# Efficient Energy Management in Wireless Sensor Networks

DISSERTATION

Presented in Partial Fulfillment of the Requirements for

the Degree Doctor of Philosophy in the

Graduate School of The Ohio State University

By

Rahul Srivastava, B. Tech., M. S.

Graduate Program in Electrical and Computer Engineering

The Ohio State University

2010

Dissertation Committee:

Prof. Can Emre Koksal, Adviser

Prof. Ness B. Shroff

Prof. Eylem Ekici

# ABSTRACT

Recent advances in wireless networking and data acquisition have enabled us with a unique capability to remotely sense our environment. Data acquisition networks can be used to sense natural as well as human-created phenomena. As these applications may require deployment in remote and hard-to-reach areas, it is critical to ensure that such wireless sensor networks are capable of operating unattended for long durations. The lack of easy access to a continuous power source in most scenarios and the limited lifetime of batteries have hindered the deployment of such networks. Consequently, the central objective in wireless sensor network design is to utilize the available energy as efficiently as possible. In this thesis, we study the design of optimal or near-optimal energy management schemes for various wireless sensor networks composed of nodes with different capabilities.

Firstly, we derive theoretical upper bounds on the performance of a transmission scheduler for sensor networks. We do this by calculating the information theoretic channel capacity of finite-state Markov channels with imperfect feedback containing different grades of channel state information including that, obtained through Automatic Repeat Request (ARQ) feedback. Secondly, we consider the problem of energy optimal transmission scheduling over a finite state Markov channel with imperfect feedback. We propose a transmission controller that utilizes different "grades" of channel state information to schedule packet transmissions in an energy-optimal

way, while meeting a deadline constraint for all packets waiting in the transmission queue. Our scheduler is readily implementable and it is based on the dynamic programming solution to the finite-horizon transmission control problem. We illustrate that our scheduler achieves a given throughput at a power level that is fairly close to the information-theoretic limit. Finally, we consider the problem of energy management in nodes with energy replenishment capabilities. Here, we derive the performance limits of sensor nodes with limited energy, being replenished at a variable and random rate. We provide a simple localized energy management scheme for nodes with limited energy storage space, and show that our scheme achieves a performance asymptotically close to that available with an unlimited energy source. Based on the insights developed, we address the problem of energy management for energy-replenishing nodes with finite data buffer capacities as well as limited energy storage space. To this end, we give an energy management scheme that is provably asymptotically optimal.

***to Mamma and Papa***

# ACKNOWLEDGMENTS

I would like to thank many people who have made this thesis work a learning and enjoyable experience. First and foremost, I would like to express my gratitude towards my thesis advisor, Prof. Can Emre Koksal, for all his advice, support, guidance and encouragement during my research. Emre has been a constant source of motivation and extremely approachable throughout my graduate studies. I have benefited a lot from his critical feedback on my research and writing. I would like to thank Prof. Ness Shroff and Prof. Eylem Ekici for serving on my Candidacy and Thesis committee. Their feedback, both on my thesis and on other areas were very fruitful to me. I am also very thankful to Professor Rama Yedavalli for serving as the graduate faculty representative for my oral committee.

I would like to thank the faculty here at The Ohio State University, and in particular, the faculty of the Department of Electrical and Computer Engineering for the academic training I received here. I would like to thank Jeri McMichael for her help and counsel throughout my stay here.

I would like to thank the members of the IPS Lab, past and present, for their support and providing a thoughtful, pleasant and conducive atmosphere for research. In particular, I would like to thank Aditi Kothiyal, Arun Kannu, Petru Cociorva, Hernan Romero, Arul Murugan, Praveen Gopala, Sibasish Das, Lifeng Lai, Amrita Ghosh, Subhojit Som, Naveen Ramakrishnan, Sugumar Murugesan, Ahmed Fasih,

Jason Palmer, Justin Ziniel, Ozan Koyluoglu, Onur Gungor, Srikanth Hariharan, Yang Yang, and Mohammad Shahmohammadi. I am especially thankful to Arun Sridharan and Rohit Aggarwal for their friendship as well as our discussions regarding research and almost everything else.

I am grateful to Rohini and Ravi Srivastava for being around and providing a home away from home. I would like to thank my wife Sindhu for her warmth, understanding and support through all these years. Finally, I would like to thank my parents, Romla and Ashok Srivastava, for their unconditional love, support and sacrifices.

# VITA

| | |
|---|---|
| November 13, 1979 | Born - New Delhi, India |
| 2002 | Bachelor of Technology in Electrical Engineering,<br>Indian Institute of Technology, Madras, India |
| 2005 | Master of Science in Electrical and Computer Engineering. ,<br>Rice University, Houston, Texas, USA |
| 2005-2010 | Graduate Student,<br>Graduate Teaching Associate,<br>Graduate Research Associate,<br>The Ohio State University, Columbus, Ohio, USA |

# PUBLICATIONS

### Research Publications

Rahul Srivastava and Can Emre Koksal, "Basic Tradeoffs for Energy Management in Rechargeable Sensor Networks", *Submitted to IEEE/ACM Transactions on Networking.*

Rahul Srivastava and Can Emre Koksal, "Energy Optimal Transmission Scheduling in Wireless Sensor Networks", *IEEE Transactions on Wireless Communications*, Vol. 9, No. 5, pp. 1550 - 1560, May 2010.

Sha Liu, Rahul Srivastava, Can Emre Koksal and Prasun Sinha, "Pushback: A Hidden Markov Model Based Scheme for Energy Efficient Data Transmission in Sensor Networks", *Ad Hoc Networks*, Vol. 7, No. 5, pp. 973-986, July 2009.

Sha Liu, Rahul Srivastava, Can Emre Koksal and Prasun Sinha, "Achieving Energy Efficiency with Transmission Pushbacks in Sensor Networks", *Proceedings of IEEE IWQoS*, June 2008.

## FIELDS OF STUDY

Major Field: Electrical and Computer Engineering

# TABLE OF CONTENTS

**Page**

Appendices:

# LIST OF TABLES

# LIST OF FIGURES

# CHAPTER 1

# INTRODUCTION

Recent advances in wireless networking and data acquisition have enabled us with a unique capability to remotely sense our environment using sensor networks. A sensor network is defined as a data acquisition network that consists of a number of spatially-distributed sensor nodes that monitor natural or human-created phenomena. Sensor networks can be employed for various tasks including:

- Environmental study and monitoring [1,2,3] e.g., monitoring earthquakes, wildfires, glacial movements, pollution, etc.
- Health care [4] e.g., patient monitoring, drug administration, etc.
- Home automation [5, 6, 7] e.g., habitat monitoring and smart environments.
- Security [8, 9] e.g., battlefield surveillance, intrusion detection, infrastructure security, urban monitoring, etc.
- Other applications [10] e.g., inventory control, vehicle tracking, traffic monitoring, etc.

The emergence of sensor networks can be attributed to two factors, (a) development of sensor nodes that can sense natural/man-made phenomena, e.g., temperature,

humidity, pressure, noise, and vehicular movement, and (b) development of network architectures and algorithms that allow for the operation of such networks [11]. While progress has been made in the development of sensor nodes, network architectures and algorithms, the scarcity of energy limits the lifetime of sensor networks and limits their performance in many crucial applications. In this work, we will look at developing algorithms that enable long-term sensor network operation via efficient use of energy resources. Note that sensor networks usually consist of a large number of sensor nodes. For the network to make economic sense, sensor nodes should have a low production cost which, in most cases, translates to simplicity in design and limited capabilities. Due to the low-cost design, sensor nodes have limited communication, computational and data storage capabilities. Therefore, we focus our attention on energy management schemes that do not require sophisticated modulation schemes and can be implemented with minimal computational power.

The objective of this introductory chapter is to provide motivation for the underlying theme of this thesis - efficient energy management in wireless sensor networks. We provide a high-level overview of the different problems addressed in this area and summarize our contributions. Detailed literature reviews and problem statements are presented in individual chapters.

## 1.1 Energy Management Problem and Our Contributions

The energy management problem faced by a single node is illustrated in Fig. 1.1. Here, we have a sensor node with an energy source represented by the battery that can store a finite amount of energy. The battery has access to some energy replenishment mechanism that replaces the energy at a variable rate. The node has a finite data

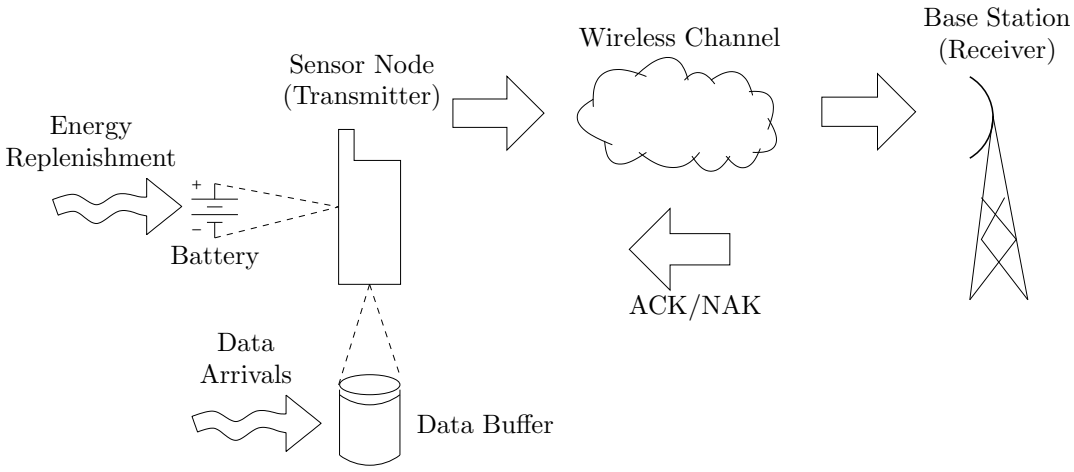



Fig. 1.1: Illustration of the energy management problem in sensor networks.

buffer that stores the data arriving at the sensor node for transmission. Stored data has to be transmitted subject to a deadline or a throughput constraint. The transmitter sends data packets over an "unreliable" wireless channel. An unreliable channel implies that there is a non-zero probability of transmission failure. The receiver acknowledges a successful packet transmission by sending an ACK (acknowledgment) to the transmitter and a failed packet transmission is indicated by a NAK (negative acknowledgment).

Energy consumption in sensor nodes can be classified under four broad categories: idle listening, communication, sensing, and data processing. We will limit the scope of our thesis to exploring problems related to developing energy-efficient communication algorithms. Within the problems of our interest, there are a multitude of scenarios that a network designer might choose to solve. Here, we consider two specific problem setups.

In the **first part** of this thesis, we design an energy-efficient transmission scheduler that uses the least amount of energy to reliably transmit the data contained in the buffer by a specified deadline. Here, we assume that the battery does not have access to energy replenishment. To understand the crucial role of a transmission scheduler, consider one that does not adapt to channel variability. This scheduler will be unable to anticipate bad channel conditions and, consequently, waste a high amount of energy in multiple retransmissions. An intuitive way to improve the performance of such a scheduler would be to use channel information to intelligently schedule packet transmissions. However, due to hardware constraints on the sensor node design, it may not be possible to perfectly sense the channel before making the decision to transmit. In fact, the only information available to the sensor nodes in practical deployments might be in the form of previous error realization sequence, i.e., ACK/NAK sequence of the previous transmissions.

**Our contributions:** **(1)** In Chapter 2, we use tools from *information theory* to derive the throughput limits of an energy-efficient transmission scheduler over an FSMC with ARQ feedback. We use the notion of capacity-cost function for communication channels with feedback to find these theoretical bounds. Each state of the FSMC corresponds to an erasure channel with different erasure probability. Erasure channels are particularly suited to model packetized transmissions as a packet loss is detectable, and it can be treated as an erasure. With these capacity results, we evaluate the potential energy savings achieved by a transmission scheduler that adapts to channel variability using the available channel state information (CSI).

A special case of this setup is imperfect CSI, such as that obtained from the ACK/NAK sequence of the previous transmissions. As mentioned previously, this

type of feedback is freely available at the link layer of most network architectures. For a FSMC with different erasure probabilities, ACK or NAK is a randomized feedback, which gives coarse information about the channel state. One of the main contributions of this thesis is the derivation of the capacity-cost function for this channel. Using the capacity-cost function, we illustrate the value of using ACK/NAK relative to other kinds of feedback including no feedback.

**(2)** With these insights, in Chapter 3 we design an energy-efficient transmission scheduler that uses available channel information to reliably communicate data subject to deadline constraints. Our scheduler is based on the *dynamic programming* (DP) solution to a finite-horizon transmission control problem with perfect and imperfect CSI. We evaluate the performance of our scheduler numerically and illustrate that it achieves a given throughput at a power level that is fairly close to the information theoretic limits evaluated in Chapter 2. Since sensor nodes have computational power constraints, we approximate the proposed transmission scheduler with a lookup table based scheduler that requires minimal computations. We show that the performance loss associated with the approximation is small even with reasonably low table sizes.

**(3)** In Chapter 4, we give a MAC-layer implementation of a practical energy management scheme. Using the data collected from an actual sensor network testbed, we study the temporal characteristics of the channel variations and interference. We use this data to estimate the parameters of a *hidden Markov-model* based channel. A novel concept called "pushbacks" is introduced at the MAC layer. Pushback delays packet transmissions appropriately to overcome periods of poor channel quality and high interference. This mechanism increases the packet transmission success rate

while ensuring that the throughput requirement of the node is met. Using extensive numerical and experimental evaluations, we show that significant gains in energy and/or throughput can be observed using the pushback technique.

In the **final part** of this thesis, we consider the case where the sensor node has a finite-sized battery that is being replenished at a variable rate. In addition, the data buffer has a constant stream of incoming data. An efficient energy management scheme should control the transmission power in such a way that the probabilities of battery discharge and data queue overflow are minimized while achieving an optimal or near-optimal performance. Typically, for nodes with energy replenishment, aggressive use of energy can lead to frequent energy outages causing communication failures (for time-critical data) and lack of coverage in sensor networks, whereas energy conservative schemes may lead to large number of data queue overflows as well as missed recharging opportunities. In addition to this challenge of trying to find the right balance, we note that there is a strong coupling between the battery and the data queue processes that complicates the performance evaluation of energy management schemes.

**Our Contributions:** In Chapter 5, we identify the performance limits of sensor nodes with energy replenishment and provide guidelines to approach these limits. **(1)** First, we assume a finite battery size and an infinitely backlogged data queue. The success of the node in achieving communication and/or sensing tasks is measured in terms of a concave and non-decreasing utility function of the consumed energy. Intuitively, increasing the battery size should lead to a decay in the battery discharge probability and convergence to the maximum utility function. However, we

identify a basic trade-off by showing that it is not possible to simultaneously observe an exponential decay (with the battery size) for the discharge probability and achieve maximum utility. With this limitation in mind, we develop a "simple" energy management scheme that achieves asymptotic convergence to the maximum average utility achieved by a scheme that has access to an infinite energy buffer at the same power level. In addition, the probability of energy outage decays quadratically for this scheme. Our simple energy management scheme uses local information available to the node and does not require intensive computations.

**(2)** Based on the insights developed, we also address the problem of energy management for energy replenishing nodes with finite battery and finite data buffer capacities. Since the drifts of the battery and data queue are dependent, evaluation of battery discharge and buffer overflow probabilities simultaneously for any energy management scheme is highly complex. To that end, we use the *diffusion limit* to approximate these expressions. We develop an localized energy management scheme that achieves convergence to the maximum utility achieved by a scheme that has access to an infinite data and energy buffers at the same power level. In addition, we show that the buffer overflow probability decays quadratically with the buffer size and battery discharge probability decays exponentially with the battery size for our energy management scheme.

The summary of our contributions and tools used to solve different variants of the energy management problem are summarized in Table 1.1.

## 1.2 Relevant Publications

The material of Chapters 2 and 3 is based on:

| **Problem** | **Tools Used** | **Chapter** |
|---|---|---|
| Performance Bounds<br>for Transmission Schedulers | Information Theory<br>Optimization Theory | 2 |
| Design a<br>Transmission Scheduler | Dynamic Programming | 3 |
| Implement an<br>energy-aware protocol | Hidden Markov Models<br>Maximum Likelihood Estimation<br>MAC layer implementation | 4 |
| Energy Management in<br>Energy Replenishing Networks | Large Deviations Theory<br>Diffusion Approximation | 5 |

Table 1.1: Summary of the thesis contributions

Rahul Srivastava, and Can Emre Koksal,

"Energy Optimal Transmission Scheduling in Wireless Sensor Networks,"

IEEE Transactions on Wireless Communications, Vol. 9, No. 5, pp. 1550 - 1560, May 2010.

An extended version of this work is also available as:

Rahul Srivastava and Can Emre Koksal,

"Energy Optimal Transmission Scheduling in Wireless Sensor Networks,"

Technical Report, The Ohio State University, February 2009.

Available: `http://arxiv.org/abs/1003.0054`

The material of Chapter 4 is based on:

Sha Liu, Rahul Srivastava, Can Emre Koksal, and Prasun Sinha

"Pushback: A Hidden Markov Model Based Scheme for Energy Efficient Data Transmission in Sensor Networks,"

Ad Hoc Networks, Vol. 7, No. 5, pp. 973-986, July 2009.

And:

Sha Liu, Rahul Srivastava, Can Emre Koksal, and Prasun Sinha
"Achieving Energy Efficiency with Transmission Pushbacks in Sensor Networks,"
Proceedings of IEEE IWQoS, June 2008

The material of Chapter 5 is based on:

Rahul Srivastava, and Can Emre Koksal,
"Basic Tradeoffs for Energy Management in Rechargeable Sensor Networks,"
*Under Submission* IEEE/ACM Transactions on Networking.

