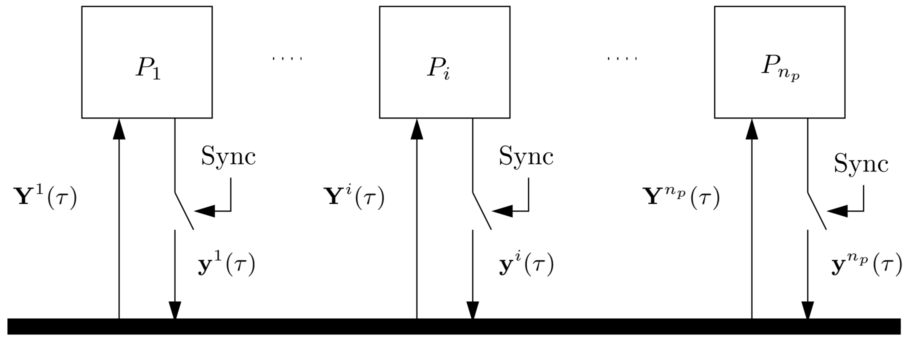


Figure 7.15: At a synchronization time $\tau \in \mathcal{T}$, each pursuer P_i broadcasts $\mathbf{y}^i(\tau)$ to the network and receives $\mathbf{Y}^i(\tau) = \{\mathbf{y}^j(\tau)\}_{j \neq i}$.

for element $x = (k, \ell) \in \mathcal{X}$ of the probabilistic map of P_i and we let $M^i(t)$ denote the corresponding matrix. Inspired by information theory [CT91] we make the following definitions.

Definition 7.13 *The map entropy H of a probabilistic map M is*

$$H(M) = -\frac{1}{2 \log n_c} \sum_{x \in \mathcal{X}} m_x \, \log m_x$$

Definition 7.14 *The relative map entropy D of two probabilistic maps M and N is*

$$D(M||N) = \sum_{x \in \mathcal{X}} m_x \, \log \frac{m_x}{n_x}$$

In particular, the relative map entropy between the probabilistic map M and the uniform probabilistic map $N = n_c^{-2} \, \mathbb{1}_{n_c \times n_c}$ is given by

$$D(M||N) = \frac{1 - H(M)}{2 \log n_c}$$

In the following we consider two event-triggered synchronization schemes: one based on the map entropy and the other based on the relative map entropy.

7.3.1 Synchronization based on dynamic threshold

Let a synchronization event be triggered whenever $H(M^i(t)) < \lambda(t)$, where $\lambda(t)$ is the synchronization threshold. The threshold is updated according to

$$\lambda(t+1) = \begin{cases} \alpha \, \lambda(\tau_k), & \text{if } t+1 = \tau_k; \\ \lambda(t), & \text{otherwise.} \end{cases}$$

with $\lambda(0) = \lambda_0 > 0$

with τ_k being the time for synchronization k and $0 < \alpha < 1$. The decreasing dynamic threshold is natural since the map entropy is in most cases decreasing with time.

7.3.2 Synchronization based on relative map entropy

Let us consider a synchronization scheme based on the relative map entropy D that triggers a synchronization event whenever the local probabilistic map differs sufficiently much from the previously broadcasted map. Synchronization k is carried out when

$$D(M(t)||M(\tau_{k-1}) > \xi \qquad\qquad \tau_{k-1} < t$$

i.e., a synchronization is performed when the relative entropy between the current probabilistic map and the last synchronized probabilistic map is larger than a positive constant ξ.

7.3.3 Discussion

The idea behind event-triggered synchronization based on the map entropy is that the map entropy should reflect the amount of information in the probabilistic map. It is then natural to expect that the map entropy decreases as a pursuer moves around gathering more and more information about the environment. Simulations show that this is often the case, in particular in the earlier part of a game. However, it is easy to construct counter examples in which the map entropy increases at one or more times before the evader is captured; thus in general $H(M^i(t))$ is not a decreasing function. On the other hand, for a static evader it is straightforward to show that $H(M^i(t))$ is a decreasing function. This follows from that at each step the number of zero elements $z(t)$ of $M^i(t)$ is non-decreasing, while all other elements are equal to $1/(1-z(t))$.

