ABSTRACT

Title of dissertation: REMOTE ESTIMATION OVER USE-DEPENDENT CHANNELS

David Ward, Doctor of Philosophy, 2017

Dissertation directed by: Professor Nuno C. Martins
Department of Computer and Electrical Engineering

This dissertation investigates communication and estimation over channels whose transmission characteristics change with previous channel utilization and transmissions. We define three classes of channels: 1) Use-dependent discrete switching channels, 2) Use-dependent packet-drop channels, and 3) Shared-resource multiple packet-drop channels. In each of these classes of channels, there is a channel state that determines the channel's transmission characteristics.

For use-dependent discrete switching and packet-drop channels, there is a channel transmission policy that calculates the input to the channel state system. There is also an encoding policy that calculates the data to transmit over the channel. For these channels, we explore the properties, structure, and calculation of optimal channel transmission and encoding policies.

A discrete channel and a finite state machine, the channel state, form a use-dependent discrete switching channel. For each channel state, the discrete channel has different symbol transmission statistics. The transmission policy has access to

the output of the discrete channel. For a remote estimation problem with a conditional entropy cost over these channels, we show a partial separation between the design of transmission policies and encoding policies. Also, the optimal transmission and encoding policy are calculated for a specific use-dependent discrete switching channel.

A Bernoulli packet-drop link and a finite state machine, the channel state, form a use-dependent packet-drop channel. The channel state influences transmission performance by adjusting the probability of a packet-drop on the Bernoulli link. Each channel state corresponds to a specific drop probability. For a remote estimation problem with an expected mean-squared error cost over these channels, the structure of optimal transmission policies is explored.

For shared-resource multiple packet-drop channels, the channel has various modes of operation for transmitting multiple sensor measurements to an estimator over Bernoulli packet-drop links. Each mode of operation, or channel state, prioritizes the transmission of some sensor measurements over others. The channel state sets transmission priorities by adjusting the probability of packet-drop for each Bernoulli packet-drop link. In a given channel state, one sensor's drop probability is low, while another sensor's drop probability is high. For a remote estimation problem of transmitting the state of multiple systems over these channels, algorithms are presented to design the transition between transmission prioritization, channel states, to simultaneously stabilize the expected mean-squared estimation error of all the systems.

A detailed application of these results to operator support system design and a literature review of systematic design methods for decision support tools are presented.

REMOTE ESTIMATION OVER USE-DEPENDENT CHANNELS

by

David Ward

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2017

Advisory Committee:
Professor Nuno C. Martins, Chair/Advisor
Professor Michael Rotkowitz
Professor Sennur Ulukus
Professor Nikhil Chopra
Professor Richard J. La

Acknowledgments

At the conclusion of my time as a student, I am immensely grateful for my experiences and the time that I have had to learn and grow. I would like to greatly thank Nuno Martins, my advisor, for the guidance, care, and concern he has shown for me, as well as for the knowledge and tremendous growth I have had while under his mentorship. He has had a great impact on my life for which I am very grateful. Numerous faculty members played critical roles during my time as a graduate student. For their friendly demeanor, patience, and the feedback they provided, I am grateful. I would also like to thank Joe Notaro (NAVAIR) for the insightful discussions, support, and feedback that helped shape my dissertation. Brian Sadler (ARL) provided guidance at an important time during my graduate studies, and I am grateful for his guidance. My undergraduate advisor, Sean Warnick (BYU), has continued to be a great mentor and friend. I am immensely grateful for him and his impact in my life.

To my friends and fellow students, thank you for the fun, support and experiences we have had. To my family, thank you for the love and support. It is an amazing gift to always have people rooting for your success.

To my love, Candice, for the fun, love, and growth that we have experienced I am inexpressibly grateful and excited for the future.

To my two unconquerable boys, Peter and Michael, I love you and thank you for being awesome.

I am also grateful for the financial support my research received. Research reported in this dissertation was partially funded by grants AFOSR FA95501510367, NSF ECCS 1408320, ECCS 1135726 and (011367) Naval Air Warfare Center Aircraft Division / UMD Cooperative Agreement.

Table of Contents

Chapter 1: Introduction

This chapter introduces and motivates the problem formulations addressed in this dissertation. A motivating application for this research is the dynamic management and interaction with human operators and the systematic design of decision support systems. A literature review of relevant human operator phenomena is supplied. Also, presented is a review of control system design techniques for human operator controlled systems. For each of the problem formulations, a literature review specific to that formulation is presented in its corresponding chapter.

1.1 Motivation

The aim of this research is to improve the control and estimation of systems that communicate over channels whose performance is impacted by previous channel usage. Control of systems over communication channels is a challenging problem: the control of systems over channels with memory that determines channel performance even more so. This research focuses on remote estimation problems over specific classes of channels with memory. The classes of channels under investigation are motivated by applications to communication systems with energy harvesting capabilities and human operator support system design. These channels have an associated finite state machine (FSM) whose current state influences the transmission characteristics of the channel. Both packet-drop and discrete channels are considered in different problem formulations.

Often human operators are integral to the operation of engineered systems; however, engineered systems are often not designed with explicit considerations for the impact of a human operator. In specific scenarios, operators' performance level is well modeled by an FSM. For example, in [1] an FSM models the performance of unmanned air vehicle operators performing a visual search of images. In this dissertation, an operator's response is modeled as the output of a channel. Operators' performance is related to recent and current workload. Thus, the operator's response is modeled as a channel with memory that determines channel performance. By explicitly incorporating the impact of operator performance phenomena into control system design, this research seeks to improve the performance of the entire system, which includes both the human operator and the engineered system.

In battery-operated wireless communication systems with energy harvesting, the decision of whether to attempt a transmission must be made time and again at each time-step. The charge-level of the battery induces memory in the channel. An attempted transmission affects the battery-level and consequently impacts the current and future performance of the channel. In this dissertation, the battery dynamics are modeled using an FSM, which determines the instantaneous performance of the channel. Transmission policies are designed with explicit considerations for the energy harvesting capabilities and limitations of the communication system.

Guided by these primary motivating examples, the following problem formu-

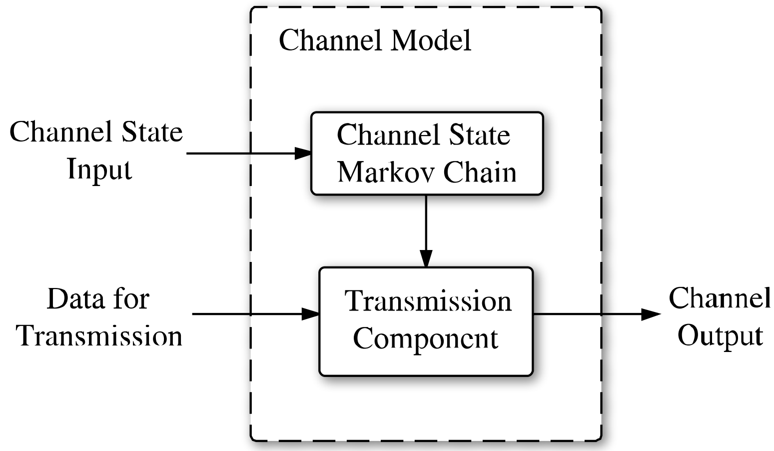

Figure 1.1: The general structure of the channels under investigation.

lations seek to improve the estimation of systems that communicate over channels with usage dependent performance.

1.2 Overview of Problem Formulations

This dissertation investigates remote estimation problems over channels with the general structure shown in Figure 1.1. This channel has a transmission component whose instantaneous performance is determined by the current state of a Markov chain. We refer to the Markov chain as the channel state. The channel has a channel state input and an input for the data to be transmitted. The channel output is the output of the transmission component. Many problems can be formulated using this channel structure depending on the choice of objective function, transmission component, Markov chain and channel state input. Guided by the motivating applications of channels with energy harvesting capabilities and operator support system applications, several specific research problems using this channel structure are discussed below.

The problem formulations presented are divided into two categories based on if the Markov chain's transition matrix is fixed or is one of the optimization variables. With a fixed transition matrix, the optimization variables are the channel state input and transmission variable input. If the Markov chain transition matrix is not fixed, the optimization variables are the data used for transmission and the transition matrix.

1.2.1 Problems with Fixed Markov Chain Dynamics

Two different problem formulations with fixed FSM dynamics are presented. Chapter 2 discusses use-dependent switching channels where the transmission component is a discrete channel, as shown in Figure 1.2. We formulate a problem where the objective is to minimize the conditional entropy of a source random variable given the channels output symbols. Chapter 3 discusses use-dependent packet-drop channels where the transmission component is a Bernoulli packet-drop channel, as shown in Figure 1.3. A problem is formulated where the objective is to minimize the mean-squared estimation error.

In Chapter 2, we investigate transmitting the realization of a random variable by multiple uses of a use-dependent switching channel, shown in Figure 1.2. The channel state input represents a transmission attempt, and the FSM dynamics of the channel state are fixed. The instantaneous statistics of the discrete channel depend on the current channel state. With a finite number of channel uses the encoder,

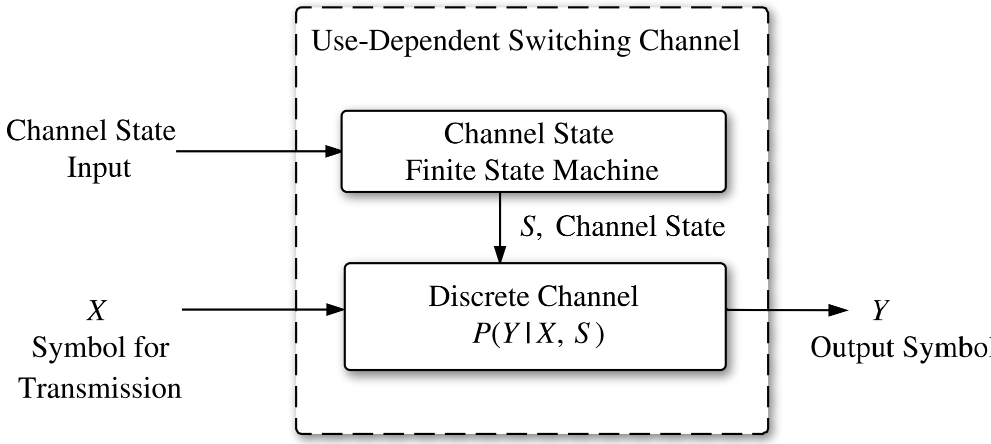

Figure 1.2: The structure of use-dependent switching channels. The current state of the finite state machine or channel state determines the statistics of the discrete channel.

decoder and transmission policy are designed to minimize the conditional entropy of the source random variable given the channel output symbols.

In Chapter 3, a remote estimation problem over a use-dependent packet-drop channel, as shown in Figure 1.3, is explored. This channel has a Bernoulli packet-drop transmission component and fixed channel state dynamics. The channel state input represents a transmission attempt. The encoder, decoder, and transmission policy are designed to track a random process by minimizing the mean square estimation error.

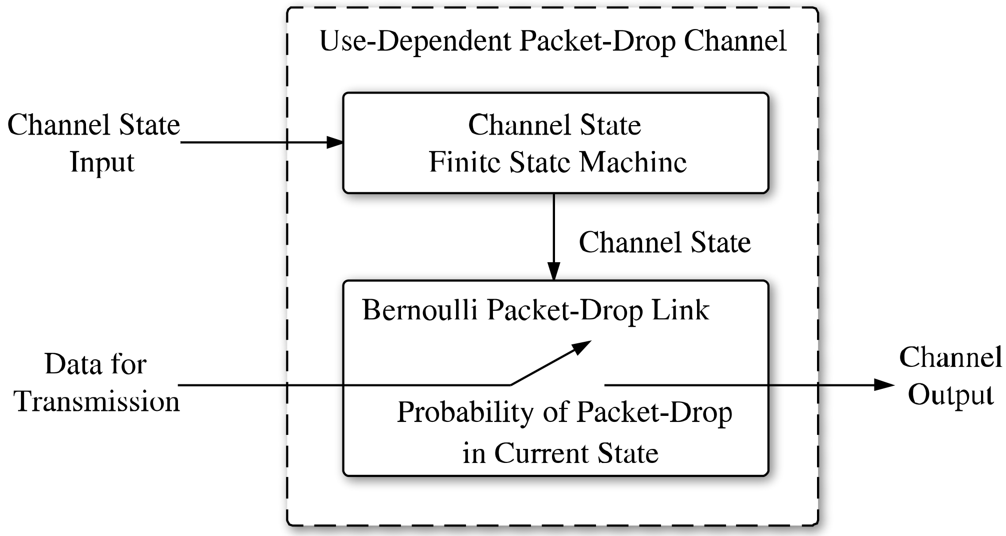

Figure 1.3: The structure of use-dependent packet-drop channels. The current state of the finite state machine or channel state determines the probability that the packet is dropped.

1.2.2 Designing the Transition Matrix of the Channel State Markov Chain Dynamics

In Chapter 4, a remote estimation problem of tracking multiple random processes by transmitting over a shared-state multiple packet-drop channel, as shown in Figure 1.4, is considered. No channel state input is used and the channel state is modeled as a homogeneous Markov chain. Each random process to be transmitted is remotely estimated over a Bernoulli packet-drop link. The current channel state determines the probability of a dropped packet for each link. This probability may be different for each link. The channel state transition matrix is designed to stabilize the mean-square estimation error of all systems.

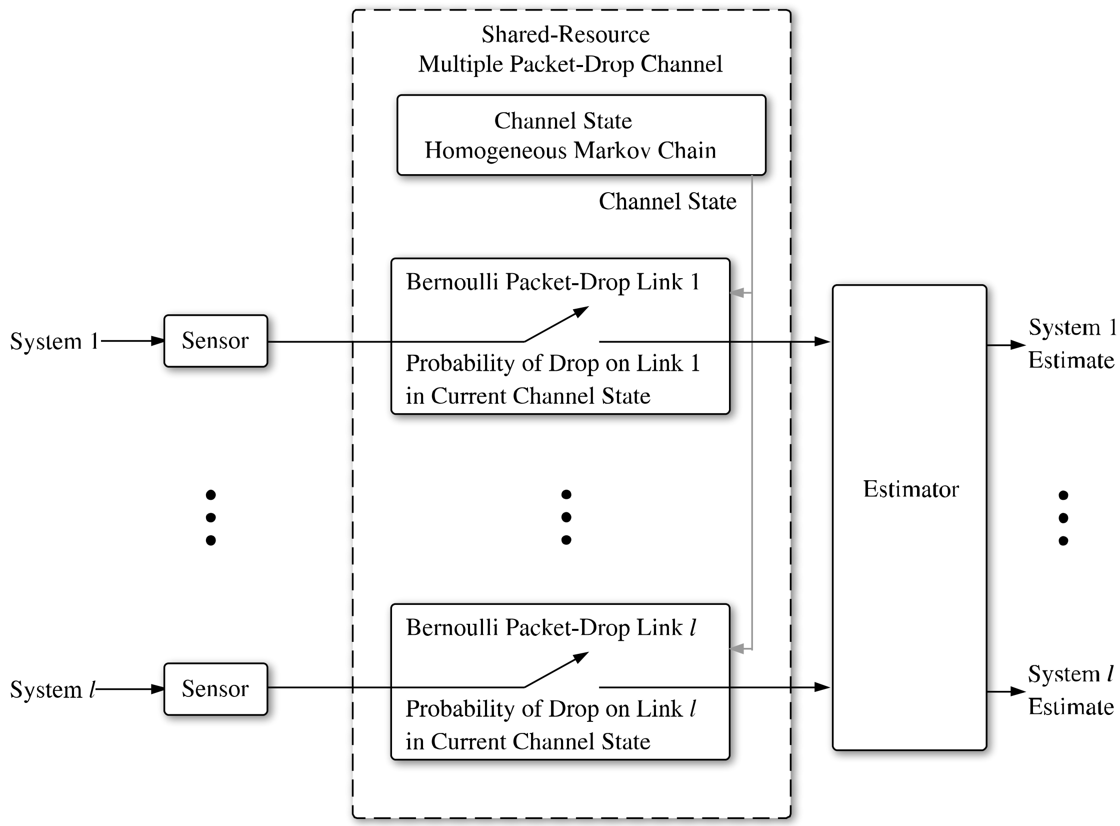

Figure 1.4: The structure of the channels investigated in Chapter 4. Methods of designing the Markov chain transition matrix to stabilize the mean square estimation error of all l systems transmitted are explored.

1.3 Dissertation Structure

Each subsequent chapter presents a detailed problem formulation, a literature review, and a research contribution. Also, numerical examples and applications to support system design and energy harvesting channels are presented where applicable.

In Chapter 2, a remote estimation problem over a use-dependent switching channel, see Figure 1.2, with an entropy cost, is investigated. Technical results isolating the design of the transmission policy from encoding and decoding policies is presented. Also, the optimal transmission policy for an average cost infinite horizon problem with a specific class of channel state FSMs is found.

In Chapter 3, a remote estimation problem over a use-dependent packet-drop channel, see Figure 1.3, with a mean-square error cost is presented. Under two different sets of assumptions, structural results characterizing the optimal solutions are presented.

In Chapter 4, a remote estimation problem over shared-state multiple packet-drop channels, as shown in Figure 1.4, is studied. We present algorithms that design the transition matrix of the channel state Markov channel to stabilize the estimation error of all transmitted systems in the mean-square stability sense or certify that no stabilizing transition matrix exists.

1.4 Literature Review

This section is a general introduction and literature review of mixed-initiative human-in-the-loop problems and applications. After further motivation and introduction, a literature review of pertinent human operator traits and models is given. This section is concluded by a literature review of systematic design techniques for human operator support systems.

The following applications exhibit strong collaboration between control systems and human operators: nuclear power plant monitoring [2], command and control of multiple unmanned air vehicles (UAVs) [3] [4], manufacturing [5], architectural design [6] [7], and a mobility augmenting jet-pack for low-gravity missions (i.e. manned Mars exploration) [8]. Understanding how to design control systems better suited for interaction with a human operator is desirable for different reasons in each application. In nuclear power plant monitoring, improved control system and operator collaboration lead to safer operation. In the command and control of multiple UAVs, improved collaboration leads to a higher UAV to operator ratio as well as improved mission performance. In architectural design, decision support systems can ease the burden of recalling and locating applicable building codes and material specifications; leading to faster and safer building designs.

The impact of control system interaction on operator performance is difficult to predict; however, key operator traits have been identified and well studied. The general framework and channel structures proposed in this dissertation have the capability to model operator performance that is affected by workload, biasing, speed-accuracy tradeoff, among other human phenomena. These human phenomena are discussed in Section 1.4.1. In the framework of this dissertation, as shown in Figure 1.1, the channel state dynamics are used to model the desired human phenomena. The channel's output represents the operator's responses or actions.

1.4.1 Literature Review - Human operator phenomena

The intent of this literature review is to highlight fundamental human phenomena critical to short term performance. Long term phenomena, such as learning, are not included. The problem formulations seek to design interaction during a single operation lasting minutes or hours. This literature review is skewed toward well-studied operator phenomena with mathematical models. Also, models of phenomena on very short time scales, such as physiological models of visual attention [9], are not included. The primary human phenomena motivating our problem formulations in Section 1.2 are workload, situational awareness, speed-accuracy tradeoff, and bias. Below these phenomena are discussed as separate independent mechanisms; however, they are connected. For example, situational awareness and bias are related: also, situational awareness and workload are arguably different descriptions of the same mechanism [10]. For this dissertation, the phenomena are considered separate mechanisms.

Operator workload refers to the phenomena of operator performance degradation under situations of high or low utilization. Simple discrete time models of operator workload are given in [11] and [12]. Operator performance degrades at

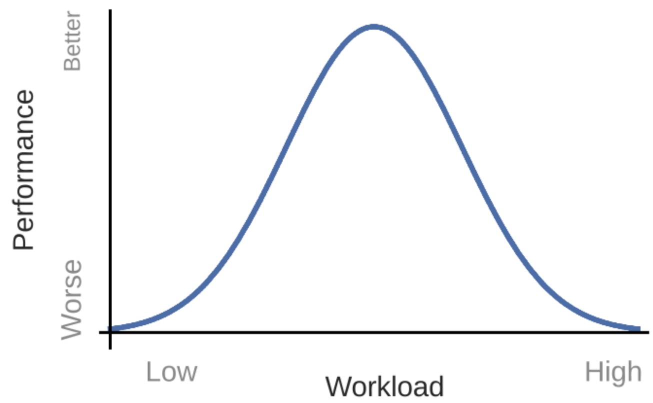

Figure 1.5: The Yerkes-Dodson law states that operator performance suffers at low and high workloads.

high workloads. Interestingly, operator performance also degrades at low workloads. This is referred to as the Yerkes-Dodson law [13] and is shown in Figure 1.5. Different experiments and applications use different interpretations of performance and workload [10]. A common interpretation for workload is the utilization ratio, the percentage of time the operator is busy. A common interpretation for decreasing performance is increasing task completion times [14]. Variations of these common interpretations are used in [1] and [15]. Alternatively, performance reduction can take the form of decision errors, less precise decisions, and the omission of tasks or subtasks [16] [17]. An alternative measure of workload is physiological metrics such as pulse, pupil dilation, sweating, or electroencephalogram (EEG) [18] [19]. Another common workload metric is the NASA multi-attribute task battery or the NASA task load index, which are questionnaires gauging task difficulty. These questionnaires are completed by operators during an intermission in the experiment or at its conclusion.

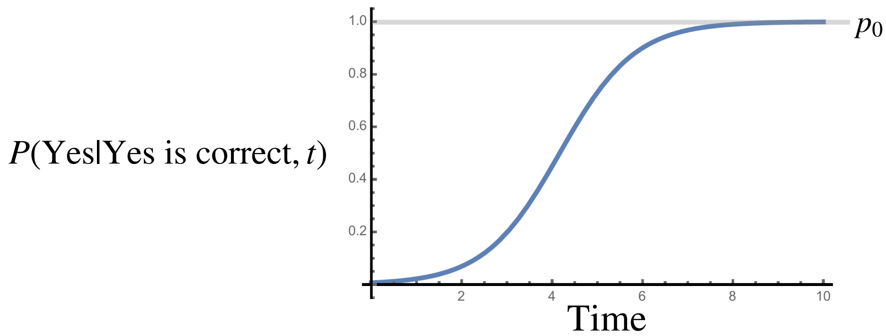

Figure 1.6: The time-accuracy tradeoff in human decision making is commonly modeled with a sigmoid function as highlighted by Pew's modeled shown here.

Two alternative forced choice tasks (TAFCs), which are tasks that require the operator to choose between two responses (i.e., Yes or No), are commonly used to studied human decision making phenomena. The responses to general TAFCs are arbitrarily referred to as 'Yes' and 'No'. Experiments and models of TAFCs have many variations. For example, the operator may be required to supply a response within a fixed amount of time or the operator may be allowed an indefinite amount of time to respond [20]. Two common models for TAFCs decisions, Pew's model and drift diffusion models (DDMs), are discussed below.

Operators often make better decisions when given more time for decision making. This speed-accuracy tradeoff for TAFCs is explicitly modeled by Pew's model [21]. Pew's model uses three parameters $p_0 \in [0, 1]$, a, $b \in \mathbb{R}$ to describe the probability of a 'Yes' response given that 'Yes' is the correct response and t seconds are used to make the decision,

$$P(\text{Yes}|\text{Yes is correct}, t) = \frac{p_0}{1 + e^{-(at+b)}}.$$

Figure 1.6 plots this probability versus the time taken for a decision. This probability as a function of time is sigmoidal. In [21] and [22], a generalized speed-accuracy model is employed that simply assumes this probability is a sigmoidal function.

Drift diffusion models (DDMs) also capture the speed-accuracy tradeoff in TAFCs by modeling the accumulation of evidence for a 'Yes' response [23]. A basic DDM uses two parameters, a drift rate $a \in \mathbb{R}$ and a diffusion rate $\sigma \in \mathbb{R}_{++}$, and the evidence $x \in \mathbb{R}$ evolves according to a stochastic differential equation

$$dx(t) = adt + \sigma dW(t), \quad x(0) = x_0,$$

where W is the standard Wiener process and $x_0 \in \mathbb{R}$ the initial condition. In an untimed response setting a 'Yes' is declared if x grows larger than an upper threshold and a 'No' is declared if x is smaller than a lower threshold. DDMs are used to model a variety of phenomena from neuron activation in the retina to strategic decisions. Although DDMs do not provide a simple closed-form expression for the probability of a correct decision, they provide a model of response times as well as decisions.

There are many forms of bias. We focus on sequential bias in TAFCs that occur due to the influence previous responses have on future responses [24]. An example of sequential biasing is when all the previous responses to unrelated TAFCs have all been 'Yes' by happen-stance. The operator is biased toward a future decision of 'Yes' and will also decide 'Yes' with a shorter response time. The model of sequential bias for TAFCs used in Chapter 2 is detailed in [25] and [26].

Situational awareness is an important human operator trait well represented in the literature. "[Situational Awareness is] the perception of elements in the environment, the comprehension of their meaning in terms of task goals, and the projection of their status in the near future." [27] Situational awareness is a descriptive model of the operator's understanding of the state of operations. Numerous examples illustrate the potential for catastrophic failure of aircraft or process control systems when operators lack situational awareness. A taxonomy of different operator-support system interactions, referred to as levels of autonomy, are commonly used in experiments to study the impact of support systems on operator situational awareness [5] [28] [29]. In [30], a descriptive model of how operators focus attention to maintain situational awareness in complex settings is given. Human operator limitations of finite working memory, limited attention and the impact of stress induce the development of strategic behavior to overcome these limitations such as operators creating mental models and goal-oriented behavior [31]. Also, situational awareness explains why switching between disparate tasks has a performance cost of accuracy and response time. This, as well as the performance benefits of a period of preparation before task switching is discussed in [32].

Memory retention is a well-studied phenomenon and has a critical impact on operator performance and situational awareness [33]. Two common experimental procedures test either the operators recall or recognition. In a recognition experiment, a sequence of words is presented to the operator who is tasked with identifying if a word was previously shown. In a recall experiment, a sequence of word pairs is presented to the operator who must later recall the second word when the first word is shown. The lag refers to either the number of intervening words or the amount of time between initial presentation and a recall or retention task. Common models for the probability of recall or retention are the exponential model

$$P(\text{Correct Recall}) = Ae^{-bT}$$

or the power model

$$P(\text{Correct Recall}) = AT^{-b},$$

where A and b are model parameters and T is the lag in either seconds or number of intervening tasks. In [33], a multi-term model is proposed

$$P(\text{Correct Recall}) = a_1e^{-T/t_1} + a_2e^{-T/T_2} + a_3,$$

where the first term represents short term memory and the other terms represent long term memory. Similar models are available for memory retention's impact on response time. Interestingly, [34] argues that the environment often follows an exponential decay similar to operators memory. In particular, the appearance of words in the New York Times headlines, the chance of receiving an email from a particular sender, and the frequency of words parents say to children, follows a similar exponential decay in probability.

The primary motivating human phenomena for this research are workload, speed-accuracy tradeoff, sequential bias and situational awareness; however, there are numerous other human phenomena that impact the design of mixed-initiative human-in-the-loop teams. Several of these other human phenomena are mentioned below. Multiple resource theory models an operator's ability to process different types of information in parallel [35]. For example, an operator can process auditory and visual information, listening to an audio book while driving, without significant performance degradation; however, processing auditory and visual information that both require symbol and logic processing, reading a book and listening to a different audio book, induces a performance degradation. In [36], this ability to simultaneously process information in different modalities is utilized to alert an operator and communicate the urgency and geographic location of the disturbance using either visual or haptic cues. In TAFC tasks, operator performance and decision making strategies vary depending on the precise nature of the TAFC structure. For example, operators exhibit different decision making strategies if the task has sensor uncertainty or outcome uncertainty [37]. Alertness cycles are another human phenomena impacting performance [20]. Stress response can have an impact of operator performance [38] [39]. The low-level physiological mechanisms governing operator vision and image parsing can impact perception [9]. Lastly, a critical human phenomena vital to consider in the design of support systems is the impact of personal differences and expertise [40] [20]. Some models of the phenomena mentioned in this paragraph may fit the general framework of the research presented.

Operator trust of the support system is vital but difficult to quantify. Trust is discussed in much of the literature cited above, particularly the situational awareness literature. In [41], an explicit simple linear system model of trust is presented.

1.4.2 Literature Review - Design of support systems

In this section, a literature review of systematic design for support systems that explicitly incorporates models of operator traits discussed in Section 1.4.1 is given. Approaches to the systematic design of operator support systems are hugely varied ranging from models predicting the number of UAVs an operator can manage [42] to fitting hidden Markov models of operator's attention allocation [43] and simulation based design [44]. The focus of this literature review is works that investigate fundamental design principles using simple operator models that address workload, bias and attention allocation. These are the most related literature to the contributions of this dissertation.

In a spirit similar to Chapter 2 and Chapter 3, the authors of [45], [11], and [1] consider operator workload management; however, the objective and workload models are different. In [45], [11], and [1] task release policies are designed to guarantee the stability of incoming queued tasks and maximize throughput. The utilization ratio, a ratio of the amount of time an operator has recently been busy, quantifies the operator's workload. With high utilization, the operator performs tasks more slowly. This model was verified via an experiment where the operator tasks were analogy questions. A task release policy that is a threshold function of the operator's utilization ratio can achieve the maximum throughput for tasks with a constant amount of work. Surprisingly, tasks with a random amount of work only increase the maximum throughput.

In [15] and [12], surveillance tasks are processed by a human operator who experiences speed-accuracy tradeoffs and workload effects. These tasks are generated by a support system which determines the region the operator should survey and for how long. Several incremental problems are addressed building up to a solution for this surveillance problem. In [22], a characterization of optimal time allotments for operator tasks that have a sigmoidal speed-accuracy trade-off model is developed. Also, an order N^2 algorithm, that is within a bound of optimal, is proposed that produce time allotments for sequential tasks that have a sigmoidal time-accuracy tradeoff penalty. In [15], a receding horizon solution to the optimal time allotment problem where operator tasks that have both a sigmoidal speed-accuracy tradeoff and the operator processes tasks slower with increased utilization ratio. A solution to the original surveillance problem includes a sequential probability ratio test based region selection policy that minimizes an upper bound of the average time to anomaly detection and the receding horizon solution from [15] to allocate time for searching the generated tasks. In [46], a similar problem of designing time allotments for operators with speed-accuracy trade-offs and workload considerations is addressed with the additional capability of the support system to re-queue tasks for additional analysis.

Task assignment problems are well studied in manufacturing and assembly line design [47]. Recently in [48], a greedy task assignment algorithm was augmented with operator workload and speed-accuracy trade-off models when assigning tasks that required both the attention of an operator and use of an unmanned vehicle. This algorithm was used to study the performance impact of different updating frequencies for task assignment.

Simulation based approaches to support system design are common especially in application areas where experiments are expensive, difficult or impractical, such as military operations. This design procedure is iterative and involves: task analysis and decomposition [44] [49], determination of models and parameters [50], discrete-event simulation [51] [52] and analysis of simulation results. A thorough example of this design process for a military vehicle driving through hostile territory is detailed in [52].

Numerous support system design techniques combine modeling, optimization, and experimentation. In [14], operators are tasked with finding and removing objects from a maze with 2,4,6 or 8 unmanned vehicles (UVs). The UVs operate in one of two modes: manual (Idle unless assigned) or automatic (Search unexplored areas if idle). This experiment was used to verify and fit a model of operator performance and workload. This model was then used to predict operator workload and performance while managing a more complex team of both manual and automatic UVs. In a different experiment where operators utilize a team of UVs to search and destroy targets, the impact of different levels of autonomy are explored [53]. Another aspect of support system design is the impact of multiple operators using the support system. In [54], the impact of two operators controlling the same pool of UVs is experimentally explored.