An extended version of this work is also available as:

Rahul Srivastava and Can Emre Koksal,
"Basic Tradeoffs for Energy Management in Rechargeable Sensor Networks,"
Technical Report, The Ohio State University, August 2010.
Available: `http://arxiv.org/abs/1009.0569`

# CHAPTER 2

# THROUGHPUT LIMITS FOR TRANSMISSION SCHEDULERS IN FINITE STATE MARKOV CHANNELS

## 2.1 Introduction

The objective of this chapter is to find throughput bounds for energy-efficient transmission schedulers. As we have discussed in Chapter 1, design of transmission schedulers is a challenging research problem due to the limited computational and energy resources available to the sensor node. Also, nodes do not have access to accurate CSI that would guide the scheduler to attempt packet transmissions at times with best chance of success. For instance, nodes often have access to only the randomized CSI in the form of ACK/NAK information about previous transmissions. Moreover, sensor nodes typically use an ON-OFF type transmission scheduler, which can either transmit a packet at a fixed power level or defer the packet transmission. It is under these general settings that we propose to find the fundamental limits on the performance of transmission schedulers in sensor networks. Here we define the performance of a scheduler as the maximum achievable throughput, subject to an average power constraint.

A transmission scheduler provides a strategy to consume the minimum energy, $\mathcal{P}T$, to attempt successful transmission of $B$ packets by a specified deadline $T$. This corresponds to a throughput of $B/T$ at power $\mathcal{P}$. In this chapter, we relax the deadline constraint and find the maximum achievable *asymptotic* throughput subject to a power constraint $\mathcal{P}$. This quantity is known as the capacity-cost function $C(\mathcal{P})$ [12]. The inverse of the capacity-cost function gives the minimum power $\mathcal{P}$ for a given asymptotic throughput $C$ [13], linking it to the objective of the DP problem. Inverting the axes of $\mathcal{P}$-$C(\mathcal{P})$ plot will give the plot of $\mathcal{P}$ as a function of $C$.

The capacity-cost function will give an upper bound on the achievable throughput of any transmission scheduler for the corresponding wireless channel. To understand this statement, we note that the information theoretic calculations are asymptotic results, i.e., number of channel uses tend to infinity while meeting a throughput requirement. Any scheduler that transmits $B$ packets in $T$ time slots will achieve an asymptotic throughput of $B/T$ packets/time slot by replicating itself every $T$ time slots. However, the converse is not true, as ensuring an asymptotic throughput does not guarantee a throughput for every block of $T$ time slots. As a result, the capacity-cost function requires less power to achieve the same throughput and provides an upper bound on the throughput of the transmission scheduler. The wireless channel is modeled as a finite state Markov channel (FSMC).

Capacity of FSMCs with CSI is a well studied topic. Goldsmith and Varaiya [14] gave the channel capacity of FSMCs with perfect CSI at the transmitter and receiver. In this case, CSI implies accurate knowledge of the instantaneous state of the Markov channel. The optimal power adaptation for such a channel model implies "water-filling" in time, which is analogous to the water-filling result in frequency in a

frequency-selective channel [15]. Viswanathan [16] extended these results for delayed perfect CSI at the transmitter. Here, CSI is defined as the delayed knowledge of the Markov channel state. Yüksel and Tatikonda derived the capacity of a FSMC with imperfect deterministic feedback [17]. Imperfect deterministic feedback implies that the transmitter has inaccurate (quantized) information about the channel state. The quantizer used is a deterministic function of the channel state. Although these results are useful for a wide variety of capacity calculations, they can not be applied directly to channels with discrete power levels and a long term average power constraint on the input symbols. This is due to the fact that the aforementioned results assume continuous transmission or transmission in very time slot. However, our transmitter might have to defer transmission, due to the presence of discrete power levels, in order to meet the power constraints.

To incorporate skipped transmission attempts in the classical capacity results, we introduce a new discrete channel, which we call *channel with vacation*. We associate a channel with vacation to each state of the FSMC, and calculate the capacity-cost function for this special Markov channel. The general problem of finding the capacity of a feedback channel with memory has been notoriously difficult in information theory; and the problem of feedback capacity containing a CSI, which is a random function of the channel state has not been addressed before. We use this kind of a feedback in our channel model because the probability of a transmission success is channel state dependent, which implies that the feedback is a random function of the channel state. We generalize [17] and combine it with [16] and [18] in order to calculate the capacity-cost function of our channel with feedback. We compare this capacity-cost function with the performance achieved by our scheduler and observe

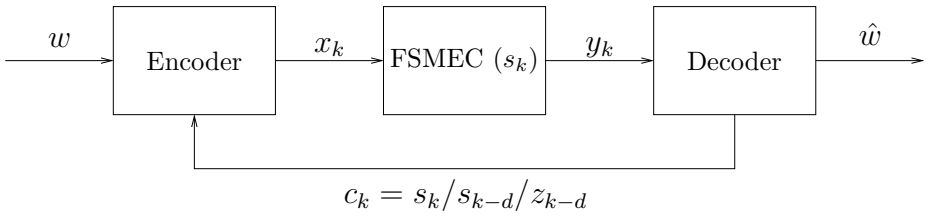



Fig. 2.1: Block diagram showing the system model used for the capacity calculations for the FSMEC with CSI at the receiver and feedback.

that for a given throughput requirement, the power expended by our scheme is close to the theoretical limit of the system.

The rest of the chapter is organized as follows. First, we provide some basic definitions and system model in Section 2.2. We present the capacity-cost results for this system model in Section 2.3. In Section 2.4, we evaluate and compare the capacity-cost function for different types of channel feedback.

## 2.2 Definitions and System Model

The system model used in this work is shown in Figure 2.1. We consider a slotted time model indexed by $k$. Each symbol requires one time slot for transmission. The channel model that we consider for our analysis is a finite-state Markov erasure channel (FSMEC). A more detailed discussion of FSMECs is provided in Section 2.2.1. For now, we note that FSMECs are a special case of the FSMCs. A FSMC is a channel model which, during each symbol duration, is in one of a finite number of states with Markov transition between these states. While not as simple as a Bernoulli or an independent loss channel model, Markov models are used to approximate channels with memory. FSMCs were first used to model a channel with bursty errors [19], and are popular in the literature [20].

The channel input is denoted by $X_k \in \mathcal{X}$ and output is denoted by $Y_k \in \mathcal{Y}$. The messages, chosen from a set of equally probable set of messages $\mathcal{W}$, are denoted by $W$ and the decoded messages are denoted by $\hat{W}$. The size of the message set $\mathcal{W}$ is $M$. The channel state $\{S_k \in \mathcal{S},\ k = 1, 2, \ldots\}$ determines the channel transition (or error) probability during the $k$th symbol duration and the CSI $C_k$ is a function of $S_k$. Here $\mathcal{S}$ is the state space of the Markov channel. We derive capacity results for different "grades" of CSI at the transmitter and the possible grades of CSI are classified as:

1. **Non-causal Perfect CSI:** the scheduler knows the instantaneous channel state, i.e., $C_k = S_k$.

2. **Causal Perfect CSI:** the scheduler knows the channel state delayed by $d$ time units, i.e., $C_k = S_{k-d}$.

3. **Causal Partial CSI:** in this case, the controller knows the ACK/NAK upto and including the $d$th previous transmission, i.e., $C_k = Z_{k-d}$.

Before presenting the specifics of the capacity calculation, we define some information theoretic quantities.

*Encoder:* For a given message set $\mathcal{W}$, an encoder is a sequence of code functions $\{f_k(w, c_0^{k-1})\}_{k\geq 1}$, where the symbol to be sent at time $k$ is given by $f_k(w, c_0^{k-1})$.

*Channel Code:* For a given message set $\mathcal{W}$, the $(T, M, \mathcal{P}, e)$ channel code consists of: **(1)** The block-length of the codewords is equal to $T$, i.e., the number of channel uses to transmit a message is equal to $T$. **(2)** Each codeword $\{f_k(w, c_0^{k-1})\}_{1\leq k\leq T}\ \forall\ w,\ c_0^{T-1}$, satisfies the constraint, $\frac{1}{T}\sum_{i=1}^{T} b(f_i(w, c_0^{i-1})) \leq \mathcal{P}$, where $b(x)$ is the non-negative

cost associated with the input symbol $x \in \mathcal{X}$. Therefore $\mathcal{P}$ is a hard power constraint imposed on every sample outcome of the channel state or ACK/NAK sequence. **(3)** The average probability of incorrectly decoding a message is bounded as $\frac{1}{|\mathcal{W}|}\sum_{w\in\mathcal{W}} p(\hat{w} \neq w|W = w) \leq e$.

*Capacity-Cost Function:* Given $0 \leq e < 1$ and $\mathcal{P} > 0$, a non-negative number $R$ is an $e$-achievable rate with the average cost per symbol not exceeding $\mathcal{P}$ if for every $\delta > 0$ there exists $T_0$ such that if $T \geq T_0$, then an $(T, M, \mathcal{P}, e)$ code can be found whose rate satisfies $\log M > T(R - \delta)$. Furthermore, $R$ is said to be achievable if it is $e$-achievable for all $0 < e < 1$. The maximum achievable rate with average cost per symbol not exceeding $\mathcal{P}$ is the channel capacity denoted by $C(\mathcal{P})$, which is referred to as the capacity-cost function.

We cannot directly apply classical channel capacity results as the system that we consider has special properties. The factors to consider while evaluating the channel capacity are:

1. **Skipped Channel Uses:** An ON-OFF scheduler is limited to two discrete power levels corresponding to the decision to transmit (at a fixed power) or not to transmit. As a result, under a power constraint, we should consider an "outage capacity" in which the decision not to transmit is accounted for in the capacity calculations.

2. **Markov Channel with Delayed Feedback:** Deterministic perfect and quantized feedback in such channels has been treated in literature [16, 17], but randomly quantized feedback has not. We have to calculate the capacity-cost function of a Markov channel with feedback in the form of ACK/NAK. In our

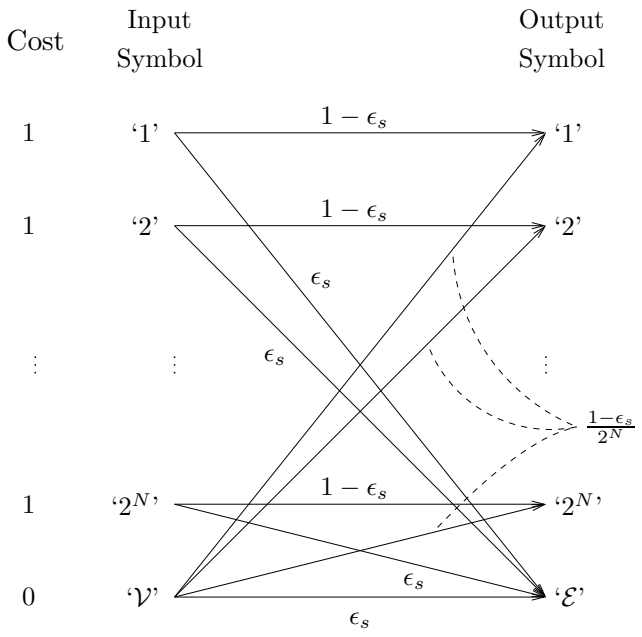



Fig. 2.2: The Erasure Channel with Vacations (ECV). '$\mathcal{V}$' is a zero-cost, zero-information symbol. {'1','2',. . .,'$2^N$'} incur a unit cost on transmission. The transition labels represent the appropriate probabilities.

channel model, ACK/NAKs are a random function of the channel state and hence need to be considered accordingly.

### 2.2.1 Finite State Markov Erasure Channels (FSMECs)

A FSMEC is a special case of FSMC, where each state has an erasure channel (EC) associated with it. The channel state $S_k$ determines the channel erasure probability during the $k$th symbol. Note that we describe an entire packet (or the lack of it) as a channel symbol and model a detectable packet error/loss using an erasure. In this problem, since we assume an ON-OFF transmitter, the input symbols are restricted to discrete power levels and have a long term average power constraint. Consequently, it is not possible to transmit data at every time instant. Discrete ECs [15] do not provide

any provision for calculating mutual information rate under such power constraints. As a result, the channel model has to be designed in such a way that it incorporates the "OFF-period" of the transmitter in the capacity calculations.

To integrate the idea of no transmission in our channel model, we add an input symbol ('$\mathcal{V}$') to the traditional EC as shown in Figure 2.2 and call the new channel erasure channel with vacations (ECV). Each input symbol has an associated cost (power) of 1 unit and the cost required to transmit '$\mathcal{V}$' is 0. The transition probabilities of '$\mathcal{V}$' are chosen according to the statistical requirement rather than the physical signaling system. We will show in Section 2.3 that the capacity-cost function $C(\mathcal{P})$ of the ECV in Figure 2.2 is $\mathcal{P}N(1-\epsilon_s)$. This can be intepreted as the capacity of an EC that is active for $\mathcal{P}$ fraction of time. One can observe that the unconstrained capacity (i.e., $\mathcal{P}=1$) of this ECV is the same as that of an EC with $2^N$ input symbols and erasure probability $\epsilon_s$. As a result, addition of '$\mathcal{V}$' to the EC does not change the mutual information between the input and output symbols. Therefore, we infer that '$\mathcal{V}$' is a zero-cost, zero-information symbol which can replace the OFF-period of the transmitter for the capacity calculation. For the ON-period, we consider transmission of $N$-bit (fixed-size) packets that require unit cost to transmit. Hence, there will be $2^N$ distinct symbols, $\{\text{'1'},\ldots,\text{'}2^N\text{'}\}$, shown in Figure 2.2 representing the ON-period of the transmitter. We assume that the probability of packet error is independent of the packet "content". As a result in state $s$, each packet (symbol) is transmitted correctly with probability $1-\epsilon_s$.

## 2.3 Capacity Results

In this section we state the capacity results for the FSMEC with different CSI at the transmitter. More specifically, we are interested in the capacity cost function $C(\mathcal{P})$, which is defined as,

$$C(\mathcal{P}) = \sup_{\substack{X \\ E[b[X]] \leq \mathcal{P}}} I(X;Y).$$

Where $b[X]$ is the cost of transmitting symbol $X$. We apply this general definition to specific channel scenarios to find the capacity-cost functions.

### 2.3.1 Capacity-Cost Function for Causal Perfect CSI

First, we derive the capacity-cost function for FSMEC channels with causal perfect CSI at the transmitter. Each state $s \in \mathcal{S}$ is associated with an ECV (shown in Figure 2.2) with parameter $\epsilon_s$ and $2^N + 1$ input symbols. We define $\pi(s)$ as the steady state distribution of states.

**Theorem 2.1.** *The capacity-cost function for a FSMEC with d-step transition probability matrix $P^d$, each state $s$ associated with an ECV with parameter $\epsilon_s$ and causal perfect CSI at the encoder is given by,*

$$C_{FSMEC\text{-}CSI}(\mathcal{P}) = \sup_{\mathcal{P}(\tilde{s})} \sum_{\tilde{s}\in\mathcal{S}} \sum_{s\in\mathcal{S}} \pi(\tilde{s}) P^d(\tilde{s}, s)(1-\epsilon_s) N\mathcal{P}(\tilde{s}),$$
$$s.t. \begin{array}{c} \sum_{\tilde{s}} \pi(\tilde{s})\mathcal{P}(\tilde{s}) \leq \mathcal{P} \\ 0 \leq \mathcal{P}(\tilde{s}) \leq 1, \quad \forall \tilde{s} \in \mathcal{S} \end{array}. \tag{2.1}$$

*The optimization variable is $\mathcal{P}(\tilde{s})$, which can be thought of as the power allocation policy for state $\tilde{s}$. It is the fraction of unit cost symbols in the codebook for state $\tilde{s}$.*

*Proof.* The capacity-cost function of a cost constrained memoryless stationary channel is given by,

$$C(\mathcal{P}) = \sup_{\substack{X \\ E[b[X]] \leq \mathcal{P}}} I(X;Y). \tag{2.2}$$

The transmission of one bit of information requires $1/C(\mathcal{P})$ symbols at a cost of $\mathcal{P}/C(\mathcal{P})$. In order to find the capacity-cost function, first we find the Capacity per unit cost, $C_{\text{Cost}}$, using the expression given in [18],

$$\begin{aligned} C_{\text{cost}} &= \sup_{X} \frac{I(X;Y)}{E[b[X]]} \\ &= \sup_{x} \frac{D(P_{Y|X=x}||P_{Y|X=\mathcal{V}})}{b[x]} \\ &= (1-\epsilon_s)N. \end{aligned} \tag{2.3}$$

In our case, $b[x] = 1$ for $x = \{1, \ldots, 2^N\}$ and $b[x] = 0$ for $x = \mathcal{V}$. We find that for an ECV, $D(P_{Y|X=1}||P_{Y|X=\mathcal{V}}) = \ldots = D(P_{Y|X=2^N}||P_{Y|X=\mathcal{V}}) = 1 - \epsilon_s$. Using Eq. (2.3) and incorporating the average power constraint of $\mathcal{P}$ we can find the capacity-cost function of the ECV,

$$C(\mathcal{P}) = E[b[X]]C_{\text{cost}} = \mathcal{P}\left[N(1-\epsilon_s)\right]. \tag{2.4}$$

The FSMEC at each instant, is in some state $s \in \mathcal{S}$. In each of these states, the channel is an ECV with the corresponding transition probabilities. The capacity-cost calculations made earlier are used in conjunction[1] with results from [16] to find the capacity of FSMECs with delayed feedback. For a average symbol power constraint $\mathcal{P}$ and unit feedback delay, the channel capacity $C_{\text{FSMEC}}(\mathcal{P})$ for a finite-state Markov

[1]It might appear that the capacity-cost calculations which have been made for a memoryless channel are not applicable for Markov channels (which are not memoryless). However [16] derives capacity results which use the mutual information rate conditioned on the present state of the channel, which makes the channel memoryless for Markov channels.

channel is given by [16],

$$C_{\text{FSMEC}}(\mathcal{P}) = \max_{P(X|\tilde{S})} I(X;Y|S,\tilde{S}) \quad \text{s.t.} \begin{array}{c} \sum_{\tilde{s}} \pi(\tilde{s})\mathcal{P}(\tilde{s}) \leq \mathcal{P} \\ 0 \leq \mathcal{P}(\tilde{s}) \leq 1, \quad \forall \tilde{s} \in \mathcal{S} \end{array}. \tag{2.5}$$

Where $\tilde{s} \in \mathcal{S}$ represents the state information fed back ($\tilde{s} = s_{t-d}$) to the transmitter from the receiver and $s \in \mathcal{S}$ is state at the time of transmission. $\mathcal{P}(\tilde{s})$ is the average cost per symbol (channel use) incurred (or power allocation strategy) while in state $\tilde{s}$. And we define,

$$\begin{aligned} I(X;Y|S,\tilde{S}) &= \sum_{\tilde{s}\in\mathcal{S}} \pi(\tilde{s}) I(X;Y|S,\tilde{S}=\tilde{s}) \\ &= \sum_{\tilde{s}\in\mathcal{S}} \pi(\tilde{s}) \sum_{s\in\mathcal{S}} \mathrm{P}^d(\tilde{s},s) I(X;Y|S=s,\tilde{S}=\tilde{s}). \end{aligned}$$

We have previously defined $\pi(s)$ as the steady state distribution of state $s$ and $\mathrm{P}^d$ as the $d$-step transition probability matrix of the Markov chain representing the channel. To sum up, we can write the capacity of the energy constrained channel as follows,

$$C_{\text{FSMEC}}(\mathcal{P}) = \sup_{P(X|\tilde{S})} \sum_{\tilde{s}} \pi(\tilde{s}) \sum_{s} \mathrm{P}(\tilde{s},s) I(X;Y|S=s,\tilde{S}=\tilde{s}) \quad \text{s.t.} \begin{array}{c} \sum_{\tilde{s}} \pi(\tilde{s})\mathcal{P}(\tilde{s}) \leq \mathcal{P} \\ 0 \leq \mathcal{P}(\tilde{s}) \leq \mathcal{P} \quad \forall \tilde{s} \end{array}. \tag{2.6}$$

Using the results derived in Eq. (2.4), the optimization problem (2.6) becomes a power allocation problem and can be written as,

$$C_{\text{FSMEC}}(\mathcal{P}) = \sup_{\mathcal{P}(\tilde{s})} \sum_{\tilde{s}} \sum_{s} \pi(\tilde{s}) \mathrm{P}(\tilde{s},s)(1-\epsilon_s) N \mathcal{P}(\tilde{s}), \quad \text{s.t.} \begin{array}{c} \sum_{\tilde{s}} \pi(\tilde{s})\mathcal{P}(\tilde{s}) \leq \mathcal{P} \\ 0 \leq \mathcal{P}(\tilde{s}) \leq 1, \quad \forall \tilde{s} \in \mathcal{S} \end{array}. \tag{2.7}$$

□

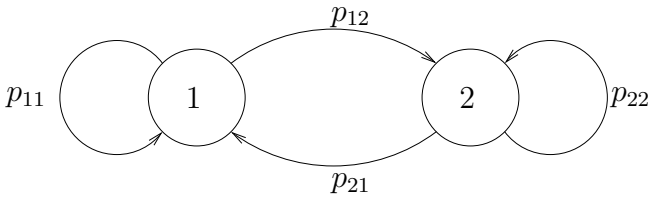



Fig. 2.3: A 2-state Markov erasure channel with two states, "1" and "2". The associated erasure probabilities are $\epsilon_1$ and $\epsilon_2$ and the transitions have been labeled with the respective probabilities.

