Controller Design for Underwater Vehicle Systems with Communication Constraints

by

Brooks Louis-Kiguchi Reed

S.B., Massachusetts Institute of Technology (2009)

S.M., Massachusetts Institute of Technology (2011)

Submitted to the Joint Program in Applied Ocean Science and Engineering,
Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

and the

WOODS HOLE OCEANOGRAPHIC INSTITUTION

February 2015

Author .....
Joint Program in Applied Ocean Science and Engineering,
Department of Mechanical Engineering
December 18, 2014

Certified by .....
Franz S. Hover
Associate Professor, MIT
Thesis Supervisor

Certified by .....
Dana R. Yoerger
Senior Scientist, WHOI
Thesis Supervisor

Accepted by .....
David E. Hardt
Chairman, Department Committee on Graduate Students
MIT

Accepted by .....
Henrik Schmidt
Chairman, Joint Committee for Applied Ocean Science and Engineering
MIT/WHOI

Controller Design for Underwater Vehicle Systems with Communication Constraints

by

Brooks Louis-Kiguchi Reed

Submitted to the Joint Program in Applied Ocean Science and Engineering,
Department of Mechanical Engineering
on December 18, 2014, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Real-time cooperation between autonomous vehicles can enable time-critical missions such as tracking and pursuit of a dynamic target or environmental feature, but relies on wireless communications. Underwater, communication over distances beyond about one hundred meters is almost exclusively accomplished through acoustics, which bring challenges such as propagation delays, low data rates, packet loss, and scheduling constraints due to interference and limited bandwidth. These limitations make underwater pursuit missions preeminent applications of networked control. Motivated by such applications, this thesis presents contributions towards multi-vehicle feedback control in the presence of severe communication constraints.

The first major area of work considers the formulation and solution of new underwater multi-vehicle tracking and pursuit problems using closed-loop control. We begin with a centralized robust optimization approach for multicast routing and power control which is suitable for integration with vehicle control. Next, we describe field experiments in range-based target pursuit at high tracking bandwidths in a challenging shallow-water environment. Finally, we present a methodology for pursuit of dynamic ocean features such as fronts, which we validate using hindcast ocean model data. The primary innovation is a projection algorithm which carries out linearization of ocean model forecast dynamics and uncertainty directly in vehicle coordinates via a forward model technique. The resulting coupled linear stochastic system is suitable for networked control.

The second area of work presents a unified formalism for multi-vehicle control and estimation with measurement, control, and acknowledgment packets all subject to scheduling, delays and packet loss. The modular framework we develop is built around a jump linear system description incorporating receding horizon optimization and buffering at actuators. Integration of these elements enables synthesis of a novel technique for estimation using delayed and lossy control acknowledgments—a desirable and practical capability of fielded systems that has not been considered to date. Simulations and field experiments demonstrate the effectiveness of our approach.

Thesis Supervisor: Franz S. Hover
Title: Associate Professor, MIT

Thesis Supervisor: Dana R. Yoerger
Title: Senior Scientist, WHOI

Acknowledgments

I have been lucky to have two co-advisors. Each has his own style, but both were instrumental in my development as a researcher. First of all, I would like to thank Franz Hover, who I first met my sophomore year. We have been working together ever since, and he has been an incredibly supportive, inspirational, and helpful teacher, advisor, and mentor during my entire journey at MIT. Dana Yoerger's energy and enthusiasm opened my eyes to the importance of ocean science and the possibilities for making an impact using vehicles. Every conversation with Dana is exciting, and I am thankful for his high-level perspective.

My fellow graduate students in the Hover Group made time in the lab enjoyable. Josh T, Brendan, and Eric showed me the way, Mei was there for many interesting discussions, and Chris and Pedro provided some much-needed entertainment during the final stages of my thesis. Josh Leighton gets special mention for his ability to keep our experimental equipment happy. Rob has been a good friend of mine for even longer than Franz, and together we started the multi-vehicle surface testbed which has played a major role in all of my work.

I want to thank my committee members, Jim Preisig and Milica Stojanovic, for their constructive feedback and help with acoustic communications. Pierre Lermusiaux and the MSEAS group graciously taught me about ocean models and gave the data used in Chapter 5. On the experimental side, I thank Mike Benjamin for his work and support related to MOOS-IvP, Toby Schneider for his Goby software, and Keenan, Sandipa, Jim and Lee from the WHOI Acomms group for excellent technical support. The MIT Sailing Pavilion was instrumental in our field experiments, as well as being the site of many great sailing memories during my time at MIT.

I am lucky to have a great group of friends who made my graduate years exciting and memorable, but it would be unfair to single out just some of you! If you are reading this, thanks. And finally, I want to thank my family, for supporting me in all of your various goofy ways.

Work is supported by:
Office of Naval Research, Grant N00014-09-1-0700, and
National Science Foundation, Contract CNS-1212597.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation and Problem Overview

The ocean plays a crucial role in the global climate and ecosystem, and understanding the complex interactions between humans and the ocean is important for the long-term prosperity of society. However, the ocean is incredibly difficult to study, as remote sensing below the surface is difficult, measurements from ships are very expensive, and stationary or drifting platforms do not provide dense coverage.

Robotic systems have emerged as important tools for efficient, low-cost, and low-risk study of the ocean. Early robotic vehicles used tethers to a ship for power and remote-controlled operation. While remotely operated vehicles (ROVs) are still workhorses of marine industry and research, their connection to the ship makes them expensive for monitoring missions, and additionally their large size limits maneuverability and speed. For many survey applications, autonomous underwater vehicles (AUVs) can cover more ground and deliver a more desirable data product.

In recent years, technology for underwater propulsion, sensing, energy storage, imaging, and navigation has been maturing—resulting in highly capable vehicles. Many flavors of AUVs have been invented, ranging from slow yet efficient gliders, to large survey vehicles featuring many sensors, to highly maneuverable vehicles meant for operating in tight near ships or seafloor structures [170]. Basic water properties are routinely measured today from mobile robots, while sophisticated chemical

and biological analyses *in situ* are becoming mature technologies, for example DNA probes [216] and mass spectrometers [37]. AUVs themselves as well as components are increasingly becoming commercialized, resulting in lower costs and more widespread use. These systems have made an impact in naval operations, undersea oil and gas, underwater archeology, and ocean research.

With the successes of single-vehicle AUV operations, the focus has begun to move towards more advanced team behavior and collaboration [18]. Already exploited regularly in the terrestrial and air domains, networks of mobile agents are an attractive means for tracking and pursuit of dynamic processes over mixed spatial scales [68], although wireless communication inevitably brings fundamental challenges in networked control [14]. Surfacing to use satellite comms is very expensive in terms of time and energy, and while optical communications are a great new technology, at present they are only suitable for links up to one or two hundred meters in clear water. For longer distances, acoustics are the preferred method of wireless underwater communication, however, there are fundamental limitations to this channel: limited and distance-dependent bandwidth, time-varying multipath propagation that makes decoding packets difficult, and the low speed of sound in water (1500 m/s as opposed to the speed of light) [116]. For multiple vehicle networks, acoustics are subject to scheduling constraints due to interference and limited bandwidth.

These communication constraints have limited the use of acoustic communications in high-performance, real-time tasks. This is for good reason—assets are expensive, the ocean environment is risky, and large benefits can come from even basic uses of communications. For example, integrated data assimilation has been aided by coordinated adaptive sampling at slow update rates via surfacing and satellite comms [195]. Non-time-critical acoustic communications has increased the effectiveness of many missions, both via uplink of sensor info to operators on a ship, and downlink of basic commands such as updated waypoint lists. However, this thesis argues that *in order to enable new dynamic missions in the ocean, we must consider closed-loop control with acomms in the feedback loop.* Some examples of such missions are tight formation flying, cooperative pursuit of targets such as marine animals, or pursuit of an

oceanographic feature like a plume from an oil well or underwater volcano eruption. In such scenarios, multiple cooperating vehicles can be highly effective, even necessary, for achieving spatial and temporal resolution simultaneously. To enable these capabilities, communication constraints must be at the forefront of control design.

Much work in acoustic communications has been aimed at traditional communication systems, which focus on achieving reliable transmission at high throughput. With any error correction scheme, the price to be paid for increased reliability is increased latency due to coding delay, and decreased throughput due to added redundancy. In practical communication systems, handshaking and retransmissions are usually used for reliable transport, such as the TCP protocol often used in the internet. However, due to the propagation delays of acomms, TCP is not effective underwater.

For feedback control, the needs of a communication system are different. Control systems operate in real-time, so latency becomes arguably the most important consideration. Long coding delays and/or retransmissions are not effective, as old information is not very useful to a controller. Instead, it is often desirable to sacrifice reliability and/or throughput for short delays. Of course, traditional feedback control assumes that information moves around the control loop with no constraints or errors. The field of communication-constrained control is also known as networked control, and has received considerable theoretical attention. Motivated by the challenges laid out so far related to dynamic underwater missions, this work aims to bring advanced networked control algorithms into the field of marine robotics.

We believe that dynamic control of multiple underwater vehicles communicating with acoustics is a preeminent application of networked control. To this end, this thesis presents contributions towards centralized multi-vehicle feedback control in the presence of severe communication constraints typical of underwater acoustics. A block diagram of such a scenario is shown in **Figure 1-1**. The first major area of work considers the formulation and solution of new underwater pursuit problems using closed-loop control, namely field experiments in multi-vehicle target pursuit, and "oceanographic pursuit" of dynamic ocean features. The second area of work presents a unified formalism for multi-vehicle control and estimation with measure-

ment, control, and acknowledgment packets all subject to scheduling, delays and packet loss.

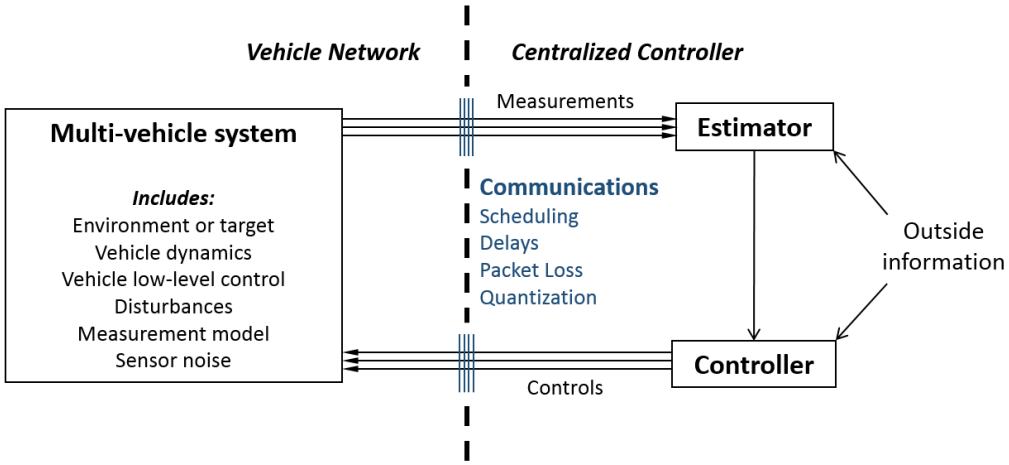


Figure 1-1: Networked control system with centralized estimator/controller and multiple vehicles, connected via constrained communication links.

1.2 Background and Prior Work

The vision of a dynamic multi-vehicle cooperative ocean monitoring system builds on a number of diverse topics: vehicle autonomy, navigation and control, underwater communications, numerical ocean models, and networked control. In this section, we present background and prior work in these topics, as well as relevant vehicle operations underwater. We provide an overview here and give more detailed literature review in the specific chapters. **Chapter 3** discusses a collection of experimental works in the specific context of target pursuit, **Chapter 4** includes more background on numerical ocean models, and **Chapter 5** reviews some specific networked control work in more detail.

1.2.1 Vehicle Control and Navigation

Onboard flight control is developed and tuned specific to the vehicle design (shape, control surfaces, thruster placement), and ranges from simple PID controllers to

highly nonlinear MIMO control systems for vehicles with complex dynamics. Often vehicles use trackline-following, path-planning, or trajectory-generation methods as an outer loop around low-level thrust and attitude control [84]. Above these low-level controllers there is some form of an autonomous decision-maker. This software ranges from simple modules that execute preplanned missions (for example, visiting a series of waypoints), to powerful adaptive mission planners such as MOOS-IvP [25], or T-REX [159]. Additionally, due to acoustic links to a ship, many AUVs rely on some aspect of human-in-the-loop decision making for low-frequency high-level planning, leveraging the economical mobility and data-gathering capabilities of the AUV combined with the experience and knowledge of human scientists [30, 215, 257].

The primary impediment to navigation underwater is that GPS does not work below the ocean surface. Depth, magnetic heading, and orientation are relatively easily obtained underwater in the open ocean, however methods for accurately determining geo-referenced position are challenging. Advanced odometry-based navigation can be quite accurate when expensive sensors are used. Navigation systems relying on inertial measurement units (IMU) and Doppler velocimetry (DVL) are frequently used in the underwater environment [132]. These systems have been reported to give sub-meter navigational accuracy, and also work well when combined with low frequency updates from a global navigation system (such as the acoustic methods described in the next section). However, these systems have significant drawbacks. A high-end IMU costs $150,000, while a DVL costs $30,000 or more depending on depth-rating, and Doppler velocimetry is only useful within range of a solid boundary. DVL bottom-lock range is frequency-dependent and is inversely proportional to the accuracy of measured velocities. As with very high-end IMUs, these units are prohibitively expensive and large in size for use in small, economical AUVs. Price and form factor aside, inertial and Doppler methods suffer from drift over time—errors accumulate as acceleration and velocity are integrated to give position. The latest high performance inertial and Doppler methods have drift rates as low as 0.1% of distance traveled, a 'good' system could have drift on the order of 0.5%, and obviously, as cheaper and smaller components are used, performance degrades further.

Acoustics can provide GPS-like drift-free globally referenced navigation underwater, albeit with other limitations. There are two main classes of acoustic navigation underwater that provide drift-free global reference: Long baseline (LBL) [162] and Ultra-short baseline (USBL) [243]. These systems use the travel time of sound in water to determine distance and therefore track acoustic pingers.

The most effective underwater navigation is achieved using drift-free acoustic systems combined with IMUs and DVLs to achieve accuracy on the order of one meter [132, 136, 203, 248]. With multiple-vehicle fleets, collaborative navigation using inter-vehicle ranging can help improve position estimation accuracy [13, 73, 75].

1.2.2 Underwater Communications

Radio-frequency wireless communications, the workhorse of terrestrial systems, are infeasible underwater due to severe attenuation. Attenuation is less dramatic at low frequencies, however systems running as low as 433 MHz have only been reported to propagate just over one meter underwater [7]. Transmissions at extremely low frequencies (ELF, 30-300 Hz) can propagate through conductive seawater, and are commonly used for communications by US Navy submarines [113], however transmission at these frequency bands requires large antennas and high power, making it impractical for use by small autonomous vehicles. Optical communications using lasers or LEDs have also been considered for high-bandwidth underwater communications [137] and can offer high throughput in certain conditions (several Mbits/sec at ranges up to 100-200 m [65, 76, 77, 130]), however optical links are affected by high scattering due to particles in the water and have limited range. They are also challenged by ambient light in shallow water operations.

Similarly to navigation, underwater communications are primarily accomplished through acoustic links. Acoustics are unique due to distance-dependent bandwidth [228] and long propagation delays. Various technologies exist for acoustic modems, usually operating in the 10-30 kHz range. Performance of acoustic modems varies significantly based on the modulation type used and the channel characteristics. Frequency shift keying is a simple noncoherent modulation technique which is relatively

reliable and low-power, but offers low communication throughput. Phase-shift keying (PSK) with channel equalization is a more complex coherent modulation method that requires more processing but offers the possibility of orders of magnitude higher throughput [87]. Channel characteristics can vary in different ocean applications based on the water depth, bottom topography, oceanographic water properties, sea surface conditions, ambient noise, and the direction of communication [229]. Deep water vertical channels offer the best conditions for acoustic communication due to low ambient noise and scattering in the mid-water column, less difficulty with multipaths, and lower variance on delays [221]. The shallow water channel is much more difficult due to multipaths from surface and bottom effects, high delay spreads, and a high Doppler spread [7]. A rough performance limitation for vertical channels in deep water is 100 km $\cdot$ kbps for the range-rate product [131], while in shallow horizontal channels achievable throughput can be as low as 80 bps, and sometimes channel availability can completely vanish for tens of minutes [179]. Recent work has focused on signal processing such as multiple input-multiple output channel estimation and spread-spectrum techniques for improving the performance of phase-coherent methods [47].

There are a number of commercial off-the-shelf acoustic modems available [6], such as the WHOI micromodem [85], models by Teledyne Benthos [5], LinkQuest [3], EvoLogics [2] and DSPComm [1]. Additionally, USBL navigation units include acoustic modem capabilities integrated into the transceiver and transponders, such as with the Sonardyne Ranger USBL system used with the NDSF vehicle Sentry [4]. These USBL units support transmission of position data obtained by the USBL interleaved with short data or control packets.

Acoustic communications are half duplex, as transducers can not send and receive at the same time [150]. Additionally, due to collisions of acoustic packets at the receiver, great care must be taken with acoustic modem systems if communications with multiple nodes must be achieved. Research is being conducted with multiple access (MAC) schemes, however the most widely used method in practice is simple Time Division Multiple Access (TDMA), where a time slot is allocated for each transponder

to communicate [116]. Specialized "spatial reuse" geographic routing and scheduling techniques exist for TDMA where improved performance can be obtained by considering interference caused by distant transmissions, instead of requiring a strictly collision-free schedule [64, 125, 180].

Finally, we note that the use of acoustic communications for *dynamic control* purposes has a different set of requirements and goals compared to the traditional view of communication theory and networking. Since control is real-time, time-averaged throughput is not the primary metric of interest. Old packets are not as useful to a control system, and long block codes cannot be used to improve throughput because they increase the latency of the measurements and commands in the control loop. This affects choices of packet size, forward error correction codes, as well as transport protocols. Much of the research on optimization of these choices for communication networks is not relevant for networked control systems, where communication constraints are incorporated into control design in an integrated manner.

1.2.3 Ocean Science and Numerical Ocean Models

Similar to numerical weather prediction (NWP) for the atmosphere [200], numerical ocean models now play a major role in our understanding of ocean science [106]. Originally, these models were global-scale, and did not have resolution sufficient to study smaller and more dynamic features. As models improve, the situation is changing. The behavior of ocean fronts and similar features such as plumes and filaments has long been of interest to oceanographers [79, 92]. Recent measurements in a front off Japan have revealed sub-mesoscale structure that figures unexpectedly large in the energy balance [62]. Fronts and plumes are implicated in foundational work on Lagrangian coherent structures [173], and can show dramatic physical, chemical, and biological variability that is critical to understanding ocean-atmospheric coupling, ecological systems, and pollution [37, 78]. Despite continual advances in modeling of complex natural processes, ocean fronts at the mesoscale and smaller remain challenging [41, 111], and hence have emerged as a primary focus area for mobile sensing systems.

The quality of predictions is of course a perennial concern in modeling any stochastic, nonlinear process [133,144,177,204,230]. Large-scale data assimilation is often accomplished using ensemble Kalman filter techniques [74,120]. An ensemble of monte-carlo model runs with variable forcing and initial conditions is a popular means for describing forecast uncertainty [209]; we use such forecast techniques in **Chapter 4**.

1.2.4 Relevant Vehicle Operations Underwater

We lay out some background in vehicle operations in the ocean in the following sections. Prior work with multiple-vehicle operations is the most directly related to this thesis, however, we also give background on some single-vehicle operations where relevant sampling, path-planning, and control designs are used. We focus primarily on experimental work here, although notable theoretical and simulation works are included as well.

Multiple Vehicles

An overview of multi-vehicle operations in the ocean is given in **Figure 1-2**, showing two axes: reliance on communications, and use of environmental models. A selection of particularly relevant works are placed in appropriate locations within this two-dimensional space. These works as well as more are described in more detail below.

Early references on the benefits of multiple vehicles for ocean surveys include Willcox *et al.* [251] and Curtin *et al.* [59]. Leonard *et al.* have studied coordinated control with multiple gliders extensively, including field experiments in Monterey Bay [83, 142, 143, 176]. Coordination was performed via surfacing and satellite communication with a centralized control center on shore, often including human input.

Schneider and Schmidt present a command and control architecture for coordinating multiple vehicles from a ship using both RF and acoustic communications [215]. Multi-vehicle relays using acoustics were studied by Murphy *et al.* [167] and Cheung *et al.* [46]. Various acoustic communication infrastructures for multi-vehicle deployments have been presented, *e.g.* Grund *et al.* for PLUSNet [107], the GREX project [8, 33],

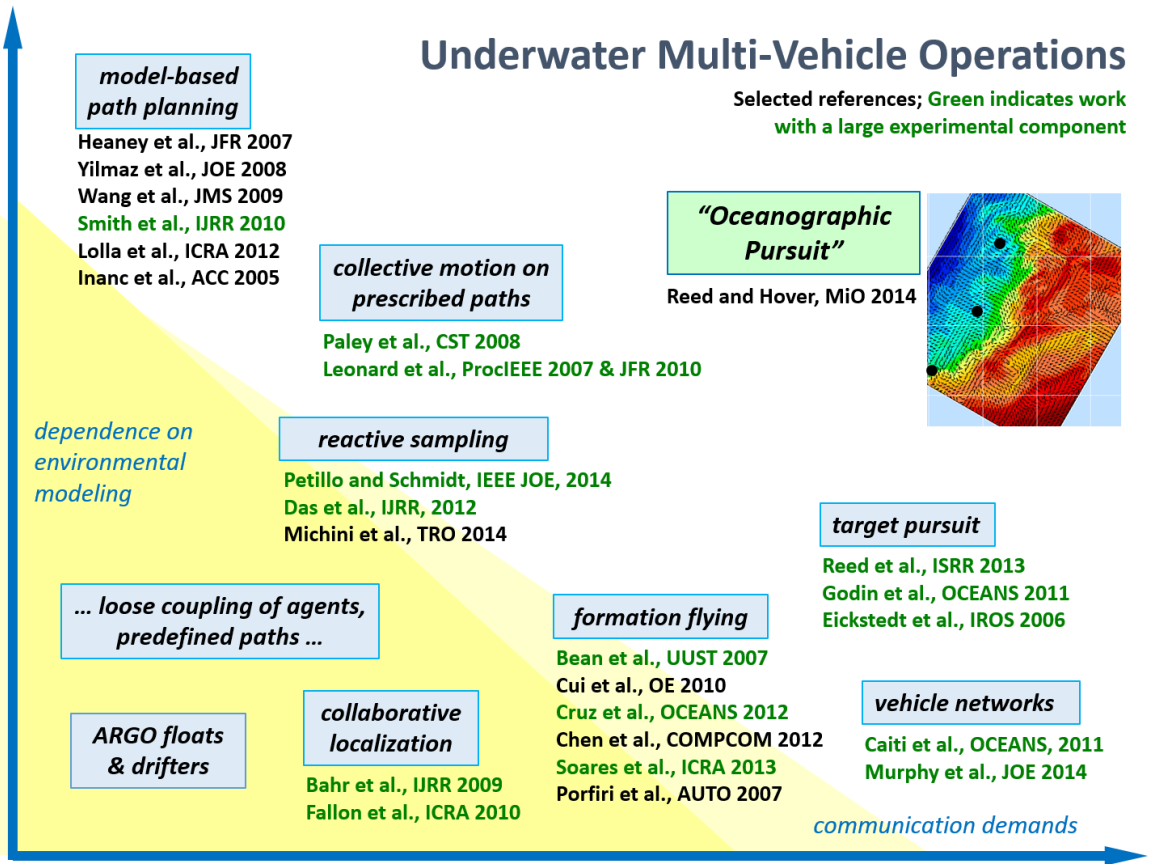


Figure 1-2: Literature review of multi-vehicle operations in the ocean, plotted according to their use of environmental modeling and reliance on communications. Image adapted from F. Hover.

and Caiti *et al.* for the Underwater Acoustic Network (UAN) project [35,36]. Packet loss rates in mobile networking experiments are studied in [34,38].

Distributed navigation using acoustic ranging is studied in [13,58,75,129], while formation control and leader-follower experiments at relatively slow speeds are presented in [17,33,53,226]. Advanced control approaches for leader-follower missions are studied via simulations in Cui *et al.* [56]. Coordinated tracking of acoustic signals using multiple vehicles and arrays is considered experimentally in [69,151,152]. Das *et al.* consider coordinated deployments of vehicles and drifters for Lagrangian tracking, including field results [61]. Petillo & Schmidt give experimental results with two AUVs performing coordinated adaptive surveys for detecting internal waves in [181].

Sampling and path-planning

There has been considerable work studying advanced planning optimizations for multiple vehicles via simulations. Sampling strategies for data assimilation are presented by Heaney *et al.* [114], and related path-planning optimization is considered by Yilmaz *et al.* [256]. Collaborative control for tracking Lagrangian coherent structures is studied by Michini *et al.* [160]. Petillo *et al.* present a distributed simulation approach for plume and thermocline tracking in [182].

Although not multi-vehicle, there has been related experimental work with single vehicles that make reactive decisions based on measurements. A single vehicle has successfully tracked a plankton bloom [98]. Tracking of internal waves and the thermocline has been performed by Cruz & Matos [54], Cazenave *et al.* [43], and Zhang *et al.* [260]. Similarly, Zhang *et al.* present tracking of upwelling fronts in Monterey Bay [259,261,262].

Path-planning under knowledge of current forecasts has been studied extensively, for example by Smith *et al.* [225] and Lolla *et al.* [149]. Lagrangian coherent structures have also been used for path-planning in currents [124].

Teleoperation

While not directly related to autonomous vehicles, underwater teleoperation is a relevant field [201]. ROVs of course are remotely controlled via the fiber optic tether, which offers lossless high bandwidth communication with very low latency. However, when vehicles are remote controlled via acoustic communications, communication constraints are a very important aspect of the system. Interest has been increasing for using un-tethered AUVs for manipulation tasks. While autonomous manipulation is in its early phases [153], the possibility of supervised or semi-autonomous manipulation via wireless communication is intriguing.

Most approaches for acoustic-based remote control do not perform closed-loop dynamic control (such as force-feedback teleoperation). The human operator gives open-loop commands, possibly with some local closed-loop assistance to avoid disturbances [212]. Often, model-based prediction is used to reduce the effects of the communication delays and give the operator an up-to-date representation of the pose of a manipulator, an example of such a system is described by Sayers [211].

More recently, there have been experiments with wireless operation of the NEREUS vehicle, both over acoustic and optical links [32, 249]. In these experiments, feedback loops for the manipulator arms were closed onboard the vehicle, with only joint positions and parameter settings sent over the wireless link to avoid stability issues. Still, the pilots reported latency as the most challenging aspect of controlling the vehicle and manipulator.

1.2.5 Networked Control

Traditional control theory assumes that signals between sensors, controllers and actuators are perfect, *e.g.* there are no communication constraints. When communication is not ideal, such as with wireless communications, challenges arise for control [14]. There are many ways of representing communication constraints, ranging from fundamental information-theoretic bounds on channel capacity to practical abstractions that model the behavior of specific packet-based protocols as seen by the control

system.

Theoretical questions of performance and stability of even very simple plants subject to communication constraints have been the subject of seminal papers in networked control. The famous Witsenhausen counterexample [252] was an early result in networked control, and has attracted considerable research interest even to this day. Regarding stability, Tatikonda & Mitter related the channel capacity to the unstable eigenvalues of a dynamic system, and presented design techniques for encoding and decoding [232]. Martins & Dahleh studied disturbance rejection and present a new Bode-like integral relating unstable dynamics and channel capacity [154]. Sahai & Mitter present the notion of anytime capacity and discuss the relationship between coding delay and control performance in [208].

For application to real-life systems, the theoretical bounds of information theory are less useful, and most work considers some variation of the packet-based network abstraction. While specific network protocol stacks can become very complex, a simplified explanation is as follows. Packets include a certain amount of information and take a certain amount of time to transmit. Usually, the field of networked control does not consider the physical layer in detail, working instead with higher level abstractions. After encoding and transmission across a lossy channel, packets are decoded successfully with some probability, and dropped if decoding is unsuccessful. Delay includes the time to encode, transmit, and decode the packet, on top of the propagation delay through the wireless medium (much longer in water than air!). The packet size and schedule for a particular link in the network determines the throughput of that communication link; packet size, encoding, and modulation are determined based on a tradeoff between bit rate and packet loss. The quantization of the information (*e.g.* sensor measurements, or actuator commands) depends on the packet size chosen.

This thesis (and the majority of the work described below) uses the abstraction of packet loss, delay (and scheduling if multiple communication links are used), and quantization to describe a communication link. We first will give an overview of constructive techniques for networked control, and then discuss Model Predictive Control and its application to networked control. **Figure 1-3** shows the relationships

and differences between some major results in networked control, focusing on the main elements of our jump linear system (JLS) control technique developed in **Chapter 5**, namely Model Predictive Control, scheduling and delays, and robustness to packet loss.

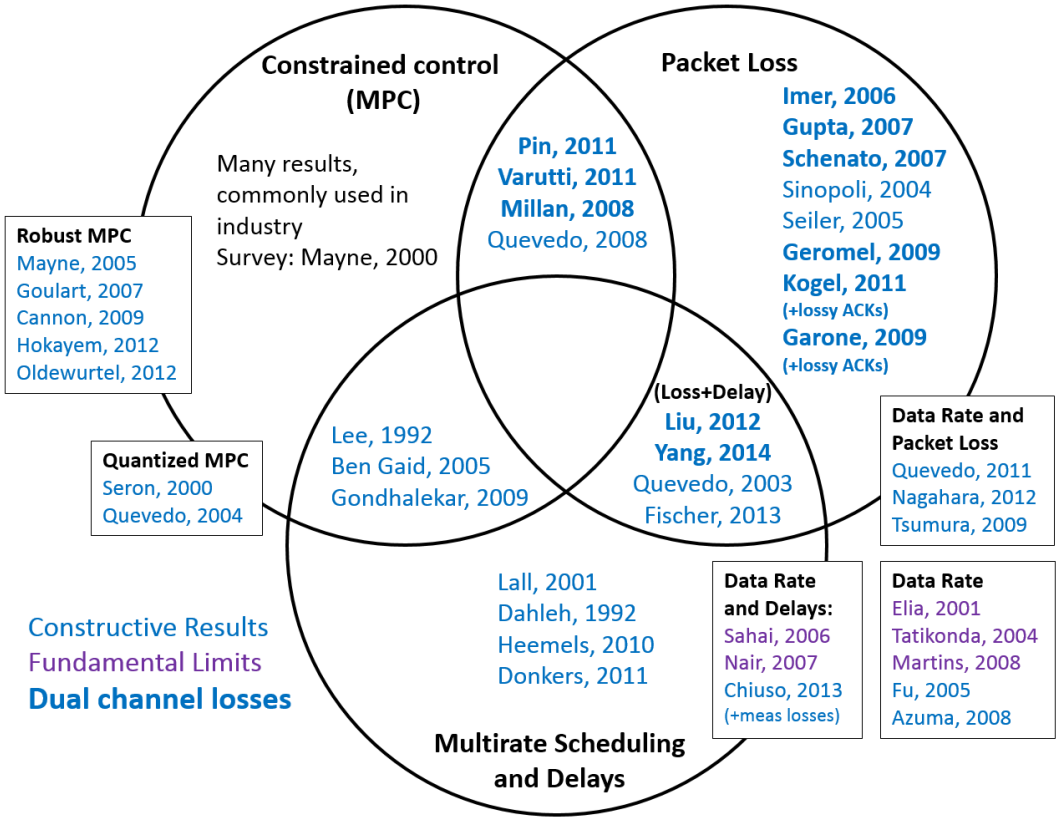


Figure 1-3: Literature review of networked control techniques, focusing on Model Predictive Control, scheduling and delays, and robustness to packet loss. The works in purple consider fundamental information-theoretic limits, and the works shown in blue develop constructive techniques using packet-based abstractions. Constructive works shown in bold consider losses in both the sensor-estimator and controller-actuator links of a feedback loop.

Constructive Techniques

While literature on networked control is extensive, the focus of this thesis is on connecting advanced control techniques to real-world applications. The literature reviewed in this section is relevant to this goal.

Signal delays are commonly considered in control design; deterministic delays are easily handled with standard techniques, and there is extensive literature on stochastic delays (see Nilsson *et al.* for a survey [172]). Similarly, estimation subject to delays has been addressed in [139, 227].

Related to delays, deterministic scheduling is an aspect of networked control systems that arises when multiple vehicles must share a communication medium. As mentioned in Section 1.2.2, interference considerations and bandwidth limitations of underwater acoustic networks mean that the most common approach for multiple access is TDMA scheduling. This scenario results in a multirate control system, where measurements and commands are sent at different times, and possibly at different rates. Some results in multirate control include the ℓ_1 optimization approach of Dahleh *et al.* [60] and the LMI approach of Lall & Dullerud [138].

Packet loss when there are no delays or rate limits has been studied extensively for the case of quadratic cost. The Kalman Filter is easily set up to handle intermittent measurements; performance in this situation is analyzed in [222]. Gupta *et al.* present the Modified Information Filter for the scenario where there are packet losses between the sensor and controller in an LQG control loop [108]. Alternative encoding schemes for dealing with packet loss include multiple-description coding [126, 175] and temporal packet coding [205], although each of these increases latency due to coding delay. Control techniques for the scenario with lossy channels between the sensor and controller and the controller and actuator have been presented by Schenato *et al.* [214] using linear matrix inequalities (LMIs), and by Imer *et al.* [123] using dynamic programming; Imer's dynamic programming approach results in a highly tractable recursion. We extended the work of Imer *et al.* to the case of independent multi-channel packet losses [196]. A major consideration in these schemes is whether or not control packet acknowledgments are available. If they are, then the usual separation principle holds and estimation and control can be designed independently. If they are not, then the "dual effect" is present—control packet uncertainty affects estimation. A middle ground is lossy control acknowledgments, studied in [93, 94, 134]. With underwater acoustic networks subject to long propagation delays and interfer-

ence constraints, acknowledgments may be very costly in time. Notably, the case of delayed and lossy acknowledgments has not been considered in the networked control literature, motivating our treatment of this case in Chapter 5.

Many works consider quantized control where all packets are successful without delay. Fu presents a sector bound approach where a logarithmic quantizer is treated as a norm-bounded uncertainty within a robust control scheme [80, 88]. An alternate approach is to use dynamic quantizers that adjust the quantization window based on the plant state [10, 11, 163]. These sort of dynamic quantizers rely on lossless communication, and synchronization issues can arise if packet loss is present.

The Markov jump linear systems (MJLS) community has studied control within $\mathcal{H}_2$ and $\mathcal{H}_\infty$ frameworks. MJLS descriptions can incorporate complicated jump behavior, including packet loss. The usual assumption is that the "mode" or "jump variable" is available to the controller, which in the networked control setting means that control packet acknowledgments are available. MJLS approaches based on LMIs for this case are studied in [52, 82, 96, 99, 217, 218]. The case with no mode observations is significantly more difficult and has received limited attention, often via difficult and suboptimal iterative optimizations [238]. MJLS approaches can also handle deterministic schedules, but the tractability of the LMI solutions does not scale well with problem size/schedule length [218].

There have been limited works in networked control that consider more than one of the above communication constraints simultaneously, as theory becomes quite complex. Results have been limited to stability tradeoffs, as opposed to performance bounds. Tsumura *et al.* study tradeoffs between packet loss and quantization [234], Chiuso *et al.* study packet loss and delay [48], Donkers *et al.* study scheduling constraints and delays [66], and Heemels *et al.* consider packet loss, quantization, and delays [115].

1.2.6 Model Predictive Control

Model Predictive Control (MPC) is a control approach that leverages real-time online optimization to compute a trajectory of optimal control commands over a finite

horizon. Traditionally, the first command is executed each step, and the process is repeated with a rolling horizon. The primary advantage of MPC over linear control is that the optimization approach allows for state and input constraints as well as certain types of nonlinearities to easily be handled [158]. The drawback is computational complexity, however as computing power improves and optimization solvers become more efficient, MPC can be used effectively in many applications [26]. MPC is widely used in industry, although the constraints and rolling horizon make theoretical analysis more difficult [121, 187].

One disadvantage to MPC is that it assumes perfect state information. A common approach is certainty equivalence (CE-MPC): use an estimator and design control under the assumption that the estimate is the true state and there will be no future disturbances. In this case, the cost function is deterministic. The logic follows from the separation principle in LQG control, and often works well in practice [49, 223, 247]. Various approaches for robust MPC have been presented, *e.g.* [21, 103, 104, 118, 157, 174, 253]. These techniques are discussed in more detail in Section 5.9.5, however they are often considerably more computationally intensive than deterministic MPC.

MPC has also been applied to networked control, most often to the case of packet loss between the controller and actuator. For this scenario, a natural approach called packetized predictive control (PPC) is to send an entire trajectory of commands to a buffer at the actuator [19, 109, 193]. If future packets are lost, the actuator executes commands from the buffered trajectory. Obviously, disturbances cannot be rejected if packets are lost, however, this approach offers many advantages compared to the usual approaches of zero or hold-input control. Variations on PPC study the rate-distortion tradeoff when quantization is present [192], and formulate the optimization to generate sparse control trajectories using the ℓ_1 norm [168]. The PPC idea of sending buffered trajectories has been applied in a number of networked control settings [57, 81, 100, 146, 161, 183, 184, 241]. These approaches are discussed in more detail in Section 5.2.4.

Quantized MPC is a related area of research. Explicitly including quantization levels in the optimization is an option, however this results in an intractable com-

binatorial optimization problem. In a series of related works, Goodwin *et al.* derive optimal closed-form solutions for quantized MPC with vector codebooks, and study the partition of the state space that characterizes the solutions [102, 189–191, 219]. These approaches scale poorly with long trajectories, as the number of codebook entries scales with the trajectory length and lookup becomes an expensive operation.

MPC is quite flexible for modification to different problems, with the tractability of the underlying optimization being the most important consideration. Other extensions of MPC include hybrid autonoma [20], MJLS [239, 240], and distributed MPC [169, 213].

1.3 Field Experiment Setup

The field experiments in **Chapter 3** and **Chapter 6** both use our autonomous surface craft and acoustic modem testbed, with operations in the Charles River. We describe this testbed here.

1.3.1 Autonomous Surface Vehicles

We use autonomous kayaks as shown in **Figure 1-4** for our experiments; they are also described in [97]. Each craft is $1.8m$ long, weighs about 40 kg, and has a rotating thruster near the bow for propulsion and steering. The maximum speed of the vehicles is approximately 1.7 m/s. The relevant navigation sensors available on each vehicle are a tilt-compensated compass and RTK GPS. We use Novotel GPS antennas, uBlox GPS receivers, and the RTKlib software package [231], and have observed position variances on the order of 10^{-4} m^2. Raw compass measurements are passed through a first-order low-pass filter with time constant 2 s, and the noise variance on this signal is estimated as 10 deg^2.

The vehicles run MOOS-IvP autonomy software [25] integrated with custom control algorithms and modem interfaces. We rely on the the MOOS heading PID controller, which runs at five Hz, and the MOOS trackline controller, which runs at two Hz. Step response experiments with the kayak under closed-loop heading control

indicate a rise time of roughly four seconds, and 30% overshoot; we also note the kayaks are able to turn 180 degrees in approximately three seconds. The MOOS trackline controller is an inner-outer loop that modulates the desired vehicle heading so as to steer it toward a point on the trackline, some lead distance l_d ahead. When the waypoint is closer than the lead distance, the vehicle simply drives towards the waypoint. For longer distances the result for small errors is a proportional map for desired heading: $\phi^d \simeq e_x/l_d$, where e_x is the cross-track error in meters and ϕ^d is in radians.[1]

Figure 1-4: The Charles River Basin in Cambridge/Boston, MA, and the autonomous kayak Nostromo. Water depth is 2-12 m.

1.3.2 Acoustic Communications

We use the WHOI Micro-Modem [85], a well-established and commercially available technology for underwater acoustic communication. Modems are towed by the vehicles, suspended at a depth of about 1.5 meters; this gives us realistic shallow-water

[1]The linear form written is based on approximation of the tangent function. For errors less than one meter, the MOOS Trackline controller increases the lead distance proportionally, effectively lowering the gain to limit oscillations.

acoustic performance, but with direct access to GPS and RF wireless connectivity at the surface for conducting controlled tests. Along with messaging, we use the modem for one-way travel-time ranging [73]. For messaging, the Micro-Modem has six different packet types with different lengths and data capacities. In **Chapter 3** we use the FSK mini-packet ("MP"), which is regarded as the most robust of the packet types, but contains only thirteen bits of information. The mini-packets take slightly over one second to transmit. We also use the full-sized Rate 0 FSK packets ("FSK0"), which carry thirty-two bytes of information and take approximately five seconds to transmit. All Micro-Modem packets are sent with an acoustic source level of 190 dB rel μPa.

We note that the upcoming MicroModem 2 will include new flexible PSK mini-packets, which are available at different PSK data rates and are flexible in terms of the packet size. This capability will help give more options when designing a system, as it fills the gap between the 13-bit minipacket and the full-sized FSK and PSK packets in terms of latency, packet size, and reliability.

The experimental work in this thesis took place in the Charles River Basin, shown in **Figure 1-4**. This domain has fresh water 2-12 m deep, a complex bathymetry, and some hard surfaces on the boundaries (seawalls and bridges); our working space is about 1500 m long and 500 m wide. Acoustic performance in this environment is different from an open-water deep ocean scenario, where multipath and reverberation are much lower, but the ranges are higher. Operations in the Basin can have highly variable acoustic performance, as shown in **Figure 1-5**. Our conditions are multipath-limited and travel times are short.

We use TDMA scheduling for multi-vehicle communications. Scheduling and timing is especially important for closed-loop control. We enforce the fixed time slots with a number of timeouts, as indicated in **Figure 1-6**. We synchronize clocks using the network time protocol; in the absence of clock synchronization, we note that precision clocks are becoming increasingly practical for use on underwater vehicles [73].

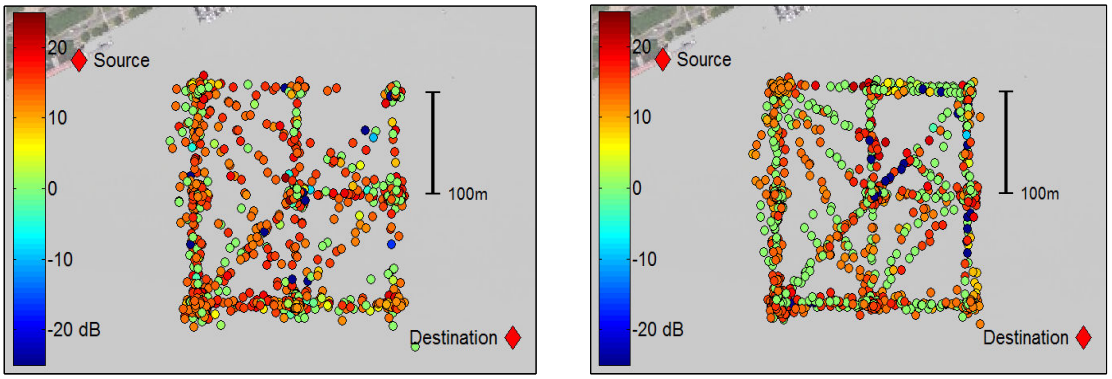


Figure 1-5: Micro-Modem performance data in the Charles River Basin, an environment limited by multipath, not power. The left plot shows transmissions from the source to a mobile relay, and the right plot shows transmissions from the relay to the destination. The SNR value indicates sound pressure level relative to ambient noise.

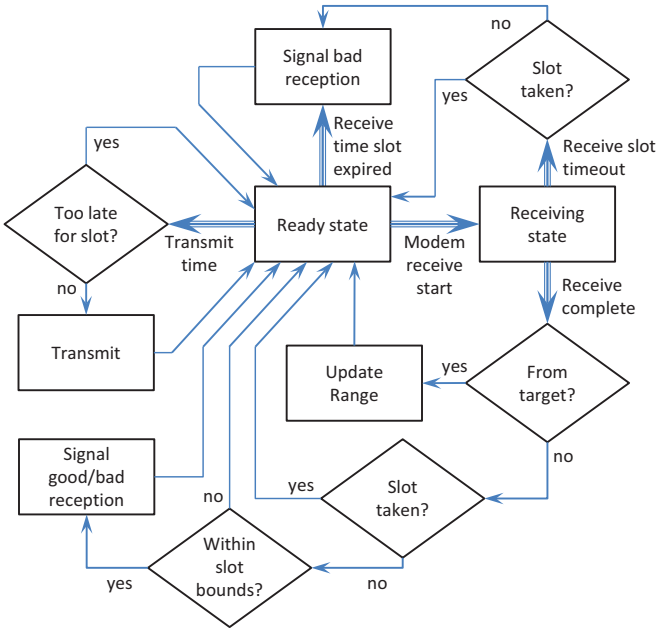


Figure 1-6: The internal state machine used on each vehicle to maintain consistent timing with respect to predefined transmission and reception slots. Thick arrows distinguish acoustic events that initiate state changes or other actions from normal logic flow. Special operations are indicated to handle detection of erroneous multipath receptions, which frequently occur in this environment. For example, a good reception for a time slot T_i will follow the "Receive complete" path (bottom) to a good signal. A trailing multipath reception will return to the receiving state, but the end of time slot T_i will arrive before the end of the packet. In the top right, slot T_i is already taken by the good reception, so we return to the ready state with no action taken.

1.4 Overview of Approach

This thesis considers networked control of multiple-vehicle ocean systems, presenting contributions towards truly dynamic integrated missions in the presence of severe communication constraints. The contributions are aimed towards implementation and experimentation with real-world testbed systems.

Although navigation is challenging and important for all underwater vehicles, we focus on the multi-vehicle control aspects and *consider navigation as a given.* The uncertainty of whatever navigation system may be used is incorporated into our estimation approach as measurement noise for vehicle positions. Similarly, we consider abstract models of vehicle motion that capture high bandwidth low-level dynamic control onboard the vehicle. The effects of disturbances as well as unmodeled dynamics (of the vehicle under low-level control) are considered process noise for our estimation purposes.

We consider communication constraints using the abstraction of packet loss, delays, scheduling, and quantization. We do not consider optimization of the lower layers of the communication system—choices such as modulation, channel equalization, and error correction coding. Except for the specific range-based target pursuit experiments in **Chapter 3**, we do not consider quantization of sensor commands, as this is a highly application-specific problem. In **Chapters 5-6**, our main focus with quantization is the tradeoff between different control packet types (broadcast vs. individual), as well as the tradeoff between control quantization and control packet loss.

We use a centralized control architecture (as diagrammed in **Figure 1-1**) for a number of reasons. First, our vision for a comprehensive ocean monitoring system involves integration of networked control with lower-frequency procedures in data assimilation and model forecasting. These procedures are computationally-intensive and draw on data from many sources, making a centralized computation center attractive. Second, ocean systems are expensive, and operation is risky. Operators prefer to have the ability to watch over the system and intervene if necessary, which

is much more difficult with decentralized control architectures. Additionally, it is often beneficial to blend expert human input (*e.g.* from oceanographers) with autonomous capabilities. The third reason is that many decentralized control schemes are designed based on centralized methods as starting points. As multi-vehicle systems grow in numbers and spatial coverage, network constraints will begin to drive control in a more decentralized direction.

Regarding control techniques, we note that many more theoretical results of networked control consider stability, versus performance. The ocean monitoring systems we consider are not unstable in the traditional sense—vehicles will not be accelerating arbitrarily fast across the ocean. These systems are more concerned with estimation accuracy, reference following and disturbance rejection. If performance in these metrics is low, a form of practical instability can be observed where the feature or target to be tracked is lost. However, networked control results on stabilizing unstable systems are not directly relevant. Due to this reason, as well as the lack of networked control results that consider all of the communication constraints present in acoustic communication networks, our control approach is to build a practical and effective framework that handles all of the aspects we desire, while sacrificing some theoretical rigor in order to do so. We demonstrate the effectiveness of our approach empirically in field experiments and simulations.

1.5 Summary of Contributions

While the rest of this thesis focuses on dynamic missions with multiple cooperating vehicles, **Chapter 2** introduces some fundamental aspects of acoustic communications, and presents a robust approach to a major acoustic networking problem: multicast routing and power control. Specifically, we consider the minimum energy wireless transmission problem [MET], augmented by the practical condition that constraints on link power must be satisfied in probability. For this, we formulate the robust counterpart of the multicommodity mixed-integer linear programming (MILP) model from Haugland and Yuan [112], and derive scaled power levels that account for uncer-

tainty. While not undertaken in this thesis, the optimization approach for routing and power control is suitable for future integration with control and scheduling design. This chapter is based on work published in [199].

In **Chapter 3** we address through experiments the capability of acoustics to sustain highly dynamic, multi-agent missions, in particular range-only pursuit in a challenging shallow-water environment. As opposed to a traditional control and estimation design scenario, the mission here is accomplished through a highly integrated vehicle system performing full joint estimation and coordination through lossy, rate-limited acoustic communications underwater. The three experimental configurations studied show the effects of cycle time, quantization, and acomms performance on the frequency response of the system. In particular, we show that for tracking highly dynamic targets it is beneficial to trade-off quantization for low cycle time. These outcomes show definitively that aggressive dynamic control of multi-agent systems underwater is tractable today. This chapter is based on work published in [198].