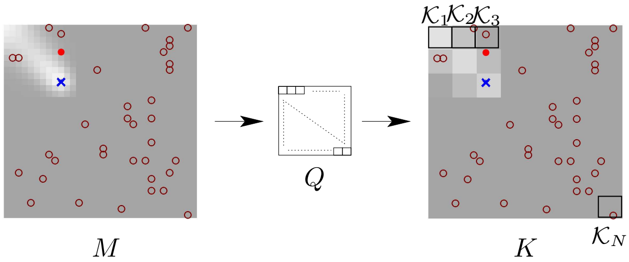
7.4 Bandwidth limitations

In order to cope with communication bandwidth limitations, it is natural to send only the part of the probabilistic map M that contains most of the information[3]. The idea is to transform the map M into a new map K, denoted reproduction probabilistic map. The map K should contain almost all information in M but should require less bits to be encoded. We consider a vector quantization

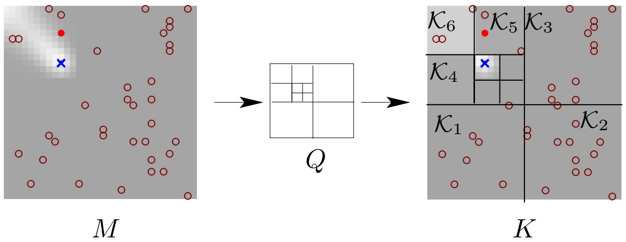
$$Q : \mathbb{R}^{n_c \times n_c} \rightarrow \mathbb{R}^{n_c \times n_c} : M \mapsto K = Q(M)$$

where Q defines a complete partition of the matrix M into square sub-matrices $\mathcal{M}_1, \ldots, \mathcal{M}_N$ of order $n_1, \ldots, n_N$ such that $\sum_{i=1}^{N} n_i = n_c$. The reproduction prob-

[3]In this section, the pursuer index i and the time dependence are suppressed.



(a) Uniform quantization



(b) Non-uniform quantization

Figure 7.16: Vector quantization $K = Q(M)$ of probabilistic map M.

abilistic map K is block partitioned correspondingly into $\mathcal{K}_1, \ldots, \mathcal{K}_N$ with

$$\mathcal{K}_i = \frac{1}{n_i^2} \mathbb{1}_{n_i \times 1} \mathcal{M}_i \mathbb{1}_{1 \times n_i} \mathbb{1}_{n_i \times n_i} \tag{7.27}$$

Each element of $\mathcal{K}_i$ is thus given by the average of the elements of $\mathcal{M}_i$. Associated with the quantization Q, we define a distortion measure

$$d(M, K) = |H(M) - H(K)|$$

The choice of granularity in the block partition, i.e., the order of the sub-matrices, should be chosen such that $d(M, K)$ is small. This corresponds to a small loss of information in the quantization. Trade-off between quantization granularity and distortion is treated by the rate distortion theory [Ber70]. An analytical characterization of this trade-off seems to be hard to obtain in our case. We therefore consider two heuristic approaches.

7.4.1 Uniform quantization

For uniform quantization, the block partition of Q is such that the order of all blocks are equal, $n_i = n$. An illustrative example is shown in Figure 7.16(a). In this case the number of elements of K is equal to n_c^2/n^2, while the number of elements of M is n_c^2.