Theorem 2.1 gives the expression for the capacity of a FSMEC when causal perfect CSI is available at the encoder. The solution to the optimization problem (2.1) gives the optimal power allocation policy $\{\mathcal{P}^*(s), \forall s \in \mathcal{S}\}$. Substituting this optimal power allocation into the original expression gives the channel capacity. Also note that the capacity-cost function for the non-causal perfect CSI case is given by substituting $d = 0$. In this case, $\mathrm{P}^d$ becomes an identity matrix of size $|\mathcal{S}| \times |\mathcal{S}|$.

**Example 2.1 2-state Markov erasure channel (2SMEC) with perfect causal CSI**

We consider a 2SMEC which at each instant, is in one of the two states, i.e. $\mathcal{S} = \{1, 2\}$ as shown in Figure 2.3. For this channel, $\pi_1 = \frac{p_{21}}{p_{12}+p_{21}}$ and $\pi_2 = \frac{p_{12}}{p_{12}+p_{21}}$, and, $\mathrm{P} = \begin{bmatrix} 1-p_{12} & p_{12} \\ p_{21} & 1-p_{21} \end{bmatrix}$.

Applying Theorem 2.1 for $s \in \{1, 2\}$ and $d = 1$, we can solve this problem for $\{\mathcal{P}^*(1), \mathcal{P}^*(2)\}$ as a constrained optimization problem [21]. The problem has to be divided into two cases and the power allocation policy for these cases are:

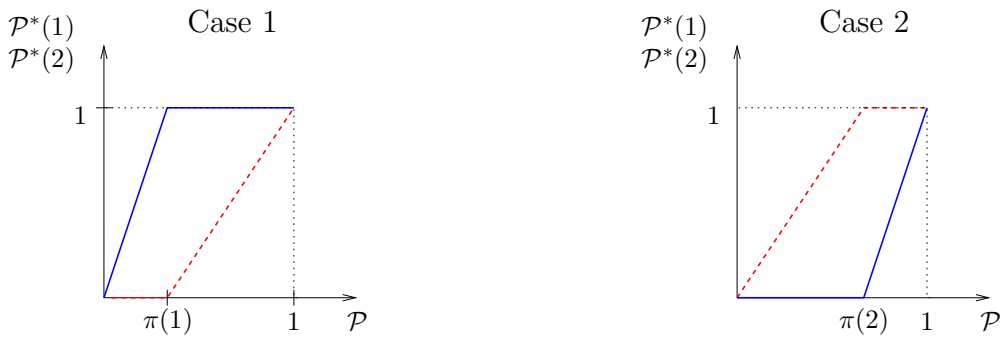



Fig. 2.4: The power allocation policy as a function of the power constraint $\mathcal{P}$ for the 2SMEC with delayed CSI at the transmitter. The solid (dashed) trace corresponds to the power allocation for $\tilde{s} = 1(2)$.

- **Case 1:** $(1-p_{12})(1-\epsilon_1)+p_{12}(1-\epsilon_2) \geq p_{21}(1-\epsilon_1)+(1-p_{21})(1-\epsilon_2)$

$$\mathcal{P}^*(1) = \begin{cases} \frac{\mathcal{P}}{\pi(1)} & \text{if} \quad \mathcal{P} \leq \pi_1 \\ 1 & \text{if} \quad \mathcal{P} > \pi_1 \end{cases}$$

$$\mathcal{P}^*(2) = \begin{cases} 0 & \text{if} \quad \mathcal{P} \leq \pi_1 \\ \frac{\mathcal{P}-\pi_1}{\pi_2} & \text{if} \quad \mathcal{P} > \pi_1 \end{cases}.$$

- **Case 2:** $(1-p_{12})(1-\epsilon_1)+p_{12}(1-\epsilon_2) < p_{21}(1-\epsilon_1)+(1-p_{21})(1-\epsilon_2)$

$$\mathcal{P}^*(1) = \begin{cases} 0 & \text{if} \quad \mathcal{P} \leq \pi_2 \\ \frac{\mathcal{P}-\pi_2}{\pi_1} & \text{if} \quad \mathcal{P} > \pi_2 \end{cases}$$

$$\mathcal{P}^*(2) = \begin{cases} \frac{\mathcal{P}}{\pi_2} & \text{if} \quad \mathcal{P} \leq \pi_2 \\ 1 & \text{if} \quad \mathcal{P} > \pi_2 \end{cases}.$$

The plot for the power allocation policy as a function of the power constraint for the two cases are shown in Figure 2.4. For case 1, when the power constraint $\mathcal{P} \leq \pi_1$, power is allocated only to $\tilde{s} = 1$. This is represented by the solid blue trace. For $\mathcal{P} > \pi_1$, there are more opportunities to transmit then there are "good states" available. As a result, $(\mathcal{P} - \pi_1)$ part of the power constraint is now allocated to $\tilde{s} = 2$ shown by the dashed red trace. This continues till $\mathcal{P} = 1$, at which point all the

states are utilized for transmission. These traces can be interpreted as level-filling results for power allocation in discrete channels, analogous to the one derived in [14]. Case 2 has a similar power allocation process.

### 2.3.2 Capacity-Cost Function for Causal Partial CSI

Next, we present similar results when causal partial CSI is available to the encoder. We denote $z_1^n \in \{0,1\}^n$ as an ACK/NAK sequence of length $n$ and $p(z_1^n)$ is the probability of a particular sequence $z_1^n$.

**Theorem 2.2.** *The capacity-cost function for FSMEC with each state $s \in \mathcal{S}$ associated with an ECV with parameter $\epsilon_s$ and causal partial CSI (ACK/NAK) at the encoder is given by,*

$$C_{FSMEC\text{-}ARQ}(\mathcal{P}) = \lim_{n\to\infty} \sum_{\substack{z_1^n \in \{0,1\}^n \\ s\in\mathcal{S}}} p(z_1^n) \sup_{\mathcal{P}(z_1^n)} \{p(S_{n+d} = s|z_1^n)(1-\epsilon_s)N\mathcal{P}(z_1^n)\}$$
$$s.t. \quad \begin{matrix} \sum_{z_1^n} p(z_1^n)\mathcal{P}(z_1^n) \leq \mathcal{P} \\ 0 \leq \mathcal{P}(z_1^n) \leq 1, \quad \forall\, z_1^n \in \{0,1\}^n \end{matrix} . \tag{2.8}$$

*Here, the optimization is done over $\mathcal{P}(z_1^n)$, which can be thought of as the power allocation policy for the ACK/NAK observation $z_1^n$.*

*Proof.* We calculate the capacity of an energy-constrained finite state Markov erasure channel with causal imperfect CSI by extending the proof given by Yüksel and Tatikonda in [17]. The problem formulation in that work is very similar to ours. The main difference comes from the fact that we consider a causal probabilistic quantizer (i.e., ACK/NAK) for state feedback. In order to incorporate this feature, we first make some modifications to converse of the channel theorem.

If we consider a coding scheme such that $\epsilon \to 0$, Fano's inequality gives,

$$H(W|Y_1^T, S_1^T, Z_1^T) \leq h(p_e) + p_e \log_2(M), \tag{2.9}$$

where $p_e$ gives the probability of error. Also,

$$\begin{aligned} H(W|Y_1^T, S_1^T, Z_1^T) &= H(W) - I(W; Y_1^T, S_1^T, Z_1^T) \\ &= \log_2(M) - I(W; Y_1^T, S_1^T, Z_1^T). \end{aligned} \tag{2.10}$$

Combining Eqs. (2.9) and (2.10) we have,

$$\frac{\log_2(M)}{T} \leq \frac{I(W; Y_1^T, S_1^T, Z_1^T) + h(p_e)}{T(1-p_e)}.$$

As $N \to \infty$, $p_e \to 0$ and we can write,

$$\limsup_{N\to\infty} \frac{\log_2(M)}{T} \leq \limsup_{N\to\infty} \frac{1}{T} I(W; Y_1^T, S_1^T, Z_1^T). \tag{2.11}$$

We focus our attention on the right side of Eq. (2.11),

$$
\begin{aligned}
&\frac{1}{T} I(W; Y_1^T, S_1^T, Z_1^T) \\
&= \frac{1}{T} \sum_{k=1}^{T} I(W; Y_k, S_k, Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}) \\
&\overset{(a)}{=} \frac{1}{T} \sum_{k=1}^{T} I(W, Z_1^{k-1}; Y_k, S_k, Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}) \\
&\qquad - \underbrace{I(Z_1^{k-1}; Y_k, S_k, Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}, W)}_{=0} \\
&= \frac{1}{T} \sum_{k=1}^{T} H(Y_k, S_k, Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}) - H(Y_k, S_k, Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}, W) \\
&\overset{(b)}{=} \frac{1}{T} \sum_{k=1}^{T} H(Y_k, S_k, Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}) - H(Y_k, S_k, Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}, W, X_k) \\
&\overset{(c)}{=} \frac{1}{T} \sum_{k=1}^{T} H(S_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}) + H(Y_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}, S_k) \\
&\qquad + H(Z_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}, S_k, Y_k) - H(S_k | S_1^{k-1}, Z_1^{k-1}, W, X_k) \\
&\qquad - H(Y_k | S_1^{k-1}, Z_1^{k-1}, W, X_k, S_k) - H(Z_k | S_1^{k-1}, Z_1^{k-1}, W, X_k, S_k, Y_k) \\
&= \frac{1}{T} \sum_{k=1}^{T} H(Y_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}, S_k) + \underbrace{H(S_k | S_{k-1})}_{(d)} + H(Z_k | S_k) \\
&\qquad - H(S_k | S_{k-1}) - H(Y_k | S_k, X_k) - \underbrace{H(Z_k | X_k, Y_k)}_{(e)} \\
&= \frac{1}{T} \sum_{k=1}^{T} \underbrace{H(Y_k | Y_1^{k-1}, S_1^{k-1}, Z_1^{k-1}, S_k) - H(Y_k | S_k, U_k)}_{\text{I}} + \underbrace{H(Z_k | S_k)}_{\text{II}}
\end{aligned}
$$

Here, $(a)$ and $(c)$ follow from the chain rule of Mutual Information and entropy respectively, and $(b)$ is due the fact that the channel input $X_k$ is a function of the channel observations $Z_1^{k-1}$ and the message $W$. The above expression can be divided into two parts I and II. Part I is the same as that given by [17] and can be written as a cost which is a function of the coding policy $p(x_k | z_1^{k-1})$. This cost expression is used to recast the maximization problem (2.11) as a Markov Decision Process. Consequently,

the argument for the converse just follows the original proof given in [17]. Part II of the expression is just a constant that does not depend on the coding policy and can be ignored. The direct theorem can be used without modification to complete the proof. □

Theorem 2.2 extends Theorem 2.1 to the case when only ACK/NAK information is available to the encoder. In this case the power allocation policy becomes a function of the observed ACK/NAK sequence instead of the observed state.

## 2.4 Numerical Evaluations

### 2.4.1 Capacity-Cost for Different Feedback Schemes

Figure 2.5 shows the comparison of the theoretical bounds for schedulers with different CSI for a 2SMEC. The state space in this case is $\mathcal{S} = \{1, 2\}$, and the channel parameters used are $\epsilon_1 = 0.2$ and $\epsilon_2 = 0.8$. The transition probability matrix is given by $\mathrm{P} = \begin{bmatrix} 0.8 & 0.2 \\ 0.1 & 0.9 \end{bmatrix}$. In all capacity cost functions with feedback a knee-point in the curve is observed when $\mathcal{P}$ is equal to 0.33. This is due to the fact that as the number of data units to be transmitted is increased, the cost will increase at a certain rate since only the good states are being used initially. However after all the good states are exhausted, the controller will use the bad states for transmission, which increases cost at a faster rate as more packets will be dropped. The dotted green trace represents a blind transmission scheduler. Blind scheduler refers to a transmission scheduler that does not use any type of CSI. A blind scheme will give the lower bound on performance of any channel aware transmission scheme. The curve $C(\mathcal{P})$ for this scheduler has a constant slope, since the controller does not adapt the transmission policy to the channel state. As the power constraint is increased to unity, making it an

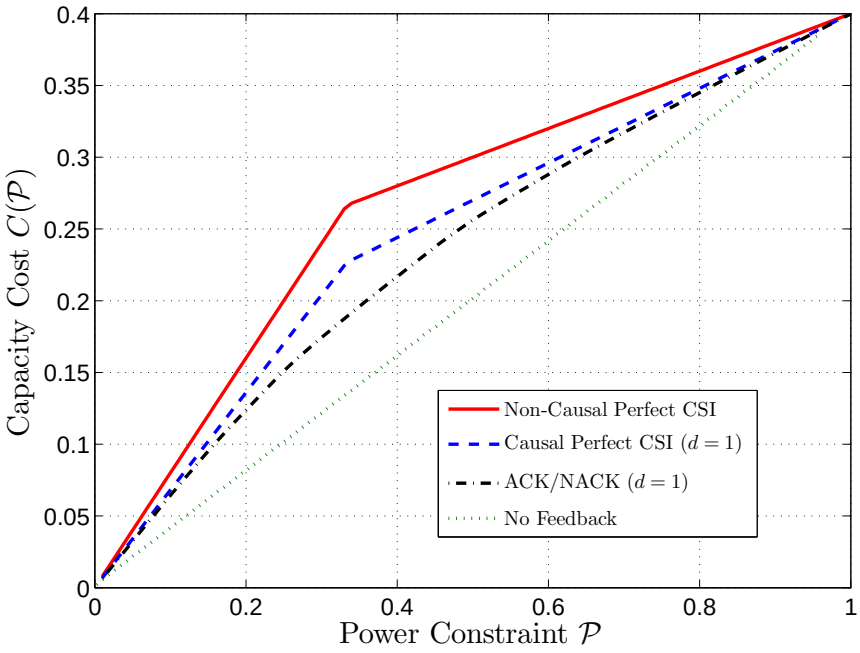



Fig. 2.5: Comparison of the capacity-cost functions for the 2SMEC with different feedback. Here $\epsilon_1 = 0.2, \epsilon_2 = 0.8, p_{21} = 0.1$ and $p_{12} = 0.2$.

unconstrained problem, the performance of all the schemes converge. This happens due to the controller attempting transmission in each slot regardless of the channel state.