Further related works are mentioned below. In [55], a task schedule for an operator with uncertain task completion times is obtained by generating a finite number of task completion time realizations and solving an integer program constrained to schedules feasible for these realizations. In [56], a switching linear quadratic regulator (SLQR) is used to determine when an underwater vehicle is autonomously piloted or operator controlled. An algorithm for online trajectory planning for multiple UAVs and numerous targets are developed in [57] and [58] that guarantees the operator a non-conflicting real-time video feed of each target for a pre-planned length of time. In [59] with multiple operators and multiple UAVs available, an operator and UAV assignment policies are designed to minimize the expected time to classify a target. A strategy for piloting multiple UAVs simultaneously by using formations and dynamic leaders is implemented and tested in [60]. In [61], experiments, modeling and analysis of human strategies for multi-armed bandit problems is investigated. A mixed-initiative support system for air traffic control, proposed in [62], determines the maximum flow rates between airspace sectors and frees the operator to schedule within these rate bounds to achieve good overall network flow.

The problem formulation and results presented in this dissertation differ from the related literature discussed above. Many of the results discussed are specific to a given operator phenomena. One of the important contributions of this dissertation is that the results apply to any operator phenomena model used as the channel's Markov chain dynamics in Figure 1.1. For example, the algorithm proposed in Chapter 4 applies to designing operator attention allocation for any operator phenomena or combination of phenomena that can be described by a controlled finite state Markov chain.

Chapter 2: Optimal Remote Estimation over Use-Dependent Switching Channels

In this chapter, communication over channels with the structure shown in Figure 2.1 is considered. More specifically, a remote estimation problem is formed by a channel and an encoder that assesses a continuous random variable denoted as the source. The internal structure of the channel has a finite state machine (FSM) whose state dictates the transmission characteristics. Each state of the FSM corresponds to a specific discrete memoryless channel (DMC). At each transmission, information is transmitted from the encoder to the channel output according to the DMC selected by the current FSM state. This class of channels is denoted as use-dependent switching channels, or UDS. A transmission policy maps the channel's output into a decision to transmit across the channel at the next time step or not, this decision is the channel state input to the FSM. This chapter investigates methods to design optimal transmission and encoder policies that minimize the differential entropy of the source conditioned on a finite number of channel outputs.

This chapter is organized as follows: Section 2.1 provides an introduction and literature review, Section 2.2 introduces notation, defines UDS channels, and formulates the problem, Section 2.3 develops a separation principle between transmission and encoding policies; Section 2.4 develops optimal transmission policies for a specific UDS with a binary symmetric channel and a linearly saturating FSM, Section 2.5 and Section 2.6 contain applications, Section 2.7 is the chapter conclusion, and Section 2.8 contains several proofs.

The contributions of this chapter are listed below.

1. A partial separation principle between transmission policies and encoding policies is developed. Specifically, it is shown that there are optimal transmission policies for which the input to the FSM is a sequence that does not depend on the channel output. Thus, the jointly optimal transmission and encoding policies are constructed by first calculating an optimal transmission policy and then building an optimal encoding policy assuming the use of the optimal transmission policy.

2. Optimal transmission and encoding policies are found for a specific UDS where the FSM parametrizes the crossover probabilities of a Binary Symmetric Channels (BSC) and the FSM is a linear chain model that degrades the crossover probabilities as a result of sequential transmissions and recovers when there are no transmissions.

3. A novel formulation for human operator support system design is presented and the technical results are used to investigate optimally querying an operator who is affected by workload performance degradation and sequential biasing.

2.1 Introduction

We seek to design remote estimation policies over a discrete channel whose transmission characteristics depend on the state of a finite state machine (FSM). Consider Figure 2.1, where X^*, X_n, Y_n and S_n represent a source random variable, the input to the channel, the channel output and the state of the FSM with input U_n, respectively. Each state of the FSM is associated with a discrete memoryless channel (DMC) specified by $p(Y_n|X_n, S_n)$, which characterizes the transmission statistics from the encoder to the channel output. The interconnection of the FSM and switching DMC, represented in Figure 2.1, is denoted as a Use-Dependent Switching (UDS) channel.

The goal is to minimize the differential entropy of the conditional distribution of the source given N channel outputs, $H(X^*|Y^N)$, where output feedback is present. The two design variables are the encoder $\mathcal{E}_n$ and the transmission policy $\mathcal{K}_n$, which assesses current and past channel outputs to produce the input symbol X_n and the channel state input U_n, respectively. The channel state input is the input to the FSM of the channel. This input U_n controls the evolution of the channel state, which takes the form

$$S_n = \mathcal{F}_n(S_{n-1}, U_n), \tag{2.1}$$

with known, deterministic initial condition s_0.

For UDS channels, a partial separation occurs between the design of channel state inputs and the design of encoders. In Section 2.3, Theorem 1 shows that there

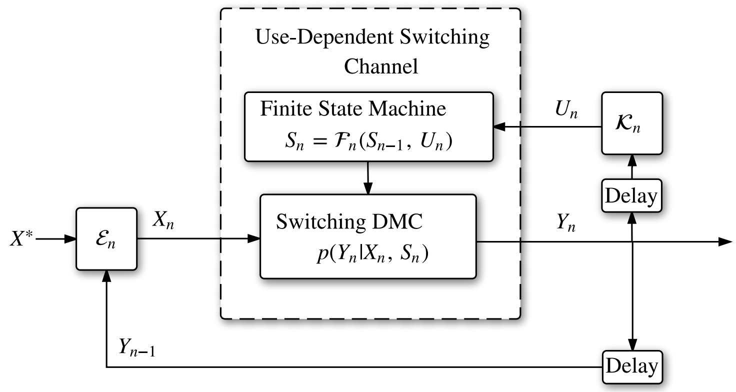

Figure 2.1: Use-Dependent Switching (UDS) channels are represented by the dashed box. UDS channels are comprised of a switching Discrete Memoryless Channel (DMC) whose transition probabilities switch according to the state process which is controlled by the channel state inputs U_n. We seek to design the encoder $\mathcal{E}^N$ and transmission policy $\mathcal{K}^N$ to minimize the entropy $H(X^*|Y^N)$ when feedback is present.

exists an optimal input sequence U_n that can be designed without regard to the encoding and does not depend on the channel output. Using this channel state input sequence, an optimal encoding is any encoding that induces the capacity achieving input distribution for the DMC determined by the current channel state. Thus, the optimal encoding does depend on the optimal channel state input sequence.

A simple model of a battery-operated wireless channel with energy harvesting is a compelling example of a UDS channel. Let the channel state be the energy level of the battery and the channel state input the decision to transmit across the channel. When an input to transmit occurs, the channel transmission characteristics depend on the battery level, $P(Y_n|X_n,\ S_n)$, and the batteries energy level reduces,

$$S_{n+1} = \mathcal{F}_{n+1}(S_n,\ \text{`Transmit'}).$$

The transmission policy determines when to transmit and the encoding policy determines the symbol to transmit. Further details of this application are in Section 2.5.

This research builds upon the work reported in [63] and [64], which develop optimal encoding strategies for transmission over a BSC with a fixed cross over probability to minimize the conditional entropy of the source given the channel outputs. The results here consider the larger class of UDS channels that require jointly designing a transmission policy and encoding policy.

UDS channels are distinct from channels with action-dependent states defined in [65]. The state S_n of channels with action-dependent states is the output of a memoryless channel $P(S_n|U_n)$ with the action U_n as input; whereas with UDS channels, the channel state has memory, is deterministic given the actions U^N and evolves according to eq. (2.1). An additional distinction is that, we have assumed the input, output, and channel state alphabets, $\mathbb{X}$, $\mathbb{Y}$, and $\mathbb{S}$, to be finite.

As the example above highlights, this research is related to communication systems with energy harvesting capabilities, which have been formulated in numerous ways [66]. For instance, [67] uses a distortion metric for the cost, whereas [68] uses a throughput cost. The conditional entropy cost used here is related to the throughput cost, see Section 2.6. The proposed problem is distinct from such formulations and the related literature in two ways; first the partial separation result holds for any FSM, no specific model is assumed, and second there is no noise as an input to the FSM. So directly modeling stochastic energy harvesting is not possible.

2.2 Problem formulation

2.2.1 Notation

Double-bared fonts are used to denote sets, calligraphic fonts to denote functions, capitals for random variables and lowercase for realization of random variables. This chapter uses the following notation:

$\mathbb{S}$	Channel State alphabet$\lvert\mathbb{S}\rvert < \infty$,
$\mathbb{U}$	Channel State Input,
$\mathbb{Y}$	Output alphabet, $\lvert\mathbb{Y}\rvert < \infty$,
$\mathbb{X}$	Input alphabet, $\lvert\mathbb{X}\rvert < \infty$,
Q^n	$\{Q_1, Q_2, \ldots Q_n\}$,
H	Differential Entropy,
$\mathcal{U}[a, b]$	Uniform distribution on $[a, b]$.

Definition 1 (Use-dependent switching channels) *Use-Dependent Switching (UDS) channels are channels from $\mathbb{X} \to \mathbb{Y}$ for which the probability of the output conditioned on the input depends on the state of the channel $S_n \in \mathbb{S}$. More specifically, the output of the channel Y_n is governed by $p(Y_n|X_n,\, S_n)$. The state is governed by the finite state machine*

$$S_n = \mathcal{F}_n(S_{n-1},\, U_n), \tag{2.2}$$

with known initial state s_0.

2.2.2 Problem formulation

We seek to transmit a random variable X^* taking values in $\mathbb{R}^d$ across a UDS channel with feedback present. Our design problem consists of selecting the encoder $\mathcal{E}_n$ and policies that determine U_n from the available information y^{n-1}. The encoding strategy $\mathcal{E}_n : \mathbb{R}^d \to \mathbb{X}$ is an assignment of X^* to channel symbols X_n. Let $\mathcal{G}^N$ and $\mathcal{K}^N$ be the policies for selecting channels state input and encoding strategies as a function of the channel output,

$$\begin{aligned} \mathcal{E}_n &= \mathcal{G}_n(y^{n-1}) \text{ and} \\ u_n &= \mathcal{K}_n(y^{n-1}). \end{aligned}$$

The belief of X^* after n transmissions is

$$p_n \stackrel{def}{=} P(X^*|Y^n = y^n).$$

Assume that the distribution of X^* is known and denote it p_0. The cost of policy $(\mathcal{G}^N, \mathcal{K}^N)$ is

$$J_N(\mathcal{G}^N, \mathcal{K}^N) \stackrel{def}{=} \frac{1}{N} H(X^*|Y^N),$$

when $(\mathcal{G}^N, \mathcal{K}^N)$ is used to generate u_n, $\mathcal{E}_n$ and consequently x_n and y_n. Using the notation that $H(f)$ is the differential entropy of the density function f, the cost is written as

$$\begin{aligned} J_N(\mathcal{G}^N, \mathcal{K}^N) &= \frac{1}{N} E[H(X^*|Y^N = y^N)] \\ &= \frac{1}{N} E[H(p_N)], \end{aligned}$$

where the expectation is taken with respect to Y^N.

Problem 1 *(Finite Time Horizon) Solve the following for finite N:*

$$\min_{(\mathcal{G}^N, \mathcal{K}^N)} J_N(\mathcal{G}^N, \mathcal{K}^N).$$

The cost in the infinite horizon case is $J(\mathcal{G}, \mathcal{K}) \stackrel{def}{=} \limsup_{N\to\infty} \frac{1}{N} H(X^*|Y^N)$, with respect to policies $\mathcal{G} = \{\mathcal{G}_n\}_{n=1}^{\infty}$ and $\mathcal{K} = \{\mathcal{K}_n\}_{n=1}^{\infty}$.

Problem 2 *(Infinite Time Horizon) Solve the following:*

$$\min_{(\mathcal{G},\mathcal{K})} J(\mathcal{G},\,\mathcal{K}).$$

It will be convenient to define the following. Let $s_n(u^n)$ be the state that is generated by the sequence of inputs u^n. Let $C(u^n)$ be the capacity of the channel $p(Y_n|X_n,\,S_n = s_n(u^n))$ that results from the sequence of inputs u^n.

2.3 Design separation principle

In this section, the optimal policy $(\mathcal{G}^N, \mathcal{K}^N)$ for the finite time horizon is characterized. It is shown that the optimal transmission policy $\mathcal{K}^N$ can be determined independent of the optimal encoding policy and does not depend on the channel output. This is accomplished through dynamic programming by finding the form of the cost-to-go functions in Lemma 1. The optimal transmission policy is a function of the FSM dynamics, $\mathcal{F}_n$, and transition probabilities $p(Y_n|X_n,\,S_n)$ alone.

For finite N, the dynamic programming recursion for Problem 1 is written as

$$\begin{aligned} V_N(u^N,\,y^N) = &\quad H(X^*|Y^N = y^N) = H(p_N), \\ V_n(u^n,\,y^n) = &\quad \inf_{\mathcal{E}_{n+1},\,u_{n+1}} E[V_{n+1}(u^{n+1},\,y^{n+1})], \end{aligned} \tag{2.3}$$

for $n = 0 \ldots N-1$, where the expectation is over $Y_{n+1}|u^n,\,Y^n = y^n$. The current belief of X^* and the previous actions $(p_{n-1},\,U^{n-1})$ form a sufficient statistic at time n. This is because V_N is a function of p_N and there are no stage costs. The previous actions U^{n-1} are necessary to calculate the channel transition probabilities $P(Y_n|X_n, S_n)$ and thus to update p_n via Bayes' rule.

Lemma 1 *The cost-to-go-functions, eq. (2.3), associated with Problem 1 take the form*

$$V_n(u^n,\, p_n) = \min_{u_{n+1}^N} H(p_n) - \sum_{i=n+1}^{N} C(u^i). \tag{2.4}$$

Proof 1 *See proof 6 in Section 2.8.*

Theorem 1 *The jointly optimal encoding and transmission policy is characterized by the following two step process. First, calculate the optimal transmission policy by solving*

$$\underset{\mathcal{K}^N}{\text{maximize}} \sum_{i=1}^{N} C(u^i) = \underset{u^N}{\text{maximize}} \sum_{i=1}^{N} C(u^i), \tag{2.5}$$

where $C(u^i)$ is the channel capacity of the faulty channel at time i if the actions u^i have been used.

Second, the optimal encoding strategy $\mathcal{E}^N$ achieves at each time step n the channel capacity $C(u^n)$. Let the input symbol distribution p_n^* achieve the channel capacity $C(u^n)$ for the channel $p(Y_n|X_n, S_n = s_n(u^n))$. At time n the encoding policy $\mathcal{E}_n$ is chosen such that for each input symbol x the following holds,

$$p(\mathcal{E}_n(X^*) = x) = p * (X_n = x).$$

Proof 2 *See proof 7 in Section 2.8.*

Interestingly, even with feedback present, the design of channel state input does not utilize feedback information. Feedback information is used in the encoding strategy, since the encoder $\mathcal{E}_n$ is selected to create the capacity achieving input distribution based on p_{n-1}.

2.4 Binary symmetric channel with a linearly saturating finite state machine

A specific UDS channel that degrades with use and recovers when not in use is considered in this section. The channel state input U_n specifies whether the channel transmits. The channel state controls the crossover probability q_n of a binary symmetric channel (BSC), see [69] for the definition of a BSC. We adopt the convention that the crossover probability q_n increases (decreases) by a positive α if a transmission is (not) sent. The crossover probability is one half when the channel is not used to transmit, rendering the BSC useless.

Because the channel has memory previous query strategies for similar formulations [64] and [63] do not immediately apply and can perform poorly.

2.4.1 Optimal encoding policy, $\mathcal{G}_1^N$

The optimal encoding policy described below is denoted $\mathcal{G}_1^N$. The input distribution that achieves the channel capacity for any BSC assigns equal probability to both input symbols irrespective of the crossover probability q_n. Thus, any encoding $\mathcal{E}_n$ that produces

$$P(X_n = 1|Y^{n-1} = y^{n-1}) = P(X_n = 0|Y^{n-1} = y^{n-1}) = \frac{1}{2}$$

is optimal.

Following [64] and [63], let the encoder $\mathcal{E}_n$ equal $\mathbf{1}_{\mathbb{A}_n}$ where $\mathbf{1}_{\mathbb{A}}(x)$ is the indi-

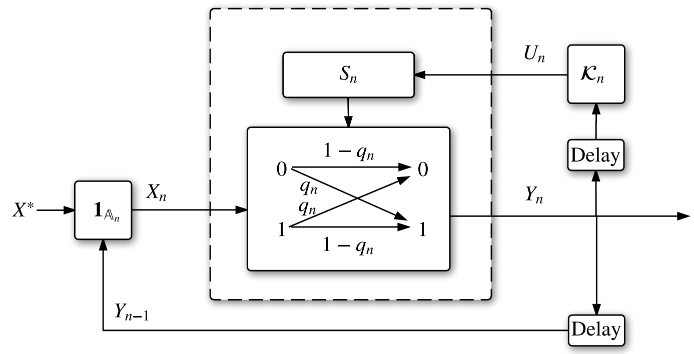

Figure 2.2: The specific UDS channel under investigation in Section 2.4 has a binary symmetric channel (BSC) with a linearly saturating channel state model. The switching discrete channel is a BSC whose crossover probability q_n is determined by the FSM as described in eq. (2.6) and eq. (2.7). The encoder $\mathcal{E}_n$ is chosen as the indicator function of the set $\mathbb{A}_n$, with $\mathbb{A}_n$ chosen to achieve the capacity of $P(Y_n|X_n,\, S_n = s_n(u^n))$ which ensures optimality of the encoder $\mathcal{E}_n$.

cator function of $x \in \mathbb{A}$. The transmission symbol is now defined as

$$X_n \overset{def}{=} \mathbf{1}_{\mathbb{A}_n}(X^*).$$

In order to be optimal, $\mathbb{A}_n$ is selected such that $P(X^* \in \mathbb{A}_n|Y^{n-1} = y^{n-1})$ equals one half.

For this particular UDS channel the design of transmission policy $\mathcal{K}^N$ and encoding policies $\mathcal{G}^N$ have separated into two independent problems. Since the capacity achieving input distribution for BSCs is agnostic to the crossover probabilities q_n, the design of the encoding policy $\mathcal{G}^N$ does not depend on the transmission policy $\mathcal{K}^N$. Also, the optimal transmission policy $\mathcal{K}^N$ does not depend on $\mathcal{G}^N$ by Theorem 1.

2.4.2 Linearly saturating channel state model

In this section, the linearly saturating FSM is defined. It is depicted in Figure 2.3. The channel state input U_n takes values 1 or -1, depending on whether the channel is used or not, respectively. The crossover probability q_n and memory to track the possible crossover probability z_n form the state S_n as follows,

$$S_n = \begin{bmatrix} z_n \\ q_n \end{bmatrix}.$$

When the channel is not used, $u_n = -1$, q_n equals one half, rendering the BSC useless. When there is a transmission, $u_n = 1$, q_n equals the possible crossover probability z_n. The parameters $\underline{\epsilon}$ and $\bar{\epsilon}$ take values in the interval $[0, .5]$ with $\underline{\epsilon}$ less than $\bar{\epsilon}$. The maximum crossover probability is $\bar{\epsilon}$ and the minimum crossover probability is $\underline{\epsilon}$. For a fixed parameter α, the dynamics of the channel state are

$$z_{n+1} = z_n + u_n\alpha, \tag{2.6}$$

$$q_n = \begin{cases} z_n & \text{if } u_n = 1 \\ \frac{1}{2} & \text{if } u_n = -1, \end{cases} \tag{2.7}$$

where z_n saturates at the endpoints of $[\underline{\epsilon}, \bar{\epsilon}]$.

The following assumptions are made for ease of exposition.

Assumption 1 $m \stackrel{def}{=} \frac{\bar{\epsilon}-\underline{\epsilon}}{\alpha} \in \mathbb{N}$.

Assumption 2 $s_0 = \underline{\epsilon} + i\alpha$, *for some* $i = 0, 1, \ldots, m$.

2.4.3 Optimal transmission policy, $\mathcal{K}_1^N$

First, near optimal finite horizon policies for Problem 1 are presented. Second, these policies are shown to be optimal in the infinite horizon case, Problem 2.

2.4.3.1 Finite horizon problem

The task is to find a $u^N \in \{-1, 1\}^N$ that maximizes

$$\sum_{i=1}^{N} C(u^i). \tag{2.8}$$

Dynamic programming can be used; however, the simple transmission policy $\mathcal{K}_1^N$, defined below, is near optimal for this linearly saturating channel state model. For this channel state model the capacity $C(u^n)$ reduces to the channel capacity of a BSC and so we overload the notation and let $C(q_n)$ denote the channel capacity of a BSC with cross-over probability q_n.

Definition 2 *The transmission policy $\mathcal{K}_1^N$ is a deterministic policy that does not depend on channel outputs. If $2C(\bar{\epsilon}) \geq C(\underline{\epsilon})$, define the transmission policy $\mathcal{K}_1^N$ to always transmit,*

$$U_n = 1, \qquad n = 1,\, 2,\, \ldots N.$$

If $2C(\bar{\epsilon}) < C(\underline{\epsilon})$, define the transmission policy $\mathcal{K}_1^N$ to transmit only when $z_n = \underline{\epsilon}$,

$$U_n = \begin{cases} 1 & \text{If } z_n = \underline{\epsilon}, \\ -1 & \text{otherwise.} \end{cases}$$

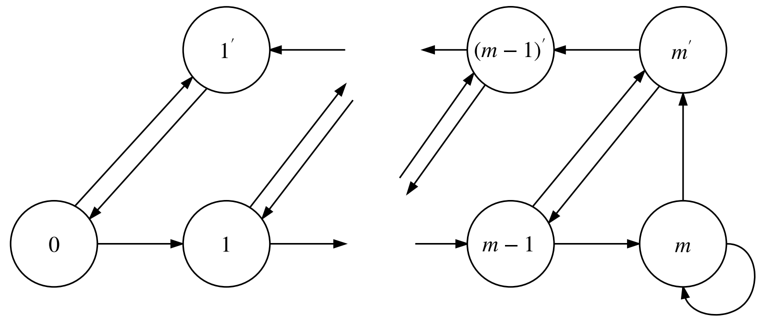

Figure 2.3: Graphical representation of eq. (2.6) and eq. (2.7) with saturation at the endpoints.

To motivate this transmission policy a graph for the dynamics of z_n is shown in Figure 2.3. Node i represents $z_n = \underline{\epsilon} + i\alpha$ and a transmission is sent. Node i' represents $z_n = \underline{\epsilon} + i\alpha$ and a transmission is not sent. The transitions in the figure represent the choice to transmit or not at $n+1$, the following time step. In node i since a transmission is sent at n, $z_{n+1} = \underline{\epsilon} + (i+1)\alpha$ and the next node will be $i+1$ (if a transmission is sent at $n+1$) or $(i+1)'$ (if a transmission is not sent at $n+1$). In node i' since a transmission is not sent at n, $z_{n+1} = \underline{\epsilon} + (i-1)\alpha$ and the next node will be $i-1$ (if a transmission is sent at $n+1$) or $(i-1)'$ (if a transmission is not sent at $n+1$).

The transmission policy in Definition 2 is motivated as follows. In all of the primed nodes of Figure 2.3, the choice was made to not transmit and thus a reward of $C(.5) = 0$ is gained. But, in the unprimed node i the choice was made to transmit gaining a reward of $C(\underline{\epsilon}+i\alpha)$. On average, a reward higher than $C(\bar{\epsilon})$ is only possible by previously not transmitting. The idea in Theorem 2's proof is that to spend time in a node besides node m, the policy must transmit about as much as it does not transmit. If this is the case, the best choice is to alternate transmissions such that the transmission happens in node 0.

Theorem 2 *There exists a constant w such that the policy $(\mathcal{G}_1^N, \mathcal{K}_1^N)$ is within a $\frac{w}{N}$ bound of optimal for Problem 1. More specifically,*

$$|J_N^* - \tilde{J}_N| \leq \frac{w}{N}$$

where J_N^* is the optimal cost and $\tilde{J}_N$ is the cost when policy $(\mathcal{G}_1^N, \mathcal{K}_1^N)$ is used.

Proof 3 *See proof 8 in Section 2.8.*

2.4.3.2 Infinite horizon problem

Definition 3 *The policy $(\mathcal{G}_1, \mathcal{K}_1)$ is policy $(\mathcal{G}_1^N, \mathcal{K}_1^N)$ with $N = \infty$. This is possible because policy $(\mathcal{G}_1^N, \mathcal{K}_1^N)$ is the same policy for every time step n regardless of N.*

Leveraging Theorem 2, in this section it is shown that policy $(\mathcal{G}_1, \mathcal{K}_1)$ is optimal for Problem 2. The cost of the infinite horizon problem under policy $(\mathcal{G}_1, \mathcal{K}_1)$ is also calculated.

Theorem 3 *For Problem 2, policy $(\mathcal{G}_1, \mathcal{K}_1)$ is optimal;*

$$J(\mathcal{G}_1, \mathcal{K}_1) \leq J(\mathcal{G}, \mathcal{K}) \qquad \forall\, \mathcal{G}, \mathcal{K}.$$

Also,

$$J(\mathcal{G}_1, \mathcal{K}_1) = H(p_0) - \max\left(C(\bar{\epsilon}), \frac{C(\underline{\epsilon})}{2}\right). \tag{2.9}$$

Proof 4 *See proof 9 in Section 2.8.*

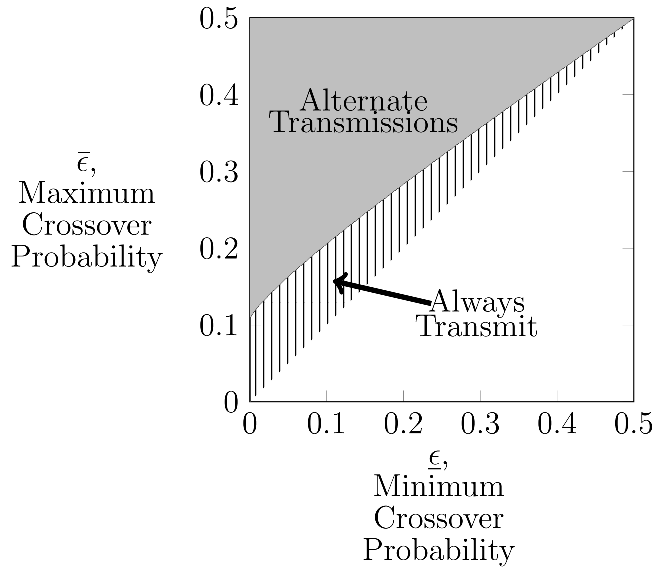

Figure 2.4: Plot of different transmission policy regimes for the transmission policy $\mathcal{K}_1^N$, in Definition 2, depending on the problem parameters. The light gray region corresponds to a policy that transmits only in channel state 0, which leads to a sequence of alternating decisions to transmit and not transmit. The region of vertical lines denotes the parameter region for which the optimal policy is to alway transmit.

In Figure 2.4, a diagram shows when the optimal transmission strategy $\mathcal{K}_1^N$ is either to always transmit or alternate transmissions depending on the parameters $\underline{\epsilon}$ and $\bar{\epsilon}$.

2.5 Application: operator support system design

Human operators are interacting with and supervising increasingly complex autonomous systems. Examples include: unmanned vehicles [70], medical treatments [71], and crowd sourcing [72]. Incorporating human considerations into the design of the autonomy is important. Utilizing the framework of [63], two such problems are given in the following subsections. These applications use a UDS channel to model human operator phenomena.

First, the tradeoff between operator workload and performance will be modeled using the formulation of Section 2.4. Thus, Theorem 3 provides the optimal policy for balancing workload and performance. Second, operator biasing and error modeled as in [25] is shown to fit the framework of UDS channels. Thus, the optimal encoding strategy is provided in Theorem 1.

In these applications, the operator's response to queries are represented by the UDS channel's output. The channel state dynamics are used to model different human operator phenomena. In the two examples below, the channel state dynamics model either the impact of workload or biasing on operator accuracy in answering *yes-no* questions.

2.5.1 Operator workload

Consider the channel's output to be the operator's responses to a sequence of *yes-no* questions. If the operator's workload becomes too high, the probability of a mistake in answering the questions increases. Meaning that the cross-over probability of the BSC increases. Operators under high workload tend to make more mistakes. This is precisely the model in Section 2.4. By Theorem 3, the optimal transmission policy is $\mathcal{K}_1^N$. Using the optimal transmission policy of Section 2.4 leads to the most informative operator inquiries.

The framework of [63] is used in this application. The problem under consideration in [63] is to localize a target X^* by minimizing the conditional entropy $H(X^*|Y^N)$ by asking the operator if the target is in the area $\mathbb{A}_n$. The difference in this application from [63] is that here the crossover probability is affected by the channel's transmission history. In other words, a UDS channel is used instead of a BSC.

2.5.2 Operator sequential bias

An application to support system design for human operators that incorporates operator models of sequential bias is presented. Operator sequential bias is illustrated in the following situation. When asking a human a sequence of *yes-no* questions, the answers to the previous questions will bias the operator's answers to the questions that follow. For example, a human operator that is asked a series of *yes-no* questions whose answers all happen to be *yes* will become biased toward responding *yes* to future questions. A model is given in [26] and [25] for operator sequential bias. This operator model fits into the framework of UDS channels as shown in Figure 2.5. The optimal encoder $\mathcal{E}_n$ is provided by Theorem 1.

In this model, the operator's responses to binary queries are the channels output Y_n. The choice to transmit (or not) U_n is the choice to ask the operator a question (or not). The channel state dynamics model the sequential biasing dynamics. The optimal encoding policy $\mathcal{G}_1$ determines what questions to ask.

The model proposed in [25] for the human operator is called the Dynamic Belief

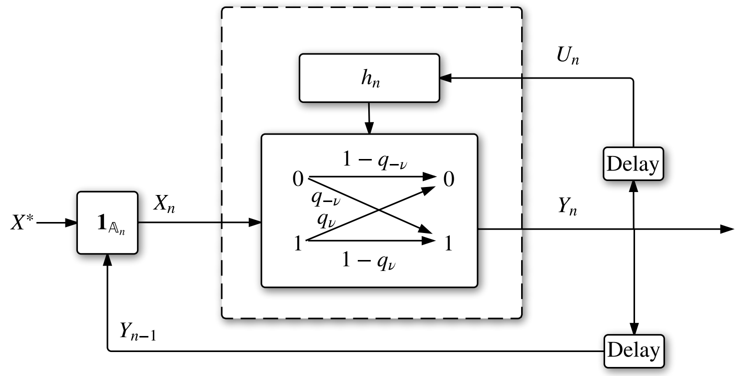

Figure 2.5: The UDS channel modeling the sequential bias of human operators. The operator biasing model used is proposed in [26] and [25]. The optimal encoding policy is given in Theorem 1.

Model (DBM). DBM postulates that the operator believes that the question $\mathbf{1}_{\mathbb{A}_n}$ is Bernoulli distributed with parameter γ_n. The Bernoulli parameter γ_{n+1} follows dynamics such that it is unchanged with probability τ or is re-drawn from a given distribution, p^0_γ with probability $1-\tau$.

The operator's belief that $\mathbf{1}_{\mathbb{A}_n}$ is Bernoulli with parameter γ_n impacts the operator's answer to the next question given their pervious responses. This is denoted, $h_n \stackrel{def}{=} P(x_n = 1 | y^{n-1})$. The operator's belief is updated by

$$h_{n+1} = \frac{1}{2}(1-\tau) + \frac{1}{3}\tau y_n + \frac{2}{3}\tau h_n.^{1} \tag{2.10}$$

This bias affects the operator's decision in the following manner. In [26], a

[1]A slight modification to the model in [25] is made. In [25], y_n is replaced by x_n in eq. (2.10). Here instead of the correct answers, x_n, updating the operator's bias. It is the operator's answers, y_n, that update their bias.