7.4.2 Non-uniform quantization

A possible non-uniform quantization is illustrated in Figure 7.16(b). This corresponds to a “divide-and-conquer” scheme, which is known as vector quantization with QuadTree map [GG91]. The partition $\mathcal{M}_1, \dots, \mathcal{M}_N$ imposed by the quantization Q is in this case carried out recursively, such that

$$\dim \mathcal{M}_1 = \frac{1}{4} \dim M$$
$$\dim \mathcal{M}_{i+1} = \begin{cases} \dim \mathcal{M}_i, & \text{if } i \bmod 3 = 0; \\ \frac{1}{4} \dim \mathcal{M}_i, & \text{otherwise} \end{cases} \tag{7.28}$$

In each recursion step, the current block is divided into four sub-matrices. Three of them are quantized using (7.27), while the remaining sub-block is partitioned into four sub-blocks, and so on. The recursion stops when the smallest block has reached a preassigned dimension n. Compared with uniform quantization, one advantage of the non-uniform quantization is the possibility of an on-line termination of the quantization if the loss of information is too high, i.e., if the distortion measure $d(M, K)$ is too large. Solving the recursion (7.28), we find that the number of values to transmit is in the order of $\log n_c^2 + n^2$.

7.5 Simulation results

Sets of hundred Monte Carlo simulations have been performed in order to evaluate the proposed synchronization and quantization strategies. The capture time T^* and the number of synchronization instances S are used as performance indices. Figures 7.17 and 7.18 show the results for a game with two pursuers and one evader on a grid with $n_c^2 = 24^2$ cells. Three different synchronization schemes are compared: time-triggered, event–triggered based on dynamic threshold, and event-triggered based on relative map entropy. The time-triggered synchronization has a synchronization period of $\Delta = 20$. We see in Figure 7.17 that the capture time T^* is varying considerably over the set of experiments. The mean capture time $\overline{T}^*$ is similar for all synchronization schemes as indicated by the dashed lines (dashed-dotted lines indicate the standard deviations). The result is collected in the following table:

Synchronization schemes	$\overline{T}^*$	$\overline{S}$
Time-triggered	68	3.9
Event-triggered dynamic threshold	64	2.1
Event-triggered relative map entropy	66	2.6

Note that the mean number of synchronization times $\overline{S}$ is much smaller for the event-triggered schemes than for the time-triggered, while the average capture time $\overline{T}^*$ is about the same. Hence, event-triggered synchronization allows a more efficient utilization of the communication channel. This fact is also illustrated in Figure 7.18. The main difference between the two event-triggered schemes is mainly the distribution of synchronization times. We see that when using relative map entropy the pursuers tend to communicate more regularly. This is due to that new information is available quite regularly for the pursuers and this information triggers the synchronization events in this scheme. In Figure 7.19 the uniform and non-uniform quantization strategies are compared. The results are for a game with two pursuers and one evader on an environment that consists of $n_c^2 = 32^2$ cells. The synchronization is time-triggered with period $\Delta = 20$. The quantization map Q has been chosen so that the dimension of the sub-matrices $\mathcal{K}_i$ is $n^2 = 8^2$. Figure 7.19 shows T^* for the following cases: no quantization ($n = 1$), uniform quantization with $n = 8$ and non-uniform quantization with $n = 8$. The experimental results are collected in the following table:

Quantization	$\overline{T}^*$	$\overline{d}(M, K)$	$\overline{V}$
Uniform ($n = 1$)	91	0.	1024
Uniform ($n = 8$)	141	0.1	16
Non-uniform ($n = 8$)	120	0.04	70

Here $\overline{d}(M, K)$ denotes the average distortion over all experiments and $\overline{V}$ the average number of broadcasted matrix elements at each synchronization. Note that $\overline{V}$ is one to two magnitudes smaller for the quantized cases compared to the non-quantized case. Still the mean capture time is only about 50% larger. The uniform quantization compared with the non-uniform quantization has quite high average distortion $\overline{d}(M, K)$. This implies a relevant loss of information that makes $\overline{T}^*$ larger in this case. On the other hand the average number of transmitted data $\overline{V}$ is considerably reduced.