Chapter 4 presents an integrated framework for "Oceanographic Pursuit"—joint estimation and pursuit of dynamic features in the ocean, over large spatial scales and with multiple collaborating vehicles relying on limited communications. We present a unique multi-vehicle frontal point description and control methodology that leverages numerical ocean model forecast ensembles. Our primary innovation is a projector algorithm that carries out linearization of ocean model forecast uncertainty directly in vehicle coordinates via a forward model technique. The outcome is a clean stochastic system representation that captures coupling between sites and is suitable for advanced techniques in networked control. Simulations using three model datasets demonstrate the proof-of-concept. This chapter is based on work published in [196] and [197].

Chapter 5 presents a unified formalism for multi-vehicle control and estimation with control, measurement, and acknowledgment packets all subject to schedules, delays and packet loss. The modular framework is built around a jump linear system (JLS) description that includes Packetized Predictive Control (PPC), a technique that combines the receding horizon optimization of Model Predictive Control with

buffering at the actuator. Integration of these elements enables synthesis of a novel technique for estimation using delayed and lossy control acknowledgments—a desirable and practical capability of fielded systems that has not been considered in work to date. This chapter describes the framework, the estimation and control technique, a simple illustrative example, and a few possible extensions.

In **Chapter 6** we present simulation and field experiments demonstrating the JLS-PPC controller in pursuit missions. The field experiments use three autonomous surface vehicles towing acoustic modems, tracking a simulated feature. To focus on control performance, "hybrid" measurements are created using the vehicles positions and simulated gradients. The acoustic communications are fully realistic, using TDMA scheduling and quantized packets, and subject to packet loss. We also present simulation results demonstrating the performance improvements of JLS-PPC over independent vehicles, comparison of two schedule paradigms, and scalability to larger fleet sizes. A design tradeoff study between control quantization and packet loss is demonstrated using the simulation framework, and finally, we present results showing the benefits of using piggybacked ACKs.

We conclude and summarize the contributions of the thesis in **Chapter 7**, along with a discussion of areas of future work.

Chapter 2

A Centralized Optimization Approach for Robust Minimum Energy Multicast Routing and Power Control

Most uses of acoustic modems have been in static or quasi-static deployments, where energy, range, and time-averaged throughput are the major considerations. While the rest of this thesis focuses on dynamic missions with multiple cooperating vehicles, this chapter presents a robust approach to a major acoustic networking problem: multicast routing and power control. Multicast is an important component of vehicle networks as it is often beneficial to send commands to many vehicles at once to save both time and energy. The multiplexed schedule in **Chapter 5** is an example of the use of broadcast in a multi-vehicle control system. Furthermore, acoustic modems have recently become more prevalent on small low-power vehicles such as gliders, where energy considerations are very important [86].

In this chapter, we consider the minimum energy wireless transmission problem [MET], augmented by the practical condition that constraints on link power must be satisfied in probability. For this, we formulate the robust counterpart of the mul-

ticommodity mixed-integer linear programming (MILP) model from Haugland and Yuan [112], and derive scaled power levels that account for uncertainty. Our main result is that the deterministic formulation with these scaled power levels recovers exactly the optimal robust solution in the absence of correlations, and therefore allows for efficient solution via MILP. The approach developed here relies on centralized global optimization, which is more amenable to mobile networks than iterative algorithms where convergence may be difficult while nodes are constantly moving. While not undertaken in this thesis, the optimization approach for routing is suitable for future integration with control and scheduling design.

2.1 Introduction and Prior Work

With underwater acoustic communications, range and data throughput depend on modem power and carrier frequency [228], and as a result, ocean network deployments are often over-powered or limited in scale to improve robustness. However, excess power causes interference and depletes limited energy sources in untethered vehicles and nodes [180].

This chapter considers underwater acomms routing with power control via a centralized robust approach, with emphasis on multicast. While the large size and ad-hoc nature of many RF wireless applications motivate distributed routing methods based on network discovery [206], the high latency and unreliability of acomms suggests that these algorithms could exhibit poor convergence in the underwater domain. However, centralized optimization requires that all data go to the central location which itself uses communications resources. Considering large-scale ocean missions, data assimilation and planning are typically centralized today and the marine assets are expensive and tracked carefully [195]. These aspects of acomms and ocean missions motivate optimization methods which can take into account motion plans, global channel information, and operator input [119].

Wireless network design via centralized approaches is of course a rich and active area of research. Convex optimization for routing in multi-hop RF wireless networks

is presented in [55]; see also [12] for an approach specific to acomms. These works do not consider robustness, however. Most prior work in robust network design, *e.g.* [9], has considered uncertainty in traffic demands. Chang *et. al.* consider robustness to uncertain packet success rates in lossy network coding subgraph generation [44]. Regarding power control in routing, several non-robust, acoustics-focused approaches have been proposed, including [127]. Quek *et. al.* consider robust power allocation for two-hop RF wireless relay networks [188] for a single source to single destination, using multiple two-hop relay channels. Our approach shares the idea that power can be traded off for robustness; we note that for acoustic communications, this tradeoff is most clear with low-rate FSK modulation. Other factors such as time-varying multipath become more important for higher rate techniques.

In this work, we consider multicast over arbitrary numbers of hops using acoustic channel models. Although acomms possesses the broadcast advantage, multicast has received little attention in underwater acoustic networks [171]. We base our approach on the multicommodity MET-F2 formulation by Haugland and Yuan [112], and the main idea is to use robust convex optimization to account for uncertainty in required power levels for acomms. We give the problem statement in **Section 2.2**. Stochastic acomms models motivated by data are discussed in **Section 2.3**. The supporting formulations are outlined in **Section 2.4**, and our new approach for Robust MET is presented in **Section 2.5**. **Section 2.6**, we show that the deterministic formulation with properly-scaled power data can be used to solve the robust problem. We present computational results in **Section 2.7**, and discuss conclusions and some realistic extensions to our formulation in **Section 2.8**.

2.2 Approach and problem definition

We consider a single source transmitting to multiple destinations, and design minimum-power broadcast trees and node power levels which meet individual connectivity requirements with a specified probability. Node locations are considered static and known; the primary sources of uncertainty are in transmission loss and noise at the

receiver and transmitter. While we recognize the importance of protocol effects, we do not consider link throughput rates, impacts of interference on medium access, nor correlated uncertainty across links in this work in order to focus on the key aspects of robust minimum-power routing. However, the formulation of Robust MET via convex optimization is a key underlying construction onto which protocol aspects may be added and analyzed.

Since we are designing power levels at the nodes, we choose to model uncertainty in the transmit power necessary to achieve a minimum SNR at the receiver: $p_{ij} = \bar{p}_{ij} + \tilde{p}_{ij}$. The mean power for link (i, j) to have successful transmission is $\bar{p}_{ij}$ (the no-uncertainty power), and the normal random variable describing the uncertainty in the power is $\tilde{p}_{ij}$.[1] The mean and variance for each link, along with the desired probability of link connectivity, are inputs to the optimization.

Robust optimization considers the worst-case realization of the random variable p_{ij}; under the assumption of a Gaussian distribution we use the mean power plus a properly-scaled addition to account for uncertainty. Our solution is thus feasible for the worst-case realization within a certain probabilistic bound. We call the mean power plus the scaled power $\hat{p}_{ij}$ and will show in **Section 2.6** that it can be set deterministically.

2.2.1 Definitions

The wireless network is described by a graph $G(V, E)$, where E is set of possible (undirected) edges and V is the set of nodes. The set of directional arcs derived from E is A. The multicast source node is s and the set of destinations is D. The transmit

[1] As will be discussed in **Section 2.4.2**, the assumption of a Gaussian distribution is simply used to formally size the uncertainty sets used in the optimization. Other distributions can be better-suited for acoustic channel variability, and the size of the uncertainty sets could be approximated under different distributions or based directly on data.

power of node i is P_i. Additionally,

$$x_{ij}^t = \text{Flow on arc } (i,j) \in A \text{ for commodity } t \in D$$

$$y_{ij} = \begin{cases} 1 & \text{if the power of node } i \geq \hat{p}_{ij} \\ 0 & \text{otherwise} \end{cases}$$

The x variables are binary and an arc is included in the routing if it has flow for any commodity.

2.2.2 Deterministic Minimum Energy Transmission [MET]

The minimum energy transmission [MET] problem was first introduced in [250], and concerns the optimal node transmission powers and associated routing tree for a wireless single-source broadcast or multicast network. To be consistent with our notation we use $\bar{p}_{ij}$ to denote the deterministic power model. The formal problem statement is:

[MET] Find a power vector $(P_1, P_2, \ldots, P_N) \in \mathbb{R}_+^N$ *of minimum sum, such that the induced graph* (V, E^P)*, where* $E^P = \{(i,j) \in A : P_i \geq \bar{p}_{ij}\}$*, has a path from* s *to each* $t \in D$*.*

Broadcast has $D{=}V\backslash\{s\}$ while multicast has $D \subset V\backslash\{s\}$. The MET problem can be transformed into an equivalent Steiner tree problem and is thus NP-complete [112].

2.2.3 Robust Minimum Energy Transmission [Robust MET]

The robust formulation of MET requires the power constraints, which relate the power P_i at a node to the inter-node minimum power levels p_{ij}, to be satisfied in probability:

$$E^P = \{(i,j) \in A : \mathbf{prob}\,(P_i \geq p_{ij}) \geq \eta\} \tag{2.1}$$

Successful transmission occurs when the power at the receiver exceeds a minimum SNR threshold.

2.3 Acoustic communications model

The unique characteristics of the acoustic communications channel leave many transmission parameters to be optimized, such as center-frequency, bandwidth, frequency allocation, power level, and modulation schemes [228]. Our models are aimed towards practical implementation using currently available hardware. We assume center frequencies, bandwidth, and frequency allocation to be fixed in our propagation models. New versions of the WHOI MicroModem allow for transmit power to be set in the range of 140-150 dB, whereas the standard source level is 185 dB [91].

2.3.1 Mean power model

For our mean power model we use classical descriptions of underwater acoustic propagation, as well as the conversion from sound pressure level (traditionally denoted in acoustics in dB rel μPa) to absolute power in Watts. To reach a threshold SNR of SNR_0 decibels, with ambient noise N_{RX} dB rel μPa, the transmit power in Watts as a function of distance r meters is approximated as

$$\bar{p}(r) = Ar^{\kappa}\left(10^{(\alpha r)/10}\right)\left(10^{(SNR_0+N_{RX}+60-185)/10}\right) + B \tag{2.2}$$

The first term (r^{κ}) is due to spreading ($\kappa = 2$ for spherical), while the second term is a linear approximation of absorption loss in seawater [12]. Following the literature, at 10 kHz, $\alpha = 10^{-3}$ (this corresponds to attenuation of one dB per kilometer). The constant factor that is a function of SNR_0 and N_{RX} represents the desired power at the receiver, and $(60 - 185)$ represents the conversion from dB rel μPa to W, the (-185) is the conversion from dB to W, and the $(+60)$ is due to the 1μPa reference for sound pressure. The linear gain in the transmission loss model A and zero-mean additive term B will be used in the next section. As an example, with $SNR_0 = 20$, and ambient noise of $N_{RX} = 40$, 25 W of transmit power is required to transmit $r = 5$ km. This approximation roughly matches performance which has been observed with the WHOI MicroModem [85].

2.3.2 Uncertainty Models

Uncertainty derives from different types of nodes (static sensor nodes, AUVs, surface ships), different operating locations (harbor, open-ocean, shipping lane) and different ocean conditions (mixing water masses, varying wind/wave conditions, varying bathymetry). These can all affect both the ambient noise at the receiver and the transmission loss. Consequently, we define multiplicative and additive uncertainty on each link: $A_{ij} = 1 + \tilde{A}_{ij}$, and $B_{ij} = 0 + \tilde{B}_{ij}$, with $\tilde{A}_{ij}$ and $\tilde{B}_{ij}$ as zero-mean Gaussian random variables. To first order, multiplicative uncertainty can approximate uncertainty in path loss (large-scale fading), or uncertainty in distance. Additive uncertainty corresponds to ambient noise at the receiver.

References [119, 186] discuss two specific MicroModem datasets which are supportive of the mean power model in **Equation 2.2**, and have a path loss variance in decibels which is constant with distance. Constant variance in decibels roughly equates with our multiplicative uncertainty model in Watts. These data were taken in moderately deep water and in relatively good channel conditions. Conversely, **Figure 1-5** shows data with higher variability obtained in experiments with MicroModems in the Charles River (Boston, MA), a very shallow acoustic environment. Statistical analysis of modem performance in this environment is ongoing work; we note that our formulation can accommodate link-by-link means and variances from any model.

2.4 Supporting Formulations

2.4.1 MET-F2 MILP formulation

Here we summarize a compact integer programming model for MET introduced by Haugland and Yuan [112]; our notation matches theirs. The strength of "MET-F2" over previous formulations comes from multi-commodity flows: each commodity

corresponds to a unique destination. We define a multicommodity flow vector $\boldsymbol{x}$:

$$x_{ij}^t = \begin{cases} 1 & \text{if flow to destination } t \in D\backslash\{s\} \text{ on arc } (i,j) \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

Continuity is defined in a standard way by relating the flows of each commodity, $\boldsymbol{x}^t$, the graph G, and the supply/demand vector $\boldsymbol{b}_{st}$: $\boldsymbol{x}^t \in \mathcal{F}(G, \boldsymbol{b}_{st}), t \in D\backslash\{s\}$, where $\mathcal{F}$ is the set of admissible flows. For each commodity, the source has a supply of one, and the destination has a demand of one. Supplies and demand for each commodity t and node i are set according to:

$$b_{st}(i) = \begin{cases} 1 & \text{if } i = t \\ -1 & \text{if } i = s \\ 0 & \text{otherwise} \end{cases}$$

The multicommodity flow formulation allows for the broadcast advantage to be represented compactly, using constraints which relate the y_{ij} variables to the flows x_{ij}^t using a specific ordering of power levels. For any node $i \in V$, let $\pi_i : \{1, \ldots, N-1\} \mapsto V\backslash\{i\}$ be a bijection such that $p_{i,\pi_i(1)}, \ldots, p_{i,\pi_i(N-1)}$ is monotonically non-decreasing. As shorthand, the subscript (i,k) defines the variables in non-decreasing order of power required, where k refers to the kth-closest node to node i. The formal problem [MET-F2] is given below [112]:

$$\underset{\boldsymbol{y}}{\text{minimize}} \quad \sum_{\{i,j\} \in A} p_{ij} y_{ij} \tag{2.4}$$

$$\text{subject to} \quad \boldsymbol{x}^t \in \mathcal{F}(G, \boldsymbol{b}_{st}), t \in D\backslash\{s\}, \tag{2.5}$$

$$\sum_{l=k}^{N-1} x_{(il)}^t \leq \sum_{l=k}^{N-1} y_{(il)},$$

$$i \in V, k \in 1, \ldots, N-1, t \in D\backslash\{s\}, \tag{2.6}$$

$$\boldsymbol{y} \in \{0,1\}^{|A|}, \tag{2.7}$$

$$\boldsymbol{x} \in \{0,1\}^{|A||D|}, \tag{2.8}$$

where the minimum mean link powers p_{ij}, the sets A and D, the source s, and the ordering π_{ij} are given. The node powers are then set as $P_i = \sum_{j \in V} p_{ij} y_{ij}$.

2.4.2 Robust LP

We start with a standard linear program:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{aligned} \tag{2.9}$$

A deterministic LP uses constraints of the form $\boldsymbol{a}_i^T \boldsymbol{x} \leq b_i$, where $\boldsymbol{a}_i^T$ and b_i are known. The robust optimization framework of Ben-Tal and Nemirovski [24] requires the solution to hold for all constraint parameters in an uncertainty set. We use the second-order cone program (SOCP) formulation from [148], which models $\boldsymbol{a}_i$ as Gaussian random variables and sizes the uncertainty sets such that the constraints are met in probability. We desire:

$$\mathbf{prob}(\boldsymbol{a}_i^T \boldsymbol{x} \leq b_i) \geq \eta. \tag{2.10}$$

The corresponding SOC constraint is:

$$\bar{\boldsymbol{a}}_i^T \boldsymbol{x} + \Phi^{-1}(\eta) \left|\left| Q_i^{1/2} \boldsymbol{z} \right|\right|_2 \leq b_i \tag{2.11}$$

where Φ^{-1} is the inverse cdf of the standard normal distribution. The probability η must be ≥ 0.5, which results in $\Phi^{-1}(\eta) \geq 0$, making (2.11) a valid SOC constraint. Q_i is the covariance matrix of the independent Gaussian random vectors $\boldsymbol{a}_i$; there are no correlations between $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$ represented. Notice that this formulation uses continuous decision variables, while there are binary variables in MET-F2. We will address this in the next section.

2.5 Robust LP for MET-F2

In the deterministic MET-F2 formulation, the P_i variables are not used, since they are redundant with p_{ij} and y_{ij}. In order to pose the problem as a robust LP, we re-introduce them, which allows for P_i to become larger than the mean minimum powers:

$$P_i \geq \sum_{j \in V} p_{ij} y_{ij}. \tag{2.12}$$

Substituting the stochastic definition of p_{ij} from **Section 2.3**, and enforcing the power constraint probabilistically, we require

$$P_i \geq \sum_{j \in V} (\bar{p}_{ij} + \tilde{p}_{ij}) y_{ij}, \text{ with probability } \eta. \tag{2.13}$$

We define the vector of decision variables, with N P_i variables, $|A||D|$ x_{ij}^t variables, and $|A|$ y_{ij} variables:

$$\boldsymbol{z} = [P_1, \ldots, P_N, x_{12}, \ldots, x_{N-1,N}, y_{12}, \ldots, y_{N-1,N}] \tag{2.14}$$

Following the procedure of **Section 2.4.2**, we can manipulate the constraints of [MET-F2] into the form $\boldsymbol{a}_i^T \boldsymbol{z} \leq b_i$, and arrive at a new set of SOC constraints:

$$-P_i + \sum_{j=1}^{N} (\bar{p}_{ij} y_{ij}) + \Phi^{-1}(\eta) \left|\left| Q_i^{\frac{1}{2}} \boldsymbol{z} \right|\right|_2 \leq 0, \quad i = 1, \ldots, N \tag{2.15}$$

For the Robust MET-F2 problem, Q_i is a large matrix with blocks corresponding to the constituents of $\boldsymbol{z}$ (P_i, x_{ij}^t, and y_{ij}). For a given node i, y_{ij} is a singleton vector which we denote $\boldsymbol{y}_i$. Since uncertainty is modeled in the parameter p_{ij}, multiplying the variables y_{ij}, the only nonzero block of Q_i is the one corresponding to $\boldsymbol{y}_i$. We denote this block $Q_{i,yy}$, and restrict it to be diagonal.

With inter-node variances of p_{ij} denoted as σ_{ij}^2, we define the vector of variances from node i to each other node $\boldsymbol{\sigma}_i^2 = [\sigma_{i1}^2, \ldots, \sigma_{iN}^2]$. Thus, $Q_{i,yy} = \text{diag}(\boldsymbol{\sigma}_i^2)$.

The full robust MET-F2 optimization problem is:

[Robust MET-F2]

$$\begin{aligned}
\underset{\boldsymbol{y},\boldsymbol{P}}{\text{minimize}} \quad & \sum_{i \in V} P_i && (2.16)\\
\text{subject to} \quad & (2.5), (2.6), (2.7), (2.8), \\
& -P_i + \sum_{j=1}^{N} (\bar{p}_{ij} y_{ij}) + \Phi^{-1}(\eta) \boldsymbol{\sigma}_i^T \boldsymbol{y}_i \leq 0, \\
& i = 1, \ldots, N, && (2.17)\\
& P_i^{min} \leq P_i \leq P_i^{max}, \quad i = 1, \ldots, N && (2.18)
\end{aligned}$$

This model has two major features. First, the diagonal $Q_{i,yy}$ restriction reduces the second-order cone constraint of the robust counterpart to a linear constraint. In addition to the robust constraint (2.17), we have added maximum and minimum node power levels to this formulation to more accurately describe constraints due to real hardware. Second, the ordering based on power used in constraint (2.6) must be modified to use $\hat{p}_{ij}$ instead of the deterministic (or mean) powers in order to account for the effects of uncertainty. In the next section we show exactly how to set $\hat{p}_{ij}$.

2.6 Analysis and determination of scaled powers

2.6.1 Determination of $\hat{p}_{ij}$

We show that the scaled powers $\hat{p}_{ij}$ are a function of the mean and variance of p_{ij}, and further, that if $\hat{p}_{ij}$ is used as input to the deterministic MET-F2 MILP formulation, the results are the optimal solution to Robust MET.

We assume that the optimal routing y_{ij} has been determined, and define $j^*(i) = j$ s.t. $y_{ij} = 1$; $j^*(i)$ is the node in the routing which requires the largest power for connectivity with node i. The robust constraint (2.17) reduces to:

$$P_i \geq \bar{p}_{ij^*(i)} + \Phi^{-1}(\eta) \sigma_{ij^*(i)}, \tag{2.19}$$

where $\sigma_{ij^*(i)}$ is the standard deviation of the uncertainty for the transmit power of link $ij^*(i)$. Since the objective is to minimize the sum of the node powers P_i, and P_i appear only in this constraint, the inequality (2.19) is tight. The resulting equality relation for P_i allows for substitution of the RHS of (2.19) in the objective,[2] which becomes:

$$\text{minimize } \sum_{i=1}^{N} P_i = \sum_{i=1}^{N} \left(\bar{p}_{ij^*(i)} + \Phi^{-1}(\eta)\sigma_{ij^*(i)}\right) \tag{2.20}$$

The added robust constraint (2.17) has been moved to the objective.[3] The only remaining difference between the constraint sets of the deterministic MET-F2 formulation and the robust version is that the ordering used in constraint (2.6) is different.[4] Robust MET requires ordering based on the scaled powers $\hat{p}_{ij}$, while ordering in deterministic MET-F2 is set based on the deterministic (or mean) powers. However, by the same equality argument as for (2.19), it is clear that:

$$\hat{p}_{ij} = \bar{p}_{ij} + \Phi^{-1}(\eta)\sigma_{ij}. \tag{2.21}$$

Substituting $\hat{p}_{ij}$ for p_{ij} in deterministic MET-F2 results in an equivalent formulation to Robust MET. This is important computationally because MET-F2 (a MILP) solves much faster than the general robust counterpart of a MILP (a MISOCP). We refer to [112] for solution times; networks up to fifty nodes are tractable to solve to optimality today.

The case of a nondiagonal $Q_{i,yy}$ represents correlations, which is outside our current scope. However, correlations could be treated approximately by solving the MISOCP with constraint (2.15), using the ordering based on $\hat{p}_{ij}$ as given above. If it

[2]Substituting the robust definition of P_i into the objective can also be viewed as a special case of the robust optimization approach for cost coefficients with ellipsoidal uncertainty sets by Bertsimas and Sim [28].

[3]This is a simple variable substitution since the inequality is argued to be tight, however this procedure can also be interpreted via Lagrangean duality.

[4]Additionally, if maximum and minimum node power levels are desired, the P_i variables must be retained; the effect on overall problem size and tractability is negligible.

is desired, a fully linear approximation could also be made through the relation:

$$\left|\left|Q_{i,yy}^{1/2}\boldsymbol{y}_i\right|\right|_2 \leq \left|\left|Q_{i,yy}^{1/2}\right|\right|_2. \tag{2.22}$$

2.6.2 Special case: constant multiplicative uncertainty

Multiplicative uncertainty (described by $\tilde{A}$ in **Section 2.3**) which is constant across all links is amenable to further analysis. This model would be valid if all nodes have similar characteristics and the ocean conditions are approximately uniform across the operating region. The uncertainty for link ij in absolute power [W] at the sender becomes a simple fraction of the mean power for the link in [W]:

$$\sigma\left(\tilde{A}_{ij}\right) = \sigma_{ij} = \frac{\bar{p}_{ij}}{\mathbb{C}} \tag{2.23}$$

We show that under these conditions the optimal routing solution (x_{ij}^t and y_{ij}) obtained through the deterministic MET-F2 program with the mean powers $\bar{p}$ is in fact optimal for the robust formulation as well. The node powers are set with a simple linear scaling of $\bar{p}_{ij^*(i)}$ that depends on η, and that the scaling is the same across all nodes.

We insert this model for σ_{ij} into the objective as defined in (2.20) and collect terms:

$$\sum_{i=1}^{N} P_i = \left(1 + \frac{\Phi^{-1}(\eta)}{\mathbb{C}}\right)\left(\sum_{i=1}^{N} \bar{p}_{ij^*(i)}\right) \tag{2.24}$$

Since $\Phi^{-1}(\eta)$ and $\mathbb{C}$ are both constants, it is clear that this objective is the same as the deterministic MET-F2 objective, with a constant scaling factor. With the constant multiplicative uncertainty, the ordering based on $\hat{p}_{ij}$ is the same as the ordering based on $\bar{p}_{ij}$. Thus, this formulation has the same feasible set as deterministic MET-F2 and the optimal solution to Robust MET is:

- The optimal routing x_{ij}^t and y_{ij} from deterministic MET-F2

- Node powers set according to:

$$P_i = \left(1 + \frac{\Phi^{-1}(\eta)}{\mathbb{C}}\right)\left(\sum_{j=1}^{N} \bar{p}_{ij} y_{ij}\right) \tag{2.25}$$

The optimal topology and routing are invariant, but the power levels change with the uncertainty level. This is important practically as only changes in the power scaling parameter must be broadcast to all the nodes as conditions change, as opposed to a complete routing table.

2.7 Computational Results

We ignore absorption losses and present results for the spherical spreading model $\bar{p}_{ij} = Gr_{ij}^2$ in order to be consistent with literature on MET. Results were computed using AMPL/CPLEX. The results we show are all for a single multicast instance with $N = 30$ nodes, and $|D| = 15$ destinations randomly located in the unit square. We present example results for multiplicative and additive uncertainty separately, all with $\eta = 0.99$. We normalize the powers such that the deterministic objective ($\sigma = 0$) has total power of one. We did not set maximum or minimum power levels for any of these cases, in order to focus on the effects of the robust constraints.

2.7.1 Multiplicative uncertainty

The left side of **Figure 2-1** shows the deterministic routing, and the right side shows a scenario where all links going into destinations have a multiplicative uncertainty of $\sigma_{ij} = \bar{p}_{ij}/2$ and all links going into optional router nodes have a multiplicative uncertainty of $\sigma_{ij} = \bar{p}_{ij}/20$. The routing is notably different between the two cases. The deterministic case would be infeasible with uncertainty.

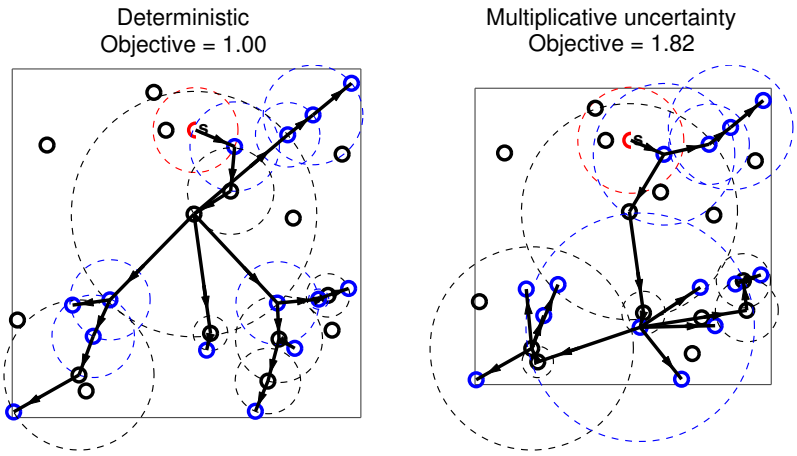
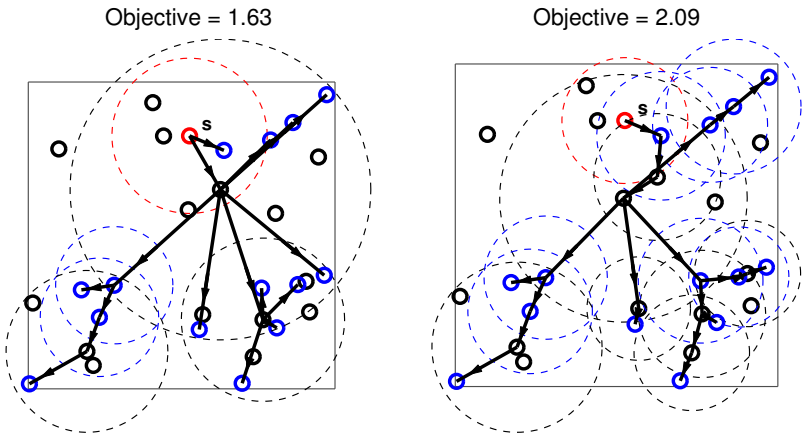


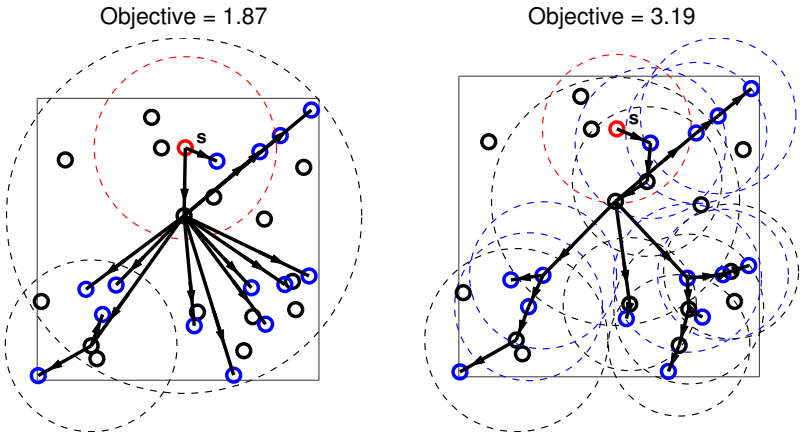
Figure 2-1: The left plot is the deterministic solution (shown for reference). The red node labeled s is the source. The right plot is the solution when destination nodes (blue) have multiplicative uncertainty of $\sigma_{ij} = \bar{p}_{ij}/2$ and optional routers (black) have multiplicative uncertainty of $\sigma_{ij} = \bar{p}_{ij}/20$. Note that the deterministic solution would be infeasible for the scenario with uncertainty.

2.7.2 Constant additive uncertainty

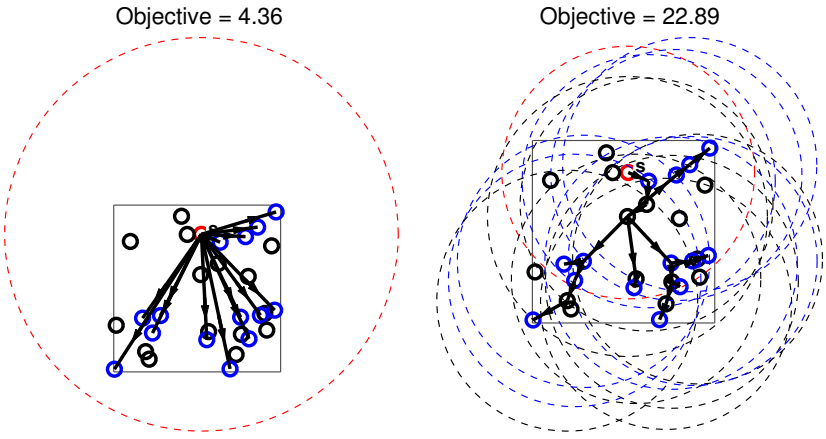
We consider next uncertainties in transmit power for all links as a single constant: $\sigma(\tilde{B}_{ij}) = \sigma_{ij} = \sigma_C$. **Figure 2-2** shows three cases. The uncertainty is normalized such that a standard deviation of one is equal to the power required to transmit the edge length of the domain. The optimal solutions are compared to the prior heuristic, which takes the deterministic design and increases node power levels in order to meet the robust constraints. The heuristic applied in this case is very poor. As uncertainty increases, the true solution moves from the optimal deterministic solution towards a star network. We present results up to large uncertainties to show the extreme behavior of the routing trees. **Figure 2-3** shows a summary comparison. Even at low uncertainty, for $\sigma_C = 1/50$ shown in **Figure 2-2b**, Robust MET achieves an objective which is 41% better than that of the heuristic. We note that the optimal solution is piecewise-linear in between changes in routing and topology, although **Figure 2-3** does not directly show each discrete change.



(a) $\sigma_C = \frac{1}{100}$



(b) $\sigma_C = \frac{1}{50}$



(c) $\sigma_C = \frac{1}{5}$

Figure 2-2: Robust MET solution (left) compared to baseline heuristic (right) for three different values of constant additive uncertainty. $\sigma_C = 1$ corresponds to uncertainty equal to the power to transmit the distance of an edge of the box. The objective is normalized such that the optimal deterministic objective ($\sigma = 0$) is equal to one.

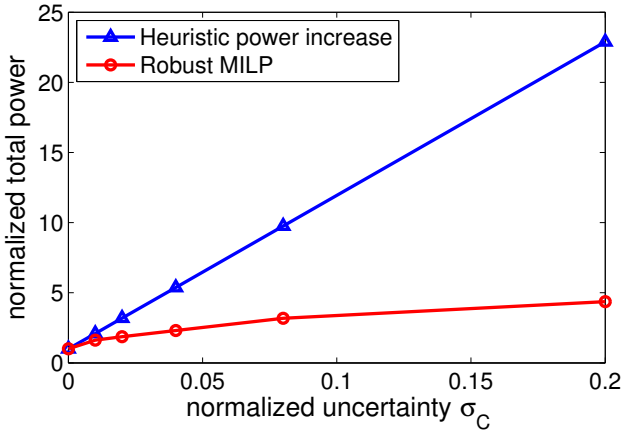


Figure 2-3: Normalized sum of transmit powers as a function of constant additive uncertainty for $N = 30$ and $D = 15$. The total power with no uncertainty is 1. Uncertainty with a standard deviation equal to the mean power required to transmit the edge length of the domain is one.

2.8 Summary

Robust MET provides a tractable means for designing efficient geographic routing subject to power uncertainty, a capability which is especially useful in power-constrained marine robotic networks that rely on unreliable acoustic communications. We have shown that with proper scaling of input power levels, a deterministic MILP formulation may be used to find the optimal robust solution; MILP solvers are faster than mixed-integer SOCP solvers. Additionally, in the case of constant multiplicative uncertainty the deterministic routing solution plus a linear scaling of node powers is optimal. This suggests that the routing table does not always need to be updated as conditions change. In this case or between shifts in topology for arbitrary uncertainty scenarios, adaptive power-control schemes using feedback, such as in [185], could be used for additional performance benefits as the routing is locally optimal.

Chapter 3

Field Experiments in Multi-vehicle Dynamic Target Pursuit

In this chapter we address through experiments the capability of acoustics to sustain highly dynamic, multi-agent missions, in particular range-only pursuit in a challenging shallow-water environment. As opposed to a traditional control and estimation design scenario, the mission here is accomplished through a highly integrated vehicle system performing full joint estimation and coordination through lossy acoustic communications underwater. The waypoint-based control used in this chapter does not consider detailed vehicle dynamics or timing aspects that we focus on in **Chapter 5**. Nevertheless, the three experimental configurations studied show the effects of cycle time, quantization, and acomms performance on the frequency response of the closed-loop system. In particular, the MP and FSK0 experiments demonstrate that for tracking highly dynamic targets it is beneficial to trade-off quantization for low cycle time. These outcomes show definitively that aggressive dynamic control of multi-agent systems underwater is tractable today. More broadly, the pursuit mission presented in this chapter is one special case of a much larger picture where multiple vehicles track *features* in the ocean, as opposed to point targets. We discuss an approach for such missions in **Chapter 4**.

3.1 Introduction

Truly dynamic missions of interest in the ocean include networked ocean vehicles following a submarine or a marine animal; the latter has been a dream of biologists for decades. Major gaps exist in our understanding of the life cycles of many important marine animals, such as jellyfish [202], sharks [224, 245], lobsters [242], and more.

In an effort to lay some groundwork for exploiting advanced algorithms in a real-world ocean application, this chapter addresses with experiments an approach for joint estimation and pursuit of a moving target using acoustic communications; see **Figure 3-1**. Needless to say, the general pursuit problem has held high interest for decades; it is a canonical mission in space and air, on land, and at sea. Probabilistic pursuit-evasion games have been studied extensively in the robotics literature [244], and pursuer and evader dynamics as well as nonlinear estimation are important factors in these algorithms [145, 263]. The effects of communication constraints have not received much attention [166]. These are often addressed indirectly via decentralized approaches that require minimal exchange of information between agents [50]; see [70, 89] for ocean-specific implementations.

There have been some recent experimental works that are related to our pursuit scenario. Perhaps most intriguing is tracking a leopard shark in extremely shallow water, using a single autonomous vehicle with a hydrophone array of 2.4 m spread [51]. The system was successful but the shark evidently moved only 200 m or so in 48 minutes reported. Bean *et al.* (2007) studied range-based leader–follower regulation with Micro-Modem mini-packets and 1 m/s speeds [17], while Brignone *et al.* (2009) study a similar problem with DSPComm modems and two vehicles operating at 0.7 and 3 m/s [33]. Both works present data from proof-of-concept field trials with mostly straight trajectories. Soares *et al.* (2013) consider a vehicle following two leaders in a triangle formation, with ranges of about fifteen meters, speeds around 0.5 m/s, and a total loop time of four seconds [226]. In contrast, Cruz *et al.* (2012) consider a complete feedback system—in the sense of two-way communications—for which a stationary controller transmits commands for two mobile followers, who then transmit

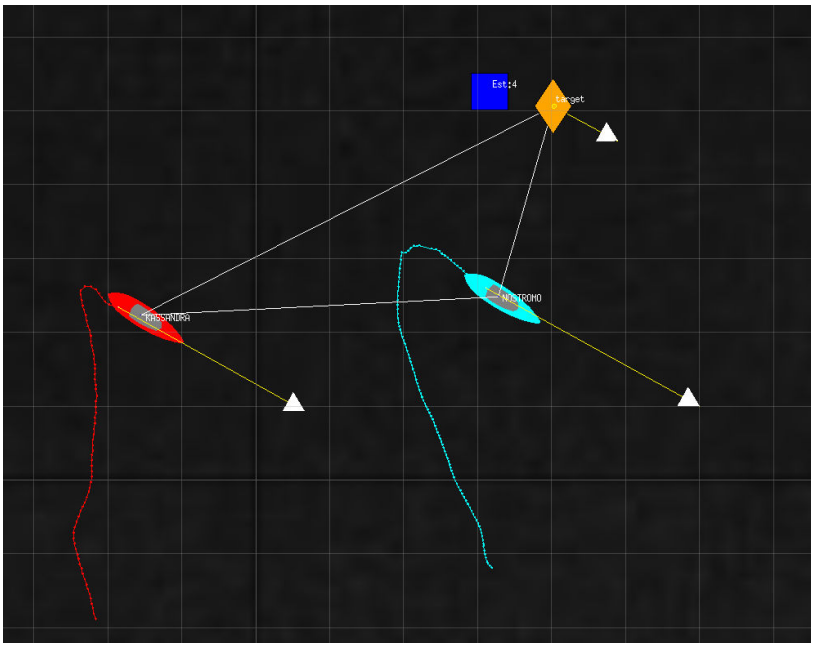


Figure 3-1: Screenshot from an active localization and pursuit experiment with acoustic communications. The two vehicles jointly estimate the target location based on range measurements, and move to stay in formation relative to it.

back their positions [53]. The vehicle speeds are slow, in the neighborhood of 10 cm/s, and the cycle time is around twenty seconds. Through analysis, Chen and Pompili (2012) addressed optimization of acoustic communications in coordinated flight of ocean gliders, where currents are especially important [45].

None of these prior works explicitly deal with designing and improving closed-loop frequency response of an integrated multi-vehicle feedback system. This is exactly our objective here. Our design does not rigorously account for stability margins, the multi-rate nature of acoustic communications, inherent geometric nonlinearities, or the fact that autonomous marine vehicles are not ideal actuators. On the other hand, our approach demonstrates practical closed-loop performance at half the Nyquist rate, with little evidence of stability breakdown.

We detail the experiment setup in the following section with descriptions of the vehicles and communication hardware used, the experimental domain, and the estimation and control strategies and parameters. We then give results from three integrated tests, demonstrating the performance achieved.

3.2 Experimental Setup

Our experiment in joint localization and pursuit has two mobile agents sharing sensor information and commands through acoustic links. We make scalar range measurements at each agent, and thus tracking is impossible without their coordination. One agent is designated as the leader that coordinates the measurements and the actions of the followers. This arrangement involves lossy channels at both locations in the feedback loop of **Figure 3-2**. In the general case, a centralized architecture such as this allows integration with remote sensing, large-scale computations (such as data assimilation), and human-in-the-loop decision-making. The mobile agents attempt to stay close to the target, and in a formation conducive to good sensor performance.

The autonomous surface vehicles and acoustic modem system used are described in **Section 1.3**. The next three subsections detail the arrangement and operation of the multi-vehicle joint estimation and pursuit system.

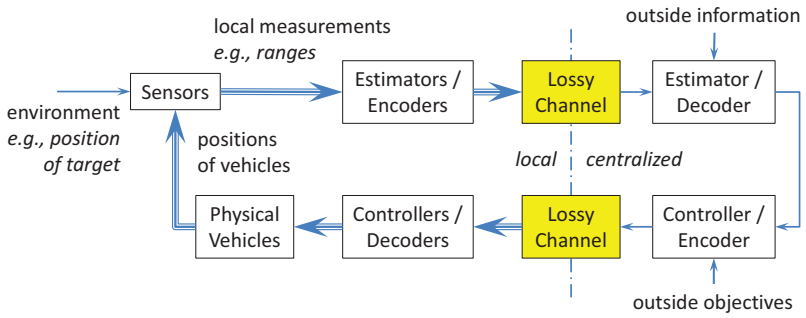


Figure 3-2: Block diagram of a generic multi-vehicle feedback system with a centralized estimator and controller, and communication channels at two locations within the loop. Vehicles act as mobile sensors.

3.2.1 Physical Layout

The two-vehicle pursuit mission encompasses limited communication performance in both the sensing and control channels. In this experiment there is a target to be tracked, "Icarus", and two cooperating agents "Silvana" and "Nostromo". We will denote these three nodes with the symbols $\mathcal{I}$, $\mathcal{S}$, and $\mathcal{N}$, respectively. $\mathcal{N}$ can be thought of as a leader, and $\mathcal{S}$ a follower. The sensing objective is a simple one: to maintain $\mathcal{S}$

and $\mathcal{N}$ in fixed triangular configuration relative to the estimated location of $\mathcal{I}$, so that measurements will be of high fidelity, *i.e.*, in the sense of a good HDOP [29], and in the sense of a short range. Our pursuit arrangement models the general situation where range or other target sensing degrades with distance, but a high level of tracking precision is desired. Maintaining a close pursuit formation keeps ranges close to a nominal value, allowing for more precise quantization.

An "unstable" situation is encountered if the target crosses the baseline (the line in between the two vehicles acting as a moving long baseline network)—the estimate begins to diverge from the target location. Thus, the disadvantage of a small pursuit formation is that it is easier for the target to cross the baseline, bringing up a tradeoff between robustness of a larger formation and accuracy of a smaller formation (which requires good closed-loop performance).

3.2.2 Cycle Description, Timing and Quantization

We detail the stages of the control loop for the MP and FSK0 cases. Within a cycle step, $\mathcal{S}$ and $\mathcal{N}$ each receive a measurement of range to $\mathcal{I}$ via the Micro-Modems in ranging mode. After a guard period, $\mathcal{S}$ transmits its current location and range data to $\mathcal{N}$ through acoustic communication. $\mathcal{N}$ combines this information with its own location and range information to generate an estimated location of $\mathcal{I}$. $\mathcal{N}$ calculates control actions for itself and for $\mathcal{S}$, and transmits the latter back to $\mathcal{S}$. The cycle includes three separate transmissions and there are no acknowledgments.

For feedback control, there is a problem-dependent tradeoff to be made between time-averaged throughput (usually achieved with long coding blocks) and timeliness of the information (shorter messages). We present data using both 13-bit mini-packets and 32-byte FSK0 packets as an initial study of this tradeoff. The MP scenario minimizes cycle time at the expense of data quantization; we achieved a total cycle time of 12 seconds in this configuration.[1] With the FSK0 configuration, packets

[1]When range measurements do not interfere with modem packets and the cycle consists of just two-way communications (*e.g.* using GPS and wifi for ranges), we have achieved a six-second total cycle time with mini-packets in the field.

require no quantization for the data types we send, but do require a 9.5-second time slot for each transmission, resulting in a total cycle time of 23 seconds. The "wifi" scenario involves a four-second slot for acoustic ranging, as detailed above. However, the inter-vehicle communications are handled instantaneously via wifi, so the estimate is available immediately upon reception of ranges.

For the message from $\mathcal{S}$ to $\mathcal{N}$ in the MP case, we used three bits for the range, and five bits each direction for $\mathcal{S}$'s location in a 32×32 discretized workspace; this workspace had ten-meter resolution. The range data were logarithmically quantized relative to a desired range of 50 m, with seven bin edges located at [19.2 32.5 42.5 50 57.5 67.5 80.8] m, and the three-bit messages decoded as [11.5 26.8 38.2 46.8 53.2 61.8 73.2 88.5] m. This correlates with the density $\rho = 0.75$ [80]. For the message from $\mathcal{N}$ back to $\mathcal{S}$, we used five bits in each of x and y for the desired location in the workspace. This left three bits unused. Note that with quantization, there is a tradeoff between range and precision. With this choice, any range larger than 80.8 m is decoded as the furthest range bin, so when ranges are very large, estimation suffers. Increasing this outer range would come at the expense of resolution of the bins near the 50 m nominal range; it is the control system's job to keep the vehicles in the desired formation so that small bins can be used.

3.2.3 Settings and User Choices

The tracking system contains a nonlinear sigma-point filter (SPF) [128], well-suited for this type of application.[2] The nonholonomic target $\mathcal{I}$ (a small motorboat) was assumed to be moving at constant 1.55 m/s, with stochastic low-pass, zero-mean turning rate with variance Q. The observation vector contains the two noisy ranges, with variances $R_{\mathcal{S}}$ and $R_{\mathcal{N}}$ for range measurements to Silvana and Nostromo, respectively. The sensor noise for range measurements was chosen based on prior characterizations of the WHOI Micro-Modem ranging capability [58, 85] and our own observed LBL performance. The sensor noise for the follower range measurement ($\mathcal{I}$ to $\mathcal{S}$) in the MP experiment was set to a higher value to account for the effects of quantization during

[2]Other nonlinear, range-only filters, such as particle filters, could also be used [63].

communication of the measurement from $\mathcal{S}$ to the filter running on $\mathcal{N}$. Settings for the three configurations are given in **Table 3.1**.

When a measurement is not available (either due to a missed LBL range, or a dropped measurement packet from $\mathcal{S}$ to $\mathcal{N}$), we take the standard approach of setting the noise of the lost measurement to infinity [222]. In the MP and FSK0 configurations, when a control command from $\mathcal{N}$ to $\mathcal{S}$ is dropped, the previously-received command for $\mathcal{S}$ remains the desired waypoint. This approach is chosen to ensure safe operation in the case of many missed packets. In the MP case, three bits are left unused in the command packet which could encode contingency plans.

The desired observation triangle has a sixty-degree vertex at $\mathcal{J}$. For the MP and wifi cases, the ranges to each of $\mathcal{S}$ and $\mathcal{N}$ were 50 m; for the FSK0 case the desired ranges were 100 m due to the slower cycle time.[3]

Table 3.1: Settings and results for the three configurations. DesRange is the length of the legs in the desired sensing formation. The columns with R are the sensor noise variances for the range measurements to each vehicle. Q is the target heading rate variance. BW is the closed-loop tracking bandwidth, and Atten is the tracking error attenuation at 0.065 rad/s. Also see **Figure 3-6**.

Config	Cycle Time sec	DesRange m	$R_{\mathcal{S}}$ m^2	$R_{\mathcal{N}}$ m^2	Q $(rad/s)^2$	BW rad/s	Atten dB
FSK0	23	100	0.25	0.25	0.01	0.065	0
Wifi	4	50	0.25	0.25	0.05	0.5	18
MP	12	50	0.25	9	0.05	0.13	7

3.3 Experimental Results

We compare the tracking performance of three different communication configurations: full-sized packets ("FSK0") with negligible quantization and a 23-second cycle, RF wireless communication ("wifi") with a four-second cycle, and 13-bit mini-packets ("MP") with a 12-second cycle. The "wifi" configuration roughly represents a single vehicle towing a long two-element array, as inter-vehicle communication is lossless

[3]The ranges are set relative to the distance the target can drive in a time step, so that the target is unlikely to cross the baseline before the control system can react.

and immediate. However, a true "single-vehicle with array" would be less maneuverable than vehicles without arrays, and could not pursue the target as closely without risking the target crossing the baseline. For close pursuit with multiple vehicles, we can view the "wifi" case as a lower bound on performance.

The experiments we report were conducted on 8-9 July 2013, both days with light winds.[4] **Figures 3-3, 3-4, and 3-5** give results from the FSK0, wifi and MP tests, respectively. In each test, $\mathcal{I}$ moved in a largely random trajectory, as shown in the birds-eye view in the upper left (Subplot **a**) and the time traces in Subplot **c**. The upper right (Subplot **b**) shows the sensing formation every fifteen time steps; we see that while the ideal triangle configuration was rarely achieved in the FSK0 and MP tests, the target did not cross the baseline (the red straight line between the two nodes acting as a moving LBL network), nor did the geometry ever stay poor for a sustained period. The tracking and pursuit system did not lose the target.