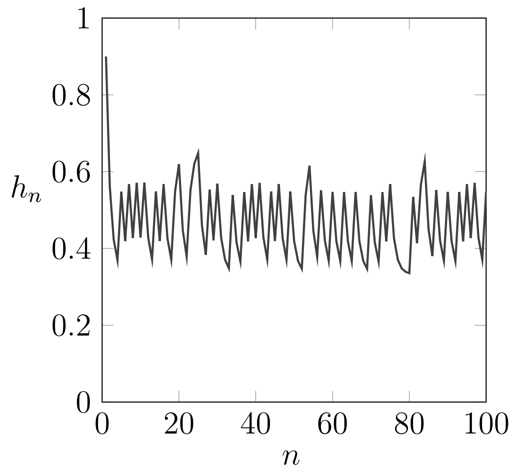

Figure 2.6: A simulation of an operator's bias with initial bias $h_0 = .9$ and the optimal encoding policy $\mathcal{K}_1$ which satisfies eq. (2.13).

drift diffusion model (DDM) is used with the upper, ν, and lower $-\nu$ thresholds determined by the probability of type I error that is tolerated, ϵ. The threshold is $\nu \stackrel{def}{=} \log(1-\epsilon) - \log(\epsilon)$.

We can nicely approximate the probability of making errors in this DDM model [73]. If the mass beyond the chosen thresholds is negligible, then the probability of $x_n = 0$ and the operator declaring $y_n = 1$ is approximately

$$q_{-\nu} \stackrel{def}{=} \frac{e^{-\nu} - e^{-I(h_n)}}{e^{-\nu} - e^{\nu}}, \tag{2.11}$$

where $I(h_n) \stackrel{def}{=} \log(h_n) - \log(1-h_n)$. Similarly, the error of $x_n = 1$ and the operator declaring $y_n = 0$ is

$$q_{\nu} \stackrel{def}{=} \frac{e^{I(h_n)} - e^{-\nu}}{e^{\nu} - e^{-\nu}}. \tag{2.12}$$

To summarize this model, the operator's belief that the next answer will be 1 is h_n and is governed by previous answers y^{n-1}. This belief skews the probability of making an error, q_ν and $q_{-\nu}$.

This operator model is a UDS channel with a binary asymmetric faulty channel and a FSM governed by eq. (2.10), eq. (2.11) and eq. (2.12). Per the model in [26] and [25], the transmission policy is chosen to be the identity.

Using Theorem 1 and the capacity achieving distribution for Binary Asymmetric Channels, the optimal encoding policy is given by choosing $\mathbb{A}_n$ such that

$$p_n(x_n = 0) = \begin{cases} \frac{1-q_{-\nu}(1+v_1)}{(1-q_\nu-q_{-\nu})(1+v_1)} & \text{if } q_{-\nu} \le q_\nu \\ \\ \frac{1-q_\nu(1+v_2)}{(1-q_\nu-q_{-\nu})(1+v_2)} & \text{if } q_{-\nu} > q_\nu, \end{cases} \tag{2.13}$$

where $v_1 \overset{def}{=} 2^{\frac{h(q_{-\nu})-h(q_\nu)}{1-q_\nu-q_{-\nu}}}$ and $v_2 \overset{def}{=} 2^{\frac{h(q_\nu)-h(q_{-\nu})}{1-q_\nu-q_{-\nu}}}$. Figure 2.6 is a typical example of the operator's bias h_n through a sequence of binary queries.

2.6 Application: channels with energy harvesting capabilities

Consider remotely estimating a source over a battery-operated channel with an energy harvesting device that deterministically charges the battery with k energy units each time step. Transmitting consumes energy and the channel state input determines how much energy to use. This, in turn, determines the statistics of the transmission channel. Also, assume that the decoder decides when and at what power to transmit. This occurs in practice when the encoder has no or limited processing capabilities.

For a specific example consider a UDS channel comprised of a BSC and the Markov chain structure shown in Figure 2.7 with the cross-over probabilities of the BSC determined by the chart in Figure 2.8. The objective function for this applica-

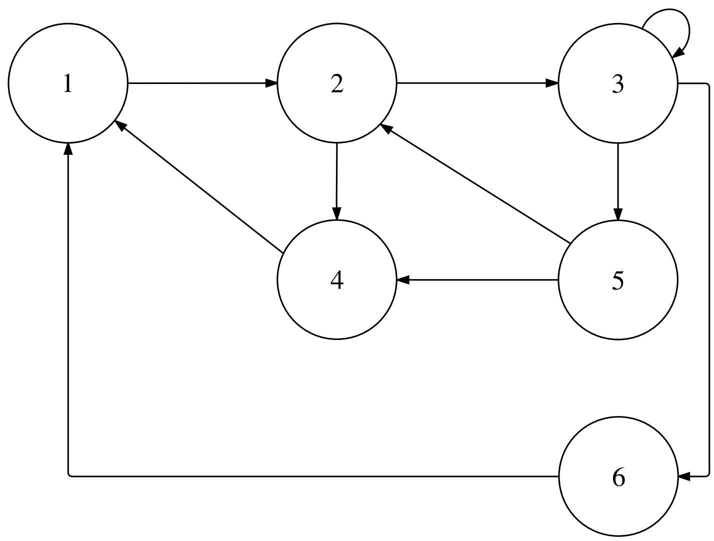

Figure 2.7: A Markov chain modeling the dynamics of a battery for an energy harvesting communication channel. The battery has a capacity of 2 energy units and the transmission policy determines when to charge the battery, state 1 and 2, when to transmit with 1 unit of energy, states 4 and 5, and when to transmit with 2 units of energy, state 6.

Channel State	BSC cross-over probability
1	.5
2	.5
3	.5
4	.25
5	.25
6	0

Figure 2.8: Each state of the Markov chain in Figure 2.7 determines the cross-over probability for the BSC as shown here. States 1, 2 and 3 correspond to charging the battery. States 4 and 5 correspond to transmitting with 1 unit of energy and state 6 to transmitting with 2 units of energy.

tion is the condition entropy $H(X^*|Y^N)$. From Theorem 1, minimizing $H(X^*|Y^N)$ is equivalent to designing the transmission policy to maximize

$$\sum_{i=1}^{N} C_i \ ,$$

where C_i is the capacity of the energy harvesting channel at time i. In [68], this objective function is called the throughput cost.

By Theorem 1, the jointly optimal transmission policy and encoder is found by solving the optimization problem in eq. (2.5) for the transmission policy that maximizes the throughput cost and constructing an encoding policy to induce the channel capacity achieving input distribution.

2.7 Conclusion

Use-Dependent Switching (UDS) channels are defined. UDS channels have two inputs: a channel state input and a transmission symbol input. The current channel state determines the which discrete channel is available for transmission at the current time step. For this class of channels, a partial separation of the design of transmission policies and encoding policies was shown for an entropy cost. For a UDS channel composed of a linearly saturating FSM and BSC, an optimal encoding policy and transmission policy was developed for the infinite horizon problem. The application of the optimal policy to problems in the areas of human operator support system design and communication over channels with energy harvesting capabilities is presented.

2.8 Proofs

Lemma 2 *[Single Step Entropy Reduction] Using the notation of Problem 1,*

$$\min_{\mathcal{E}_{n+1}} E_{Y_{n+1}|u_{n+1},p_n,\mathcal{E}_{n+1}}[H(p_{n+1})] = H(p_n) - C(u^{n+1}).$$

Furthermore, any encoder $\mathcal{E}_{n+1}$ that induces the input distribution that achieves the capacity of the channel $p(Y_{n+1}|X_{n+1}, s_{n+1}(u^{n+1}))$ is optimal.

Proof 5 (Lemma 2) *This proof is an extension of [64, Theorem 1] and follows the structure of that proof. For ease of notation the conditioning of the expectation will be dropped from the notation in the remainder of this proof. The actions u^{n+1} in the conditioning will also be dropped from the notation.*

First, the capacity of the channel at time $n+1$, $C(u^{n+1})$, will be written in a convenient way. Let f_i be the distribution of the output symbol y_{n+1} given the channel state and an input symbol i, $f_i \stackrel{def}{=} p(Y_{n+1}|X_{n+1} = i, s_{n+1}(u^{n+1}))$. The capacity of the discrete channel at time $n+1$ is

$$\begin{aligned} C(u^{n+1}) &= \sup_{p(X_{n+1})} I(X_{n+1}; Y_{n+1}|s_{n+1}(u^{n+1})) \\ &= \sup_{p(X_{n+1})} H\left(\sum f_i p(X_{n+1} = i)\right) \\ &\quad - \sum H(f_i)p(X_{n+1} = i). \end{aligned}$$

Rewrite

$$\begin{aligned} E[H(p_{n+1})] &= H(p_n) - I(X^*;\, Y_{n+1}|p_n,\, \mathcal{E}_{n+1}) \\ &= H(p_n) - H(Y_{N+1}|p_n,\, \mathcal{E}_{n+1}) \\ &\quad + H(Y_{N+1}|X^*,\, p_n,\, \mathcal{E}_{n+1}) \\ &= H(p_n) - H\left(\sum f_i p'(X_{n+1} = i)\right) \\ &\quad + \sum H(f_i) p'(X_{n+1} = i), \end{aligned}$$

where p' is the distribution of X_{n+1}. By choice of encoder $\mathcal{E}_{n+1}$, p' can be any distribution on $\mathbb{X}$. Minimizing over $\mathcal{E}_{n+1}$ is the same as minimizing over p'. Thus, $\min_{\mathcal{E}_{n+1}} E[H(p_{n+1})] = H(p_n) - C(u^{n+1})$.

Proof 6 (Lemma 1) *The proof is by induction. Equation (2.4) holds for N since*

$$\begin{aligned} V_N(u^N,\, p_N) &= \min_{u_{N+1}^N} H(p_N) - \sum_{i=N+1}^{N} C(u^i) \\ &= H(p_N). \end{aligned}$$

Now assume eq. (2.4) holds for $n+1$. It must be shown that it holds for n which is accomplished as follows.

$$\begin{aligned} V_n(u_n,\, p_n) &= \min_{\mathcal{E}_{n+1},\, u_{n+1}} E[V_{n+1}(u^{n+1},\, p_{n+1})] \\ &= \min_{\mathcal{E}_{n+1},\, u_{n+1}} E[\min_{u_{n+2}^N} H(p_{n+1}) - \sum_{i=n+2}^{N} C(u^i)] \\ &= \min_{\mathcal{E}_{n+1},\, u_{n+1}} E[H(p_{n+1})] - \max_{u_{n+2}^N} \sum_{i=n+2}^{N} C(u^i). \end{aligned}$$

where the expectation is with respect to $Y_{n+1}|u^{n+1},\, p_n,\, \mathcal{E}_{n+1}$. From Lemma 2, we have $\min_{\mathcal{E}_{n+1}} E[H(p_{n+1})] = H(p_n) - C(u^{n+1})$. The cost-to-go function now becomes

$$V_n(u^n,\, p_n) = H(p_n) - \max_{u_{n+1}^N} \sum_{i=n+1}^{N} C(u^i).$$

Concluding the proof.

Proof 7 (Theorem 1) *This is immediate. The optimal encoding strategy is shown in Lemma 2. The optimal transmission policy is seen from Lemma 1.*

Proof 8 (Theorem 2) *Channel state inputs u^N specify the states S^N. Let N_i be the number of times a transmission occurred with $q_n = \underline{\epsilon} + i\alpha$ (i.e. node i was visited in Figure 2.3), and N_β the number of times a transmission did not occur (i.e. a primed node is visited in Figure 2.3). Let $N_\alpha = \sum_{i=1}^{m-1} N_i$. Note $N_\alpha + N_\beta + N_m = N$.*

Using Lemma 1, NJ_N^* can be written

$$\begin{aligned} NJ_N^* &= \inf_{\mathcal{G}^N, \mathcal{K}^N} E\left[H(p_N)\right] \\ &= \inf_{\mathcal{K}^N} H(p_0) - \sum_{i=1}^{N} C(q_n) \\ &= \inf_{u^N} H(p_0) - N_0 C(\underline{\epsilon}) - N_1 C(\underline{\epsilon} + \alpha) - \dots \\ &\qquad -N_m C(\underline{\epsilon} + (m)\alpha) - N_\beta C(.5) \\ &= \inf_{\{N_i\} \in \mathbb{C}} H(p_0) - N_0 C(\underline{\epsilon}) - N_1 C(\underline{\epsilon} + \alpha) - \dots \\ &\qquad -N_m C(\underline{\epsilon} + (m)\alpha) \end{aligned} \tag{2.14}$$

Let $\mathbb{W}$ denote the set of all possible $\{N_i\}_{i=1}^m$, N_α, N_β configurations that can be generated by the dynamics in eq. (2.6) and eq. (2.7). In $\mathbb{W}$, N_α can be upper and lower bounded. This is accomplished by approximating N_α. It will be shown

$$\bar{w} + \frac{N - N_m}{2} \geq N_\alpha \geq \underline{w} + \frac{N - N_m}{2} \tag{2.15}$$

With this we can lower bound $NJ_N^* \geq H(p_0) - (\bar{w} + \frac{N-N_m}{2})C(\bar{\epsilon}) - N_m C(\bar{\epsilon})$. Since $N = N_\alpha + N_\beta + N_m$ and possible values for N_m are $0, 1, \dots N$ further lower bounds *can be obtained by minimizing with respect to* N_m. *If* $\frac{C(\underline{\epsilon})}{2} > C(\bar{\epsilon})$, *then* $N_m = 0$ *minimizes and the lower bound becomes*

$$NJ_N^* \geq H(p_0) - (\bar{w} + \frac{N}{2})C(\underline{\epsilon}) \tag{2.16}$$

If $\frac{C(\underline{\epsilon})}{2} \leq C(\bar{\epsilon})$, *then* $N_m = N$ *minimizes and the lower bound becomes*

$$NJ_N^* \geq H(p_0) - \bar{w}C(\underline{\epsilon}) - NC(\bar{\epsilon}) \tag{2.17}$$

The variables with a tilde are the quantities defined above that are generated by policy $(\mathcal{G}_1^N, \mathcal{K}_1^N)$. *Under the transmission policy in Definition 2, if* $\frac{C(\underline{\epsilon})}{2} > C(\bar{\epsilon})$, *transmission only occurs when* $q_n = \underline{\epsilon}$ *so* $\tilde{N}_0 = \tilde{N}_\alpha$ *and* $\tilde{N}_i = 0$ *for* $i = 1, \ldots, m$. *The relationship between* $\tilde{J}_N$ *and* $\tilde{N}_\alpha$ *is* $N\tilde{J}_N = H(p_0) - \tilde{N}_\alpha C(\underline{\epsilon})$. *From the inequality in eq.* (2.16), *we have*

$$\begin{aligned} NJ_N^* - N\tilde{J}_N &\geq \left(\tilde{N}_\alpha - (\bar{w} + \frac{N}{2})\right) C(\underline{\epsilon}) \\ &\geq (\underline{w} - \bar{w})\, C(\underline{\epsilon}). \end{aligned}$$

Using this lower bound and the fact that the optimal cost J_N^* *is less than* $\tilde{J}_N$, *we arrive at*

$$0 \geq J_N^* - \tilde{J}_N \geq \frac{1}{N}\left[(\underline{w} - \bar{w})\, C(\underline{\epsilon})\right]. \tag{2.18}$$

Under the transmission policy in Definition 2, if $C(\underline{\epsilon}) \leq 2C(\bar{\epsilon})$ *transmission always occurs so* $\tilde{N}_m \geq N - m$ *and* $\tilde{N}_i = 0$ *or* 1 *for* $i = 1, \ldots, m$ *depending on the initial condition. The relationship between* $\tilde{J}_N$ *and* $\tilde{N}_\alpha$ *is* $N\tilde{J}_N \geq H(p_0) - mC(\underline{\epsilon}) - NC(\bar{\epsilon})$. *From the inequality in eq.* (2.17) *we have*

$$NJ_N^* - N\tilde{J}_N \geq (m - \bar{w})\, C(\underline{\epsilon}).$$

Since $J_N^* \le \tilde{J}_N$, we arrive at

$$0 \ge J_N^* - \tilde{J}_N \ge \frac{1}{N}\,(m - \bar{w})\,C(\underline{\epsilon}). \tag{2.19}$$

Thus if the bounds on N_α hold, the proof is complete.

To prove the approximation of N_α, consider when transmission occurs and $q_n = \bar{\epsilon}$ (node m's self loop), namely $q_{n+1} = \bar{\epsilon}$. This transmission does not affect N_α which is the number of transmissions that cause a change in z_n. Not transmitting always affects z_n. Thus, z_N is the initial condition plus α times the number of transmissions affecting positive change minus α times the number of non-transmissions

$$z_N = z_0 + N_\alpha \alpha - N_\beta \alpha. \tag{2.20}$$

This together with the facts that $z_N \in [\underline{\epsilon}, \bar{\epsilon}]$, and $N = N_\alpha + N_\beta + N_m$ yield the following

$$\frac{\bar{\epsilon} - s_0}{2\alpha} + \frac{N - N_m}{2} \quad \ge N_\alpha \quad \ge \frac{\underline{\epsilon} - s_0}{2\alpha} + \frac{N - N_m}{2}.$$

Proof 9 (Theorem 3) *The value of the cost for policy $(\mathcal{G}_1, \mathcal{K}_1)$ is readily seen from the proof of Theorem 2. Below we will show the policy $(\mathcal{G}_1, \mathcal{K}_1)$ is optimal.*

To begin, note that the $\limsup$ *is becomes a limit because for any fixed $\mathcal{G}$ and $\mathcal{K}$,*

$$\frac{1}{N} H(X^*|Y^N) \ge \frac{1}{K} H(X^*|Y^K) \qquad K \ge N.$$

Since $J(\mathcal{G}_1, \mathcal{K}_1) < \infty$ we need only consider $(\mathcal{G}, \mathcal{K})$ pairs such that $J(\mathcal{G}, \mathcal{K}) < \infty$.

Consider the difference

$$\begin{aligned} J(\mathcal{G}_1, \mathcal{K}_1) - J(\mathcal{G}, \mathcal{K}) &= \lim_{N \to \infty} J_N(\mathcal{G}_1^N, \mathcal{K}_1^N) \\ &\qquad - J_N(\mathcal{G}^N, \mathcal{K}^N), \\ &\leq \lim_{N \to \infty} J(\mathcal{G}_1^N, \mathcal{K}_1^N) - J_N^*, \\ &\leq \lim_{N \to \infty} \frac{w}{N} = 0. \end{aligned}$$

The first inequality arises from $J_N^* \leq J_N(g^N, u^N)$ *and the second from Theorem 2.*

Chapter 3: Optimal Remote Estimation over Use-Dependent Packet-Drop Channels

In this chapter, a remote estimation problem over channels with the structure shown in Figure 3.1 is considered. A discrete-time remote estimation system formed by an encoder, a transmission policy, a channel, and a remote estimator is investigated. The encoder assesses a random process that the remote estimator seeks to estimate based on information sent to it by the encoder via the channel. The channel is affected by Bernoulli drops. The instantaneous probability of a drop is governed by a finite state machine (FSM). The state of the FSM is denoted as the channel state. At each time step, the encoder decides whether to attempt a transmission through the packet-drop link. The sequence of transmission decisions is the input to the FSM. This chapter seeks to design an encoder, transmission policy and remote estimator that minimize a finite-horizon mean squared error cost.

This chapter is organized as follows: Section 3.1 introduces the problem and has a literature review, Section 3.2 formulates the problem, Section 3.3 presents two structural results, Section 3.4 presents applications, Section 3.5 is the chapter conclusion, and Section 3.6 contains the proofs of the technical results.

The contributions of this chapter are listed below.

1. Assume that the process to be estimated is white and Gaussian, we show that there is an optimal transmission policy governed by a possibly asymmetric threshold on the estimation error.

2. The optimal symmetric transmission policies are characterized for the case when the measured process is the state of a scalar linear time-invariant plant driven by white Gaussian noise under assumptions on the drop probabilities.

3. A novel application to human operator support system designed is detailed.

3.1 Introduction

Encoders often select varying channel modes to enhance transmission performance in the presence of power and energy constraints. For example, in battery-operated wireless communication systems with energy harvesting, the decision of whether to attempt transmission must be made time and again at each time-step. The charge-level of the battery induces memory in the channel. We define a class of *use-dependent packet-drop channels* to model the effect of attempted transmissions on current and future performance, which in our case is quantified by the probability that an attempted transmission is dropped. The memory in use-dependent packet-drop channels is modeled by a finite state machine (FSM). The state of the FSM, or channel state, determines the instantaneous probability of drop. In our formulation the input to the FSM is the time-sequence of decisions of whether to attempt a transmission.

We consider a system formed by a remote estimator, a transmission policy, a use-dependent packet-drop channel and an encoder. The estimator produces an estimate of the state of a linear time-invariant plant that is accessible to the encoder. The estimate is based on information transmitted from the encoder to the estimator via the channel. The encoder and transmission policy also have access to past transmission decisions and channel feedback on the realization of current and past drops. The encoder determines what to transmit over the channel and the transmission policy determines when to attempt a transmission. The goal of this chapter is to investigate encoders, transmission policies and remote estimators that jointly minimize the mean squared state estimation error over a finite time-horizon.

3.1.1 Outline of the main results

The following are our two main results characterizing the structure of optimal transmission policies for our problem. In both results, no restrictions are placed on the dynamics or size of the FSM.

In the first result, we assume that the process to be estimated is white and Gaussian. We show that the optimal transmission policy is of the threshold type, meaning that the encoder chooses to attempt transmission when the process takes values outside a certain interval $[\underline{\tau}, \bar{\tau}]$. The characteristics of the use-dependent packet-drop channel determine the values of $\bar{\tau}$ and $\underline{\tau}$. In general, $\bar{\tau}$ may not equal $-\underline{\tau}$, even when the process is zero-mean.

In the second result, the process to be estimated is the state of a scalar linear time-invariant plant driven by white Gaussian noise, for which we seek to obtain

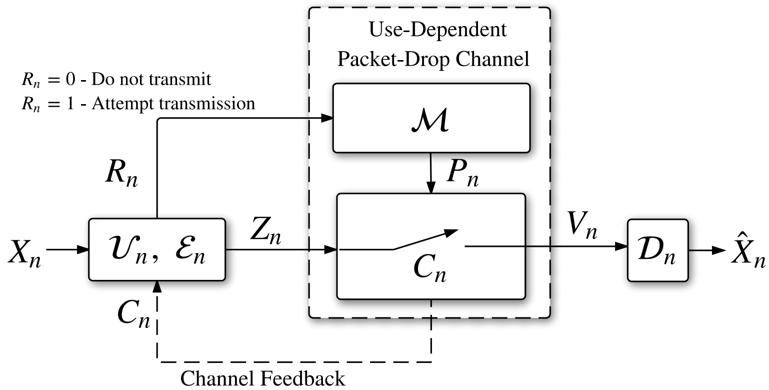

Figure 3.1: The problem under investigation is a remote estimation problem over a packet-drop channel, whose probability of drop P_n is governed by the Finite State Machine $\mathcal{M}$.

an optimal symmetric transmission policy. We show that if the channel performs satisfactorily in all channel states, then there exists at least one symmetric threshold that, when applied to the estimation error, leads to a transmission policy that is optimal among all symmetric strategies.

3.1.2 Related literature

In [74] and [75], an estimation problem over a packet drop channel with communication costs is considered. In contrast to [74] and [75], here we introduce a channel state and do not consider explicit communication costs in the objective function. In our formulation, the channel state, which depends on current and past transmission decisions, and its impact on channel performance create an implicit communication cost. For example, in the energy harvesting application discussed in Section 3.4, there is no explicit cost for attempting a transmission. However, attempting a transmission reduces the energy available for future transmissions, which causes performance degradation that can be viewed as an implicit cost for attempting a transmission.

Considering costly measurements (or transmissions) in estimation and control problems has a long history and has been modeled in many ways. In [76], one of several possible measurements with different observation costs is selected to minimize a combination of error and observation cost. In [77], a subset of the measurements is selected in order to minimize the log-determinant of the error covariance. In [78], the arrival of observations is a random process and the convergence of the error covariance is studied. In [79], the task is to locate a mobile agent and the observation cost is the expected number of observations that must be made to do so.

In [65], the capacity of channels with action-dependent states is studied. Although our problem formulation is similar to that of [65] in motivation, it differs in several accounts. In contrast to [65], we consider finite time horizons, a mean-squared error cost and a new class of packet-drop channels.

3.2 Problem formulation

3.2.1 Notation

We use calligraphic font ($\mathcal{F}$) to denote deterministic functions, capital letters (X) to represent random variables and lower case letters (x) to represent realizations of the random variables. Let $\mathcal{N}(0, \sigma^2)$ denote the Gaussian distribution with zero mean and variance σ^2. We use Q^n to denote the finite sequence $\{Q_1, Q_2, \ldots Q_n\}$.

The real line is denoted with $\mathbb{R}$ and a subset of $\mathbb{R}$ is denoted with double barred font, such as $\mathbb{A}$. The indicator function of a set $\mathbb{A}$ is defined as

$$\mathbf{1}_{\mathbb{A}}(x) \stackrel{def}{=} \begin{cases} 1 & x \in \mathbb{A}, \\ 0 & \text{Otherwise.} \end{cases}$$

The expectation operator is denoted with $E[\cdot]$. By $\lim_{\delta \downarrow 0} \mathcal{F}(\delta)$ we mean the limit of $\mathcal{F}(x)$ at 0 from the right.

3.2.2 Problem formulation

Consider the following scalar linear time-invariant system

$$X_{n+1} = aX_n + W_n, \quad n \geq 0, \quad X_0 = x_0,$$

where X_n is the state, a is a real constant, W_n is independent and identically distributed Gaussian noise with zero mean and variance σ^2. The initial state $x_0 \in \mathbb{R}$ is known.

Observations are made by the encoder and transmitted to the remote estimator over a use-dependent packet-drop channel, which is defined below. In Figure 3.1, the dotted box represents the use-dependent packet-drop channel.

Definition 4 (Use-dependent packet-drop channels) *Let* $\mathcal{M}^s : \mathbb{Q} \times \{0,1\} \rightarrow \mathbb{Q}$ *and* $\mathcal{M}^o : \mathbb{Q} \rightarrow [0,1]$ *be given, where* $\mathbb{Q} = \{1, \dots, m\}$ *represents the state space for the finite state machine (FSM). The channel inputs are* Z_n *and* R_n*, which take values in* $\mathbb{R}$ *and* $\{0,1\}$*, respectively. In this model* Z_n *represents the information to be transmitted, while the decision to attempt a transmission (or* *not) is represented by* $R_n = 1$ *(*$R_n = 0$*). The channel output* V_n *takes values in* $\mathbb{R} \cup \mathfrak{E}$ *and is determined as follows*

$$V_n = \begin{cases} Z_n & \text{if } L_n = 1 \\ \mathfrak{E} & \text{if } L_n = 0, \end{cases}$$

where $L_n \stackrel{def}{=} R_n C_n$*. Here,* C_n *is a Bernoulli process characterized by* $p(C_n = 0) = \mathcal{M}^o(Q_n)$*, where* Q_n *is the state of the FSM updated by*

$$Q_{n+1} = \mathcal{M}^s(Q_n, R_n).$$

The FSM's initial state $q_1 \in \mathbb{Q}$ *is known. Here,* $\mathcal{M}^s$ *and* $\mathcal{M}^o$ *model the effect of the input on the transitions among channel states and the probability of drop as a function of the channel state, respectively.*

At time n, the transmission policy $\mathcal{U}_n : \mathbb{R}^n \times \{0,1\}^{n-1} \to \{0,1\}$ determines whether a transmission is attempted,

$$R_n = \mathcal{U}_n(X^n, C^{n-1}),$$

based on the plant history X^n and drop history C^{n-1}. The remote estimator $\mathcal{D}_n : \mathbb{R}^n \times \{0,1\}^n \to \mathbb{R}$ produces the state estimate,

$$\hat{X}_n = \mathcal{D}_n(V^n, R^n),$$

based on the channel output history V^n and the transmission history R^n. The encoder $\mathcal{E}_n : \mathbb{R}^n \times \{0,1\}^{n-1} \to \mathbb{R}$ determines what is transmitted,

$$Z_n = \mathcal{E}_n(X^n, C^{n-1}),$$

based on the plant history X^n and drop history C^{n-1}.

We seek to solve the following problem.

Problem 3 *For finite N, solve*

$$\min_{\mathcal{U}^N,\mathcal{E}^N,\mathcal{D}^N} \sum_{n=1}^{N} E\left[(X_n - \hat{X}_n)^2\right].$$

Remark 1 *For any encoder and transmission policy, the optimal remote estimator is the conditional mean, $\mathcal{D}_n(V^n, R^n) = E[X_n|V^n, R^n]$. Also, an optimal encoder policy transmits only the current state, $\mathcal{E}_n(X^n, C^{n-1}) = X_n$. This is evident from the Markov nature of X_n and the information already available to the remote estimator. The channel drops can be calculated from (V^{n-1}, R^{n-1}); thus, the only new information to send the remote estimator is X_n.*

Because of Remark 1, Problem 3 is equivalent to the following problem.

Problem 4 (Main Problem) *For finite N, solve*

$$\min_{\mathcal{U}^N} \sum_{n=1}^{N} E\left[(X_n - \hat{X}_n)^2\right],$$

where the optimal encoder, $\mathcal{E}_n(X^n, C^{n-1}) = X_n$, and optimal remote estimator, $\mathcal{D}_n(V^n, R^n) = E[X_n|V^n, R^n]$, are used.

3.3 Structural results

In this section, we present our technical results. We begin by defining threshold transmission policies.

3.3.1 Definitions

Estimation error is denoted as $E_n \stackrel{def}{=} X_n - \hat{X}_n$.

Definition 5 *A function $\mathcal{G} : \mathbb{R} \to [0, 1]$ is a threshold function if there are constants $\underline{\tau}$ and $\bar{\tau}$, such that:*

$$\mathcal{G}(e) = \begin{cases} 1 & \text{if } \underline{\tau} \leq e \leq \bar{\tau} \\ \\ 0 & \text{Otherwise.} \end{cases}$$

Definition 6 *A function $\mathcal{G} : \mathbb{R} \to [0, 1]$ is a symmetric threshold function if there is a constant τ, such that:*

$$\mathcal{G}(e) = \begin{cases} 1 & \text{if } |e| \leq \tau \\ \\ 0 & \text{Otherwise.} \end{cases}$$

Definition 7 *A transmission policy $\mathcal{T}^N$ is a threshold policy if the decision to transmit depends only on the current error and channel state (e_n, q_n) in the following manner*

$$\mathcal{T}_n(x^n, c^{n-1}) = \begin{cases} 1 & \text{if } \underline{\tau}_n(q_n) \leq e_n \leq \bar{\tau}_n(q_n) \\ \\ 0 & \text{Otherwise,} \end{cases}$$

for some $\underline{\tau}_n(q_n)$, $\bar{\tau}_n(q_n) \in \mathbb{R}$.

Notice that the current error and channel state (e_n, q_n) are calculated from the history $(x^n,\ c^{n-1})$ and previous policies $\mathcal{T}^{n-1}$.

Definition 8 *A transmission policy $\mathcal{T}^N$ is a symmetric threshold policy if the decision to transmit depends only on the current error and channel state (e_n, q_n) in the following manner*

$$\mathcal{T}_n(x^n, c^{n-1}) = \begin{cases} 1 & \text{if } |e_n| \leq \tau_n(q_n) \\ 0 & \text{Otherwise,} \end{cases}$$

for some $\tau_n(q_n) \in \mathbb{R}$.