### 2.4.2 Maximum Power Penalty

Next, we find the maximum power penalty incurred by the feedback schemes using ACK/NAK information. We define the maximum power penalty as $A_{\text{dB}} \triangleq \max_R 10 \log_{10} \left( \frac{\text{Cost-Capacity function for rate } R \text{ using ACK/NAK feedback}}{\text{Cost-Capacity function for rate } R \text{ using causal CSI feedback}} \right)$. Cost-Capacity function is the inverse of the Capacity-Cost function. Maximum power penalty quantifies the worst-case extra cost (power) required by an ACK/NAK based scheme to achieve

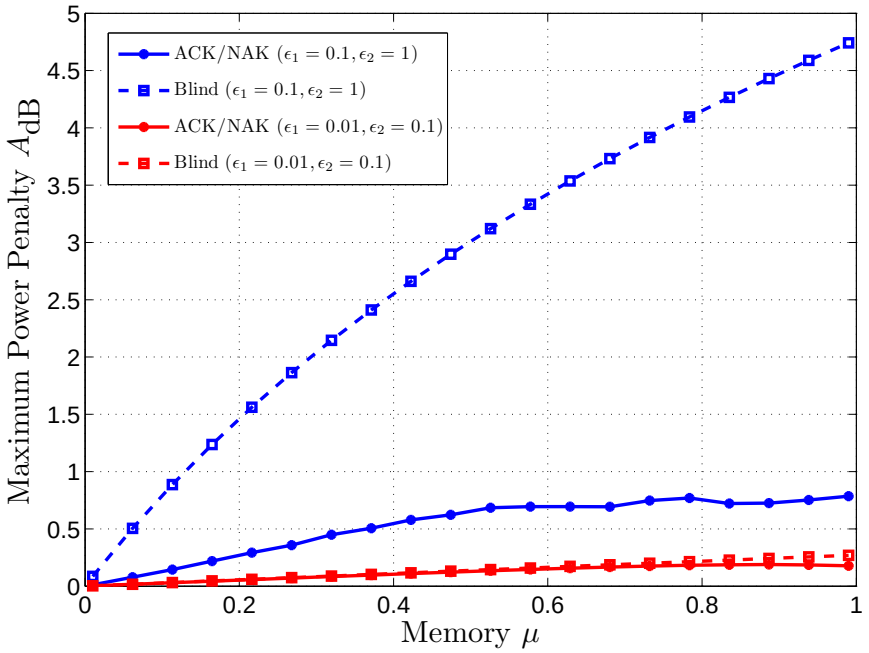



Fig. 2.6: The maximum power penalty as a function of the channel memory for different values of packet loss probabilities $\{\epsilon_1, \epsilon_2\}$ with $\pi(1) = 1/3$ and $\pi(2) = 2/3$.

the same transmission rate achieved by a causal perfect CSI based scheme. To complete the comparison, we define a similar quantity for blind scheme, i.e., the case when no feedback is used for scheduling transmissions.

Figure 2.6 compares the maximum power penalties incurred by ACK/NAK and blind schemes as a function of channel memory $\mu \triangleq 1 - p_{12} - p_{21}$. We plot this quantity for different values of loss probabilities $\{\epsilon_1, \epsilon_2\}$, while keeping the steady state probabilities $\{\pi(1), \pi(2)\}$ constant. For the range of values considered in our evaluations, the ACK/NAK based schemes do not incur a maximum power penalty of more than 1 dB. On the other hand, in case of no feedback (blind) the maximum power penalty can be as much as 4.5 dB. An interesting observation is that when the channel conditions are "good" (i.e., $\epsilon_1 = 0.01$ and $\epsilon_2 = 0.1$) the advantage of having

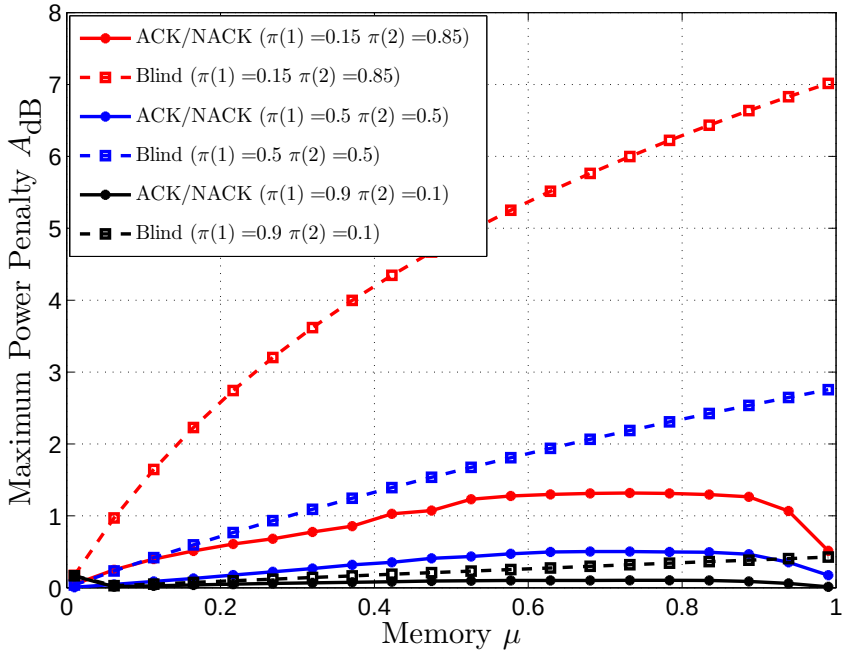



Fig. 2.7: The maximum power penalty as a function of the channel memory for different values of steady state probabilities $\{\pi(1), \pi(2)\}$ with $\epsilon_1 = 0.1$ and $\epsilon_2 = 0.99$.

ACK/NAK feedback over causal CSI is negligible. The difference in maximum power penalties for the two cases is less than 0.1 dB in this case. This can be explained by noting that under good channel conditions, there will be negligible packet drops and consequently, the value of estimating the channel (using feedback) is minimal.

In Figure 2.7, we plot the maximum power penalties for different values of the steady state probability $\{\pi(1), \pi(2)\}$, while keeping the loss probabilities constant $(\epsilon_1 = 0.1, \epsilon_2 = 0.95)$. In this case, the ACK/NAK feedback based scheduler incurs a maximum power penalty of upto 1.5 dB when $\{\pi(1) = 0.15, \pi(2) = 0.85\}$. When no feedback is provided to the transmitter, the maximum power penalty can be as high as 7 dB. We also note from Figures 2.6 and 2.7 that as the channel memory $(1-p_{12}-p_{21})$ increases, the difference in maximum power penalties between the case of ACK/NAK

feedback and no feedback also increases. This is explained by the fact that for higher values of channel memory, the ACK/NAK feedback will provide estimates of the channel state which are close to that of causal perfect CSI. From these result, we can draw the conclusion that using ACK/NAK feedback to schedule transmissions will lead to substantial energy savings as compared to using no information at all. Channel memory is an indicator of burstiness in Markov channel. This leads to the conclusion that ACK/NAK feedback based schedulers will provide best results when the wireless channel is bursty.

# CHAPTER 3

# ENERGY OPTIMAL TRANSMISSION SCHEDULER FOR SENSOR NETWORKS

## 3.1 Introduction

As we observed in Chapter 2, without any effort for adapting to the variability in the channel, the energy resources are consumed inefficiently. Intuitively, if bad channel conditions are not anticipated, a high fraction of the node energy can be consumed by multiple retransmissions per correctly decoded packet. To avoid such inefficiencies, sensor nodes use transmission schedulers. The objective of an energy-efficient transmission scheduler is to reliably communicate data using minimal amount of energy, while meeting deadline and/or throughput constraints.

We propose a transmission scheduler that utilizes different grades of CSI on the state of a FSMC to schedule packet transmissions in order to meet a deadline constraint for the packets waiting in the transmission queue. We are specifically interested in the highly imperfect CSI that can be obtained based solely on the ACK/NAK sequence for the past transmissions. Note that, this level of information is available at the link layer in almost all wireless networks without any extra effort (such as sending special non-information carrying physical-layer pilot symbols over the channel). Our

transmitter has a single transmit power level and the coding and modulation schemes are fixed. In every transmission opportunity, the transmitter can choose to attempt the packet transmission or defer it.

Our scheduler is based on the dynamic programming (DP) solution to a finite-horizon transmission control problem. We show how a simple version of it can be implemented in sensor nodes. We will evaluate the performance of our scheduler and illustrate that it achieves a given throughput at a power level that is fairly close to the fundamental limit achievable over the channel.

In the transmission scheduling problem, the parameter we minimize is the number of transmission attempts, subject to a deadline constraint for the packets in the queue. One can realize that, with the limitation of binary power control (transmit or do not transmit), the number of transmissions is proportional to the energy consumed per correctly decoded packet and the deadline constraint can be translated into a throughput constraint at every point in time[2]. Hence the problem can be viewed as minimizing the energy per packet subject to a throughput constraint.

Transmission strategies based on channel estimation in FSMCs have been considered by Zorzi and Rao [22] as well as Chiasserini and Meo [23]. These papers assume a two-state Markov chain and detect the bad channel state upon receipt of a NAK. Both schemes reduce the transmission rate and constantly probe the channel as a response. In [23], the transmitter switches to a greedy transmission mode when the buffer level exceeds a certain threshold, regardless of the channel state. Our scheme is based on a dynamic program, which takes the queue and the estimated channel states jointly into account to schedule packet transmissions in an optimal manner.

[2]For instance if the buffer contains 5 packets with a deadline constraint of 1 sec, then the scheduler has to guarantee a minimum of 5 packets/sec for the next second

Johnston and Krishnamurthy [24] give an algorithm that minimizes the transmission energy and latency while transmitting over a fading channel by formulating the problem as a partially observed Markov decision process (POMDP) search problem. However, their threshold-based policy result is optimal for a 2-state channel only. In addition, the channel model is assumed to have a unity packet loss probability in one state. This may not seem like a fundamental difference, but in such a scenario, an ACK implies a "good" state. The corresponding time dependence is finite and the associated solution can exploit it. Our results are valid for a general FSMC.

Uysal, et. al., [25] and Zafer, et. al., [26] have considered rate control policies for transmission scheduling similar to ours. However, [25] uses a static channel model for deriving the control policy. On the other hand, [26] considers a Markov channel, but assumes knowledge of the channel state prior to transmission. In our model, this information is not available to the controller. Haleem, et. al., [27] use ACK/NAK feedback in a learning automata algorithm to schedule transmissions. This method is shown to converge to the optimal throughput in stationary channels. Ho, et. al., [28] give a sub-optimal rate adaptation scheme to maximize throughput that uses ACK/NAK feedback. Karmokar, et. al., [29] pose the problem of rate adaptation in Type-I Hybrid ARQ systems as a POMDP and provide some heuristic solutions. The authors develop this idea further and propose a linear programming approach to solve the POMDP problem [30, 31].

The chapter is organized as follows. We give our system model and the assumptions in Section 3.2. We present the DP formulation of the problem in Section 3.3. We discuss some implementation issues in Section 3.4. In Section 3.5, we analyze the

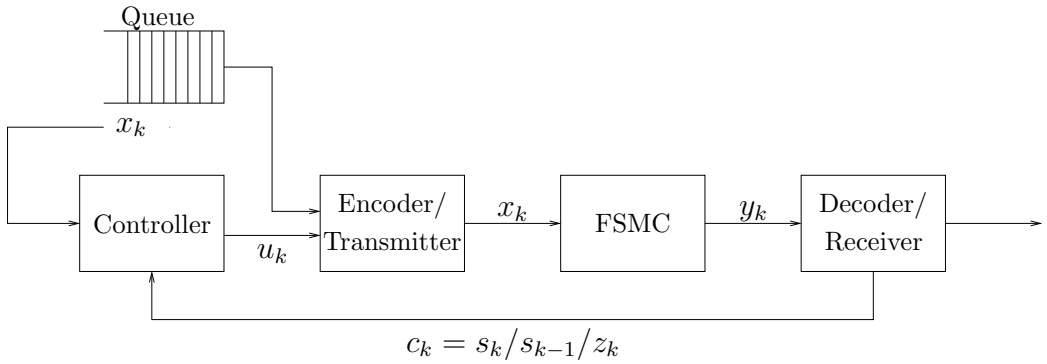



Fig. 3.1: Block diagram showing the system model considered in this work.

performance of the DP scheduler and compare it with the theoretical bounds derived in Chapter 2.

## 3.2 System Model

The system model used in this work is shown in Figure 3.1. Here, we consider a single point-to-point wireless link with a feedback path. The transmitter has $B$ packets in a queue to be sent to the receiver within $T$ time slots. The transmitter is assumed to transmit at a fixed rate and we define a time slot as the time it takes to transmit a packet and receive the associated ACK/NAK. The controller, which is attached to the transmitter, determines the decision to attempt a transmission or to defer transmission of a packet in every time slot. For the $k$th time slot, we denote this packet (or lack of it) as $x_k$. The receiver decodes the packet $y_k$ and sends feedback $c_k$ about the channel state to the controller over an error-free channel. The different types of feedback will be discussed shortly. At time $k$, the controller has access to the queue state information (i.e., number of packets in the queue, $q_k$) and the feedback $c_k$. We use the following notation for a sequence, $c_m^n \triangleq \{c_m, c_{m+1}, \ldots, c_n\}$. The

information vector $I_k = \{q_0^k, c_0^k\}$ is defined as the information available at time $k$ to the controller. The controller makes control decision $u_k \in \{0, 1\}$ based on the information vector, $I_k$. Here '0' corresponds to the decision not to transmit and '1' corresponds to the decision to transmit packet $x_k$.

As previously, the channel model that we consider for this problem is a FSMC. The channel state at time $k$ is denoted by $s_k \in \mathcal{S}$, determines the packet loss probability during the $k$th time slot. The state space of the Markov channel, $\mathcal{S}$ is a discrete set containing the different channel states of the FSMC. Each channel state $s \in \mathcal{S}$ has a certain packet loss probability denoted by $\epsilon_s$. The channel state is assumed to be constant during the period of packet transmission, i.e., one time slot. The steady state probability of state $s$ is represented by $\pi(s)$.

We assume that the controller has accurate estimates of the channel parameters, i.e., transition probabilities and the packet loss probabilities associated with each state. Estimation of these channel parameters is beyond the scope of this paper. We note that this topic has been treated widely in literature, where an FSMC with unknown parameters is posed as a hidden Markov model (HMM). Iterative procedures for the Maximum Likelihood estimation of HMM parameters are well-understood, e.g., Baum-Welch algorithm [32].

The receiver is assumed to have an error detection scheme with a negligible probability of undetected error. The decoder identifies whether a packet is decoded correctly or not, i.e., $y_k = x_k$ or $y_k \neq x_k$. Packet error process at time $k$ is denoted by $z_k$, where $z_k = 1$ for a correct transmission and $z_k = 0$ for an incorrect transmission. The receiver sends the feedback containing the CSI to the controller on a channel

assumed to be error-free. Like chapter 2, we will consider the following cases/grades of receiver feedback to the controller:

1. **Non-causal Perfect CSI:** the controller knows the instantaneous channel state when it determines $u_k$, i.e., $c_k = s_k$ and $I_k = \{q_0^k, s_0^k\}$.

2. **Causal Perfect CSI:** the controller knows the delayed channel state when it determines $u_k$, i.e., $c_k = s_{k-1}$ and $I_k = \{q_0^k, s_0^{k-1}\}$.

3. **Causal Partial CSI:** in this case, the controller knows the ACK/NAK for the previous transmissions, i.e., $c_k = z_{k-1}$ and $I_k = \{q_0^k, z_0^{k-1}\}$.

For clarity of presentation, we use a unit delay for the causal CSI cases. The results contained in this chapter can be easily extended to a general fixed delay.

## 3.3 The Dynamic Programming Solution

In this section we develop an algorithm/decision rule for the controller that minimizes the energy expended by the transmitter and the receiver for the successful delivery of a certain number of data packets while maintaining a deadline requirement. Intuitively, without a deadline constraint, the controller would be inclined to transmit at times only when it is almost sure of the good channel state to achieve energy efficiency. However, with the deadline constraint, if there are packets remaining in the queue close to the deadline, it may need to take chances with the bad state occasionally as well. Thus the controller has to make decisions on packet transmission, based jointly on the queue and channel states. Note that the controller needs to consider the effect of the current decision on future decisions. For instance, if the controller decides not to transmit at a time slot, there will be no feedback on the

channel state for that time slot. This makes the available information more outdated, affecting the success of subsequent decisions made. In our system the CSI used by the controller is provided solely by the receiver. We assume that the initial queue state is also known by the receiver, hence the receiver also has the entire information vector. Consequently, the receiver has the knowledge of whether a transmission will be attempted at any given point in time and it can remain inactive to conserve energy during skipped transmission attempts.

Since the packet-loss process is a stochastic process, this problem can be viewed as one of sequential decision making under uncertainty. In our system model, both the queue evolution and channel evolution have a Markov structure. The queue state can be observed completely by the controller, however for the channel state, we have different cases of observation (complete and partial). The combination of these factors necessitates the use of a finite horizon DP approach [33] to achieve energy optimality and meet the deadline constraint at the same time.

### 3.3.1 The Dynamic Program

Using the notation introduced in the previous section, we can write the state equation for the queue occupancy $q_k$ at the beginning of time slot $k = 0, 1, \ldots, T-1$, as follows,

$$q_{k+1} = f(q_k, s_k, u_k) = q_k - 1_{\epsilon_{s_k}} u_k, \tag{3.1}$$

where $u_k \in \{0, 1\}$. The packet loss process is denoted by $1_{\epsilon_{s_k}}$, where $1_{\epsilon_{s_k}} = 0$ with probability $\epsilon_{s_k}$ and $1_{\epsilon_{s_k}} = 1$ otherwise.