The measured ranges are reported in Subplot **e** in each figure, including quantization of raw values sent to $\mathcal{N}$ from $\mathcal{S}$ in the subsequent measurement packet for the MP case. Range losses in all cases are low, as the Micro-Modem ranging ping is fairly robust; see figure captions for loss statistics. Subplot **d** shows the north and east tracking error over time, along with dropped communication packets for the MP and FSK cases. The packet losses are significantly higher for the FSK0 test. Most of the larger errors occur following packet losses, but some large spikes (such as around 500 seconds in the mini-packet test) are not near packet losses—errors can also occur due to poor sensing geometry, and in the MP case, quantization.

Recalling our broad objective to achieve dynamic control through mobile acoustic networks, it is revealing to ask what is the effective closed-loop estimation bandwidth achieved. A direct FFT-based empirical transfer function for the estimation error divided by target motion is shown for each test in **Figure 3-6**; spectra have been smoothed with a five-point centered moving average. The FSK0 test has a break frequency for tracking the motion of $\mathcal{I}$ at approximately 0.065 rad/s, slightly less

[4]This data set, along with videos, is publicly available at `http://web.mit.edu/hovergroup/resources.html` .

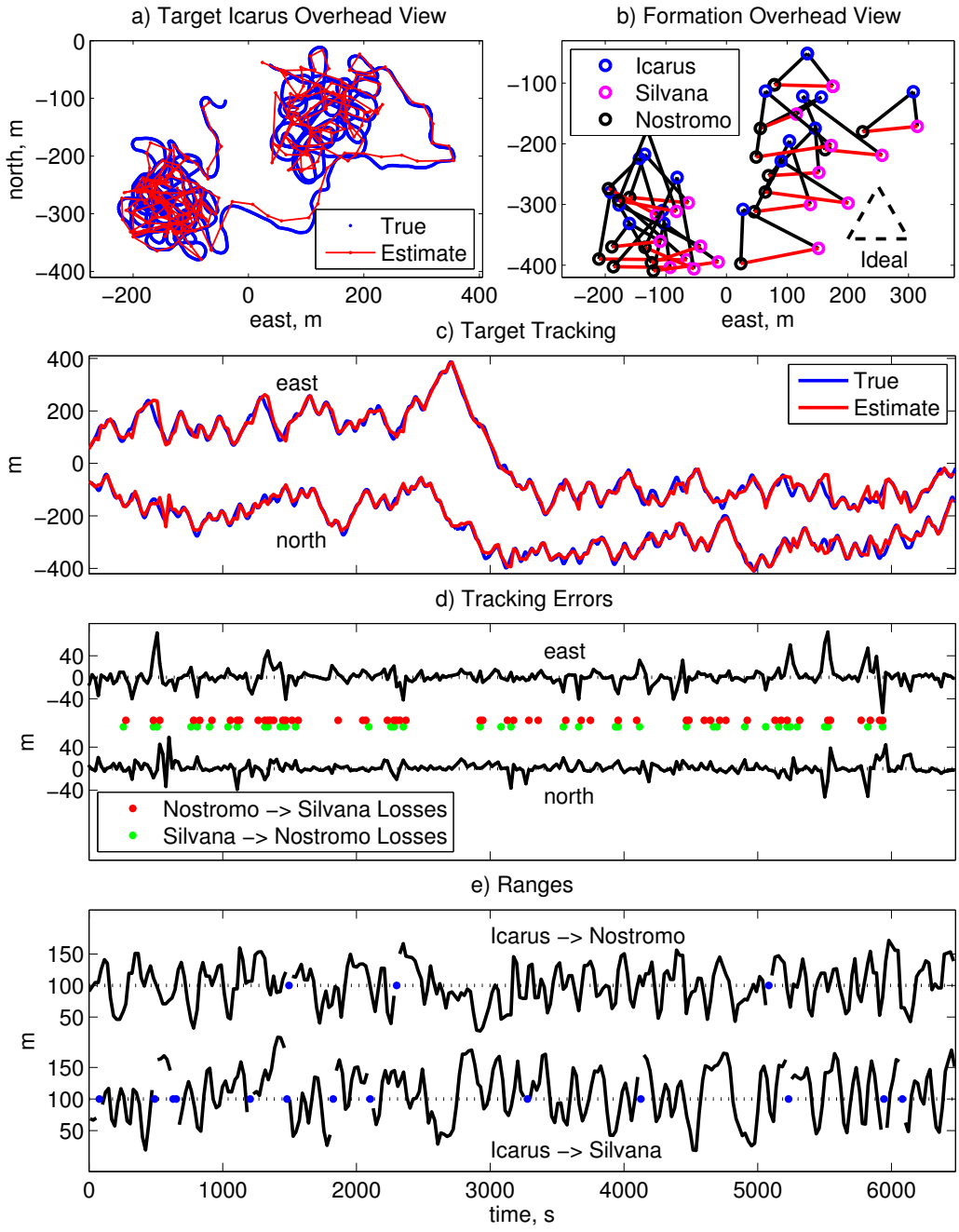


Figure 3-3: FSK0 test results (6463 seconds, 281 cycles). a) Overview of true and estimated trajectories of the target Icarus. b) Sensing formation every 15 time steps. c) Actual (GPS) and estimated trajectory of target Icarus. d) Estimation error of Icarus' location. The RMS radius of estimation errors was 20.2 m. Data packet losses are also shown; loss rates were: $\mathcal{N} \rightarrow \mathcal{S} = 19.9\%$, $\mathcal{S} \rightarrow \mathcal{N} = 14.0\%$. e) Range measurements from Icarus to each kayak, and losses. Range loss rates were: $\mathcal{I} \rightarrow \mathcal{N} = 1.1\%$, $\mathcal{I} \rightarrow \mathcal{S} = 4.8\%$.

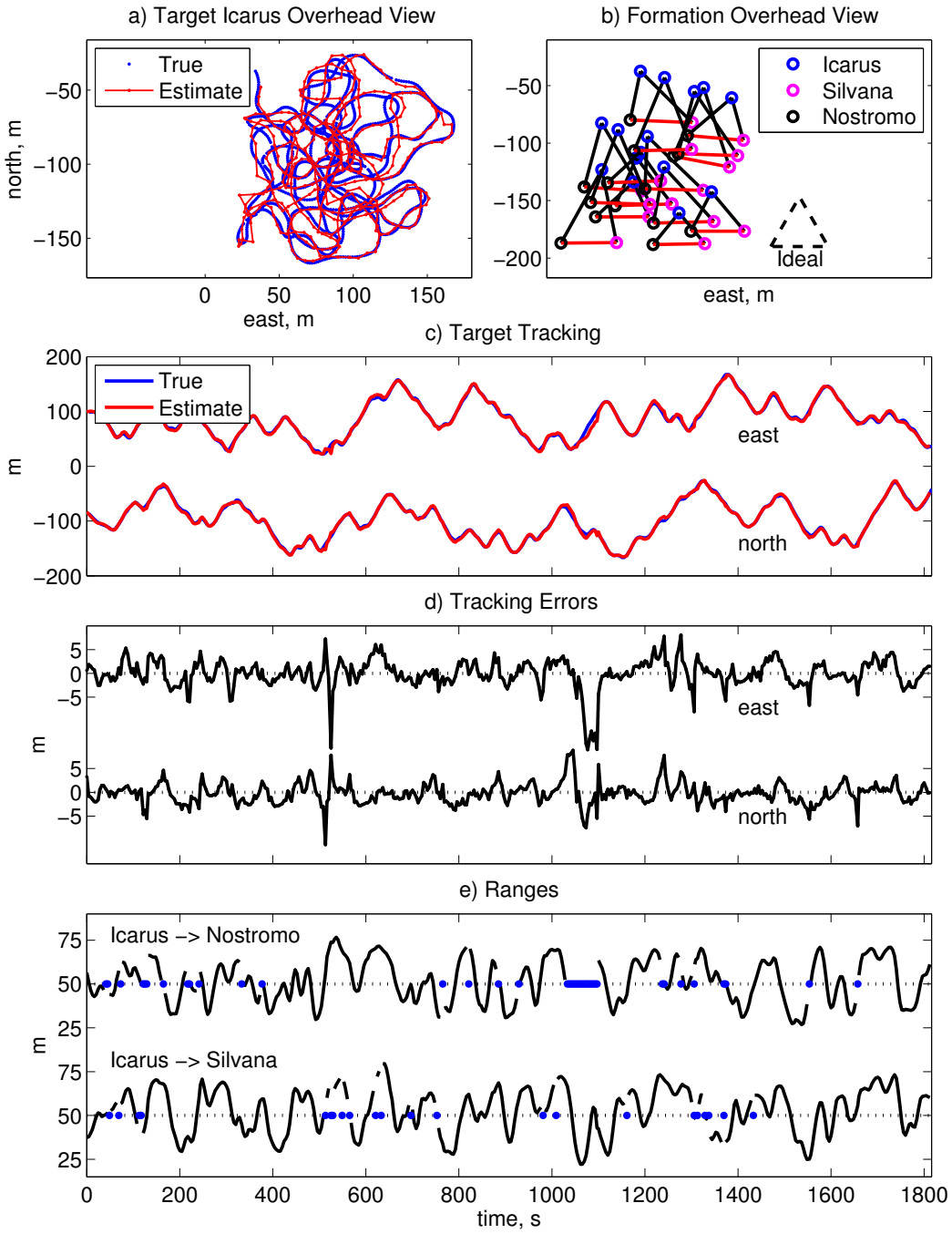


Figure 3-4: Wifi test results (1820 seconds, 455 cycles). a) Overview of the true and estimated trajectories of the target Icarus. b) Sensing formation every 30 time steps. c) Actual (GPS) and estimated trajectory of the target Icarus. d) Estimation error of Icarus' location. The RMS radius of estimation errors was 3.8 m. e) Range measurements from Icarus to each vehicle, and losses. Range loss rates were: $\mathcal{I} \rightarrow \mathcal{N} = 9.0\%$, $\mathcal{I} \rightarrow \mathcal{S} = 4.8\%$.

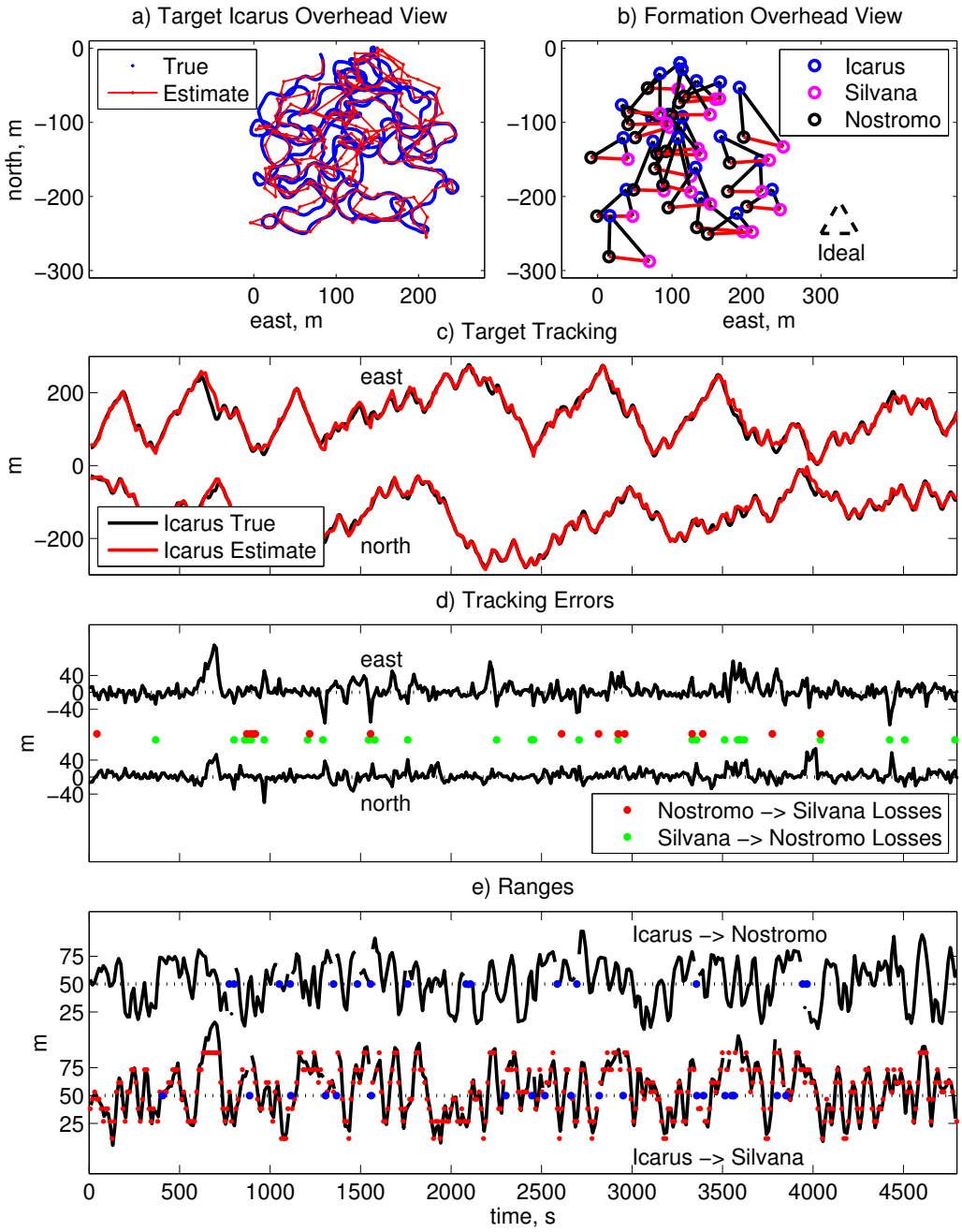


Figure 3-5: MP test results (4800 seconds, 400 cycles). a) Overview of the true and estimated trajectories of the target Icarus. b) Sensing formation every 15 time steps. c) Actual (GPS) and estimated trajectory of the target Icarus. d) Estimation error of Icarus' location. The RMS radius of estimation errors was 12.7 m. Data packet losses are also shown; the loss rates were: $\mathcal{N} \rightarrow \mathcal{S} = 3.8\%$, $\mathcal{S} \rightarrow \mathcal{N} = 6.5\%$. e) Range measurements from Icarus to each vehicle, and losses. Range loss rates were: $\mathcal{I} \rightarrow \mathcal{N} = 3.8\%$, $\mathcal{I} \rightarrow \mathcal{S} = 4.8\%$. Quantized measurements sent from Silvana to Nostromo are shown in red on top of the true measured ranges.

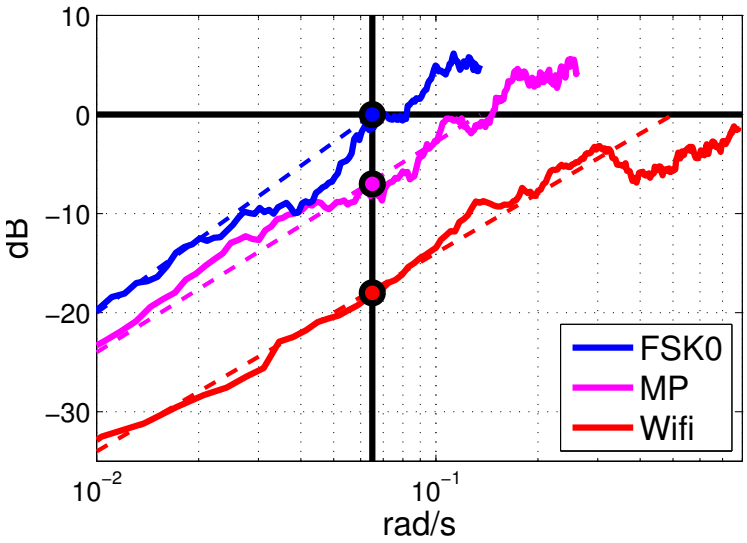


Figure 3-6: Empirical FFT-based transfer function for estimator error divided by target motion. The solid lines show the mean of the X and Y spectra. The dashed lines show an approximate linear fit for low-frequency attenuation. Dots show the approximate attenuation at 0.065 rad/s.

than half the Nyquist rate for the twenty-three-second cycle. The wifi test has a break frequency of approximately 0.5 rad/s. The MP test has a break frequency of approximately 0.13 rad/s. We can also compare the attenuation of tracking error for each configuration at 0.065 rad/s. FSK0 has zero attenuation, wifi has 18 dB attenuation, and MP has 7 dB.

3.4 Summary

Our experiment has achieved aggressive target pursuit in the underwater environment. We performed tracking of a moving target using two vehicles and acoustic range measurements. The vehicles collaborate in order to jointly estimate the target's position, and move to stay in formation relative to it. Real-time communication is an integral aspect of the estimation and control loop. We have presented in detail results comparing the tracking performance of three different communication configurations, at operating speeds near 1.5 m/s. The primary result of the experiments is high bandwidth tracking of the target. Comparing the frequency response of two dif-

ferent communication schemes demonstrates that for tracking highly dynamic targets it is beneficial to trade-off quantization for low cycle time. The results given with the 13-bit minipacket illustrate an extreme end of this tradeoff, and new developments in more flexible short packets will offer further options for exploring the tradeoff.

Chapter 4

Oceanographic Pursuit: Networked Control of Multiple Vehicles Tracking Dynamic Ocean Features

Extending the target pursuit mission of the previous chapter to oceanographic applications, this chapter considers distributed and dynamic tracking of moving ocean features, such as eddies, plumes and fronts. This capability will allow observation of important chemical, biological, and physical processes over larger physical scales and at faster temporal scales than possible with a single vehicle.

This chapter presents an integrated framework for joint estimation and pursuit of dynamic features in the ocean, using multiple collaborating vehicles relying on limited communications. We present a unique multi-vehicle frontal point description and control methodology that leverages numerical ocean model forecast ensembles. Our primary innovation is a projector algorithm that carries out linearization of ocean model forecast uncertainty directly in vehicle coordinates via a forward model technique. The outcome is a linear time-invariant stochastic system representation that captures coupling between sites. Simulations using three model datasets demonstrate the proof-of-concept. While the results of this chapter use the loss-robust control technique of Imer *et al.* [123], the modeling approach and system description we develop is suitable for the JLS-PPC control techniques we present in the next chapter.

4.1 Introduction

The behavior of ocean fronts and similar structures such as plumes and filaments has long been of interest to oceanographers [79, 92]. Recent measurements in a front off Japan have revealed sub-mesoscale structure that figures unexpectedly large in the energy balance [62]. Fronts and plumes are implicated in foundational work on Lagrangian coherent structures [173], and can show dramatic physical, chemical, and biological variability that is critical to understanding ocean-atmospheric coupling, ecological systems, and pollution [37, 78].

Despite continual advances in modeling of complex natural processes, ocean fronts at the mesoscale and smaller remain challenging [41, 111], and hence have emerged as a primary focus area for mobile sensing systems. Here, progress has been rapid, *e.g.* [83, 246]. Zhang *et al.* [261, 262] carried out at-sea experiments where measurements both drove trajectory decisions and triggered collection of large samples. A single vehicle has successfully tracked a plankton bloom [98], while a distributed simulation approach for plume and thermocline tracking is presented in [182]. Supporting all these developments, basic water properties are routinely measured today from mobile robots, while sophisticated chemical and biological analyses *in situ* are becoming mature technologies, for example DNA probes [216] and mass spectrometers [37]. In turn, ocean modeling is becoming integrated with real-time sampling systems, e.g., [110, 225, 251], and is increasingly taking on multi-disciplinary aspects [230]. Path-planning under knowledge of current forecasts has been studied extensively, for example by Smith *et al.* [225] and Lolla *et al.* [149]. None of these works, however, consider cooperation between vehicles, nor communication constraints. Elsewhere, coordinated sampling using drifters and vehicles has been studied in [61, 105], and collaborative control for tracking Lagrangian coherent structures in [160]. But these papers do not address global dynamic models nor communication constraints.

Already exploited regularly in the terrestrial and air domains, networks of mobile agents are an attractive means for tracking and pursuit of dynamic processes over mixed spatial scales [68], although wireless communication inevitably brings funda-

mental challenges in control [14]. Underwater, wireless communication over distances beyond about one hundred meters is made almost exclusively via acoustics, which suffer packet losses caused by ambient noise, multipath, and other environmental conditions [116]. This packet loss, combined with low data rates and long delays, has limited the use of acoustic communications in high-performance, real-time tasks. Our own experience with acoustic modems [198] leads us to assert that control system design should encompass communication limits from the beginning.

To this end, there has been considerable recent work in the field of control under communication constraints. Constructive results exist for lossy estimation [108, 222], lossy commands [192] and H_∞ sampled-data control [138]. We extended the work of Imer *et al.* [123] to the case of independent multi-channel packet losses [196]; Imer's dynamic programming approach results in a highly tractable recursion. These principled methods for networked control design, however, usually require LTI system representations.

In this chapter, we combine the themes above to focus on tracking and pursuit of dynamic ocean fronts by multiple unmanned vehicles, posing the problem in such a way as to accommodate the most promising developments in communication-constrained feedback control. As diagrammed in **Figure 4-1**, our approach fits as an intermediary between high-bandwidth vehicle flight control (at the seconds time scale) and lower-frequency procedures in numerical ocean modeling, assimilation, and adaptive sampling. Notably, we are using linearization for a completely different purpose here than the norm in physical oceanography, where it has helped characterize instability and maximum sensitivity directions through adjoint models [164]. As described in full below, our approach explicitly leverages *ocean forecast ensembles, a projection onto vehicle coordinates, and stochastic system identification*, yielding a dynamics description that is directly suitable for control system design. These elements all enable a *reactive control methodology for dynamically sampling the ocean*, that may be faster than many approaches used today. Looking forward, we hope that our framework may provide a basis for tradeoff studies in designing complex deployments.

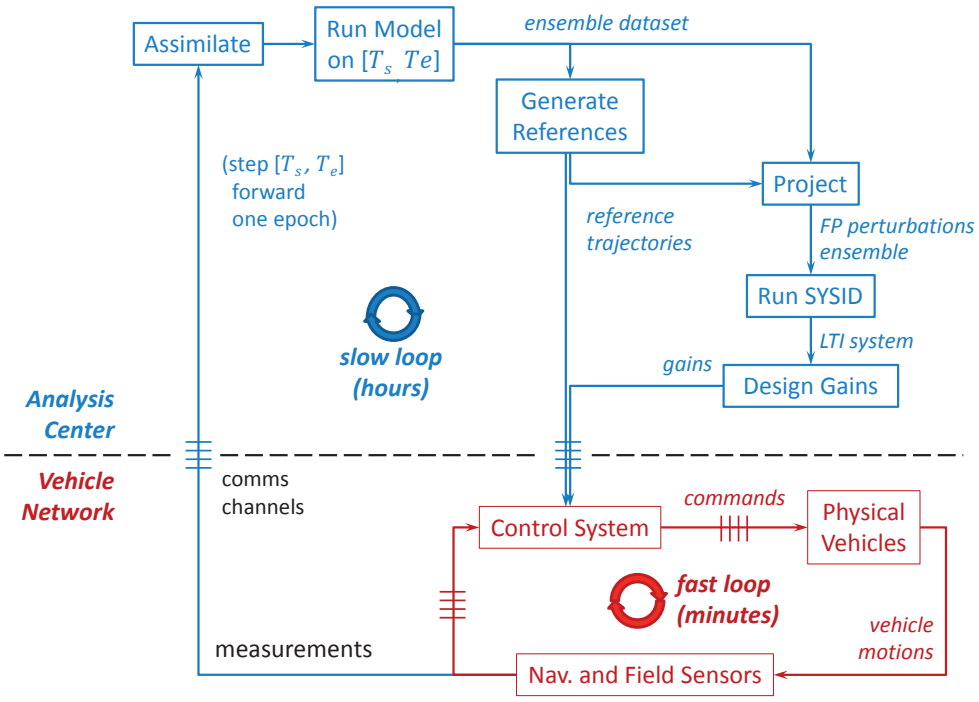


Figure 4-1: High-level overview of the ocean front tracking system. Inputs on the Analysis Center side include human decisions and other data sources not available to the vehicle network; these may or may not be embedded in the projector. Additional inputs on the Vehicle Network side include channel losses, sensor noise, and physical disturbances.

We describe the overall technical framework in **Section 4.2**, with some additional background comments on forecasting and linearization. Projection is detailed in **Section 4.3**, and the integration of projection, system identification, estimation and control in **Section 4.4**. Projection and identification outcomes with test and model data are described in **Section 4.5**, and **Section 4.6** has control simulations. **Section 2.8** concludes.

4.2 Technical Setup

4.2.1 Scope of the Field Operation

We consider a dynamic scalar field ϕ in two dimensions, although there is nothing inherent to our methodology that prohibits three dimensions, or a vector field. We will focus on field variables that can be adequately sampled by autonomous vehicles while they are in motion.

Two key assumptions are that the instantaneous field contains areas of spatial gradient, which induce a favorable measurement gain for data-driven servoing, and that the operation of vehicles on gradients is desirable. This is certainly true if we want to track a front, and characterize the water properties close to it. Beyond following gradients as a primary mission objective, there is also a broader scenario in which one set of vehicles might have the task of monitoring feature boundaries – characterized by a threshold value and a gradient – while another set operates within the feature, where structure is much harder to detect.

4.2.2 Known and Unknown Frontal Points

In loose terms, a front in the field variable at a given time is an elongated region of high gradient magnitude; in the plane, a front would comprise a tightly-packed set of contour lines for field variable $\phi(t)$. A *projector* expresses such a front in terms of desirable locations for vehicles – which we term *frontal points* (FP's) $\boldsymbol{p}(t)$ – and as the physical front evolves over time, so do the frontal points. Taking ϕ and $\boldsymbol{p}$

to denote a finite time series of the field and a set of FP's, respectively, our overall strategy in this chapter is to develop *a process that quickly generates acceptable* $\boldsymbol{p}$ *from given* ϕ, and separately *a process by which an actual vehicle group pursues a physical realization of* $\boldsymbol{p}$. The second process perhaps has a conceptual jump in the sense that we are asserting a physical $\boldsymbol{p}$ to go with a physical ϕ, and of course neither is fully known.

For a given ϕ, frontal points are defined by the projector $\mathbb{D}$:

$$\boldsymbol{p} = \mathbb{D}(\phi).$$

$\mathbb{D}$ can comprise algorithms as well as human input; it encodes any explicit dependence on time, and initial conditions for $\boldsymbol{p}$. $\mathbb{D}$ need not be causal, and as a function could be surjective or injective or both: different field evolutions ϕ could easily lead to the same desired vehicle trajectory, while there also exist useful vehicle trajectories that do not depend on ϕ at all. Most importantly, $\mathbb{D}$ has full access to ϕ and is thus *omniscient*.

Example 1. A causal, omniscient projector for N vehicles might have the following key operations at each time step:

1. Given $\phi(t)$, choose a level Φ and calculate regions of the physical space that satisfy $\phi(t) = \Phi$.

2. On this subset of interest, propagate the N FP's, $\boldsymbol{p}(t)$, with constraints on spacing and on speed.

This example captures the paradigm we explore in computations, but it is by no means the only one possible. For instance, one could just as easily identify areas of high gradients in $\phi(t)$ to put the FP's.

An acceptable $\boldsymbol{p}$ satisfies a number of constraints, $\boldsymbol{p} \in \mathbb{P} = \mathbb{P}_1 \cap \mathbb{P}_2 \cap \cdots$. Already we know from statements so far that:

$\mathbb{P}_1$*:* $\boldsymbol{p}$ *occurs in gradient areas, and observations near* $\boldsymbol{p}$ *are desirable.*

$\mathbb{P}_1$	general desirability of gradient areas
$\mathbb{P}_2$	feasibility
$\mathbb{P}_3$	accuracy and strength of reference gradients
$\mathbb{P}_4$	slowly varying reference FP's and gradients
$\mathbb{P}_5$	consistency of major features
$\mathbb{P}_6$	coupling among perturbations

Table 4.1: Summary of general constraints for projector $\mathbb{D}$.

> $\mathbb{P}_2$: $\boldsymbol{p}$ *is feasible for the vehicles and their control system, in terms of maximum transit speed, maneuvering, closed-loop bandwidth, collision avoidance, and communication or other operational constraints.*

To support closed-loop field operations, we will add to $\mathbb{P}$'s specification below, as summarized in **Table 4.1**.

Consider now the *physical instance* of the field ϕ_o; the "o" subscript is used from here on to indicate the physical instance. $\mathbb{D}$ does not apply because we do not know ϕ_o. Yet desirable FP's $\boldsymbol{p}_o$ still exist, and an estimate of them plays out according to

$$\hat{\boldsymbol{p}}_o^t = \mathbb{E}_o(\boldsymbol{z}_o^t),$$

where $\mathbb{E}_o$ is a causal, actuated estimator, and sparse measurements $\boldsymbol{z}_o$ are its driver. The "t" superscript indicates times up to t. Underlying $\boldsymbol{z}_o$ is the physical instance itself, ϕ_o, and a set of physical vehicles and their control system(s), process and sensor noise channels, and so on. $\mathbb{E}_o$ is understood to embed all available information but it has a structure different from $\mathbb{D}$: it does not know the true field ϕ_o, and thus cannot carry out the first step in *Example 1* above.

Our goal then is to exploit the information available at start of mission to ensure that $\boldsymbol{p}_o \in \mathbb{P}$, and that $\hat{\boldsymbol{p}}_0 \approx \boldsymbol{p}_o$ through a sampling, estimation and control strategy. These are achieved through both $\mathbb{D}$ and $\mathbb{E}_o$.

4.2.3 Designing $\mathbb{D}$ via Stochastic Prediction

Without prior information about ϕ_o the construction from $\mathbb{E}_o$ would be extremely weak. We can use a stochastic model of the ocean process to strengthen it. Consider a set of specific *predicted* instances (realizations) of the field, indexed $i \in \{1, \cdots, M\}$. The associated set of frontal points is generated by the same omniscient projector as above: $\boldsymbol{p}_i = \mathbb{D}(\phi_i)$. Since ϕ_i is in hand, the $\boldsymbol{p}_i$ defined by $\mathbb{D}$ is easy to create and visualize. Thus a predicted instance provides a direct assessment of $\mathbb{D}$, and an ensemble of such instances allows us to design $\mathbb{D}$ such that the constraints in **Table 4.1** are satisfied.

Let the distribution of the true field be $h(\phi_o)$, and make the assumption that $h(\phi_o) = h(\phi_i)$, i.e., there are no systematic errors in the field prediction. Formally, this assertion rests on Leith's landmark paper in climate studies, that by the ergodic hypothesis good performance in an ensemble occurs when the ensemble distribution matches the true climate distribution [141]. This assumption underlies virtually all modern weather prediction procedures, and in our context it implies that $h(\boldsymbol{p}_o) = h(\boldsymbol{p}_i)$, since $\mathbb{D}$ is agnostic on the input data. Then $\text{prob}(\boldsymbol{p}_i \in \mathbb{P}) = \text{prob}(\boldsymbol{p}_o \in \mathbb{P})$, and consequently $\mathbb{D}$ designed via $\boldsymbol{p}_i$ achieves $\text{prob}(\boldsymbol{p}_o \in \mathbb{P}) \to 1$ if the design process fully models random $\boldsymbol{p}$, and if $\mathbb{D}$ yields a high fraction of successful trajectories. In other words, the probability that the physical instance FP's $\boldsymbol{p}_o$ will satisfy the constraints increases as the projector takes more information about the field into account.

4.2.4 Geometry of $\mathbb{E}_o$

Assuming the desirability ($\mathbb{P}_1$) and feasibility ($\mathbb{P}_2$) conditions are met by $\boldsymbol{p}_o$, the next question is how to make $\hat{\boldsymbol{p}}_o \approx \boldsymbol{p}_o$. In this and the next subsection, we consider geometric aspects, from which conditions $\mathbb{P}_3$, $\mathbb{P}_4$, and $\mathbb{P}_5$ will follow. As noted, $\mathbb{E}_o$ includes the vehicle system, and hence following $\boldsymbol{p}_o$ is a complicated function of many parameters as well as random processes. At each time step, however, $\mathbb{E}_o$ can be thought of as having a geometric part that establishes $\hat{\boldsymbol{p}}'_o \approx \boldsymbol{p}_o$ through algebraic constraints only, and a feedback and estimation part that in turn establishes $\hat{\boldsymbol{p}}_o \approx \hat{\boldsymbol{p}}'_o$,

when the vehicles stay close by. This second approximation is essentially set by the closed-loop performance of the integrated system, the topic of a later section.

Regarding the first approximation, suppose for the moment that there are no constraints from the vehicle system, i.e., that all measurements and communications are perfect, and the vehicles' physical motion is unconstrained. All that is left in $\mathbb{E}_o$ is the causal mapping from perfect measurements at locations $\boldsymbol{q}$ to the FP estimates $\hat{\boldsymbol{p}}'_o$. Let us restrict our attention to a single FP and a single time step.

The geometry of $\mathbb{E}_o$ is built on one of two simple linearization models. Consider **Figure 4-2**(middle), which illustrates the first. Suppose that p_o instantaneously satisfies the level condition $\phi(p_o) = \Phi$ plus constraints $p_o \in \mathcal{C} = \mathcal{C}_1 \cap \mathcal{C}_2$.[1] To resolve ambiguities in a complicated field $\mathcal{C}_1$ restricts p_o to the neighborhood of a reference point $\bar{p}$; for uniqueness, $\mathcal{C}_2$ is a line that sufficiently constrains p_o along the level set. The intuition is that if this FP coincides with a strong gradient, and if tracking errors are small, then a vehicle located at q would be able to directly servo to p_o, by measuring $[\phi(q) - \Phi]$ and enforcing $q \in \mathcal{C}_2$. Note that $\hat{p}'_o$ as drawn is not at p_o because the estimator does not know the true level set.

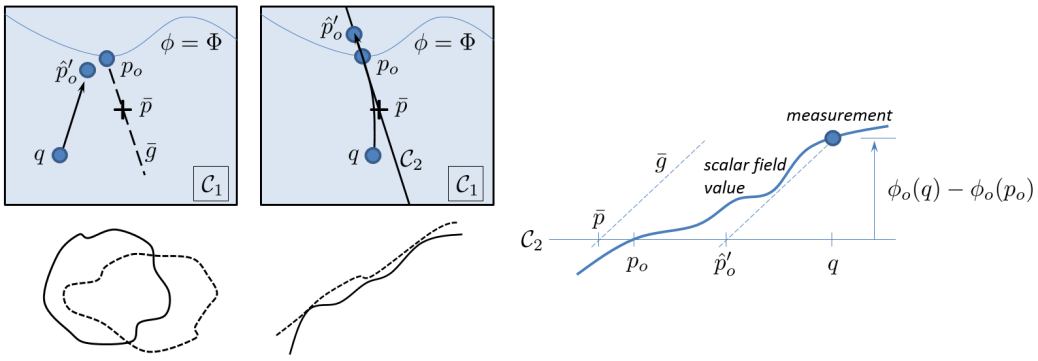


Figure 4-2: Spatial linearization geometry. **Left boxes:** The pursuit control system should drive a vehicle at q to the estimated frontal point $\hat{p}'_o$. The top boxes show two linearizations for $\mathbb{E}_o$; the right one has a line constraint $\mathcal{C}_2$ and is our main focus in computations. The left box has a reference gradient imposed and thus requires some additional constraint. The bottom sketches illustrate the perturbation type for which each linearization is suited; the solid line is the reference feature and the dotted line is a physical instance. **Right figure:** Geometry along $\mathcal{C}_2$ (*e.g.* the reference gradient direction) illustrating how $\bar{g}$ establishes $\hat{p}'_o$.

[1] As a slight abuse of notation, we will use p_o and q to indicate 2-vectors of coordinates in $\mathbb{R}^2$; boldface will be used to refer to sets of FP's and vehicles, respectively.

From the stochastic prediction let us define a reference field $\bar{\phi}$ and an associated set of reference FP's $\bar{\boldsymbol{p}}$. One choice, described in **Section 4.3.1**, is to set $\bar{\phi}$ as the ensemble mean, with reference FP's equally spaced along a level set. Note that although $\bar{\boldsymbol{p}}$ does depend on a rule set, we do not write $\bar{\boldsymbol{p}} = \mathbb{D}(\bar{\phi})$, because $\mathbb{D}$ operates on instances only, and actually contains both $\bar{\boldsymbol{p}}$ and $\bar{\phi}$ implicitly. From the stochastic prediction we also define a reference gradient vector $\bar{\boldsymbol{g}}$ to go with $\bar{\boldsymbol{p}}$; both $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{g}}$ can change over time. Now each linearization at a site in $\mathbb{E}_o$ uses the nominal gradient $\bar{g}$, which is supposed to apply in the neighborhood of p_o. This is a strong assumption, because gradient variability can be quite large. A robust stochastic prediction serves as an adequate indicator, however, and our simple approach of using $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{g}}$ directly is shown to be effective for the datasets described in **Section 4.5**.

Some simple manipulations give a key analytic relationship, illustrated in **Figure 4-2**(right):

$$\begin{aligned} \phi_o(q) - \phi_o(p_o) &= \bar{g}^T(q - \hat{p}'_o) \\ &= \int_{p_o}^{q} \nabla\phi_o(s)^T ds, \text{ to yield} \\ \int_{p_o}^{q} (\nabla\phi_o(s) - \bar{g})^T ds &= \bar{g}^T(p_o - \hat{p}'_o), \end{aligned} \tag{4.1}$$

where s denotes any path in $\mathbb{R}^2$ from p_o to q. The formula is exact, and says that the error in extrapolating a measurement at q to the anticipated measurement at (unknown) p_o is equal to the projection of the reference gradient onto the error in FP position. It offers two crucial insights. First, with all of q, p_o and $\hat{p}'_o$ assumed to lie within $\mathcal{C}_1$, for the error to be small we need $\nabla\phi_o$ to be near $\bar{g}$ in $\mathcal{C}_1$, at least in the average sense and at least in the direction of $q - p_o$. Gradient error that is orthogonal to $q - p_o$ has no bearing on the FP error. Second, stronger $|\bar{g}|$ makes for a smaller error in $\bar{g}$'s direction. Error orthogonal to $\bar{g}$ is not controlled by the integral (nor by $|\bar{g}|$), and this is why we require the constraint $\mathcal{C}_2$ or coupling in our models.

Example 2. To illustrate specific $\mathbb{D}$ and $\mathbb{E}_o$, suppose there are two vehicles and that $\mathbb{D}$ codes for one vehicle to follow the northernmost extreme of the level set $\phi_i = \Phi$,

and the other vehicle to follow the southernmost extreme. This is an easy program to write if ϕ_i is known. Assume $\bar{\boldsymbol{p}}$ is representative of each instance; then $\mathbb{E}_o$ fixes the east-west coordinate for each vehicle according to that of $\bar{\boldsymbol{p}}$ – this is the constraint $\mathcal{C}_2$ at each site – and drives each vehicle north or south depending on the error signal $[\phi_o(q) - \Phi]$ and some control law.

We now state more specifically our two major categories of linearization;

1. The first involves perturbations relative to a reference point and its reference gradient; see **Figure 4-2**(middle). Perturbations are only allowed along the reference gradient line through the reference point. This constraint is suitable when deviations perpendicular to a level set are the dominant effect, e.g., undulations of a long front.

2. The second linearization allows unconstrained translations relative to the reference FP, but maintains the reference gradient, as in **Figure 4-2**(left). This is useful when the shape of the tracked feature is consistent but its translation is not. In the absence of a given lateral constraint, this linearization requires coupling between FP's, which will be described below.

In this chapter we give application-related studies concentrating on the first linearization.

As an aside, we note that range-only tracking and pursuit of a point target can be seen as a special case of the second linearization, in which speed limits and nonholonomic constraints for the target are replacing the ocean model. That the observing vehicles have to maintain a wide aperture, and a suitable distance for sensing and communication, comprises a simple rule set within the real-time system $\mathbb{E}_o$. This is the scenario studied in **Chapter 3**.

4.2.5 Gradients for $\mathbb{E}_o$

An instance ϕ_i induces p_i at each site through $\mathbb{D}$, and the associated gradient is $\nabla\phi_i(p_i)$. If this gradient is close to the reference gradient $\bar{g}$, then we can use the latter in the real-time operation. Desirable properties for $\mathbb{P}$ in this regard are:

$\mathbb{P}_{3a}$: *$\|\bar{g}\|_2 \geq \kappa$ for each FP. The reference gradient is strong, improving SNR of a noisy field measurement.*

$\mathbb{P}_{3b}$: *$\|\nabla\phi_i(p_i + \Delta) - \bar{g}\|_2 \leq \epsilon\kappa$ for each FP, where Δ is a positioning error, and $0 < \epsilon \ll 1$. The instance gradient is close to the reference and is spatially robust.*

Both of these properties conform to minimizing the error $\hat{p}'_o - p_o$ in **Equation 4.1**, and also relate to control system performance, as discussed in **Section 4.6.2**. An additional issue is that $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{g}}$ can vary over the prediction horizon. Consistent with control systems practice, we require that these variations are slow enough that they do not interfere with the closed-loop system near its break frequency:

$\mathbb{P}_4$: *$\bar{p}$ and $\bar{g}$ for each FP are slowly-varying relative to the control system.*

Despite best efforts to corral the frontal points through advantageous trajectories, they may still fail to capture the dynamic behavior of interest. For example, suppose that a simple, positive-sloping front like that shown in **Figure 4-2**(right) develops a local minimum in the vicinity of $\bar{p}$, leading to an "S"-shaped slice. We have observed this and other failure mechanisms in our work with ocean model data, and they lead to a general constraint on feature consistency which is intrinsically related to the gradients:

$\mathbb{P}_5$: *An instance $\boldsymbol{p}_i$ stays on the major features of interest.*

4.2.6 Dynamics Linearization

Our stochastic understanding of the field is a reference trajectory $\bar{\phi}$ and a limited variation of it ϕ_o. This applies to the FP's as well, with $\bar{\boldsymbol{p}}$ and $\boldsymbol{p}_o$. For the purpose of designing and implementing an observation system, we now assume that the perturbations $\boldsymbol{p}_o - \bar{\boldsymbol{p}}$ have a locally linear dynamic behavior, driven by both known inputs (e.g., wind forcing if the estimate is accurate) and unknown (long-term nonlinear behavior of ocean flows). Linear dynamics brings access to mature and scalable

multi-variable identification, analysis and synthesis tools, but the assertion is clearly a tradeoff against the descriptive power of fully nonlinear modeling. We think the tradeoff worthwhile since communication constraints and navigation are preeminent problems in application. That said, any linearization of a nonlinear process, and certainly one that projects onto a low-dimensional space, has limitations. Our approach is advocated only for persistent features within the ability of the network and vehicle system to pursue. Large deformations may occur on long time scales, while smaller and shorter ones need to be nearly linear.

The physics of ocean processes introduce spatial and dynamic structure into the field, which may be reflected in the motions of the frontal points. Such physical aspects include the length scales of turbulence vs. smoothness in diffusion, and geostrophic or tidal forcing at large scales. Establishing coupling between the frontal points is powerful since an integrated control and estimation strategy would enable coupled vehicles to perform better than vehicles making decisions based only on local information. To emphasize this point, we observe that in general the Kalman filter systematically reduces error covariance trace as the coupling between subsystems is increased, with all other parameters held fixed. This trend holds up to the point where the system's frequency content is beyond the capability of the filter. Thus, we set:

$\mathbb{P}_6$*: An instance* $\boldsymbol{p}_i$ *exhibits coupling.*

To establish coupling, we apply system identification techniques to the stochastic frontal points perturbations; some more details of our identification approach are given in **Section 4.4.2**. This method may be criticized first because it is equivalent to claiming that the coupled dynamic behavior of the ocean field can be expressed as a model of very low order; second because we claim that these dynamics are nearly linear; and third because stochastic identification (i.e., the system is driven by unknown forcing) is difficult enough even for illustrative problems in the literature. At the same time, linear analysis has in the past been extremely useful in elucidating fundamental behavior of ocean systems, e.g., [122,233], and indeed a full linear tangent

dynamic behavior has been developed in [164], and implemented in the widely-used *ROMS* modeling system. Along the lines of our current objectives, linear estimation of ocean systems using sensor networks has been considered before by [258].

4.2.7 Modeling Framework and Other Assumptions

Ensemble description of uncertainty: The quality of predictions is of course a perennial concern in modeling any stochastic, nonlinear process [177, 204]. An ensemble of model runs with variable forcing and initial conditions is a popular means for describing uncertainty, and this will be our language in the rest of the chapter. More generally, however, any description of uncertainty could be accommodated insofar as it allows projection into the vehicle space. Uncertainty in prediction arises primarily from initial condition errors, modeling errors, and unknown disturbances; we shall assume that all three elements are unbiased in each epoch.

Limited time scales and renewal: No climate nor ocean model enjoys sustained accuracy as the prediction horizon lengthens. For employing an ocean model over long time scales, we assume a data assimilation schedule as found in numerical weather prediction; predictions are made for a given forward horizon, on which we immediately carry out our entire design process, and implement it. When a new prediction becomes available, the process can be run again. An alternative information paradigm would be to employ repeated sections from longer simulations of cyclic processes, e.g., tidal flows in a coastal area.

Timing: The integrated observation system has several time scales. Following **Figure 4-1**, we define an epoch as the period between T_s and T_e over which a given model prediction (and the associated ensemble data) is valid, and the pursuit task is undertaken. For our examples in stochastic identification and controller performance, we study a single epoch.

The model prediction for each epoch consists of T unit time steps, and thus, $T_e - T_s = T$. The integrated observation and control system, *i.e.* all sensing, communications, and actions in the feedback loop, operates at its own time step, δt. In the current study, we use $\delta t = 1$, so that the networked control system matches the model

time step. This is not far from reality, since ocean model assimilation systems often run at intervals of several hours, while the command and control cycle of a vehicle system – at least over large scales and if multi-hop acoustics are involved – could be many minutes. The coincidence of time steps is a convenient choice for this chapter, but not required.

Centralized control architecture: For the multi-vehicle front tracking problem, we assume a centralized approach for the design stage, i.e., the upper half of **Figure 4-1**, with measurement packets sent to a fusion center, and control commands sent back out to the vehicles. Many multi-vehicle systems deployed at sea today have a similarly centralized architecture [195,215]. At the same time, there is no reason that the multi-vehicle control system (in the lower right of **Figure 4-1**) has to have a centralized architecture.

Breakdown: The integrated feedback system may fail for many reasons, some of which we allude to above, and illustrate below. In general, however, a failure does not compromise the basic abilities of individual vehicles, sensors, or communication system – a great many problems will reside in the modeling and the FP's. This suggests that on a failure of the integrated system, we can still operate assets in the field, and in many cases recover the survey and group capabilities that are available today. Recovery after breakdown is related to the problem of initial detection and convergence to the feature.

Navigation and jump aspects: Although we can accommodate certain operational considerations in our unified statement, for the most part we will consider navigation to be a separate problem, except as described through a standard LTI model with disturbances and sensor noise. Additionally, while the stochastic identification procedure gives an LTI model over the prediction horizon, in practice the system matrices change at every prediction update. This implies a jump linear system, which requires special treatment in control design. Operationally, there are also issues of vehicles entering and leaving the fleet (for example to charge batteries), which require a jump system approach.

4.3 Details of the Projector

4.3.1 Fully Constrained Perturbations

We first describe instance frontal points constrained to the reference gradient direction, passing through the reference frontal point. This model is suitable when perturbations are primarily in the gradient direction, as in when following moderate undulations perpendicular to a long front. Tracking the lateral motion of a cable or chain is a useful analogy for coupled behavior. For M prediction instances (realizations) and N vehicle sites, the projection problem on $r \in \mathbb{R}^2$, $t \in \{T_s, \cdots, T_e\}$, is:

given	$\phi_i(r,t)$	(scalar field instance) $i \in \{1, \cdots, M\}$
find	$\bar{\phi}(r,t)$	(reference field)
	$\Phi_j(t)$	(target level for ϕ at each vehicle) $j \in \{1, \cdots, N\}$
	$\bar{p}_j(t)$	(reference FP for each vehicle) $j \in \{1, \cdots, N\}$
	$\tilde{p}_{i,j}(t)$	(scalar FP perturbations) $i \in \{1, \cdots, M\}$, $j \in \{1, \cdots, N\}$

such that (dropping the time argument and the ranges for i and j)

i) $\bar{g}_n(r) := \nabla\bar{\phi}(r)/||\nabla\bar{\phi}(r)||_2$ (definition of normalized reference gradient vector)

ii) $p_{i,j} := \bar{p}_j + \bar{g}_n(\bar{p}_j)\tilde{p}_{i,j}$ (definition of constrained frontal point instance)

iii) $\bar{\phi}(\bar{p}_j) = \Phi_j$ (reference frontal point identity)

iv) $\phi_i(p_{i,j}) = \Phi_j$ (frontal point identity per instance)

v) $p_{i,j} \in \mathbb{P}$ (see **Table 4.1**)

Selection of the baseline field $\bar{\phi}$ is a matter of user choice, and could be as simple as the mean field. The target level Φ_j for a given vehicle j is possibly time varying, and defines both the reference FP as well as the instance FP, (*iii*) and *iv*)). Constraint *ii*) establishes that instance FP's lie on a line passing through the reference FP, and

in the reference gradient direction.