3.3.2 Optimal transmission policies are threshold when the process is white and Gaussian ($a = 0$)

To investigate the structure of solutions to Problem 4, we start with the case when $a = 0$. The system state becomes

$$X_n = W_n.$$

The estimation error is independent at each step, thus there are optimal transmission policies that only depend on the current error and channel state.

With $a = 0$, we reformulate Problem 4 as a dynamic program to show that there are optimal transmission policies of the threshold type, which may not be symmetric. An optimal transmission policy that is not symmetric in the estimation error is surprising since the objective function is symmetric in the error and the random process is zero-mean and symmetric.

We utilize the results in [80]. In [80], a single stage estimation problem over a collision channel with two transmitters is studied. If both transmit then the remote estimator receives a collision symbol and if neither transmits a no-transmission symbol is received. The result in [80] states that the optimal policy for each transmitter is of the threshold type.

Remark 2 *At each time step the remote estimator can distinguish three cases: a transmission is attempted but dropped* ($V_n = \emptyset, R_n = 1$)*, no transmission is attempted* ($V_n = \emptyset, R_n = 0$) *and a successful transmission* ($V_n = X_n, R_n = 1$)*. The remote estimator can glean* 1*-bit of information when* $V_n = \emptyset$ *by distinguishing between an attempt* $R_n = 1$ *and a non-attempt* $R_n = 0$*. This ability to transmit information when* $V_n = \emptyset$ *causes the optimal policies to be asymmetric [80].*

Remark 3 *In Problem 4 with* $a \neq 0$*, the remote estimator's ability to distinguish a dropped transmission attempt and a non-attempt leads to belief densities of* X_n *given* V^n, R^n *that are possibly asymmetric and multi-modal. This greatly complicates the problem when* $a \neq 0$*.*

Problem 4 is a sequential problem; distinguishing it from [80], which is a static problem. Notice that our problem cannot be converted into a sequence of static problems because the transmission policies depend on the channel memory.

Following [80], the stage cost at time n can be written as

$$\begin{aligned} E[(X_n - \hat{X}_n)^2] =& E[(X_n - \hat{X}_n)^2 | L_n = 0] p(L_n = 0) \\ =& E[(X_n - \hat{X}^0_n)^2 | R_n = 0] p(R_n = 0) \\ &+ p_n E[(X_n - \hat{X}^1_n)^2 | R_n = 1] p(R_n = 1) \end{aligned} \tag{3.1}$$

where $p_n \overset{def}{=} \mathcal{M}^o(q_n)$, $\hat{X}^0_n \overset{def}{=} E[X_n | R_n = 0]$ and $\hat{X}^1_n \overset{def}{=} E[X_n | R_n = 1]$.

Proposition 1 *The stage cost at time n is a function of the current channel state Q_n and transmission policy $\mathcal{U}_n$.*

Proof 10 *See proof 17 in Section 3.6.*

With $a = 0$, Problem 4 can be written as a Markov chain with $\mathcal{U}_n$ as the input, (X_n, C_n) as the noise, (Q_n, X^{n-1}, C^{n-1}) as the state, and $E[(X_n - \hat{X}_n)^2]$ as the stage cost. Note the input is not r_n, the decision to transmit, as may have been expected. The transmission policy $\mathcal{U}_n$ is taken as the input because the density $f_{X_n|R_n}$ depends on the entire policy $\mathcal{U}_n$: not just the specific decision r_n.

Using Proposition 1 and the independence of the system states over time, without loss of performance, we consider only transmission policies that are functions of the current system state and channel state, $\mathcal{U}_n(X_n, Q_n)$. Consequently, the Markov decision process can be simplified with $\mathcal{U}_n$ as the input, X_n as the noise, Q_n as the state, and $E[(X_n - \hat{X}_n)^2]$ as the stage cost. The associated dynamic programming recursion is shown in eq. (3.2) and eq. (3.3) on the next page.

Theorem 4 *Let X_n be independent and identically distributed $\mathcal{N}(0, \sigma^2)$. The optimal transmission policy for Problem 4 is of the threshold type.*

Proof 11 *See proof 18 in Section 3.6.*

3.3.3 Optimal solutions within the class of symmetric policies

We now investigate the structure of the best symmetric transmission policies. We seek conditions under which the optimal symmetric transmission policy is a

$$V_{N+1}(q_{N+1}) = \qquad 0, \tag{3.2}$$

$$\begin{aligned} V_n(q_n) = & \min_{\mathcal{U}_n} E[(X_n - \hat{X}_n^0)^2 | R_n = 0] p(R_n = 0 | q_n) + \\ & p_n E[(X_n - \hat{X}_n^1)^2 | R_n = 1] p(R_n = 1 | q_n) + \\ & E[V_{n+1}(q_{n+1}) | q_n], \quad n \in \{1, \ldots, N\}, \end{aligned} \tag{3.3}$$

$$\begin{aligned} \frac{V_n(a(c+\delta)+w,q) - V_n(ac+w,q)}{(c+\delta)^2 - c^2} \leq & \frac{(a(c+\delta)+w)^2 - (ac+w)^2}{(c+\delta)^2 - c^2} + \\ & \frac{h_{n+1}^{q^1}(c+\delta+w/a) - h_{n+1}^{q^1}(c+w/a)}{(c+\delta)^2 - c^2}, \quad (3.4) \\ \leq & 2a^2 + \frac{a\omega}{x} + \\ & \frac{h_{n+1}^{q^1}(c+\delta+w/a) - h_{n+1}^{q^1}(c+w/a)}{(c+\delta)^2 - c^2}. \quad (3.5) \end{aligned}$$

symmetric threshold policy. This is the case if the probability of drop is sufficiently small for all channel states. Even if the drop probabilities are not sufficiently small, symmetric threshold policies may still be optimal. This is highlighted by a numerical example, which suggests that there are classes of channel dynamics for which symmetric threshold policies are the best symmetric transmission policies.

Restricting to symmetric transmission policies, Problem 4 can be written as a dynamic program. We first show that the cost-to-go functions are quasi-convex. In order to accomplish this, we write the evolution of the error in a convenient manner.

Lemma 3 *If $\mathcal{U}^N$ is a symmetric transmission policy, then the estimation error* *evolves according to*

$$E_{n+1} = \begin{cases} aE_n + W_n & \text{if } L_{n+1} = 0, \\ 0 & \text{if } L_{n+1} = 1. \end{cases} \tag{3.6}$$

Proof 12 *This is in principle equivalent to [74, Proposition 3.1]. In [74, Proposition 3.1], a symmetric threshold policy is assumed; however, the proof only relies on the symmetric nature of the policy.*

The convenient form of the error evolution in eq. (3.6) is possible due to the symmetric assumption. For symmetric policies, when $L_n = 0$ the optimal estimate $\hat{X}_n$ is the same whether a transmission was attempted or not. The remote estimator's belief $f_{X_n|V^n,R^n}$ depends on the value of R_n; however, its mean, which is the optimal estimate, does not.

The problem can be considered a Markov decision process with state (E_{n-1}, Q_n), input R_n, and noise (W_{n-1}, C_n). The cost to be minimized is

$$\sum_{n=1}^{N} E[E_n^2].$$

The associated dynamic programming recursion is given by

$$V_{N+1}(e_N, q_{N+1}) = e_N^2,$$

$$V_n(e_{n-1}, q_n) = \min\{C_n^0(e_{n-1}, q_n), C_n^1(e_{n-1}, q_n)\}, \tag{3.7}$$

for $n = 1, \ldots N$ with

$$\begin{aligned} C_n^0(e, q) &\stackrel{def}{=} e^2 + E_W[V_{n+1}(ae + W, q^0)], \\ C_n^1(e, q) &\stackrel{def}{=} p_q e^2 + p_q E_W[V_{n+1}(ae + W, q^1)] \\ &\quad + (1 - p_q) E_W[V_{n+1}(W, q^1)], \end{aligned}$$

and $q^0 \stackrel{def}{=} \mathcal{M}^s(q,0)$, $q^1 \stackrel{def}{=} \mathcal{M}^s(q,1)$, $p_q \stackrel{def}{=} \mathcal{M}^o(q)$ and W distributed $\mathcal{N}(0,\sigma^2)$.

Lemma 4 *For $n \in \{1,\ldots,N+1\}$ and $q \in \mathbb{Q}$, the cost-to-go functions $V_n(e_{n-1},q)$ are quasi-convex and symmetric in e_{n-1}. The minimum value is $V_n(0,q)$.*

Proof 13 *See proof 19 in Section 3.6.*

Theorem 5 *There exists a $v > 0$ such that if for all $q \in \mathbb{Q}$*

$$p_q < \frac{1}{1+v},$$

then the optimal symmetric transmission policy is a threshold policy.

Proof 14 *See proof 20 in Section 3.6.*

Several lemmata will be presented to aid in the proof of this theorem. Let $h_n^q(e) \stackrel{def}{=} E_W[V_n(ae+W,q)]$. Also define,

$$\mathcal{O}_n(e,q) \stackrel{def}{=} \lim_{\delta \downarrow 0} \frac{h_n^q(e+\delta) - h_n^q(e)}{(e+\delta)^2 - e^2}.$$

Lemma 5 *For $e \geq 0$ and $q \in \mathbb{Q}$, if*

$$p_q \mathcal{O}_{n+1}(e,q^1) < (1-p_q) + \mathcal{O}_{n+1}(e,q^0), \tag{3.8}$$

then the optimal symmetric transmission policy for stage n is a threshold policy.

Proof 15 *See proof 21 in Section 3.6.*

Remark 4 *The condition in Lemma 5, guarantees that C_n^0 increases more than C_n^1 at every estimation error e. Clearly, this is a condition that leads to threshold transmission policies.*

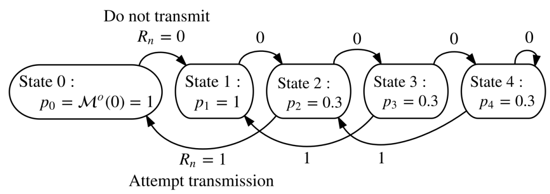

Figure 3.2: FSM model for an energy harvesting channel. State i represents the energy currently stored in the battery. The arcs represent channel state transitions which depend on whether a transmission is attempted.

Lemma 6 *For all $e \in \mathbb{R}$ and $q \in \mathbb{Q}$,*

$$\mathcal{O}_n(e, q) \leq v'_n,$$

with $v'_n \stackrel{def}{=} 2a^2(N + 1 - n) + a^2$.

Proof 16 *See proof 22 in Section 3.6.*

3.4 Applications

3.4.1 Energy harvesting channel

In this section, a wireless communication channel with energy harvesting capabilities is modeled as a use-dependent packet-drop channel. Many different problem formulations addressing remote estimation over a battery powered channel have been considered: see [66], [81] and [67].

Consider a battery operated channel with a capacity of 4 energy units. Assume energy is harvested deterministically, as in [81], at 1 energy unit per time

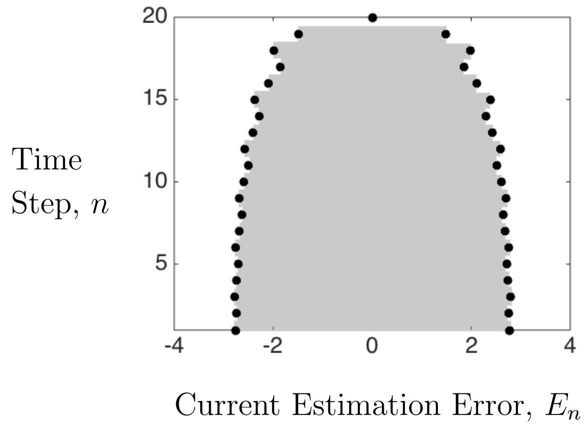

Figure 3.3: Optimal symmetric transmission policy while in channel state 2 of the use-dependent packet-drop channel as shown in Figure 3.2. This transmission policy was calculated using the values $a = 1.1$, $\sigma = 1$, and $N = 20$. For errors less than the left black dots a transmission is attempted. For errors greater than the right black dots a transmission is attempted. Inside the gray region, no transmission is sent.

step. Transmitting requires 2 units of energy and no energy is harvested during transmission. At each time step, the decision of whether to transmit is made.

The FSM in Figure 3.2 models the battery dynamics. The channel states are $\mathbb{Q} = \{0, 1, 2, 3, 4\}$. Channel state q denotes that the battery has q energy units. If a transmission is attempted $R_n = 1$, then the battery level is reduced by 2 energy units. Thus the channel state state q transitions to state $q - 2$. If a transmission is not attempted $R_n = 0$, then the battery level increases by 1 as long as the battery is not already at capacity. Thus, the channel transitions from state q to state $\min\{q+1, 4\}$. In states 0 and 1, transmitting is not allowed due to insufficient energy. The probability of drop for each state capable of transmitting is 0.3.

We assume that the encoder receives acknowledgments of the transmissions and that the remote estimator can distinguish between a drop and no transmission attempt. Interestingly, from Theorem 4 we have that the optimal transmission policy may not be symmetric in the estimation error even though the cost is symmetric in the estimation error and the noise is zero-mean and symmetric.

3.4.1.1 Numerical example

We numerically calculated, via discretization and value-iteration, the optimal symmetric transmission policies for this example when $a = 1.1$, $\sigma = 1$ and $N = 20$. The optimal symmetric transmission policy for channel state 2 is shown in Figure 3.6.

Notice that the optimal symmetric transmission policy is a threshold policy, even though the conditions of Theorem 5 are not satisfied. In fact, every p_2, p_3, $p_4 \in [0, 1]$ that we tested has an optimal symmetric transmission policy that is a threshold policy. This suggests that for the channel dynamics of Figure 3.2, threshold transmission policies are optimal among all symmetric strategies.

3.4.2 Operator task shedding

In this section, we seek to optimize a decision support system for human operators tracking a dynamic target.

Consider a human operator managing multiple UAVs. Tracking a dynamic target is one of the operator's many tasks. A video feed is presented to the operator (see Figure 3.4 for an example of the video feed). The white region is drawn on the video feed by the decision support system. The operator's task is to indicate

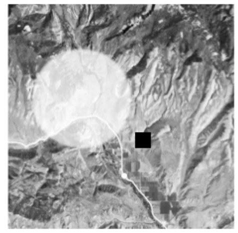

Figure 3.4: Example of a display presented to an operator for the task shedding application. The target is the black square. The visual search task consists of the operator identifying if the target is inside the white region and optionally logging its location if it is outside the region. We seek to design the white regions dynamically to help the operator manage their time.

if the target is inside this region. If outside the region the operator is requested to log the target's current location; however, the operator is allowed to not log the target's location if other tasks seem more vital. In [82], experiments are performed in a similar setting.

We seek to dynamically optimize the white regions in order to help the operator manage their time appropriately. If the regions are large, the target's location is not well known. If the regions are small, then the target's location is frequently requested. This increases the operator's workload and the likelihood the operator will ignore the request. The channel state is used to model operator workload. The optimal transmission policies define the optimal white regions and manage the tradeoff between accuracy and workload.

Yerkes-Dodson's law quantifies the tradeoff between operator performance and workload, see [13]. Yerkes-Dodson's law states that the operator performs poorly if

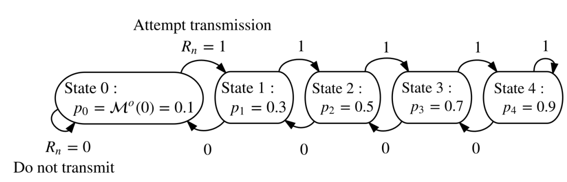

Figure 3.5: FSM model for human operator workload. The workload is a function of the average number of requests over the last 4 time steps. State i represents that i requests have occurred in that last 4 time steps. The arcs represent transitions of the channel state which depend on whether a transmission is requested.

the workload is very high or very low. Optimizing operator decision support systems using Yerkes-Dodson's law as an operator model is also investigated in [11] and [12]. In [11], the workload impacts the time to complete tasks such that under high workload situations the operator completes tasks slowly. The authors find optimal policies specifying when to present the operator with tasks in order to maximize throughput. In [12], not all tasks must be completed and the questions of which tasks to assign, for how long, and with how much rest in-between are addressed.

In contrast to [11] and [12] and motivated by [16], we assume that the operator workload impacts the likelihood that the operator will ignore a request for information.

We consider the operator's workload a function of the average number of requests over the last k time steps,

$$\frac{1}{k}\sum_{i=n-k}^{n} r_i.$$

If the average is high, the operator is prone to shed tasks. This workload model has memory and can be envisioned as the finite state machine in Figure 3.5. State q represents q requests occurring in the last k steps.

To formulate this as a use-dependent packet-drop channel we take the target's location to be the system state, X_n. The target being outside the white region represents an attempted transmission $R_n = 1$. The transmission policy $\mathcal{U}_n$ defines the white region.

We have modeled this application as a use-dependent packet-drop channel. By Theorem 5, if the operator is unlikely to ignore requests, $p_n < 1/(1+v)$, then the optimal symmetric white regions are threshold policies. This is desirable since non threshold policies represent white regions that are not connected and may mislead operators.

The numerical example below suggests that threshold policies are the best symmetric policies even if the operator is likely to ignore requests. This is due to the structure of the channel dynamics.

Note in this example X_n is two dimensional; however, in our formulation X_n is scalar. For this two dimensional example, we assume independence between the horizontal and vertical directions. Thus, Theorem 4 and Theorem 5 apply to each direction.

3.4.2.1 Numerical example

We numerically find optimal symmetric transmission policies for this example when $a = 1.1$, $\sigma = 1$ and $N = 20$. The channel dynamics and drop probabilities are

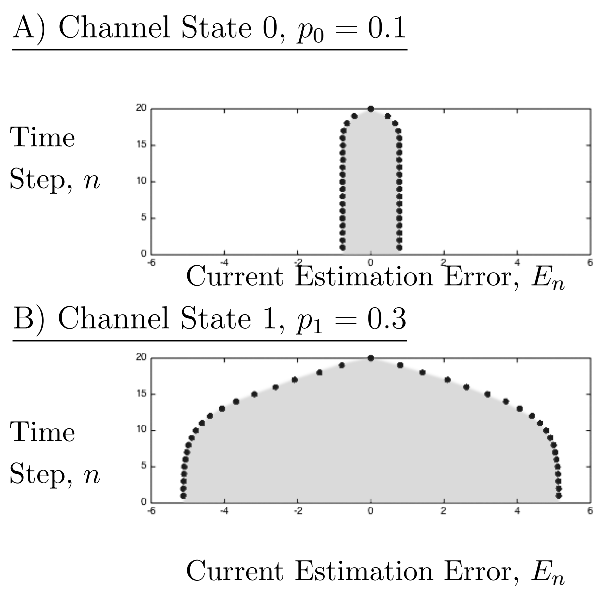

Figure 3.6: Optimal symmetric policies for the use-dependent packet-drop channel with dynamics as shown in Figure 3.5. This transmission policy was calculated using the values $a = 1.1$, $\sigma = 1$, and $N = 20$. Part A of the figure plots the transmission policy for channel state 0. Part B plots the policy for channel state 1. For errors less than the left black dots a transmission is attempted. For errors greater than the right black dots a transmission is attempted. Inside the gray region, no transmission is sent.

shown in Figure 3.5. The optimal symmetric transmission policies are calculated by approximating the value functions in eq. (3.7). In Figure 3.6, the optimal policies for channel sstates 0 and 1 are shown. It can be seen that the policies are symmetric. In fact, for all drop probabilities p_0, p_1, p_2, p_3, $p_4 \in [0, 1]$ that were simulated, the optimal transmission policies were threshold policies.

3.5 Conclusion

We investigated optimal transmission policies for a remote estimation problem over a use-dependent packet-drop channel. We presented structural results for the optimal transmission policies under two different assumptions. Also, two example applications were presented with numerical calculations. An application to energy harvesting channels and an application to mixed initiative teams with human operator's performing visual search tasks were discussed.

3.6 Proofs and background on quasi-convex functions

Proof 17 (Proposition 1) *From eq. (3.1), note that $E[(X_n - \hat{X}_n)^2]$ is a deterministic function of the channel state q_n, the probability that $R_n = 1$ and the density $f_{X_n|R_n}$. This density can be written as*

$$f_{X_n|R_n}(x_n|r_n) = \frac{p_{R_n|X_n}(r_n|x_n) f_{X_n}(x_n)}{p_{R_n}(r_n)},$$

where $p_{R_n|X_n}(r_n|x_n) \stackrel{def}{=} p(R_n = r_n|X_n = x_n)$ and $p_{R_n}(r_n) \stackrel{def}{=} p(R_n = r_n)$. Thus, eq. (3.1) is a function of q_n and the probability mass function $p_{R_n|X_n=x_n}$. The transmission policy $\mathcal{U}_n$ determines the distribution $p_{R_n|X_n=x_n}$. Therefore, the stage cost is a function of only q_n and $\mathcal{U}_n$.

Proof 18 (Theorem 4) *For an arbitrary transmission policy $\mathcal{U}^N$, we construct a threshold transmission policy $\mathcal{T}^N$, using [80], which outperforms the policy $\mathcal{U}^N$. Note, all quantities associated with the policy $\mathcal{T}^N$ have a superscript $\mathcal{T}$. Also, all quantities associated with policy $\mathcal{U}^N$ have a superscript $\mathcal{U}$.*

We expand our search for a policy $\mathcal{T}^N$ to include randomized transmission policies. For $n \in \{1, \ldots, N\}$ and $q \in \mathbb{Q}$, let $\mathcal{T}_n^q : \mathbb{R} \to [0, 1]$ be the probability of transmitting, $\mathcal{T}_n^q(x) \overset{def}{=} p^{\mathcal{T}}(R_n = 1|X_n = x, Q_n = q)$. Also, $E[\mathcal{T}_n^q(X_n)] = p^{\mathcal{T}}(R_n = 1|Q_n = q)$.

For a specific n, consider a policy $\mathcal{T}_n$ that matches the policy $\mathcal{U}_n$'s probability of transmitting,

$$p^{\mathcal{T}}(R_n = 1|Q_n = q_n) = p^{\mathcal{U}}(R_n = 1|Q_n = q_n). \tag{3.9}$$

Also, let policy $\mathcal{T}_n$ be such that it produces estimates that match those of policy $\mathcal{U}_n$,

$$^{\mathcal{T}}\hat{X}_n^0 = {}^{\mathcal{U}}\hat{X}_n^0 \tag{3.10}$$

$$^{\mathcal{T}}\hat{X}_n^1 = {}^{\mathcal{U}}\hat{X}_n^1. \tag{3.11}$$

Since $p^{\mathcal{T}}(R_n = 1|Q_n = q) = p^{\mathcal{U}}(R_n = 1|Q_n = q)$, we have $p^{\mathcal{T}}(Q_{n+1}|Q_n = q) = p^{\mathcal{U}}(Q_{n+1}|Q_n = q)$. All the quantities in eq. (3.3) are the same for both policies with the exception of $E[(X_n - \hat{X}_n^i)^2|R_n = i]$, for $i = 1, 2$. We will choose $\mathcal{T}_n^q$ to reduce $E[(X_n - \hat{X}_n^i)^2|R_n = i]$, for $i = 1, 2$.

In [80], minimizing $E[(X_n - \hat{X}_n^i)^2|R_n = i]$ for $i = 1, 2$ subject to the constraints eq. (3.9), eq. (3.10) and eq. (3.11) was cleverly rewritten as a constrained moment matching problem. It was shown that the optimal $\mathcal{T}_n^q$ was a threshold function of X_n. Using this result, we have constructed a threshold policy $\mathcal{T}_n^q$ that outperforms $\mathcal{U}_n^q$.

Thus, for every $q \in \mathbb{Q}$ and $n \in \{1, \ldots, N\}$, we can construct a threshold policy $\mathcal{T}_n^q$ that out forms $\mathcal{U}_n^q$. This threshold policy $\mathcal{T}^N$ outperforms $\mathcal{U}^N$.

Proof 19 (Lemma 4) *We show that $V_n(e_{n-1}, q)$ is a symmetric and non-decreasing function in $|e_{n-1}|$. This implies $V_n(e_{n-1}, q)$ is quasi-convex by Lemma 7. The proof* *is by induction. The claim holds for the initial case, $V_{N+1}(e_N, q_{N+1}) = e_N^2$. Assume $V_{n+1}(e_n, q_{n+1})$ is symmetric and non-decreasing in $|e_n|$. $V_n(e_{n-1}, q_n)$ is the minimum between $C_n^0(e_{n-1}, q_n)$ and $C_n^1(e_{n-1}, q_n)$. By Lemma 9, $E_W[V_{n+1}(ae_{n-1} + W, q_n^i)]$ is symmetric and non-decreasing in $|e_{n-1}|$ for $i = 0, 1$. $C_n^0(e_{n-1}, q_n)$ and $C_n^1(e_{n-1}, q_n)$ are symmetric and non-decreasing in $|e_{n-1}|$ because they are the sum of two such functions. Thus by Lemma 8, $V_n(e_{n-1}, q_n)$ is symmetric and non-decreasing in $|e_{n-1}|$.*

Proof 20 (Theorem 5) *Using the bound $v = v_1'$ from Lemma 6, we proceed by contradiction. We show that any non-threshold, symmetric transmission policy violates the assumption $p_q < \frac{1}{1+v}$.*

Following identical arguments as in Lemma 5, we have from eq. (3.12)

$$\begin{aligned} p_q \mathcal{O}_n(e, q^1) &\geq (1 - p_q) + \mathcal{O}_n(e, q^0) \\ &\geq 1 - p_q, \end{aligned}$$

since $\mathcal{O}_n(e, q) \geq 0$ by Lemma 4. Rearanging and using the bound on $\mathcal{O}_n(e, q^1)$ gives

$$p_q \geq \frac{1}{1 + \mathcal{O}_n(e, q^1)} \geq \frac{1}{1 + v}.$$

Contradicting the assumption. Thus, the optimal policy is a threshold policy.

Proof 21 (Lemma 5) *We show that if eq. (3.8) holds, any non-threshold, symmetric policy is not the optimal symmetric transmission policy.*

For a non-threshold, symmetric policy $\mathcal{S}_n$ there exists a $q \in \mathbb{Q}$ and $c \geq 0$ such that $\mathcal{S}_n(c, q) = 1$ but $\mathcal{S}_n(c + \delta, q) = 0$ for small $\delta > 0$. Since $\mathcal{S}_n(c, q) = 1$ from eq. (3.7) we have $C_n^0(c, q) \geq C_n^1(c, q)$. Also, since $\mathcal{S}_n(c + \delta, q) = 0$ we have

$C_n^0(c+\delta,q) \le C_n^1(c+\delta,q)$. By subtracting these equations we have $C_n^0(c+\delta,q) - C_n^0(c,q) \le C_n^1(c+\delta,q) - C_n^1(c,q)$. By rearranging terms this becomes

$$p_q[h_{n+1}^{q^1}(c+\delta) - h_{n+1}^{q^1}(c)] \ge (1-p_q)[(c+\delta)^2 - c^2] \\ + h_{n+1}^{q^0}(c+\delta) - h_{n+1}^{q^0}(c).$$

Dividing by $(c+\delta)^2 - c^2$ and taking the limit $\delta \downarrow 0$ yields

$$p_q \mathcal{O}_{n+1}(c,q^1) \ge (1-p_q) + \mathcal{O}_{n+1}(c,q^0). \tag{3.12}$$

Contradicting the assumption. Thus, the optimal policy is a threshold policy.

Proof 22 (Lemma 6) *We show inductively that for all $e \in \mathbb{R}$ and $q \in \mathbb{Q}$, there exists a v'_n such that $\mathcal{O}_{n+i}(e,q) \le v'_n$, for $i = 1 \ldots (N+1-n)$.*

This property holds for $N+1$, since $h_{N+1}^q = a^2e^2 + \sigma^2$ and $\mathcal{O}_{N+1}(e,q) = a^2$. Thus, $v'_{N+1} = a^2$.

Assume the property holds for $n+1$ with v'_{n+1}. We will show the property holds for n. For a specific e and ω, there are two cases $V_n(ae+\omega,q) = C^0$ or $V_n(ae+\omega,q) = C^1$, see eq. (3.7). We prove the statement for the case when $V_n(ae+\omega,q) = C^0$. The other case yields the same result and is analogous.

Equation (3.4), on the previous page, is obtained for the case using $V_n(ae+\omega,q) = C^0$ and using the bound

$$V_n(a(x+\delta)+\omega,q) \le (a(x+\delta)+\omega)^2 \\ + E[V_{n+1}(a(x+\delta)+\omega,q^1)].$$

The right hand side of eq. (3.4) is comprised of two terms. The first term is upper bounded by a $2a^2 + \frac{a\omega}{x}$.

Next, we take the expectation of eq. (3.5) with respect to ω and then the limit with respect to δ. Using the inductive hypothesis to bound the second term by v'_{n+1}, this yields

$$\mathcal{O}_n(e, q) \leq 2a^2 + v'_{n+1}.$$

Thus, with $v'_n = 2a^2 + v'_{n+1}$ the induction is complete. We see that for all n, $v'_n = 2a^2(N + 1 - n) + a^2$ is an adequate bound.

3.6.1 Quasi-convex functions

In this section, definitions and results related to quasi-convex functions are presented.

Definition 9 *A function $f : \mathbb{R} \to \mathbb{R}$ is quasi-convex if for $x, y \in \mathbb{R}$ and $\lambda \in [0, 1]$*

$$f(\lambda x + (1 - \lambda)y) \leq \max\{f(x), f(y)\}.$$

Definition 10 *A function $f : \mathbb{R} \to \mathbb{R}$ is symmetric and non-decreasing in $|x|$ if for $0 \leq x < y$,*

$$\begin{aligned} f(x) &= f(-x) \text{ and} \\ f(x) &\leq f(y). \end{aligned}$$

Lemma 7 *If $f : \mathbb{R} \to \mathbb{R}$ is symmetric and non-decreasing in $|x|$ then f is quasi-convex.*

Proof 23 *For $x, y \in \mathbb{R}$, without loss of generality let $|y| > |x|$. Note $f(y) \geq f(x)$. For $\lambda \in [0, 1]$, since $|\lambda x + (1 - \lambda)y| \leq |y|$, we have $f(\lambda x + (1 - \lambda)y) \leq f(y)$.*

Lemma 8 *Let f, g be symmetric and non-decreasing in $|x|$. The function $h(x) = \min\{f(x), g(x)\}$ is symmetric and non-decreasing in $|x|$.*

Proof 24 *First, we show h is symmetric. For $x \in \mathbb{R}$,*

$$\begin{aligned} h(-x) &= \min\{f(-x), g(-x)\} \\ &= \min\{f(x), g(x)\} \\ &= h(x). \end{aligned}$$

We now show h is non-decreasing. For $0 \leq x < y$,

$$\begin{aligned} h(x) &= \min\{f(x), g(x)\} \\ &\leq \min\{f(y), g(y)\} \\ &= h(y). \end{aligned}$$

Lemma 9 *Let f be a symmetric and non-decreasing in $|x|$, W a random variable distributed $\mathcal{N}(0, \sigma^2)$ and $a \in \mathbb{R}$. The function $h(x) = E_W[f(ax + W)]$ is symmetric and non-decreasing in $|x|$.*

Proof 25 *First, we show h is symmetric. For $x \in \mathbb{R}$,*

$$\begin{aligned} h(-x) &= \int_{-\infty}^{\infty} f(-ax + w)\eta e^{\frac{-w^2}{2\sigma^2}} dw \\ &= \int_{-\infty}^{\infty} f(-ax - w')\eta e^{\frac{-w'^2}{2\sigma^2}} dw' \\ &= h(x) \end{aligned}$$

where $\eta = \frac{1}{\sqrt{2\pi\sigma^2}}$. The second equality holds by change of variables $w' = w$.