Our objective is to minimize the energy consumption of the transmitter while transmitting $B$ packets over $T$ time slots. Consequently, the cost function should

be proportional to the total energy of the transmissions required to transmit all packets correctly. In order to do this, we set the cost incurred at time $k$, denoted by $g_k(q_k, s_k, u_k)$, equal to $u_k$. When the controller decides to transmit, it incurs a cost of 1 unit. On the other hand, if the controller decides to defer transmission, then it does not incur any cost. Since the packet loss is a stochastic process, there is always a non-zero probability associated with not transmitting all packets correctly by the deadline. To make this undesirable, we use a terminal cost $g_T(q_T) = Cq_T$, $C \gg 0$. Terminal cost implies that if there are $q_T$ packets left in the buffer at the end of time $T$, then the controller will incur a cost of $Cq_T$. One would expect that the decision rule and hence the performance is strongly tied to the choice of $C$, but we show that, somewhat surprisingly, in most simulations the choice of $C$ has little effect on the performance of the scheduler as long as $C \geq 10$. We can express the expected cost incurred from the $(T-1)$th stage to termination, also called the *cost-to-go function* $J_{T-1}(I_{T-1})$ as,

$$\begin{aligned} &J_{T-1}(I_{T-1}) \\ &\quad = \min_{u_{T-1}\in\mathcal{U}} \left\{\mathbb{E}_{s_{T-1}}\left[g_T(f(q_{T-1}, s_{T-1}, u_{T-1})) + g_{T-1}(q_{T-1}, s_{T-1}, u_{T-1})|I_{T-1}, u_{T-1}\right]\right\} \\ &\quad = \min_{u_{T-1}\in\mathcal{U}} \left\{u_{T-1} + \mathbb{E}_{s_{T-1}}\left[Cf(q_{T-1}, s_{T-1}, u_{T-1})|I_{T-1}, u_{T-1}\right]\right\}. \end{aligned} \tag{3.2}$$

The cost-to-go $J_k(I_k)$ for stages $k = 0, 1, \ldots, T-2$, can be expressed iteratively by using successive cost-to-go functions $J_{k+1}(I_{k+1})$,

$$\begin{aligned} J_k(I_k) &= \min_{u_k\in\mathcal{U}} \left\{\mathbb{E}_{s_k, c_{k+1}}\left[J_{k+1}(I_k, q_{k+1}, c_{k+1}, u_k) + g_k(q_k, s_k, u_k)|I_k, u_k\right]\right\} \\ &= \min_{u_k\in\mathcal{U}} \left\{u_k + \mathbb{E}_{s_k, c_{k+1}}\left[J_{k+1}(I_k, q_{k+1}, c_{k+1}, u_k)|I_k, u_k\right]\right\}. \end{aligned} \tag{3.3}$$

We note that these optimization problems reduce to a DP with perfect state information when non-causal perfect CSI is available. In case of causal CSI (perfect and

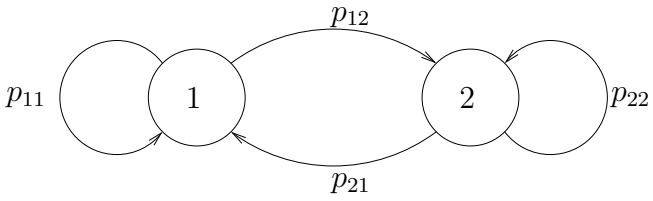



Fig. 3.2: A 2-state Markov channel with two states, "1" and "2". The associated loss probabilities are $\epsilon_1$ and $\epsilon_2$ and the transitions have been labeled with the respective probabilities.

partial) these problems are treated as DP with imperfect state information. At each $k = 0, 1, \ldots, T-1$, the optimal policy $\mu_k^*(I_k)$ maps the information vector $I_k$ to the control action $u_k \in \{0, 1\}$ that minimizes the cost-to-go given in (3.2) and (3.3). The optimal policy is obtained recursively, we first solve the optimization problem (3.2) for all possible values of the information vector $I_{T-1}$ to get $\mu_{T-1}^*(I_{T-1})$. The corresponding value of $J_{T-1}^*(I_{T-1})$ is used to calculate $\mu_{T-2}^*(I_{T-2})$ in (3.3) and $J_{T-2}^*(I_{T-2})$. This process is continued till $J_0^*(I_0) = J_0^*(q_0, c_0)$ is found. We note that, in general, at each stage $k$ the state space size of $I_k$ will grow exponentially. However, if the components of $I_k$ follow a Markov transition (e.g., perfect CSI feedback) then only the most recent state observations are required for determining the control decision keeping the state space size constant.

**Example 3.1 2-state Markov channel with perfect causal CSI**

To illustrate the DP algorithm, we present a simple example, where a transmission policy for one packet ($B = 1$) is to be optimized over two time slots ($T = 2$). We consider the two-state Markov channel shown in Figure 3.2 with $p_{12} = 0.2$ and $p_{21} = 0.1$. The possible queue states are $q_k \in \{0, 1\}$ and the possible channel states are $s_k \in \{1, 2\}$. The probability of errors are $\epsilon_1 = 0$ and $\epsilon_2 = 1$ and perfect causal

CSI is known to the controller, i.e., $c_k = s_{k-1}$. If we set $C = 20$, the terminal cost, $g_2(q_2 = 0) = 0$ and $g_2(q_2 = 1) = 20$. To obtain the optimal policy for the 1st stage, we calculate the optimal action $\mu_1^*(q_1, c_1)$ given by the minimizing argument of (3.2) for all possible values of the state $\{q_1, c_1\}$. Using this control law, we can find the corresponding optimal cost-to-go function, $J_1^*(q_1, c_1)$. The values of $\mu_1^*(q_1, c_1)$ and $J_1^*(q_1, c_1)$ will be,

$$\mu_1^*(q_1, s_1) = \begin{cases} 0, & q_1 = 0, s_1 = 1, 2 \\ 1, & q_1 = 1, s_1 = 1 \\ 1, & q_1 = 1, s_1 = 2 \end{cases} \quad \text{and} \quad J_1^*(q_1, c_1) = \begin{cases} 0, & q_1 = 0, c_1 = 1, 2 \\ 5, & q_1 = 1, c_1 = 1 \\ 19, & q_1 = 1, c_1 = 2 \end{cases},$$

i.e., the controller will transmit whenever the queue is non-empty. This expression is substituted in (3.3) to find $\mu_0^*(q_0, c_0)$ and $J_0^*(q_0, c_0)$ for the zeroth stage,

$$\mu_0^*(q_0, c_0) = \begin{cases} 1, & q_0 = 1, c_0 = 1 \\ 0, & q_0 = 1, c_0 = 2 \end{cases} \quad \text{and} \quad J_0^*(q_0, c_0) = \begin{cases} 4.8, & q_0 = 1, c_0 = 1 \\ 17.6, & q_0 = 1, c_0 = 2 \end{cases},$$

i.e., the controller will transmit only when the CSI $c_0 = 1$. Note that $q_0 = B = 1$.

Next, we derive the dynamic programming equations for the various cases of feedback discussed in Section 3.2.

### 3.3.2 DP Equation for Non-causal Perfect CSI at the Controller

Non-causal CSI is the case when the channel information is known to the controller in advance and $\{I_k = q_1^k, s_1^k\}$. Since both the queue state and channel state follow a Markov transition, we can reduce the conditioning on $I_k$ in Eqs. (3.2) and (3.3) to the most recently observed states, i.e., $\{q_k, s_k\}$. In fact, since this is the only information used by the controller, we can state that $I_k = \{q_k, s_k\}$. Using this fact, we can rewrite

these equations as,

$$J_{T-1}(q_{T-1}, s_{T-1}) = \min_{u_{T-1}\in\{0,1\}} \big\{u_{T-1} + \epsilon_{s_{T-1}} C q_{T-1} + (1-\epsilon_{s_{T-1}}) C(q_{T-1} - u_{T-1})\big\}, \tag{3.4}$$

for the terminating stage. Similarly, we can find the equations for the cost functions in the intermediate stages $k = 1, 2, \ldots, T-2$,

$$J_k(q_k, s_k) = \min_{u_k\in\{0,1\}} \Big\{u_k + \epsilon_{s_k} \sum_{s_{k+1}\in\mathcal{S}} \mathrm{P}(s_{k+1}, s_k) J_{k+1}(q_k, s_{k+1}) + (1-\epsilon_{s_k}) \sum_{s_{k+1}\in\mathcal{S}} \mathrm{P}(s_{k+1}, s_k) J_{k+1}(q_k - u_k, s_{k+1})\Big\}. \tag{3.5}$$

Here, P is the one-step transition probability matrix and we define $\mathrm{P}(s_{k+1}, s_k) \triangleq p(S_{k+1} = s_{k+1} | S_k = s_k)$. These equations can be solved iteratively to derive a decision rule for the causal CSI case.

### 3.3.3 DP Equation for Causal Perfect CSI at the Controller

For the causal CSI case, we follow a similar approach. When causal CSI is available to the controller, then $I_k = \{q_1^k, s_1^{k-1}\}$. As argued previously, since both these state variables follow a Markov chain, the cost function conditioned on the most recent state observations will be independent of the previous states. In other words, we can write $I_k = \{q_k, s_{k-1}\}$. Therefore, we can write the cost-to-go function $J_k$ given by Eqs. (3.2) and (3.3) as,

$$\begin{aligned} &J_{T-1}(q_{T-1}, s_{T-2}) \\ &= \min_{u_{T-1}\in\{0,1\}} \Big\{u_{T-1} + \sum_{s_{T-1}\in\mathcal{S}} \mathrm{P}(s_{T-1}, s_{T-2})(\epsilon_{s_{T-1}} C q_{T-1} + (1-\epsilon_{s_{T-1}}) C(q_{T-1} - u_{T-1}))\Big\}, \end{aligned} \tag{3.6}$$

where $c_k = s_{k-1}$ is the CSI available at the controller at time $k$. For $k = 1, 2, \ldots, T-2$,

$$J_k(q_k, s_{k-1}) = \min_{u_k \in \{0,1\}} \Big\{ u_k + \sum_{s_k \in \mathcal{S}} \mathrm{P}(s_k, s_{k-1}) \Big( \epsilon_{s_k} \sum_{s_{k+1} \in \mathcal{S}} \mathrm{P}(s_{k+1}, s_k) J_{k+1}(q_k, s_{k+1}) + (1 - \epsilon_{s_k}) \sum_{s_{k+1} \in \mathcal{S}} \mathrm{P}(s_{k+1}, s_k) J_{k+1}(q_k - u_k, s_{k+1}) \Big) \Big\}. \quad (3.7)$$

### 3.3.4 DP Equation for Causal Partial CSI at the Controller

Recall that with causal partial CSI, $I_k = \{q_1^k, z_1^{k-1}\}$. The queue state, $q_k$, follows a Markov transition, so we can reduce the information vector to $I_k = \{q_k, z_1^{k-1}\}$. We cannot apply the same reduction to the ACK/NAK sequence, $z_k$. This is due to the fact that, even though $z_k$ is a function of the channel state $s_k$, in general, the probability distribution of $z_k$ does not depend only on $z_{k-1}$ but all the past observations, $z_1^{k-1}$. The problem with directly applying the DP approach, (3.2) and (3.3), in this case is that the state space of the information vector will expand exponentially with the received observations. To avoid this problem, we introduce a new quantity $\boldsymbol{w_k} = \{w_k(1), \ldots, w_k(|\mathcal{S}|)\}$ which is the conditional state distribution given the past sequence, i.e., $w_k(s) \triangleq p\left(s_k = s \;\middle|\; z_1^{k-1}\right)$. The conditional state distribution follows a Markov transition, $\boldsymbol{w_k} = \Phi(\boldsymbol{w_{k-1}}, z_{k-1}, u_{k-1})$, which can be derived using a straightforward application of Bayes' rule. Since this quantity depends on observations and control actions from the previous stage only, the controller needs to track only the most recent value of this variable. Evaluating the expectation in (3.2) and (3.3), using $\boldsymbol{w_k}$, we write the cost-to-go function $J_k$. For the terminal stage,

$$J_{T-1}(q_{T-1}, \boldsymbol{w_{T-1}}) = \min_{u_{T-1}} \Big\{ u_{T-1} + \sum_{s \in \mathcal{S}} w_{T-1}(s) \left( \epsilon_s C q_{T-1} + (1 - \epsilon_s) C(q_{T-1} - u_{T-1}) \right) \Big\}, \quad (3.8)$$

and for the intermediate stages, $k = 0, \ldots, T-2$,

$$J_k(q_k, \boldsymbol{w_k}) = \min_{u_k} \Big\{ u_k + \sum_{s \in \mathcal{S}} w_k(s)\epsilon_s J_{k+1}(q_k, \Phi(\boldsymbol{w_k}, 0, u_k)) + \sum_{s \in \mathcal{S}} w_k(s)(1-\epsilon_s) J_{k+1}(q_k - u_k, \Phi(\boldsymbol{w_k}, 1, u_k)) \Big\}. \quad (3.9)$$

Note that (3.8) and (3.9) are valid for any discrete set $\mathcal{S}$. The recursive relation for $\boldsymbol{w_k}$ is given by,

$$\boldsymbol{w_k} = \Phi(\boldsymbol{w_{k-1}}, z_{k-1}, u_{k-1}) = \frac{\boldsymbol{w_{k-1}} \mathrm{A}(z_{k-1}, u_{k-1}) \mathrm{P}}{\boldsymbol{w_{k-1}} \mathrm{A}(z_{k-1}, u_{k-1}) \mathbf{1}^T}, \quad (3.10)$$

with $\boldsymbol{w_0}$ initialized as the steady state distribution of states, $\{\pi_s, \forall s \in \mathcal{S}\}$. Here $\mathrm{A}(z_{k-1}, u_{k-1})$ is a diagonal $|\mathcal{S}| \times |\mathcal{S}|$ matrix with $i$th diagonal term defined as,

$$p\left(z_{k-1} \mid S_{k-1} = i, u_{k-1}\right) = \begin{cases} \epsilon_i, & z_{k-1} = 0, u_{k-1} = 1 \\ 1 - \epsilon_i, & z_{k-1} = 1, u_{k-1} = 1 \\ 1, & u_{k-1} = 0 \end{cases},$$

P is the transition probability matrix for the FSMC, and $\mathbf{1} = \{1, \ldots, 1\}$ is a $|\mathcal{S}|$-dimensional vector. We note here that when $u_{k-1} = 0$, there is no transmission in the $(k-1)$th slot and $z_{k-1}$ is not known. When this "gap" appears in the ACK/NAK sequence, $\boldsymbol{w_k}$ is updated by vector multiplication with the transition matrix P. Intuitively, the conditional state estimate will be less accurate when there are gaps in the ACK/NAK sequence which leads to a degradation in the performance of the DP-scheduler. As expected, with increasing gap size, the state estimate converges to the steady state probability.

Next, we derive the recursive relation for the conditional state distribution $w_k(s) = p(S_k = s | z_1^{k-1})$. We consider the two possible cases individually.

*Case 1:* ACK/NAK for the $(k-1)$th slot is available

In this case $u_{k-1} = 1$, which means that there is transmission in the $(k-1)$th slot. In this case the ACK/NAK $z_{k-1}$ is available to the controller to update the

conditional state distribution. We can write,

$$\begin{aligned} p(S_{k-1}=\bar{s}|z_1^{k-1}) &= \frac{p(S_{k-1}=\bar{s}, z_1^{k-1})}{p(z_1^{k-1})} \\ &= \frac{p(z_{k-1}|S_{k-1}=\bar{s})p(S_{k-1}=\bar{s}, z_1^{k-2})}{p(z_1^{k-1})} \\ &= \frac{p(z_{k-1}|S_{k-1}=\bar{s})p(S_{k-1}=\bar{s}|z_1^{k-2})p(z_1^{k-2})}{p(z_1^{k-1})}. \end{aligned} \tag{3.11}$$

And,

$$\begin{aligned} p(z_1^{k-1}) &= \sum_{\hat{s}\in\mathcal{S}} p(z_1^{k-1}, S_{k-1}=\hat{s}) \\ &= \sum_{\hat{s}\in\mathcal{S}} p(z_{k-1}|S_{k-1}=\hat{s})p(S_{k-1}=\hat{s}, z_1^{k-2}) \\ &= \sum_{\hat{s}\in\mathcal{S}} p(z_{k-1}|S_{k-1}=\hat{s})p(S_{k-1}=\hat{s}|z_1^{k-2})p(z_1^{k-2}). \end{aligned} \tag{3.12}$$

Substituting Eq. (3.12) in Eq. (3.11), we have,

$$p(S_{k-1}=\bar{s}|z_1^{k-1}) = \frac{p(z_{k-1}|S_{k-1}=\bar{s})p(S_{k-1}=\bar{s}|z_1^{k-2})p(z_1^{k-2})}{\sum_{\hat{s}\in\mathcal{S}} p(z_{k-1}|S_{k-1}=\hat{s})p(S_{k-1}=\hat{s}|z_1^{k-2})p(z_1^{k-2})}. \tag{3.13}$$

Finally we can write $w_k(s)$ as,

$$\begin{aligned} w_k(s) = p(S_k=s|z_1^{k-1}) &= \sum_{\bar{s}\in\mathcal{S}} p(S_k=s|S_{k-1}=\bar{s})p(S_{k-1}=\bar{s}|z_1^{k-1}) \\ &= \frac{\sum_{\bar{s}\in\mathcal{S}} p(z_{k-1}|S_{k-1}=\bar{s})p(S_{k-1}=\bar{s}|z_1^{k-2})p(S_k=s|S_{k-1}=\bar{s})}{\sum_{\hat{s}\in\mathcal{S}} p(z_{k-1}|S_{k-1}=\hat{s})p(S_{k-1}=\hat{s}|z_1^{k-2})} \\ &= \frac{\sum_{\bar{s}\in\mathcal{S}} p(z_{k-1}|S_{k-1}=\bar{s})w_{k-1}(\bar{s})p(S_k=s|S_{k-1}=\bar{s})}{\sum_{\hat{s}\in\mathcal{S}} p(z_{k-1}|S_{k-1}=\hat{s})w_{k-1}(\hat{s})}. \end{aligned} \tag{3.14}$$

*Case 2:* ACK/NAK for the $(k-1)$th slot is not available

In this case $u_{k-1}=0$, which means there is no transmission in the $(k-1)$th slot. As a result, the controller does not have access to the ACK/NAK $z_{k-1}$ to calculate

the update equations for $w_k(s)$. We can write,

$$\begin{aligned} w_k(s) = p(S_k = s|z_1^{k-1}) &= \sum_{\bar{s}\in\mathcal{S}} p(S_k = s|S_{k-1} = \bar{s}, z_1^{k-2}) p(S_{k-1} = \bar{s}|z_1^{k-2}) \\ &= \sum_{\bar{s}\in\mathcal{S}} p(S_k = s|S_{k-1} = \bar{s}) p(S_{k-1} = \bar{s}|z_1^{k-2}) \\ &= \sum_{\bar{s}\in\mathcal{S}} p(S_k = s|S_{k-1} = \bar{s}) w_{k-1}(\bar{s}). \end{aligned} \tag{3.15}$$

Eqs. (3.14) and (3.15) can be written compactly in vector form given in Eq. (3.10). For a general fixed delay $d > 1$, the conditional state distribution update does not have a recursive relation. However, using simple manipulations, we can derive an evaluation process that has an iterative structure to it. We can write,

$$\begin{aligned} p(S_k = s|z_1^{k-d}) &= \sum_{\bar{s}\in\mathcal{S}} p(S_k = s, S_{k-d+1} = \bar{s}|z_1^{k-d}) \\ &= \sum_{\bar{s}\in\mathcal{S}} p(S_k = s|S_{k-d+1} = \bar{s}) p(S_{k-d+1} = \bar{s}|z_1^{k-d}) \\ &= \sum_{\bar{s}\in\mathcal{S}} p(S_k = s|S_{k-d+1} = \bar{s}) w_{k-d+1}(\bar{s}). \end{aligned} \tag{3.16}$$

Note that $w_{k-d+1}(\bar{s})$ has a recursive relation. This provides us a two-step iterative method to evaluate the conditional state distribution:

1. Update $\boldsymbol{w_{k-d}} \rightarrow \boldsymbol{w_{k-d+1}}$ using Eq. (3.10).

2. Calculate $p(S_k = s|z_1^{k-d})$ by evaluating the vector $\mathrm{P}^d \boldsymbol{w_{k-t+1}}$ and choosing the appropriate entry. Here $\mathrm{P}^d$ is the $d$-step transition matrix for the FSMC.