7.6 Conclusions and future work

In this paper, we have presented communication protocols based on time-triggered and event-triggered synchronization for a distributed pursuit–evasion game. The event-triggered schemes were based on the entropy of the probabilistic map. Simulations showed that by limiting the communication to certain events, the utilization of the communication link can be considerably improved compared to conventional time-triggered communication with uniformly distributed synchronization times. Two vector quantization maps were considered in order to cope with bandwidth limitations. A distortion measure based on the map entropy was introduced to evaluate the compression of the probabilistic map. The communication schemes developed in the paper can be applied in cases when a probabilistic map has to be sent through a network channel to a decision maker. This is common in mobile robotics: examples include localization of robots using occupancy grids [ME85, TBF98]. A related problem of our current interest is optimal localization of mobile sensors,

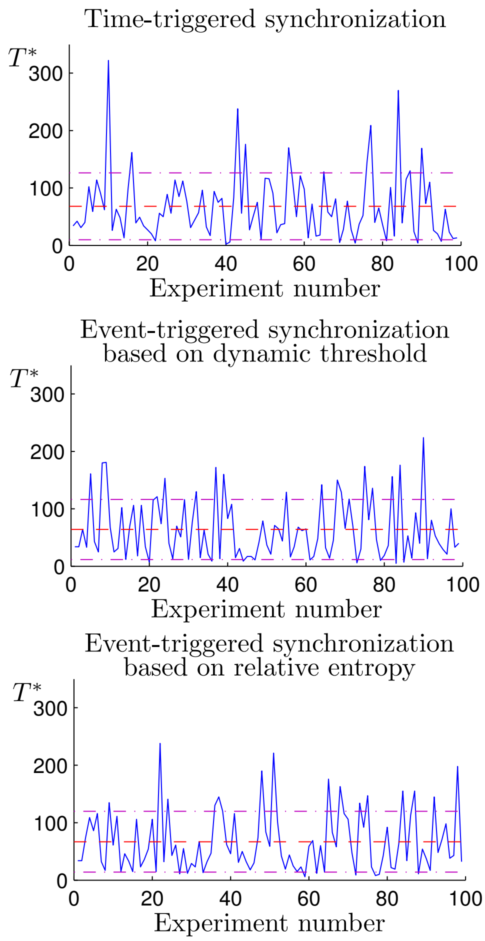


Figure 7.17: Capture time T^* for hundred Monte Carlo experiments and the three proposed synchronization schemes. The dashed line is the mean capture times $\overline{T}^*$ and the dashed-dotted is line the standard deviations. The map size is $n_c^2 = 24^2$.

which share a bandwidth limited communication channel.

7.7 Acknowledgments

The authors gratefully acknowledge Joao P. Hespanha for fruitful discussions and for providing us with a simulation environment. This work is supported by the European Commission through the RECSYS project IST-2001-32515 and by the Swedish Research Council.

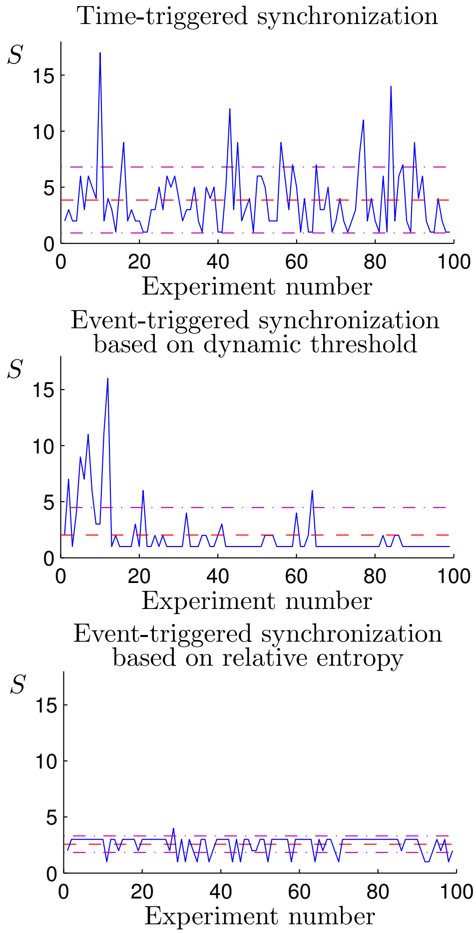


Figure 7.18: Number of synchronization instances S for the same hundred Monte Carlo experiments as in Figure 7.17.