We now show h is non-decreasing. Let $0 \leq x < y$. Using the symmetry of f, with $\eta \overset{def}{=} \frac{1}{\sqrt{2\pi\sigma^2}}$, $h(x)$ can be written,

$$h(x) = \int_0^\infty f(w)\eta[e^{\frac{-(w-ax)^2}{2\sigma^2}} + e^{\frac{-(-w-ax)^2}{2\sigma^2}}]dw.$$

Consider

$$h(y) - h(x) = \int_0^\infty f(w)\eta g(w)dw,$$

with

$$\begin{aligned} g(w) \overset{def}{=} \quad & e^{\frac{-(w-ay)^2}{2\sigma^2}} + e^{\frac{-(-w-ay)^2}{2\sigma^2}} \\ & - \left[e^{\frac{-(w-ax)^2}{2\sigma^2}} + e^{\frac{-(-w-ax)^2}{2\sigma^2}} \right]. \end{aligned}$$

There exists a $\bar{w} > 0$ such that $g(w) < 0$ for $0 < w < \bar{w}$ and $g(w) \geq 0$ for $w \geq \bar{w}$. So

$$\begin{aligned} h(y) - h(x) &= \int_0^{\bar{w}} f(w)\eta g(w)dw + \int_{\bar{w}}^\infty f(w)\eta g(w)dw \\ &\geq f(\bar{w}) \int_0^{\bar{w}} \eta g(w)dw + f(\bar{w}) \int_{\bar{w}}^\infty \eta g(w)dw \\ &= f(\bar{w})[1-1] = 0. \end{aligned}$$

Thus, $h(y) \geq h(x)$.

Chapter 4: Remote Estimation of Multiple Systems Over Shared-State Multiple Packet-Drop Channels

A discrete-time remote estimation problem formed by a transmission system transmitting sensor measurements of l different systems to an estimator is investigated. The transmission system has different operating modes. Each mode of operation prioritizes transmitting the l sensor measurements differently. For non-trivial problems, no mode of operation simultaneously prioritizes transmitting all measurements. The transmission system is formed by a homogeneous Markov chain and a Bernoulli packet-drop communication link for each of the l sensors. The mode of operation is the current Markov chain state and determines each sensor's transmission priority. The probability of a packet-drop for a sensor quantifies that sensors priority. If a sensor has a low priority in a given mode of operation, the probability of a drop on its transmission link is high. We seek to design the transition probabilities between modes of operation, Markov chain states, to stabilize the estimation error for all l systems. For specific cases of this problem, algorithmic solutions are presented that will design a stabilizing transition matrix if one exists and certify infeasibility if one does not. For the general problem formulation, an algorithm is proposed to design a stabilizing transition matrix. If this algorithm converges, it converges to a solution. Also, this algorithm scales well with problem size, makes efficient updates, and in certain settings is guaranteed to find a stabilizing transition matrix if one exists. For several example applications, a numerical study compares the proposed algorithms to a branch and bound method.

This chapter is organized as follows. Section 4.1 provides an introduction. Section 4.2 formulates the problem and discusses related technical results. Section 4.3 considers the case of stabilizing the estimation error of one linear system, l is 1. Section 4.4 discusses an important characterization of the problem that is utilized to develop the algorithms. For a transmission system with only two modes of operation, Section 4.5 presents an algorithm to design the transition matrix between the modes of operation. Section 4.6 considers a transmission system with an arbitrary number of operating modes as well as restrictions on transitioning between certain modes of operation; an algorithmic solution is proposed. Section 4.7 contains a numerical comparison of the proposed algorithms.

The contributions of this chapter are listed below.

1. A semidefinite program (SDP) to design the operating modes transition probabilities when the transmission system has a single Bernoulli link, l equals 1.

2. An algorithm to design the operating modes transition probabilities when the transmission system has two modes of operation, m equals 2, and two Bernoulli links, l equals 2.

3. A proposed algorithm for the general case of designing the operating modes

transition probabilities when the transmission system has an arbitrary number of modes m and an arbitrary number of Bernoulli links l.

4. Several desirable properties of the algorithm proposed for the general case, including a correctness proof for this algorithm in the case when l equal 1.

5. Applications of these algorithms to operator support system design and a numerical comparison are provided.

4.1 Introduction

An estimator receives measurements of l linear systems over a transmission system that has multiple modes of operation, see Figure 4.1. The estimator produces state estimates for each linear system that minimizes the expected mean-squared estimation error.

At each time step, a sensor measures the state of each linear system. These sensor measurements are the input to the transmission system. Each of the l linear systems is driven by noise.

The transmission system has multiple modes of operation. Each mode prioritizes the transmission of some sensor measurements over the other sensor measurements. A different Bernoulli packet-drop link transmits each measurement to the estimator. The transmission priority level for a sensor measurement as specified by the current mode of operation determines the probability of packet-drop for that sensor measurement. For example, an operating mode with a high priority level for a given measurement designates a low drop probability for that measurement.

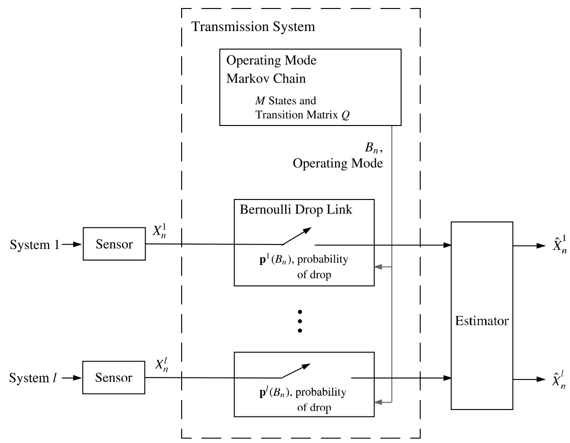

Figure 4.1: The structure of the transmission system that has multiple modes of operation and transmits the measurements of l system's states to the estimator.

A homogeneous Markov chain models the modes of operation and the probability of transitioning to a new operating mode. The Markov chain, transmission drop processes and noise driving the linear systems are assumed mutually independent.

We seek to design the transition probabilities between operating modes and thus transitions between transmission priorities to simultaneously stabilize the expected mean-squared estimation error for all l linear systems. A simple concrete example of the problem formulation is given in Section 4.2.3.

4.2 Problem Formulation

4.2.1 Notation

In this chapter, bold capital letters represent matrices such as $\mathbf{A}$, bolded lower-case letters represent vectors such as $\mathbf{f}$, and lower-case letters represent scalar quantities such as l. The entry in the i^{th} row and j^{th} column of matrix $\mathbf{A}$ is denoted

$\mathbf{A}(i,j)$. The i^{th} entry of the vector $\mathbf{f}$ is denoted $\mathbf{f}(i)$. The set of real numbers is represented by $\mathbb{R}$. The set of non-negative real numbers is represented by $\mathbb{R}_{\geq 0}$. The set of positive real numbers is denoted $\mathbb{R}_{+}$. Vector and matrix inequalities are defined entry-wise. For instance, for $\mathbf{A}$ and $\mathbf{B}$ in $\mathbb{R}^{m\times l}$, $\mathbf{A} \geq \mathbf{B}$, holds when $\mathbf{A}-\mathbf{B}$ is in $\mathbb{R}^{m\times l}_{\geq 0}$. Strict inequalities are defined similarly. Positive definite matrices are denoted by $\mathbf{A} \succ 0$. For a sequence of scalars d_n indexed by n, we use $d_{1:n}$ to denote the set $\{d_1, \ldots, d_n\}$. The set $\{1, \ldots, l\}$ is denoted by $\mathbb{L}$ and similarly $\mathbb{M}$ denotes the set $\{1, \ldots, m\}$. The following table summarizes this notation:

$\mathbf{A}(i,j)$	the entry in the i^{th} row and j^{th} column of matrix $\mathbf{A}$,
$\mathbf{f}(i)$	the i^{th} entry of the vector $\mathbf{f}$,
$\mathbf{A} \geq \mathbf{B}$	if $\mathbf{A}-\mathbf{B} \in \mathbb{R}^{m\times l}_{\geq 0}$,
$\mathbf{A} \succ 0$	matrix $\mathbf{A}$ is positive definite,
$\rho(\mathbf{A})$	spectral radius of matrix $\mathbf{A}$,
$d_{1:n}$	$\{d_1, \ldots, d_n\}$,
$\mathbb{L}$	$\{1, \ldots, l\}$,
$\mathbb{M}$	$\{1, \ldots, m\}$.

The k^{th} canonical basis vector of $\mathbb{R}^n$ is denoted $\mathbf{e}_k$. A column vector of dimension k whose entries are all one (zero) is denoted $\mathbf{1}_k$ ($\mathbf{0}_k$). The identity matrix is denoted as $\mathbf{I}$. The set of $m \times m$ matrices with entries in $\{0,1\}$ is denoted with $\{0,1\}^{m\times m}$. The diagonal matrix constructed from a k dimensional vector $\mathbf{f}$ by placing $\mathbf{f}(i)$ in the $(i,i)^{th}$ position is denoted with $\mathrm{diag}(\mathbf{f})$ or $\mathrm{diag}(\mathbf{f}(1), \ldots, \mathbf{f}(k))$. The variable n is reserved as an iteration and time index.

Throughout the chapter we frequently refer to the individual rows and columns of a matrix. To accommodate this, the following notation is used. For $\mathbf{H}$ in $\mathbb{R}^{m\times l}$,

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \dots & \mathbf{h}_l \end{bmatrix} = \begin{bmatrix} \mathbf{h}^{(1)} \\ \mathbf{h}^{(2)} \\ \vdots \\ \mathbf{h}^{(m)} \end{bmatrix},$$

where $\mathbf{h}^{(i)}$ in $\mathbb{R}^{1\times l}$ is the i^{th} row of the matrix $\mathbf{H}$ and $\mathbf{h}_i$ in $\mathbb{R}^{m\times 1}$ is the i^{th} column.

We reserve the variable X to denote the linear systems' state, W to denote the process noise, B for the transmission systems state, F for the Bernoulli drop processes, and V for the conditional expectation of estimation error. The quantities, X, W, B, F, and V are random variables and use a slightly different notation from deterministic quantities. For random variables, a capital letter such as X refers to a scalar random variable while a lower-case letter such as x refers to its realization. Also for random variables, bolded capital letters such as $\mathbf{X}$ refer to a vector random variable while bolded lower-case letters such as $\mathbf{x}$ refer to its realization.

4.2.2 Formulation

Consider l scalar linear time invariant systems with dynamics given by

$$X_{n+1}^i = a_i X_n^i + W_n^i, \tag{4.1}$$

where i is in $\mathbb{L}$, X_n^i is the state of system i at time n, a_i is a real-valued constant, W_n^i is independent and identically distributed Gaussian noise with zero mean and variance σ^2. The initial state of system i is a random variable X_0^i with known distribution.

Scalar systems are used for convenience, see Section 4.2.4 for an extension to non-scalar linear-time invariant systems.

A transmission system with different modes of operation attempts to transmit the states of these l systems to an estimator, at each time step. A homogenous Markov chain B_n with m states and transition matrix $\mathbf{Q}$, defined for i and j in $\mathbb{M}$ by

$$\mathbf{Q}(i,j) \stackrel{def}{=} P(B_{n+1} = j | B_n = i),$$

models the transmission system's current mode of operation and transitions to new modes of operation. At each time step n, system i's state X_n^i is measured and sent over a Bernoulli drop link to the estimator. Bernoulli link i successfully transmits the state X_n^i to the estimator if the drop process F_n^i is 1. It drops the transmission and does not transmit the state X_n^i to the estimator if the drop process F_n^i is 0. The current mode of operation B_n determines the probability of dropping system i's measurement

$$P(F_n^i = 0) \stackrel{def}{=} \mathbf{p}^i(B_n),$$

where the probability of drop $\mathbf{p}^i(B_n)$ specifies measurement i's transmission priority while in operating mode B_n. For interesting problem formulations, there is no single mode of operation B_n for which drop probabilities $\mathbf{p}^i(B_n)$ are low for all i in $\mathbb{L}$.

As indicated in Figure 4.1, at time n for each i in $\mathbb{L}$ the estimator receives X_n^i if F_n^i is 1 and a drop symbol if F_n^i is 0. We assume that the operating mode Markov chain, the drop processes, and the noise are mutually independent. The estimator is chosen to be the minimum mean-square error estimator for each system. Due to the independence between random variables the conditional mean of the mean-square estimator becomes

$$\hat{X}_n^i = \begin{cases} X_n^i & \text{if } F_n^i = 1, \\ a_i \hat{X}_{n-1}^i & \text{if } F_n^i = 0. \end{cases}$$

Let the initial state estimate $\hat{x}_0^i$ equal the mean $E[X_0^i]$. The estimation error for each system at time n is

$$\tilde{X}_n^i \overset{def}{=} X_n^i - \hat{X}_n^i.$$

Definition 11 (Expected Mean-square Error Stability) *For i in $\mathbb{L}$, the estimation error for system i is expected mean-square stable, if the following holds for every initial error $\tilde{X}_0^i$ and mode of operation B_0,*

$$\limsup_{n\to\infty} E\left[(\tilde{X}_n^i)^2\right] < \infty. \tag{4.2}$$

For many applications, it is important to restrict certain transitions between operating modes. It may not be physically possible to transition between all operating modes. Let $\mathbf{S}$ in $\{0,1\}^{m\times m}$ specify the allowable state transitions of the Markov chain. Transitioning from Markov chain state i to j is allowable if $\mathbf{S}(i,j)$ equals 1. If $\mathbf{S}(i,j)$ equals 0, then transitions from state i to j are not allowed and $\mathbf{Q}(i,j)$ is 0. These conditions can be imposed using the following sparsity constraint:

$$\mathbf{Q} \leq \mathbf{S}. \tag{4.3}$$

Problem 5 *Let $\boldsymbol{S}$ be a given matrix in $\{0,1\}^{m\times m}$. Find a stochastic matrix $\boldsymbol{Q}$ satisfying eq. (4.3) for which the stability condition in eq. (4.2) holds for all i in $\mathbb{L}$.*

The minimum mean-squared error for each system is a function of the Markov chain history. Taking the expectation with respect to the noise and packet-drop process conditioned on the Markov chain state history we define,

$$V_n^i(B_{1:n}) \stackrel{def}{=} E[(\tilde{X}_n^i)^2 | B_{1:n}].$$

Using standard conditional expectation arguments, we arrive at the following recursion for i in $\mathbb{L}$,

$$V_{n+1}^i(B_{1:n+1}) = a_i^2 \mathbf{p}^i(B_{n+1}) V_n^i(B_{1:n}) + \sigma^2. \tag{4.4}$$

Similar to the approach taken in the jump Markov linear systems literature [83], we use the following identity:

$$E_B[V_n^i(B_{1:n})] = \sum_{j=1}^{m} E_B[V_n^i(B_{1:n}) \chi_{\{B_n=j\}}],$$

where $\chi_{\{B_n=j\}}$ is the indicator function of the event B_n equals j. Define the vector of expectations as

$$\mathbf{r}_n^i \stackrel{def}{=} \begin{bmatrix} E_B[V_n^i(B_{1:n}) \chi_{B_n=1}] \\ \vdots \\ E_B[V_n^i(B_{1:n}) \chi_{B_n=m}]] \end{bmatrix}.$$

Using eq. (4.4) and standard transformations, the following recursion holds for i in $\mathbb{L}$,

$$\mathbf{r}_{n+1}^i = \mathbf{Q}^T \mathbf{D}_i \mathbf{r}_n^i + \sigma^2 \mathbf{1}_m,$$

where $\mathbf{D}_i \stackrel{def}{=} a_i^2 \operatorname{diag}([\mathbf{p}^i(1), \ldots, \mathbf{p}^i(m)])$. Thus, the expected estimation error of system i is mean-square stable if and only if $\rho(\mathbf{Q}^T \mathbf{D}_i)$ is less than 1. This implies the estimation error is mean square stable for all systems if and only if the following inequalities are satisfied,

$$\rho(\mathbf{Q}^T \mathbf{D}_i) < 1, \qquad i \in \mathbb{L}.$$

Using our analysis so far, we can restate Problem 5 in more precise terms as follows:

Problem 6 (Main Problem) *Let $\boldsymbol{S}$ in $\{0,1\}^{m\times m}$ and positive constants γ_i, with i in $\mathbb{L}$ be given. Find a stochastic matrix $\boldsymbol{Q}$ that satisfies the following inequalities i in $\mathbb{L}$,*

$$\begin{aligned} \boldsymbol{Q} &\leq \boldsymbol{S}, \\ \rho(\boldsymbol{Q}^T \boldsymbol{D}_i) &\leq \gamma_i, \qquad i \in \mathbb{L}, \end{aligned} \tag{4.5}$$

where $\boldsymbol{D}_i$ are diagonal matrices defined as

$$\boldsymbol{D}_i \overset{def}{=} a_i^2 \operatorname{diag}([\boldsymbol{p}^i(1), \ \ldots \ , \ \boldsymbol{p}^i(m)]).$$

Remark 5 *The γ_i parameters are introduced to generalize the problem formulation. A parameter choice of γ_i less than 1 implies stability for the mean-squared estimation error of system i. Also, this parameter provides a new design mechanism. Choice of γ_i sets the slowest possible rate of the mean estimation error's convergence. Selecting a smaller γ_i will force the design of a transition matrix $\boldsymbol{Q}$ with a smaller slowest possible convergence rate for the error of system i. Also, γ_i may be different from γ_j for i not equal j, leading to different slowest possible convergence rates of the errors for different systems. Setting these parameters with differing values allows for prioritization of different system's estimation error.*

4.2.3 Motivating Examples

To clarify the formulation, consider an illustrative example of monitoring two systems whose states are separately measured and transmitted over a transmission system with two Bernoulli drop channels. The transmission system for this example has two operating modes. In the first operating mode, the transmission of the first system's state is prioritized and the second system's state is not transmitted, meaning that the first system's state is transmitted with a low probability of drop $\mathbf{p}^1(1)$ and the second system's state is transmitted with a drop probability $\mathbf{p}^2(1)$ of 1. In the second operating mode, the second system's state is prioritized for transmission, meaning its drop probability $\mathbf{p}^2(2)$ is low and the first system is transmitted with a drop probability $\mathbf{p}^1(2)$ of 1. We seek a Markov chain transition matrix $\mathbf{Q}$ between these modes of operation that leads to mean square stability for the error estimate of both systems.

Sensor dynamics are added to this example by jointly operating the Bernoulli drop links on a battery with a charge capacity of a single measurement. The battery is recharged during periods without sensor use by energy harvesting. The probability of recharging the battery during a single time-step is r. The Markov chain in Figure 4.3 jointly describes both the operating modes and battery dynamics. State 3 represents that the battery is charged. State 4 represents that the battery is depleted. When the battery is charged, we seek to design a randomized policy of when to transmit system 1's state, or system 2's state or wait to take a measurement and preserve the battery. Equivalently, we seek to design a transition matrix with

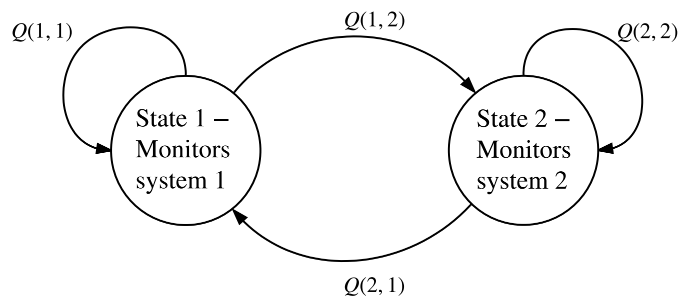

Figure 4.2: An example of a two operating mode chain determining the transmission of two sensors monitoring independent systems is shown. In state 1, system 1's state is successfully transmitted with probability $1 - \mathbf{p}^1(1)$ and system 2's state is not transmitted, $\mathbf{p}^2(1)$ equals 0. In state 2, system 2's state is successfully transmitted with probability $1 - \mathbf{p}^2(2)$ and system 1's is not transmitted, $\mathbf{p}^1(2)$ equals 0.

the graph structure of the Markov chain in Figure 4.3,

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ Q(3,1) & Q(3,2) & Q(3,3) & 0 \\ 0 & 0 & 1-r & r \end{bmatrix},$$

and leads to mean square stability for both systems' estimation error.

This example demonstrates how the Markov chain's states are used to model transmission operating modes and the allowable structures of the transition matrix is used to model the allowable dynamics of the operating modes. In general, we investigate problems with l systems to monitor, a Markov chain with m modes of operation, and arbitrary restrictions on the allowable transitions between operating modes.

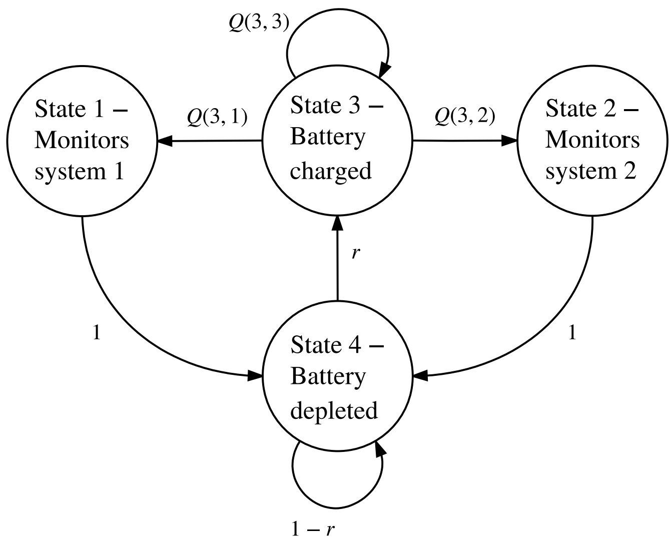

Figure 4.3: An example Markov chain is shown of operating the example in Figure 4.2 on battery power. The battery has capacity for only a single transmission. In state 1 (2), system 1 (2) is transmitted successfully with probability $1 - \mathbf{p}^1(1)$ $(1 - \mathbf{p}^2(2))$ and system 2 (1) is not transmitted, respectively. In state 4, the battery is depleted. The battery recharges with probability r. In state 3, the battery is charged.

In Section 4.7, the examples used for numerical testing the algorithms demonstrate the formulations used in human operator attention allocation problems. An additional possible application is that of monitoring mammalian cell cultures. This requires both costly equipment and methods of processing large amounts of real-time data [84]. Guiding the dynamic allocation of both physical and computational resources can increase the efficacy and efficiency of monitoring such cell cultures. The algorithms presented systematically design the dynamic prioritization of transmissions.

4.2.4 Non-scalar linear-time invariant systems

Instead of scalar systems, we can also consider systems, as follows:

$$\mathbf{X}_{n+1}^i = \mathbf{A}_i \mathbf{X}_n^i + \mathbf{W}_n^i, \tag{4.6}$$

where $\mathbf{X}_n^i$ may have dimension greater than one and i is in $\mathbb{L}$ and the noise $\mathbf{W}_n^i$ is independently and identically distributed zero-mean Gaussian with positive definite covariance matrix $\boldsymbol{\Sigma}$. The remainder of this section shows that Problem 5 with the the systems of eq. (4.6) instead of eq. (4.1) can be restated as follows.

Problem 7 (Non-scalar Problem) *Let $\boldsymbol{S}$ in $\{0,1\}^{m \times m}$ and γ_i greater than 0 for each i in $\mathbb{L}$ be given. Find a stochastic matrix $\boldsymbol{Q}$ that satisfies the following inequalities for each i in $\mathbb{L}$,*

$$\boldsymbol{Q} \leq \boldsymbol{S},$$

$$\rho(\boldsymbol{Q}^T \boldsymbol{D}_i) \leq \gamma_i,$$

where $\boldsymbol{D}_i \stackrel{def}{=} \rho^2(\boldsymbol{A}_i) diag([\boldsymbol{p}^i(1), \ \ldots \ , \ \boldsymbol{p}^i(m)])$ *are parameter matrices.*

Note the only difference between Problem 6 and Problem 7 are the parameter matrices.

The arguments leading from Problem 5 with the systems of eq. (4.6) to a restatement as Problem 7 are very similar to the arguments used in the scalar case of Section 4.2.2. However, instead of analyzing the dynamics of the mean-squared error conditioned on the operating mode history, now defined as

$$V_n^i(B_{1:n}) \stackrel{def}{=} E\left[(\mathbf{X}_n^i - \hat{\mathbf{X}}_n^i)^T(\mathbf{X}_n^i - \hat{\mathbf{X}}_n^i)|B_{1:n}\right], \tag{4.7}$$

for i in $\mathbb{L}$, we define a more convenient system Y_n^i. Define the scalar systems

$$Y_{n+1}^i(B_{1:n+1}) = \rho^2(\mathbf{A}_i)\mathbf{p}^i(B_{n+1})Y_n^i(B_{1:n}) + trace(\mathbf{\Sigma}), \tag{4.8}$$

for i in $\mathbb{L}$. This system Y_n^i is different from the mean-squared error conditioned on the operating mode history V_n^i. However, we show that the two systems V_n^i and Y_n^i have the same asymptotic behavior. We require the following assumption.

Assumption 3 *For each i in $\mathbb{L}$, assume that for each eigenvector of $\boldsymbol{A}_i$ there exists an eigenvector of $\mathbf{\Sigma}$ that is not orthogonal to it. In other words, for each i in $\mathbb{L}$, assume that the noise excites all of the modes of $\boldsymbol{A}_i$.*

Lemma 10 *Under Assumption 3, for system i in eq. (4.6), the expected mean-squared error V_n^i remains bounded*

$$\lim_{n\to\infty} E_B[V_n^i(B_{1:n})] < \infty,$$

if and only if the system Y_n^i remains bounded

$$\lim_{k\to\infty} E_B[Y_n^i(B_{1:n})] < \infty.$$

Proof 26 *See proof 38 in Section 4.9.*

Now, following the arguments of Section 4.2.2 leads to Problem 7 if the systems in eq. (4.8) are used instead of the scalar mean-squared error systems in eq. (4.4).

4.2.5 Literature Review

The problem formulation is related to the sensor selection, control over noisy channels, jump Markov linear systems (JMLS), and generalized bilinear programming literatures. The algorithms proposed as a solution to this problem utilize results and analysis from the positive systems literature, Bender's decomposition techniques, and fixed-point iteration results. The connections to these works is discussed below.

Accounting for costly sensor measurements has a long history. In [76], one of several different sensors is selected, each with a different cost, to minimize a combination of error and sensor costs. In [78] and [85], sensor measurements are made over Bernoulli drop channels and the relationship between the estimation error covariance and Bernoulli channel throughput is studied via upper and lower bounds on the error covariance. A problem similar to our proposed problem is considered in [86] [87], where an algorithm is proposed to design a distribution for randomly selecting one of several sensors at each time step. The algorithm in [86] is based upon minimizing an upper bound of the estimation error covariance. In [88], the next measurement of a nonlinear system is scheduled to guarantee stability of the closed loop system based on the current measurement. In [89], a measurement criterion is proposed that preserves the Gaussian property of the innovations process. Both [90] and [91] analyze estimating processes over multiple event-trigged channels under different measurement or triggering policies. Our proposed problem differs from those in the literature in the following ways: our formulation can account for partially utilizing multiple faulty sensors with varying prioritization levels allocated to each sensor, and our formulation accounts for dynamics, such as a battery, in the sensor apparatus.

Control of systems over noisy channels [92] or packet-drop channels [93] [94] has been investigated and fundamental stability characterizations discovered. In [95], estimation over a Bernoulli drop channel with fixed drop probability is considered and optimal transmission policies developed. A related finite time horizon problem is considered in [96]. A connection between noisy channels and packet-drop channels is made in [97]. The work presented in this chapter differs in that here the packet-drop probabilities have dynamics and we seek to design the transition between operating modes that determine the packet-drop probabilities.

When monitoring a single scalar system, l equals 1, a direct connection between Problem 6 and JMLS [83] [98] is made by creating a stacked state of each systems' estimation error and expanding the Markov chain state space to explicitly enumerate all measurement drop possibilities in each original state. This constructed JMLS is quite large and highly structured, even for modestly sized problems. However, when monitoring multiple systems, l greater than 1, such a direct connection is not possible as is highlighted in the non-scalar case where mean-squared error stability is characterized in terms of the spectral radii of the system matrices $\mathbf{A}_i$, see Section 4.2.4.

The proposed algorithms are developed by formulating the problem as a feasibility problem with bilinear inequality constraints and linear equality constraints. We then exploit the specific structure of the problem to develop the solution algorithms. In the literature, approaches have been discussed for optimization problems with non-convex quadratic objective functions over polygonal feasible regions [99] [100], and problems with bilinear objective functions over polygonal feasible regions [101] [102], as well as non-convex quadratically constrained quadratic problems (QCQP) [103]. Also, solution methods for problems with a bilinear cost and separable polygonal feasibility regions have been developed [104] [105]. These are very general and difficult problems. Solution techniques commonly include relaxation techniques [106], decomposition methods [107], and branch and bound/cut algorithms [108]. These solutions methods tailored to a specific example in farm management is presented in [109]. A framework is presented in [110], [111], and [112] for a branch and bound algorithm specifically designed for an optimization problem with a bilinear objective function and bilinear inequality constraints. The branch and bound algorithm used for comparison to our proposed algorithm in Section 4.7 was developed from these works. In contrast to these general approaches we utilized the structure of our problem and developed an iterative algorithm that scales well with problem size and the number of constraints or problem partitions does not grow with each iteration.

We leverage results from the positive systems literature characterizing the spectral radius of entry-wise nonnegative matrices [113] [114]. A related result in this literature that was not used, states that certain general quadratic optimization problems defined by entry-wise nonnegative quadratic terms are equivalent to a semidefinite program [115]. Our problem does not satisfy the conditions of this result, or the slightly generalized version [116], due to the row stochastic constraint and zero pattern constraint on the transition matrix.

Bender's decomposition methods decompose an optimization problem into a master problem and numerous decomposed sub-problems [117] [118] [119]. It is commonly used for mixed-integar problems [120] [121]. Directly applying Bender's decomposition to a problem may not yield a solution [122] and feasibility at each iteration must be considered [123]. In fact, Bender decomposition produces an algorithm with a non-convex optimization problem to solve at each iteration for Problem 6. However, inspired by this method's use of Farkas lemma to partition the problem, we use Farkas lemma and the specific structure of our bilinear feasibility problem to derive a equivalent feasibility problem in terms of a subset of the original variables.

As highlighted by the example in the introduction, problems involving communication over energy harvesting sensors have related problem formulations and solution techniques [66]. In [124], similar to our formulation a linear system driven by noise is remotely estimated over a Bernouli drop channel operating on battery power. The decision of the transmission power level, which determines the drop probability, is formulated as a Markov decision process and an approximate optimal solution by investigating the form of the value function. In [125], a transmission policy is proposed that attempts to balance the transmission rate, battery level, and estimation performance. Objective functions other than mean square estimation error have also been explored: [67] uses a distortion metric and [68] uses a throughput cost.

4.3 Semidefinite Program To Design $\mathbf{Q}$ For a Single System ($l = 1$)

In this section, Problem 6 with a single system, l equals 1, is rewritten as a semi-definite program (SDP) [115]. Even with only a single system, designing the transition matrix $\mathbf{Q}$ between operating modes can be an important and interesting problem. For example, the operating mode structure of a battery-operated communication link shown in Figure 4.3 demonstrates the use of designing operating mode transitions even when only one Bernoulli link is present. The battery in that example only had a charge capacity of a single transmission. With a larger charge capacity or a more refined model of battery dynamics, the problem can become quite complex. In general, with a large number of operating modes and depending on the restrictions placed on the transition matrix, Problem 6 with a single system can be nontrivial.