## 3.4 Implementation Issues

Next, we discuss some implementation issues associated with the DP-scheduler. First, consider the example of a two-state Markov channel with ACK/NAK information. The optimization problems (3.8) and (3.9) can be simplified for a 2-state

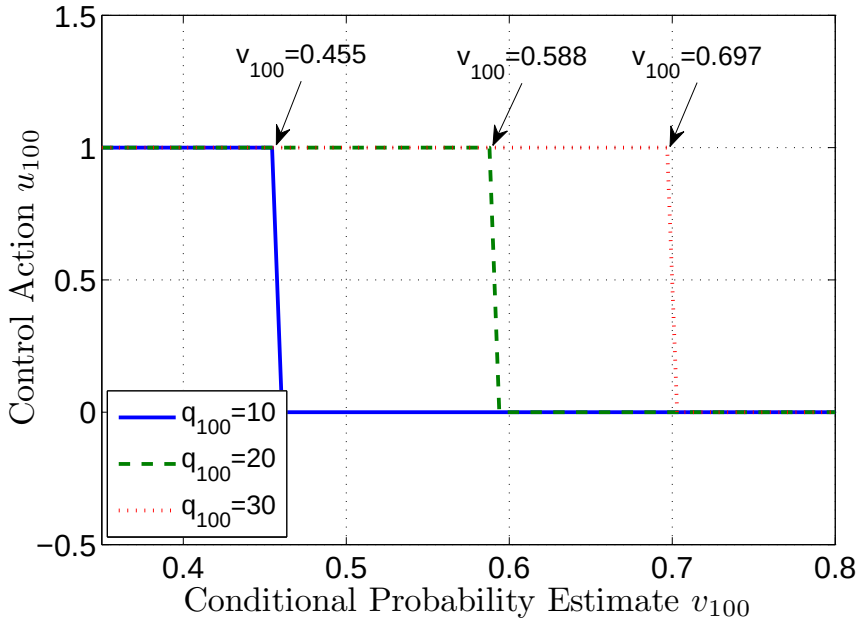



Fig. 3.3: Comparison of the thresholds for the control action $u_k$ with respect to conditional probability estimate $q_k$ for different values of $x_k$.

Markov channel by defining $v_k = p(Z_k = 0|z_1^{k-1}) = w_k(1)(\epsilon_1 - \epsilon_2) + \epsilon_2$, the probability of packet loss in the $k$th slot, given the previous ACK/NAK sequence. The recursive relation for $v_k$ is given by,

$$\begin{aligned} v_{k+1} &= \Psi(v_k, z_k, u_k) \\ &= \begin{cases} \epsilon_1 + p_{12}(\epsilon_2 - \epsilon_1) + (1 - p_{12} - p_{21})(v_k - \epsilon_1)\frac{(1-\epsilon_2)}{(1-v_k)}, & z_k = 0, u_k = 1 \\ \epsilon_1 + p_{12}(\epsilon_2 - \epsilon_1) + (1 - p_{12} - p_{21})(v_k - \epsilon_1)\frac{\epsilon_2}{v_k}, & z_k = 1, u_k = 1 \\ \epsilon_1 + p_{12}(\epsilon_2 - \epsilon_1) + (1 - p_{12} - p_{21})(v_k - \epsilon_1), & u_k = 0 \end{cases} . \end{aligned} \tag{3.17}$$

The control action at time $k$ is a function of three variables: $k$, $q_k$ and $v_k$. On numerically solving the DP equations, it is observed that for a fixed $k$ and $q_k$, the decision rule $u_k(q_k, v_k)$ is a threshold rule with respect to $v_k$. The intuition for this is that, since $v_k$ is the estimated probability of losing a packet, the controller would decide to transmit if it is below a certain value and not to transmit if it is above

a certain value. We call this threshold $\mathcal{Q}_k(q_k)$, which is a function of $k$ and $q_k$ for a particular set of channel parameters. When the calculated $v_k < \mathcal{Q}_k(q_k)$ then the controller transmits a packet ($u_k = 1$), and defers transmission otherwise ($u_k = 0$). We illustrate this property of the decision rule in Figure 3.3. In this case, $B = 40$ packets need to be transmitted in $T = 200$ time slots. The channel parameters are: $p_{21} = 0.1, p_{12} = 0.2, \epsilon_1 = 0.2$ and $\epsilon_2 = 0.8$. Here, the control $u_k$ is plotted as a function of $v_k$ for different values of $q_k$ and $k = 100$. When $q_{100} = 10$, then $u_{100} = 1$ for $v_{100} \leq 0.455$. Increasing $q_{100}$ to 20, we have $\mathcal{Q}_{100}(20) = 0.588$ and for $q_{100} = 30$, it increases to 0.697. Clearly $\mathcal{Q}_k(q_k)$ is monotonic with $q_k$ for a fixed $k$. Note that $\mathcal{Q}_k(q_k)$ is also monotonic with $k$ for a given $q_k$, since there are more packets in the queue waiting to be transmitted by a fixed deadline, the "anxiety" of the controller to transmit them increases.

The DP solution described above provides an energy optimal transmission scheduler for sensor networks. However, the computation required to solve the DP equations is beyond the capability of most sensor network transmitters. Instead, we make use of lookup tables to implement the scheduler. This approach requires storage resources rather than computation resources. A lookup table $\mathrm{T}_{\mathcal{Q}}(p_{21}, p_{12}, \epsilon_1, \epsilon_2, k, q_k)$ which has been pre-computed and loaded in the sensor memory is used to determine the control rule. This table contains the value of the threshold $\mathcal{Q}$ for different channel conditions and queue conditions. The controller updates the probability estimate $v_k$, compares it with the appropriate threshold $\mathcal{Q}_k(q_k)$ and returns the control variable $u_k \in \{0, 1\}$. Since it is not practical to have such a table for all channel realizations, we store tables for a discrete set of channel realizations, i.e., $\{p_{21}, p_{12}, \epsilon_1, \epsilon_2\}$.

The major design parameters on the transmitter/controller in this approach is the amount of memory to store the look-up table $\mathrm{T}_{\mathcal{Q}}(p_{21}, p_{12}, \epsilon_1, \epsilon_2, k, q_k)$. The size of memory will depend on the number of channel realizations ($\{p_{21}, p_{12}, \epsilon_1, \epsilon_2\}$) stored in the look-up table. For instance, a user can store 5 values each of $p_{21}$, $p_{12}$, $\epsilon_1$ and $\epsilon_2$ to get 625 different realizations of the channel. In addition, for each channel realization, each value of $q_k$ and $k$ will have a threshold value $\mathcal{Q}_k(q_k)$ associated with it. If the DP algorithm provides a solution for up to 200 data packets transmitted over a duration of up to 500 time slots, the memory requirement for one channel realization will be 100 Kbytes (since each threshold value will need only one byte of storage). In this case, the total memory requirement for the controller will be 62.5 Mbytes. The computational requirement of this approach is minimal, since it only involves the evaluation of a simple algebraic expression given by Eq. (3.17) at every time step.

Table 3.1 shows the pseudocode for the proposed scheduler. The channel parameters are required for initializing the scheduler and we assume that they are already provided to the controller. A lookup table $\mathrm{T}_{\mathcal{Q}}(p_{21}, p_{12}, \epsilon_1, \epsilon_2, k, q_k)$ which has been pre-computed and loaded in the sensor memory is used to determine the control rule. This table contains the value of the threshold $\mathcal{Q}$ for different channel conditions and queue conditions. The controller updates the probability estimate $v_k$ according to Eq. (3.17), compares it with the appropriate threshold $\mathcal{Q}_k(q_k)$ and returns the control variable $u_k \in \{0, 1\}$. In this case, $u_k = 1$ when the probability estimate is lower than the threshold and vice versa.

For FSMCs with $|\mathcal{S}| > 2$, the look-up table $\mathrm{T}_{\mathcal{U}}(\mathrm{P}, \epsilon_1, \epsilon_2, \ldots, \epsilon_{|\mathcal{S}|}, k, q_k, \boldsymbol{w_k})$ will be slightly different in structure. Here the table contains the control action $u_k$

**Input:** $p_{21}$, $p_{12}$, $\epsilon_1$ and $\epsilon_2$ from Initialization/Calibration phase.

**Output:** Control $u_k \in \{0, 1\}$.

**Given:** A six-dimensional lookup table, $\mathrm{T}_{\mathcal{Q}}(p_{21}, p_{12}, \epsilon_1, \epsilon_2, k, q_k)$ containing values of threshold $\mathcal{Q}$ for various channel conditions (characterized by $\{p_{21}, p_{12}, \epsilon_1, \epsilon_2\}$) and queue conditions (characterized by $k$ and $x_k$).

**The DP Scheduler**

**if** Transmission in $(k-1)$th slot **then**
  $z_{k-1} \leftarrow$ ACK/NAK received for the $(k-1)$th packet
  $v_k \leftarrow \Psi(v_{k-1}, z_{k-1}, 1)$
**else**
  $v_k \leftarrow \Psi(v_{k-1}, z_{k-1}, 0)$
**end if**
$q_k \leftarrow$ From the Queue
**if** $v_k > \mathrm{T}_{\mathcal{Q}}(p_{21}, p_{12}, \epsilon_1, \epsilon_2, k, q_k)$ **then**
  **return** $u_k \leftarrow 0$
**else**
  **return** $u_k \leftarrow 1$
**end if**

Table 3.1: Pseudocode for the proposed controller.

for different channel condition vectors $\boldsymbol{w_k}$. The controller updates $\boldsymbol{w_k}$ according to Eq. (3.10) and looks up the control action $u_k$, according to this vector. As previously, it is not practical to have such a table for all channel realizations, we store tables for a discrete set of channel realizations, $\mathcal{A}$. Consequently the look-up table $\mathrm{T}_{\mathcal{U}}(\mathrm{P}, \epsilon_1, \epsilon_2, \ldots, \epsilon_{|\mathcal{S}|}, k, q_k, \boldsymbol{w_k})$ will have values for the channel realizations $\{\mathrm{P}, \epsilon_1, \epsilon_2, \ldots, \epsilon_{|\mathcal{S}|}\} \in \mathcal{A}$. The size of memory will depend on the number of channel realizations $|\mathcal{A}|$ stored in the look-up table. In addition, each value of $q_k$ and $k$ will have a set of $\boldsymbol{w_k}$ associated with it. Let each component of $\boldsymbol{w_k}$ be quantized into 10 levels. If the DP algorithm provides a solution for up to 100 data packets transmitted over a duration of up to 500 time units, the memory requirement will be $|\mathcal{A}| \times 50 \times 10^{|\mathcal{S}|-1}$ Kbits (since each the control action will need only one bit of storage). For $|\mathcal{A}| = 100$ and $|\mathcal{S}| = 3$, the memory requirement for the controller will be 62.5 Mbytes.

Before concluding this section, we would like to point out that there is an inherent tradeoff between accuracy and complexity with increasing number of states. The memory requirements for storing the controller action will increase exponentially with the number of FSMC states. On the other hand, the channel model will become more "accurate" as more channel states are taken into account. However, there is no universal rule for selecting the number of states in the FSMC model for all possible physical fading processes. We direct the reader to [20] and references therein for a discussion on the selection of number of FSMC states.

## 3.5 Performance Evaluation

We conduct simulations on FSMCs to compare the various transmission schedulers and compare them against their respective theoretical bounds. The metric used to

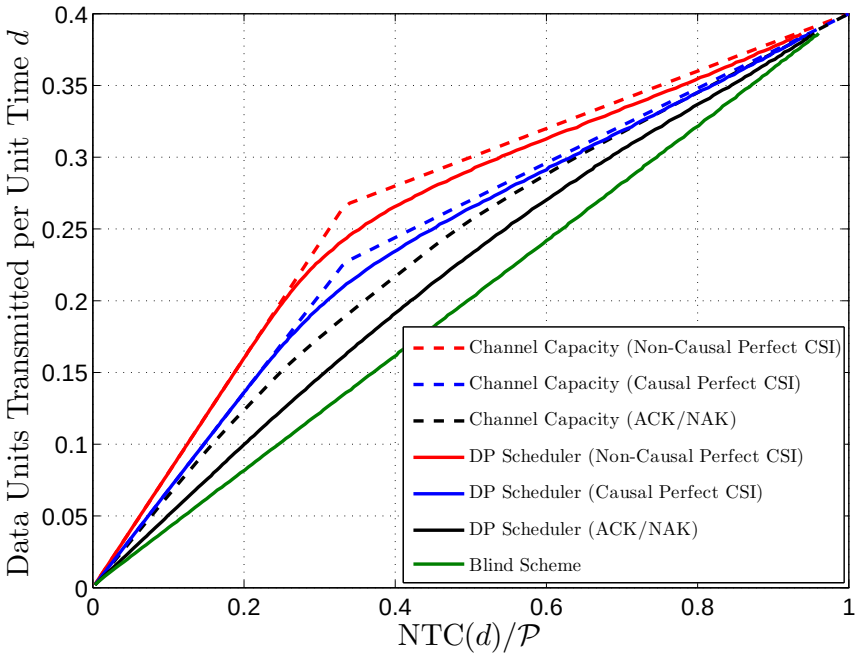



Fig. 3.4: Comparison of the energy efficiency performance of various schemes with their theoretical bounds for 2-state Markov channel. Here $\epsilon_1 = 0.2$, $\epsilon_2 = 0.8$, $p_{21} = 0.1$ and $p_{12} = 0.2$.

compare the scheduler performance is termed *normalized transmission cost* (NTC). We define $d$ as the number of data packets successfully transmitted in one time slot and NTC($d$) is the number of transmission attempts required to achieve this. For the theoretical bounds, the capacity-cost function $C(\mathcal{P})$, given by Theorem 2.1 and Corollary 2.2, is plotted as a function of the power constraint $\mathcal{P}$. Here $\mathcal{P}$ is equivalent to the number of packet transmission attempts made in a time slot. The capacity-cost function $C(\mathcal{P})$ is divided by the packet length $N$ to represent the capacity in terms of data packets and offer a direct comparison to the NTC.

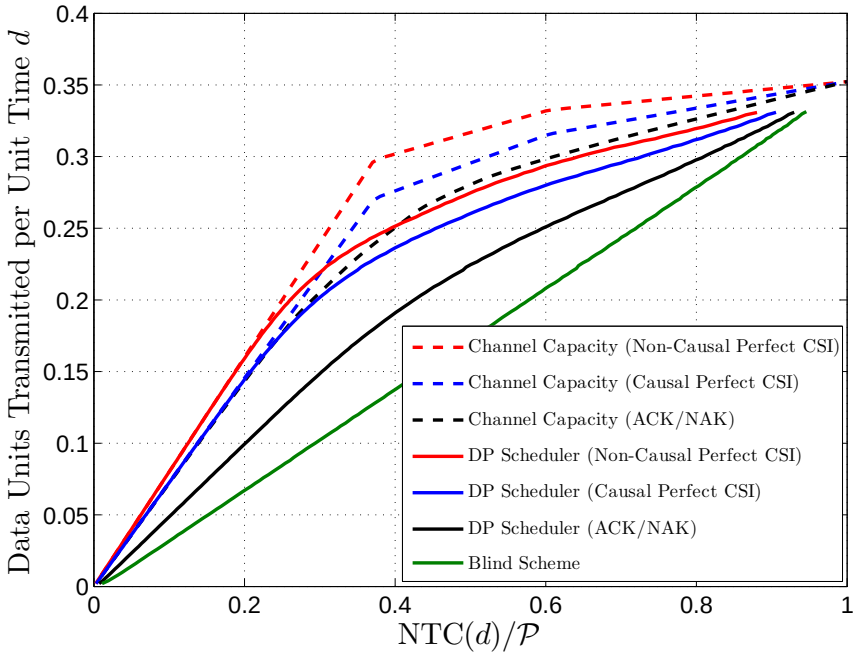



Fig. 3.5: Comparison of the energy efficiency performance of various schemes with their theoretical bounds for 3-state Markov channel. Here $p_{12} = 0.025$, $p_{13} = 0.075$, $p_{21} = 0.075$, $p_{23} = 0.05$, $p_{31} = 0.05$ and $p_{32} = 0.05$. The loss probabilities are $\epsilon_1 = 0.2$, $\epsilon_2 = 0.85$ and $\epsilon_3 = 0.95$.

### 3.5.1 Comparison of DP-Schedulers for FSMCs

Figures 3.4 and 3.5 show the performance comparison of various DP-schedulers for 2- and 3-state Markov channels. The transition probabilities for the 2-state Markov channel (2SMC) are $p_{21} = 0.1$ and $p_{12} = 0.2$. The loss probability in each state is $\epsilon_1 = 0.2$ and $\epsilon_2 = 0.8$. For the 3-state Markov channel (3SMC), the transition probabilities are $p_{12} = 0.025$, $p_{13} = 0.075$, $p_{21} = 0.075$, $p_{23} = 0.05$, $p_{31} = 0.05$ and $p_{32} = 0.05$. The loss probabilities are $\epsilon_1 = 0.2$, $\epsilon_2 = 0.85$ and $\epsilon_3 = 0.95$. The time horizon $T$ for the simulations is 500 time slots and $10^5$ realizations of the channel are simulated. The packet error process is simulated by first generating a Markov process 500 time slots long and then setting the loss variable $z_k$ according to the given state $s_k$.

First, we plot the performance of DP-scheduler (solid red trace) and the corresponding bound (dashed red trace) for the non-causal perfect CSI case. For both cases (2SMC and 3SMC), the bound is tight up to the point when the deadline constraint requires a throughput of 0.2 packets/time slot. This can be due to the fact that for a lightly loaded system, the effect of queue state on decision making is minimal, and the scheduler considers CSI for decision making leading to a performance close to the theoretic bound. In all performance curves with feedback, for a 2SMC, a knee-point is observed when the NTC (or $\mathcal{P}$) is equal to 0.33 (for 3SMC, knee points at NTC = 0.37 and NTC = 0.6). As the number of packets to be transmitted is increased, the cost will increase at a certain rate since only the good states are being used initially. However after all the good states are exhausted, the scheduler will use the bad states for transmission, which increases cost at a faster rate as more packets will be dropped. For a throughput requirement of 0.25 packets/time slot in

a 2SMC, causal perfect CSI scheduler requires 25% more transmission attempts than a non-causal perfect CSI scheduler (10% for a 3SMC).