We show how this construction allows for an explicit estimate of the distance between the vehicle and the frontal point, from measurement of the field variable. Let $\tilde{\phi}_i := \phi_i - \bar{\phi}$. For an instance in the field (again dropping the time argument), invoking a Taylor series gives

$$\begin{aligned}
\phi_i(p_{i,j}) &= \bar{\phi}(p_{i,j}) + \tilde{\phi}_i(p_{i,j}) \\
&= \bar{\phi}(\bar{p}_j + \bar{g}_n(\bar{p}_j)\tilde{p}_{\,i,j}) + \tilde{\phi}_i(\bar{p}_j + \bar{g}_n(\bar{p}_j)\tilde{p}_{\,i,j}) \\
&= \bar{\phi}(\bar{p}_j) + \|\nabla\bar{\phi}(\bar{p}_j)\|_2\, \tilde{p}_{\,i,j} + \tilde{\phi}_i(\bar{p}_j) + \text{h.o.t.}
\end{aligned}$$

"h.o.t." indicates higher-order terms, which we drop, and the second term here results from the simple fact that $\nabla\bar{\phi}(\bar{p}_j)$ is parallel to $\bar{g}_n(\bar{p}_j)$. Now let $q_{\,i,j}$ be the location of vehicle j; with $q_{\,i,j} := \bar{p}_j + \bar{g}_n(\bar{p}_j)\tilde{q}_{i,j}$, the same expansion above gives

$$\phi_i(q_{\,i,j}) \approx \bar{\phi}(\bar{p}_j) + \|\nabla\bar{\phi}(\bar{p}_j)\|_2\, \tilde{q}_{i,j} + \tilde{\phi}_i(\bar{p}_j)$$

Finally, define the noisy measurement, expressed as a deviation from the reference value:

$$\begin{aligned}
z_j^{\phi} &= \left[\phi_i(q_{o,j}) + \nu_j^{\phi}\right] - \Phi_j \\
&\approx \|\nabla\bar{\phi}(\bar{p}_j)\|_2(\tilde{q}_{o,j} - \tilde{p}_{o,j}) + \nu_j^{\phi},
\end{aligned} \tag{4.2}$$

where ν_j^{ϕ} is the measurement noise of the scalar field sensor on vehicle j. The distance between the true frontal point and the vehicle is thus captured in the field measurement.

4.3.2 Unconstrained Translation

Now we look briefly at the more general case, in which the instance FP's do not have to remain on the reference gradient line through the reference FP; they can translate arbitrarily. Such a projector could be used to follow a consistently-shaped structure

moving through space, for example by placing FP's along fixed directions from its centroid. The problem statement of the previous section is modified only slightly. Scalar $\tilde{p}_{i,j}$ is changed to a two-vector, and constraints i) and ii) are replaced with $p_{i,j} = \bar{p}_j + \tilde{p}_{i,j}$. We then simply write the vector form of the earlier measurement equation:

$$z_j^{\phi} \approx [\nabla\bar{\phi}(\bar{p}_j)]^T(q_{o,j} - p_{o,j}) + \nu_j^{\phi}. \tag{4.3}$$

It is important to note that this approach, by **Equation 4.1**, cannot follow $\boldsymbol{p}_o$ in the absence of coupling. The target pursuit analogue is illustrative. A vehicle monitoring target range in the N-S direction plus a vehicle monitoring range in the E-W direction are clearly sufficient to pursue the target if they communicate. Implicit at each tracker is the fact that the range gradient points exactly away from the target, and this is parallel to $q-p$. The explicit model coupling is simply that the two trackers are seeing the same target, or $p_1 = p_2$.

4.4 System Integration Steps

As we discuss integration of the system described so far, one should bear in mind that beginning with the system identification step, all measurements and state variables now relate to *perturbations.*

4.4.1 Implementing Projection

As noted earlier, projection can be made generic or problem-specific; the main objective is to provide oceanographers with a good balance of automation and interface. We will take here the key ideas from **Section 2**, and the notation of **Section 3**, to describe the projection used in our example cases.

First, we specify the reference field $\bar{\phi}$ as the ensemble mean. The set of interest on $\bar{\phi}$ is taken as a front defined by the scalar field value Φ, constant through the epoch; we set $\Phi_j = \Phi$ for all the FP's j. Second, a confined area (the $\mathcal{C}_1$ box from **Section 4.2.2**)

is chosen at each time step, containing desirable parts of the Φ-level set. We look through the ensemble data by hand (as an oceanographer might), and select for realistic vehicle motions ($\mathbb{P}_2$), strong gradients ($\mathbb{P}_3$), stable reference features ($\mathbb{P}_4$), and feature clarity ($\mathbb{P}_5$). $\mathbb{P}_2$ focuses on satisfying vehicle speed constraints, for which we limit the distance any reference FP can move between time steps: $|\bar{p}_j(t+1) - \bar{p}_j(t)| \leq K_u u_{max}, \forall j \in \{1, \ldots, N\}, t \in \{T_s, \ldots, T_e - 1\}$, where u_{max} is the vehicle maximum speed, and $0 < K_u < 1$ is a user parameter. Actual vehicles have to follow dynamic perturbations, not only the reference, so K_u is typically well below one. This approach for $\mathbb{P}_1$ can be extended to more complex and practical constraints specific to a given deployment, such as expected distance traveled, mission time, energy, and so on.

To pick the specific $\bar{\boldsymbol{p}}$ at each time step, we developed an interface in which the user draws rays across the reference front within $\mathcal{C}_1$, and the intersections then define $\bar{\boldsymbol{p}}$. The reference gradient directions $\nabla\bar{\phi}(\bar{p}_j)$ are computed in the same process, defining a set of N line constraints $\mathcal{C}_{2,j}$. We next project instance variations of the front onto scalar FP perturbations $\tilde{p}_{i,j}$, using **Algorithm 1**. The next two paragraphs detail variations of this algorithm that account for grid interpolation and complicated local contours.

Algorithm 1 FP Projector for constrained linearization

Require: $\phi_i(t)$ for $i \in \{1, \cdots, M\}$, and
$[\bar{p}_j(t), \mathcal{C}_1(t), \mathcal{C}_{2,j}(t)]$ for $j \in \{1, \cdots, N\}$, all for $t \in \{1, \cdots, T\}$
for all epoch time steps $t = 1, \ldots, T$ **do**
  for all instances $i \in \{1, \ldots, M\}$ **do**
    for all FP's $j \in \{1, \ldots, N\}$ **do**
      $\mathbb{S}_0 \leftarrow \Phi$-contour of field ϕ_i in $\mathcal{C}_1(t)$
      $p_{i,j}(t) \leftarrow$ unique intersection of $\mathbb{S}_0$ and $\mathcal{C}_{2,j}(t)$
      $\tilde{p}_{i,j}(t) \leftarrow p_{i,j}(t) - \bar{p}_j(t)$
    end for
  end for
end for

Computed contours consist of vertices and straight segments, so ensuring smooth interpolation to find the intersection of a local contour $\mathbb{S}_0$ and a line $\mathcal{C}_2$ takes some care. We choose a number of points on $\mathbb{S}_0$ in the immediate neighborhood of $\bar{p}_j$, compute the signed distance of each point to $\mathcal{C}_2$, and record the closest point on either side of

$\mathcal{C}_2$. $p_{i,j}$ is then computed as the intersection of $\mathcal{C}_2$ with the line connecting these two closest points. This neighborhood around $\bar{p}_j$ is unique to each FP and thus tighter than $\mathcal{C}_1$ (which covers the entire feature of interest)—this helps to ensure uniqueness of frontal points.

The instance front is not always well-behaved. Eddies and similar fluctuations occur, and these create complicated $\mathbb{S}_0$, which can induce multiple intersections with $\mathcal{C}_2$. We attempt to smooth these out by point selection based on the median filter, well-known in signal and image processing for removing outliers [90]. First, find a set of n tight contours for ϕ-levels very close to Φ; we refer to this as a contour family $\mathbb{S}(\supset \mathbb{S}_0)$. For a given frontal point, compute the intersection of $\mathcal{C}_{2,j}$ with each member of $\mathbb{S}$; there may be several intersections for a given member, but we only keep the one closest to $\bar{p}_j$. Compute the signed distance to $\bar{p}_j$ for each of these n closest intersections, and then take the median across $\mathbb{S}$ as $\tilde{p}_{i,j}$. In the extreme case where $\mathbb{S} \cap \mathcal{C}_{2,j} = \emptyset$ (that is, no crossing contours are available in the area of interest), we expand $\mathcal{C}_1$, and with it $\mathbb{S}$, enough to find at least one intersection, so the FP can be placed. For our computations, this overall approach has proven very effective for limiting sudden jumps in $\tilde{p}_{o,j}$.

In some instances the front breaks down to a point that the real-time system $\mathbb{E}_o$ simply cannot follow, and so creates major outliers in the FP's. This is a failure to satisfy $\mathbb{P}_5$. Linear system identification does not handle outliers well, and if needed we can concede to *exclude outlier realizations from system identification*. If the fraction of excluded trials in the ensemble is high, however, we clearly cannot expect the real-time system to do well. Practically speaking, any operational vehicle system should be able to switch to a "recovery" mode if such conditions are encountered in a physical experiment; we expect that even if the oceanographic pursuit mission fails, the ensemble can still aid failure-detection algorithms.

Once the population of FP perturbations, $\tilde{p}_{i,j}(t)$ has been created, it should to be reviewed to ensure that all (or an acceptable fraction) are in $\mathbb{P}$. $\mathbb{P}_2$ is notable because it relates to vehicle control, speed and maneuvering capabilities. For example, after running the projector, we check the vehicle speeds: $|p_{i,j}(t+1) - p_{i,j}(t)| \leq u_{max}$.

Corrections in the process are likely to be made at the level of the reference FP's, since **Algorithm 1** even with its variations has few tuning parameters.

4.4.2 Stochastic Identification of Instance Frontal Points

Linear system identification [147] – the determination of an LTI dynamical system model to explain measurements – is highly effective when the input signals are known. For single-input, single-output systems, strong results are regularly produced by robust time- and frequency-domain approaches; for multi-input, multi-output systems, subspace identification algorithms [237] are widely used. The stochastic analogue, where we wish to explain the data as the output of a coupled system driven by noise inputs, is much less developed. In fact, aside from the two key references above, there seem to be few technical results on the topic in the last fifteen years. Intuitively, one can appreciate that the difficulty of stochastic identification reflects a tenuous optimization problem – the raw constraints are noisy output traces, the belief that the driving signals are Gaussian, and a system order specification. System order selection is aided by computing singular values of the Hankel matrix on the input data; this is very similar to what is done for balanced realization and model reduction [165]. We describe our specific choices for model order selection in **Section 4.5**.

Given the order, stochastic subspace methods involve first estimating a sequence of *states* from the data; this is in strong contrast to the input-output viewpoint of classical system identification. The states are interpreted as the output of Kalman filter predictors, and they can be obtained through QR factorization and singular value decomposition (SVD). Once these states are identified, a linear least-squares regression yields the unknown system matrices. The specific numerical algorithm we use is Matlab's `n4sid` [237], providing a model in innovations form. In our case, the outputs are the linearized perturbations $\tilde{\boldsymbol{p}}$ (with t denoting the time step):

$$\begin{aligned} \boldsymbol{x}_p(t+1) &= A_p \boldsymbol{x}_p(t) + K\boldsymbol{e}(t) \\ \tilde{\boldsymbol{p}}(t) &= C_p \boldsymbol{x}_p(t) + \boldsymbol{e}(t), \end{aligned}$$

and A_p, K, C_p, and $\mathrm{Cov}(\boldsymbol{e}) \triangleq R_e$, are generated by the algorithm. We reiterate that $\tilde{\boldsymbol{p}}$ is available only for the model ensemble, not for the physical instance.

Several specific challenges that practitioners of stochastic algorithms face include high sensitivity to pre-filtering of data, and, perhaps more seriously, the requirement of long signals. Indeed guarantees of optimality and asymptotic unbiasedness only exist for an infinite number of samples, and statistical analysis with finite sample length remains an open problem [237]. In practice `n4sid` can generate useful results with short sequences, and this would be a necessity for oceanographic pursuit, where typically few, if any, repeated events are observed. One could conjecture that from a system identification point of view, longer data traces are desirable even if the latter portions of them are useless for forecasting; this question is beyond our scope.

`n4sid` includes a few user parameters that can be tweaked to improve performance: SVD weighting, forward prediction horizon, and number of past inputs used for prediction. In both [147, 237], however, it is made clear that optimal choices for most of these settings are still open subjects of research.

4.4.3 Connecting with the Vehicle System

We take in our framework a linear time-invariant model of vehicle system behavior; see [132] for a recent review of underwater vehicle navigation systems that support control. Our formulation is developed for a *group* of vehicles, which may or may not share physical disturbances and navigation aspects. This aggregate vehicle system is described by the state-space dynamics matrix A_q, gain B_q, and output C_q; the vehicles' process noise vector $\boldsymbol{w}_q$ has covariance Q_q, and measurement noise $\boldsymbol{\nu}_q$ has covariance R_q. A_q and B_q are block diagonal, since the vehicles have no coupling except through the control and possibly the disturbances. The integrated open-loop system is:

$$\left\{\begin{array}{c}\boldsymbol{x}_p(t+1)\\ \boldsymbol{x}_q(t+1)\end{array}\right\} = \left[\begin{array}{c|c} A_p & 0 \\ \hline 0 & A_q \end{array}\right] \left\{\begin{array}{c}\boldsymbol{x}_p(t)\\ \boldsymbol{x}_q(t)\end{array}\right\} + \left[\begin{array}{c} 0 \\ \hline B_q \end{array}\right] \boldsymbol{u}(k) + \left[\begin{array}{c|c} K & 0 \\ \hline 0 & I \end{array}\right] \left\{\begin{array}{c}\boldsymbol{e}(t)\\ \boldsymbol{w}_q(g)\end{array}\right\}. \quad (4.4)$$

As written, dynamics of the frontal point perturbations and vehicle perturbations are decoupled, giving the macro block-diagonal structure shown. Clearly $\boldsymbol{x}_p$ is affected by neither $\boldsymbol{x}_q$ nor $\boldsymbol{u}$, but is $\boldsymbol{x}_q$ affected by $\boldsymbol{x}_p$? Although in principle one could establish such coupling, the fact that A_p is developed from stochastic identification, whereas A_q is not, implies that simulation of vehicles and their control system would have to be added into $\mathbb{D}$. This turn toward a self-referential projector is left for future work. Off-diagonal blocks in the process noise gain matrix can be left at zero by the same rationale.

The output equations are where the oceanographic and the vehicle systems most strongly interact:

$$\boldsymbol{z}(t) = \begin{Bmatrix} \boldsymbol{z}^{\phi}(t) \\ \tilde{\boldsymbol{q}}(t) \end{Bmatrix} = \left[\begin{array}{c|c} -GC_p & GC_q \\ \hline 0 & C_q \end{array}\right] \begin{Bmatrix} \boldsymbol{x}_p(t) \\ \boldsymbol{x}_q(t) \end{Bmatrix} + \left[\begin{array}{c|c|c} -G & I & 0 \\ \hline 0 & 0 & I \end{array}\right] \begin{Bmatrix} \boldsymbol{e}(t) \\ \boldsymbol{\nu}^{\phi}(t) \\ \boldsymbol{\nu}_q(t) \end{Bmatrix}, \quad (4.5)$$

with $G = \mathrm{diag}(\|\nabla\bar{\phi}(\bar{p}_1)\|_2, \ldots, \|\nabla\bar{\phi}(\bar{p}_N)\|_2)$, and it is assumed that G varies slowly enough not to interfere with the feedback controller ($\mathbb{P}_4$). Note that the disturbance $\boldsymbol{e}(t)$ enters the output equation, in accordance with the innovations form. Its inclusion highlights a subtlety in **Equation 4.2**; whereas $\tilde{q}$ represents a physical vehicle perturbation via a strictly proper system, $\tilde{p}$ does not. $\tilde{p}$ is the output of our projection algorithm with no low-pass constraints at all, and as defined in the stochastic identification it has "jitter."

4.4.4 Estimation of the Integrated System

The aggregate process noise vector (dropping the time argument) is $\boldsymbol{w} = [(K\boldsymbol{e})^T,\ (\boldsymbol{w}_q)^T]^T$ and the total measurement noise vector is $\boldsymbol{\nu} = [(-G\boldsymbol{e} + \boldsymbol{\nu}^{\phi})^T,\ (\boldsymbol{\nu}_q)^T]^T$. Process noise for the FP's may be correlated with the vehicles' process noise, for example due to currents; this is captured by the matrix Q_{eq}. We assume, however, that beyond $\boldsymbol{e}$, no process noise is correlated with any sensor noise. For use in the generalized Kalman

filter [220], expanding out the expectations gives:

$$\mathbb{E}\left[[\boldsymbol{w}^T,\boldsymbol{\nu}^T][\boldsymbol{w}^T,\boldsymbol{\nu}^T]^T\right] = \left[\begin{array}{cc|cc} KR_eK^T & Q_{eq} & -KR_eG^T & 0 \\ * & Q_q & 0 & 0 \\ \hline * & * & (GR_eG^T+R_\phi) & 0 \\ * & * & * & R_q \end{array}\right]. \tag{4.6}$$

4.5 Examples of Projection and Identification

We now present three projection and stochastic identification examples, for the fully constrained linearization of **Section 4.3.1**. The first case is an LTI chained mass system, for validation. Next, we study a double gyre model, simulated using a finite-volume Navier-Stokes solver [210,235]. The double gyre is a canonical fluid mechanics problem, highly nonlinear and unstable; while these factors would seem to position it poorly for system identification, there is also a dominant wave-like behavior that greatly contributes to coupling, and to a locally linear behavior. More broadly the double gyre is a generic and dimensionless scenario with few physical parameters, and thus useful for benchmarking. The third data set is substantially more difficult and realistic, focusing on three-dimensional flows north and east of Taiwan. This particular set was part of a larger study [95] emphasizing prediction and uncertainty for an ocean setting with complex multiscale dynamics, from internal tides and waves to large-scale currents. One of the main features noted was sporadic intrusions by the Kuroshio Current into the so-called "Cold Dome." These elements make it a challenging case for oceanographic pursuit.

4.5.1 Identification of an LTI Chained Mass

The chained mass system has eight masses arranged nominally along a line and undergoing lateral perturbations. Each mass is tied to ground lightly by a spring k_g and damper b_g, connected to adjacent neighbors with springs k_n and dampers b_n, and forced by zero-mean white noise of variance w_j. The parameters are tuned to give magnitudes and frequencies similar to those in the double gyre FP perturbations; see

Table 4.2. The initial condition for each point is $p_j(0) = 0.1 \cdot \sin\left(j\frac{2\pi}{N-1}\right) + 0.5q,\ \forall j \in \{1, \ldots, N\}$, where $q = \mathcal{N}(0, 1)$.

Table 4.2: Parameter values used for chained mass system.

k_n	k_g	b_n	b_g	$w_{1,\ldots,7}$	w_8
0.02	0.048	0.3	0.01	0.13	0.26

The identification procedure here is the same as used for the ocean model data. We chose a model order twice the number of sites; this is of course the order of the generating system, but also consistent with our choices later. With the `n4sid` algorithm, we used a forward prediction horizon of four, and four past inputs. **Figure 4-3** compares time series for two "data" instances (left) to two time series constructed by stochastic simulation of the identified system and its noise statistics (middle). The "data" perturbations show a dominant resonant mode and tight coupling in phase; this strong coupling is supported by the correlation coefficients shown in **Figure 4-4**. Time series and correlations from the identified model show virtually the same properties. The Hankel singular values of the input data (**Figure 4-5** (left)), however, show that the stochastic ID does not exactly recover the original system. We also ran a local second-order identification for each mass. This is shown in **Figure 4-3** (right), where all relative phase information has been lost. This model is what we use below in a "loners" control scheme made up of non-communicating vehicles.

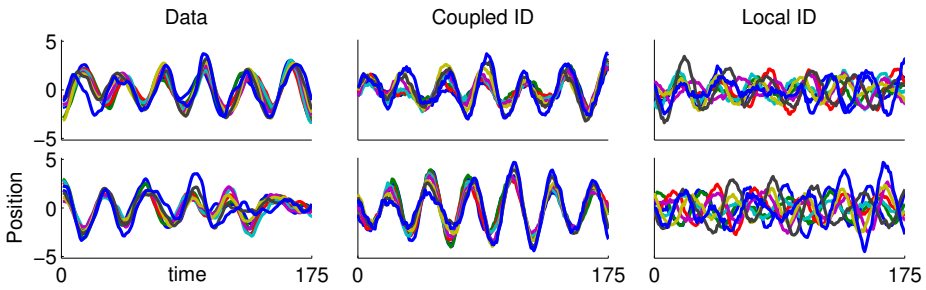


Figure 4-3: Time series of selected chained mass perturbations. The eight sites are shown in different colors on each plot. Left: two selected instances of original simulated data. Middle: simulations using coupled system identification. Right: simulations using local system identification.

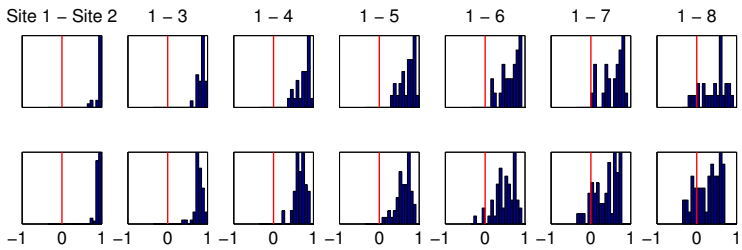


Figure 4-4: Top: histograms of FP correlation coefficients from original chained mass simulated data (25 instances). Bottom: from identified model simulations (50 instances). Only correlations between Site 1 and the others are shown.

4.5.2 Identification of a Double Gyre Front

The feature of interest in the double gyre model is a vorticity contour; although vorticity is not a typical scalar measurement taken at sea, turbulence scales and shear stress are. We study a 176-timestep period that takes place when the gyre system is just starting to become unstable, soon after which all structure is lost. Eight reference FP trajectories were designed, maintaining equal spacing along the reference front. We then ran the projector on 25 randomly-selected ensemble members, and performed stochastic identification. The perturbations were lightly low-pass filtered before being passed into `n4sid`, so as to reduce grid effects. The Hankel singular values shown in the middle plot of **Figure 4-5** have only one clear break point, at a model order of *two*. This choice would allow only for one oscillator, with differences between the behavior at each site merely representing phase shifts. The actual response is somewhat more complicated, and we chose a model order of sixteen. A higher number than this would risk overfitting, while sixteen can be justified at least from a comparison point of view: the uncoupled dynamics is naturally described with eight independent pairs of states. We used a forward prediction horizon of four, and four past inputs in `n4sid`.

As for the chained mass, in **Figure 4-6** we compare perturbations for two instances (left) to two time series constructed by stochastic simulation of the identified system and its noise statistics (middle). The instance fluctuations are oscillatory and coordinated, with a traveling wave characteristic; amplitude, phase relations, and frequency content are all well-captured in the identified model. Correlation coefficients shown in **Figure 4-7** confirm the strong coupling. The local system identification

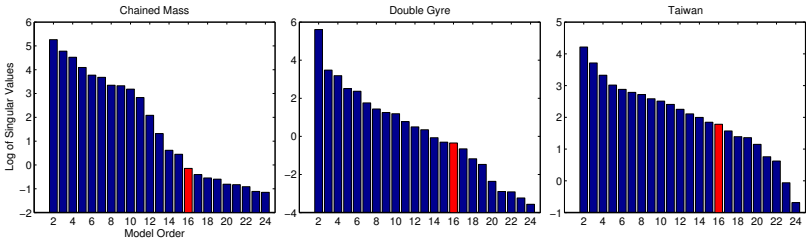


Figure 4-5: Hankel singular values for identification data. Red bars indicate the model order chosen for each case study.

shown in **Figure 4-6** (right) does not describe coupling and relative phase between points.

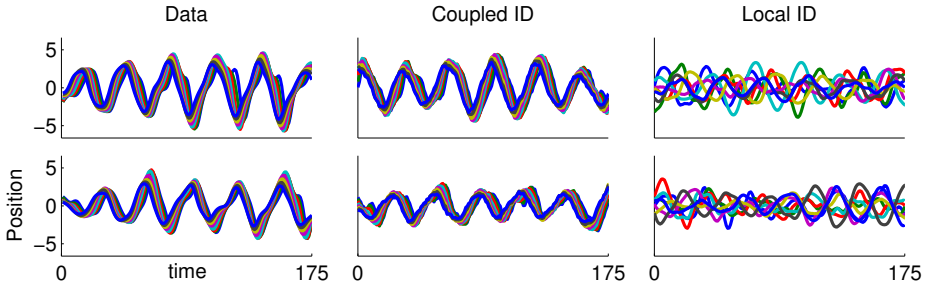


Figure 4-6: Time series data, same format as **Figure 4-3**.

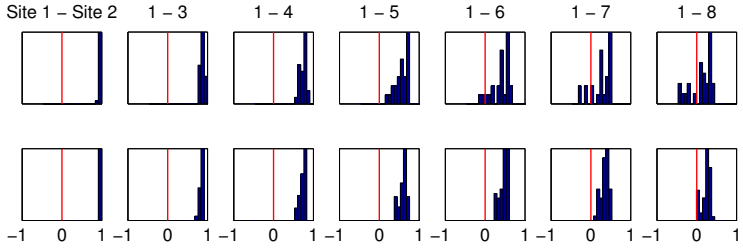


Figure 4-7: Correlation coefficients, same format as **Figure 4-4**.

4.5.3 Identification of a Front off Taiwan

The Taiwan ensemble has fifty instances, the time step is three hours, the run is 68 steps long (about eight days), and the model grid is 4.5 km. The feature of interest here is a persistent temperature front at 50 m depth. Compared with the double gyre case above, this front exhibits much smaller perturbations relative to motion of the reference, and far less structure. We followed the same general procedure in the

projector, picking a temperature level and a physical area where gradients are strong and the front stays well-formed. Snapshots over time of a single realization are shown in **Figure 4-8**.

To illustrate the key aspects of this scenario pertinent to identification and oceanographic pursuit, **Figure 4-9** shows slices of the temperature field along the reference gradient cut, and centered at each reference FP. These slices at time step 30 are from twenty-five instances used for projection and identification. Most of them have a clean albeit nonlinear shape, confirming definition of the front and strong gradients. Some sharp corners are visible, caused by interpolation on the model grid.

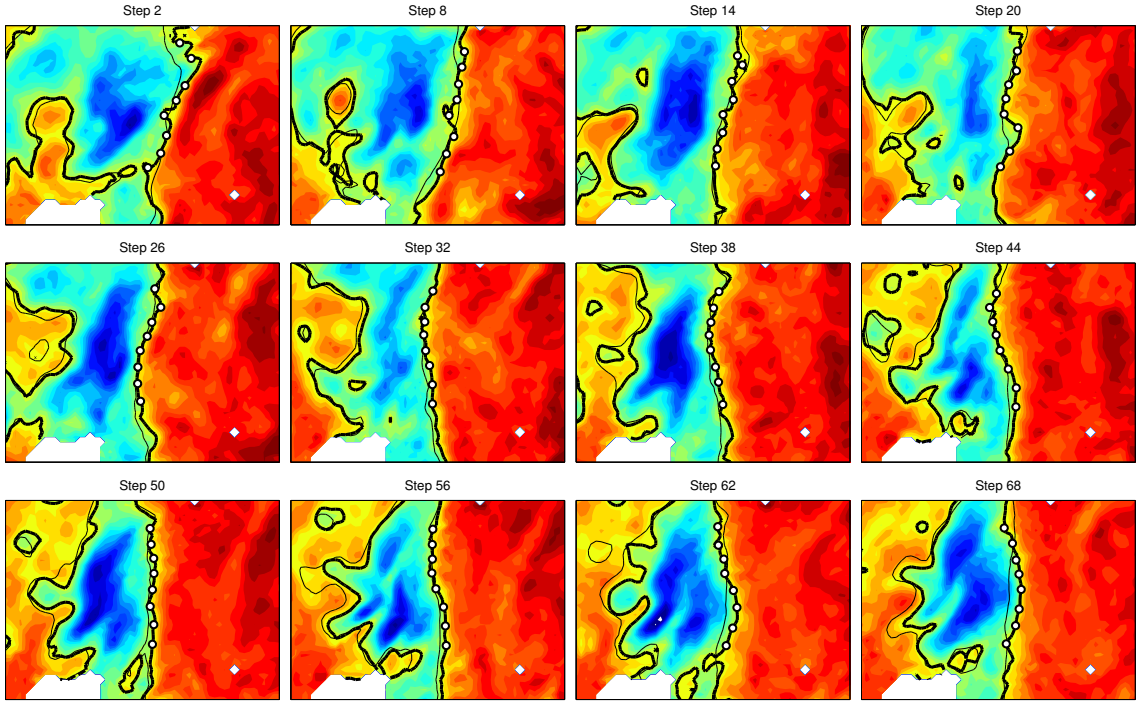


Figure 4-8: Snapshots over time of a single instance of the Taiwan dataset; the northern edge of Taiwan is visible at the bottom of each frame. This instance was used for projection and system identification. The reference contour is a thin black line, and the true contour is a thick black line. True FP's are white with black outlines. The domain of each box is 248 × 180 km. The time between snapshots is 18 hours, which makes out-of-phase semi-diurnal (and diurnal) internal tidal effects visible.

Hankel singular values on the right side of **Figure 4-5** show no break point for any order; this flatter shape indicates the difficulty and lack of structure in the dataset.

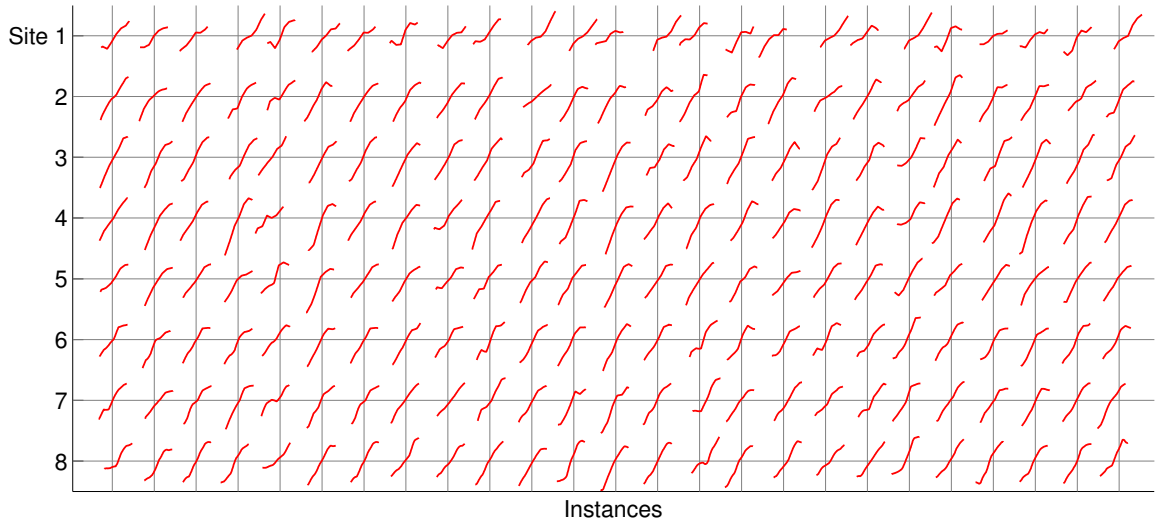


Figure 4-9: Slices of temperature field along constraint $\mathcal{C}_2$ for each FP at time step 30. Twenty-five ensemble instances used for projection and identification are shown. The horizontal axis represents distance along $\mathcal{C}_2$ and the vertical axis represents temperature. Slices are positioned horizontally relative to the reference front, which is the origin of each subplot.

We chose a system order of sixteen, for consistency with twice the number of FP's. With the `n4sid` algorithm, we used a forward prediction horizon of six, and three past inputs. Time series are plotted in **Figure 4-10**, and these confirm reduced coupling. Yet there are still meaningful correlations between neighboring points, as seen in **Figure 4-11**. These correlations are replicated in the identified system, and we will show in the next section that even this modest level brings a benefit to communication-constrained control.

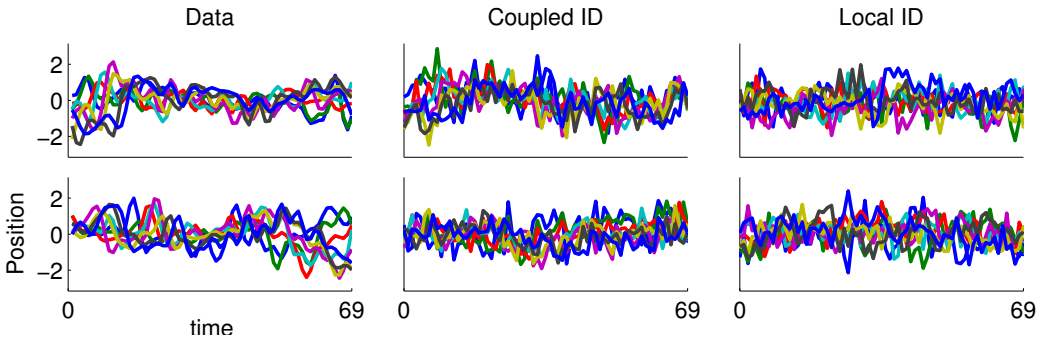


Figure 4-10: Time series data, same format as **Figure 4-3**.

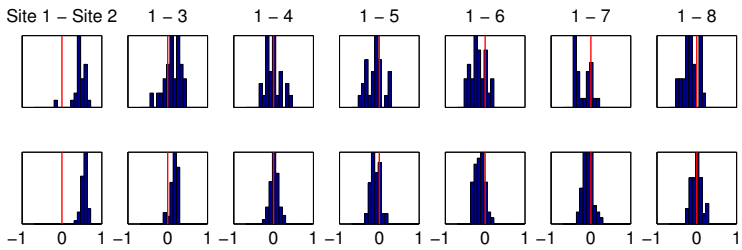


Figure 4-11: Correlation coefficients, same format as **Figure 4-4**. Adjacent points show significant correlation and far-away points are largely uncorrelated.

4.6 Closed-loop Control Examples

For each of the three scenarios above, our controller design uses an LTI system model found by identification on a subset of ensemble instances; the controller design is evaluated on separate subsets. We emphasize that these cases are considered only proof of concept; the control settings and operational parameters used do not necessarily reflect real conditions.

4.6.1 Explanation of Controllers

As stated in the Introduction, one of our main objectives is to develop controllers suited for communication constraints; one of the most difficult is packet loss. We define four vehicle communication models:

- **NC**: no communication between vehicles for the purpose of oceanographic pursuit.
- **PC**: perfect communication (lossless and instantaneous).
- **IL**: independent losses. In a cycle, vehicles report measurements instantaneously to a fusion center, which then sends out a command set instantaneously. There are thus N inbound sensor packets and N outbound sensor packets. Losses in the sensor packets are binary, and described as a set of independent Bernoulli processes, having success probabilities $[\beta_1, \cdots, \beta_N]$. Losses in the command packets are similarly described by the Bernoulli parameters $[\alpha_1, \cdots, \alpha_N]$.

- **TL**: total losses. In a cycle, vehicles report measurements instantaneously to a fusion center, which then sends out a command set instantaneously. The sensor packets all succeed with probability β, or they all fail. The command packets all succeed with probability α, or they all fail. These total failures on the command and the sensor sides are independent.

This setup ignores the role of communication delays, interference, and scheduling—aspects that, like packet loss, can be handled rigorously from an LTI framework. We compare five controllers, each subject to vehicle navigation noise and physical disturbances:

- "Non-reacting" uses local linear quadratic Gaussian controllers (LQG, comprising a Kalman filter (KF) for state estimation and a linear quadratic regulator (LQR) for full-state feedback) to place vehicles at the reference FP's. The vehicles do not alter their trajectories based on field measurements, and thus do not have to communicate: **NC**. The local model from stochastic identification is used for estimation with a KF at each vehicle. This approach gives a practical upper bound on real-time estimation error.

- "Loners" uses a set of N independent LQG controllers to servo to the estimated front. Each vehicle has only a local model derived from the stochastic identification: **NC**.

- "Naïve" applies a standard LQG controller design, given a fully coupled model and assuming **PC**. The simulation uses **IL**, while the KF takes the standard approach for handling missed measurements.

- "All-or-none" uses the dynamic programming procedure of Imer *et al.* [123], on a fully coupled model and where **TL** is assumed. In simulation, we impose **IL**. The control design uses the means of the α and β vectors that make up the simulation **IL** model. A regular missed-measurement KF applies, but the **TL** scalar α invokes an adjustment to the prior due to the uncertainty of the control action [94].

- "Lower Bound" applies a standard LQG controller design, given a fully coupled model and assuming **PC**. The simulation is **PC** also, and hence this controller is expected to be the best.

When control packets are not received, vehicles stay in place. The control objective is to minimize the squared positioning error over time, for all sites/vehicles: $\Sigma_{j=1}^{N}\Sigma_{t=1}^{T}\left(\tilde{q}_{o,j}(t)-\tilde{p}_{o,j}(t)\right)^2$. For LQR design, we weight positioning error ten times more heavily than control.

4.6.2 Results

Gain Margins

We return briefly to the gradient condition $\mathbb{P}_{3b}$: $\|\nabla\phi_i(p_i+\Delta)-\bar{g}\|_2 \leq \epsilon\kappa$ for each FP, where Δ is a positioning error, and $0<\epsilon\ll 1$. ϵ is related explicitly to gain margin, a standard means for quantifying robustness in the closed loop, while κ ultimately limits the bandwidth of the closed-loop system. The first ten instances of the Taiwan data (from **Figure 4-9**) are re-plotted in **Figure 4-12**, now showing more specifically the slices (red) overlaid on top of twice the reference gradient slope (blue), and centered at the instance FP. Differences between the reference gradient (not shown) and the instance slice can be interpreted as sensor gain variations from the point of view of an estimator. Although we use a time-varying KF, it is noteworthy that the steady-state KF and the LQR each have a guaranteed per-channel gain margin of zero to two [207] and thus the blue line and the horizontal define a [0,2]-sector that is highly desirable.[2] We see that all of these instantaneous slices fall within, or very close to within, the sector, at least near the origin.

Controller Performance

For our simulations, we assume that all of the vehicles have identical dynamics and sensors. The scalar field measurement noise covariance R_ϕ is the most sensitive pa-

[2]Despite its wide use in applications, the generic LQG does not have guaranteed stability margins [67].

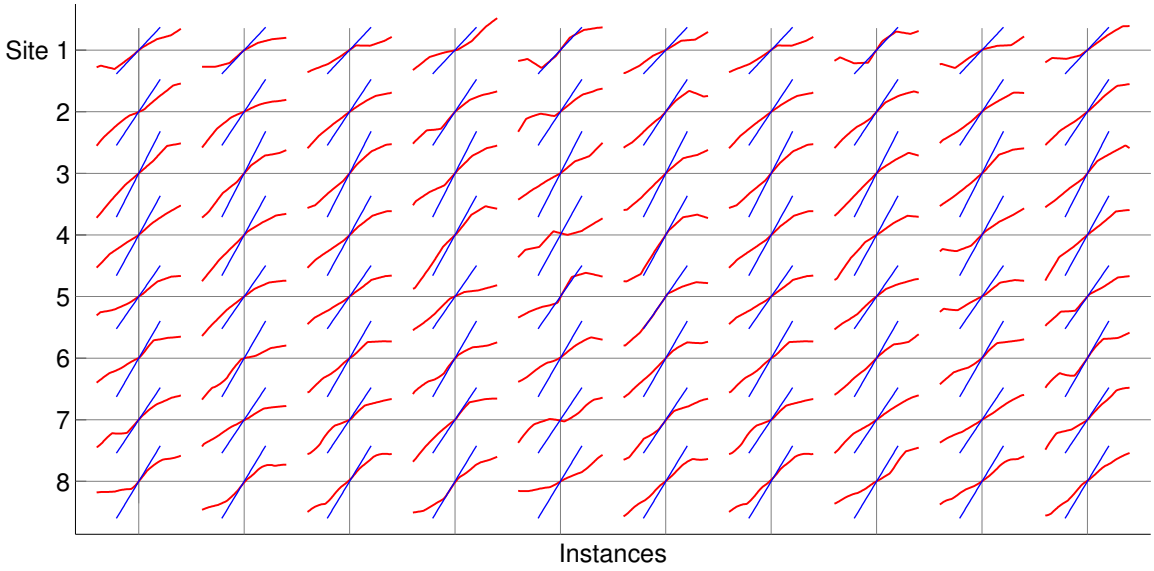


Figure 4-12: Ten instances of Taiwan temperature slices (red) along $\mathcal{C}_2$, overlaid on top of twice the reference gradient (blue). The horizontal axis represents distance along $\mathcal{C}_2$, centered at the instance FP, and the vertical axis represents temperature. The [0,2] sector defines a region of stability for the Kalman filter.

rameter, and our results compare estimation performance varying R_ϕ, with other parameters fixed. For all cases, we simulate control on instances separate from those used for projection and identification, and use $\tilde{p}_{o,j}$ generated by the omniscient projector for ground-truth. We use one hundred simulation instances for the chained mass, and twenty-five instances each for the double gyre and Taiwan datasets. Vehicle noise parameters are set as $R_q = 0.01$ and $Q_q = 0.01$. These are chosen to roughly describe a physical scenario, but in reality would depend on the specific vehicles and environmental conditions. Packet success probabilities are set as $\bar{\alpha} = [0.7, 0.8, 0.6, 0.9, 0.7, 0.8, 0.6, 0.9]$, $\bar{\beta} = [0.6, 0.9, 0.7, 0.6, 0.8, 0.9, 0.7, 0.8]$.[3] From examination of empirical transfer functions for positioning error in all three scenarios with a variety of noise settings, we verified that the control system bandwidth is sufficient to track the dominant motion of the feature.

We show results comparing performance with varying R_ϕ in **Figure 4-13**. R_ϕ in the plots is scaled with the mean gradient squared, so as to capture the importance

[3]In other work, we have investigated a mixed-loss controller that explicitly designs for **IL** [196]; however, mixed-loss control performs similarly to "All-or-none" with this set of loss probabilities.

of noise relative to the ambient signal.[4] We compute the RMS (over time) estimation error $(\hat{\tilde{p}} - \tilde{p})$, averaged across sites and across simulation instances, and this error is normalized by the same statistic on the perturbations used for system identification, in order to compare results across datasets. The leftmost plot of **Figure 4-13** shows the performance of Lower Bound. Major differences underscore the relative difficulties mentioned earlier, and in particular Taiwan has less structure in its perturbations and considerably more noise, caused both by grid effects as well as erratic projections in the noisy field. At the other extreme, Lower Bound for the chained mass, designed using the exact system model, performs only slightly better than Lower Bound based on the identification—demonstrating that the control system resulting from identification is successful. Lower Bound for the chained mass does far better than in either the double gyre or Taiwan cases, largely because the chained mass has an ideal projector and exact gradients, such that estimation performance is independent of positioning.

In the right plots of **Figure 4-13**, we plot differences between the estimation error of each control method and that of Lower Bound. We will describe a few key aspects: positioning, coupling, and packet-losses.

Non-reacting and Loners use the same local model for estimation; the difference is that Loners positions vehicles reactively to pursue the front. Non-reacting and Loners are identical in the chained mass case, but with the ocean model datasets, estimation becomes more difficult as positioning worsens in the nonlinear field (see **Figure 4-2**(right)). This fact yields a dramatic improvement in estimation for Loners over Non-reacting in the double gyre, where slices are smooth but slope deforms with distance from the front. In the Taiwan dataset, Loners also outperform Non-reacting, but to a lesser extent and mostly at low noise values. The Taiwan front has a more variable shape and shallower slope, along with more pronounced grid effects. For every scenario, the bandwidth of the estimator decreases with large R_ϕ—the entire system becomes sluggish, vehicles move less, and thus the differences between Loners

[4] A loose definition of SNR is gradient magnitude multiplied by the standard deviation of the navigation noise, divided by the standard deviation of the scalar field measurement noise. We have fixed R_q, so that these figures have essentially 1/SNR on the horizontal axis.

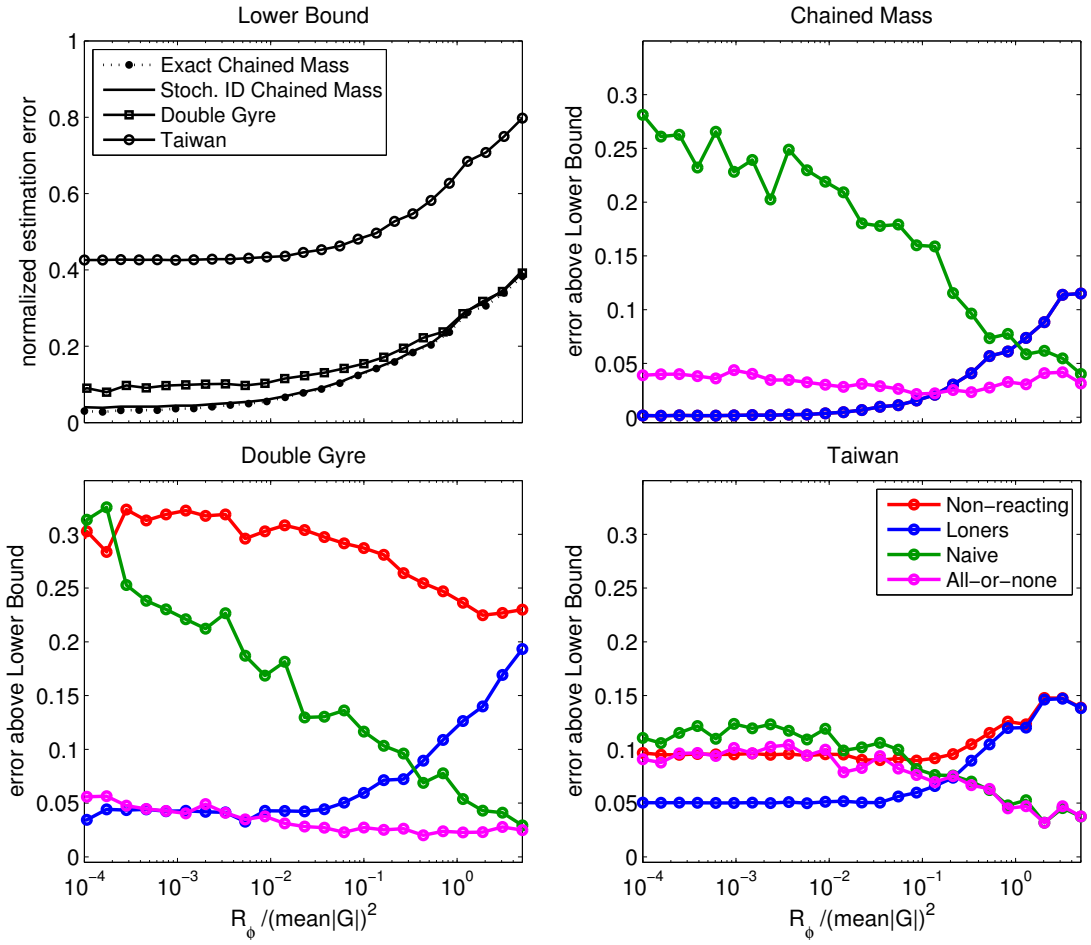


Figure 4-13: Summarized estimation performance. Estimation error is the RMS over time of $(\hat{\tilde{p}} - \tilde{p})$, averaged across points and realizations. **Upper left plot:** performance of Lower Bound in each dataset, as function of R_ϕ. For the chained mass, the "Exact" controller is designed using the true system and noise. All other controllers use the outcome of stochastic identification. **Upper right, and lower plots:** difference in estimation error relative to Lower Bound for all controllers in each dataset.

and Non-reacting disappear.

The role of coupling can be understood by comparing performance of Loners and Lower Bound. In the chained mass model, the improvement of Lower Bound over Loners increases significantly from small values as R_ϕ gets larger, reflecting clearly that coupling can offset the bad effects of sensor noise. In the two ocean model datasets, Lower Bound outperforms Loners even at very low R_ϕ, which may seem surprising but is due to the nonlinear shapes of the slices. The KF does the best it can with the given noise level, but the slice nonlinearities invariably create a "pseudo-noise," so the the filter is mis-tuned. Loners and Lower Bound both suffer, but Lower Bound again has the advantage of coupling.

A more nuanced effect of coupling relates to packet loss, and an interesting comparison can be seen between All-or-none and Loners. Communication losses hurt estimation because of missed measurements, but also hurt positioning directly through dropped control commands. At low levels of sensor noise, Loners outperform the controllers that are subject to packet loss because uncertain communication deteriorates the benefits of coupling. At larger R_ϕ, above a crossover, Loners cannot succeed with only local modeling, and All-or-none prevails, even with lossy communication.

Comparing now the two controllers subject to packet loss, Naïve and All-or-none use the same coupled model for estimation, and are subject to the same stochastic packet loss sequence in each simulation. All-or-none, however, explicitly takes loss into account in both controller design and in the uncertainty of the KF prior. This is a strong stabilizing effect, as Naïve shows much more sensitivity to sequences of packet losses; its errors can be very small or very large, giving rise to the jumps up and down in **Figure 4-13**. More broadly, Naïve performs poorly especially at low R_ϕ, the error converging to that of All-or-none as R_ϕ increases. This convergence simply reflects decreased estimator bandwidth.