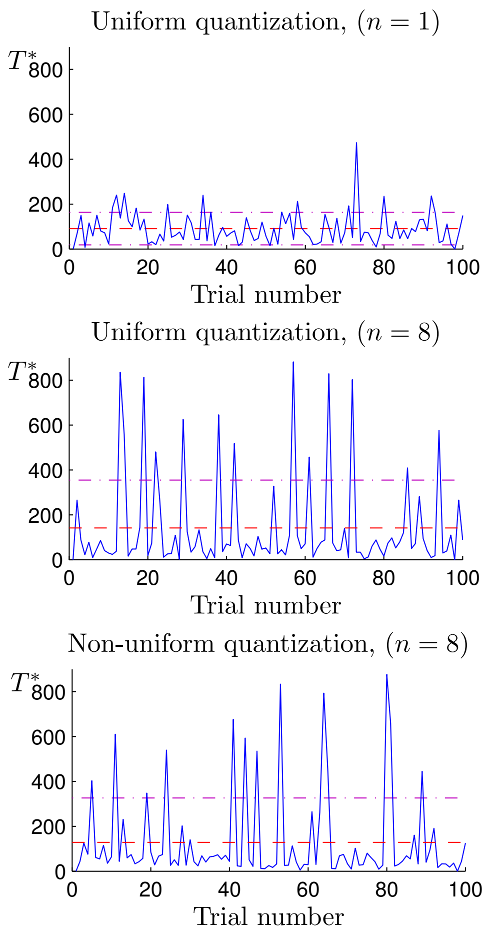


Figure 7.19: Capture time T^* for hundred Monte Carlo experiments. In this case the size of the map is $n_c^2 = 32^2$.

References

[ÅB02] K.J. Åström and B.M. Bernhardsson. Comparison of Riemann and Lebesgue sampling for first order stochastic systems. In *IEEE Conference on Decision and Control*, 2002.

[BA94] T. Balch and R.C. Arkin. Communication in reactive multiagent robotic systems. *Autonomous Robots*, pages 27–52, 1994.

[Ber70] T. Berger. *Rate distortion theory.* Prentice-Hall, NJ, 1970.

[Ber85] J.O. Berger. *Statistical decision theory and bayesian analysis.* Springer-Verlag, Berlin, 1985.

[BO95] T. Basar and J.G. Olsder. *Dynamic noncooperative game theory.* Academic Press, 2nd edition, 1995.

[CT91] T.M. Cover and J.A. Thomas. *Elements of information theory.* Wiley Interscience, 1991.

[GG91] A. Gersho and R.M. Gray. *Vector quantization and data compression.* Kluwer, 1991.

[HK02] J.P. Hespanha and H. Kizilocak. Efficient computation of dynamic probabilistic maps. In *Mediterranean Conference on Control and Automation*, 2002.

[HKA00] J.P. Hespanha, H.H. Kizilocak, and Y.S. Ateskan. Probabilistic map building for aircraft-tracking radars. Technical report, University of Southern California, Los Angeles, CA, 2000.

[HKS99] J.P. Hespanha, H. J. Kim, and S. Sastry. Multiple-agent probabilistic pursuit–evasion games. In *IEEE Conference on Decision and Control*, volume 3, pages 2432–2437, 1999.

[HP02] JP. Hespanha and M. Prandini. Optimal pursuit under partial information. In *Mediterranean Conference on Control and Automation*, 2002.

[Isa67] R. Isaac. *Differential Games*. Robert E. Kriger Publisher Company, 2 edition, 1967.