With l equals 1, Problem 6 simplifies to the following: Let $\mathbf{S}$ in $\{0, 1\}^{m \times m}$ and $\mathbf{D}$ be a diagonal matrix with strictly positive diagonal entries be given. Find a transition matrix $\mathbf{Q}$ that satisfies

$$\mathbf{Q} \leq \mathbf{S},$$

$$\rho\left(\mathbf{Q}^T\mathbf{D}\right) < 1.$$

The following proposition allows us to cast the spectral radius condition as an SDP.

Proposition 2 *Let $\boldsymbol{S}$ in $\{0,1\}^{m\times m}$ and a positive definite diagonal matrix $\boldsymbol{D}$ be given. There exists a stochastic matrix $\boldsymbol{Q}$ that satisfies*

$$\boldsymbol{Q} \leq \boldsymbol{S},$$

$$\rho(\boldsymbol{Q}^T\boldsymbol{D}) < 1, \tag{4.9}$$

if and only if there exists $\tilde{\boldsymbol{Q}}$ and diagonal $\boldsymbol{P}$ such that

$$\begin{bmatrix} \boldsymbol{P} & \tilde{\boldsymbol{Q}}\boldsymbol{D} \\ \boldsymbol{D}\tilde{\boldsymbol{Q}}^T & \boldsymbol{P} \end{bmatrix} \succ 0, \tag{4.10}$$

$$\tilde{\boldsymbol{Q}}\boldsymbol{1}_m = \boldsymbol{P}\boldsymbol{1}_m,$$

$$\tilde{\boldsymbol{Q}} \leq \boldsymbol{P}\boldsymbol{S},$$

$$\tilde{\boldsymbol{Q}} \geq 0.$$

Moreover, if such a $\tilde{\boldsymbol{Q}}$ and $\boldsymbol{P}$ exist, then eq. (4.9) is satisfied by $\boldsymbol{Q}$ equals $\boldsymbol{P}^{-1}\tilde{\boldsymbol{Q}}$.

Proof 27 *Since all the entries of $\boldsymbol{Q}$ and $\boldsymbol{D}$ are nonnegative, the following are equivalent:*

- $\rho(\boldsymbol{Q}^T\boldsymbol{D}) < 1$,
- $\rho(\boldsymbol{D}\boldsymbol{Q}^T) < 1$,

- $\rho(\boldsymbol{QD}) < 1$,
- $\exists \boldsymbol{\xi} \in \mathbb{R}^m$ *s.t* $\boldsymbol{\xi} \geq 0,\ \boldsymbol{QD\xi} < \boldsymbol{\xi}$, *and*
- $\exists$ diagonal $\boldsymbol{P} \succ 0$ *s.t.*

  $[\boldsymbol{QD}]^T \boldsymbol{P}\ \boldsymbol{QD}\ \prec \boldsymbol{P}$.

Defining $\tilde{\boldsymbol{Q}} \overset{def}{=} \boldsymbol{PQ}$*, the last inequality,* $\boldsymbol{DQ}^T\boldsymbol{P}\ \boldsymbol{QD} \prec \boldsymbol{P}$*, is re-written as:*

$$\boldsymbol{D}\tilde{\boldsymbol{Q}}^T\boldsymbol{P}^{-1}\tilde{\boldsymbol{Q}}\boldsymbol{D} - \boldsymbol{P} \prec 0.$$

Using Schur's complement this becomes

$$\begin{bmatrix} \boldsymbol{P} & \tilde{\boldsymbol{Q}}\boldsymbol{D} \\ \boldsymbol{D}\tilde{\boldsymbol{Q}}^T & \boldsymbol{P} \end{bmatrix} \succ 0.$$

The above is a linear matrix inequality (LMI) in $\boldsymbol{P}$ *and* $\tilde{\boldsymbol{Q}}$*. However,* $\boldsymbol{Q}$ *must be a transition matrix and satisfy the sparsity constraint in eq. (4.3),* $\boldsymbol{Q} \leq \boldsymbol{S}$*. Since* $\boldsymbol{P}$ *is diagonal and positive definite, adding the following constraints enforces these requirements.*

1. $\boldsymbol{Q}$ *is row stochastic, if the following constraint is added*

   $$\tilde{\boldsymbol{Q}}\boldsymbol{1}_m = \boldsymbol{P}\boldsymbol{1}_m.$$

2. *The sparsity constraint in eq. (4.3),* $\boldsymbol{Q} \leq \boldsymbol{S}$*, is enforced by adding the constraint*

   $$\tilde{\boldsymbol{Q}} \leq \boldsymbol{PS}.$$

   Because eq. (4.3) is an entry-wise inequality, it is vital that $\boldsymbol{P}$ *is diagonal to establish the equivalent constraint stated above.*

3. The entries of $\boldsymbol{Q}$ must be nonnegative, $\boldsymbol{Q}$ in $\mathbb{R}_{\geq}^{m \times m}$. This is enforced by

$$\tilde{\boldsymbol{Q}} \geq 0.$$

This concludes the proof.

4.4 Preliminary Problem Characterization For Monitoring Multiple Systems ($l > 1$)

This section presents a critical result used to develop solution algorithms for the case when l is larger than 1. Before we present this characterization of the problem, three observations highlight the challenge and need to search in the entire space of stochastic matrices for Problem 6.

First: Proposition 2 cannot be extended to multiple systems. The variable transformation enabling Proposition 2,

$$\tilde{\mathbf{Q}} \overset{def}{=} \mathbf{P}\mathbf{Q},$$

does not immediately lead to a solution for l larger than 1. The analysis in proof 27 leading from the spectral radius condition $\rho(\mathbf{Q}^T\mathbf{D})$ less than 1 to an LMI in $\tilde{\mathbf{Q}}$ and $\mathbf{P}$ can be repeated for each $\mathbf{D}_i$, leading to l LMI's in

$$\tilde{\mathbf{Q}}_i \overset{def}{=} \mathbf{P}_i\mathbf{Q}$$

and $\mathbf{P}_i$. However with l greater than 1, the variable transformation requires the additional constraints

$$\mathbf{P}_i^{-1}\tilde{\mathbf{Q}}_i = \mathbf{P}_j^{-1}\tilde{\mathbf{Q}}_j,$$

for i and j in $\mathbb{L}$, in the feasibility problem of eq. (4.10). With these constraints, the problem is no longer an SDP and standard transformations cannot alleviate this complication.

Second: Solutions to Problem 6 cannot be characterized by their stationary probability mass functions, as demonstrated by the following example.

Example 1 *Consider that l is 2 and that $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ are given, as follows:*

$$\boldsymbol{D}_1 = \begin{bmatrix} .45 & 0 \\ 0 & 2 \end{bmatrix}, \qquad \boldsymbol{D}_2 = \begin{bmatrix} 2 & 0 \\ 0 & .45 \end{bmatrix}.$$

Although alternative $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ given below have the same stationary probability mass function, the first solves Problem 6, with γ_1 and γ_2 equal to 1, while the latter does not. The two transition matrices are:

$$\boldsymbol{Q}_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad \boldsymbol{Q}_2 = \begin{bmatrix} .5 & .5 \\ .5 & .5 \end{bmatrix}.$$

Indeed, the stationary distribution of $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ is $[.5 \quad .5]$. For $\boldsymbol{Q}_1$, the spectral radii are less than 1,

$$\rho(\boldsymbol{D}_1\boldsymbol{Q}_1) = \rho(\boldsymbol{D}_2\boldsymbol{Q}_1) = .9487.$$

For $\boldsymbol{Q}_2$, the spectral radii are greater than 1,

$$\rho(\boldsymbol{D}_1\boldsymbol{Q}_2) = \rho(\boldsymbol{D}_2\boldsymbol{Q}_2) = 1.225.$$

Third: Randomization may be necessary. A transition matrix in $\{0,1\}^{m\times m}$ defines a Markov chain that makes deterministic state transitions. A transition matrix with some entries that are not identically 0 or 1 defines a Markov chain that makes randomized state transitions while in some states. The following example demonstrates that randomization may be necessary.

Example 2 *Consider that l is 2 and that $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ are given, as follows:*

$$\boldsymbol{D}_1 = \begin{bmatrix} .1 & 0 \\ 0 & 2 \end{bmatrix}, \qquad \boldsymbol{D}_2 = \begin{bmatrix} 2 & 0 \\ 0 & .6 \end{bmatrix}.$$

Below a randomized transition matrix $\boldsymbol{Q}_1$ that solves Problem 6, with γ_1 and γ_2 equal to 1, and a transition matrix $\boldsymbol{Q}_2$ in $\{0,1\}^{2\times 2}$ are given.

$$\boldsymbol{Q}_1 = \begin{bmatrix} 0 & 1 \\ .66 & .33 \end{bmatrix}, \qquad \boldsymbol{Q}_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

For $\boldsymbol{Q}_1$, the spectral radii conditions are:

$$\rho(\boldsymbol{D}_1 \boldsymbol{Q}_1) = .8286, \qquad \rho(\boldsymbol{D}_2 \boldsymbol{Q}_1) = .9998.$$

For $\boldsymbol{Q}_2$, the spectral radii conditions are:

$$\rho(\boldsymbol{D}_1 \boldsymbol{Q}_2) = .4472, \qquad \rho(\boldsymbol{D}_2 \boldsymbol{Q}_2) = 1.0954.$$

Thus, $\boldsymbol{Q}_2$ does not stabilize the estimation error of system 2. Also, all other matrices in $\{0,1\}^{2\times 2}$ do not simultaneously stabilize both estimation errors.

4.4.1 A Re-characterization of Problem 6

In light of the challenges posed by Problem 6, we seek algorithmic solutions, presented in Section 4.5 and Section 4.6, which leverage the preliminary re-characterization of Problem 6 detailed in Theorem 6 below. Before Theorem 6 is presented, we introduce a supporting lemma and some necessary notion.

First, without loss of generality, let γ_i equal 1 for i in $\mathbb{L}$ in the remainder of this chapter. By dividing by γ_i, eq. (4.5) becomes

$$\rho\left(\mathbf{Q}^T \frac{1}{\gamma_i}\mathbf{D}_i\right) \leq 1,$$

for i in $\mathbb{L}$. By redefining the diagonal parameter matrices $\mathbf{D}_i$ as

$$\frac{1}{\gamma_i}\mathbf{D}_i,$$

without loss of generality, for i in $\mathbb{L}$ we let γ_i equal 1 in Problem 6.

The following lemma transforms the spectral radii conditions into a system of inequalities.

Lemma 11 *A stochastic matrix $\boldsymbol{Q}$ satisfies eq. (4.5) if and only if there exists $\boldsymbol{H}$ in $\mathbb{R}_+^{m\times l}$ that satisfies*

$$\boldsymbol{D}_i \boldsymbol{Q} \boldsymbol{h}_i \leq \boldsymbol{h}_i, \tag{4.11}$$

for i in $\mathbb{L}$. Recall, $\boldsymbol{h}_i$ is the i^{th} column of $\boldsymbol{H}$.

Proof 28 *A direct consequence of Lemma 18 in Section 4.9.*

The following notation is required. The parameters matrices $\mathbf{D}_i$ for i in $\mathbb{L}$ are re-stacked by the mode of operation k in $\mathbb{M}$,

$$\mathbf{D}^{(k)} \stackrel{def}{=} \operatorname{diag}\left([\mathbf{D}_1(k,k), \ldots \mathbf{D}_l(k,k)]\right).$$

Note $\mathbf{D}^{(k)}$ comprises the parameter weights associated with the Markov state k. $\mathbf{D}_i$ comprises the parameter weights associated with system i. Let $ON(k)$, defined below, denote the set of states to which the Markov chain may transition away from state k,

$$\mathrm{ON}(k) \stackrel{def}{=} \{ \ i \mid \mathbf{S}(k,i) = 1 \ \}.$$

Theorem 6 *Let $\boldsymbol{S}$ in $\{0,1\}^{m\times m}$ be given. There exists $\boldsymbol{H}$ in $\mathbb{R}_+^{m\times l}$ and a (row) stochastic matrix $\boldsymbol{Q}$ that satisfy,*

$$\begin{aligned} \boldsymbol{Q} &\leq \boldsymbol{S}, \\ \boldsymbol{D}_i \boldsymbol{Q} \boldsymbol{h}_i &\leq \boldsymbol{h}_i, \end{aligned} \tag{4.12}$$

for i in $\mathbb{L}$ if and only if there exists $\boldsymbol{H}$ in $\mathbb{R}_+^{m\times l}$ such that the following inequalities are satisfied for all $\boldsymbol{Z}$ in $\mathbb{R}_{\geq 0}^{l\times m}$,

$$\boldsymbol{h}^{(k)} \boldsymbol{z}_k \geq \min_{i\in \mathrm{ON}(k)} \boldsymbol{h}^{(i)} \boldsymbol{D}^{(k)} \boldsymbol{z}_k, \tag{4.13}$$

for k in $\mathbb{M}$. Recall that $\boldsymbol{z}_k$ is the k^{th} column of $\boldsymbol{Z}$ and $\boldsymbol{h}^{(k)}$ is the k^{th} row of $\boldsymbol{H}$.

Proof 29 *See proof 42 in Section 4.9.*

Note that the condition in eq. (4.13) does not involve $\mathbf{Q}$. Theorem 6 provides conditions solely in terms of $\mathbf{H}$ that are equivalent to the conditions in eq. (4.12) and therefore, by Lemma 11, equivalent to Problem 6. The remainder of this chapter uses the conditions in eq. (4.13) to develop iterative algorithms for Problem 6.

4.5 Algorithm to Design $\mathbf{Q}$ for $l = 2$ and $m = 2$

This section presents an algorithm to design the transitions between two modes of operation, l equals 2, that prioritize between the transmission of two different system's state, m equals 2, see Figure 4.2. No restrictions are placed on the transitions between modes of operation,

$$\mathbf{S} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}. \tag{4.14}$$

For this case, Problem 6 is solved by finding a feasible solution to the conditions in eq. (4.13). The conditions in eq. (4.13) become the feasibility problem that there exists an $\mathbf{H}$ in $\mathbb{R}_{+}^{2\times 2}$ such that the following inequalities hold for all $\mathbf{Z}$ in $\mathbb{R}_{\geq 0}^{2\times 2}$,

$$\mathbf{h}^{(1)}\mathbf{z}_1 \geq \min\left(\mathbf{h}^{(1)}\mathbf{D}^{(1)}\mathbf{z}_1,\ \mathbf{h}^{(2)}\mathbf{D}^{(1)}\mathbf{z}_1\right), \tag{4.15}$$

$$\mathbf{h}^{(2)}\mathbf{z}_2 \geq \min\left(\mathbf{h}^{(1)}\mathbf{D}^{(2)}\mathbf{z}_2,\ \mathbf{h}^{(2)}\mathbf{D}^{(2)}\mathbf{z}_2\right). \tag{4.16}$$

We solve Problem 6 by identifying parameter matrices, $D^{(1)}$ and $D^{(2)}$, ranges that lead to trivial solutions and provide an algorithm that tests eq. (4.13) feasibility for the non-trivial cases.

4.5.1 Trivial cases

The following problem parameter cases lead to trivial solutions:

- $\mathbf{D}_1 \leq \mathbf{I}$,
- $\mathbf{D}_1 > \mathbf{I}$,
- $\mathbf{D}^{(1)} \leq \mathbf{I}$,
- $\mathbf{D}^{(1)} > \mathbf{I}$.

Similarly, trivial solutions are available when the $\mathbf{D}_2$ or $\mathbf{D}^{(2)}$ parameter matrices satisfy one of these cases. These trivial cases correspond to a single mode of operation adequately transmitting all systems measurements simultaneously or one of the system's state not being adequately transmitting in any mode of operation. Equation (4.15) and eq. (4.16) are analyzed for each of the cases:

$\mathbf{D}_1 \leq \mathbf{I}$: For both Markov chain states the parameters for system 1, $a_1^2\mathbf{p}_1(1)$ and $a_1^2\mathbf{p}_1(2)$ are less than or equal to 1. Thus, any stochastic matrix $\mathbf{Q}$ satisfies $\rho(\mathbf{Q}^T\mathbf{D}_1)$ less than or equal to 1. The estimation error for system 1 is mean square stable for any $\mathbf{Q}$.

$\mathbf{D}_1 > \mathbf{I}$: For both Markov chain states the parameters for system 1, $a_1^2\mathbf{p}_1(1)$ and $a_1^2\mathbf{p}_1(2)$ are greater then 1. Thus, $\rho(\mathbf{Q}^T\mathbf{D}_1)$ is greater than 1 for any stochastic matrix $\mathbf{Q}$. Problem 6 is feasible. The estimation error for system 1 is not mean square stable for any $\mathbf{Q}$.

$\mathbf{D}^{(1)} \leq \mathbf{I}$: In Markov chain state 1, the parameters for both systems, $a_1^2\mathbf{p}_1(1)$ and $a_2^2\mathbf{p}_2(1)$ are less than or equal to 1. Both systems are mean-square stable by choosing to always stay in Markov chain state 1, $\mathbf{Q}(1,1)$ equals 1 and $\mathbf{Q}(2,2)$ equals 0.

$\mathbf{D}^{(1)} > \mathbf{I}$: In Markov chain state 1, the parameters for both systems, $a_1^2\mathbf{p}_1(1)$ and $a_2^2\mathbf{p}_2(1)$ are greater than 1. Thus, Markov chain state 1 does not help stabilize either system and without loss of generality the only remaining hope of solving Problem 6 is the transition matrix $\mathbf{Q}$ with $\mathbf{Q}(1,1)$ equals 0 and $\mathbf{Q}(2,2)$ equals 1.

The only remaining cases are

$$\mathbf{D}_1(1,1) > 1, \qquad \mathbf{D}_1(2,2) \leq 1,$$

$$\mathbf{D}_2(1,1) \leq 1, \qquad \mathbf{D}_2(2,2) > 1,$$

and

$$\mathbf{D}_1(1,1) \leq 1, \qquad \mathbf{D}_1(2,2) > 1,$$

$$\mathbf{D}_2(1,1) > 1, \qquad \mathbf{D}_2(2,2) \leq 1.$$

By a relabeling of the states these two cases are equivalent. Furthermore, Problem 6 is not feasible if $\mathbf{D}_1(1,1)$ or $\mathbf{D}_2(1,1)$ equals 1. If $\mathbf{D}_1(1,1)$ equals 1, since $\mathbf{D}_2(2,2)$ is greater than 1, the transition matrix that stabilizes system 1's estimation error always remains in state 1, $\mathbf{Q}(1,1)$ equals 1 and $\mathbf{Q}(2,2)$ equals 0; however, since $\mathbf{D}_2(1,1)$ is greater than 1, this transition matrix does not stabilize system 2's error. Identical arguments hold if $\mathbf{D}_2(1,1)$ equals 1.

4.5.2 Algorithm to Solve Non-trivial Cases

The only non-trivial case is

$$\mathbf{D}_1(1,1) < 1, \qquad \mathbf{D}_1(2,2) > 1, \tag{4.17}$$

$$\mathbf{D}_2(1,1) > 1, \qquad \mathbf{D}_2(2,2) < 1. \tag{4.18}$$

This case corresponds to each operating mode adequately transmitting one of the system's state, but not the other. In this section, we present an algorithm to test the feasibility of the conditions in eq. (4.15) and eq. (4.16) assuming that the parameter matrices satisfy eq. (4.17) and eq. (4.18). This section's structure is as follows: first, preliminary notation and definitions are presented, second, the algorithm is presented, and third, properties of the algorithm are discussed culminating in a theorem that the algorithm accurately characterizes the feasibility of the conditions in eq. (4.15) and eq. (4.16).

The algorithm below requires only the parameter matrices $\mathbf{D}_1$ and $\mathbf{D}_2$ as input to test the feasibility of Problem 6. The algorithm uses the following functions, defined on the domain d greater than 0 and not equal to 1 and k greater than 0,

$$\mathcal{L}(d,\ k) \stackrel{def}{=} \frac{d}{(d-1)k} - \frac{d+1}{d-1} \qquad \text{and}$$
$$\mathcal{R}(d,\ k) \stackrel{def}{=} \frac{d}{(d-1)(1-k)} - \frac{d+1}{d-1}.$$

To save space in the following iterative algorithm, the iteration index n is suppressed. The variable values at the current time step n, for example $\underline{k}_1(n)$, are denoted without the iteration index $\underline{k}_1$. Variable values at the next time $n+1$, for example $\underline{k}_1(n+1)$, are denoted with a superscript plus sign $\underline{k}_1^+$.

Algorithm 1: Interval Iteration for $l = 2$ and $m = 2$

Initialize $\bar{k}_1 = \frac{1}{\mathbf{D}_1(2,2)+1}$, $\underline{k}_1 = 0$, $\bar{k}_2 = \frac{1}{\mathbf{D}_2(1,1)+1}$, $\underline{k}_1 = 0$.

while *true* **do**

1. Find new $(\underline{k}_1^+, \bar{k}_1^+)$ bounds.

   Solve $\mathcal{L}(\mathbf{D}_1(1,1),\ k_1) = \mathcal{R}(\mathbf{D}_2(1,1),\ \underline{k}_2)$ for k_1. $\underline{k}_1^+ = \max\{k_1, \underline{k}_1\}$

   Solve $\mathcal{L}(\mathbf{D}_2(2,2),\ \bar{k}_2) = \mathcal{R}(\mathbf{D}_1(2,2),\ k_1)$ for k_1. $\bar{k}_1^+ = \min\{k_1, \bar{k}_1\}$

2. Find new $(\underline{k}_2^+, \bar{k}_2^+)$ bounds.

   Solve $\mathcal{L}(\mathbf{D}_1(1,1),\ \bar{k}_1) = \mathcal{R}(\mathbf{D}_2(1,1),\ k_2)$ for k_2. $\bar{k}_2^+ = \min\{k_2, \bar{k}_2\}$

   Solve $\mathcal{L}(\mathbf{D}_2(2,2),\ k_2) = \mathcal{R}(\mathbf{D}_1(2,2),\ \underline{k}_1)$ for k_1. $\underline{k}_2^+ = \max\{k_2, \underline{k}_2\}$

3. If $\underline{k}_1 \geq \bar{k}_1$ or $\underline{k}_2 \geq \bar{k}_2$, then return infeasible.

4. Let $k_1^{**} = \frac{\bar{k}_1 + \underline{k}_1}{2}$. If $\exists k_2^{**}$ such that (k_1^{**}, k_2^{**})

   satisfies the feasibility problem in eq. (4.19),

   then return (k_1^{**}, k_2^{**}) .

Algorithm 1 characterizes the feasibility of the conditions in eq. (4.15) and eq. (4.16) as two intervals on $\mathbb{R}$. The algorithm initially over approximates these intervals and iteratively reduces them as needed. If eventually no reduction is possible, the conditions in eq. (4.15) and eq. (4.16) are feasible. If one of the two intervals becomes empty then the conditions are not feasible. The theorem below provides the result that allows two intervals on $\mathbb{R}$ to describe the conditions in eq. (4.15) and eq. (4.16).

Theorem 7 *For l equals 2, m equals 2, and parameter matrices satisfying eq. (4.17) and eq. (4.18), $\boldsymbol{H}$ in $\mathbb{R}_+^{2\times 2}$ satisfies the condition in eq. (4.13) if and only if the following inequalities hold:*

$$\begin{aligned} k_1 &\le \frac{1}{\boldsymbol{D}_1(2,2)+1}, \\ k_2 &\le \frac{1}{\boldsymbol{D}_2(1,1)+1}, \\ \mathcal{L}(\boldsymbol{D}_1(1,1),\ k_1) &\ge \mathcal{R}(\boldsymbol{D}_2(1,1),\ k_2), \\ \mathcal{L}(\boldsymbol{D}_2(2,2),\ k_2) &\ge \mathcal{R}(\boldsymbol{D}_1(2,2),\ k_1), \end{aligned} \tag{4.19}$$

where

$$\begin{aligned} k_1 &\overset{def}{=} \frac{\boldsymbol{H}(1,1)}{\boldsymbol{H}(1,1)+\boldsymbol{H}(2,1)}, \\ k_2 &\overset{def}{=} \frac{\boldsymbol{H}(2,2)}{\boldsymbol{H}(1,2)+\boldsymbol{H}(2,2)}. \end{aligned}$$

Proof 30 *See proof 43 in Section 4.9.*

Theorem 7 determines the feasibility of an $\mathbf{H}$ by two scalar quantities k_1 and k_2. Algorithm 1 searches for feasible intervals for k_1 and k_2. By definition k_1 and k_2

only take positive values. Several properties of the functions $\mathcal{L}$ and $\mathcal{R}$ are needed in the proof that Algorithm 1 accurately reports the feasibility of eq. (4.15) and eq. (4.16).

Lemma 12 *For given d less than 1, $\mathcal{L}(d, k)$ is monotonically increasing and concave on k in $(0, 1)$.*

Proof 31 *See proof 44 in Section 4.9.*

Lemma 13 *For given d greater than 1, $\mathcal{R}(d,\ k)$ is monotonically increasing and convex on k in $(0, 1)$.*

Proof 32 *See proof 45 in Section 4.9.*

Theorem 8 *For l equals 2, m equals 2, $\boldsymbol{S}$ given by eq. (4.14), and parameter matrices satisfying eq. (4.17) and eq. (4.18), Algorithm 1 returns a feasible point to the conditions in eq. (4.19) or gives a certificate of infeasibility.*

Proof 33 *First, we show if the algorithm returns 'infeasible', then eq. (4.19) is infeasible and by Theorem 7 the conditions in eq. (4.13) are infeasible. We show this by proving that there is no feasible choice for k_1 outside the set $(\underline{k}_1(n), \bar{k}_1(n))$. We proceed to show that there is no feasible choice for k_2 outside the set $(\underline{k}_2(n), \bar{k}_2(n))$. These claims are proven recursively. By eq. (4.19) and the definition of initial conditions, there is no feasible k_1 outside the initial interval $(\underline{k}_1(0), \bar{k}_1(0))$ and no feasible k_2 outside the initial interval $(\underline{k}_2(0), \bar{k}_2(0))$. Now to prove the inductive step, assume the claim holds for n. In Algorithm 1, Step 1 removes k_1 for which there is* *no feasible k_2. Step 2 removes k_2 for which there is no feasible k_1. Thus the claim holds for $n+1$.*

Second, we show that if the condition in eq. (4.19) is feasible then the algorithm returns a feasible point $\boldsymbol{H}$ to the conditions in eq. (4.13). This statement is proven by contradiction. Assume that both the infeasibility criterion in Step 3 and the criterion in Step 4 are never met. By the monotone convergence theorem, $\underline{k}_1(n) \to \underline{k}_1^*$, $\bar{k}_1(n) \to \bar{k}_1^*$, $\underline{k}_2(n) \to \underline{k}_2^*$ and $\bar{k}_2(n) \to \bar{k}_2^*$. From steps 1 and 2,

$$\mathcal{L}(\boldsymbol{D}_1(1,1),\ \bar{k}_1^*) = \mathcal{R}(\boldsymbol{D}_2(1,1),\ \bar{k}_2^*),$$

$$\mathcal{L}(\boldsymbol{D}_1(1,1),\ \underline{k}_1^*) = \mathcal{R}(\boldsymbol{D}_2(1,1),\ \underline{k}_2^*),$$

$$\mathcal{L}(\boldsymbol{D}_2(2,2),\ \bar{k}_2^*) = \mathcal{R}(\boldsymbol{D}_1(2,2),\ \bar{k}_1^*),$$

$$\mathcal{L}(\boldsymbol{D}_2(2,2),\ \underline{k}_2^*) = \mathcal{R}(\boldsymbol{D}_1(2,2),\ \underline{k}_1^*).$$

Assume $\underline{k}_1^* = \underline{k}_2^*$ and $\bar{k}_1^* = \bar{k}_2^*$. The case for $\underline{k}_1^* \neq \underline{k}_2^*$ and/or $\bar{k}_1^* \neq \bar{k}_2^*$ uses the same principle arguments.

Since $\mathcal{L}$ is concave and $\mathcal{R}$ is convex

$$\mathcal{L}(\boldsymbol{D}_1(1,1),\ k) > \mathcal{R}(\boldsymbol{D}_2(1,1),\ k),$$

$$\mathcal{L}(\boldsymbol{D}_2(2,2),\ k) > \mathcal{R}(\boldsymbol{D}_1(2,2),\ k).$$

Let $k_1^{**} = \frac{\bar{k}_1^* - \underline{k}_1^*}{2}$ and $k_2^{**} = k_1^{**}$. Thus, by the above set of inequalities (k_1^{**}, k_2^{**}) is a feasible point of eq. (4.19). Thus, eventually the criterion in Step 4 is met, contradicting our assumption that it was never met.

4.6 Algorithm For Problem 6 With Arbitrary l and m

In this Section, we present an algorithm to find a transition matrix $\mathbf{Q}$ between m modes of operation that prioritize the transmissions of l system's state. This algorithm iteratively searches for a feasible $\mathbf{H}$ that satisfies eq. (4.13) by updating a previously infeasible one. In the algorithm and analysis that follow, the updated iterate is denoted by ${}^{+}\mathbf{H}$. The sequence of iterates generated by the algorithm is denoted $\mathbf{H}_n$. Also, the algorithm generates the sequence $\mathbf{Z}_n^*$, which are the minimizing arguments to the optimizations in Step 2 of the below algorithm.

All steps of the algorithm are readily solvable. Step 1 is a linear program and Step 2 can be written as m convex programs with a linear objective, at most $2m$ linear inequality constraints, and a normalization constraint.

Algorithm 2 and Algorithm 1 are very different algorithms. Unfortunately, Algorithm 1 does not extend to the general case. Algorithm 1 is an iterative algorithm that reduces the initially over-estimated feasibility intervals that cover the true feasibility intervals. Algorithm 2 is similar to fixed point algorithms, in that it iteratively updates an initial guess at a feasible solution.

Problem 6 is a difficult non-convex problem. In numerical studies, Algorithm 2 finds an $\mathbf{H}$ that satisfies eq. (4.13) if the condition is feasible. If eq. (4.13) is not feasible, $\mathbf{H}_n$ diverges to infinity. We have not found a case of this algorithm failing to find a solution, if it is known that Problem 6 has a solution. This algorithm does not have a formal guarantee of correctness; however, Algorithm 2 has the following desirable properties discussed in the Sections below:

Algorithm 2: General Algorithm for Problem 6

Initialize $\mathbf{H} > 0$.

while *true* **do**

1. With $\mathbf{H}$ fixed, solve the feasibility problem in $\mathbf{Q}$,

$$\mathbf{Q} \leq \mathbf{S},$$

$$\mathbf{D}_i \mathbf{Q} \mathbf{h}_i - \mathbf{h}_i \leq 0, \qquad i \in \mathbb{L}.$$

   If feasible, exit.

2. For $k \in \mathbb{M}$, solve

$$\beta_k = \min_{\mathbf{z}_k} \mathbf{h}^{(k)} \mathbf{z}_k - \min_{i \in \mathrm{ON}(k)} \mathbf{h}^{(i)} \mathbf{D}^{(k)} \mathbf{z}_k,$$

$$\text{subject to } \mathbf{z}_k \geq 0,$$

$$||\mathbf{z}_k|| \leq 1.$$

3. Set $\beta_k = \max(-\beta_k, 0)$, for $k \in \mathbb{M}$.

4. ${}^{+}\mathbf{h}^{(k)} = \beta_k \mathbf{z}_k^* + \mathbf{h}^{(k)}$, for $k \in \mathbb{M}$,

   where $\mathbf{z}_k^*$ achieves β_k in the optimization

   problem in Step 2.

- If Algorithm 2 converges, it converges to a solution of eq. (4.13) (Section 4.6.1),
- Algorithm 2 is provably correct for l equals 1 (Section 4.6.3),
- Algorithm 2 makes the minimum 2-norm update of $\mathbf{H}_n$ (Section 4.6.2),
- Algorithm 2 scales well with problem size (Above),
- Algorithm 2 can guarantee infeasibility of Problem 6 in specific scenarios (Section 4.6.1).