The estimation performance over time from one example site and realization for each dataset is shown in **Figure 4-14**. The double gyre and chained mass cases show all methods clearly tracking the dominant oscillations, although Loners and Naïve often exhibit more over- or undershoot than the coupled methods. In the double gyre,

estimation errors of Non-reacting are clear, especially near the peaks of perturbation magnitude. The Taiwan case is drastically different, again showing the difficulty of this dataset. None of the controllers is able to track the high-frequency variations, but Loners and Non-reacting are much more sluggish than the coupled methods. These time series also illustrate system startup (in the first ten or so samples), an important factor to consider in real-world operations. We have had no difficulties in initializing to the reference FP's in any of our test cases.

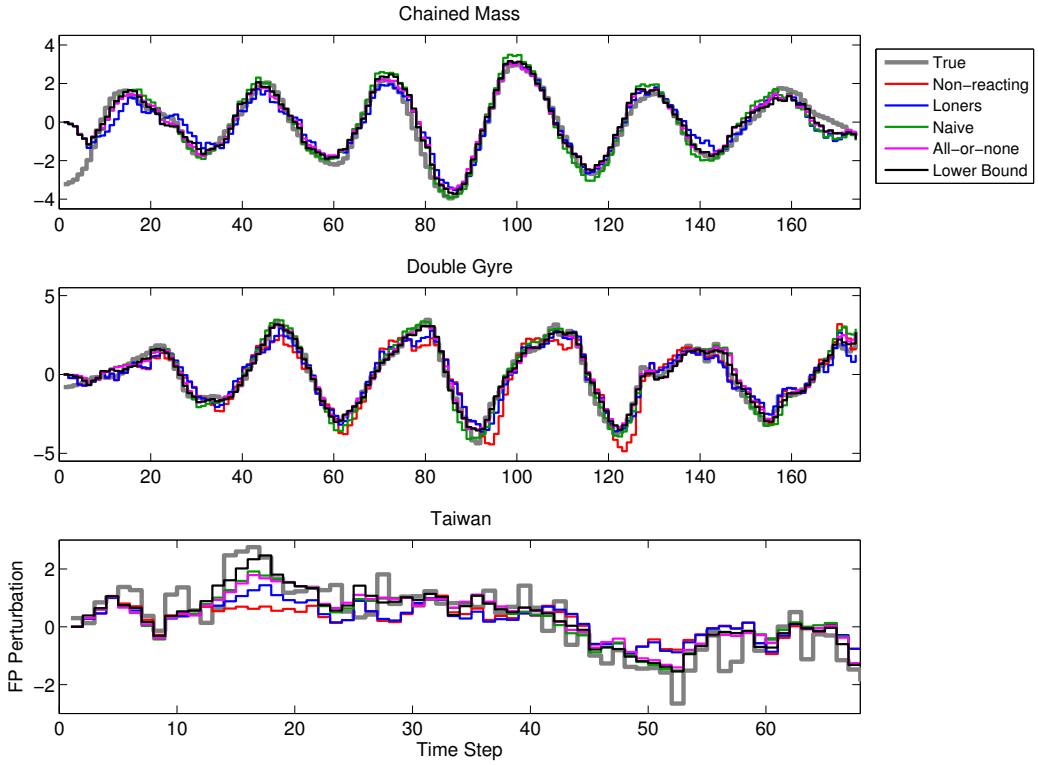


Figure 4-14: Estimation performance for one example site and realization. These cases are in the region where All-or-none outperforms Loners: $R_\phi/(\text{mean}(|G|))^2 = 0.82$.

In the control simulation results presented, we can clearly see the importance of actively pursuing the front as opposed to passive tracking, and of global model-based estimation. Moreover, these outcomes illustrate performance of a pursuit system in different conditions, illuminating an interesting trade-off space for designing deployments. An important dimension in it is the number of vehicles and frontal points compared to the length of the feature, as this affects both spatial resolution of the reconstruction as well as the expected level of coupling between sites. These factors

have to be weighed in turn against the costs and capabilities of the individual vehicles. For example, if highly accurate sensors are available, it may not be necessary to set up communication at all for the purposes of oceanographic pursuit. On the other hand, vehicles with less expensive, lower-quality sensors can be deployed in larger numbers and will likely benefit from a coupled model and hence collaboration.

4.7 Summary

We have articulated an integrated framework for dynamically sampling the ocean using a group of communicating mobile agents. Our new concept is that locally linear behavior of an ocean process admits strong estimation and control techniques on short time scales; this will allow multiple cooperating vehicles to decompose spatial and temporal variations in the field, and track a dynamic feature of interest. The stochastic dynamical model supporting our controller design is created via a projection from an ocean forecast ensemble directly into vehicle coordinates, and this is the main innovation of our work. We have demonstrated that control and estimation designs resulting from the identified models are successful, in studies with three example datasets.

Chapter 5

JLS-PPC: A Jump Linear System Framework for Multirate Packetized Predictive Control

The examples of target tracking and oceanographic pursuit have illustrated cooperative missions posed in forms suitable for networked feedback control. This chapter presents a unified formalism for multi-vehicle control and estimation with measurement, control, and control acknowledgment packets all subject to schedules, delays and packet loss. The modular framework is built around a jump linear system (JLS) description that includes Packetized Predictive Control (PPC), a technique that combines the receding horizon optimization of Model Predictive Control with buffering at the actuator. Integration of these elements enables synthesis of a novel technique for estimation using delayed and lossy control acknowledgments—a desirable and practical capability of fielded systems that has not been considered in work to date. This chapter describes the framework, the estimation and control technique, a simple illustrative example, and a few possible extensions. The following chapter presents simulation and field experiment results with JLS-PPC.

5.1 Introduction

Acoustic communication links are a crucial component in the feedback loop for underwater multi-vehicle systems operating under centralized estimation and control. A multi-node system using acoustic communications is subject to multiple-access constraints; the most straightforward approach is interference-free scheduling. Due to bandwidth limitations as well as the fact that most modem hardware today operates in a single frequency band, the most common approach is TDMA scheduling.[1] This results in a multi-rate control system, where measurements arrive at the estimator at different times, and control arrives to an individual vehicle at different times. Packet delays resulting from acoustic propagation time and transmission time are (nearly) deterministic but substantial, especially if relays must be used. Stochastic packet loss complicates control design further.

The difficulty of analysis in networked control has limited work primarily to situations where only a subset of practical conditions are considered, *e.g.* packet loss and disturbances, or schedules and constraints [14, 117]. In this work, we develop an estimation and control framework that handles:

- deterministic TDMA scheduling with communication delays,
- stochastic packet loss in multiple sensor and control links,
- the possibility of delayed or missing control acknowledgments,
- control quantization and saturation constraints,
- process and sensor noise.

To our knowledge this is the first technique that integrates all of these aspects. Our approach builds on aspects of existing control and estimation techniques: the Kalman filter (KF), model predictive control (MPC), and buffering of control trajectories. The

[1] Staggered or overlapping TDMA schedules that exploit position diversity are an area of active research, *e.g.* [135, 178]

primary contributions are a novel approach for handling delayed or missing acknowledgments within the framework of multirate scheduling and delays with control trajectory buffers, and a modular description of the integrated system in a compact jump system form. We describe the framework and our approach for the various blocks in the subsequent sections, a diagram of the system is shown in **Figure 5-1**. The unified formalism we develop helps maintain clean notation and system description across different multi-vehicle systems and communication scenarios. The modular framework will allow future improvements or modifications to specific blocks without the need to develop an entire system from scratch.

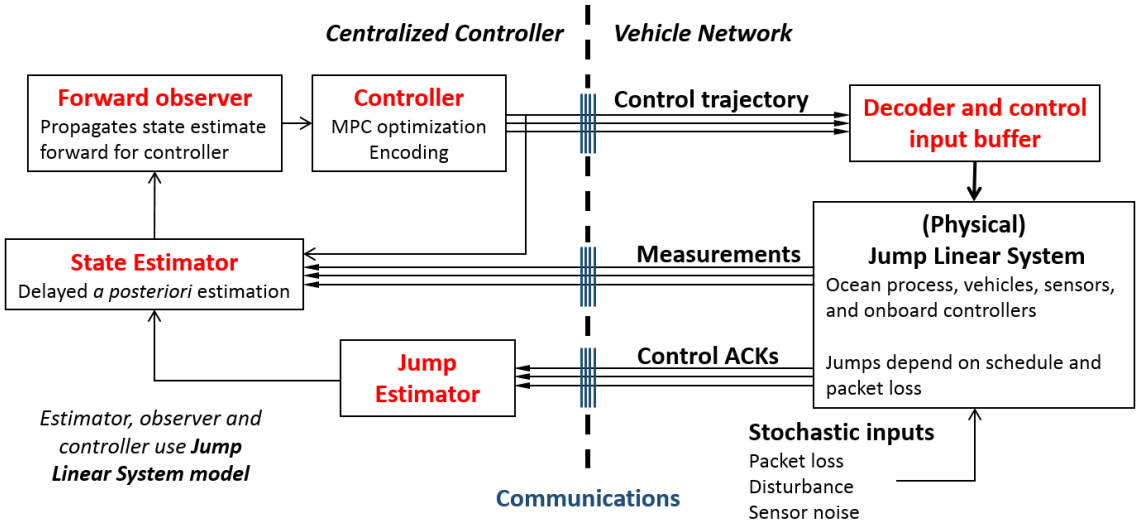


Figure 5-1: Block diagram of the modular JLS framework for multi-vehicle communication and control.

5.2 Overview of approach and prior work

We give background on crucial elements to our approach.

5.2.1 Model Predictive Control (MPC)

Constraints on inputs and states, such as speed saturation, islands, or proximity to ship, are important to model in real physical systems, yet they are not handled by

conventional linear control techniques; *constrained* LQG cannot be solved tractably by dynamic programming. Model predictive control (or receding-horizon control) is a successful technique that involves explicit solution of a finite-horizon optimal control problem at each time step. This comes at a (tractable) computational expense online, but allows significant flexibility in modeling. MPC is often successful in practice but lacks theoretical guarantees [26].

A primary drawback of MPC is that the standard formulation assumes noiseless state feedback. Various techniques for Robust MPC have been proposed, however no current approaches are suitable for use with networked control (primarily due to constraints on estimator structure). A common approach is certainty equivalence (CE-MPC): use an estimator and design control under the assumption that the estimate is the true state and there will be no future disturbances. In this case, the cost function is deterministic. The logic follows from the separation principle in LQG control, and often works well in practice [223,247]. We choose to use CE-MPC for this reason, along with the belief that communication constraints are more important than state constraints, which are where approaches explicitly formulated for uncertainty provide the largest benefit. We discuss Robust MPC in more detail in **Section 5.9.5**.

5.2.2 Packetized Predictive Control (PPC)

A flavor of MPC for stochastic packet-loss between the controller and actuator is called Packetized Predictive Control (PPC) [193], where buffered control trajectories are used in the case of dropped packets. This is an intuitive idea, originally proposed by Bemporad for predictive control of teleoperated systems with unbounded delays [19]. A receding-horizon optimal control problem is solved, and the entire trajectory (or at least some length trajectory—the sent trajectory only must be $\leq$ the computed trajectory) is sent in each control packet. This trajectory is stored in a buffer at the actuator (vehicle), and in the case that a new packet is not available at the next time-step, the next control in the buffered trajectory is implemented; see **Figure 5-2**. Since the communication protocol is time-synchronized, the actuator (vehicle) knows whether a packet was received or not (an implicit ACK is available).

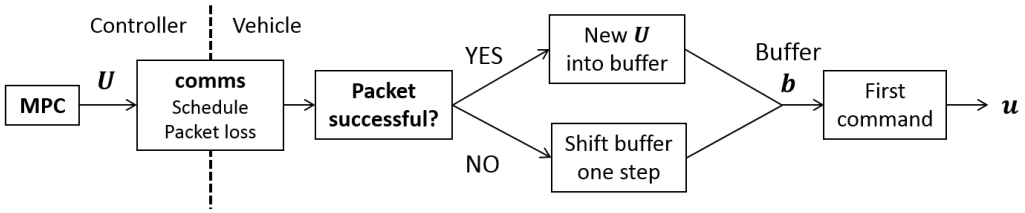


Figure 5-2: Diagram illustrating the Packetized Predictive Control concept.

Benefits of this approach are that no *a priori* specification of control packet success probability is required, any optimization method can be used "under the hood" to generate the control sequences, and the only tuning parameter is the length of the trajectory to be sent. Practically, the length of the trajectory *sent* as well as command resolution can be varied based on the trajectory contents: if the control dies to zero quickly, early commands can be sent at high-resolution; if the control commands are large for a number of steps, they can be sent with less resolution with minimal effect on performance. Extensions have been proposed for sparse control plans [168], and rate-distortion analysis with dithered vector quantizers [192].

5.2.3 Scheduling, delays, ACKS, and the dual effect

Due to the propagation delays of acoustic communications, control cannot be computed based on up-to-date information. The measurement is old by the time it reaches a centralized estimator, and the control command does not reach the actuator, or vehicle, until after computation time plus propagation delay of the control packet. Scheduling introduces additional delays. This timing structure is different than the conventional setup in discrete-time control, where control is computed and applied immediately based on instantaneous measurements.

A major issue in systems with packet loss in the sensor and controller channels is the *dual effect*, which arises when no acknowledgment of control packet success is available [15]. In this case, the control prior to be used in the estimator is uncertain. The information flow of the scheduled control loop adds further complication, as the delays due to scheduling exaggerate the dual effect.

The following basic options may be possible for control packet ACKs (we will explore these later):

- Control ACK is available instantaneously and without loss (usually an unrealistic assumption).
- Control ACK is "piggybacked" with measurement packets (a lossy, delayed ACK).
- Dedicated slots for control ACKs are built into the schedule (likely more reliable and less delayed than a measurement packet).
- No ACKs are used.

We note that some subtle variations are also possible. For example, ACKs may be piggybacked as part of a large measurement packet, but in their own separate block with a different error correction code. In this case, the piggybacked ACK would share the schedule and delay with the measurement packet, but would have a different (likely higher) packet success probability. Another example is if hardware allows for a separate, non-interfering ACK channel in addition to the primary data packet channel (for example, data transmission in a higher frequency band, and ACK transmission in a lower frequency band). In this case, the ACK schedule could have overlap with the control and/or measurement schedules. The JLS framework can handle scenarios such as these as long as they can be described by the scheduling, delay, and loss notation introduced later.

5.2.4 Related Work

Other constructive approaches for handling packet loss have been proposed in various settings. Of particular interest are scenarios where there are losses in both the sensor and measurement channels. Imer *et al.* solves an LQG-like control problem via dynamic programming, for the cases when acknowledgments (ACKs) are present, and not present [123]. Schenato *et al.* solved a similar problem via LMIs [214]. Both of these works consider single "all-or-none" communication channels; we have extended

this to the independent multi-channel case in [196]. Other approaches have taken the Markov Jump Linear System (MJLS) viewpoint, often with an $\mathcal{H}_\infty$ objective. This problem has been solved for the output feedback case with mode observations (ACKs) [96], via an LMI method. The case of no mode observations is not tractable to solve to optimality, but some approaches have been proposed, *e.g.* [238]. MJLS approaches can also handle deterministic schedules, but the tractability of the LMI solutions does not scale well with problem size/schedule length [218]. None of these approaches are able to handle system and control constraints like MPC, nor do they take advantage of PPC buffering schemes.

Multirate control has been studied extensively for unconstrained systems with no packet loss [138]. An early reference on multirate MPC is Lee *et al.* , 1992 [140]. This work considers a measurement schedule only, and is primarily concerned with designing a suitable steady-state estimator.

Millan *et al.* , 2008 consider constrained predictive control with buffering for the case of noiseless state feedback. Liu *et al.* , 2012 present an unconstrained predictive control scheme using buffering that handles multirate and out-of-order measurements [146]. Both of these works assume immediate lossless acknowledgments, and all-or-none losses in the sensor and controller channels. For the case of noisy but lossless measurements, Cunguara *et al.* show that the separation principle holds even without acknowledgments when buffering of extended input schemes is used [57].

The closest work in terms of delays and lack of acknowledgments is described in Pin & Parisini, 2011 [184], and Pin *et al.* , 2009 [183]. These works build on similar work by Varutti & Findeisen [81, 241], and consider stochastic delays and packet-loss with no acknowledgments. By constraining the beginning of a new control trajectory to be equal to the buffered trajectory, the dual effect is circumvented as the estimator knows the input. However, these approaches do not consider multiple communication channels nor multirate scheduling, and may be slow to reject disturbances due to the control constraints during the beginning of each trajectory.

5.3 List of assumptions

We make a number of simplifying assumptions in developing our framework. We collect them all here, but each is explained in context in the relevant section. Many of these assumptions can be relaxed without loss of generality (but with an increase in notational complexity). However, a few are fundamental to our approach and are listed in **bold**.

- **All vehicles have accurate clocks; schedule is time-synchronized**.
- **The communication schedule is given**.
- **All vehicles know the communication schedule.**
- **Packets are coded and have a cyclic redundancy code (CRC)—packets are either received correctly, or dropped completely**.
- Transmissions occur at the start of a slot. Receptions arrive sometime within the slot, and are assumed to be available at the beginning of the next slot.
- Each of the N_v agents has the same number of control inputs N_u and measurements N_y.
- Controls for all vehicles are computed using the same trajectory horizon N_p.
- Control commands and measurements to and from a single vehicle are contained in one packet each.
- The communication schedule is periodic and constant in time.
- All propagation delays are equal to an integer multiple (≥ 1) of the time slot length, and are equal for all similar types of communication links (*e.g.* control delays to all vehicles are the same).
- **There is sufficient memory onboard vehicles and at the controller to keep a buffers of old signals.**

The system model is discretized with the TDMA slot length as the time step. A lower bound for TDMA slot length depends the packet lengths and necessary guard times of the communication system used, and expected propagation delays given the size of the vehicle deployment area. Longer slots can be used if lower duty cycles are desired for energy savings, although at the expense of a longer cycle and slower control system bandwidth. If variable length packets or short computation slots are needed, the TDMA schedule can be constructed with a small "fundamental" slot length, and longer packets can use multiple slots.

5.4 System Definitions

We begin by defining the system under consideration.

5.4.1 Underlying system

We consider a standard discrete-time LTI state-space system:

$$\begin{aligned} \boldsymbol{x}_{t+1} &= A\boldsymbol{x}_t + B^u\boldsymbol{u}_t + B^w\boldsymbol{w}_t \\ \boldsymbol{y}_t &= C\boldsymbol{x}_t + \boldsymbol{\nu}_t, \end{aligned} \tag{5.1}$$

where $\boldsymbol{x}_t \in \mathbb{R}^{N_x \times 1}$, $\boldsymbol{u}_t \in \mathbb{R}^{(N_v N_u)\times 1}$, $\boldsymbol{y}_t \in \mathbb{R}^{(N_v N_y)\times 1}$, $\boldsymbol{w}_t \in \mathbb{R}^{n_w}$ and $\boldsymbol{\nu}_t \in \mathbb{R}^{(N_v N_y)\times 1}$. For simplicity in future notation, we have assumed that there are N_v separate agents (*e.g.* vehicles), each with N_u control inputs and N_y measured outputs.

We will consider minimum and maximum allowed control values as functions of time ($N_u \times 1$ vectors) for each individual agent:

$$\underline{\boldsymbol{u}}_t^i \leq \boldsymbol{u}_t^i \leq \bar{\boldsymbol{u}}_t^i, \quad \forall t \in 1, \ldots, N, \;\; \forall i \in 1, \ldots, N_v,$$

where N is the length of the mission. For brevity, we will refer to these as $\boldsymbol{u} \in \mathcal{U}$.

5.4.2 Flow of information

Deterministic schedules

We consider control design under the assumption that a given schedule has already been designed. Comparison of various schedules can then be performed empirically.

We ignore inter-time-step delays and assume transmissions arrive at the end of their slot within the TDMA schedule. We assume that the system operates in a time-synchronized manner and set the system discretization time step equal to the network time step (one slot of the TDMA schedule).

For simplicity, we will assume all propagation delays are equal to an integer multiple (≥ 1) of the slot length:

- τ_c is the control delay
- τ_m is the measurement delay
- τ_a is the ACK delay.

These can be vectors with different delays for each vehicle without loss of generality. These delays include the time to encode and transmit the packet, the physical propagation delay of the information traveling through space, and any time necessary to decode. As a simplifying assumption we will consider that all measurements (which can be vectors) to and from a single vehicle (denoted $i \in 1, \ldots, N_v$) are contained in one packet; this assumption can easily be relaxed with more complicated notation.[2]

We model the deterministic control, measurement, and ACK packet scheduling

[2]Measurements may also come from non-actuated nodes; these can also be incorporated into the estimator.

policy with indicator variables (each for all $i \in 1, \ldots, N_v$):

$$\pi_t^i = \begin{cases} 1 & \text{if a control packet is planned to be sent at step } t \\ 0 & \text{otherwise} \end{cases}$$

$$\xi_t^i = \begin{cases} 1 & \text{if a measurement packet is planned to be sent at step } t \\ 0 & \text{otherwise} \end{cases}$$

$$\lambda_t^i = \begin{cases} 1 & \text{if a control ACK packet is planned to be sent at step } t \\ 0 & \text{otherwise} \end{cases}$$

Note that the time indices t refer to transmission at the start of a time slot, so that the packet will have arrived at the beginning of the time slot τ_c, τ_m, or τ_a steps in the future.

We additionally define $\tau_{a'}^i$ as the time between a (planned) control packet reception, and the transmission of the corresponding ACK. If $\pi_{t-\tau_c}^i = 1$, then a control packet was *planned to be* received at t, and $\lambda_{t+\tau_{a'}^i}^i = 1$ (an ACK is sent at $t + \tau_{a'}^i$). We assume periodic schedules, so $\tau_{a'}^i$ is a property of the schedule design (although in principle $\tau_{a'}^i$ could vary in time, assuming it is known at the vehicle and controller).

Schedule examples

We will describe two canonical scheduling paradigms for illustration (these will later be compared in simulation/experimental results).

The "Multiplexed" (MX) schedule first sends measurements from each vehicle to the controller, then sends a single broadcast packet containing control trajectories for all vehicles. The "Interleaved" (IL) schedule sends a measurement from a vehicle, then a control packet to that vehicle, repeated for all vehicles. **Figure 5-4** shows the control packets for each type of schedule. For a given N_p, the MX control packet includes N_v times more control actions than the IL packet, making quantization more coarse for a given packet size.

In addition to choice of MX or IL control packets, schedules vary depending on whether control ACKs are "piggybacked" with measurement packets, or sent in their

own dedicated packet. Examples of MX and IL schedules with both of these options are illustrated in **Figure 5-3**. Even with these simple schedule examples, it is clear that scheduling design involves complicated tradeoffs between quantization of control packets, overall length of schedule T^s, ACK reliability, time between measurement and control, and time between control and ACKs $\tau_{a'}^i$.

Stochastic packet-loss

Similar to the deterministic schedule, we define (random) indicator variables to indicate whether packets are successfully received (each for all $i \in 1, \ldots, N_v$):

$$\alpha^i_{t-\tau_c} = \begin{cases} 1 & \text{if a control packet sent at step } t-\tau_c \text{ is received at step } t \\ 0 & \text{otherwise} \end{cases}$$

$$\beta^i_{t-\tau_m} = \begin{cases} 1 & \text{if a measurement packet sent at step } t=\tau_m \text{ is received at step } t \\ 0 & \text{otherwise} \end{cases}$$

$$\gamma^i_{t-\tau_a} = \begin{cases} 1 & \text{if a control ACK packet sent during at } t-\tau_a \text{ is received at step } t \\ 0 & \text{otherwise} \end{cases}$$

We denote the probability of packet success in control channel i as $\bar{\alpha}^i$, in measurement channel i as $\bar{\beta}^i$, and in control ACK channel i as $\bar{\gamma}^i$.[3] In the development of the framework, we do not consider any particular distribution on the packet loss sequences, although common assumptions are that the losses are Bernoulli or Markov.

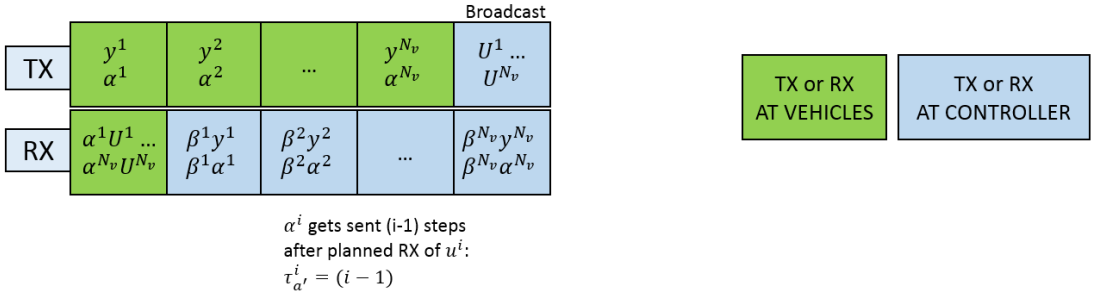
New information is available only if a packet is scheduled *and* received correctly. A measurement *taken* at time $t-\tau_m$ is available at the estimator at time t when $\xi^i_{t-\tau_m}\beta^i_{t-\tau_m} = 1$. Similarly, a control ACK packet sent at time $t-\tau_a$ is available at the estimator at time t when $\lambda^i_{t-\tau_a}\gamma^i_{t-\tau_a} = 1$. A control packet sent at time t is available at the actuator (vehicle) at time $t+\tau_c$ when $\pi^i_t\alpha^i_t = 1$.

Vectors for all vehicles are bold: $\boldsymbol{\pi}_t = [\pi^1_t, \ldots, \pi^{N_v}_t]^T$, and similar for $\boldsymbol{\xi}$, $\boldsymbol{\lambda}$, $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\beta}$.

[3] These probabilities could be time-varying (for example if a channel estimator is being run based on observed performance); for simplicity we leave them as time-invariant.

Multiplexed w/ piggybacked ACK

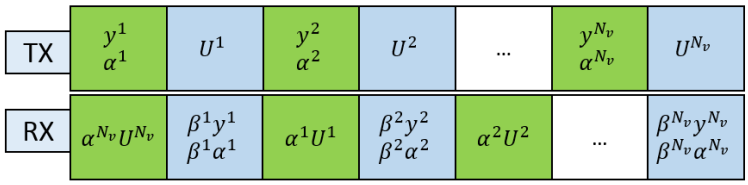
$T^s = N_v + 1, \tau_m = \tau_c = \tau_a = 1, \lambda = \xi$



α^i gets sent (i-1) steps after planned RX of u^i: $\tau^i_{a'} = (i-1)$

Interleaved w/ piggybacked ACK

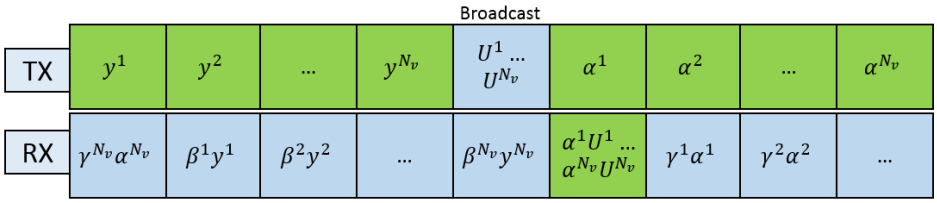
$T^s = 2N_v, \tau_m = \tau_c = \tau_a = 1, \lambda = \xi$



α^i gets sent ($2N_v$-2) steps after planned RX of u^i: $\tau^i_{a'} = (2N_v - 2)$

Multiplexed w/ dedicated immediate ACK

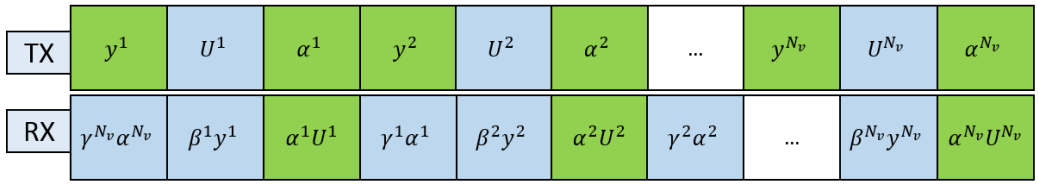
$T^s = 2N_v + 1, \tau_m = \tau_c = \tau_a = 1$



α^i gets sent (i-1) steps after planned RX of u^i: $\tau^i_{a'} = (i-1)$

Interleaved w/ dedicated immediate ACK

$T^s = 3N_v, \tau_m = \tau_c = \tau_a = 1$



α^i gets sent immediately after planned RX of u^i: $\tau^i_{a'} = 0$

Figure 5-3: MX and IL schedules with piggybacked ACKs, and dedicated ACKs

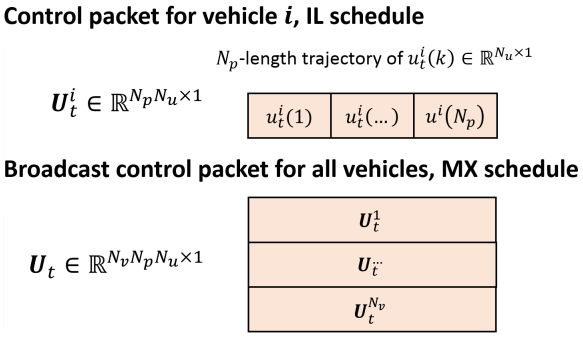


Figure 5-4: Control packets for MX and IL schedules. For a given prediction horizon N_p, the MX broadcast packet contains N_v times more commands than an individual IL packet.

5.5 Stochastic Jump Linear System Framework

We present a stochastic jump linear system framework (JLS) for multi-vehicle packetized predictive control with multirate scheduling and packet loss. This framework describes the evolution of the system with given control commands, schedule, and packet loss sequences. We develop estimation and control strategies in subsequent sections.

5.5.1 Packetized Predictive Control (PPC)

We propose a straightforward extension to the packetized predictive control concept where the combination of deterministic scheduling and stochastic packet loss determines the control to be implemented. This is especially useful when considering control constraints; the vehicle will still drive in the best (open-loop) manner in between packets.

Control plans of length N_p are *computed* whenever $\max \boldsymbol{\pi}_t = 1$, *e.g.* whenever there is a planned control transmission to any vehicle at time t. The control sequence computed at t is designed to take effect at time $t + \tau_c$, as indicated below. We set the trajectory length for the control packet equal to the MPC prediction horizon, although the transmitted trajectory length can be shortened if desired. Control plans are set to dummy variables when they are not computed—the indicator variable for packet success takes the schedule into account and will not update the buffer state

when no control is computed.

Some definitions for a single input system:

- $d_t^i = \pi_{t-\tau_c}^i \alpha_{t-\tau_c}^i$: scalar indicator variable to indicate when a control packet is received at vehicle i at time t.
- $u_t^i \in \mathbb{R}^1$: control command for vehicle i to be executed at time t.
- $U_t^i \in \mathbb{R}^{N_p \times 1}$: vector containing the planned control trajectory for vehicle i
- $b_t^i \in \mathbb{R}^{N_p \times 1}$: buffer state for vehicle i

We first describe the PPC buffer in state space form for a scalar system, following [192]:

$$\begin{aligned} b_t^i &= M b_{t-1}^i (1 - d_t^i) + d_t^i U_t^i \\ u_t^i &= e_1^T b_t^i, \end{aligned} \tag{5.2}$$

with e_1 and M defined as follows.

$$e_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T \in \mathbb{R}^{N_p \times 1}$$

$$M = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ 0 & \cdots & \cdots & \cdots & m_f \end{bmatrix} \in \mathbb{R}^{N_p \times N_p}$$

If zero-control is to be applied in the case that the buffer runs out, then $m_f = 0$. If the previous input is to be held, then $m_f = 1$.

5.5.2 Multi-vehicle PPC

Moving to a multi-vehicle system, with the option for N_u control inputs for each vehicle, requires construction of some larger matrices. We use col to denote the

column-stacking operator, $\otimes$ to denote the Kronecker product, and $.*$ to denote the elementwise product of two vectors.[4]

- $\boldsymbol{D}_t = \text{diag}(\boldsymbol{\pi}_{t-\tau_c} . * \boldsymbol{\alpha}_{t-\tau_c}) \otimes \left(I_{N_p} \otimes I_{N_u}\right) \in \mathbb{R}^{(N_v N_p N_u)\times(N_v N_p N_u)}$: success matrix for controls applied at t.
- $\boldsymbol{u}_t^i = \left[u_t^{(i,1)}, \ldots, u_t^{(i,N_u)}\right]^T \in \mathbb{R}^{N_u \times 1}$: control input for vehicle i.
- $\boldsymbol{U}_t^i = \text{col}\left[\boldsymbol{u}_t^i, \ldots, \boldsymbol{u}_{t+N_p-1}^i\right] \in \mathbb{R}^{(N_p N_u)\times 1}$: stacked vector containing the planned control trajectory for vehicle i.
- $\boldsymbol{U}_t = \text{col}\left[\boldsymbol{U}_t^1, \ldots, \boldsymbol{U}_t^{N_v}\right] \in \mathbb{R}^{(N_v N_p N_u)\times 1}$: stacked vector of control trajectories for all vehicles/channels.
- $\boldsymbol{u}_t = \text{col}\left[\boldsymbol{u}_t^1, \ldots, \boldsymbol{u}_t^{N_v}\right] \in \mathbb{R}^{(N_v N_u)\times 1}$: vector of control commands for all vehicles.
- $\boldsymbol{b}_t^i \in \mathbb{R}^{(N_p N_u)\times 1}$: buffer state for vehicle i (N_p vectors of size $\mathbb{R}^{N_u \times 1}$ are stacked).
- $\boldsymbol{b}_t = \text{col}\left[\boldsymbol{b}_t^1, \ldots, \boldsymbol{b}_t^{N_v}\right] \in \mathbb{R}^{(N_v N_p N_u)\times 1}$: stacked vector of all buffer states.
- $\boldsymbol{M} = I_{N_v} \otimes M \otimes I_{N_u} \in \mathbb{R}^{(N_v N_p N_u)\times(N_v N_p N_u)}$: augmented buffer shift matrix.
- $\boldsymbol{E}_1 = I_{N_v} \otimes \left(e_1^T \otimes I_{N_u}\right) \in \mathbb{R}^{(N_v N_u)\times(N_v N_p N_u)}$: augmented selection matrix.

The buffer state $\boldsymbol{b}_t$ and control $\boldsymbol{u}_t$ for all vehicles is updated with

$$\boldsymbol{b}_t = \boldsymbol{M}(I - \boldsymbol{D}_t)\boldsymbol{b}_{t-1} + \boldsymbol{D}_t \boldsymbol{U}_t$$
$$\boldsymbol{u}_t = \boldsymbol{E}_1 \boldsymbol{b}_t.$$

[4]When column-stacking and constructing augmented matrices, we use the convention that control inputs for a single vehicle go together in the "innermost" grouping, trajectories of (stacked) input vectors are grouped next, and finally blocks of these stacked trajectories for each vehicle are stacked.

5.5.3 Augmented System Dynamics

We augment the system dynamics to include the PPC buffer. The augmented state vector $\boldsymbol{X}$ includes the system states $\boldsymbol{x}$ as well as the buffer $\boldsymbol{b}$:

$$\boldsymbol{X}_t = \begin{Bmatrix} \boldsymbol{x}_{t+1} \\ \boldsymbol{b}_t \end{Bmatrix}.$$

The system update is described in the form $\boldsymbol{X}_{t+1} = \mathcal{A}_t \boldsymbol{X}_t + \mathcal{B}_t^u \boldsymbol{U}_t + \mathcal{B}^w \boldsymbol{w}_t$, given in **Equation 5.3**. We formulate the output equations from the perspective of the centralized estimator. Measurement $\boldsymbol{y}_{t-\tau_m}$ arrives at the estimator at time t, according to the success matrix $\boldsymbol{S}_{t-\tau_m}$ described below. The output equation is given in **Equation 5.4**.

ACKs from each vehicle are an additional output. Since ACKs may be lossy or not available, we define $\tilde{\boldsymbol{\alpha}}_t$ as the control packet success *as known to the estimator*. If $\lambda_t^i \gamma_t^i = 1$, then the ACK $\alpha^i_{t-\tau_c-\tau^i_{a'}}$ is available at the estimator at time $t+\tau_a$. Since ACKs may be lossy, we define a success matrix for ACKs, $\boldsymbol{a}_t$, in a similar manner as for measurements. With slight abuse of notation, $\underline{\boldsymbol{\alpha}}_{t-\tau_a}$ is an array of ACKs sent from all vehicles at $t-\tau_a$: $\left[\alpha^1_{t-\tau_a-\tau^1_{a'}-\tau_c}, \ldots, \alpha^N_{t-\tau_a-\tau^N_{a'}-\tau_c}\right]^T$. Note that each vehicle's ACK may correspond to a different control trajectory, depending on $\tau^i_{a'}$. If ACKs are sent back with measurements, $\boldsymbol{\lambda} = \boldsymbol{\xi}$, $\boldsymbol{\gamma} = \boldsymbol{\beta}$, and $\tau_a = \tau_m$.

Further definitions:

- $\boldsymbol{S}_t = I_{N_y} \otimes \text{diag}(\boldsymbol{\xi}_t . * \boldsymbol{\beta}_t) \in \mathbb{R}^{(N_v N_y)\times(N_v N_y)}$: success matrix for all measurement channels, sent at $t-\tau_m$.

- $\boldsymbol{a}_t = \text{diag}(\boldsymbol{\lambda}_t . * \boldsymbol{\gamma}_t) \in \mathbb{R}^{N_v \times N_v}$: ACK success indicator matrix for all vehicles, sent at $t-\tau_a$.

$$\begin{Bmatrix} \boldsymbol{x}_{t+1} \\ \boldsymbol{b}_t \end{Bmatrix} = \begin{bmatrix} A & B^u \boldsymbol{E}_1 \boldsymbol{M}(I - \boldsymbol{D}_t) \\ 0 & \boldsymbol{M}(I - \boldsymbol{D}_t) \end{bmatrix} \begin{Bmatrix} \boldsymbol{x}_t \\ \boldsymbol{b}_{t-1} \end{Bmatrix} + \begin{bmatrix} B^u \boldsymbol{E}_1 \boldsymbol{D}_t \\ \boldsymbol{D}_t \end{bmatrix} \boldsymbol{U}_t + \begin{bmatrix} B^w \\ 0 \end{bmatrix} \boldsymbol{w}_t \quad (5.3)$$

$$\begin{Bmatrix} \boldsymbol{y}_{t-\tau_m} \\ \tilde{\boldsymbol{\alpha}}_{t-\tau_a} \end{Bmatrix} = \begin{bmatrix} \boldsymbol{S}_{t-\tau_m} C & 0 \\ 0 & \boldsymbol{a}_{t-\tau_a} \end{bmatrix} \begin{Bmatrix} \boldsymbol{x}_{t-\tau_m} \\ \underline{\boldsymbol{\alpha}}_{t-\tau_a} \end{Bmatrix} + \begin{bmatrix} \boldsymbol{S}_{t-\tau_m} \\ 0 \end{bmatrix} \boldsymbol{\nu}_{t-\tau_m} \quad (5.4)$$

Note that the $\mathcal{A}$ and $\mathcal{B}^u$ matrices are time-varying with $\boldsymbol{D}_t$, which itself has a deterministic component (based on $\boldsymbol{\pi}$) and a stochastic component ($\boldsymbol{\alpha}$).

5.6 Estimation

Our estimation approach is based on the Kalman filter (KF), with W and V as the covariance matrices for process and measurement noise, respectively. The goal of estimation is to prepare the state estimate for use when computing control. At time t, a (partial) measurement of the state at time $t - \tau_m$ may be available, so the *a posteriori* estimate $\hat{\boldsymbol{x}}_{t-\tau_m|t-\tau_m}$ is computed.[5] However, control computed at time t will be received and applied at time $t + \tau_c$, so the estimate must be propagated forward open-loop to this time before use in control computations. An example timing diagram for a two-vehicle MX schedule with no packet loss (deterministic scheduling only) is given in **Figure 5-5**.

Due to scheduling, losses, and delays of measurements and ACKs, the information known at the estimator varies each step in a stochastic manner. The KF can handle intermittent measurements as well as time-varying system matrices [222]. The primary challenge is determining the correct control prior to use when there are control packet losses, delays, and delayed/lossy acknowledgments.

We first represent the basic KF in jump system form, used for both *a posteriori* and open-loop estimation. We then describe novel approaches for handling control priors.

[5] More complex methods for mixed delays can also be used, for example using the methods of Stanway, 2010 [227].

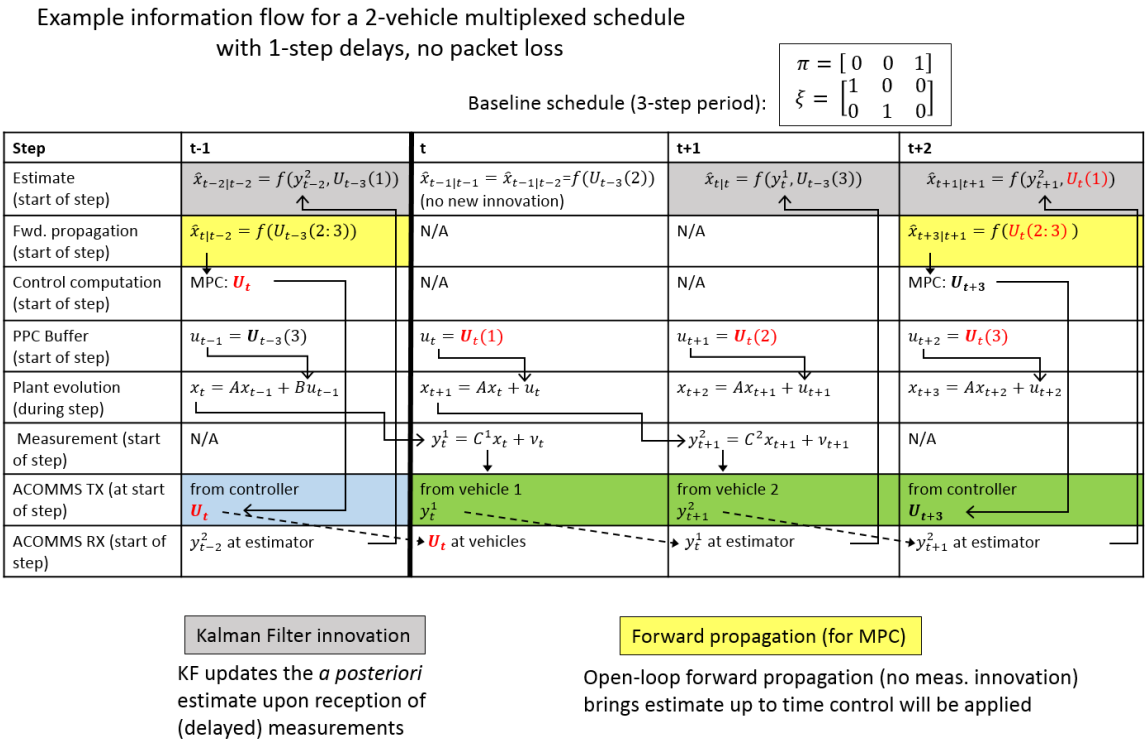


Figure 5-5: Example timing diagram for two-vehicle MX schedule and one-step delays. This diagram assumes that no packet loss is occurring. The capital red U is a control trajectory that is sent and placed into the PPC buffer onboard the vehicle. The subscript refers to the time the trajectory arrives, and the indices inside of the parentheses refer to specific element(s) of that control trajectory.

5.6.1 KF in Jump System Form

Following the system (5.3), we formulate the KF in jump system form. The KF jump system depends on $\hat{\boldsymbol{D}}_t$, which is an estimate of the success matrix $\boldsymbol{D}_t$ defined in **Section 5.5.2**. As described later, $\hat{\boldsymbol{D}}_t$ is set in different ways depending on the availability of control ACKs.

The augmented state estimate is $\hat{\boldsymbol{X}}_{t+1|t+1} = [\hat{\boldsymbol{x}}_{t+1|t+1}^T, \hat{\boldsymbol{b}}_t^T]^T$, where $\hat{\boldsymbol{x}}_{t+1|t+1}$ is the *a posteriori* state estimate and $\hat{\boldsymbol{b}}_t$ is the buffer estimate. The estimated buffer state is used when generating the state estimate priors, but is not updated with the KF innovation. The combination of the buffer and system states in the same JLS description make the KF prediction step straightforward—the usual matrix operations automatically ensure that the buffer is used for the control priors.

The innovation updates only the system states; the error covariance $P \in \mathbb{R}^{N_x \times N_x}$ describes the uncertainty of these states. The standard covariance prior is used $P_{t+1|t} = AP_{t|t}A^T + W$. The Kalman gain $L_{t+1} \in \mathbb{R}^{(N_v N_y) \times N_x}$ is computed in the usual way, with columns wiped out for missing measurements using $\boldsymbol{S}_{t+1}$:

$$L_{t+1} = (P_{t+1|t}C^T)(CP_{t+1|t}C^T + V)^{-1}\boldsymbol{S}_{t+1}. \tag{5.5}$$

The covariance update for $P_{t+1|t+1}$ is standard, using L_{t+1}: $P_{t+1|t+1} = (I - L_{t+1}C)P_{t+1|t}$.

We now give the jump system form of the KF:

$$\begin{Bmatrix} \hat{\boldsymbol{x}}_{t+1|t+1} \\ \hat{\boldsymbol{b}}_t \end{Bmatrix} = \begin{bmatrix} (I - L_{t+1}C)A & (I - L_{t+1}C)B^u\boldsymbol{E}_1\boldsymbol{M}(I - \hat{\boldsymbol{D}}_t) \\ \boldsymbol{0}_{(N_v N_p N_u)\times(N_x)} & \boldsymbol{M}(I - \hat{\boldsymbol{D}}_t) \end{bmatrix} \begin{Bmatrix} \hat{\boldsymbol{x}}_{t|t} \\ \hat{\boldsymbol{b}}_{t-1} \end{Bmatrix} + \begin{bmatrix} (I - L_{t+1}C)B^u\boldsymbol{E}_1\hat{\boldsymbol{D}}_t \\ \hat{\boldsymbol{D}}_t \end{bmatrix} \boldsymbol{U}_t + \begin{bmatrix} L_{t+1} \\ 0 \end{bmatrix} \boldsymbol{y}_{t+1} \tag{5.6}$$

The KF runs each time step using whatever information is available. The result is the *a posteriori* estimate of the system state for the time that the most recently received measurement was *taken—e.g.* at each step t, the estimator updates $\hat{\boldsymbol{x}}_{t-\tau_m|t-\tau_m}$.

5.6.2 Forward propagation for control computation

If control is to be computed and sent at time t, it will arrive and be applied at time $t + \tau_c$. The state estimate input to the MPC solver must be for time $t + \tau_c$. Since the most up-to-date *a posteriori* estimate at time t is $\hat{\boldsymbol{x}}_{t-\tau_m|t-\tau_m}$, the system is run forward open-loop based on the current estimated control buffer (and estimated jump variables $\hat{\boldsymbol{D}}_{t-\tau_m}$ to $\hat{\boldsymbol{D}}_{t+\tau_c-1}$). This generates $\hat{\boldsymbol{X}}_{t+\tau_c|t-\tau_m}$, which is passed to the MPC solver. The timing of the forward propagation for a two-vehicle MX schedule is diagrammed in **Figure 5-5**.

This forward propagation is performed using $\hat{\boldsymbol{X}}_{k+1} = \mathcal{A}_k(\hat{\boldsymbol{D}}_k)\hat{\boldsymbol{X}}_k + \mathcal{B}_k^u(\boldsymbol{D}_k)\boldsymbol{U}_k, \forall k = t - \tau_m, \ldots, t + \tau_c - 1$. (This can also be interpreted as the above jump system KF with $L = 0$).

5.6.3 Setting the jump variable $\hat{\boldsymbol{D}}_t$

The goal when setting the jump variable $\hat{\boldsymbol{D}}_t$ is to exploit all control information available at the estimator. When immediate, lossless ACKs are available, this is simple: the jump variable $\hat{\boldsymbol{D}}_t$ is equal to the true $\boldsymbol{D}_t$. The combination of delayed ACKs and control buffering means that an ACK arrival gives information about the past—back to the time the control buffer that is being acknowledged arrived at the vehicle. Due to the buffer, the ACK is useful for determining the control applied over a number of steps instead of just one control action. To make use of this information, the estimator backs up and re-runs its estimate based on the updated jump variable (using a stored history of past *a posteriori* estimates as well as computed controls and measurements). This procedure is described in **Figure 5-6**.

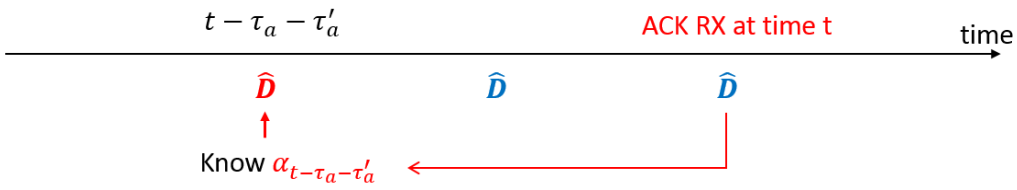


Figure 5-6: Schematic showing the update of the appropriate past $\hat{\boldsymbol{D}}$ upon reception of a delayed ACK.