[Kop93] H. Kopetz. Should responsive systems be event triggered or time triggered? *IEICE Transaction on Information and Systems*, pages 1525–1532, 1993.

[ME85] H.P. Moravec and A.E. Elfes. High resolution maps from wide angle sonar. In *IEEE International Conference on Robotics and Automation*, 1985.

[Mit00] S.K. Mitter. Control with limited information: the role of systems theory and information theory. *IEEE Information Theory Society Newsletter*, 4(50):122–131, 2000.

[Sha48] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423,623–656, 1948.

[TBF98] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*, 31:29–53, 1998.

[WB99] W.S. Wong and R.W. Brockett. Systems with finite communication bandwidth constraints ii: stabilization with limited information feedback. *IEEE Transaction on Automatic Control*, 44(5):1049–1052, 1999.

Part III

Conclusions

Chapter 8

Conclusions and future work

This chapter briefly summarizes the key ideas presented in the three papers. Each summary ends with some future research directions.

Paper A

In Paper A we have studied the rendezvous problem for a team of robots that communicate over quantized channels. Two different types of communication channels are considered: uniformly and logarithmically quantized. The second one is preferable since in general less bits need to be communicated compared to the uniform quantized. For a class of feasible communication topologies, control laws that solves the rendezvous problem are derived. A stochastic approximation of the logarithmic quantization as multiplicative noise is also considered. The analysis, limited to the case of two vehicles, shows that when the quantization error becomes very large, the information communicated over logarithmically quantized channels does not affect the control action of the vehicle that receives it.

The communication topologies seem to influence the performances, for example the speed of convergence to the rendezvous point. A theoretical analysis that would allow the comparison between topologies seems to be a needed step. A possible way to address this problem could be the stochastic approximation of the logarithmic quantizer. Formation control under quantized communication, where complex geometries are consider is a interesting future research direction. Furthermore we would like to explore the possibility of having time-varying communication topologies.

Paper B

A hierarchical control structure for a team of autonomous underwater vehicles employed in finding the minimizer of a scalar field is discussed. The controller is composed of two layers. The upper layer is the team controller which is modeled as discrete-event system. It generates waypoints based on the simplex search optimization algorithm. The waypoints are used as target points by the lower control layer, which continuously steer each vehicle from the current to the next waypoint. Since underwater communication is costly in terms of energy, a protocol which reduces the amount of data to be exchanged is also designed.

The hierarchical control structure seems to be promising for modeling multi-robot systems. The possibility of dividing the design problem in layers and including the communication as a part of the discrete layer seems to help in tackle the limitations it imposes in a easier way. Verification that specifications are fulfilled when we have interaction among different layers is an important direction to be investigated. In particular specifications of interest regard the capability of the system to handle reconfiguration of the vehicles, to recover from faults in the communication and vehicles. In the paper we have studied only a particular example and a generalization to other optimization algorithms would strength the framework.

Paper C

A distributed probabilistic pursuit–evasion game is considered in this paper. The environment is divided in cells and a probabilistic map i.e., a map that associate to each cell a value which is the probability the evader to be there, is constructed. The pursuers can synchronize their probabilistic maps through a common communication channel. The problem we address is the dependence of the capture time with the frequency of the synchronization. Two different approaches are compared: time-triggered synchronization and event-triggered synchronization. The first is driven by a clock with fixed frequency, the second depends on the information content of the probabilistic map. Monte Carlo simulations show that the event-triggered synchronization do not increase the capture time despite the fact that the synchronization happens seldom. A discussion on how to cope with possible bandwidth limitations during synchronization is also included.

There are many robotics problem where the robots build probabilistic maps. For example in mapping an environment or for navigation, occupancy grids are still used. A possible future work direction could be the analysis of the proposed communication schemes in such applications. Further analysis is also needed for the investigation of event-based and time-triggered communication. It would be interesting to trade-off how often and how much you communicate.