4.6.1 Convergence and Feasibility

Lemma 14 *If the iterates $\boldsymbol{H}_n$ converge to a $\boldsymbol{H}^*$, then $\boldsymbol{H}^*$ satisfies the conditions in eq.* (4.13).

Proof 34 *Since $\boldsymbol{H}_n \to \boldsymbol{H}^*$, for $k \in \mathbb{M}$, $\beta_k(\boldsymbol{H}_n) \to 0$, where the notation $\beta_k(\boldsymbol{H})$ is used to highlight that β_k is a function of $\boldsymbol{H}$. Note, $\beta_k(\boldsymbol{H})$ is a continuous function, the proof of this relies on the facts that for each $\boldsymbol{z}_k$ the objective function in Step* 2 *is Lipschitz continuous in $\boldsymbol{H}$ with a Lipschitz constant that depends on $\boldsymbol{z}_k$ linearly and that the feasible set of $\boldsymbol{z}_k$ is bounded. So, $\beta_k(\boldsymbol{H}_n) \to \beta_k(\boldsymbol{H}^*) = 0$, which implies $\boldsymbol{H}^*$ satisfies the conditions in Equation* (4.13).

Remark 6 *Often, if $\boldsymbol{H}_n$ converges to $\boldsymbol{H}^*$, there is a neighborhood around $\boldsymbol{H}^*$ that also satisfies the conditions in eq.* (4.13). *Step 1 returns with the first satisfying $\boldsymbol{H}_n$ without waiting for convergence.*

The following lemma identifies a scenario, where $\mathbf{H}_n$ diverging implies the condition in eq. (4.13) is not feasible. This scenario has occurred repeatedly in numerical experimentation.

Definition 12 *The variable $\boldsymbol{Z}_n^*$ enters a limit cycle of matrices in $\{0,1\}^{l\times m}$, if there exists a k' and N such that for n greater than or equal to N, $\boldsymbol{Z}_n^*$ is in $\{0,1\}^{l\times m}$ and*

$$\boldsymbol{Z}_{n+k'}^* = \boldsymbol{Z}_n^*.$$

Lemma 15 *If $\boldsymbol{Z}_n^*$ enters a limit cycle of permutation matrices, then the condition in eq. (4.13) is infeasible.*

Proof 35 *The condition in eq. (4.13) is for all $\boldsymbol{Z} \geq 0$. So, eq. (4.13) must hold for $\boldsymbol{Z}_N^*$, ... , $\boldsymbol{Z}_{N+k'-1}^*$. Consider the system of inequalities created by these k' choices of $\boldsymbol{Z}$. Because these $\boldsymbol{Z}$ are in $\{0,1\}^{l\times m}$, this system of inequalities takes a form similar to eq. (4.13) when $l = 1$, as seen in eq. (4.20). Thus, by following arguments analogous to those presented in Section 4.6.3, we conclude that $\boldsymbol{H}_n$ converges if and only if the condition in eq. (4.13) is feasible. By assumption, $\boldsymbol{H}_n$ diverges since $\boldsymbol{Z}_n^*$ has entered a limit cycle. Therefore, the condition in eq. (4.13) is infeasible.*

4.6.2 Algorithm 2 Updates $\mathbf{H}_n$ in the Minimum Two Norm Direction

The iterate update rule in Step 4 of Algorithm 2 is the minimum potentially feasible $^{+}\mathbf{H}$, defined below, that is also larger than the current $\mathbf{H}$. Observe that replacing $\mathbf{h}_i$ with a positively scaled version $\alpha\mathbf{h}_i$, for α greater than 0, does not change the inequalities in eq. (4.12). So, if the problem in eq. (4.12) is feasible then for any $\mathbf{H}$ in $\mathbb{R}_+^{m\times l}$, there exists a feasible $\mathbf{H}'$ such that

$$\mathbf{H}' \geq \mathbf{H}.$$

Thus with out loss of generality, for any iterate $\mathbf{H}_n$, the search for a feasible $\mathbf{H}'$ is restricted to $\mathbf{H} \geq \mathbf{H}_n$.

For each, β_k less than 0, a known $\mathbf{z}_k^*$ certifies $\mathbf{H}_n$ is infeasible. Therefore, if the new iterate ${}^+\mathbf{h}^{(k)}$ is feasible, it satisfies

$${}^+\mathbf{h}^{(k)}\mathbf{z}_k^* \geq \min_{i\in \mathrm{ON}(k)} {}^+\mathbf{h}^{(i)}D^{(k)}\mathbf{z}_k^* \geq \min_{i\in \mathrm{ON}(k)} \mathbf{h}^{(i)}D^{(k)}\mathbf{z}_k^*.$$

The iterate update rule is the solution to the following problem

$$\begin{aligned} \min_{{}^+\mathbf{h}^k} ||{}^+\mathbf{h}^k||_2 \\ {}^+\mathbf{h}^k > \mathbf{h}^k, \\ {}^+\mathbf{h}^k\mathbf{z}_k^* \geq \min_{i\in \mathrm{ON}(k)} \mathbf{h}^{(i)}D^{(k)}\mathbf{z}_k^*, \end{aligned}$$

which has the solution

$${}^+\mathbf{h}^k = \beta_k\mathbf{z}_k^* + \mathbf{h}^k.$$

The minimum two norm update is a good update policy and direction because it increases $\mathbf{h}^k$ towards feasibility with the minimum possible over-shoot. An update in the direction of $\mathbf{z}_n^*$ that overshoots the minimum 2-norm update can cause $\mathbf{H}_n$ to diverge, even if eq. (4.13) has a feasible solution.

Remark 7 *We could have chosen to update $\boldsymbol{h}^k$ in a direction other than $\boldsymbol{z}_n^*$. For all other iterate update policies tested an example was constructed where the condition* *in eq. (4.13) is feasible but the iterates $\boldsymbol{H}_n$ diverged. No such example is known for the minimum two-norm update policy. In applications of Bender's decomposition techniques, selecting different cuts, which is analogous to selecting an update direction, has an impact on both convergence [123] and convergence rate [126].*

Remark 8 *A minimum step size parameter α can be introduced by replacing Step 3 with the logic:*

$$\beta_k = \begin{cases} \max(-\beta_k,\ \alpha) & \text{if } \beta_k < 0, \\ 0 & \text{otherwise.} \end{cases}$$

The minimum step size parameter α has an impact on the speed of convergence. In numerical experimentation, a minimum step size of .1 was much smaller than minimum step sizes that had an impact whether or not the algorithm converged. A minimum step size of .1 is used for the numerical examples presented.

4.6.3 For $l = 1$, Algorithm 2 Converges if and only if Problem 6 is Feasible

In Section 4.3, Problem 6 for l equals 1 is re-written as a SDP. Although, this case is solved by that SDP, it is important that the algorithm for the general, non-convex Problem 6 reduces to a provably correct algorithm for this case. With l equal to 1, the condition in eq. (4.13) is equivalent to the statement that there exists an $\mathbf{H}$ in $\mathbb{R}_+^{m\times 1}$ such that the following inequality holds,

$$\mathbf{H}(k,1) \geq \mathbf{D}(k,k) \min_{i\in \mathrm{ON}(k)} \mathbf{H}(i,1), \tag{4.20}$$

for k in $\mathbb{M}$. Step 2 of Algorithm 2 simplifies to

$$\mathbf{z}_k = \begin{cases} 0 & \mathbf{H}(k,1) \geq \mathbf{D}(k,k) \min_{i \in \mathrm{ON}(k)} \mathbf{H}(i,1), \\ \\ 1 & \text{Otherwise,} \end{cases} \tag{4.21}$$

for k in $\mathbb{M}$. Step 4 simplifies to

$$\mathbf{H}(k,1) = \max\left(\mathbf{H}(k,1), \mathbf{D}(k,k) \min_{i \in \mathrm{ON}(k)} \mathbf{H}(i,1)\right) \tag{4.22}$$

for k in $\mathbb{M}$.

Lemma 16 *For l equals 1, if*

$$\boldsymbol{H'} \geq \boldsymbol{H}$$

then

$$^{+}\boldsymbol{H'} \geq {}^{+}\boldsymbol{H}.$$

Proof 36 *If $\boldsymbol{H'} \geq \boldsymbol{H}$, then for k in $\mathbb{M}$*

$$\boldsymbol{D}(k,k) \min_{i \in \mathrm{ON}(k)} \boldsymbol{H'}(i,1) \geq \boldsymbol{D}(k,k) \min_{i \in \mathrm{ON}(k)} \boldsymbol{H}(i,1).$$

This along with $\boldsymbol{H'}(k,1) > \boldsymbol{H}(k,1)$, yields

$$^{+}\boldsymbol{H'}(k,1) \geq {}^{+}\boldsymbol{H}(k,1)$$

for k in $\mathbb{M}$.

Theorem 9 *For l equals 1, $\boldsymbol{H}_n$ converges if and only if the condition in eq. (4.13) is feasible.*

Proof 37 *By Lemma 14, if $\boldsymbol{H}_n$ converges then the condition in eq. (4.13) is feasible.*

If there exists an $\boldsymbol{H}^*$ that satisfies the conditions in eq. (4.13) then for any initialization point $\boldsymbol{H}(0)$, we select an α such that $\alpha\boldsymbol{H}^* \geq \boldsymbol{H}(0)$. Consider an execution of Algorithm 2 with the alternative initialization point $\boldsymbol{H}'(0) = \alpha\boldsymbol{H}^*$. By the monotone property of Lemma 16 at iteration n,

$$\boldsymbol{H}'_n \geq \boldsymbol{H}_n.$$

Since $\boldsymbol{H}'(0)$ satisfies the conditions in eq. (4.13), $\boldsymbol{H}'_n = \alpha\boldsymbol{H}^*$ for all n. $\boldsymbol{H}_n$ is a nondecreasing sequence that is bounded above, $\alpha\boldsymbol{H}^* \geq \boldsymbol{H}_n$. Thus we conclude $\boldsymbol{H}_n$ converges.

4.7 Examples

Two numerical examples are given that demonstrate the problem formulation and we detail a typical execution time of Algorithm 1, Algorithm 2, and for comparison a branch and bound approach. The numerical tests were performed on a 2015 Macbook Air and the algorithms are implemented in Matlab.

4.7.1 Example monitoring two systems ($l = 2$) and a two state Markov chain ($m = 2$)

The example shown in Figure 4.2 is used for this numerical example. The zero pattern structure of this Markov chain is

$$\mathbf{S} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

The diagonal parameter matrices are

$$\mathbf{D}_1 = \begin{bmatrix} .2 & 0 \\ 0 & 2 \end{bmatrix} \quad \text{and} \quad \mathbf{D}_2 = \begin{bmatrix} 2 & 0 \\ 0 & .55 \end{bmatrix}.$$

The chart in Table 4.1 lists each algorithms execution time for this example. A stabilizing transition matrix exists for these parameter matrices: an illustrative stabilizing transition matrix is

$$\mathbf{Q} = \begin{bmatrix} .027 & .973 \\ .663 & .337 \end{bmatrix}.$$

4.7.2 Example monitoring 6 systems ($l = 6$) and a 38 state Markov chain ($m = 38$)

Systematically designing human operator interaction and support systems is an emerging application area [20]. Human phenomena impacting an operator's performance such as performance degradation with low or high workloads [13] or

	Execution Time (seconds)
Algorithm 1	2.6
Algorithm 2	6.6
Branch and bound	14.2
Grid refine	23.6
Uniform random search	381.9

Table 4.1: Execution times of each algorithm for the example discussed in the Section 4.2.3 of monitoring two systems, l equals 2, by a transmission system with two states, m equals 2. For the random algorithm the average execution time over ten executions is shown.

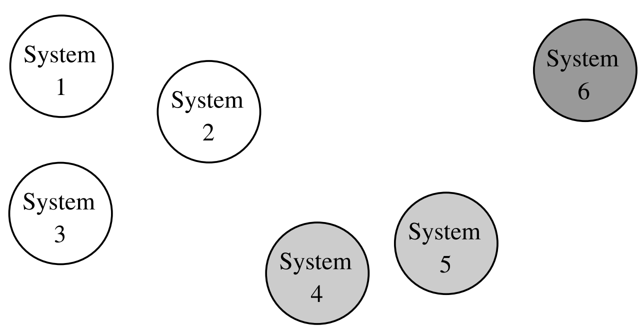

Figure 4.4: An example of monitoring 6 geographically dispersed systems. Systems 1, 2, and 3 are closely located. Systems 4 and 5 are also closely located. See Section 4.7.2 for details of the operating mode Markov chain that models a human operator monitoring these systems.

	Execution Time
Algorithm 2	477.8 secs
Branch and bound	$>$ 3hrs
Grid refine	$>$ 3hrs
Uniform random search	$>$ 3hrs

Table 4.2: Execution times of each algorithm for a human operator attention allocation problem with a model for situational awareness and workload. Given the large dimensions of the problem, the branch and bound, grid refine, and uniform random search algorithms were terminated after 3 hours of execution. These calculations were performed on a 2015 Macbook Air.

situational awareness [31] can be modeled using a Markov chain with a specific allowable graph structure [43].

A human operator attention allocation design problem is considered for this numerical example. A human operator is tasked with monitoring the location of 6 targets by performing visual search tasks. Target's 1, 2, and 3 are near each other. Target's 4 and 5 are near each other but distant from other targets, see Figure 4.4. Due to proximity and terrain changes, if the operator focuses on locating target 1 there is a modest probability that the operator will also locate target 2 or 3. If the operator focuses on locating target 5, there is a modest probability that the operator will also locate target 4. Similarly, if the operator focuses attention on locating one of the other targets.

The sensor state Markov chain is constructed to model human operator characteristics. A sensor Markov chain state exists for each target representing the operator's attention is focused on locating that target. Due to operator situational awareness, operator performance temporarily degrades as attention changes focus. This is modeled by a sensor state with higher drop probabilities when transitioning away from focusing on a target. Operator workload also affects performance. Sensor states are added that model a low, medium and high workload. Each workload level increases the sensor drop probability. These workload states are added to each of the states associated with focusing on a target as well as the situational awareness states. This example has 38 sensor Markov chain states and a moderately sparse $\mathbf{S}$ in $\mathbb{R}^{n \times n}$.

Table 4.2 lists the execution time of each algorithm for this example. Note that the branch and bound algorithm is searching in a thirty eight dimensioned hypercube.

Remark 9 *Although a framework is presented in [110], [111], and [112] for a branch and bound algorithm specifically designed for an optimization problem with a bilinear objective function and bilinear inequality constraints; substantial analysis and algorithm development were required to produce a viable branch and bound algorithm for comparison. These works have two critical shortcomings: 1) the suggested manner to produce an upper-bound does not account for the feasibility of that proposed upper-bound, 2) the suggested branching method does not properly account for product spaces. In the branch and bound algorithm developed for comparison, these shortcomings are overcome by the following. 1) Problem 6 can be written as a pro-*

gram with a linear objective and bilinear inequality constraints with slack variables, exploiting these slack variables, we can produce an upper-bound in the feasible set. 2) A branching method that properly traverses the regions for product spaces is used.

4.8 Conclusions

In this chapter, we considered the problem of designing a Markov chain transition matrix that governs the transmission mode of operation used to monitor multiple systems. Algorithmic solutions were presented for several cases that design stabilizing transition matrices if possible and certify infeasibility of the problem otherwise. An algorithm for the general problem is presented that scales well with problem size, and make efficient updates. These algorithms are compared to a branch and bound method in two example problems.

4.9 Proofs

Proof 38 (Lemma 10) *The recursion of system i in eq. (4.8) is re-written*

$$Y_n^i(B_{1:n}) = \mathrm{trace}(\boldsymbol{\Sigma}) \sum_{j=0}^{n-1} \rho^{2j}(\boldsymbol{A}_i) \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v).$$

Also, the expected estimation error for system i conditioned on the operating mode history can be written as

$$V_n^i(B_{1:n}) = \mathrm{trace}(E[(\boldsymbol{X}_n^i - \hat{\boldsymbol{X}}_n^i)(\boldsymbol{X}_n^i - \hat{\boldsymbol{X}}_n^i)^T | B_{1:n}]).$$

By Lemma 17, the limit of the expected estimation error is bounded

$$\lim_{n\to\infty} E_B[V_n^i(B_{1:n})] < \infty$$

if and only if the following limit is bounded

$$\lim_{n\to\infty} \frac{1}{\text{trace}(\mathbf{\Sigma})} E_B[Y_n^i(B_{1:n})] < \infty.$$

Let the error covariance conditioned on the operating mode history be denoted as

$$\mathbf{\Phi}_n^i(B_{1:n}) \stackrel{def}{=} E[(\mathbf{X}_n^i - \hat{\mathbf{X}}_n^i)(\mathbf{X}_n^i - \hat{\mathbf{X}}_n^i)^T | B_{1:n}].$$

Lemma 17 *The expected mean-squared error limit is bounded*

$$\lim_{n\to\infty} E_B[\text{trace}(\mathbf{\Phi}_n^i(B_{1:n}))] < \infty$$

if and only if the following limit is bounded

$$\lim_{n\to\infty} E_B\left[\sum_{j=0}^{n-1} \rho^{2j}(\boldsymbol{A}_i) \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v)\right] < \infty.$$

Proof 39 *The error covariance matrix for the non-scalar system i follows the recursive dynamics*

$$\mathbf{\Phi}_{n+1}^i(B_{1:n+1}) = \boldsymbol{p}^i(B_{n+1}) \boldsymbol{A}_i \mathbf{\Phi}_n^i(B_{1:n}) \boldsymbol{A}_i^T + \mathbf{\Sigma}$$

which are re-written as

$$\mathbf{\Phi}_n^i(B_{1:n}) = \sum_{j=0}^{n-1} \left[\boldsymbol{A}_i^j \mathbf{\Sigma} (\boldsymbol{A}_i^T)^j \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v)\right].$$

Taking the trace of both sides yields

$$\text{trace}(\mathbf{\Phi}_n^i(B_{1:n})) = \sum_{j=0}^{n-1} \left[\text{trace}(\boldsymbol{A}_i^j \mathbf{\Sigma} (\boldsymbol{A}_i^T)^j) \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v)\right].$$

Since $\mathbf{\Sigma}$ is symmetric and positive definite there exists a $\boldsymbol{U}$ with orthonormal columns and a positive definite diagonal matrix $\mathbf{\Lambda}$ such that $\mathbf{\Sigma} = \boldsymbol{U}\mathbf{\Lambda}\boldsymbol{U}^T$. The k^{th} column of $\boldsymbol{U}$ is written in the basis of orthonormal eigenvectors $\boldsymbol{v}_t$ of $\boldsymbol{A}_i$ as $\sum_t \alpha_{t,k}\boldsymbol{v}_t$ for appropriate $\alpha_{t,k}$. In this notation,

$$\mathrm{trace}(\boldsymbol{A}_i^j \mathbf{\Sigma}(\boldsymbol{A}_i^T)^j) = \sum_k \mathbf{\Lambda}(k,k) \sum_t \lambda_t^{2j} \alpha_{t,k}^2,$$

where λ_t is the eigenvalue of $\boldsymbol{A}_i$ associated with eigenvector $\boldsymbol{v}_t$. So by rearranging the finite summations the trace of the conditional error covariance $\mathrm{trace}(\mathbf{\Phi}_n^i(B_{1:n}))$ *equals,*

$$\sum_k \mathbf{\Lambda}(k,k) \sum_t \alpha_{t,k}^2 \left[\sum_{j=0}^{n-1} \lambda_t^{2j} \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v) \right].$$

Assumption 3 implies that for each t there exists a k such that $\alpha_{t,k}$ is not zero. Thus, the limit of the expected estimation error is bounded

$$\lim_{n\to\infty} E_B[\mathrm{trace}(\mathbf{\Phi}_n^i(B_{1:n}))] < \infty$$

if and only if for each t, the following limits are bounded

$$\lim_{n\to\infty} E_B \left[\sum_{j=0}^{n-1} \lambda_t^{2j} \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v) \right] < \infty.$$

The proof is completed by noting that for some t', $\lambda_{t'} = \rho(\boldsymbol{A}_i)$ and thus

$$E_B \left[\sum_{j=0}^{n-1} \rho^{2j}(\boldsymbol{A}_i) \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v) \right]$$

is an upper bound for $E_B[\sum_{j=0}^{n-1} \lambda_t^{2j} \prod_{v=n-j}^{n-1} \boldsymbol{p}^i(B_v)]$ with any other $t \neq t'$.

Lemma 18 *For an entry-wise non-negative matrix $\boldsymbol{A} \geq 0$, $\rho(\boldsymbol{A}) \leq 1$ if and only if an entry-wise positive vector $\boldsymbol{h} > 0$ exists such that $\boldsymbol{A}\boldsymbol{h} \leq \boldsymbol{h}$.*

Proof 40 *If $\rho(\boldsymbol{A}) \leq 1$ then $\boldsymbol{Ah} \leq \boldsymbol{h}$ for all $\boldsymbol{h}$. So clearly, $\boldsymbol{Ah} \leq \boldsymbol{h}$ for $\boldsymbol{h} > 0$ holds.*

The other direction is [127, Theorem 2.1.11].

Farkas Lemma is stated for reference.

Lemma 19 (Farkas Lemma)

$$\boldsymbol{Ax} \leq \boldsymbol{b} \qquad \boldsymbol{Cx} = \boldsymbol{d}$$

and

$$\boldsymbol{z} \geq 0, \qquad \boldsymbol{A}^T\boldsymbol{z} + \boldsymbol{C}^T\boldsymbol{y} = 0$$

$$\text{and} \qquad \boldsymbol{b}^T\boldsymbol{z} + \boldsymbol{d}^T\boldsymbol{y} < 0$$

are strong alternatives, meaning one system of equations is feasible and the other is not feasible.

Proof 41 *See [128, Section 4.7]. This specific form of Farkas lemma is found in [129, Page 5-5].*

Proof 42 (Theorem 6) *Write the transition matrix in terms of its allowable non-zero entries using the following notation: $\boldsymbol{Q} = \sum_{v=1}^{n_q} \boldsymbol{q}(v)\mathbb{Q}_v$, where n_q is the number of possible non-zero entries of $\boldsymbol{Q}$, $\boldsymbol{q}$ is a vector and $\mathbb{Q}_v = \boldsymbol{e}_j\boldsymbol{e}_t^T$ for appropriate j and t such that $\boldsymbol{Z}(j,t) > 0$. The system of eq. (4.11) is written as*

$$\sum_v \boldsymbol{q}(v)\boldsymbol{D}_i\mathbb{Q}_v\boldsymbol{h}_i \leq \boldsymbol{h}_i, \qquad i \in \mathbb{L},$$

or separating each row of this entry-wise inequality yields

$$\sum_v \boldsymbol{q}(v)\boldsymbol{e}_k^T\boldsymbol{D}_i\mathbb{Q}_v\boldsymbol{h}_i \leq \boldsymbol{e}_k^T\boldsymbol{h}_i, \qquad i \in \mathbb{L},\ k \in \mathbb{M}.$$

This system of equations is stacked as follows

$$\boldsymbol{q}^T \begin{bmatrix} \boldsymbol{e}_k^T \boldsymbol{D}_i \mathbb{Q}_1 \\ \vdots \\ \boldsymbol{e}_k^T \boldsymbol{D}_i \mathbb{Q}_{n_q} \end{bmatrix} \boldsymbol{h}_i \leq \boldsymbol{e}_k^T \boldsymbol{h}_i, \qquad i \in \mathbb{L},\ k \in \mathbb{M}.$$

By the appropriate definition of $\boldsymbol{V}_{i,k}$ this is written as the bilinear system,

$$\boldsymbol{q}^T \boldsymbol{V}_{i,k} \boldsymbol{h}_i - \boldsymbol{e}_k^T \boldsymbol{h}_i \leq 0, \qquad i \in \mathbb{L},\ k \in \mathbb{M}.$$

With the $\boldsymbol{h}_i$ fixed, apply Farkas lemma to the following feasibility problem: does there exist $\boldsymbol{q}$ such that

$$\begin{aligned} &\boldsymbol{q}^T \boldsymbol{V}_{i,k} \boldsymbol{h}_i \leq \boldsymbol{e}_k^T \boldsymbol{h}_i, \qquad i \in \mathbb{L},\ k \in \mathbb{M}, \\ &\boldsymbol{q} \geq 0, \\ &\boldsymbol{f}_j^T \boldsymbol{q} = 1, \qquad j \in \mathbb{M}, \end{aligned} \tag{4.23}$$

where

$$\boldsymbol{f}_j \stackrel{def}{=} \begin{bmatrix} \boldsymbol{0}_{n_1} \\ \vdots \\ \boldsymbol{1}_{n_j} \\ \vdots \\ \boldsymbol{0}_{n_m} \end{bmatrix},$$

and n_j is the number of parameters on the j^{th} row of $\boldsymbol{Q}$. So $n_q = \sum_1^m n_j$. The strong alternative to this system of equations is the following feasibility problem: does there *exist $\boldsymbol{Z}, \boldsymbol{\lambda}, \boldsymbol{\gamma}$ such that*

$$\boldsymbol{Z} \geq 0, \tag{4.24}$$

$$\boldsymbol{\lambda} \geq 0, \tag{4.25}$$

$$\sum_{i,k} (\boldsymbol{V}_{i,k}\boldsymbol{h}_i)\boldsymbol{Z}(i,k) - \boldsymbol{\lambda} + \sum_j \boldsymbol{f}_j \boldsymbol{\gamma}(j) = 0, \tag{4.26}$$

$$\sum_{i,k} \boldsymbol{h}_i(k)\boldsymbol{Z}(i,k) + 0^T\boldsymbol{\lambda} + \sum_j \boldsymbol{\gamma}(j) < 0. \tag{4.27}$$

This feasibility problem is infeasible if and only if $\forall \boldsymbol{Z} \geq 0$, $\forall \boldsymbol{\lambda} \geq 0$, and $\forall \boldsymbol{\gamma}$ that satisfy

$$\sum_{i,k} (\boldsymbol{V}_{i,k}\boldsymbol{h}_i)\boldsymbol{Z}(i,k) - \boldsymbol{\lambda} + \sum_j \boldsymbol{f}_j \boldsymbol{\gamma}(j) = 0,$$

also satisfy

$$\sum_{i,k} \boldsymbol{h}_i(k)\boldsymbol{Z}(i,k) + \sum_j \boldsymbol{\gamma}(j) \geq 0.$$

Note that the feasibility problem restricted to eq. (4.24), eq. (4.25), and eq. (4.26) is always feasible by an appropriate choice of $\boldsymbol{\lambda}$. The above condition is equivalent to the condition that $\forall \boldsymbol{Z} \geq 0$, and $\forall \boldsymbol{\gamma}$ that satisfy

$$\sum_{i,k} (\boldsymbol{V}_{i,k}\boldsymbol{h}_i)\boldsymbol{Z}(i,k) + \sum_j \boldsymbol{f}_j \boldsymbol{\gamma}(j) \geq 0,$$

also satisfy

$$\sum_{i,k} \boldsymbol{h}_i(k)\boldsymbol{Z}(i,k) + \sum_j \boldsymbol{\gamma}(j) \geq 0. \tag{4.28}$$

Thus by Farkas lemma the system of eq. (4.23) is feasible if and only if the system of eq. (4.24) is not feasible. The system of eq. (4.24) is not feasible if and *only if the conditions in eq. (4.28) hold. So the system of eq. (4.23) is feasible if and only if the conditions in eq. (4.28) hold.*

Define the optimization problem

$$\Gamma(\boldsymbol{H}) \stackrel{def}{=} \min_{\boldsymbol{Z},\boldsymbol{\gamma}} \sum_{i,k} \boldsymbol{h}_i(k)\boldsymbol{Z}(i,k) + \sum_j \boldsymbol{\gamma}(j), \tag{4.29}$$

subject to $\boldsymbol{Z} \geq 0$,

$$\sum_{i,k} (\boldsymbol{V}_{i,k}\boldsymbol{h}_i)\boldsymbol{Z}(i,k) + \sum_j \boldsymbol{f}_j \boldsymbol{\gamma}(j) \geq 0. \tag{4.30}$$

If the condition in eq. (4.28) holds, then $\Gamma(\boldsymbol{H}) = 0$*. If the condition in eq. (4.28) does not hold,* $\Gamma(\boldsymbol{H}) = -\infty$*.*

By using the structure of $\boldsymbol{V}_{i,k}$ *and* $\boldsymbol{f}_j$*, the constraint in eq. (4.30) is written as the* m *separate constraints*

$$\boldsymbol{\gamma}(k) \geq \max_{t \in \mathrm{ON}(k)} -\boldsymbol{h}^{\{t\}} D^{\{i\}} \boldsymbol{z}_k, \tag{4.31}$$

for $k \in \mathbb{M}$*, where* $\mathrm{ON}(k)$ *denotes the columns of the non-zero entries in the* k^{th} *row of* $\boldsymbol{Z}$ *or equivalently the Markov chain states with an incoming arc that originated from state* k*, out neighbors. The above statement is justified by recalling each* w *parameter of* $\boldsymbol{Q}$ *corresponds to specific* j *and* t *such that* $\mathbb{Q}_w = \boldsymbol{e}_j \boldsymbol{e}_t^T$*. If* $j \neq k$ *then the* j^{th} *row of* $\boldsymbol{V}_{i,k}$ *is*

$$\boldsymbol{e}_k^T \boldsymbol{D}_i \mathbb{Q}_w = \boldsymbol{0}_m^T.$$

If $j = k$ *then the* j^{th} *row of* $\boldsymbol{V}_{i,k}$ *is*

$$\boldsymbol{e}_k^T \boldsymbol{D}_i \boldsymbol{e}_k \boldsymbol{e}_t^T = \boldsymbol{D}_i(k,k)\boldsymbol{e}_t^T.$$

Additionally, if $\boldsymbol{f}_j(w) = 0$ *then* $\mathbb{Q}_w = \boldsymbol{e}_v \boldsymbol{e}_t^T$ *with* $v \neq j$*. If* $\boldsymbol{f}_j(w) = 1$ *then* $\mathbb{Q}_w = \boldsymbol{e}_j \boldsymbol{e}_t^T$ *with some* t*. Thus, the constraint in Equation (4.30) becomes* m *separate constraints*

$$\boldsymbol{f}_k \boldsymbol{\gamma}(k) \geq -\sum_i (\boldsymbol{V}_{i,k} \boldsymbol{h}_i) \boldsymbol{Z}(i,k), \qquad k \in \mathbb{M}.$$

Writing this entry-wise vector inequality as a scalar inequalities and removing the zeros, leaves the following set of inequalities: for each $k \in \mathbb{M}$ *and* $t \in \mathrm{ON}(k)$

$$\boldsymbol{\gamma}(k) \geq -\sum_i \boldsymbol{D}_i(k,k) \boldsymbol{h}_i(t) \boldsymbol{Z}(i,k), \qquad k \in \mathbb{M}.$$

Since $\boldsymbol{h}_i(t) = \boldsymbol{H}(t,i)$*, the summation on the right hand side of the above equation equals*

$$\sum_i \boldsymbol{D}_i(k,k) \boldsymbol{h}_i(t) \boldsymbol{Z}(i,k) = \boldsymbol{h}^{\{t\}} \boldsymbol{D}^{\{i\}} \boldsymbol{z}_k.$$

Thus Equation (4.31) follows from the structure of $\boldsymbol{V}_{i,k}$ *and* $\boldsymbol{f}_j$*.*

In problem $\Gamma(\boldsymbol{H})$*, given the constraints in eq. (4.31), minimizing* $\boldsymbol{\gamma}(k)$ *for fixed* $\boldsymbol{Z}$ *leads to an optimal* $\boldsymbol{\gamma}^*(k)$*,*

$$\boldsymbol{\gamma}(k)^* = -\min_{t \in \mathrm{ON}(k)} \boldsymbol{h}^{\{t\}} \boldsymbol{D}^{\{i\}} \boldsymbol{z}_k$$

Note that since $\boldsymbol{h}_i(k) = \boldsymbol{H}(k,i)$*, the following summation equals*

$$\sum_i \boldsymbol{h}_i(k) \boldsymbol{Z}(i,k) = \boldsymbol{h}^{\{k\}} \boldsymbol{z}_k.$$

Thus the optimization problem $\Gamma(\boldsymbol{H})$ *becomes*

$$\min_{\boldsymbol{Z} \geq 0} \sum_{k=1}^{m} \left[\boldsymbol{h}^{\{k\}} \boldsymbol{z}_k - \min_{t \in \mathrm{ON}(k)} \boldsymbol{h}^{\{t\}} \boldsymbol{D}^{\{i\}} \boldsymbol{z}_k \right].$$

The optimization problem $\Gamma(\boldsymbol{H})$ *can be separated into the sum of* m *separate optimizations as* $\Gamma(\boldsymbol{H}) = \sum_k \Gamma_k(\boldsymbol{H})$ *where for* $k \in \mathbb{M}$ *these optimizations are defined* *by*

$$\Gamma_k(\boldsymbol{H}) \stackrel{def}{=} \min_{\boldsymbol{z}_k \geq 0} \left[\boldsymbol{h}^{\{k\}} \boldsymbol{z}_k - \min_{t \in \mathrm{ON}(k)} \boldsymbol{h}^{\{t\}} \boldsymbol{D}^{\{i\}} \boldsymbol{z}_k \right].$$

Since the objective function of $\Gamma_k(\boldsymbol{H})$ *is linear in* $\boldsymbol{z}_k$*, either* $\Gamma_k(\boldsymbol{H}) = 0$ *or* $-\infty$.