When dealing with missing ACKs (either lost ACK packets, or no planned ACKs), we formulate the estimator priors based on the expected value of the control action. This follows from the approach for zero control when packets are lost (as considered by Imer *et al.* , Schenato *et al.* , Garone *et al.* , and others), where the control prior is simply the computed control command weighted by $\bar{\alpha}$. With the PPC buffer and MJLS system, the buffer state is estimated by updating the jump system with $\hat{\boldsymbol{D}}_t = \text{diag}(\boldsymbol{\pi}_{t-\tau_c} . * \bar{\boldsymbol{\alpha}}) \otimes \left(I_{N_p} \otimes I_{N_u}\right)$ instead of $\boldsymbol{D}_t$. (Other approaches for handling missing ACKs could be used, but lack any formal justification).

Handling lossy and delayed acknowledgments with arbitrary schedules requires care when performing the "back-up and re-run" strategy. The amount of time to back up depends on all ACKs received at a given time step, since $\tau^i_{a'}$ may be different for each vehicle, and the system is coupled. We initialize $\hat{\boldsymbol{\alpha}}$ to $\bar{\boldsymbol{\alpha}}$ for all time steps, and overwrite when an ACK is available. **Algorithm 2** describes the procedure for updating a sequence of $\hat{\boldsymbol{D}}$ based on lossy, delayed acknowledgments. A timing diagram showing the operation of the algorithm is shown in **Figure 5-7**, for the specific scenario when the previous control trajectory is successfully received at $t - \tau_a - \tau_{a'}$, a measurement and ACK are successfully received at t, and control is to be computed at time t.

Algorithm 2 Determining jump variable with lossy and delayed acknowledgments

Initialize $\hat{\boldsymbol{\alpha}}_t \leftarrow \bar{\boldsymbol{\alpha}}, \forall t = 1, \ldots, N$
for all Time steps $t = 1, \ldots, N$ **do**
  $\boldsymbol{a}_{t-\tau_a} \leftarrow \boldsymbol{\lambda}_{t-\tau_a} . * \boldsymbol{\gamma}_{t-\tau_a}$
  ACKInds $\leftarrow \{i | a^i_{t-\tau_a} = 1\}$
  $t^i_{ack} \leftarrow t - \tau_a - \tau^i_{a'}, \ \forall i \in$ ACKInds
  for all $i \in$ ACKInds **do**
    $\hat{\alpha}^i_{t^i_{ack}-\tau_c} \leftarrow \alpha^i_{t^i_{ack}-\tau_c}$
  end for
  for all $i \in$ ACKInds **do**
    $\hat{\boldsymbol{D}}_{t^i_{ack}} \leftarrow \text{diag}(\boldsymbol{\pi}_{t^i_{ack}-\tau_c} . * \hat{\boldsymbol{\alpha}}_{t^i_{ack}-\tau_c}) \otimes (I_{N_p} \otimes I_{N_u})$
  end for
end for

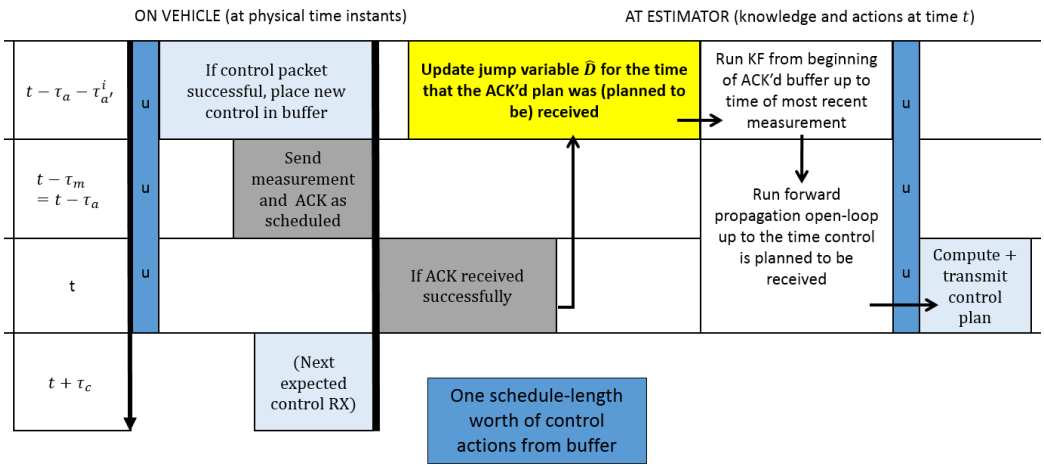


Figure 5-7: Diagram showing timing and operation of the delayed ACK algorithm when the previous control trajectory is successfully received at $t-\tau_a-\tau_{a'}$, a measurement and ACK are successfully received at t, and control is to be computed at time t. The blocks left of the thick black line show events that happen on the physical vehicle at real time instants. Blocks to the right of the thick black line show actions taken at time t by the estimator and controller.

Accuracy of priors

This strategy results in correct KF priors for vehicle i when the following occur:

- A new ACK is received successfully
- The received ACK indicates the previous control packet was successful (positive ACK)
- $\tau_a+\tau^i_{a'} \leq T^s$, where T^s is the length of the periodic schedule.

If the schedule is shorter than $\tau_a+\tau^i_{a'}$, then there are steps near the end of the re-run section where the control priors must be estimated (a new control plan is planned to arrive inside of the re-run section, but after the recent ACK was sent). Priors remain correct for the forward propagation section if $\tau_a+\tau^i_{a'}+\tau_c \leq T^s$.

If an ACK is not received successfully, the priors are incorrect since the expected value of the buffer is used. If an ACK is received successfully, but the ACK indicates that the previous control packet was *not* received successfully (a negative acknowledgment, or NACK), then the current buffer estimate is used for the prior. As such, the accuracy of the priors depends on the prior state of the buffer estimate.

When the priors are wrong, the estimation accuracy suffers. The "resetting" of priors to their accurate values when a positive ACK is received helps estimation performance, however the negative effect of previous dropped measurements and/or ACKs can linger. One approach around this is to send back a history of ACKs and/or measurements in each packet. This way, if a measurement and/or ACK packet arrives, the estimator can be backed-up to the last point in time where the priors were correct, and re-run from there. ACK packets from each vehicle could be of different lengths, *e.g.* one vehicle's ACK could report about a long history if space allows and if we know the control channel is unreliable. Obviously the length of the ACK history (as well as strength of the error correction coding for the ACKs) affects the size of the packet, bringing up the tradeoff between packet size and reliability. This tradeoff invites future optimization, but is outside of the current scope.

5.7 Multirate MPC formulation

The solution of the MPC optimization is complicated by the schedule and delays—control updates may not be scheduled for all vehicles at once. We compute control actions for a prediction horizon of N_p steps, repeating the process with the new state estimate at each time step. While MPC assumes perfect state information, we initialize the optimization with the state estimate provided by the KF and forward propagation, and solve a deterministic optimal control problem assuming this is the actual state. This certainty-equivalent (CE-MPC) approach is effective and commonly used in industry [247], however, robust MPC techniques could be used as well. Robust MPC is discussed further in **Section 5.9.5**.

The cost function for a mission (steps $t = 1, \ldots, N$) considers a quadratic cost on states (which can be reformulated for any sort of output equation), and quadratic cost on control (which could alternatively be reformulated with an ℓ_1 norm if sparse control is desired):

$$\mathcal{J} = \sum_{t=1}^{N} \left(\boldsymbol{x}_t^T Q_t \boldsymbol{x}_t + \boldsymbol{u}_t^T R_t \boldsymbol{u}_t \right)$$

The value of $\mathcal{J}$ depends on the initial state $\boldsymbol{x}_0$, the measurement and control sequences $\boldsymbol{\pi}$ and $\boldsymbol{\xi}$, and the stochastic variables $\boldsymbol{w}$, $\boldsymbol{\nu}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The design variables are the control sequence $\boldsymbol{u}$ as well as the choice of estimator.

If only partial control updates are to be sent at time t, control actions that will not be updated until future times are not new decision variables—they are constrained to equal the actions that will be executed out of the buffer. For each vehicle we compute k_p^i, the number of time steps to use control priors within the MPC prediction, following $k_p^i = \min\{k|\pi_{t+\tau_c+k}^i = 1, k \geq 0\}$. The MPC prediction uses $\hat{\boldsymbol{b}}_{t+\tau_c-1}$ and $\boldsymbol{k}_p$ to constrain the control priors. We describe the multirate MPC algorithm in **Algorithm 3**, which includes solution of the optimization problem **(5.7)**.

Algorithm 3 Multirate MPC

Require: t, $\boldsymbol{\pi}$, $\boldsymbol{\tau}_c$, τ_m, τ_a, $\boldsymbol{\tau}_{a'}$, N_p, A, B^u, Q, R, $\underline{\boldsymbol{u}}$, $\bar{\boldsymbol{u}}$

AT ESTIMATOR/CONTROLLER, AT STEP t**:**

Receive $\boldsymbol{S}_{t-\tau_m}\boldsymbol{y}_{t-\tau_m}$, and $\boldsymbol{a}_{t-\tau_a}\tilde{\boldsymbol{\alpha}}_{t-\tau_a}$.

Run **Algorithm 2** to compute $\hat{\boldsymbol{D}}_{\min(\boldsymbol{t}_{ack})-\tau_c-1}$ through $\hat{\boldsymbol{D}}_{t+\tau_c-1}$.

Run the KF using $\hat{\boldsymbol{D}}$ computed above, giving $\hat{\boldsymbol{X}}_{t+\tau_c|t-\tau_m}$.

Determine additional length of control priors to use in MPC prediction:

for all Vehicles $i \in 1, \ldots, N_v$ **do**

$k_p^i \leftarrow \min\{k|\pi_{t+\tau_c+k}^i = 1, k \geq 0\}$: first planned update for vehicle i.

end for

Give MPC optimization $\hat{\boldsymbol{X}}_{t+\tau_c|t-\tau_m}$, $\boldsymbol{k}_p$, and system parameters.

Run MPC optimization (5.7), obtain $\boldsymbol{U}_{t+\tau_c^i}^i, \forall i$ s.t. $\pi_t^i = 1$.

Send $\boldsymbol{U}_{t+\tau_c}^i$ (control plans to be sent at time t, received at time $t+\tau_c$).

$$\underset{\boldsymbol{u}}{\text{minimize}}\ \mathcal{J} = \sum_{k=t+\tau_c+1}^{t+\tau_c+N_p} \left(\boldsymbol{x}_k^T Q_k \boldsymbol{x}_k + \boldsymbol{u}_{k-1}^T R_{k-1} \boldsymbol{u}_{k-1}\right)$$

subject to (5.7)

$$\boldsymbol{x}_{k+1} = A\boldsymbol{x}_k + B^u\boldsymbol{u}_k,\ \forall t = t+\tau_c, \ldots, t+\tau_c+N_p$$

$$\boldsymbol{x}_{t+\tau_c} = \hat{\boldsymbol{x}}_{t+\tau_c|t-\tau_m}$$

$$\text{if } k_p^i \geq 1 : \boldsymbol{u}_k^i = \boldsymbol{E}_1^i \boldsymbol{M}^{k_p^i} \hat{\boldsymbol{b}}_{t+\tau_c-1} \quad \forall k = k = t+\tau_c, \ldots, t+k_p^i+\tau_c-1,\ \ \forall i \in 1, \ldots, N_v,$$

$$\underline{\boldsymbol{u}}_k^i \leq \boldsymbol{u}_k^i \leq \bar{\boldsymbol{u}}_k^i, \quad \forall k = t+\tau_c, \ldots, t+N_p+\tau_c,\ \ \forall i \in 1, \ldots, N_v,$$

We use a basic control quantization scheme: n_q control levels in between u_{min} and u_{max} for each command of the trajectory, spaced linearly. While suboptimal, the simplest control quantization method is to solve the MPC optimization assuming no quantization, and then quantize the solution with a linear quantizer.

Alternate methods for quantized MPC and vector quantization of trajectories exist (see **Section 5.9.3**); we leave those for future work.

5.7.1 JLS-PPC Modular code

The elements that make up the JLS-PPC controller are modular, consisting of a jump variable estimator, a lossy state estimator, a forward propagation step, MPC optimization, and the PPC buffer. **Figure 5-8** shows the modular blocks used in the software implementation, including the inputs and outputs.

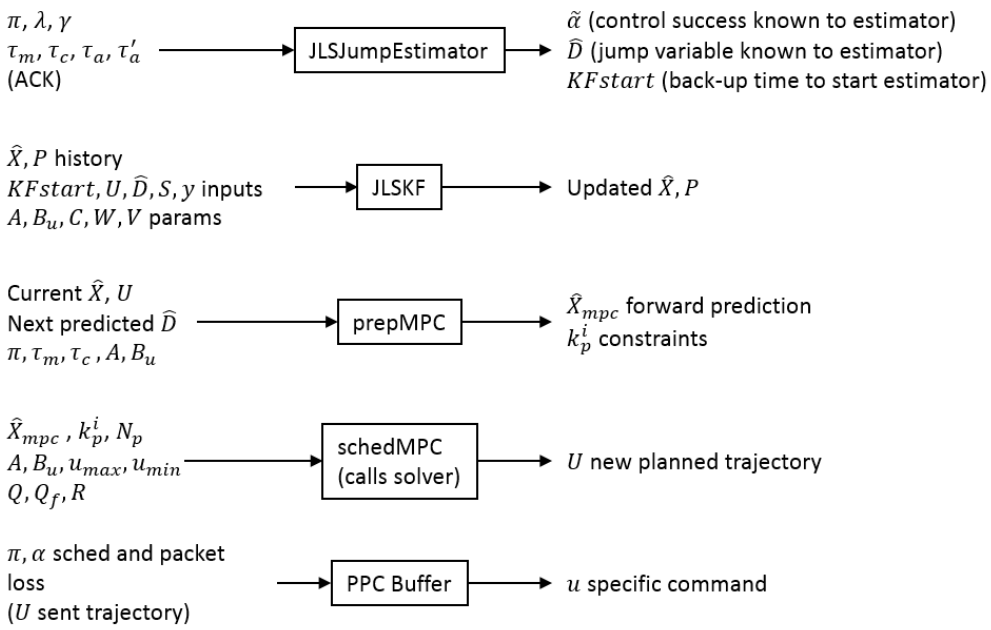


Figure 5-8: Modular code blocks of the JLS-PPC framework, including inputs and outputs.

5.8 Example

We present an example with a simple plant to demonstrate the framework, especially the performance benefits of using piggybacked ACKs over no ACKs. The plant is a double integrator with a single force input u, force disturbances w, and noisy measurement of position. We consider this plant discretized in time with a zero-order-hold on the input and unit time steps:

$$\boldsymbol{x}_{t+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \boldsymbol{x}_t + \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} (u_t + w_t) \tag{5.8}$$

$$y_t = [1 \quad 0]\boldsymbol{x}_t + v_t \tag{5.9}$$

We control the plant using JLS-PPC, subject to schedules and losses for measurements and controls. We compare performance when using piggybacked ACKs, and no ACKs. The MPC horizon N_p is 20 steps, and MPC parameters are $Q = [10, 0; 0, 1]; Q_f = 10Q$, and $R = 1$. Control constraints are set as $u_{max} = 10, u_{min} = -10$, and there are 15 quantization bins linearly spaced between the constraints. Process noise covariance is set as $W = 0.1$, and measurement noise covariance is set as $V = 4$; we note that these are chosen such that the KF priors play a large role in estimation, emphasizing the effects of using piggybacked ACKs. Communications occur in a four-step periodic cycle, according to $\pi = [1, 0, 0, 0]$ and $\xi = [0, 0, 1, 0]$. Communication delays $\tau_c = \tau_m = \tau_a = 1$, and both control and measurement packet success rates are set to sixty percent: $\bar{\alpha} = \bar{\beta} = 0.6$.

We study 300-step simulations of this system, and average results over 25 trials. For each trial, the same exact packet loss and noise sequences are used for both ACK methods. Results are given in **Table 5.1**. Dramatic benefits are observed by using piggybacked ACKs—estimation error improves by over a factor of five, and positioning error improves by over a factor of twelve.

We show time series results for a single trial in **Figure 5-9**. In this trial, the system effectively went unstable when no ACKs were used, resulting in very large estimation and positioning errors. The RMS estimation error with piggybacked ACKs was 5.46,

Table 5.1: Performance comparison between controllers using piggybacked ACKs, and controllers using no ACKs. Results are the RMS error of each trial over time, averaged over 25 trials.

Controller	Lower Bound	Piggyback ACKs	No ACKs
Estimation error	2.2	10.2	55.4
Position error	5.2	26.0	312.0

and 85.0 when using no ACKs. The RMS positioning error with piggybacked ACKs was 15.37, and 767.65 when using no ACKs. In **Figures 5-10** and **5-11**, we zoom-in and illustrate the origins of the instability when using the no ACK controller. Despite being subject to the same packet loss and noise sequences, the piggybacked ACK controller behaves well.

We note a few important observations when comparing time series of individual trials. In between measurement receptions, the estimate usually drifts away from the true state. This effect is more pronounced when disturbances are large and/or control priors are inaccurate. When measurement packets arrive successfully, the estimate jumps (sometimes quite far) back towards the true position, as expected. Measurement packets carrying piggybacked ACKs serve to improve the knowledge of the control buffer, improving the priors and resulting in more accurate estimation in between measurement receptions. Even with successful measurements, the estimate can drift considerably in the no ACKs case because control priors are constructed using the expected value of the control action. Intuition suggests successful control packet receptions are good for system performance, but with no ACKs, if the control packet success probability is low but a sequence of consecutive control receptions occurs, estimation (and subsequently positioning) will suffer since the priors are far from correct! This can be observed in steps 17-25 in **Figure 5-10**. With piggybacked ACKs, estimation is still good even when control packets are lost, (since measurements are arriving successfully, and delayed ACKs are used).

Figure 5-10 shows the beginning of the trial, zoomed-in from the beginning up to step 50. A measurement is lost at step 7, causing the estimate to lag. When a new measurement arrives successfully at step 11, the estimate snaps back on for both

methods. Between step 11 and 43, however, the estimate is much more accurate for the piggybacked ACK controller, since the expected value used in priors with no ACKs is not accurate. At step 43, another measurement is lost. The piggybacked ACK estimate remains well-behaved because positioning error was low so little control was planned, and because the control priors are known more accurately due to previous successful ACKs. The estimate with no ACK is very far off, and the effect of the incorrect control prior is visible in the incorrect slope of the estimate between steps 41 and 46. The estimation errors with the no ACK controller result in poor choices of control, driving the system away from its setpoint.

Figure 5-11 shows steps 40-80—the aftermath of the errors that begin to accumulate in the end of the previous plot. The piggybacked ACK estimate and control perform well, while the incorrect priors induce instability when no ACKs are used.

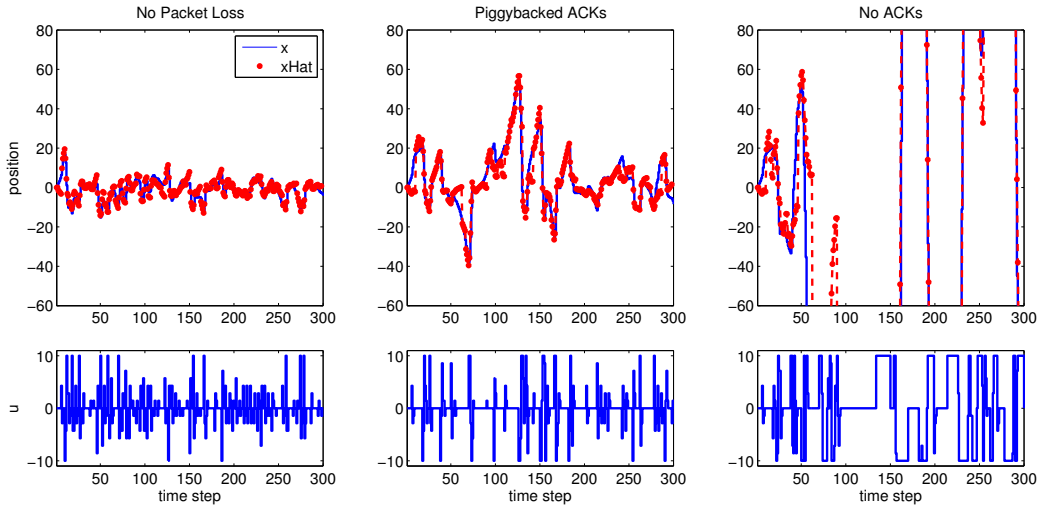


Figure 5-9: Time series results for a single trial. The lower bound using scheduled communications but no packet loss is on the left. The controller using piggybacked ACKs is in the middle, and the controller using no ACKs on the right. RMS estimation error for piggybacked ACKs was 5.46, and the RMS positioning error was 15.37. RMS estimation error for no ACKs was 85.04, and the RMS positioning error was 767.65.

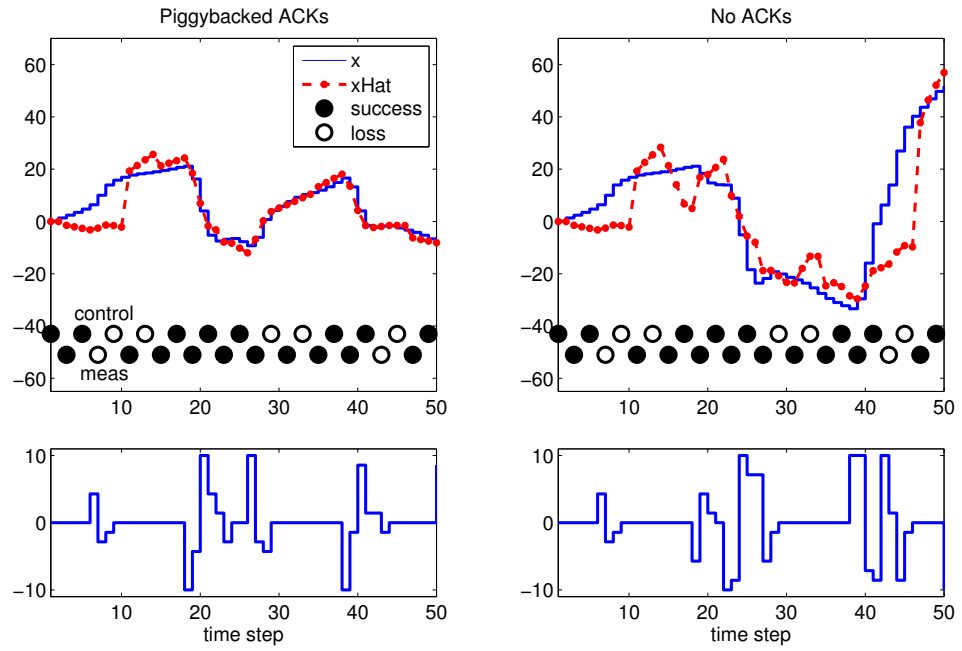


Figure 5-10: Time series results for a single trial, using piggybacked ACKs on the left, and using no ACKs on the right.

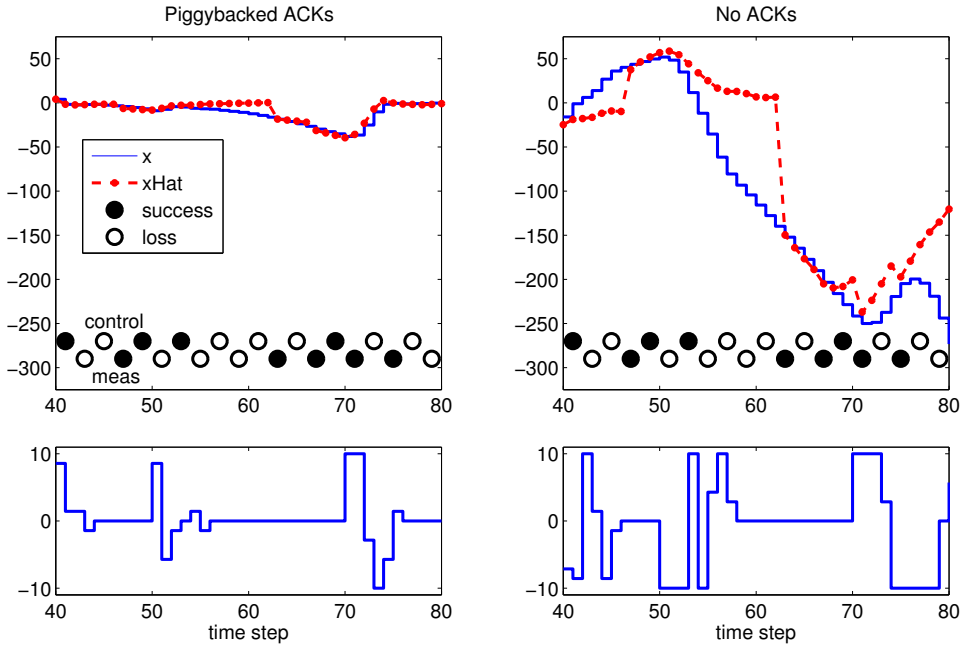


Figure 5-11: Time series results for a single trial, using piggybacked ACKs on the left, and using no ACKs on the right.

5.9 Alternate approaches

This section will articulate some alternate approaches and extensions that are not used in the JLS-PPC framework described above, but may be useful.

5.9.1 Improved estimation when no ACKs are available

The approach developed in **Section 5.6.3** uses the expected value of the control action for the control prior when ACKs are not available. A drawback of this approach is that the expected value of the control is a mix between the previous buffer and the new command, and will never actually be applied.

An alternate approach is to use ideas from maximum likelihood estimation and compare the estimator priors for different scenarios of packet losses using residuals. This idea is used in communications for simultaneous channel estimation and decoding, known as per-survivor processing [194]. Similar approaches using banks of estimators are used for fault-tolerant control when sensors or actuators may break down [254, 255]. Specific to the no-ACK networked control scenario, Epstein *et al.* study mode observer techniques for recovering the "fate" of the control signal applied at the plant [71, 72]; the accuracy of their scheme depends on the magnitude of the control compared to noise and disturbances. Blind & Allgower extend the methods of Epstein *et al.* to the case where measurements are lost [31]; however the fate estimates can only be updated when measurements are available. In summary, all of the above methods use measurement information (output of the plant) to try to infer the actual control applied and improve state estimation.

We do not consider these approaches in our current formulation because we focus on the "piggybacked" ACK case, where ACKs are added onto measurement packets.[6] With "piggybacked" ACKs, the ACK arrives only when measurements arrive, and thus there is little or no information to use for improving estimation compared

[6] We choose to focus on "piggybacked" ACKs because they are the most practical approach for the underwater multi-vehicle missions we consider. Dedicated ACK packets take up time in the schedule, and optimized ACK packets are often not built in to commercial modem systems. Data packets used for measurements are often large, however, and the addition of one extra ACK bit should be negligible.

to the expected-value-control prior approach. There is never a case of a successful measurement but an unsuccessful ACK, which is the case considered by the above methods.

5.9.2 Covariance modifications when control is uncertain

When immediate, lossless ACKs are not available, the actual control executed onboard the vehicles is uncertain due to packet loss. It is possible to derive an additional term to add to the KF covariance computation that takes this into account. In simulations, this has not proven to consistently improve estimation performance, so the results of the next chapter do not include this adjustment. However, we believe with some tweaking this approach may prove useful in the future, possibly in combination with the robust MPC techniques described in **Section 5.9.5**.

For simplicity, we will first develop the covariance modifications for a system with a scalar control action and packet success variable α_t, with $\mathrm{E}(\alpha_t) = \bar{\alpha}$. Consider a time t, with the most recent *planned* control arrival at $t-\tilde{k}_t$. $\tilde{k}_t = \min\{k|\pi_{t-\tau_c-k} = 1, k \geq 0\}$. There are two options: if the control packet is successful (*e.g.* $\alpha_{t-\tilde{k}_t-\tau_c} = 1$), then the appropriate action of the *recent computed/sent* control trajectory $\boldsymbol{U}_{t-\tilde{k}_t}$ is executed. We denote this as $\tilde{u}_t = e_1^T M^{\tilde{k}_t} U_{t-\tilde{k}_t}$. If the control packet is lost (*e.g.* $\alpha_{t-\tilde{k}_t-\tau_c} = 0$), then control is executed from the vehicle buffer (which is a function of the history of U and α). We denote this control as $\hat{u}_t = e_1^T M b_{t-1}$.

Depending on the ACK structure, b_{t-1} may be known exactly (for example, if there are delayed ACKs that arrive in between the sending of the previous control trajectory and the current time), or a buffer estimate $\hat{b}_{t-1}$ may need to be used (following the no-ACK KF priors developed above). In this case, is there an "extra" added covariance due to the fact that the buffer is not known exactly.[7]

The system evolves according to

$$x_{t+1} = Ax_t + \alpha_k B\tilde{u}_t + (1-\alpha_t)B\hat{u}_t + w_t \tag{5.10}$$

[7]This added uncertainty should be quantified in some way; this topic is outside the current scope.

We can write the prediction error as follows:

$$e_{t+1|t} = Ae_{t|t} + (\alpha_t - \bar{\alpha})B\tilde{u} - ((1-\alpha_t) - (1-\bar{\alpha}))\,B\hat{u} + w_t \tag{5.11}$$

The covariance prior is the expected value: $P_{t+1|t} = \mathrm{E}\left[e_{t+1|t}e_{t+1|t}^T\right]$. Following the approach used in Garone *et al.* , 2010 [94], this can be rewritten as follows:

$$\begin{aligned} P_{t+1|t} &= \mathrm{E}\left[Ae_{t|t}e_{t|t}^T A^T\right] + \mathrm{E}[w_t w_t^T] + \mathrm{E}\left[(\alpha_t - \bar{\alpha})^2\right] B\tilde{u}_t\tilde{u}_t^T B^T + \mathrm{E}\left[((1-\alpha_t) - (1-\bar{\alpha}))\right] B\hat{u}_t\hat{u}_t^T B^T \\ &= AP_{t|t}A^T + Q + (1-\bar{\alpha})\bar{\alpha}B\tilde{u}_t\tilde{u}_t^T B^T - (1-\bar{\alpha})\bar{\alpha}B\hat{u}_t\hat{u}_t^T B^T \\ &= AP_{t|t}A^T + Q + (1-\bar{\alpha})\bar{\alpha}B(\tilde{u}_t - \hat{u}_t)(\tilde{u}_t - \hat{u}_t)^T B^T \end{aligned}$$

For the multi-channel case, we define $\tilde{\boldsymbol{u}}_t$ and $\hat{\boldsymbol{u}}_t$ as the appropriate $(N_v N_u) \times 1$ control vectors corresponding to control actions from the recent computed plan, vs. previous buffer. With $q = (\tilde{\boldsymbol{u}}_t - \hat{\boldsymbol{u}}_t)(\tilde{\boldsymbol{u}}_t - \hat{\boldsymbol{u}}_t)^T$, we have

$$P_{t+1|t} = AP_{t|t}A^T + Q + B(E\{\boldsymbol{\alpha}_t' q \boldsymbol{\alpha}_t'\} + q - \bar{\boldsymbol{\alpha}}' q - q\bar{\boldsymbol{\alpha}}')B^T, \tag{5.12}$$

where $\alpha' = \mathrm{diag}(\boldsymbol{\alpha})$ (and similarly for $\bar{\boldsymbol{\alpha}}'$), and using

$$E\{\alpha' Z \alpha'\} = \begin{cases} \bar{\alpha}' Z \bar{\alpha}' & \text{for the off-diagonal elements} \\ \bar{\alpha}' Z & \text{for the diagonal elements,} \end{cases} \tag{5.13}$$

5.9.3 Closed-form quantized MPC

Quantization of control commands is important with PPC since long trajectories of control commands are sent in each packet. Additionally, the multiplexed schedule described in **Section 5.4.2** uses a broadcast control packet sent to all vehicles, which increases the number of commands in a single packet by a factor of N_v. Our simulations and experiments in **Chapter 6** look at the effects of control packet size with a naïve quantization scheme (first solve a non-quantized MPC optimization, then quantize the solution with a linear nearest-neighbor quantizer).

Closed form solutions for constrained MPC give analytic expressions for the con-

trol law depending on which partition of the state space the system is in [22, 219]. Building on this idea, solutions for optimal quantized MPC exist, where vector quantization is used on trajectories of commands. Quevedo, Goodwin and De Dona present a closed form solution for this scenario, where a coordinate transform plus nearest-neighbor vector quantizer is shown to be optimal, and a similar state space partition can be used to determine the appropriate control law [189]. The drawback of this approach is that the codebook (for one control input) is of size $n_u^{N_p}$, where n_u is the number of quantization levels for control commands, and N_p is the trajectory length. The codebook size and nearest-neighbor search used for encoding do not scale well, and become quite computationally intensive as the number of inputs and length of trajectory increase.

5.9.4 Application-specific source coding and quantized estimation

Application-specific source coding could potentially be used for both control commands as well as measurements. For control commands, a codebook could be constructed based off of a pdf of the (expected) speeds of the perturbations (or alternatively, off of the non-quantized commands from simulations). This sort of codebook could used with modifications of the closed-form MPC described above.

We specifically choose not to consider quantization of measurements in this work, as this is a highly application-specific problem. Measurements could be of many flavors—a simple scalar measurement, average of a scalar measurement over time, more complex "metameasurements" such as a gradient estimate from a short local survey, etc. A given problem could benefit from a focused source coding design, either lossless or compressed. If measurement quantization is severe, quantized estimation techniques can be used.

5.9.5 Robust MPC

A drawback of MPC in its standard form is that the problem is formulated for noiseless state feedback. However, often all states are not measured, and there is sensor noise and system disturbances. A common approach is certainty equivalence (CE-MPC): use an estimator and design control under the assumption that the estimate is the true state and there will be no future disturbances. In this case, the cost function is deterministic. The logic follows from the separation principle in LQG control, and often works well in practice. Wang and Boyd [247] derive performance bounds for stochastic control and show that CE-MPC is close to optimal in three examples. Skaf and Boyd [223] show in one numerical example of stochastic unconstrained supply chain optimization that the performance of CE-MPC is very close to that of affine-recourse MPC, and that both are close to the prescient optimal value. Chuang *et al.* [49] study the optimality of certainty equivalence for expected value problems and provide an algorithm for determining partitions of the state space where the certainty equivalent controller is optimal.

We focus on deterministic MPC computations for a number of reasons. First, robust or stochastic approaches often do not have significant gains in cost (expected or worst-case) compared to CE-MPC solutions for systems driven by Gaussian noise (a reasonable assumption for OP, since we use stochastic identification to capture uncertainty). Second, robust or stochastic approaches are not as flexible in terms of adding in scheduling and packet-loss (specifically time-varying estimators). Third, we believe the communication constraints are more important than state constraints, which are where approaches explicitly formulated for uncertainty provide the largest benefit. However, we give a brief discussion on the available techniques for robust MPC for context, and in the hopes that they can be integrated into the JLS framework in the future.

Robust MPC models have been presented that include parametric uncertainty in the system model as well as additive disturbances, both bounded and stochastic. Mayne *et al.* present an overview of MPC with disturbances [158], and a Goodwin

et al. give a survey of more recent work [101]. Cost functions for robust MPC have been posed in min-max (worst-case) as well as stochastic (expected value) forms. Additionally, uncertainty and disturbances bring up issues of feasibility with state constraints, which are usually considered in a probabilistic sense.

A major topic of theoretical interest in robust MPC is that the effects of feedback should be considered in the robust control design to combat against excessive conservatism. Feedback offers recourse for future disturbances – the control policy can be made less conservative if feedback is designed in. As with multi-stage stochastic programming, this is a very difficult (or intractable) problem to solve optimally. Based on ideas of the Affinely-adaptive Adjustable Robust Counterpart (AARC) from Ben-Tal and Nemirovski [23], approaches have been proposed to restrict the optimization to affine feedback policies, beginning with [236]. Goulart *et al.* formulate the affine feedback policy as a convex problem using disturbance feedback [104]. The affine feedback method is considered for time-invariant observers in [103], and the Kalman Filter in Hokayem *et al.* , 2012 [118]. Mayne *et al.* take a slightly different approach where the initial state is used as an optimization parameter in order to increase the domain of attraction for robust state feedback MPC [157], as well as for time-invariant observers [155]. Transient dynamics of the observer is considered in [156], but the observer gain itself must be time-invariant. Work by Cannon *et al.* look at "tube-based" robust MPC under closed-loop feedback [39,40]. Unfortunately, all closed-loop prediction methods are (as far as we aware) unable to handle time-varying estimator structures, which are crucial for handling intermittent measurements.

Bertsimas and Brown take a different approach and present a tractable robust open-loop prediction technique [27]. Their method allows for all system parameters to be time-varying, with bounded uncertainty sets that can be tuned heuristically for a desired level of conservatism. The resulting optimization is a SDP, and they derive an SOCP approximation as well. The min-max objective is computed in an open-loop manner, *e.g.* under the assumption that no feedback is available throughout the horizon to reject disturbances. The open-loop prediction of their approach results in considerable conservatism and constraints must still be implemented in a probabilis-

tic (as opposed to worst-case) manner. We have computationally investigated this technique, and initial findings indicate that the gains from the robust objective are minimal in the Gaussian noise setting.

5.10 Summary

We have described JLS-PPC, a unified framework and control technique for centralized estimation and control with stochastic packet-loss, deterministic delays, and a pre-designed schedule for transmissions of measurements, controls, and control acknowledgments. The ability to consider all of these communication constraints is a new capability in the literature, and our framework helps manage the complexity of system description and notation. We have outlined the notation and assumptions, described candidate scheduling paradigms, presented the jump linear description of the system, and finally developed specific estimation and control algorithms. The elements that make up the JLS-PPC controller are modular, consisting of a jump variable estimator, a lossy state estimator, a forward propagation step, MPC optimization, and the PPC buffer. Our jump estimator algorithm uses a novel approach for using delayed and lossy control packet acknowledgments, made possible by the use of the PPC buffer and the JLS system description. An illustrative example with a simple system demonstrates that the benefits of using lossy, delayed ACKs can be dramatic. We have described the algorithms that our JLS-PPC implementation is based on, and noted that future improvements to any individual block may be used with the rest of the existing framework.

Chapter 6

Pursuit Experiments with JLS-PPC

We present simulation and field experiments demonstrating the JLS-PPC controller in pursuit missions. The feature is a simulated chained mass front similar to the one studied in **Chapter 4**. The field experiments use three autonomous surface vehicles towing acoustic modems, tracking and pursuing the simulated feature. To focus on control performance, "hybrid" measurements are created using the vehicles positions and simulated gradients. The acoustic communications are fully realistic, using TDMA scheduling and quantized packets, and subject to packet loss. We also present simulation results demonstrating the performance improvements of JLS-PPC over independent vehicles, comparison of two scheduling paradigms, and scalability to larger fleet sizes. A design tradeoff study between control quantization and packet loss is demonstrated using the simulation framework, and finally, we present results showing the benefits of using piggybacked ACKs.

6.1 Setup

We study the performance of the JLS-PPC controller with different communication schedule types (MX and IL), packet loss rates, quantization levels, and sensor noise. We use piggybacked ACKs for all results in this chapter, except for the comparison

with no ACKs in **Section 6.2.5**. The JLS-PPC controllers are compared to Loners (vehicles operating independently with local uncoupled models) and Lower Bound (coupled model with ideal communications) controllers. The Loners and Lower Bound controllers are similar to those used in the Oceanographic Pursuit studies, described in **Section 4.6.1**.

The system model for simulations and field experiments is kinematic vehicles with speed constraints tracking and pursuing perturbations generated by a chained-mass driven by noise; see the system description in **Section 4.4.3** of **Chapter 4**. The chained mass system has N_v masses arranged nominally along a line and undergoing lateral perturbations. Each mass is tied to ground lightly by a spring k_g and damper b_g, connected to adjacent neighbors with springs k_n and dampers b_n, and forced by zero-mean white noise of variance w_j. As in **Chapter 4**, the chained mass model uses perfectly linear gradients, which decouples vehicle positioning from estimation. For this reason, we consider both estimation and positioning performance in our results. When dealing with true oceanographic features, the arguments of **Chapter 4** will hold—that the linearization of of the projection procedure will be more accurate near the feature itself, and improved positioning should also improve estimation due to more accurate gradients.

In order to evaluate controller performance clearly, the coupled controllers (all except Loners) use the exact chained mass model, as opposed to the outcome of system identification; **Chapter 4** studies projection and identification on ocean model datasets. We note that in contrast to the simulations of **Chapter 4**, the experiments of this chapter include scheduling, delays, and control packet quantization in addition to packet loss. We use the multiplexed (MX) and interleaved (IL) schedules described in **Section 5.4.2** for the JLS-PPC controllers. For fairness, we compare the Loners and Lower Bound controllers with measurement schedules, such that the frequency of measurement updates for each vehicle is the same as for the MX schedule (Loners and Lower Bound are not subject to delays or packet loss, and control is computed and applied every step).

While the Loners in **Chapter 4** used the outcome of uncoupled system identifi-

cation on their specific frontal point, we choose not to use system identification in this chapter for a more clean evaluation of controller performance. For experiments in this chapter, we tried two different model options for Loners. The first is a simple double integrator, and the second is two masses tied together by a spring (not attached at all to ground). The natural frequency of the second model was tuned to match the dominant natural frequency of the chained mass. In simulations, the integrator model consistently outperformed the drifting oscillator model. For some perturbations the drifting oscillator was very accurate, however due to coupling and higher modes, Loners with this model was often considerably out of phase. For this reason, we show Loners results using the double integrator model for the rest of this Chapter.

We focus our studies on *fast dynamics*, and set the parameters of the chained mass system such that there are relatively few control cycles per natural period of the perturbations. For the majority of our experiments, we set the chained mass parameters such that there are three cycles of the IL schedule within the natural period, since the IL schedule is longer than the MX schedule. Parameters of the chained mass system that give this natural frequency are given in **Table 6.1**, for the three vehicle case. Note the light damping and very light spring to ground (to keep perturbations from drifting too far). For other numbers of masses/vehicles, the cycle time increases and we scale k_n appropriately to give the same number of schedule cycles per natural period.

The driving noise of the perturbations are set such that the speeds of the perturbations are usually below but often close to the vehicle speed constraints. If perturbation speeds are larger than the speed constraints, the vehicles are always chasing the feature and the pursuit problem is ill-posed. If perturbation speeds are much slower, than the importance of using MPC over unconstrained control techniques is diminished.

We use simple linear quantization scheme of the previous chapter, with levels set between the max and min speed constraints. We set the MPC prediction horizon as a multiple of the schedule length: $N_p = N_p^{mult} T^s$. This choice is due to the spirit of

Table 6.1: Parameter values used for chained mass system in JLS experiments for $N_v = 3$. The dominant natural frequency of the system is $\omega_n = 0.3487$, for a natural period of 18 steps.

k_n	k_g	b_n	b_g	w_p
0.0405	1e-4	4e-3	0	0.35

packetized predictive control—the vehicle buffer will have a trajectory lasting N_p^{mult} cycles, which can be used for $N_p^{mult} - 1$ consecutive packet losses. A side effect of this choice is that it results in a different horizon for MX vs. IL schedules. If desired, it is possible set MPC horizons longer than the length of the trajectory sent in the PPC packet.

When comparing MX vs. IL schedules, we keep the number of total commands in a packet constant between the schedules. The IL control packet has the trajectory of just one vehicle, although the trajectory is a bit longer than MX due to the longer IL schedule. The MX broadcast control requires squeezing all N_v trajectories into the packet, resulting in coarser quantization.

6.1.1 Setup of Field Experiments

Field experiments were conducted in the Charles River Basin using the kayak system described in Introduction and **Chapter 3**. The vehicles introduce real maneuvering and physical disturbances (*e.g.* wind, waves) into the system. Most importantly, the vehicles are towing acoustic modems that are communicating according to a MX or IL TDMA schedule and subject to real packet loss. Our experimental setup uses synchronized timing on all vehicles, and the TDMA schedule uses five-second communication slots followed by two-second guard time slots. The guard time slots are used for computation. Running the estimator is near-instantaneous, and the MPC control optimization took on average 0.25 s, easily fitting into the guard slots. The discrete-time control system is run using time steps of $\delta t = 7$ s, matching the TDMA slots.

For our hybrid experiments, we simulated the perturbations and constructed hy-

brid measurements based on linear gradients and position of the perturbation and vehicle. Gaussian noise for the hybrid field measurements was added. Vehicle 2D locations were determined using RTK GPS, and for the experiment we projected the position onto the $\mathcal{C}_2$ line (the line normal to the feature, explained in **Section 4.2.4**) to give the vehicle position in the perturbation frame, q.

Vehicle low-level controllers, timing, and acoustic communications were handled within MOOS using custom C++ code. The control algorithms and hybrid feature measurements were implemented on the shore computer in Matlab, using the mex-moos connection utility to pass messages back and forth from the MOOS database. The MPC optimization is modeled in CVX, a Matlab toolbox for prototyping convex optimization problems, and solved using Gurobi, a state-of-the-art commercial solver.

The controller outputs speed commands, which were encoded and sent to the vehicles. The PPC buffer was run onboard each vehicle, outputting a speed command each time step. The vehicle computed a desired waypoint $u \times \delta t$ meters along the $\mathcal{C}_2$ from the *previous* desired waypoint. Using this method, any positioning errors do not accumulate over time. The maximum vehicle speed was 1.2 m/s. Speed constraints for the controller were set as $(1.2\text{m/s}) \times (7\text{s/step}) = 8.4$ m/step, in the positive and negative directions.

For vehicle control, we made a few adjustments to the baseline MOOS waypoint and trackline control. Vehicle speed control was accomplished via an open-loop lookup table for thrust, plus a outer loop adaptation based on GPS speed. GPS speed is a very noisy measurement so this loop was run with heavy filtering, slow bandwidth, and was turned off when sharp turns were made. For trackline control, we adjusted the setup so that the $\mathcal{C}_2$ line was used as the trackline, instead of the line between the vehicle and the next waypoint. Since vehicles often turn around on the $\mathcal{C}_2$ line, we modified the trackline controller for improved performance when making sharp turns: when the heading error is large, the trackpoint moves far away from the vehicle before coming back to its usual position slowly. Additionally, we shrunk the waypoint capture radius for more precise performance.

6.1.2 Vehicle and Comms simulations

Simulations use the same feature model and measurement technique as the field experiments. Vehicles are simulated using (constrained) kinematic models along the $\mathcal{C}_2$ line, *e.g.* no orthogonal positioning effects. Gaussian process noise and measurement noise is added. The same speed constraints were used: $+/-8.4$ horizontal units per step.

Comms are simulated with TDMA schedule with slot equal to the control system time setup, and Bernoulli packet losses. For simplicity, in our simulations we set the packet loss rates equal for all vehicles, and for controls and measurements. Heterogeneous loss rates are easily handled by the JLS-PPC framework; the only loss rate estimate required is for control packets (used by the jump estimator for KF priors when no ACK is available).

6.2 Results

We first present experimental results, to emphasize that the JLS-PPC controller is suitable for field implementation. We then discuss a few major outcomes of simulations.

6.2.1 Field Experiments

We ran experimental trials testing different controllers on the same perturbation instance. The scalar field measurement noise variance was $V_\phi = 0.1$. For estimation of the vehicle position, we set $V_q = 0.1$ m^2 and $W_q = 0.1$ m^2. We then compare these with the performance expected from simulations with the same settings.

Time Series Results

Time series of selected controller runs are shown in **Figures 6-1, 6-2, 6-3, and 6-4**. The true perturbations $\tilde{p}$, estimated perturbations $\hat{p}$, and vehicle positions q are plotted, color-coded by site. The lower subplots show estimation error $(\hat{\tilde{p}} - \tilde{p})$, and

position error $(q - \tilde{p})$. To evaluate performance, we compute the RMS (over time) of the estimation error and position error, averaged across sites.

The Lower Bound controller performs very well, estimation and positioning errors are typical of LQG control, with some larger positioning errors due to vehicle speed constraints. The Loners controller has much larger errors, especially near the peaks of the perturbation oscillations, where the integrator model takes a few schedule cycles to make the turn. These errors combined with speed constraints causes the vehicles lag behind the perturbations considerably, despite the Loners controller not having to deal with communication delays. The JLS-PPC controller with MX schedule and 9-level quantization demonstrates good estimation performance, nearly as good as Lower Bound. Vehicles exhibit a similar phase lag as with the Loners controller, this time caused primarily by the communication delays.

Due to traffic on the Charles River, the experiments were conducted relatively close to the dock, where the modem for the centralized estimation and control center was located. This resulted in few packet losses. To demonstrate the ability of the controller to handle larger packet losses, we ran a field trial where additional Bernoulli packet losses were added in simulation, with a loss probability of 20%. Results for this trial are shown in **Figure 6-4**. With more packet losses, there are some large excursions where the estimate is inaccurate, such as near step 60. When a measurement successfully arrives, the estimate error drops dramatically. Positioning errors can also occur when estimation is relatively accurate, but control packets are dropped. An example of this is before step 120, where three consecutive control packets are lost for the red perturbation/vehicle, causing the vehicle to drive too far in the negative direction. Since $N_p^{mult} = 3$, after these three packet losses, the buffer runs out and vehicle speeds are set to zero, evident at step 120. The next control packet arrives successfully, but the position error remains relatively large while the vehicle is at its speed constraint attempting to catch up to the perturbation.