We observe that the performance of the causal imperfect CSI scheduler or the ACK/NAK scheduler (solid black traces) is within 80% of the theoretical bound (dashed black traces) for the 2SMC (75% for the 3SMC). The bound is not very tight as the capacity calculations assume knowledge of the complete ACK/NAK sequence, whereas the ACK/NAK scheduler has access to the incomplete sequence (since ACK/NAK is unavailable for the time slots when no transmission is attempted). In both the cases, the ACK/NAK scheduler, at worst, requires 50% more transmission attempts than the non-causal perfect CSI based scheduler for achieving a throughput of 0.25 packets/time slot. The green trace represents a blind transmission scheme. In this case, the scheduler does not have any channel feedback, instead it just keeps transmitting till the queue is emptied out. This can be the lower bound on performance of any feedback scheme. The performance of this scheduler has a constant slope, since the scheduler does not adapt the transmission policy to the channel state. For a 2SMC, ACK/NAK scheduler outperforms the blind transmission scheme by up to 25% for 0.1 packets/time slot throughput (50% for a 3SMC) and 15% for 0.25 packets/time slot throughput (20% for a 3SMC).

### 3.5.2 Look-up Table Size

Figure 3.6 compares the effect of look-up table size on the performance of the ACK/NAK scheduler. The parameters for the 2SMC are: $p_{12} = 0.18$, $p_{21} = 0.09$, $\epsilon_1 = 0.23$ and $\epsilon_2 = 0.78$. The 32 MB table contains control action for $|\mathcal{A}| = 256$ channel realizations and 20 levels of $\boldsymbol{w_k}$. The parameters for the 1 MB table are $|\mathcal{A}| = 16$ and

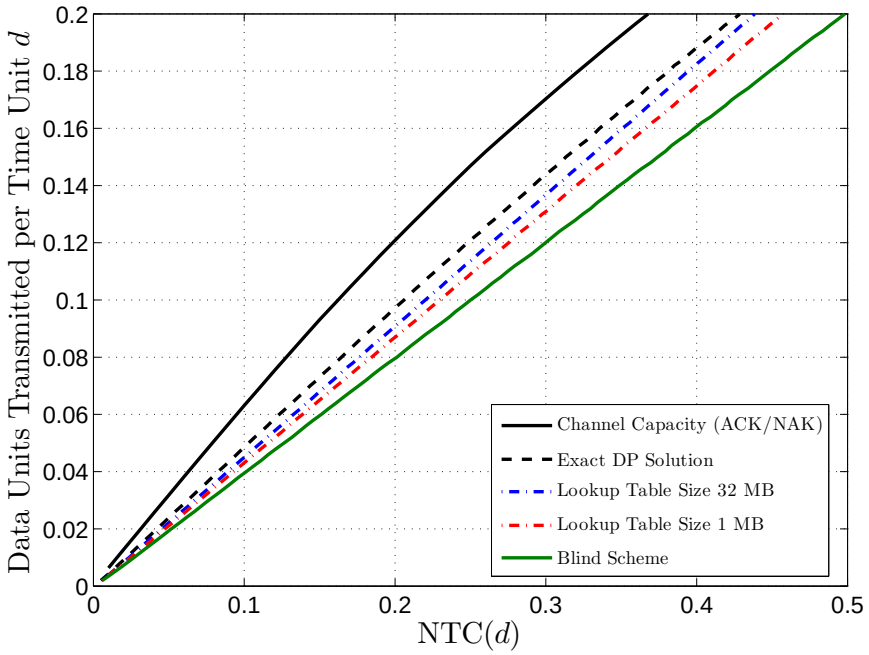



Fig. 3.6: Comparison of the performance of ACK/NAK schedulers with different table sizes. Here $\epsilon_1 = 0.23$, $\epsilon_2 = 0.78$, $p_{21} = 0.09$ and $p_{12} = 0.18$.

10 levels of $\boldsymbol{w_k}$. Both tables store control action for up to 100 packets transmitted over a duration of up to 500 time slots. We observe that the ACK/NAK schedulers using a look-up table perform close to the exact solution. For instance, the 32 MB table requires 7% extra transmission attempts (11.5% for 1 MB table) to achieve throughput of 0.1 packets/time slot. Note that we have used uniform quantization for this example, however quantization methods that perform better might also exist.

### 3.5.3 Time Horizon

Figure 3.7 compares the performance of the ACK/NAK scheduler with different time horizons $T$. The parameters for the 2SMC are: $p_{12} = 0.2$, $p_{21} = 0.1$, $\epsilon_1 = 0.1$ and $\epsilon_2 = 0.9$. We observe that with increasing $T$, the performance of the ACK/NAK

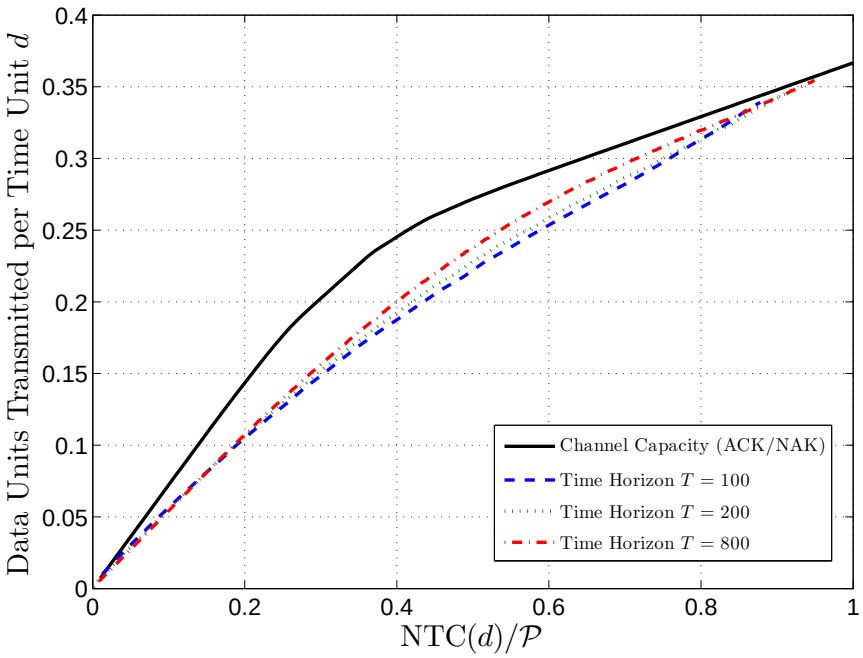



Fig. 3.7: Comparison of the performance of ACK/NAK schedulers with different time horizons $T$. Here $\epsilon_1 = 0.1$, $\epsilon_2 = 0.9$, $p_{21} = 0.1$ and $p_{12} = 0.2$.

scheduler improves. A scheduler with $T = 100$ requires, at most, 10% more transmission attempts than a scheduler with $T = 800$. The improvement in performance can be due to the scheduler having greater flexibility in scheduling transmissions with larger $T$. We also observe that the performance of the ACK/NAK schedulers (for all $T$) is within 75% of the theoretical bound.

### 3.5.4 Terminal Cost

In Figure 3.8, we compare the effect of terminal cost $C$ on the performance of the ACK/NAK scheduler for a 2SMC. The channel parameters are: $p_{12} = 0.2$, $p_{21} = 0.1$, $\epsilon_1 = 0.2$ and $\epsilon_2 = 0.8$. For $C = \{10, 50, 200\}$, we observe that there is not much of a difference in the performance of the ACK/NAK schedulers formulated using

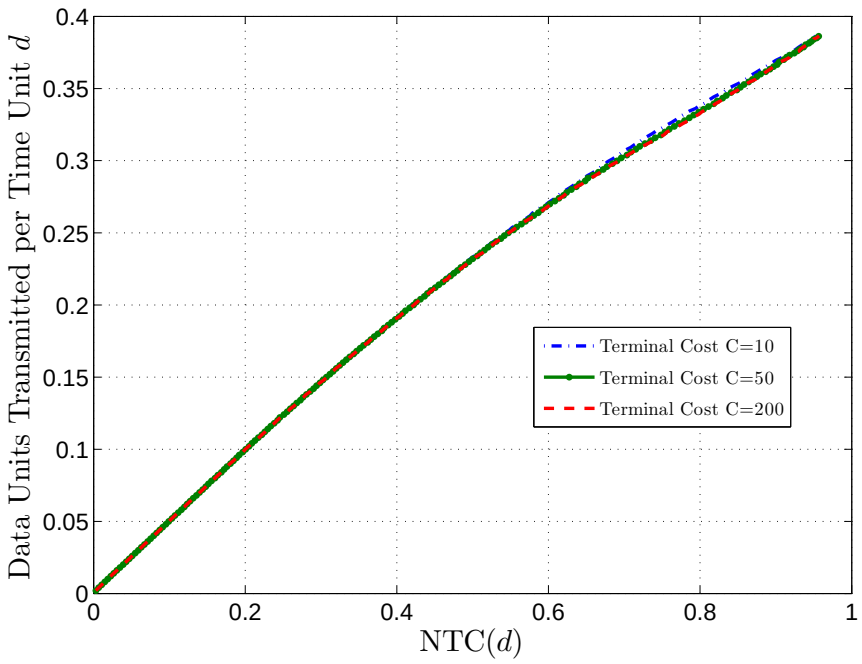



Fig. 3.8: Comparison of the performance of ACK/NAK schedulers formulated with different terminal costs $C$. Here $\epsilon_1 = 0.2$, $\epsilon_2 = 0.8$, $p_{21} = 0.1$ and $p_{12} = 0.2$.

these terminal costs. Other channel realizations also exhibit a similar behavior. This leads to the observation that the choice of $C$ does not impact the performance of the scheduler, as long as $C > 10$.

### 3.5.5 Actual Power Penalty

Next, we find the actual power penalty incurred by an ACK/NAK scheduler, $L_{\text{dB}} \triangleq 10 \log_{10} \left( \frac{\text{attempts made by ACK-NAK scheduler}}{\text{attempts made by causal CSI scheduler}} \right)$, for a given throughput. We define a similar quantity for the blind transmission scheme. Figure 3.9 shows the power penalty incurred by the ACK/NAK based and blind schedulers for 2SMCs as a function of channel memory $\mu$. We plot this quantity for different values of packet loss probabilities, $\{\epsilon_1, \epsilon_2\}$ keeping the steady state probabilities $\{\pi(1), \pi(2)\}$ constant.

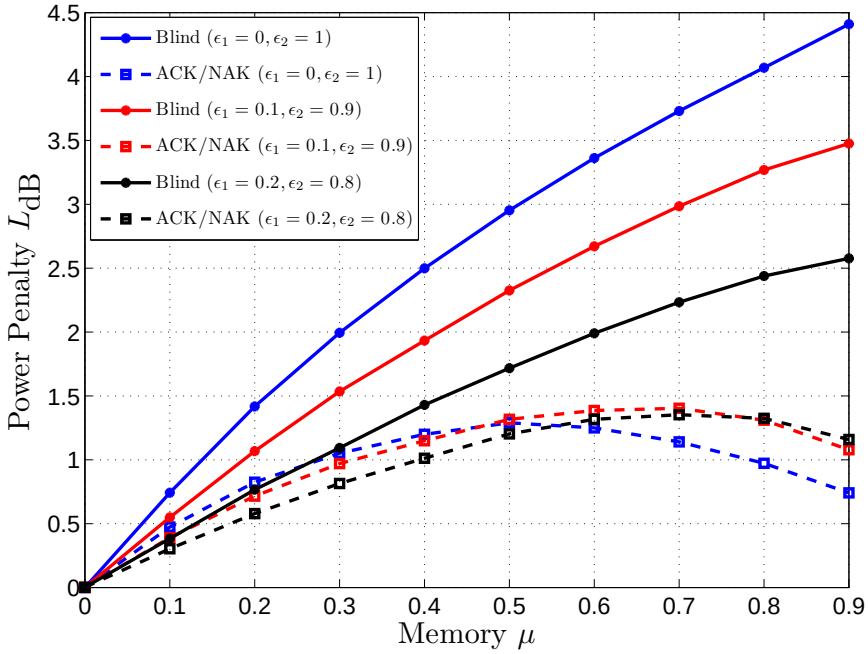



Fig. 3.9: The power penalty $L_{\mathrm{dB}}$ as a function of the channel memory $\mu$ for different values of packet loss probabilities $\{\epsilon_1, \epsilon_2\}$ with $\pi(1) = 1/3$ and $\pi(2) = 2/3$. The scheduler attempts to transmit 100 packets over 500 time slots for all cases.

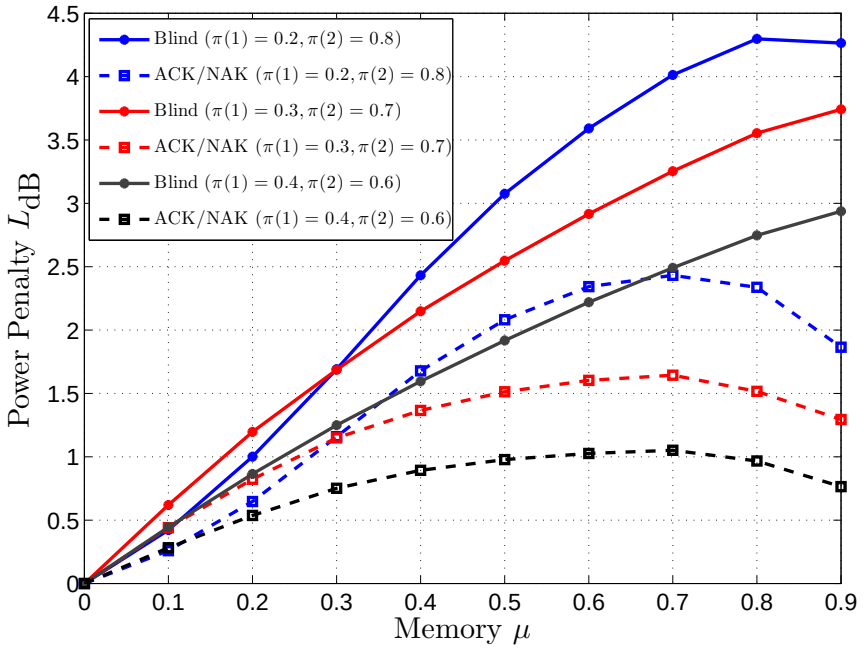



Fig. 3.10: The power penalty $L_{\text{dB}}$ as a function of the channel memory $\mu$ for different values of steady state probabilities $\{\pi(1), \pi(2)\}$ with $\epsilon_1 = 0.1$ and $\epsilon_2 = 0.9$. The scheduler attempts to transmit 50 packets over 500 time slots for all cases.

For the range of channel parameters considered in our simulations and a throughput requirement of 0.2 packets/time slot, ACK/NAK scheduler does not incur a power penalty of more than 1.5 dB. On the other hand, a blind transmission scheme takes a power penalty of up to 4.5 dB (for $\epsilon_1 = 0$ and $\epsilon_2 = 1$).

In Figure 3.10, we consider different steady state probabilities $\{\pi(1), \pi(2)\}$, while keeping the loss probabilities constant ($\epsilon_1 = 0.1, \epsilon_2 = 0.9$). The throughput requirement is set at 0.1 packets/time slot. In the worst case, the ACK/NAK scheduler incurs a power penalty of 2.5 dB for $\pi_1 = 0.2$ and $\pi_2 = 0.8$ whereas the blind scheme incurs a power penalty of almost 4.5 dB. For $\mu = 0.9$, in all the cases, the performance

of the ACK/NAK scheduler is 2 dB better than the blind scheduler. Since $\mu$ is an indicator of the burstiness (as bursty channels have high values of $\mu$), the improvement provided by our scheme will be more pronounced in bursty channels. This confirms the trends we saw in Section 2.4.2 with the performance bounds.

# CHAPTER 4

# A PRACTICAL IMPLEMENTATION OF AN ENERGY MANAGEMENT SCHEME

## 4.1 Introduction

Sensor networking applications often have stringent requirements for QoS parameters such as throughput and delay. In battery operated sensor networks, such QoS requirements must be met while consuming the least amount of energy. Since a high percentage of the energy is spent on data communication, support for efficient and reliable communication is critical. However, high variability in channel quality caused by factors such as fading, mobility, and time-varying multiuser interference make it difficult to achieve those objectives. Indeed, Woo et. al., [34], and Zhao and Govindan [35] have both observed a significant variability in link quality in wireless sensor networks with radios in the 433 MHz band. The former paper points out that the instantaneous packet error probability varies by approximately 30% around its mean. The latter paper, as well as Willig et. al., [36], both show that the packet-error stochastic process exhibits significant long-term dependence.

The existing CSMA protocol is an example of an adaptive transmission technique. The CSMA protocol uses carrier sensing to avoid collisions and backoffs to address

the problem of contention among nearby nodes. However, packet transmissions may fail due to cumulative interference from other nodes in the network. Indeed in our testbed experiments with Mica2 nodes, we have observed that with interfering sources that are sufficiently far away, 69% of the packets for which the CSMA granted a transmission permit are lost. From this example, we can conclude that the combined effect of a large number of interfering sources can be very detrimental and the CSMA based protocols - designed to suppress collisions - are not effective in avoiding such losses. Immediate solution to this problem is reducing the carrier sense threshold that triggers a backoff and consequently increasing the carrier sense range. This, in effect, would enable a node to sense this combined interference and hidden terminals to some extent, and avoid part of the losses. However, the increase of carrier sense range makes a node overly conservative with respect to interference and it leads to a lower effective throughput. Therefore, simple adjustment of the carrier sense range is not sufficient to avoid transmissions during poor channel conditions.