The condition in Equation (4.28) *holds and hence the system of Equation* (4.23) *is feasible if and only if* $\Gamma(\boldsymbol{H}) = 0$ *or equivalently for* $k \in \mathbb{M}$

$$\Gamma_k(\boldsymbol{H}) = 0.$$

Proof 43 (Theorem 7) *For the condition in Equation* (4.13) *without loss of generality, assume the column's of* $\boldsymbol{H}$ *and* $\boldsymbol{Z}$ *sum to* 1*. Note* k_1 *and* k_2 *are the entries* $\boldsymbol{H}(1,1)$ *and* $\boldsymbol{H}(2,2)$*, respectively, after the column's of the matrix* $\boldsymbol{H}$ *has been normalized. This is without loss of generality since in Equation* (4.13) *the column's of* $\boldsymbol{Z}$ *can be multiplied by any positive number without affecting the inequality. Similarly in Equation* (4.12) *the column's of* $\boldsymbol{H}$ *can be arbitrarily scaled and by Theorem 6 this scaled* $\boldsymbol{H}$ *satisfies Equation* (4.13) *if and only if the original* $\boldsymbol{H}$ *satisfied Equation* (4.12)*. We use the convention that the diagonal matrix entry is used to define the column. For example,*

$$\boldsymbol{z}_1 = \begin{bmatrix} \boldsymbol{Z}(1,1) \\ 1 - \boldsymbol{Z}(1,1) \end{bmatrix}.$$

Under the assumption that $\boldsymbol{D}_1(1,1) < 1$ *and* $\boldsymbol{D}_2(1,1) \geq 1$*, we show that*

$$\boldsymbol{h}^{(1)} \boldsymbol{z}_1 \geq \min\left(\boldsymbol{h}^{(1)} \boldsymbol{D}^{(1)} \boldsymbol{z}_1, \boldsymbol{h}^{(2)} \boldsymbol{D}^{(1)} \boldsymbol{z}_1 \right), \tag{4.32}$$

for all $\boldsymbol{z}_1 \geq 0$ *if and only if*

$$k_2 \leq \frac{1}{\boldsymbol{D}_2(1,1)+1},$$

$$\mathcal{L}(\boldsymbol{D}_1(1,1),k_1) \geq \mathcal{R}(\boldsymbol{D}_2(1,1),k_2).$$

Note eq. (4.32) *holds for all* $\boldsymbol{z}_1 \geq 0$ *if and only if*

$$\boldsymbol{h}^{(1)}\boldsymbol{z}_1 \geq \boldsymbol{h}^{(2)}\boldsymbol{D}^{(1)}\boldsymbol{z}_1,$$

when $\boldsymbol{h}^{(1)}\boldsymbol{z}_1 < \boldsymbol{h}^{(1)}\boldsymbol{D}^{(1)}\boldsymbol{z}_1$. *The conditional*

$$\boldsymbol{h}^{(1)}\boldsymbol{z}_1 < \boldsymbol{h}^{(1)}\boldsymbol{D}^{(1)}\boldsymbol{z}_1$$

is equivalent to

$$\boldsymbol{Z}(1,1) < \frac{\boldsymbol{H}(1,2)[\boldsymbol{D}_2(1,1)-1]}{\boldsymbol{H}(1,1)[1-\boldsymbol{D}_1(1,1)]+\boldsymbol{H}(1,2)[\boldsymbol{D}_2(1,1)-1]}. \tag{4.33}$$

So, eq. (4.32) *holds for all* $\boldsymbol{z}_1 \geq 0$ *if and only if*

$$\boldsymbol{h}^{(1)}\boldsymbol{z}_1 \geq \boldsymbol{h}^{(2)}\boldsymbol{D}^{(1)}\boldsymbol{z}_1, \tag{4.34}$$

for $\boldsymbol{Z}(1,1)$ *in the range from* 0 *to this upper bound. Since both sides of Equation* (4.34) *are linear in* $\boldsymbol{Z}(1,1)$, *this inequality holds on the specific* $\boldsymbol{Z}(1,1)$ *range if and only if the inequality holds at the both end points of the range. Evaluating eq.* (4.34) *at* $\boldsymbol{Z}(1,1) = 0$ *yields*

$$k_2 \leq \frac{1}{\boldsymbol{D}_2(1,1)+1}.$$

Evaluating eq. (4.34) *at the right-hand side of equation eq.* (4.33) *yields*

$$\mathcal{L}(\boldsymbol{D}_1(1,1),k_1) \geq \mathcal{R}(\boldsymbol{D}_2(1,1),k_2).$$

Analogous arguments hold for the inequality in eq. (4.16).

Proof 44 (Lemma 12) *Since the derivative*

$$\frac{d}{dk}\mathcal{L}(d,k) = \frac{-d}{(d-1)k^2} > 0,$$

$\mathcal{L}(d,\cdot)$ is monotonically increasing. Also, the second derivative

$$\frac{d^2}{d^2k}\mathcal{L}(d,k) = \frac{2d}{(d-1)k^3} < 0$$

on $k \in (0,1)$ implying that $\mathcal{L}(d,\cdot)$ is concave.

Proof 45 (Lemma 13) *Since the derivative*

$$\frac{d}{dk}\mathcal{R}(d,k) = \frac{d}{(d-1)(1-k)^2} > 0,$$

$\mathcal{R}(d,\cdot)$ is monotonically increasing. Also, the second derivative

$$\frac{d^2}{d^2k}\mathcal{R}(d,k) = \frac{2d}{(d-1)(1-k)^3} > 0$$

on $k \in (0,1)$ implying that $\mathcal{R}(d,\cdot)$ is convex.

Chapter 5: Conclusion

The application of the results in this disseration to the design of operator support systems that account for human performance factors such as workload, speed-accuracy tradeoffs, biasing, and situational awareness are detailed. Maturing communication, sensing, and computing technologies have enabled the development of systems and tools that have expanded human capabilities. A human operator is now capable of piloting an air-craft from thousands of miles away, precisely controlling a nuclear reactor, or monitoring activity across vast regions. Optimal performance of the system depends on a well-constructed relationship with the human operator. The systematic design of this relationship is an important area of research sometimes known as human-in-the-loop. Techniques used in optimization, control, and communications can be of great value to the design of human operator support systems; however, traditional assumptions underpinning these theories may not hold for operator support system design, leading to new areas of research.

This dissertation investigates remote estimation problems over a channel where channel performance is determined by previous channel usage. Specifically, the channel's performance is determined by the state of a controlled Markov chain whose input is the channel usage. Two research directions are presented. The first includes the design of transmission and encoding policies to minimize estimation error when the channel state dynamics are fixed. The second research direction involves a problem where the channel's Markov chain structure is fixed but the transition probabilities are not and are designed to stabilize the estimation error. In the operator support system design application, the operator's decisions are modeled by the ouput of the faulty channel. The operator's state and its impact on operator performance is modeled by the channel state dynamics and the channels state's impact on channel performance.

For the first research direction, two specific problems formulations are presented: 1) Find optimal channel policies that minimize the mean squared error for use-dependent packet-drop channels, and 2) Characterize the information maximizing channel policies for a use-dependent discrete switching channel whose statistics switch in accordance with the channel state.

For the second research direction the problem of remotely estimating the state of multiple systems transmitted over separate packet-drop channels where a central channel state, evolving according to a Markov chain, determines the individual drop probabilities for all the packet-drop channels is investigated. An algorithm is presented that designs the transition matrix of the channel's Markov chain to stabilize the estimation error. This problem has application to operator attention allocation where the central Markov chain state determines the current focus of the operator's attention and the transition probabilities determine where the operator should focus next.

Bibliography

[1] Ketan Savla and Emilio Frazzoli. Maximally stabilizing task release control policy for a dynamical queue. *IEEE Transactions on Automatic Control*, 55(11):2655–2660, 2010.

[2] Randall J. Mumaw, Emilie M. Roth, Kim J. Vicente, and Catherine M. Burns. There is more to monitoring a nuclear power plant than meets the eye. *Human Factors*, 42(1):36–55, 2000.

[3] Mary L Cummings, Sylvain Bruni, S Mercier, and PJ Mitchell. Automation architecture for single operator, multiple uav command and control. Technical report, DTIC Document, 2007.

[4] Marc Steinberg. Intelligent autonomy for unmanned naval systems. In *Defense and Security Symposium*, pages 623013–623013. International Society for Optics and Photonics, 2006.

[5] Mica R Endsley. Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics*, 42(3):462–492, 1999.

[6] Vinod Baya, R Cutkosky Mark, et al. Information handling behavior of designers during conceptual design: three experiments. 1996.

[7] Clive L Dym, Alice M Agogino, Ozgur Eris, Daniel D Frey, and Larry J Leifer. Engineering design thinking, teaching, and learning. *Journal of Engineering Education*, 94(1):103–120, 2005.

[8] Celena Dopart, Kimberly Jackson, Michele D Carpenter, Babak E Cohanim, and Jeffrey A Hoffman. A mission guided investigation of operational functions and user interface for mobility augmenting jetpack with integrated cmgs (majic). In *2014 IEEE Aerospace Conference*, pages 1–9. IEEE, 2014.

[9] Laurent Itti and Christof Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3):194–203, 2001.

[10] Michael A Vidulich and Pamela S Tsang. The confluence of situation awareness and mental workload for adaptable human–machine systems. *Journal of Cognitive Engineering and Decision Making*, 9(1):95–97, 2015.

[11] K. Savla and E. Frazzoli. A dynamical queue approach to intelligent task management for human operators. *Proceedings of the IEEE*, 100(3):672–686, March 2012.

[12] V. Srivastava, A. Surana, and F. Bullo. Adaptive attention allocation in human-robot systems. In *American Control Conference (ACC), 2012*, pages 2767–2774, June 2012.

[13] Robert M Yerkes and John D Dodson. The relation of strength of stimulus to rapidity of habit-formation. *Journal of comparative neurology and psychology*, 18(5):459–482, 1908.

[14] CE Nehme, B Mekdeci, JW Crandall, and ML Cummings. The impact of heterogeneity on operator performance in futuristic unmanned vehicle systems. Technical report, DTIC Document, 2009.

[15] Vaibhav Srivastava, Ruggero Carli, Cédric Langbort, and Francesco Bullo. Attention allocation for decision making queues. *Automatica*, 50(2):378–388, 2014.

[16] Axel Schulte and Diana Donath. Measuring self-adaptive uav operators' load-shedding strategies under high workload. In *HCI (21)'11*, pages 342–351, 2011.

[17] Axel Schulte and Diana Donath. Measuring self-adaptive uav operators load-shedding strategies under high workload. In *International Conference on Engineering Psychology and Cognitive Ergonomics*. Springer, 2011.

[18] Glenn F Wilson and Christopher A Russell. Real-time assessment of mental workload using psychophysiological measures and artificial neural networks. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 45(4):635–644, 2003.

[19] Glenn F Wilson and Christopher A Russell. Performance enhancement in an uninhabited air vehicle task using psychophysiologically determined adaptive aiding. *Human factors: the journal of the human factors and ergonomics society*, 49(6):1005–1018, 2007.

[20] Jeffrey R Peters, Vaibhav Srivastava, Grant S Taylor, Amit Surana, Miguel P Eckstein, and Francesco Bullo. Human supervisory control of robotic teams: integrating cognitive modeling with engineering design. *IEEE Control Systems*, 35(6):57–80, 2015.

[21] Richard W Pew. The speed-accuracy operating characteristic. *Acta Psychologica*, 30:16–26, 1969.

[22] Vaibhav Srivastava and F. Bullo. Hybrid combinatorial optimization: Sample problems and algorithms. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 7212–7217, Dec 2011.

[23] Richard P Heitz. The speed-accuracy tradeoff: history, physiology, methodology, and behavior. *Frontiers in neuroscience*, 8:150, 2014.

[24] Raymond Y Cho, Leigh E Nystrom, Eric T Brown, Andrew D Jones, Todd S Braver, Philip J Holmes, and Jonathan D Cohen. Mechanisms underlying dependencies of performance on stimulus history in a two-alternative forced-choice task. *Cognitive, Affective, & Behavioral Neuroscience*, 2(4):283–299, 2002.

[25] Angela J Yu and Jonathan D. Cohen. Sequential effects: Superstition or rational behavior? In *Advances in Neural Information Processing Systems 21*, pages 1873–1880. Curran Associates, Inc., 2009.

[26] Shunan Zhang, Crane He Huang, and J Yu Angela. Sequential effects: a bayesian analysis of prior bias on reaction time and behavioral choice. In *Proceedings of the 36th Annual Conference of the Cognitive Science Society*, 2014.

[27] David B Kaber and Mica R Endsley. Team situation awareness for process control safety and performance. *Process Safety Progress*, 17(1):43–48, 1998.

[28] David B Kaber and Mica R Endsley. The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theoretical Issues in Ergonomics Science*, 5(2):113–153, 2004.

[29] David B Kaber, Emrah Onal, and Mica R Endsley. Design of automation for telerobots and the effect on performance, operator situation awareness, and subjective workload. *Human Factors and Ergonomics in Manufacturing*, 10(4):409–430, 2000.

[30] David D Woods and Nadine B Sarter. Capturing the dynamics of attention control from individual to distributed systems: the shape of models to come. *Theoretical Issues in Ergonomics Science*, 11(1-2):7–28, 2010.

[31] Mica R Endsley. Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):32–64, 1995.

[32] Stephen Monsell. Task switching. *Trends in cognitive sciences*, 7(3):134–140, 2003.

[33] David C Rubin, Sean Hinton, and Amy Wenzel. The precise time course of retention. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 25(5):1161, 1999.

[34] John R Anderson and Lael J Schooler. Reflections of the environment in memory. *Psychological science*, 2(6):396–408, 1991.

[35] Christopher D Wickens. Multiple resources and performance prediction. *Theoretical issues in ergonomics science*, 3(2):159–177, 2002.

[36] Shameem Hameed, Thomas Ferris, Swapnaa Jayaraman, and Nadine Sarter. Using informative peripheral visual and tactile cues to support task and interruption management. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 51(2):126–135, 2009.

[37] Angela J Yu. Decision-making tasks. *Encyclopedia of computational neuroscience*, pages 931–937, 2015.

[38] Peter A Hancock. A dynamic model of stress and sustained attention. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 31(5):519–537, 1989.

[39] Azad M Madni. Integrating humans with and within complex systems. *CrossTalk*, 5, 2011.

[40] Rolf Klomp, Clark Borst, René van Paassen, and Max Mulder. Expertise level, control strategies, and robustness in future air traffic control decision aiding. *IEEE Transactions on Human-Machine Systems*, 46(2):255–266, 2016.

[41] Xiaotian Wang, Zhenwu Shi, Fumin Zhang, and Yue Wang. Mutual trust based scheduling for (semi)autonomous multi-agent systems. In *American Control Conference (ACC), 2015*, pages 459–464, July 2015.

[42] Mary L Cummings and Paul J Mitchell. Predicting controller capacity in supervisory control of multiple uavs. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 38(2):451–460, 2008.

[43] Yves Boussemart and Mary L Cummings. Predictive models of human supervisory control behavioral patterns using hidden semi-markov models. *Engineering Applications of Artificial Intelligence*, 24(7):1252–1262, 2011.

[44] Jamie C Macbeth, Mary L Cummings, Luca F Bertuccelli, and Amit Surana. Interface design for unmanned vehicle supervision through hybrid cognitive task analysis. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 56, pages 2344–2348. SAGE Publications, 2012.

[45] K. Savla and E. Frazzoli. Dynamical queue-based task management policies for human operators. In *American Control Conference (ACC), 2011*, pages 1849–1854, June 2011.

[46] Luca F Bertuccelli and Mary L Cummings. Operator choice modeling for collaborative uav visual search tasks. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 42(5):1088–1099, 2012.

[47] Brahim Rekiek, Alexandre Dolgui, Alain Delchambre, and Antoneta Bratcu. State of art of optimization methods for assembly line design. *Annual Reviews in control*, 26(2):163–174, 2002.

[48] Christopher J Shannon, Luke B Johnson, Kimberly F Jackson, and Jonathan P How. Adaptive mission planning for coupled human-robot teams. In *American Control Conference (ACC), 2016*, pages 6164–6169. American Automatic Control Council (AACC), 2016.

[49] Holly AH Handley and Robert J Smillie. Human view dynamics the nato approach. *Systems Engineering*, 13(1):72–79, 2010.

[50] Raja Parasuraman and Ericka Rovira. Workload modeling and workload management: Recent theoretical developments. Technical report, DTIC Document, 2005.

[51] Josephine Q Wojciechowski. Validation of improved research integration tool (imprint) driving model for workload analysis. Technical report, DTIC Document, 2004.

[52] Carl E Nehme. Modeling human supervisory control in heterogeneous unmanned vehicle systems. Technical report, DTIC Document, 2009.

[53] ML Cummings and PJ Mitchell. Operator scheduling strategies in supervisory control of multiple uavs. *Aerospace Science and Technology*, 11(4):339–348, 2007.

[54] Fei Gao, Missy L Cummings, and Luca F Bertuccelli. Teamwork in controlling multiple robots. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pages 81–88. ACM, 2012.

[55] L.F. Bertuccelli and M.L. Cummings. Scenario-based robust scheduling for collaborative human-uav visual search tasks. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 5702–5707, Dec 2011.

[56] D.A. Spencer and Yue Wang. Slqr suboptimal human-robot collaborative guidance and navigation for autonomous underwater vehicles. In *American Control Conference (ACC), 2015*, pages 2131–2136, July 2015.

[57] Andres Eduardo Ortiz Rubiano. *On-line task scheduling and trajectory planning techniques for reconnaissance missions with multiple unmanned aerial vehicles supervised by a single human operator.* PhD thesis, University of Illinois at Urbana-Champaign, 2012.

[58] Andres E Ortiz and Cédric Langbort. Scheduling multiple uninhabited aerial vehicles for target classification by single human operator. *Journal of Aerospace Computing, Information, and Communication*, 8(12):328–345, 2011.

[59] K. Savla, T. Temple, and E. Frazzoli. Human-in-the-loop vehicle routing policies for dynamic environments. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 1145–1150, Dec 2008.

[60] Xu Chu Ding, M. Powers, M. Egerstedt, Shih yih Young, and T. Balch. Executive decision support. *Robotics Automation Magazine, IEEE*, 16(2):73–81, June 2009.

[61] Paul Benjamin Reverdy et al. Human-inspired algorithms for search a framework for human-machine multi-armed bandit problems. *Princeton University: PhD Dissertation*, 2014.

[62] Jerome Le Ny and H. Balakrishnan. Distributed feedback control for an eulerian model of the national airspace system. In *American Control Conference, 2009. ACC '09.*, pages 2891–2897, June 2009.

[63] T. Tsiligkaridis, B.M. Sadler, and A.O. Hero. Collaborative 20 questions for target localization. *IEEE Trans. Inform. Theory*, 60(4):2233–2252, April 2014.

[64] Bruno Jedynak, Peter I Frazier, and Raphael Sznitman. Twenty questions with noise: Bayes optimal policies for entropy loss. *Journal of Applied Probability*, 49(1):114–136, 2012.

[65] T. Weissman. Capacity of channels with action-dependent states. *IEEE Trans. Inform. Theory*, 56(11):5396–5411, Nov 2010.

[66] Sennur Ulukus, Aylin Yener, Elza Erkip, Osvaldo Simeone, Michele Zorzi, Pulkit Grover, and Kaibin Huang. Energy harvesting wireless communications: A review of recent advances. *IEEE Journal on Selected Areas in Communications*, 33(3):360–381, 2015.

[67] Ashutosh Nayyar, Tamer Basar, Demosthenis Teneketzis, and Venugopal V. Veeravalli. Optimal strategies for communication and remote estimation with an energy harvesting sensor. *CoRR*, abs/1205.6018, 2012.

[68] O. Ozel, K. Tutuncuoglu, Jing Yang, Sennur Ulukus, and A Yener. Transmission with energy harvesting nodes in fading wireless channels: Optimal policies. *IEEE J. Select. Areas Commun.*, 29(8):1732–1743, September 2011.

[69] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory.* Wiley-Interscience, 2006.

[70] Andres Ortiz, Derek Kingston, and Cédric Langbort. Multi-uav velocity and trajectory scheduling strategies for target classification by a single human operator. *J. Intell. Robotics Syst.*, 70(1-4):255–274, April 2013.

[71] S.M. Lynch and B.W. Bequette. Model predictive control of blood glucose in type i diabetics using subcutaneous glucose measurements. In *2002 American Control Conference*, volume 5, pages 4039–4043 vol.5, 2002.

[72] David R. Karger, Sewoong Oh, and Devavrat Shah. Budget-optimal task allocation for reliable crowdsourcing systems. *CoRR*, abs/1110.3564, 2011.

[73] D.R. Cox and H.D. Miller. *The Theory of Stochastic Processes*. Science paperbacks. Taylor & Francis, 1977.

[74] G.M. Lipsa and N.C. Martins. Optimal state estimation in the presence of communication costs and packet drops. In *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, pages 160–169, Sept 2009.

[75] G.M. Lipsa and N.C. Martins. Remote state estimation with communication costs for first-order lti systems. *Automatic Control, IEEE Transactions on*, 56(9):2013–2025, Sept 2011.

[76] Michael Athans. On the determination of optimal costly measurement strategies for linear stochastic systems. *Automatica*, 8(4):397 – 412, 1972.

[77] M. Shamaiah, S. Banerjee, and H. Vikalo. Greedy sensor selection: Leveraging submodularity. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 2572–2577, Dec 2010.

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

[79] B. Hajek, K. Mitzel, and Sichao Yang. Paging and registration in cellular networks: Jointly optimal policies and an iterative algorithm. *Information Theory, IEEE Transactions on*, 54(2):608–622, Feb 2008.

[80] M.M. Vasconcelos and N.C. Martins. Estimation over the collision channel: Structural results. In *Communication, Control, and Computing (Allerton), 2013 51st Annual Allerton Conference on*, pages 1114–1119, Oct 2013.

[81] O. Ozel, Jing Yang, and Sennur Ulukus. Optimal scheduling over fading broadcast channels with an energy harvesting transmitter. In *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2011 4th IEEE International Workshop on*, pages 193–196, Dec 2011.

[82] Axel Schulte and Diana Donath. Measuring self-adaptive uav operators' load-shedding strategies under high workload. In *Proceedings of the 9th International Conference on Engineering Psychology and Cognitive Ergonomics*, EPCE'11, Berlin, Heidelberg, 2011. Springer-Verlag.

[83] Oswaldo Luiz Valle Costa, Marcelo Dutra Fragoso, and Ricardo Paulino Marques. *Discrete-time Markov jump linear systems*. Springer Science & Business Media, 2006.

[84] AP Teixeira, R Oliveira, PM Alves, and MJT Carrondo. Advances in on-line monitoring and control of mammalian cell cultures: supporting the pat initiative. *Biotechnology advances*, 27(6):726–732, 2009.

[85] Xiangheng Liu and Andrea Goldsmith. Kalman filtering with partial observation losses. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 4, pages 4180–4186. IEEE, 2004.

[86] Vijay Gupta, Timothy H Chung, Babak Hassibi, and Richard M Murray. On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42(2):251–260, 2006.

[87] Vijay Gupta. *Distributed estimation and control in networked systems*. PhD thesis, California Institute of Technology, 2006.

[88] Adolfo Anta and Paulo Tabuada. To sample or not to sample: Self-triggered control for nonlinear systems. *IEEE Transactions on Automatic Control*, 55(9):2030–2042, 2010.

[89] Duo Han, Yilin Mo, Junfeng Wu, Sean Weerakkody, Bruno Sinopoli, and Ling Shi. Stochastic event-triggered sensor schedule for remote state estimation. *IEEE Transactions on Automatic Control*, 60(10):2661–2675, 2015.

[90] Sebastian Trimpe and Raffaello D'Andrea. Event-based state estimation with variance-based triggering. *IEEE Transactions on Automatic Control*, 59(12):3266–3281, 2014.

[91] Lei Zou, Zidong Wang, Huijun Gao, and Xiaohui Liu. Event-triggered state estimation for complex networks with mixed time delays via sampled data information: The continuous-time case. *IEEE Transactions on cybernetics*, 45(12):2804–2815, 2015.

[92] Sekhar Tatikonda and Sanjoy Mitter. Control over noisy channels. *IEEE transactions on Automatic Control*, 49(7):1196–1201, 2004.

[93] Vijay Gupta, Babak Hassibi, and Richard M Murray. Optimal lqg control across packet-dropping links. *Systems & Control Letters*, 56(6):439–446, 2007.

[94] Vijay Gupta, Nuno C Martins, and John S Baras. Optimal output feedback control using two remote sensors over erasure channels. *IEEE Transactions on Automatic Control*, 54(7):1463–1476, 2009.

[95] Jhelum Chakravorty and Aditya Mahajan. Remote-state estimation with packet drop. *IFAC-PapersOnLine*, 49(22):7–12, 2016.

[96] Adam Molin and Sandra Hirche. On lqg joint optimal scheduling and control under communication constraints. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pages 5832–5838. IEEE, 2009.

[97] Eduardo I Silva and Sebastián A Pulgar. Control of lti plants over erasure channels. *Automatica*, 47(8):1729–1736, 2011.

[98] Michel Mariton. *Jump linear systems in automatic control*. M. Dekker New York, 1990.

[99] Panos M Pardalos. Global optimization algorithms for linearly constrained indefinite quadratic problems. *Computers & Mathematics with Applications*, 21(6-7):87–97, 1991.

[100] Izabella Czochralska. The method of bilinear programming for nonconvex quadratic programming. *Applicationes Mathematicae*, 3(17):515–525, 1982.

[101] Giorgio Gallo and Aydin Ülkücü. Bilinear programming: an exact algorithm. *Mathematical Programming*, 12(1):173–194, 1977.

[102] Hanif D Sherali and Amine Alameddine. A new reformulation-linearization technique for bilinear programming problems. *Journal of Global optimization*, 2(4):379–410, 1992.

[103] Jeff Linderoth. A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs. *Mathematical Programming*, 103(2):251–282, 2005.

[104] Artyom G Nahapetyan. Bilinear programming., 2009.

[105] Scott Cohen and Carlo Tomasi. *Systems of bilinear equations*. Stanford University, Department of Computer Science, 1997.

[106] Kurt M Anstreicher. Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. *Journal of Global Optimization*, 43(2):471–484, 2009.

[107] Antonio J Conejo, Enrique Castillo, Roberto Minguez, and Raquel Garcia-Bertrand. *Decomposition techniques in mathematical programming: engineering and science applications*. Springer Science & Business Media, 2006.

[108] Eugene L Lawler and David E Wood. Branch-and-bound methods: A survey. *Operations research*, 14(4):699–719, 1966.

[109] Jacqueline M Bloemhof-Ruwaard and Eligius MT Hendrix. Generalized bilinear programming: An application in farm management. *European Journal of Operational Research*, 90(1):102–114, 1996.

[110] Faiz A Al-Khayyal and James E Falk. Jointly constrained biconvex programming. *Mathematics of Operations Research*, 8(2):273–286, 1983.

[111] FA Al-Khayyal. Jointly constrained bilinear programs and related problems: An overview. *Computers & Mathematics with Applications*, 19(11):53–62, 1990.

[112] Faiz A Al-Khayyal. Generalized bilinear programming: Part i. models, applications and linear programming relaxation. *European Journal of Operational Research*, 60(3):306–314, 1992.

[113] Anders Rantzer. Scalable control of positive systems. *European Journal of Control*, 24:72–80, 2015.

[114] Lorenzo Farina and Sergio Rinaldi. *Positive linear systems: theory and applications*, volume 50. John Wiley & Sons, 2011.

[115] Lieven Vandenberghe and Stephen Boyd. Semidefinite programming. *SIAM review*, 38(1):49–95, 1996.

[116] Sunyoung Kim and Masakazu Kojima. Exact solutions of some nonconvex quadratic optimization problems via sdp and socp relaxations. *Computational Optimization and Applications*, 26(2):143–154, 2003.

[117] Jacques F Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik*, 4(1):238–252, 1962.

[118] Arthur M Geoffrion. Generalized benders decomposition. *Journal of optimization theory and applications*, 10(4):237–260, 1972.

[119] Christodoulos A Floudas. *Nonlinear and mixed-integer optimization: fundamentals and applications*. Oxford University Press, 1995.

[120] L Kaufman and Fernand Broeckx. An algorithm for the quadratic assignment problem using bender's decomposition. *European Journal of Operational Research*, 2(3):207–211, 1978.

[121] Jean-François Cordeau, Goran Stojković, François Soumis, and Jacques Desrosiers. Benders decomposition for simultaneous aircraft routing and crew scheduling. *Transportation science*, 35(4):375–388, 2001.

[122] NV Sahinidis and Ignacio E Grossmann. Convergence properties of generalized benders decomposition. *Computers & Chemical Engineering*, 15(7):481–491, 1991.

[123] A Grothey, S Leyffer, and KIM McKinnon. A note on feasibility in benders decomposition. *Numerical Analysis Report NA/188, Dundee University*, 1999.

[124] Yuzhe Li, Fan Zhang, Daniel E Quevedo, Vincent Lau, Subhrakanti Dey, and Ling Shi. Power control of an energy harvesting sensor for remote state estimation. *IEEE Transactions on Automatic Control*, 62(1):277–290, 2017.

[125] J. Huang, D. Shi, and T. Chen. Event-triggered state estimation with an energy harvesting sensor. *IEEE Transactions on Automatic Control*, PP(99):1–1, 2017.

[126] Georgios KD Saharidis and Marianthi G Ierapetritou. Improving benders decomposition using maximum feasible subsystem (mfs) cut generation strategy. *Computers & chemical engineering*, 34(8):1237–1245, 2010.

[127] Abraham Berman and Robert J Plemmons. Nonnegative matrices. *The Mathematical Sciences, Classics in Applied Mathematics*, 9, 1979.

[128] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[129] Lieven Vandenberghe. Alternatives lecture notes in ee236a: Linear programming, 2013.