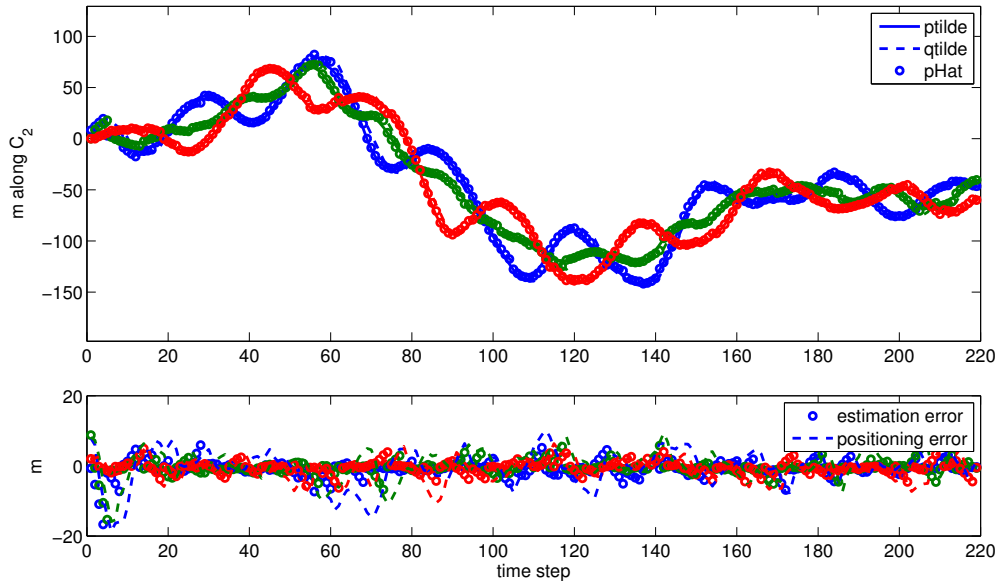


Figure 6-1: Time series of experiment run, Lower Bound. RMS estimation error was 2.21 m, and the RMS positioning error was 4.20 m.

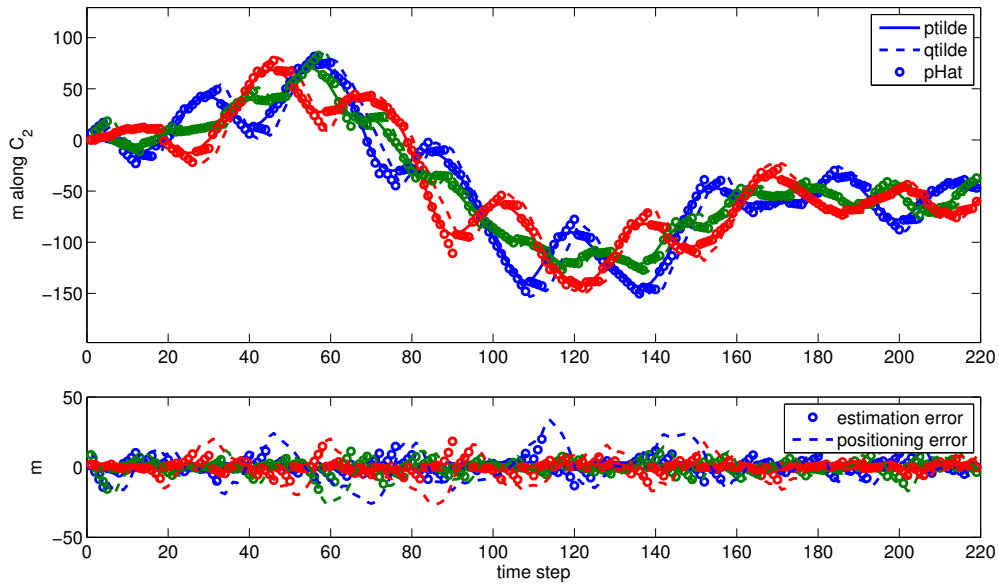


Figure 6-2: Time series of experiment run, Loners. RMS estimation error was 4.19 m, and the RMS positioning error was 10.15 m.

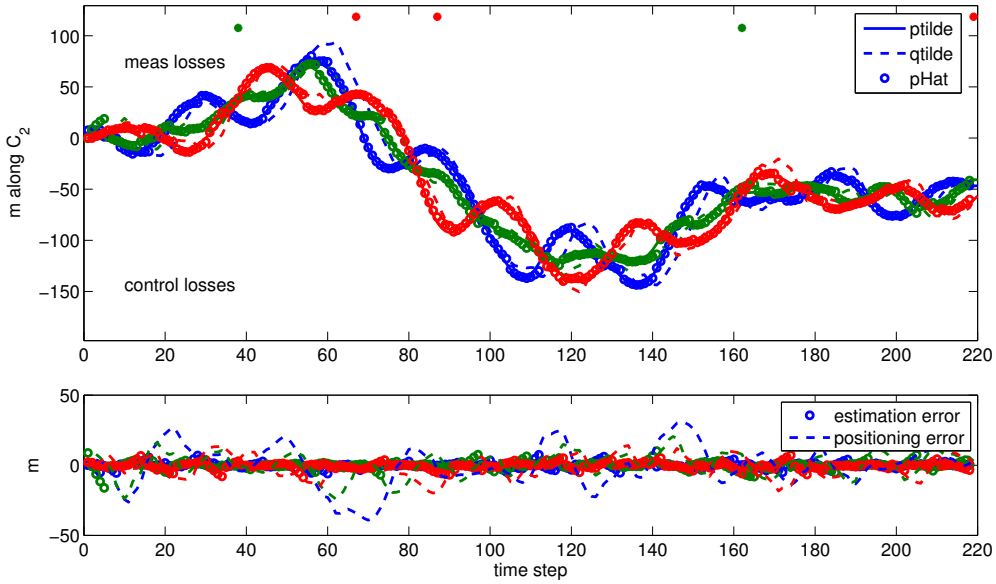


Figure 6-3: Time series of experiment run, MX schedule, 9-level quantization. RMS estimation error was 2.49 m, and the RMS positioning error was 10.33 m.

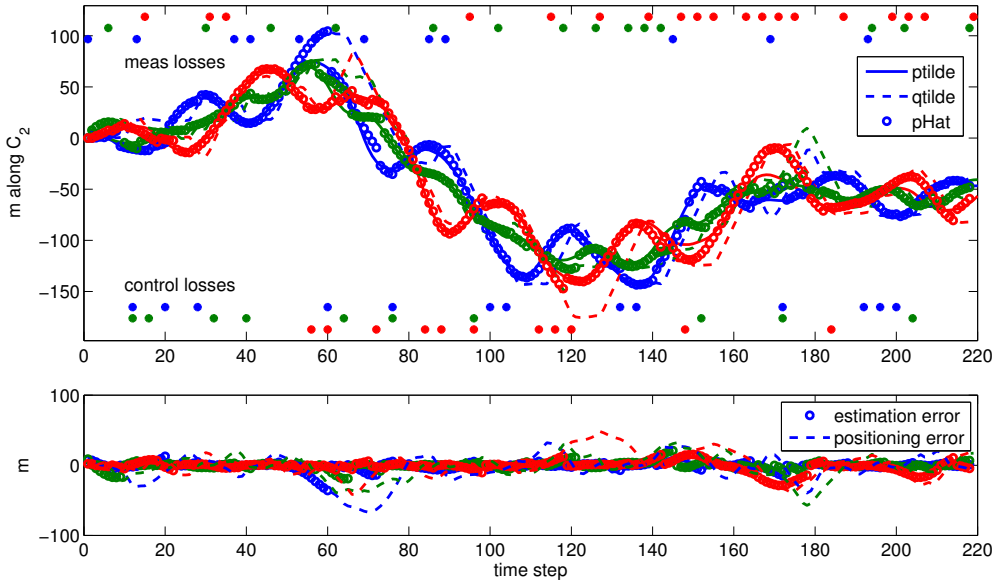


Figure 6-4: Time series of experiment run, MX schedule, 3-level quantization, with 20% additional packet loss added in software. RMS estimation error was 6.37 m, and the RMS positioning error was 18.59 m.

Comparison to simulations

We compare experimental results to simulations using the same perturbation time series, and the same exact noise and packet loss sequences. The only difference is that the vehicles are simulated using a kinematic model. **Figure 6-5** shows a scatter plot of these results, with positioning error on the x-axis, and estimation error on the y-axis.

Experimental results roughly match the simulation results. Experimental performance is consistently worse than simulations by a small amount, especially in positioning error, because our autonomous kayaks are non-ideal actuators. This is an expected source of error, especially since the experiments were conducted over a relatively small domain compared to the maneuverability of the vehicles. This causes the station-keeping behavior and turning dynamics to affect positioning. A surprising result is that the positioning inaccuracy also affects estimation of the frontal points. This was not expected because of the ideal gradient slices used in generating the hybrid measurements, which usually mean that positioning does not affect estimation performance. However, the estimation of the vehicle position does affect estimation of the feature. The vehicle positioning errors are non-Gaussian, and the KF might not have been tuned optimally for estimation of vehicle position. We note that the brown trial includes 20% extra simulated packet loss, on top of the actual packet losses. In this case, the estimator was not tuned properly for this scenario, as the control priors (when no ACKs arrived) were generated using a control packet loss probability of 5%. Comparison with the simulation results for 20% packet loss in **Figure 6-6** shows that the inaccurate packet loss estimate resulted in estimation and positioning errors that are larger by approximately a factor of two.

6.2.2 JLS-PPC MX and IL vs. independent vehicles

We show ensemble simulation results, again with scatter plots with positioning error on the x-axis, and estimation error on the y-axis. The simulation ensembles have 25 trials each and were run with a few packet loss rates and control quantization levels.

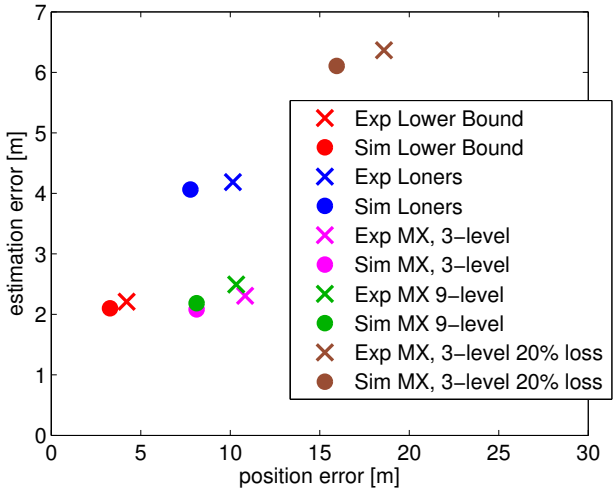


Figure 6-5: Scatter plot of experiments and experiments with the exact same perturbations, noise and packet loss sequences. The dominant natural period of the chained-mass is 18 steps. Experiment performance is slightly worse than simulations with similar settings, due to the physical vehicles being non-ideal actuators.

For the Loners and Lower Bound, we show each individual trial with a marker. For the JLS-PPC controllers, the large outlined dots show the mean of 25 trials, and the thin curves show one-standard-deviation ellipses.

We have focused the majority of our simulation studies on the three-vehicle case as it captures the essential elements of the system, but is faster to simulate and more clear to analyze than larger fleet sizes. A representative example with scalar field measurement noise variance of $V_\phi = 1$, vehicle measurement noise of $V_q = 1$, and vehicle process noise of $W_q = 0.1$ is shown in **Figure 6-6**.

Over a large range of sensor and process noise values, we have observed that the JLS-PPC controller with the MX schedule consistently outperforms the same controller using the IL schedule. JLS-PPC with MX with packet losses of roughly 20% or less are superior to vehicles acting independently (the Loners controller), and when packet loss is very low, these methods approach the performance of the scheduled Lower Bound. Notably, the JLS-PPC methods have a clear advantage in estimation, consistently exhibiting approximately 1.75 times lower estimation error than Loners when packet losses are low. The estimation error for Loners is 5.10, while it is 2.93 for the MX schedule with 3-level quantization and 10% packet loss.

The JLS-PPC methods do not perform as strongly in positioning, as they must deal with communication delays on top of packet loss. This explains why the JLS-PPC controllers with no packet loss and no quantization match the estimation error of Lower Bound, but are worse at positioning. With coarse quantization, positioning is hurt further. Quantization effects are discussed further in **Section 6.2.4**.

As a more exaggerated example of the differences between control methods, we show results when the system dynamics are extra fast in **Figure 6-7**. For these simulations, the scalar field measurement noise variance was $V_\phi = 0.1$, vehicle measurement noise was $V_q = 0.1$, and vehicle process noise was $W_q = 0.1$. The natural period of the chained-mass is 12 steps, which gives three cycles of the MX schedule, and just two cycles of the IL schedule. With the IL controller right at its Nyquist rate, its performance suffers considerably. The improvement of the MX controller over the Loners is more pronounced as well. The mean estimation for Loners is 6.85, while the mean estimation for the MX with 3-level quantization and 10% packet loss is 2.41—almost three times lower. This leads us to a design observation: as the dynamics of the feature to be tracked speed up, the benefits of the JLS-PPC controller with MX schedule over Loners become more important.

The trend of MX outperforming IL is also exaggerated as the feature dynamics become faster. We suspect the dominant factor here is that the IL schedule is longer than the MX schedule, which outweighs the finer quantization (for a given packet size) of the IL schedule. This point corroborates the main result of the target pursuit experiments of **Chapter 3**—that cycle time is more important than quantization when high tracking and pursuit bandwidth is desired.

6.2.3 Scalability

The tractability of a centralized control system is very important, as the system size and computational requirements increase with fleet size. Onboard the vehicles, very minimal computation is needed; small embedded processors will have no trouble running the PPC buffer. The large computations happen at the centralized fusion center, which runs the jump estimator, lossy KF, and controller. The dominant computa-

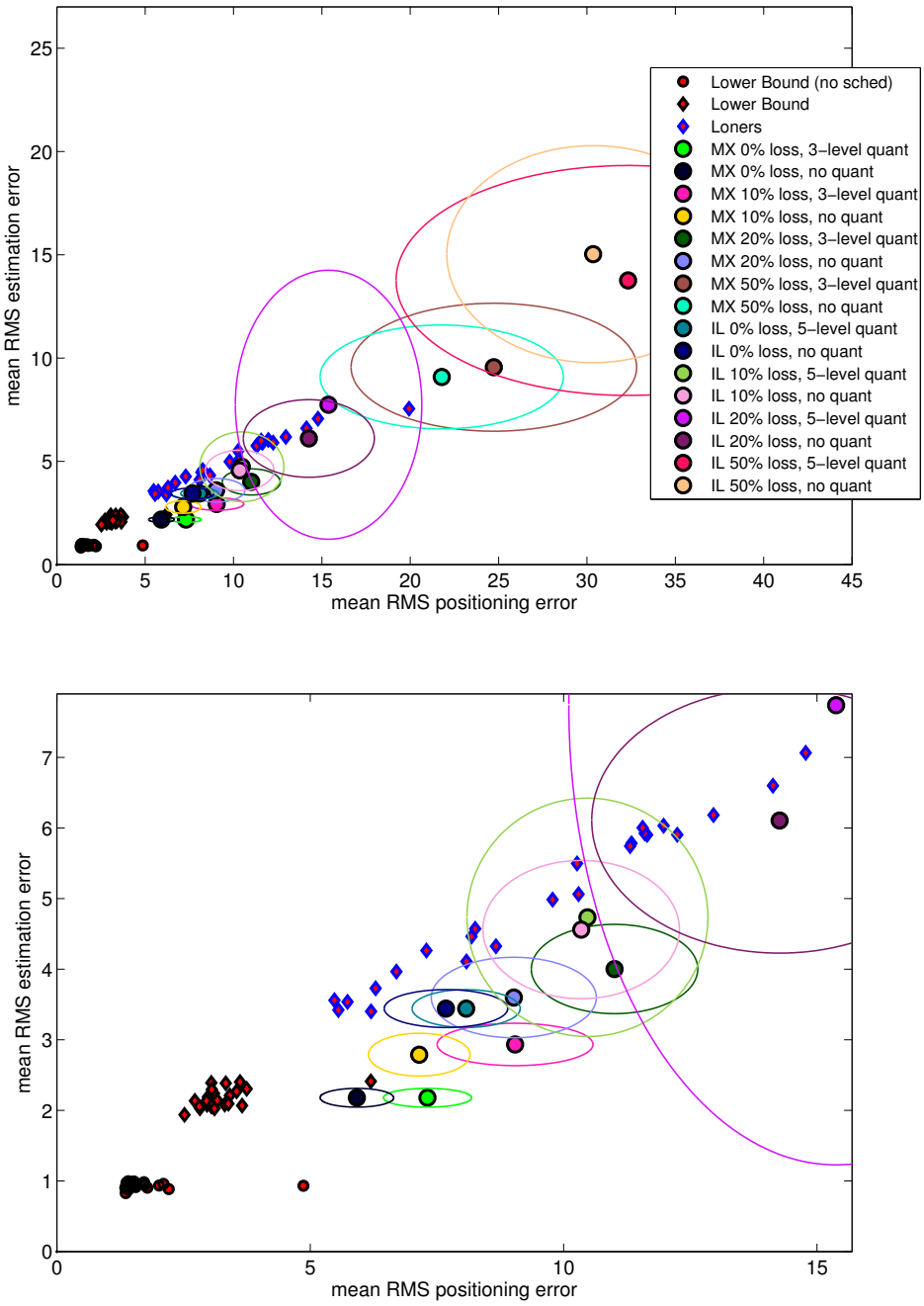


Figure 6-6: Scatter plot for $N_v = 3$. The dominant natural period of the chained-mass is 18 steps. The upper plot is zoomed-out and includes 50% packet loss results; the lower plot is a zoomed-in version of the upper plot. For the Loners and Lower Bound, we show each individual trial with a marker. For the JLS-PPC controllers, the large outlined dots show the mean of 25 trials, and the thin curves show one-standard-deviation ellipses. The main result is that the JLS-PPC controller with the MX schedule gives a factor of 1.75 improvement in estimation error over Loners when packet loss is 10%.

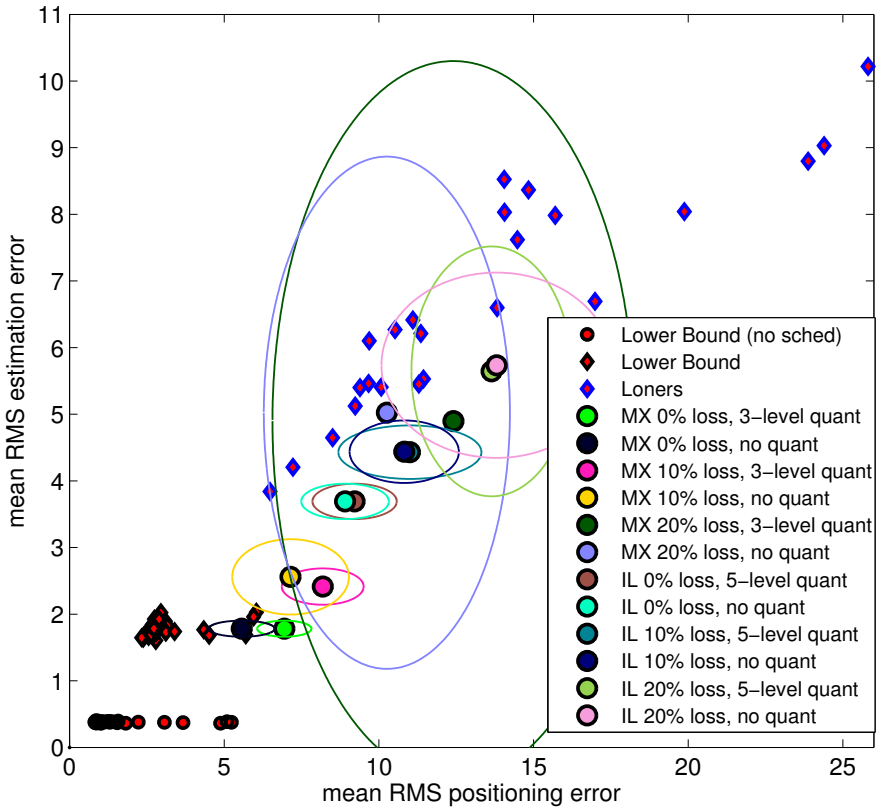


Figure 6-7: Scatter plot for $N_v = 3$, with fast perturbations. The dominant natural period of the chained-mass is 12 steps. For the Loners and Lower Bound, we show each individual trial with a marker. For the JLS-PPC controllers, the large outlined dots show the mean of 25 trials, and the thin curves show one-standard-deviation ellipses. The MX controller shows larger improvements over Loners with this fast system compared to the slower system results in **Figure 6-6**.

tional expense is running the MPC optimization, a constrained quadratic program. As noted, we solve the MPC problem using CVX and Gurobi on a standard desktop computer. This approach is easily tractable for 10-vehicle fleets, with computation times of roughly 0.5 s.

We note that more advanced solvers exist that take advantage of the special structure of MPC-style problems, as well as generate compiled code for specific problem instances. These methods could be used to effectively implement JLS-PPC if even larger fleets or low-power computers are used.

Figure 6-8 gives simulation results for $N_v = 10$, using chained mass dynamics with a dominant natural period of 60 steps (three times the length of the IL schedule of 20 steps). These results are for $V_\phi = 1$, $V_q = 1$, and $W_q = 1$. The main result is that computations were tractable, and similar trends in performance are observed compared to the $N_v = 3$ results. The mean estimation error is 15.73 for Loners, and 9.91 for MX with 3-level quantization and 5% packet loss. This is slightly less than the two-time improvement observed frequently in $N_v = 3$ trials. For the MX schedule with 3-level quantization, packets include 990 commands times levels. With the same packet size, the IL control commands consist of 15 levels. The difference in performance between no quantization and 15-level quantization is negligible, so the infinite quantization IL results are omitted from the plot. The 50% packet loss IL results are also omitted, as performance is poor. Notably, the IL schedule with *no* packet loss has much higher estimation error than the MX schedule with 20% packet loss, and performs worse in estimation and positioning compared to Loners. The differences in schedule lengths between MX and IL are exaggerated as fleet sizes grow, easily overcoming the quantization benefits of IL—making MX the obvious schedule choice.

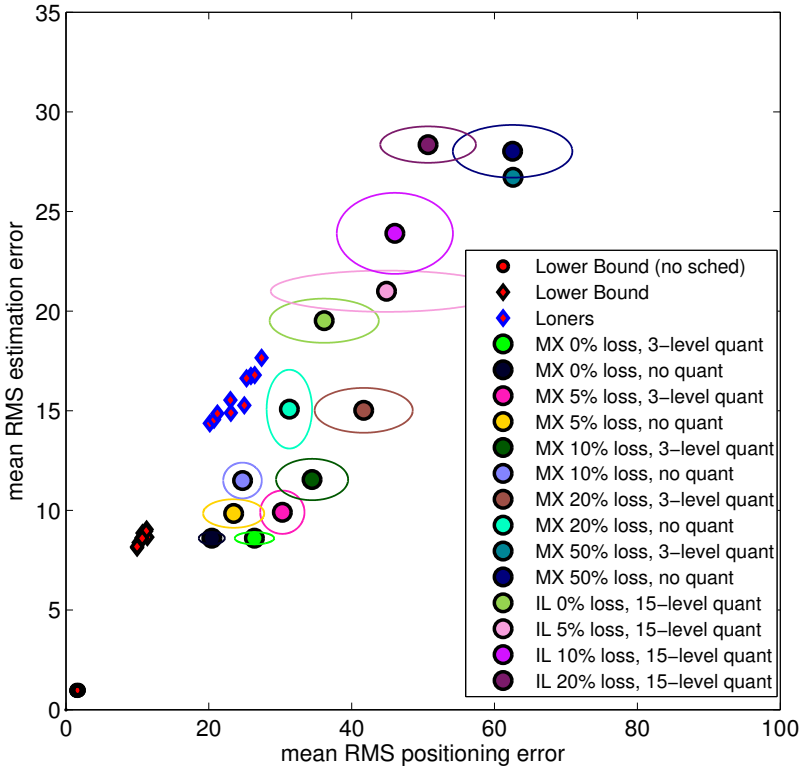


Figure 6-8: Scatter plot for $N_v = 10$. The dominant natural period of the chained-mass is 60 steps. For the JLS-PPC controllers, the large outlined dots show the mean of 10 trials, and the thin curves show one-standard-deviation ellipses. Similar trends are observed as with the three-vehicle systems, although the differences between the MX and IL schedules are exaggerated due to more vehicles and longer schedules.

6.2.4 Tradeoff between control command quantization and control packet loss

The use of buffering, and also the MX schedule, place an emphasis on quantization of control commands, as many commands are fit into one control packet. While there are many complex tradeoffs and settings involved in a wireless communication link, a fundamental tradeoff exists between the data rate and the ability of the receiver to decode information. Consider a packet-based network with fixed packet lengths in time. For given channel conditions, as the size of the packet in bits increases, modulation and/or error correction coding must adjust to increase throughput, which decreases reliability. With the control framework of this chapter, we study fixed numbers of control commands per packet, so the coarseness of quantization of each command affects the reliability of packet reception. The JLS-PPC framework does not model the choices involved in this tradeoff directly, however the simulation capability can be used indirectly as a design tool for optimizing the quantization versus packet loss tradeoff.

As an example, **Figure 6-9** shows a contour plot of positioning error over a two-dimensional domain of packet loss vs. number of quantization levels, for MX schedules. Each point on the grid is the average of 50 simulations using that combination of quantization and packet loss. To emphasize the shape, log-scale axes are used. We note that positioning error is affected by quantization much more strongly than estimation error when ideal gradients are used (estimation error is largely a function of packet loss alone).[1]

The difference in performance as quantization increases from 2 levels to 3 levels is dramatic, since three levels adds zero as an option. From 3 to 5 levels, the difference is less dramatic; a trend that continues as quantization becomes finer. We note that with the examples of this chapter, quantization is probably less of a factor than in other settings since we are looking at systems with fast dynamics, and the vehicle speed constraints are set up to be roughly equal to the max speed of the perturbations.

[1] With nonlinear fields, positioning error will affect estimation, and if measurement quantization is considered, a similar tradeoff can be studied for the vehicle-to-estimator communication link.

With this scenario, the movement of the perturbations is often quite fast, making the 3-level quantization scheme effective. With slower perturbations, there likely will be scenarios where the naive quantization scheme we use (rounding of the optimal non-quantized solution) could perform badly (large limit cycles). In such a scenario, techniques like dynamic quantization (*e.g.* [10,11,163]), or possibly the optimal closed-form quantized MPC discussed in **Section 5.9.3**, could perform better, since they can give a PWM (or delta-sigma modulator) type command sequence that isn't biased.

The positioning error contour plot is a useful tool for system design. A system designer can produce a tradeoff curve in the 2D packet loss vs. quantization space for a given communication system setup and channel condition. Factors that may go into creating such a curve are discussed below, however a reasonable assumption is the curve will be monotonically increasing from left to right (as the number of quantization levels goes up, the bit rate must increase, and packet loss will increase). The "quarter-bowl" shape of the contour plot suggests that for many such curves, there will be an interesting design tradeoff as the minimum in positioning error will occur somewhere in the middle of the plot.

A basic example of such a curve is plotted on top of the upper contour plot in **Figure 6-9**. This curve describes an exponential relationship between the number of quantization levels and the packet loss probability of the controller link: $nLevels = 1.5e^{0.1\bar{\alpha}}$. With this relationship, packet loss is low ($\approx 3\%$) when coarse 2-level quantization is used, and packet loss increases as the quantization resolution becomes finer and the data packets must be sent using higher rate transmission. The bottom plot of **Figure 6-9** gives the RMS positioning error along this curve, showing the minimum positioning error occurs when five control quantization levels are used.

As mentioned above, there are many complicated choices that go into creating a"representative" packet loss vs. quantization curve, and full treatment of the choices is beyond the scope of this thesis. High-level decisions such as communication system frequency and bandwidth, and the slot length, packet length in time, etc. for the communication schedule and control system must be considered. Once these are decided, the digital modulation scheme must be chosen (*e.g.* FSK, or PSK/QAM

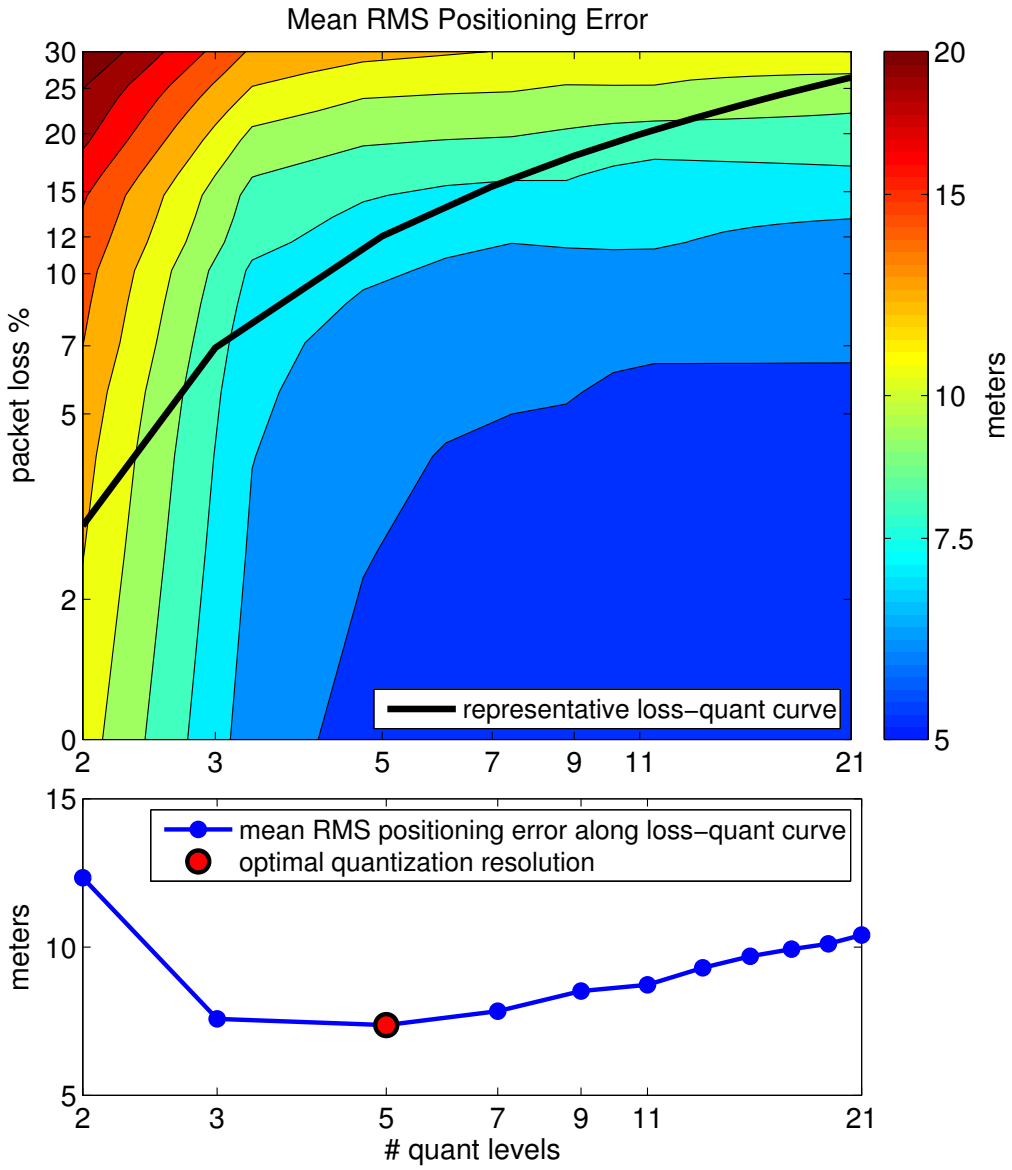


Figure 6-9: **Top:** Positioning error contour plot for JLS-PPC with MX schedule, piggybacked ACKs. The black curve shows the tradeoff between quantization and packet loss for a representative communication system. **Bottom:** Slice along the quantization vs. packet loss curve, showing the number of quantization levels that results in the lowest positioning error for this particular tradeoff curve.

with varying constellation sizes). Forward error correction coding (FEC) gain can also be adjusted. Channel conditions (noise, delay spread, path loss, etc.) vary, and must be modeled. The SNR (or post-equalizer SNR for phase-coherent methods) is then related to a bit error rate based on the specific modulation scheme chosen. The bit error rate can then be related to a packet error rate depending on the FEC coding gain.

Related to the number quantization levels, there are also issues about how to efficiently package the control trajectories into a packet. Combining the bit representations of individually-quantized commands is the simplest, although vector quantization of trajectories is also possible. The issue with vector quantization is that the codebook size is $nLevels^{N_p}$, so codebooks can grow very large when many levels and/or long trajectories are used. However, when commands are quantized individually using few numbers of bits, there are large jumps in the numbers of levels that can be represented, *e.g.* four levels with two bits, 8 levels with three bits, 16 levels with four bits, etc. Packets for serialized command trajectories will have large discrete jumps in size as the number of levels increases—quantization with 5 levels will result in the same size packet as with 8 levels. A middle-ground option is to quantize shorter sections of commands via vector quantization, and then combine those chunks to form the full trajectory. The choice of trajectory representation and compression involves engineering tradeoffs and will affect the shape of the loss-quantization curve.

All of these settings are related, and one could think of various optimizations that could be studied. Similar optimizations have been done in the literature, but usually related to multihop or reliable transport protocols (*e.g.* work by Casari & Zorzi [42], and Basagni *et al.* [16]). There is room for future work to study communication optimization in the context of real-time control.

6.2.5 Piggybacked ACKs vs. no ACKs

A major contribution of the JLS framework is the ability to handle delayed and lossy control acknowledgments. In this section, we consider the benefit of using this capability with piggybacked control ACKs versus no control ACKs. This study is

similar to that conducted with the simple SISO double integrator in **Section 5.8**, but for the multi-vehicle pursuit system.

Control ACKs help generate accurate KF priors, and thus make the biggest difference in estimation when the priors are important—large vehicle measurement noise and low vehicle process noise. We present results with such noise settings, intended to emphasize the differences between ACK schemes. The vehicle measurement noise was set to $V_q = 9$, vehicle process noise to $W_q = 0.01$, and the scalar field measurement noise to $V_\phi = 0.01$.

For these simulations, the two ACK methods were compared with identical noise and packet loss sequences for each trial. Four numbers of quantization levels were simulated. Due to the large noise values and high packet losses considered, there were often trials where the estimator diverged; we count the trials where this occurred, and discard the numerical results for a trial if either of the methods diverged. For each packet loss, quantization, and instance, the no ACK methods had either the same or slightly more (by up to 2 more) outliers. We then computed the mean estimation and position errors from the outlier-filtered results, averaged across trials and quantization levels. These results are shown in **Figure 6-10**.

The results are dramatic, showing clear benefits for the piggybacked ACK scheme. At 0% packet loss, the performance is the same, as expected. The difference in estimation and positioning errors becomes more dramatic as packet loss increases to 20%, with approximately a constant absolute difference in performance at higher packet losses. With 20% packet loss, the piggybacked ACK controller achieves a 60% reduction in positioning error and a 70% reduction in estimation error compared to the no ACK controller.

6.3 Summary

The main result of the chapter is that the MX schedule consistently outperforms vehicles acting independently ("Loners") when packet loss remains reasonable (*e.g.* below 30%). Additionally, results showed that the MX schedule is superior to the

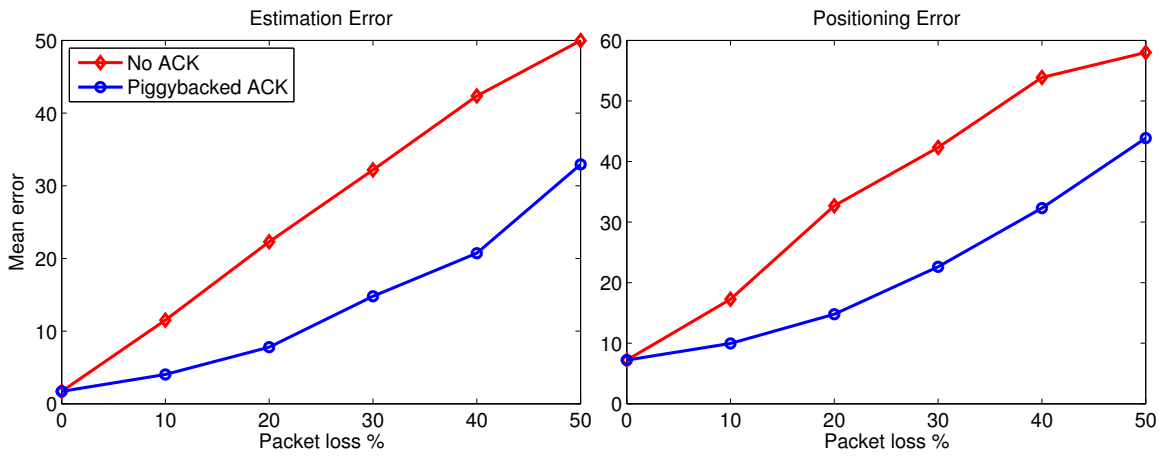


Figure 6-10: Comparison of performance with piggybacked (lossy, delayed) ACKs vs. no ACKs, averaged over 50 trials and four different quantization levels.

IL schedule in both estimation and positioning performance. We demonstrated that the JLS framework is easily scalable to 10-vehicle fleets, with the above two trends remaining true. As the dynamics of the feature to be tracked become faster, the benefits of the JLS-PPC controller over Loners become more dramatic.

We conducted field experiments with autonomous surface vehicles and real acoustic communications using TDMA schedules and subject to quantization and packet loss. Results from these trials indicated that the simulation framework is effective at illuminating trends between controllers, with the absolute performance of the physical system diminished slightly due to vehicles behaving as non-ideal actuators. This supports our assertion that the simulation framework is a useful tool for system design. We discussed one example where the simulation framework provides a method to indirectly study and optimize communication system tradeoffs, specifically the quantization versus packet loss relationship. Finally, we presented simulation results that clearly demonstrate the usefulness of piggybacked ACKs. This capability is a key aspect of the JLS-PPC framework, and is especially helpful in scenarios where the KF priors play a large role in estimation performance.

Chapter 7

Conclusion

We reiterate the primary outcomes of each chapter, and summarize how they relate to the overall goal of enabling highly dynamic multi-vehicle collaborative control in the ocean. We conclude with a discussion of promising directions for future work.

7.1 Summary of Contributions

Chapters 2-4 consider the first major area of work: the setup and solution of new underwater pursuit problems using closed-loop control.

Chapter 2 set the stage by introducing some fundamental aspects of acoustic communications, and presented a robust approach to a major acoustic networking problem: multicast routing and power control. We formulated the robust counterpart of the multicommodity mixed-integer linear programming (MILP) model from Haugland and Yuan [112], and derived scaled power levels that account for uncertainty. Our formulation, Robust MET, provides a tractable means for designing efficient geographic routing subject to power uncertainty, a capability which is especially useful in power-constrained marine robotic networks that rely on unreliable acomms. We showed that with proper scaling of input power levels, a deterministic MILP formulation may be used to find the optimal robust solution; this type of optimization can be solved tractably. The approach developed relies on centralized global optimization, which is more amenable to mobile networks than iterative algorithms

where convergence may be difficult while nodes are constantly moving.

Chapter 3 presented field experiments in target pursuit using acoustic range measurements and lossy, rate-limited acoustic communications. We performed tracking of a moving target using two vehicles and acoustic range measurements. The vehicles collaborate in order to jointly estimate the target's position, and move to stay in formation relative to it. Real-time communication is an integral aspect of the estimation and control loop. We presented in detail results comparing the tracking performance of three different communication configurations, at operating speeds near 1.5 m/s. A "lower bound" case with RF wireless communication, a four-second cycle and no quantization had a tracking bandwidth of ≈ 0.5 rad/s. When using full-sized modem packets with negligible quantization and a 23-second cycle time, the tracking bandwidth was ≈ 0.065 rad/s. With 13-bit mini-packets, we employed logarithmic quantization to achieve a cycle time of 12 seconds and a tracking bandwidth of ≈ 0.13 rad/s. In particular, the mini packet vs. full-sized packet experiments demonstrated that for tracking highly dynamic targets it is beneficial to trade-off quantization for low cycle time. These outcomes showed definitively that aggressive dynamic control of multi-agent systems underwater is tractable today.

In **Chapter 4**, we articulated an integrated framework for dynamically sampling the ocean using a group of communicating mobile agents. Our new concept is that locally linear behavior of an ocean process admits strong estimation and control techniques on short time scales; this allows multiple cooperating vehicles to decompose spatial and temporal variations in the field, and track a dynamic feature of interest. The stochastic dynamical model supporting our controller design is created via a projection from an ocean forecast ensemble into succinct vehicle coordinates, and this is the main innovation of our work. In studies with three example datasets, we demonstrated that control and estimation designs resulting from the identified models are successful and provide benefits over vehicles operating independently.

Chapter 5 presented the second primary area of work—detailed development of a networked control framework for centralized estimation and control with stochastic packet-loss, deterministic delays, and a pre-designed schedule for transmissions

of measurements, controls, and control acknowledgments. The ability to consider all of these communication constraints is a new capability in the literature, and our framework helps manage the complexity of system description and notation. We outlined the notation and assumptions, described candidate scheduling paradigms, presented the jump linear description of the system, and finally developed specific estimation and control algorithms. The elements that make up the JLS-PPC controller are modular, consisting of a jump variable estimator, a lossy state estimator, a forward propagation step, MPC optimization, and the PPC buffer. Our jump estimator algorithm uses a novel approach for using delayed and lossy control packet acknowledgments, made possible by the use of the PPC buffer and the JLS system description. An illustrative example with a simple system demonstrated that the benefits of using lossy, delayed ACKs can be dramatic. We described the algorithms that our JLS-PPC implementation is based on, and note that future improvements to any individual block may be used with the rest of the existing framework.

In **Chapter 6**, we presented simulation and field experiments demonstrating the JLS-PPC controller in pursuit missions. The main result of the chapter is that the JLS-PPC controller using the MX schedule consistently outperformed vehicles acting independently ("Loners") when packet loss remains reasonable (*e.g.* below 30%). We showed that the MX schedule is superior to the IL schedule in both estimation and positioning performance, and demonstrated that the JLS framework is easily scalable to 10-vehicle fleets, with the above two trends remaining true. We conducted field experiments with autonomous surface vehicles and real acoustic communications using TDMA schedules and quantized controls, and subject to packet loss. Results from these trials indicated that the simulation framework is effective at illuminating trends between controllers, with the absolute performance of the physical system diminished slightly due to vehicles behaving as non-ideal actuators. This supports our assertion that the simulation framework is a useful tool for system design. We discussed one example where the simulation framework provides a method to indirectly study and optimize communication system tradeoffs, specifically the quantization versus packet loss relationship. Finally, we presented simulation results that clearly demonstrate

the usefulness of piggybacked ACKs. This capability is a key aspect of the JLS-PPC framework, and is especially helpful in scenarios where the KF priors play a large role in estimation performance.

7.2 Future Work

7.2.1 Oceanographic Pursuit and similar applications

We believe that multi-vehicle networked control applications will play a large role in future understanding of the ocean. Target pursuit can help inform the study of marine animals, and oceanographic pursuit can improve our understanding of complex and dynamic ocean features. Many areas of future work exist, and we discuss a few here.

Local linearization of ocean model simulations is a critical element of the Oceanographic Pursuit procedure, that we address first with the frontal points concept and then with subspace identification methods. Neither of these is easy. Frontal point generation has an implicit ergodicity assumption, and requires domain expertise as implied in the constraints $\mathbb{P}$. At the same time, subspace identification, even for systems which are known to be linear, is difficult, and we have as well restrictions on record length, plus gridding effects. Many specializations and improvements can be made to the preliminary methods we have described. We firmly believe that linear stochastic models, however, are key to cogent analysis and design procedures when multiple vehicles are to operate with realistic navigation and communication limits. The LTI model allows for classical and scalable multivariable estimation and control, as well as rigorous contemporary approaches for lossy communications. To this end, the JLS-PPC control framework with TDMA scheduling and delays can be applied to oceanographic pursuit datasets to provide more realistic results in the same style as those presented at the end of **Chapter 4**.

Of course, full-scale experiments pursuing an actual feature in the ocean are a future goal. Next, we discuss some immediate areas that will make our Oceanographic Pursuit approach more amenable to field implementation. Initial detection

of a feature, and the associated initialization of the closed-loop pursuit system, is an important problem. Related are issues of fault detection (*e.g.* did the feature break up? or did the vehicles lose it?) and recovery. Important application-specific extensions consider the measurements themselves. Realistic field measurements from vehicles could range from time-intensive scalar measurements, to histories of fast measurements, to "metameasurements" such as a local survey to determine the gradient. Representing these measurements for communications requires efficient source coding, and incorporating such measurements into the networked control framework may require novel measurement models. Depending on the nature and update rate of the sensor, it may also be beneficial to blend the "Loners" technique with the centralized collaborative approach. Hierarchical control techniques could allow a level of reactivity locally onboard the vehicle, with high-level adjustments provided by the networked controller.

Our Oceanographic Pursuit framework and projection algorithm could be used to study the relationships between spatiotemporal resolution, number of vehicles, vehicle speeds, feature dynamics, sensor characteristics, etc. For example, if highly accurate sensors are available, it may not be necessary to set up communication at all for the purposes of oceanographic pursuit. On the other hand, vehicles with less expensive, lower-quality sensors can be deployed in larger numbers and will likely benefit from a coupled model and hence collaboration. More broadly, heterogeneous sensor and vehicle networks are increasingly likely in practice. The integrated framework we have developed should support strong trade-off studies along these lines, and further insight could be gathered from a more comprehensive and integrated approach to projection and system identification. For example, depending on the strength of coupling between sites and communication degradation with distance, it may be beneficial to group vehicles into subnetworks, where collaboration occurs at a high rate between nearby vehicles, and inter-subgroup communication occurs at longer intervals.

7.2.2 JLS-PPC and co-design of communication and control

One straightforward direction for future work is to use the JLS framework for weighing the various tradeoffs in communication design, specifically when communications are to be used for closed-loop control. As results in this thesis have shown, the tradeoff between delays, reliability, and quantization must be considered differently for feedback control than for time-averaged throughput. Simulations using the JLS framework can help determine optimal tradeoffs, similar to the quantization vs. packet loss example given in the end of **Chapter 6**. Practically, investigation of potential "sweet spots" within these tradeoff spaces can inform the design of control-specific modems and protocols to better meet the needs of networked control systems. Increased flexibility in packet types (amount of data, length in time, error correction coding, etc.) will allow more precise optimization of transport protocols aimed for the latency and reliability considerations of closed-loop control. One example of such a capability is the flexible minipacket capability of the MicroModem 2. Additionally, hardware changes could help alleviate some of the fundamental constraints typically encountered with acoustic communications and vehicles. For example, a non-interfering dual-frequency communication system could dramatically improve scheduling constraints, and a dedicated ACK capability, where very small reliable ACK packets are sent over an acoustic channel that does not interfere with the larger control and measurement packets, could help reduce estimation uncertainty caused by the the dual effect.

As mentioned, the modular nature of the JLS framework is intended to allow improvements and extensions to specific components, many of which we discuss in **Section 5.9**. Of these, development of robust MPC techniques may be the most important. Connecting the uncertainty of the state estimator with robust control techniques could offer large improvements in performance, especially if state constraints (*e.g.* islands or proximity to a ship) are considered. Robust MPC is also a promising direction for developing analytical results. Similarly, certain applications may require more detailed consideration of nonlinear vehicle dynamics and maneuvering constraints. Work towards tractable representations and approximations of

such constraints within the MPC framework would improve performance in these scenarios.

More broadly, we expect that convex optimization can provide a unifying framework for integrated communication and control design and analysis. The JLS-PPC controller is built around receding horizon optimization, so extensions in this direction are natural. The JLS-PPC controller accepts scheduling and delays as inputs; an "outer" optimization ideally would *design* schedules that give best performance given system dynamics and vehicle positions. Large deployments in terms of both spatial extent as well as vehicle fleet size will necessitate multihop networking, where routing is an additional aspect to be considered on top of scheduling. The approach for robust MET given in **Chapter 2** is based on convex optimization, and this formulation is just one possible step towards integrated co-design of control and communications, the "holy grail" of multi-vehicle networked control. Related to the co-design problem is the desire for analytical results for controller performance as a function of communication parameters, which may help connect the communication design with controller performance metrics. Bringing together practical constructive techniques in networked control and communication design with fundamental bounds on performance such as disturbance rejection will help provide a theoretical justification for operators to trust networked vehicle systems in the field.

Bibliography

[1] DSPComm Acoustic Modems. http://www.dspcomm.com/products.html.

[2] EvoLogics R Series Acoustic Modems. http://www.evologics.de/en/products/acoustics/index.html.

[3] LinkQuest SoundLink Underwater Acoustic Modems. http://www.link-quest.com/html/intro1.htm.

[4] Sonardyne Positioning and Navigation. http://www.sonardyne.co.uk/Products/PositioningNavigation/index.html.

[5] Teledyne Benthos Acoustic Modem Product Comparison. http://www.benthos.com/acoustic-telesonar-modem-product-comparison.asp.

[6] Hydro International Acoustic Modem Product Survey, 2007. http://www.hydro-international.com/productsurvey/id7-Acoustic_Modems.html.

[7] I.F. Akyildiz, D. Pompili, and T. Melodia. Underwater acoustic sensor networks: research challenges. *Ad Hoc Networks*, 3(3):257–279, 2005.

[8] J. Alves and M. Schneider. A communication infrastructure for cooperative operation of a fleet of heterogeneous autonomous marine vehicles: Concepts and developments within the grex project. In *OCEANS 2009–EUROPE*, pages 1 –8, May 2009.

[9] A. Atamturk and M. Zhang. Two-stage robust network flow and design under demand uncertainty. *Operations Research*, 55(4):662–673, 2007.

[10] S. Azuma and T. Sugie. Optimal dynamic quantizers for discrete-valued input control. *Automatica*, 44(2):396–406, 2008.

[11] S. Azuma and T. Sugie. Synthesis of optimal dynamic quantizers for discrete-valued input control. *IEEE Trans. Automatic Control*, 53(9):2064–2075, 2008.

[12] L. Badia, M. Mastrogiovanni, C. Petrioli, S. Stefanakos, and M. Zorzi. An optimization framework for joint sensor deployment, link scheduling and routing in underwater sensor networks. In *Proceedings of the 1st ACM international workshop on Underwater networks*, WUWNet '06, pages 56–63, New York, NY, USA, 2006. ACM.

[13] A. Bahr, J.J. Leonard, and M.F. Fallon. Cooperative localization for autonomous underwater vehicles. *International J. Robotics Research*, 28(6):714, 2009.

[14] J. Baillieul and P.J. Antsaklis. Control and communication challenges in networked real-time systems. *Proc. of the IEEE*, 95(1):9–28, 2007.

[15] Y. Bar-Shalom and E. Tse. Dual effect, certainty equivalence, and separation in stochastic control. *IEEE Trans. Automatic Control*, 19(5):494–500, 1974.

[16] S. Basagni, C. Petrioli, R. Petroccia, and M. Stojanovic. Optimized packet size selection in underwater wireless sensor network communications. *IEEE J. Oceanic Engineering*, 37(3):321–337, 2012.

[17] T. Bean, J. Canning, G. Beidler, M. O'Rourke, and DB Edwards. Designing and implementing collaborative behaviors for autonomous underwater vehicles. In *Proc. International Symposium on Unmanned Untethered SubmersibleTechnology*, 2007.

[18] J.G. Bellingham and K. Rajan. Robotics in remote and hostile environments. *Science*, 318(5853):1098–1102, 2007.

[19] A. Bemporad. Predictive control of teleoperated constrained systems with unbounded communication delays. In *Proc. IEEE Conf. Decision and Control*, pages 2133–2138. IEEE, 1998.

[20] A. Bemporad and S. Di Cairano. Model-predictive control of discrete hybrid stochastic automata. *IEEE Trans. Automatic Control*, 56(6):1307–1321, 2011.

[21] A. Bemporad and M. Morari. Robust model predictive control: A survey. *Robustness in identification and control*, pages 207–226, 1999.

[22] Alberto Bemporad, Manfred Morari, Vivek Dua, and E. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[23] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.

[24] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, pages 769–805, 1998.

[25] M.R. Benjamin, J.J. Leonard, H. Schmidt, and P.M. Newman. An overview of MOOS-IvP and a brief users guide to the IvP helm autonomy software. *Massachusetts Institute of Technology, MIT CSAIL, Tech. Rep. TR-2009-28-07*, 2009.

[26] D.P. Bertsekas. *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995.

[27] D. Bertsimas and D. B. Brown. Constrained stochastic LQC: a tractable approach. *IEEE Trans. Automatic Control*, 52(10):1826–1841, 2007.

[28] D. Bertsimas, D. Pachamanova, and M. Sim. Robust linear optimization under general norms. *Operations Research Letters*, 32(6):510–516, 2004.

[29] B. Bingham. Predicting the navigation performance of underwater vehicles. In *Proc. IEEE/RSJ Intelligent Robots and Systems, 2009*, 2009.

[30] B. S. Bingham, J.M. Walls, and R.M. Eustice. Development of a flexible command and control software architecture for marine robotic applications. *Marine Technology Society Journal*, 45(3):25–36, May/June 2011.

[31] R. Blind and F Allgower. Estimating the fates of the control packets for networked control systems with loss of control and measurement packets. In *IEEE Conf. Decision and Control*, pages 2687–2692. IEEE, 2009.

[32] A.D. Bowen, M.V. Jakuba, N.E. Farr, J. Ware, C. Taylor, D. Gomez-Ibanez, C.R. Machado, and C. Pontbriand. An un-tethered rov for routine access and intervention in the deep sea. In *IEEE/MTS OCEANS*, pages 1–7, Sept 2013.

[33] L. Brignone, J. Alves, and J. Opderbecke. GREX sea trials: first experiences in multiple underwater vehicle coordination based on acoustic communication. In *Proc. MTS/IEEE OCEANS*, 2009.

[34] A. Caiti, V. Calabro, G. Dini, A.L. Duca, and A. Munafo. AUVs as mobile nodes in acoustic communication networks: Field experience at the UAN10 experiments. In *Proc. MTS/IEEE OCEANS*, 2011.

[35] A. Caiti, P. Felisberto, T. Husoy, S.M. Jesus, I. Karasalo, R. Massimelli, T.A. Reinen, and A. Silva. UAN underwater acoustic network. In *MTS/IEEE OCEANS*, 2011.

[36] A. Caiti and A. Munafo. Adaptive cooperative algorithms for AUV networks. In *IEEE Conf. Communications Workshops*, 2010.

[37] R. Camilli, C.M. Reddy, D.R. Yoerger, B.A.S. Van Mooy, M.V. Jakuba, J.C. Kinsey, C.P. McIntyre, S.P. Sylva, and J.V. Maloney. Tracking hydrocarbon plume transport and biodegradation at Deepwater Horizon. *Science*, 330(6001):201, 2010.

[38] J.R. Canning, M.J. Anderson, D.B. Edwards, M. O'Rourke, T.A. Bean, J.L. Pentzer, and D.L. Odell. A low bandwidth acoustic communication strategy for supporting collaborative behaviors in a fleet of autonomous underwater vehicles. *US Navy J. Underwater Acoustics*, 2009.

[39] M. Cannon, Q. Cheng, B. Kouvaritakis, and S.V. Raković. Stochastic tube mpc with state estimation. *Automatica*, 48(3):536–541, 2012.

[40] M. Cannon, B. Kouvaritakis, and D. Ng. Probabilistic tubes in linear stochastic model predictive control. *Systems & Control Letters*, 58(10):747–753, 2009.

[41] V.M. Canuto and M.S. Dubovikov. Modeling mesoscale eddies. *Ocean Modelling*, 8(1):1–30, 2005.

[42] P. Casari and M. Zorzi. Protocol design issues in underwater acoustic networks. *Computer Communications*, 2011.

[43] F. Cazenave, Y. Zhang, E. McPhee-Shaw, J.G. Bellingham, and T.P. Stanton. High-resolution surveys of internal tidal waves in monterey bay, california, using an autonomous underwater vehicle. *Limnology and Oceanography: Methods*, 9:571–581, 2011.

[44] C.S. Chang, T. Ho, and M. Effros. On robust network coding subgraph construction under uncertainty. In *Asilomar Conf. on Signals, Systems and Computers*, 2008.

[45] B. Chen and D. Pompili. Team formation and steering algorithms for underwater gliders using acoustic communications. *Computer Communications*, 35(9):1017–1028, 2012.

[46] M. Cheung, J. Leighton, and F. Hover. Multi-armed bandit formulation for autonomous mobile acoustic relay adaptive positioning. In *Proc. IEEE International Conf. on Robotics and Automation*, pages 4165–4170, 2013.

[47] M. Chitre, S. Shahabudeen, L. Freitag, and M. Stojanovic. Recent advances in underwater acoustic communications & networking. In *IEEE/MTS OCEANS*, 2008.

[48] A. Chiuso, N. Laurenti, L. Schenato, and A. Zanella. LQG cheap control over SNR-limited lossy channels with delay. In *IEEE Conf. Decision and Control*, 2013.

[49] F. Chuang, C. Danielson, and F. Borrelli. Optimality of certainty equivalence in expected value problems for uncertain linear systems. *arXiv preprint arXiv:1404.0413*, 2014.

[50] T.H. Chung, J.W. Burdick, and R.M. Murray. A decentralized motion coordination strategy for dynamic target tracking. In *Proc. IEEE International Conference on Robotics and Automation*, 2006.

[51] C.M. Clark, C. Forney, E. Manii, D. Shinzaki, C. Gage, M. Farris, C.G. Lowe, and M. Moline. Tracking and following a tagged leopard shark with an autonomous underwater vehicle. *Journal of Field Robotics*, 30 (3):309–322, 2013.

[52] O.L.V. Costa, M.D. Fragoso, and R.P. Marques. *Discrete-time Markov jump linear systems*. Springer, 2006.

[53] N.A. Cruz, B.M. Ferreira, A.C. Matos, C. Petrioli, R. Petroccia, and D. Spaccini. Implementation of an underwater acoustic network using multiple heterogeneous vehicles. In *Proc. MTS/IEEE Oceans*, pages 1–10. IEEE, 2012.

[54] N.A. Cruz and A.C. Matos. Adaptive sampling of thermoclines with autonomous underwater vehicles. In *OCEANS 2010*, pages 1–6. IEEE, 2010.

[55] R.L. Cruz and A.V. Santhanam. Optimal routing, link scheduling and power control in multihop wireless networks. In *IEEE INFOCOM*, 2003.

[56] R. Cui, S.G. Shuzhi, B. Voon Ee How, and Y. Sang Choo. Leader–follower formation control of underactuated autonomous underwater vehicles. *Ocean Engineering*, 37(17):1491–1502, 2010.

[57] M.A. Cunguara, O. e Silva, T.A. Mendes, and P.B.R. Pedreiras. On the application of block transmissions for improving control over lossy networks. In *IEEE Conf. Emerging Technologies & Factory Automation*, 2013.

[58] J. Curcio, J. Leonard, J. Vaganay, A. Patrikalakis, A. Bahr, D. Battle, H. Schmidt, and M. Grund. Experiments in moving baseline navigation using autonomous surface craft. In *Proc. MTS/IEEE OCEANS*, 2005.

[59] T.B. Curtin, J.G. Bellingham, J. Catipovic, and D. Webb. Autonomous oceanographic sampling networks. *Oceanography*, 6(3):86–94, 1993.

[60] M.A. Dahleh, P.G. Voulgaris, and L.S. Valavani. Optimal and robust controllers for periodic and multirate systems. *IEEE Trans. Automatic Control*, 37:90–99, 1992.

[61] J. Das, F. Py, T. Maughan, T. O'Reilly, M. Messié, J. Ryan, G.S. Sukhatme, and K. Rajan. Coordinated sampling of dynamic oceanographic features with underwater vehicles and drifters. *Int. J. Robotics Research*, 31(5):626–646, 2012.

[62] E. D'Asaro, C. Lee, L. Rainville, R. Harcourt, and L. Thomas. Enhanced turbulence and energy dissipation at ocean fronts. *Science*, 332(6027):318–322, 2011.

[63] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proc. International Conf. Robotics and Automation (ICRA)*, volume 2, pages 1322–1328. IEEE, 1999.

[64] R. Diamant, G.N. Shirazi, and L. Lampe. Robust spatial reuse scheduling in underwater acoustic communication networks. In *IEEE Conf. Vehicular Technology*, 2011.

[65] M. Doniec, C. Detweiler, I. Vasilescu, and D. Rus. Using optical communication for remote underwater robot operation. In *IEEE/RSJ Conf. Intelligent Robots and Systems*, 2010.

[66] M.C.F. Donkers, W.P.M.H. Heemels, N. Van De Wouw, and L. Hetel. Stability analysis of networked control systems using a switched linear systems approach. *IEEE Trans. Automatic Control*, 56(9):2101–2115, 2011.

[67] JC Doyle. Guaranteed margins for LQG regulators. *IEEE Trans. Automatic Control*, 23(4):756–757, August 1978.

[68] M. Dunbabin and L. Marques. Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics & Automation Magazine*, 19(1):24–39, 2012.

[69] D.P. Eickstedt, M.R. Benjamin, H. Schmidt, and J.J. Leonard. Adaptive control of heterogeneous marine sensor platforms in an autonomous sensor network. In *IEEE/RSJ Conf. Intelligent Robots and Systems*, 2006.

[70] D.P. Eickstedt, M.R. Benjamin, H. Schmidt, and J.J. Leonard. Adaptive tracking of underwater targets with autonomous sensor networks. *US Navy J. Underwater Acoustics*, 56:465–495, 2006.

[71] M. Epstein, L. Shi, and R.M. Murray. An estimation algorithm for a class of networked control systems using UDP-like communication schemes. In *IEEE Conf. Decision and Control*, 2006.

[72] M. Epstein, L. Shi, and R.M. Murray. Estimation schemes for networked control systems using UDP-like communication. In *IEEE Conf. Decision and Control*, 2007.

[73] H. Eustice, R.and Singh and L. Whitcomb. Synchronous-clock one-way-travel-time acoustic navigation for underwater vehicles. *J. Field Robotics, Special Issue on State of the Art in Maritime Autonomous Surface and Underwater Vehicles*, 28(1):121–136, 2011.

[74] G. Evensen. The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics*, 53(4):343–367, 2003.

[75] M.F. Fallon, G. Papadopoulos, and J.J. Leonard. A measurement distribution framework for cooperative navigation using multiple AUVs. In *IEEE Conf. Robotics and Automation*, 2010.

[76] N Farr, A Bowen, J Ware, C Pontbriand, and M Tivey. An integrated, underwater optical/acoustic communications system. In *IEEE/MTS OCEANS*, 2010.

[77] N. Farr, A. Chave, L. Freitag, J. Preisig, S. White, D. Yoerger, and P. Titterton. Optical modem technology for seafloor observatories. In *MTS/IEEE OCEANS*, 2005.

[78] J.A. Farrell, S. Pang, and W. Li. Chemical plume tracing via an autonomous underwater vehicle. *IEEE J. Oceanic Engineering*, 30(2):428–442, 2005.

[79] R. Ferrari. A frontal challenge for climate models. *Science*, 332(6027):316–317, 2011.

[80] M. Fi and L. Xie. The sector bound approach to quantized feedback control. *IEEE Trans. Automatic Control*, 50(11):1698 – 1711, nov. 2005.

[81] R Findeisen and P Varutti. Stabilizing nonlinear predictive control over nondeterministic communication networks. In *Nonlinear model predictive control*, pages 167–179. Springer, 2009.

[82] A.R. Fioravanti, A. Gonçalves, and J.C. Geromel. Discrete-time H-Infinity output feedback for markov jump systems with uncertain transition probabilities. *International Journal of Robust and Nonlinear Control*, 2012.

[83] E. Fiorelli, N.E. Leonard, P. Bhatta, D.A. Paley, R. Bachmayer, and D.M. Fratantoni. Multi-AUV control and adaptive sampling in Monterey Bay. *IEEE J. Oceanic Engineering*, 31:935 –948, 2006.

[84] T.I. Fossen. *Guidance and control of ocean vehicles*. John Wiley & Sons Inc, 1994.

[85] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball. The WHOI Micro-Modem: An acoustic communications and navigation system for multiple platforms. In *Proc. MTS/IEEE OCEANS*, 2005.

[86] L. Freitag, P. Koski, A. Morozov, S. Singh, and J. Partan. Acoustic communications and navigation under arctic ice. In *Oceans, 2012*, 2012.

[87] L. Freitag and S. Singh. Performance of micro-modem PSK signaling under variable conditions during the 2008 RACE and SPACE experiments. In *MTS/IEEE OCEANS*, 2009.

[88] M. Fu and C.E. de Souza. State estimation for linear discrete-time systems using quantized measurements. *Automatica*, 45(12):2937–2945, 2009.

[89] A.S. Gadre, D.K. Maczka, D. Spinello, B.R. McCarter, D.J. Stilwell, W. Neu, M.J. Roan, and J.B. Hennage. Cooperative localization of an acoustic source using towed hydrophone arrays. In *Proc. IEEE/OES Autonomous Underwater Vehicles*, pages 1 –8, oct. 2008.

[90] N.C. Gallagher Jr and G.L. Wise. A theoretical analysis of the properties of median filters. *IEEE Trans. Acoustics, Speech and Signal Processing*, 29(6):1136–1141, 1981.

[91] E. Gallimore, J. Partan, I. Vaughn, S. Singh, J. Shusta, and L. Freitag. The WHOI micromodem-2: A scalable system for acoustic communications and networking. In *MTS/IEEE OCEANS*, 2010.

[92] A. Gangopadhyay and A.R. Robinson. Feature-oriented regional modeling of oceanic fronts. *Dynamics of Atmospheres and Oceans*, 36(1):201–232, 2002.

[93] E. Garone, B. Sinopoli, and A. Casavola. On the effect of packet acknowledgment on the stability and performance of networked control systems. In *Modelling, Estimation and Control of Networked Complex Systems*, pages 191–206. Springer, 2009.

[94] E. Garone, B. Sinopoli, and A. Casavola. LQG control over lossy TCP-like networks with probabilistic packet acknowledgements. *International Journal of Systems, Control and Communications*, 2(1):55–81, 2010.

[95] G. Gawarkiewicz, S. Jan, P.F.J. Lermusiaux, J.L. McClean, L. Centurioni, K. Taylor, B. Cornuelle, T.F. Duda, J. Wang, and J.Y. Yiing. Circulation and intrusions northeast of Taiwan: Chasing and predicting uncertainty in the Cold Dome. *Oceanography*, 24:110–121, 2011.

[96] J.C. Geromel, A.P.C. Gonçalves, and A.R. Fioravanti. Dynamic output feedback control of discrete-time markov jump linear systems through linear matrix inequalities. *SIAM Journal on Control and Optimization*, 48(2):573–593, 2009.

[97] E. Gilbertson, B. Reed, J. Leighton, M. Cheung, and F. Hover. Experiments in dynamic control of autonomous marine vehicles using acoustic modems. In *Proc. IEEE International Conf. on Robotics and Automation*, 2013.

[98] M.A. Godin, Y. Zhang, J.P. Ryan, T.T. Hoover, and J.G. Bellingham. Phytoplankton bloom patch center localization by the Tethys Autonomous Underwater Vehicle. In *OCEANS '11*, pages 1–6. MTS/IEEE, 2011.

[99] A.P.C. Gonçalves, A.R. Fioravanti, and J.C. Geromel. Filtering for discrete-time markov jump systems with network transmitted mode. In *IEEE Conf. Decision and Control*, pages 924–929. IEEE, 2010.

[100] R. Gondhalekar and C.N. Jones. MPC of constrained discrete-time linear periodic systemsa framework for asynchronous control: Strong feasibility, stability and optimality via periodic invariance. *Automatica*, 47(2):326–333, 2011.

[101] G.C. Goodwin, H. Kong, G. Mirzaeva, and M.M. Seron. Robust model predictive control: reflections and opportunities. *Journal of Control and Decision*, 1(2):115–148, 2014.

[102] G.C. Goodwin and D.E. Quevedo. Finite alphabet control and estimation. *Int. J. Control Automation and Systems*, 1:412–430, 2003.

[103] P.J. Goulart and E.C. Kerrigan. Output feedback receding horizon control of constrained systems. *International Journal of Control*, 80(1):8–20, 2007.

[104] P.J. Goulart, E.C. Kerrigan, and J.M. Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42(4):523–533, 2006.

[105] R. Graham, F. Py, J. Das, D. Lucas, T. Maughan, and K. Rajan. Exploring space-time tradeoffs in autonomous sampling for marine robotics. In *Experimental Robotics*, volume 88, pages 819–839. Springer Tracts in Advanced Robotics, 2013.

[106] S.M. Griffies. *Fundamentals of ocean climate models*, volume 518. Princeton University Press Princeton, 2004.

[107] M. Grund, L. Freitag, J. Preisig, and K. Ball. The plusnet underwater communications system: Acoustic telemetry for undersea surveillance. In *IEEE/MTS OCEANS*, pages 1 –5, sept. 2006.

[108] V. Gupta, B. Hassibi, and R.M. Murray. Optimal LQG control across packet-dropping links. *Systems & Control Letters*, 56(6):439–446, 2007.

[109] V. Gupta, B. Sinopoli, S. Adlakha, A. Goldsmith, and R. Murray. Receding horizon networked control. In *Proc. Allerton Conf. Commun., Control Comput*, 2006.

[110] P.J. Haley Jr, P.F.J. Lermusiaux, A.R. Robinson, W.G. Leslie, O. Logoutov, G. Cossarini, X.S. Liang, P. Moreno, S.R. Ramp, and J.D. Doyle. Forecasting and reanalysis in the Monterey Bay/California current region for the Autonomous Ocean Sampling Network-II experiment. *Deep Sea Research Part II: Topical Studies in Oceanography*, 56(3):127–148, 2009.

[111] S.R. Hanna and R. Yang. Evaluations of mesoscale models' simulations of near-surface winds, temperature gradients, and mixing depths. *Journal of Applied Meteorology*, 40(6):1095–1104, 2001.

[112] D. Haugland and D. Yuan. Compact integer programming models for power-optimal trees in ad hoc wireless networks. *Wireless Network Design*, pages 219–246, 2011.

[113] R. Headrick and L. Freitag. Growth of underwater communication technology in the U.S. Navy. *IEEE Communications Magazine*, 47(1):80 –82, 2009.

[114] K.D. Heaney, G. Gawarkiewicz, T.F. Duda, and P.F.J. Lermusiaux. Nonlinear optimization of autonomous undersea vehicle sampling strategies for oceanographic data-assimilation. *Journal of Field Robotics*, 24(6):437–448, 2007.

[115] W.P.M.H. Heemels, A.R. Teel, N. van de Wouw, and D. Nesic. Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance. *IEEE Trans. Automatic Control*, 55(8):1781 –1796, 2010.

[116] J. Heidemann, M. Stojanovic, and M. Zorzi. Underwater sensor networks: applications, advances and challenges. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 370(1958):158–175, 2012.

[117] J.P. Hespanha, P. Naghshtabrizi, and Yonggang Xu. A survey of recent results in networked control systems. *Proc. IEEE*, 95(1):138 –162, jan. 2007.

[118] P. Hokayem, E Cinquemani, D. Chatterjee, F. Ramponi, and J. Lygeros. Stochastic receding horizon control with output feedback and bounded controls. *Automatica*, 48(1):77–88, 2012.

[119] G.A. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G.S. Sukhatme, M. Stojanovic, H. Singh, and F. Hover. Underwater data collection using robotic sensor networks. *IEEE J. Selected Areas in Communications*, 30(5):899–911, 2012.

[120] P.L. Houtekamer and H.L. Mitchell. Data assimilation using an ensemble kalman filter technique. *Monthly Weather Review*, 126(3):796–811, 1998.

[121] D. Hrovat, S. Di Cairano, H.E. Tseng, and I.V. Kolmanovsky. The development of model predictive control in automotive industry: A survey. In *IEEE Int. Conf. Control Applications*, pages 295–302. IEEE, 2012.

[122] T. Huck and G.K. Vallis. Linear stability analysis of the three-dimensional thermally-driven ocean circulation: Application to interdecadal oscillations. *Tellus A*, 53(4):526–545, 2001.

[123] O.C. Imer, S. Yüksel, and T. Başar. Optimal control of LTI systems over unreliable communication links. *Automatica*, 42(9):1429–1439, 2006.

[124] T. Inanc, S.C. Shadden, and J.E. Marsden. Optimal trajectory generation in ocean flows. In *IEEE American Controls Conference*, 2005.

[125] K. Jain, J. Padhye, V.N. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. *Wireless networks*, 11(4):471–487, 2005.

[126] Z. Jin, V. Gupta, and R.M. Murray. State estimation over packet dropping networks using multiple description coding. *Automatica*, 42(9):1441–1452, 2006.

[127] J.M. Jornet, M. Stojanovic, and M. Zorzi. On joint frequency and power allocation in a cross-layer protocol for underwater acoustic networks. *IEEE J. Oceanic Engineering*, 35(4):936 –947, oct. 2010.

[128] S.J. Julier and J.K. Uhlmann. New extension of the Kalman filter to nonlinear systems. In *AeroSense'97*, pages 182–193. International Society for Optics and Photonics, 1997.

[129] J. Kalwa. The GREX-Project: Coordination and control of cooperating heterogeneous unmanned systems in uncertain environments. In *MTS/IEEE OCEANS*, 2009.

[130] M-A Khalighi, C. Gabriel, T. Hamza, S. Bourennane, P. Leon, and V. Rigaud. Underwater wireless optical communication; recent advances and remaining challenges. In *Int. Conf. Transparent Optical Networks*, 2014.

[131] D.B. Kilfoyle and A.B. Baggeroer. The state of the art in underwater acoustic telemetry. *IEEE J. Oceanic Engineering*, 25(1):4 –27, jan 2000.

[132] J Kinsey, R Eustice, and L Whitcomb. A survey of underwater vehicle navigation: recent advances and new challenges. In *IFAC Conference of Manoeuvering and Control of Marine Craft*, Lisbon, Portugal, 2006. Invited paper.

[133] R. Kleeman. Measuring dynamical prediction utility using relative entropy. *Journal of the Atmospheric Sciences*, 59(13):2057–2072, 2002.

[134] M. Kögel, R. Blind, F. Allgöwer, and R. Findeisen. Optimal and optimal-linear control over lossy, distributed networks. In *IFAC World Congress*, 2011.

[135] K. Kredo, P. Djukic, and P. Mohapatra. STUMP: Exploiting Position Diversity in the Staggered TDMA Underwater MAC Protocol. In *IEEE INFOCOM*, 2009.

[136] N.H. Kussat, C.D. Chadwell, and R. Zimmerman. Absolute positioning of an autonomous underwater vehicle using GPS and acoustic measurements. *IEEE J. Oceanic Engineering*, 30(1):153–164, 2005.

[137] P. Lacovara. High-bandwidth underwater communications. *Marine Technology Society Journal*, 42(1):93–102, 2008.

[138] S. Lall and G. Dullerud. An LMI solution to the robust synthesis problem for multi-rate sampled-data systems. *Automatica*, 37(12):1909–1922, 2001.

[139] T.D. Larsen, N.A. Andersen, O. Ravn, and N.K. Poulsen. Incorporation of time delayed measurements in a discrete-time kalman filter. In *IEEE Conf. Decision and Control*, 1998.

[140] J.H. Lee, M.S. Gelormino, and M. Morari. Model predictive control of multi-rate sampled-data systems: a state-space approach. *International Journal of Control*, 55(1):153–191, 1992.

[141] C.E. Leith. Theoretical skill of Monte Carlo forecasts. *Monthly Weather Review*, 102(6):409–418, 1974.

[142] N.E. Leonard, D.A. Paley, R.E. Davis, D.M. Fratantoni, F. Lekien, and F. Zhang. Coordinated control of an underwater glider fleet in an adaptive ocean sampling field experiment in monterey bay. *J. Field Robotics*, 27(6):718–740, 2010.

[143] N.E. Leonard, D.A. Paley, F. Lekien, R. Sepulchre, D.M. Fratantoni, and R.E. Davis. Collective motion, sensor networks, and ocean sampling. *Proc. IEEE*, 95(1):48 –74, 2007.

[144] P.F.J. Lermusiaux. Uncertainty estimation and prediction for interdisciplinary ocean dynamics. *Journal of Computational Physics*, 217(1):176–199, 2006.

[145] E. Liao, G. Hollinger, J. Djugash, and S. Singh. Preliminary results in tracking mobile targets using range sensors from multiple robots. *Distributed Autonomous Robotic Systems 7*, pages 125–134, 2006.

[146] B. Liu, Y. Xia, M.S. Mahmoud, H. Wu, and S. Cui. New predictive control scheme for networked control systems. *Circuits, Systems, and Signal Processing*, 31(3):945–960, 2012.

[147] L. Ljung. *System Identification: Theory for the User*. Prentice Hall PTR, 1999.

[148] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193 – 228, 1998.

[149] T. Lolla, M.P. Ueckermann, K. Yigit, P.J. Haley, and P.F.J. Lermusiaux. Path planning in time dependent flow fields using level set methods. In *IEEE Int. Conf. on Robotics and Automation*, pages 166–173. IEEE, 2012.

[150] D.E. Lucani, M. Stojanovic, and M. Médard. Random linear network coding for time division duplexing: when to stop talking and start listening. In *INFOCOM 2009, IEEE*, pages 1800–1808. IEEE, 2009.

[151] R. Lum, A. Balasuriya, and H. Schmidt. Multistatic processing and tracking of underwater target using autonomous underwater vehicles. In *Underwater Acoustics Measurement Conference*, 2009.

[152] D.K. Maczka, D. Spinello, D.J. Stilwell, A.S. Gadre, and W.L. Neu. Coordinated tracking of an acoustic signal by a team of autonomous underwater vehicles. In *Proc. AUVSI's Unmanned Systems North America*, 2009.

[153] G. Marani, S.K. Choi, and J. Yuh. Underwater autonomous manipulation for intervention missions auvs. *Ocean Engineering*, 36(1):15–23, 2009.

[154] N.C. Martins and M.A. Dahleh. Feedback control in the presence of noisy channels: Bode-like fundamental limitations of performance. *IEEE Trans. Autonomatic Control*, 53(7):1604–1615, 2008.

[155] D.Q. Mayne, S.V. Raković, R. Findeisen, and F. Allgöwer. Robust output feedback model predictive control of constrained linear systems. *Automatica*, 42(7):1217–1222, 2006.

[156] D.Q. Mayne, S.V. Raković, R. Findeisen, and F. Allgöwer. Robust output feedback model predictive control of constrained linear systems: Time varying case. *Automatica*, 45(9):2082–2087, 2009.

[157] D.Q. Mayne, M.M. Seron, and S.V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, 2005.

[158] J. Mayne, D.Q.and Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[159] C. McGann, F. Py, K. Rajan, H. Thomas, R. Henthorn, and R. McEwen. T-REX: A model-based architecture for AUV control. In *Workshop on Planning and Plan Execution for Real-World Systems*, 2007.

[160] M. Michini, M.A. Hsieh, E. Forgoston, and I.B. Schwartz. Robotic tracking of coherent structures in flows. *IEEE Trans. Robotics*, PP(99):1–11, 2014.

[161] P. Millán, C. Jurado, I.and Vivas, and F.R. Rubio. Networked predictive control of systems with data dropouts. In *IEEE Conf. Decision and Control*, 2008.

[162] P.H. Milne. *Underwater acoustic positioning systems*. Gulf Publishing Co., Houston, TX, 1983.

[163] Y. Minami, S. Azuma, and T. Sugie. Optimal decentralized dynamic quantizers for discrete-valued input control: A closed form solution and experimental evaluation. In *American Control Conference*, 2009.

[164] A.M. Moore, H.G. Arango, E. Di Lorenzo, B.D. Cornuelle, A.J. Miller, and D.J. Neilson. A comprehensive ocean prediction and analysis system based on the tangent linear and adjoint of a regional ocean model. *Ocean Modeling*, 7(1):227–258, 2004.

[165] B.C. Moore. Singular value analysis of linear systems. In *IEEE Conf. Decision and Control*, volume 17, pages 66–73. IEEE, 1978.

[166] Y. Mostofi, T.H. Chung, R.M. Murray, and J.W. Burdick. Communication and sensing trade-offs in decentralized mobile sensor networks: a cross-layer design approach. In *Int. Symposium on Information Processing in Sensor Networks*, 2005.

[167] C. Murphy, J.M. Walls, T. Schneider, R.M. Eustice, M. Stojanovic, and H. Singh. Capture: A communications architecture for progressive transmission via underwater relays with eavesdropping. *IEEE J. Oceanic Engineering*, 39(1):120–130, Jan 2014.

[168] M. Nagahara and D.E. Quevedo. Sparse representations for packetized predictive networked control. In *Proc. IFAC World Congress*, pages 84–89, 2011.

[169] R.R. Negenborn and J.M. Maestre. Distributed model predictive control: An overview and roadmap of future research opportunities. *IEEE Control Systems Magazine*, 34(4):87–97, 2014.

[170] J.W. Nicholson and A.J. Healey. The present state of autonomous underwater vehicle (AUV) applications and technologies. *Marine Technology Society Journal*, 42(1):44–51, 2008.

[171] P. Nicopolitidis, G.I. Papadimitriou, and A.S. Pomportsis. Adaptive data broadcasting in underwater wireless networks. *IEEE J. Oceanic Engineering*, 35(3):623 –634, july 2010.

[172] J. Nilsson, B. Bernhardsson, and B. Wittenmark. Stochastic analysis and control of real-time systems with random time delays. *Automatica*, 34(1):57–64, 1998.

[173] M.J. Olascoaga and G. Haller. Forecasting sudden changes in environmental pollution patterns. *Proceedings of the National Academy of Sciences*, 109(13):4738–4743, 2012.

[174] F. Oldewurtel, A. Parisio, C.N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and M. Morari. Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings*, 45:15–27, 2012.

[175] J Ostergaard and D.E. Quevedo. Multiple descriptions for packetized predictive control over erasure channels. In *IEEE Conf. Control and Automation*, 2011.

[176] D.A. Paley, F. Zhang, and N.E. Leonard. Cooperative control for ocean sampling: The glider coordinated control system. *IEEE Trans. Control Systems Technology*, 16(4):735–744, 2008.

[177] T.N. Palmer. Predicting uncertainty in forecasts of weather and climate. *Reports on Progress in Physics*, 63(2):71, 2000.

[178] K. Papadaki and V. Friderikos. Robust scheduling in spatial reuse TDMA wireless networks. *IEEE Trans. Wireless Communications*, 7(12):4767 –4771, december 2008.

[179] J. Partan, J. Kurose, and B.N. Levine. A survey of practical issues in underwater networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 11(4):23–33, 2007.

[180] J. Partan, J. Kurose, B.N. Levine, and J. Preisig. Spatial reuse in underwater acoustic networks using RTS/CTS MAC protocols. *University of Massachusetts Dept of Computer Science, UM-CS-2010-045*, 2010.

[181] S. Petillo and H. Schmidt. Exploiting adaptive and collaborative auv autonomy for detection and characterization of internal waves. *IEEE J. Oceanic Engineering*, 39(1):150–164, 2014.

[182] S. Petillo, H. Schmidt, and A. Balasuriya. Constructing a distributed AUV network for underwater plume-tracking operations. *International Journal of Distributed Sensor Networks*, 2012, 2011.

[183] G. Pin, M. Filippo, and T. Parisini. Networked MPC for constrained linear systems: a recursive feasibility approach. In *IEEE Conf. Decision and Control*, 2009.

[184] G. Pin and T. Parisini. Networked predictive control of uncertain constrained nonlinear systems: Recursive feasibility and input-to-state stability analysis. *IEEE Trans. Automatic Control*, 56(1):72–87, 2011.

[185] P. Qarabaqi and M. Stojanovic. Adaptive power control for underwater acoustic communications. In *MTS/IEEE OCEANS*, pages 1 –7, june 2011.

[186] P. Qarabaqi and M. Stojanovic. Modeling the large scale transmission loss in underwater acoustic channels. In *Allerton Conf. on Communication, Control, and Computing*, 2011.

[187] S.J. Qin and T.A. Badgwell. A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764, 2003.

[188] T.Q.S. Quek, H. Shin, and M.Z. Win. Robust wireless relay networks: Slow power allocation with guaranteed qos. *IEEE J. Selected Topics in Signal Processing*, 1(4):700 –713, dec. 2007.

[189] D.E. Quevedo, G.C. Goodwin, and J.A. De Dona. Finite constraint set receding horizon quadratic control. *International journal of robust and nonlinear control*, 14(4):355–377, 2004.

[190] D.E. Quevedo, G.C. Goodwin, and J.S. Welsh. Minimizing down-link traffic in networked control systems via optimal control techniques. In *IEEE Conf. Decision and Control*, volume 2, pages 1200–1205, 2003.

[191] D.E. Quevedo, C. Müller, and G.C. Goodwin. Conditions for optimality of naïve quantized finite horizon control. *International Journal of Control*, 80(5):706–720, 2007.

[192] D.E. Quevedo, J. Ostergaard, and D. Nesic. Packetized predictive control of stochastic systems over bit-rate limited channels with packet loss. *IEEE Trans. Automatic Control*, 56(12):2854 –2868, 2011.

[193] D.E. Quevedo, E.I. Silva, and G.C. Goodwin. Packetized predictive control over erasure channels. In *American Control Conference, 2007. ACC '07*, pages 1003 –1008, july 2007.

[194] R. Raheli, A. Polydoros, and C. Tzou. Per-survivor processing: A general approach to mlse in uncertain environments. *IEEE Trans. Communications*, 43(234):354–364, 1995.

[195] S.R. Ramp, R.E. Davis, N.E. Leonard, I. Shulman, Y. Chao, A.R. Robinson, J. Marsden, P.F.J. Lermusiaux, D.M. Fratantoni, and J.D. Paduan. Preparing to predict: the second Autonomous Ocean Sampling Network (AOSN-II) experiment in the Monterey Bay. *Deep Sea Research Part II: Topical Studies in Oceanography*, 56(3-5):68–86, 2009.

[196] B. Reed and F. Hover. Tracking ocean fronts with multiple vehicles and mixed communication losses. In *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2013.

[197] B. Reed and F. Hover. Oceanographic Pursuit: Networked control of multiple vehicles tracking dynamic ocean features. *Methods in Oceanography*, 10:21–43, September 2014.

[198] B. Reed, J. Leighton, M. Stojanovic, and F. Hover. Multi-vehicle dynamic pursuit using underwater acoustics. In *International Symposium on Robotics Research (ISRR)*, 2013.

[199] B. Reed, M. Stojanovic, U. Mitra, and F. Hover. Robust minimum energy wireless routing for underwater acoustic communication networks. In *Workshop on Wireless Networking & Control for Unmanned Autonomous Vehicles: Architectures, Protocols and Applications*, 2012.

[200] L.F. Richardson. *Weather prediction by numerical process*. Cambridge University Press, 2007.

[201] P. Ridao, M. Carreras, E. Hernandez, and N. Palomeras. Underwater telerobotics for collaborative research. In *Advances in Telerobotics*, pages 347–359. Springer, 2007.

[202] J.H. Rife and S.M. Rock. Design and validation of a robotic control law for observation of deep-ocean jellyfish. *IEEE Trans. Robotics*, 22(2):282 –291, april 2006.

[203] P. Rigby, O. Pizarro, and S.B. Williams. Towards Geo-Referenced AUV Navigation Through Fusion of USBL and DVL Measurements. In *MTS/IEEE OCEANS*, 2006.

[204] M. Rixen, J.W. Book, A. Carta, V. Grandi, L. Gualdesi, R. Stoner, P. Ranelli, A. Cavanna, P. Zanasca, and G. Baldasserini. Improved ocean prediction skill and reduced uncertainty in the coastal region from multi-model superensembles. *Journal of Marine Systems*, 78:S282–S289, 2009.

[205] C.L. Robinson and P.R. Kumar. Sending the most recent observation is not optimal in networked control: Linear temporal coding and towards the design of a control specific transport protocol. In *Proc. 46th IEEE Conf. Decision and Control*, pages 334 –339, dec. 2007.

[206] E.M. Royer and C.-K. Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, 6(2):46 –55, apr 1999.

[207] M Safonov and M. Athans. Gain and phase margin for multiloop LQG regulators. *IEEE Trans. Automatic Control*, 22(2):173–179, 1977.

[208] A. Sahai and S. Mitter. The necessity and sufficiency of anytime capacity for stabilization of a linear system over a noisy communication link, part i: Scalar systems. *IEEE Trans. Information Theory*, 52(8):3369 –3395, 2006.

[209] T.P. Sapsis and P.F.J. Lermusiaux. Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Physica D: Nonlinear Phenomena*, 238(23):2347–2360, 2009.

[210] T.P. Sapsis and P.F.J. Lermusiaux. Dynamical criteria for the evolution of the stochastic dimensionality in flows with uncertainty. *Physica D: Nonlinear Phenomena*, 241(1):60–76, 2012.

[211] C. Sayers. *Remote control robotics*. Springer, 1999.

[212] C.P. Sayers, R.P. Paul, L.L. Whitcomb, and D.R. Yoerger. Teleprogramming for subsea teleoperation using acoustic communication. *IEEE J. Oceanic Engineering*, 23(1):60–71, 1998.

[213] R. Scattolini. Architectures for distributed and hierarchical model predictive control–a review. *Journal of Process Control*, 19(5):723–731, 2009.

[214] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S.S. Sastry. Foundations of control and estimation over lossy networks. *Proc. IEEE*, 95(1):163–187, 2007.

[215] T. Schneider and H. Schmidt. Unified command and control for heterogeneous marine sensing networks. *Journal of Field Robotics*, 27(6):876–889, 2010.

[216] C. Scholin, S. Jensen, B. Roman, E. Massion, R. Marin, C. Preston, D. Greenfield, W. Jones, and K. Wheeler. The Environmental Sample Processor (ESP)– an autonomous robotic device for detecting microorganisms remotely using molecular probe technology. In *MTS/IEEE OCEANS*, 2006.

[217] P. Seiler and R. Sengupta. A bounded real lemma for jump systems. *IEEE Trans. Automatic Control*, 48(9):1651–1654, 2003.

[218] P. Seiler and R. Sengupta. An H-infinity approach to networked control. *IEEE Trans. Automatic Control*, 50(3):356 – 364, march 2005.

[219] M.M. Seron, J.A. De Dona, and G.C. Goodwin. Global analytical model predictive control with input constraints. In *IEEE Conf. Decision and Control*, volume 1, 2000.

[220] D. Simon. *Optimal State Estimation: Kalman, H infinity, and Nonlinear Approaches*. Wiley-Interscience, 2006.

[221] S. Singh, S.E. Webster, L. Freitag, L.L. Whitcomb, K. Ball, J. Bailey, and C. Taylor. Acoustic communication performance of the WHOI micro-modem in sea trials of the Nereus vehicle to 11,000 m depth. In *MTS/IEEE OCEANS*, 2009.

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

[223] J. Skaf and S.P. Boyd. Design of affine controllers via convex optimization. *IEEE Trans. Automatic Control*, 55(11):2476–2487, 2010.

[224] G.B. Skomal and G.W. Benz. Ultrasonic tracking of greenland sharks, somniosus microcephalus, under arctic ice. *Marine Biology*, 145(3):489–498, 2004.

[225] R.N. Smith, Y. Chao, P.P. Li, D.A. Caron, B.H. Jones, and G.S. Sukhatme. Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model. *The International Journal of Robotics Research*, 29(12):1475, 2010.

[226] J.M. Soares, A.P. Aguiar, A.M. Pascoal, and A. Martinoli. Joint ASV/AUV range-based formation control: Theory and experimental results. In *Proc. International Conf. Robotics and Automation (ICRA)*, 2013.

[227] M.J. Stanway. Delayed-state sigma point Kalman filters for underwater navigation. In *IEEE/OES Autonomous Underwater Vehicles*, pages 1 –9, sept. 2010.

[228] M. Stojanovic. On the relationship between capacity and distance in an underwater acoustic communication channel. *ACM SIGMOBILE Mobile Computing and Communications Review*, 11(4):34–43, 2007.

[229] M. Stojanovic and J. Preisig. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Communications Magazine*, 47(1):84 –89, january 2009.

[230] J. Stow, C.A.and Jolliff, D.J. McGillicuddy Jr, S.C. Doney, J. Allen, M. Friedrichs, K.A. Rose, and P. Wallhead. Skill assessment for coupled biological/physical models of marine systems. *Journal of Marine Systems*, 76(1):4–15, 2009.

[231] A. Takasu, T.and Yasuda. Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB. In *International Symposium on GPS/GNSS*, 2009.

[232] S. Tatikonda and S. Mitter. Control under communication constraints. *IEEE Trans. Autonomatic Control*, 49(7):1056 – 1068, july 2004.

[233] C.J. Thompson and D.S. Battisti. A linear stochastic dynamical model of ENSO. Part I: Model development. *Journal of Climate*, 13(15):2818–2832, 2000.

[234] K. Tsumura, H. Ishii, and H. Hoshina. Tradeoffs between quantization and packet loss in networked control of linear systems. *Automatica*, 45(12):2963–2970, 2009.

[235] M.P. Ueckermann and P.F.J. Lermusiaux. MIT 2.29 finite volume MATLAB framework documentation. *Reports in Ocean Science and Engineering*, 14, 2012.

[236] D.H. Van Hessem and O.H. Bosgra. A full solution to the constrained stochastic closed-loop mpc problem via state and innovations feedback and its receding horizon implementation. In *IEEE Conf. Decision and Control*, volume 1, pages 929–934. IEEE, 2003.

[237] P. Van Overschee and B. De Moor. *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Kluwer Academic Publishers, 1996.

[238] A.N. Vargas, E.F. Costa, and J.B.R. Val. On the control of markov jump linear systems with no mode observation: application to a dc motor device. *International Journal of Robust and Nonlinear Control*, 2012.

[239] A/N. Vargas, J.B.R. do Val, and E.F. Costa. Receding horizon control of markov jump linear systems subject to noise and unobserved state chain. In *IEEE Conf. Decision and Control*, 2004.

[240] A.N. Vargas, W/ Furloni, and J.B.R. do Val. Constrained model predictive control of jump linear systems with noise and non-observed markov state. In *American Control Conference*, 2006.

[241] P. Varutti and R. Findeisen. Event-based nmpc for networked control systems over udp-like communication channels. In *American Control Conference*, 2011.

[242] I. Vázquez-Rowe, D. Iribarren, M.T. Moreira, and G. Feijoo. Combined application of life cycle assessment and data envelopment analysis as a methodological approach for the assessment of fisheries. *The International Journal of Life Cycle Assessment*, 15(3):272–283, 2010.

[243] K. Vickery. Acoustic positioning systems: A practical overview of current systems. In *Workshop on Autonomous Underwater Vehicles*, 1998.

[244] R. Vidal, O. Shakernia, H.J. Kim, D.H. Shim, and S. Sastry. Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation. *IEEE Trans. Robotics and Automation*, 18(5):662 – 669, oct 2002.

[245] F.A. Voegeli, M.J. Smale, D.M. Webber, Y. Andrade, and R.K. O'Dor. Ultrasonic telemetry, tracking and automated monitoring technology for sharks. *Environmental Biology of Fishes*, 60(1):267–282, 2001.

[246] D. Wang, P.F.J. Lermusiaux, P.J. Haley, W.G. Eickstedt, D. P.and Leslie, and H. Schmidt. Acoustically focused adaptive sampling and on-board routing for marine rapid environmental assessment. *Journal of Marine Systems*, 78:393–407, 2009.

[247] Y. Wang and S. Boyd. Performance bounds for linear stochastic control. *Systems & Control Letters*, 58(3):178–182, 2009.

[248] L. Whitcomb, D.R. Yoerger, and H. Singh. Combined Doppler/LBL based navigation of underwater vehicles. In *International Symposium on Unmanned Untethered Submersible Technology*, 1999.

[249] L.L. Whitcomb, M.V. Jakuba, J.C. Kinsey, S.C. Martin, S.E. Webster, J.C. Howland, C.L. Taylor, D. Gomez-Ibanez, and D.R. Yoerger. Navigation and control of the nereus hybrid underwater vehicle for global ocean science to 10,903 m depth: Preliminary results. In *IEEE Conf. Robotics and Automation*, pages 594–600. IEEE, 2010.

[250] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *IEEE INFOCOM*, 2000.

[251] J.S. Willcox, J.G. Bellingham, Y. Zhang, and A.B. Baggeroer. Performance metrics for oceanographic surveys with autonomous underwater vehicles. *IEEE J. Oceanic Engineering*, 26(4):711–725, 2001.

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

[253] J. Yan and R.R. Bitmead. Incorporating state estimation into model predictive control and its application to network traffic control. *Automatica*, 41(4):595–604, 2005.

[254] A. Yetendje and M.M. De Doná, J.A.and Seron. Multisensor fusion fault tolerant control. *Automatica*, 47(7):1461–1466, 2011.

[255] A. Yetendje, M.M. Seron, and J.A. De Doná. Robust MPC multicontroller design for actuator faulttolerance of constrained systems. In *IFAC World Congress*, 2011.

[256] N.K. Yilmaz, C. Evangelinos, and N.M. Lermusiaux, P.F.J.and Patrikalakis. Path planning of autonomous underwater vehicles for adaptive sampling using mixed integer linear programming. *IEEE J. Oceanic Engineering*, 33(4):522–537, 2008.

[257] D. Yoerger, M. Jakuba, A. Bradley, and B. Bingham. Techniques for deep sea near bottom survey using an autonomous underwater vehicle. *Robotics Research*, pages 416–429, 2007.

[258] H. Zhang, J. Moura, and B. Krogh. Dynamic field estimation using wireless sensor networks: Tradeoffs between estimation error and communication cost. *IEEE Trans. Signal Processing*, 57(6):2383–2395, 2009.

[259] Y. Zhang, J.G. Bellingham, J.P. Ryan, B. Kieft, and M.J. Stanway. Two-dimensional mapping and tracking of a coastal upwelling front by an autonomous underwater vehicle. In *MTS/IEEE Oceans*, 2013.

[260] Y. Zhang, M.A. Bellingham, J.G.and Godin, and J.P. Ryan. Using an autonomous underwater vehicle to track the thermocline based on peak-gradient detection. *IEEE J. Oceanic Engineering*, 37(3):544–553, 2012.

[261] Y. Zhang, M. Godin, J.G. Bellingham, and J.P. Ryan. Using an autonomous underwater vehicle to track a coastal upwelling front. *IEEE J. Oceanic Engineering*, 37:338–347, 2012.

[262] Y. Zhang, J.P. Ryan, J.G. Bellingham, J.B.J. Harvey, and R.S. McEwen. Autonomous detection and sampling of water types and fronts in a coastal upwelling system by an autonomous underwater vehicle. *Limnology and Oceanography: Methods*, 10:934–951, 2012.

[263] K. Zhou and S.I. Roumeliotis. Optimal motion strategies for range-only constrained multisensor target tracking. *IEEE Trans. Robotics*, 24(5):1168–1185, 2008.