Our objective in this chapter is to optimize the energy consumption with constraint on the throughput. Our solution methodology is also different as it explicitly models the channel based on rigorous estimation techniques. Existing MAC layer solutions for sensor networks such as [37, 38, 39, 40], have not considered direct modeling of the time-varying channel quality for optimization of transmission attempts. MAC layer protocols such as [22, 23] model the bursty nature of wireless channel as a Markov model. However these techniques do not directly incorporate channel estimation into the transmission scheduler. Several back-pressure based mechanisms have been proposed at the link layer [41, 42, 43, 44] to address high interference and network

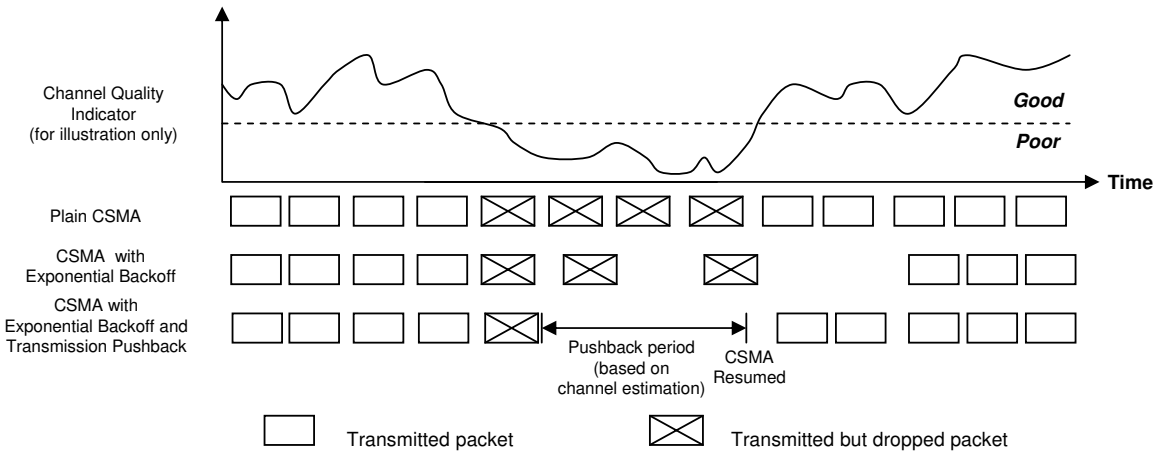



Fig. 4.1: Transmission Pushback: CSMA unnecessarily transmits during poor channel conditions. With exponential backoff, there will be fewer transmissions during poor channel conditions, however opportunities to transmit during good states will also be lost. With transmission pushback CSMA can avoid periods with poor channel quality, while making use of good channel states.

congestion. These approaches are orthogonal to our proposed solution, and can be used in conjunction with our solution to improve performance, as shown in [45].

In this chapter, we systematically study the problem of addressing packet losses due to cumulative interference, and propose a binary control technique over CSMA. Our approach is based on exploiting the temporal correlations of the interference process. We introduce a new concept called *transmission pushbacks*, which refers to an appropriately computed delay introduced at the MAC layer in order to avoid periods with bad-channel quality while considering a node's throughput requirement. Therefore, we reduce the number of transmissions per packet as well as the number of transmission attempts per unit time. In case of bursty losses, avoiding the bad channel state may also lead to a higher throughput (visible at higher layers) despite lower number of transmission attempts.

| Mechanisms | Designed for collision losses | Designed for interference and channel losses |
|---|---|---|
| **CSMA Plain** | Partially | No |
| **CSMA/EB** | Yes | Partially |
| **CSMA/EB+Pushback** | Yes | Yes |

Table 4.1: Types of losses addressed by various mechanisms

The main idea of transmission pushbacks is to defer packet transmission attempts for an appropriately selected period upon failed packet transmissions. Figure 4.1 illustrates the benefits of using transmission pushbacks in comparison with CSMA based approaches in the presence of time-varying channels. Plain CSMA leads to failed transmissions, and thus wasted energy, during periods with poor channel quality. CSMA with exponential backoff may reduce such failed transmissions, but it also cuts down the transmission attempts, even at times of improved channel quality. Our proposed transmission pushback mechanism predicts the duration for which the channel quality will remain poor. Thus, unnecessary transmissions can be avoided to conserve energy and the good channel states are taken advantage of. The contrast is also highlighted in Table 4.1 which shows that Pushback along with CSMA with exponential backoff can handle various types of packet losses. Observe that as pushback operates over the CSMA algorithm, improvements of the CSMA mechanism such as [46] are orthogonal to pushback.

To determine the pushback time, we need to estimate the channel quality and how it varies over time. We use an adaptive channel prediction technique, based on estimating the parameters of a simple hidden Markov model (HMM), which represents our channel. The parameters of the HMM are dynamically updated based

solely on the binary ACK sequence (transmission success or failure) for the previous packet transmissions. We choose the appropriate pushback period by considering the throughput requirement measured by the incoming data rate, and the predicted quality of the channel. Such an adjustment in rate, based on the throughput requirement is also seen in lazy packet schedulers (e.g., [25]). Designing an "optimal" sleep-wake scheduling solution that integrates with pushback is beyond the scope of this thesis. The main complexity of that task is caused by the missed receive opportunities: during pushback, if a node goes into sleep mode, then it will miss opportunities to receive packets from other nodes. We assume that nodes are awake at all times for potential received packets. However, nodes do not spend extra energy in sensing the channel since the calculation of the deferral time depends solely on the ACK/NACK packets. The proposed approach is simple to implement over existing CSMA based MAC solutions, as well as queue and congestion control algorithms. Therefore it is highly suited for existing sensor network platforms.

The rest of the chapter is organized as follows. Section 4.2 presents our approach to model the channel losses. Section 4.3 presents a description of our pushback algorithm.

## 4.2 Channel Model

In this section we describe our channel loss model and its parameters and give an experimental justification for our model. As we have done previously in Chapters 2 and 3, we use a Markov channel to model the packet level error process. However, in this case, we assume that the underlying transition probabilities are not known to the transmitter initially. We will call our channel a hidden Markov model (HMM)

channel. We use this channel loss model to derive the theoretical expressions for Packet Success Ratio (PSR) and throughput as functions of channel and system parameters. Our algorithm will describe how to estimate these parameters based on the available measurements at the sensor nodes in the next section.

### 4.2.1 The Hidden Markov Model (HMM) Channel

Let $A_n,\ n \geqslant 1$ be the process, which takes on an 'S' or an 'F' depending on whether a packet transmitted at time $n$ is successfully decoded at the receiver or not. We assume that only one packet can be transmitted in each time slot and that $A_n$ is a *wide-sense Markov* of order 1 for the packet error process. We will discuss the validity of this assumption later, but first we describe $A_n$ further.

A sequence $A_n, n \geq 1$ is said to be wide-sense Markov if the probability of a future value, $A_{n+m}$, is completely determined by its most recent value, i.e. $A_n$ and the time difference ($m$) between the two events. The autocovariance function for such a process is exponential. More specifically, suppose the process is in steady state[3] at time $n$ and let $\mathrm{P}\left(A_n = \mathrm{F}\right) = p$. Then the autocovariance function of $A_n$ is $K_A(m) = p(1-p)\alpha^{|m|}$ for some $\alpha$, $|\alpha| < 1$. Here $p$ is the long-term loss probability and $\alpha$ quantifies the "coherence" or correlation between a successful (failed) transmission in the future based on the present observation. The loss probability $p$ increases with the number of interfering users and the noise in the channel and $\alpha$ is a measure for the burst length for the failed transmissions. Next, we derive the conditional probabilities of a wide-sense Markov process $A_n, n \geq 1$.

[3]We assume that the process started at time $n = -\infty$. In practice the data transmission is only finite, however we make this technical assumption, since we do not have any information about the initial channel state.

**Lemma 4.1.** *For the wide-sense Markov process $A_n, n \geq 1$, the probability of a successful (failed) transmission in a future time slot, conditioned on a successful (failed) transmission in the present slot, is unique:*

$$P(A_{n+m} = F|A_n = S) = p(1 - \alpha^m) \tag{4.1}$$

$$P(A_{n+m} = S|A_n = S) = 1 - p(1 - \alpha^m) \tag{4.2}$$

$$P(A_{n+m} = F|A_n = F) = p + (1 - p)\alpha^m \tag{4.3}$$

$$P(A_{n+m} = S|A_n = F) = 1 - p - (1 - p)\alpha^m. \tag{4.4}$$

*Proof.* A wide-sense Markov process has an exponential autocovariance function. In our analysis, we assume that the autocovariance function is of the form $K_A(m) = p(1-p)\alpha^{|m|}$ and the unconditioned probability of failure ($\mathrm{P}(A_n = \mathrm{F})$) and success ($\mathrm{P}(A_n = \mathrm{S})$) are $p$ and $1 - p$ respectively.

First, we derive the probabilities conditioned on failure, i.e., Eqs. (4.3) and (4.4). To facilitate the evaluation of the expectation, we code the event of Failure (F) as '1' and the event of Success (S) as '0'. Covariance is defined as,

$$\begin{aligned} K_A(m) &= \mathrm{E}[A_{n+m}A_n] - \mathrm{E}[A_{n+m}]\mathrm{E}[A_n] \\ &= \sum_{x\in\{0,1\}} \sum_{y\in\{0,1\}} xy\mathrm{P}(A_{n+m} = x|A_n = y)\,\mathrm{P}(A_n = y) \\ &\quad - \sum_{x\in\{0,1\}} x\mathrm{P}(A_{n+m} = x) \sum_{y\in\{0,1\}} y\mathrm{P}(A_n = y) \\ &= \mathrm{P}(A_{n+m} = 1|A_n = 1)\,p - p^2. \end{aligned} \tag{4.5}$$

For $m \geq 0$, we can compare Eq. (4.5) to the autocovariance function for a wide-sense Markov process,

$$K_A(m) = p(1-p)\alpha^m = \mathrm{P}(A_{n+m} = 1|A_n = 1)\,p - p^2,$$

to get Eq. (4.3) or $\mathrm{P}\left(A_{n+m}=1|A_n=1\right)=p+(1-p)\alpha^m$. Using the fact, $\mathrm{P}\left(A_{n+m}=0|A_n=1\right)=1-\mathrm{P}\left(A_{n+m}=1|A_n=1\right)$, we get Eq. (4.4) or $\mathrm{P}\left(A_{n+m}=0|A_n=1\right)=1-p-(1-p)\alpha^m$.

To derive the probabilities conditioned on success, i.e., Eqs. (4.1) and (4.2), we follow the same steps with minor modifications. In this case, we code the event of Success (S) as '1' and the event of Failure (F) as '0'. From the definition of Covariance, we get,

$$\begin{aligned} K_A(m) &= p(1-p)\alpha^m \\ &= \mathrm{P}\left(A_{n+m}=1|A_n=1\right)(1-p)-(1-p)^2. \end{aligned} \tag{4.6}$$

On simplification of Eq. (4.6), we get Eq. (4.2) or $\mathrm{P}\left(A_{n+m}=1|A_n=1\right)=1-p(1-\alpha^m)$. Finally, $\mathrm{P}\left(A_{n+m}=0|A_n=1\right)=1-\mathrm{P}\left(A_{n+m}=1|A_n=1\right)$ gives Eq. (4.1) or $\mathrm{P}\left(A_{n+m}=0|A_n=1\right)=p(1-\alpha^m)$. □

Hence only a pair of parameters is sufficient to represent the channel with the first order Markov assumption. For the purpose of illustration, the conditional probabilities of failure for $p=0.6$ and $\alpha=0.8$ are plotted in Fig. 4.2. The "deferred time slots" represents the number, $m$, of time slots waited before the next transmission attempt after an event S or F.

We can form an underlying Markov chain corresponding to this wide-sense Markov process as follows. The Markov chain has two states S (Good) and F (Bad) and the chain makes a transition between these states depending on $A_n$ taking on an S or an F as shown in Figure 4.3. One can evaluate the parameters $p$ and $\alpha$ for the associated Markov process given the transition probabilities, $x$ and $y$ for this Markov chain and vice versa as shown in Section 4.2.2. The channel model is an HMM, since $x$ and $y$ are unknown initially and we estimate them based on the observed values of $A_n$

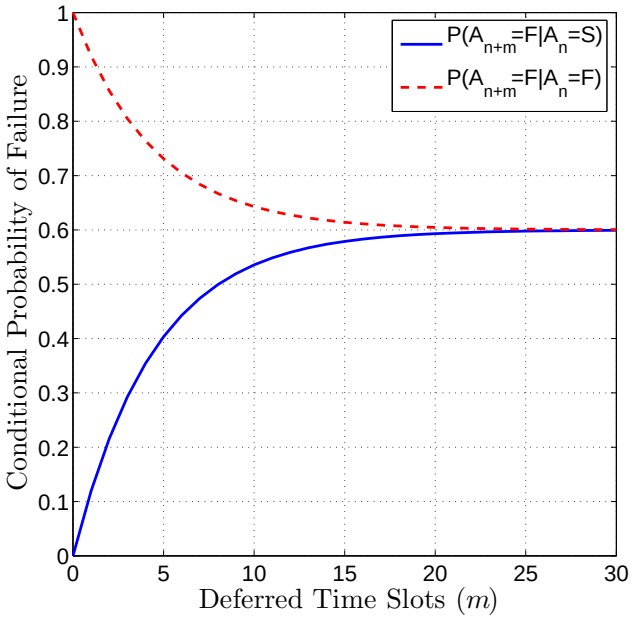



Fig. 4.2: Conditional probability of failure as a function of deferred time slots setting $p = 0.6$ and $\alpha = 0.8$.

using a maximum likelihood estimator. Based on these estimated values, we calculate the two channel parameters $\alpha$ and $p$, which is detailed in Section 4.3. Note that the reason for estimating $x$ and $y$ initially is due to the simplicity of the task (based on observation of $A_n$). The reason why we pursue our analyses with $\alpha$ and $p$ afterwards, instead of $x$ and $y$, is that the performance metrics of the pushback algorithm are tied to $\alpha$ and $p$ more naturally.

From the two curves in Fig. 4.2, one can see the reasoning behind choosing a pushback duration conditional on an event F only. If we schedule the next packet for immediate transmission (i.e., $m = 1$) after an S, we have the best chance of observing another S. Intuitively, we are taking advantage of the good channel state. On the other hand, if we defer scheduling the transmission (i.e., $m > 1$) of the

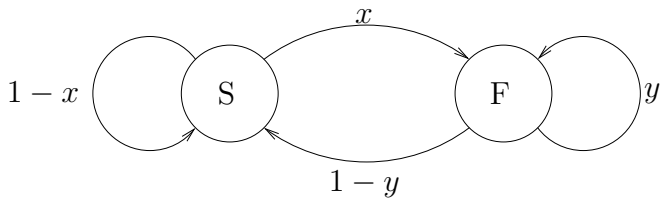



Fig. 4.3: Markov chain representation of the channel.

next packet after an F, we lower the probability of failing in that transmission. The longer the deferral time, the higher the probability of an S in the next transmission. However, waiting indefinitely can cause the throughput to drop significantly. So we need to strike a balance between these two requirements, throughput and probability of success. Hence, with the Markov channel model, the problem reduces to finding the appropriate pushback period after a failed transmission. In the next subsection, we will derive the expressions for the throughput and packet success rate for this scheme using our channel model.

### 4.2.2 Channel Parameters

To find the channel parameters, packet success ratio and throughput, we sketch the Markov chain associated with our first order Markov process. Our main objective here is to find the throughput as a function of the number of deferred time slots, $k$, on a transmission failure[4]. Once we have this function, we can choose $k$ according to the desired throughput based on the incoming data rate. To validate this model using real data, we also find the expression for the PSR.

Our Markov chain has two states, S and F as illustrated in Fig. 4.3. The current state is S if the final packet transmission is successful and F, otherwise. Note that

[4]Upon a successful transmission, the deferral time is 1 (no deferral).

a transition does not necessarily occur every time slot, rather it occurs for every packet transmission attempt. Since we schedule a transmission immediately after a successful event, the expression for $x$ is obtained by substituting $m = 1$ in (4.1). Direct application of (4.3) with $m = k$ gives the expression for $y$.

$$x = P(A_{n+1} = \mathrm{F} | A_n = \mathrm{S}) = p(1 - \alpha) \tag{4.7}$$

$$y = P(A_{n+k} = \mathrm{F} | A_n = \mathrm{F}) = p + (1 - p)\alpha^k \tag{4.8}$$

Notice that the transition probabilities at state F are functions of $k$ as well as $p$ and $\alpha$. This is due to the effect of the pushback period of $k$ time slots after the failed transmission attempts. The associated steady state probabilities are therefore functions of $k$ as well, and these probabilities for state S and state F are respectively,

$$\begin{aligned} \pi_{\mathrm{S}}(k) &= \frac{(1-p)(1-\alpha^k)}{p(1-\alpha) + (1-p)(1-\alpha^k)} \\ \pi_{\mathrm{F}}(k) &= 1 - \pi_{\mathrm{S}}(k). \end{aligned} \tag{4.9}$$

We define the packet success ratio (PSR) as the total fraction of the packets that are successfully transmitted, i.e., it is equal to the steady state probability, $\pi_{\mathrm{S}}(k)$ of state S.

Likewise, we define the throughput ($\rho$) as the number of successful packet transmissions in a unit time slot. To formulate an expression for throughput, consider the following: on a transmission attempt, we wait for $k$ time slots in state F and 1 time slot in state S for the next transmission attempt. Thus, the average number of slots per attempt is $\pi_S(k) + k\pi_F(k)$. Consequently the number of packet transmissions per slot,

$$X(k) = \frac{1}{\pi_S(k) + k\pi_F(k)} = \frac{p(1-\alpha) + (1-p)(1-\alpha^k)}{kp(1-\alpha) + (1-p)(1-\alpha^k)}. \tag{4.10}$$

The resulting throughput, $\rho(k)$, is thus

$$\begin{aligned}\rho(k) &= \pi_S(k)X(k) \\ &= \frac{(1-p)(1-\alpha^k)}{kp(1-\alpha)+(1-p)(1-\alpha^k)}.\end{aligned} \tag{4.11}$$

The theoretical value of the throughput ($\rho$) is plotted as a function of the pushback period $k$ in Fig. 4.4 for different channel parameters $p$ and $\alpha$. In Fig. 4.4, by changing the value of $p$ (keeping $\alpha$ constant), we can see that a channel with lower $p$ gives a better throughput performance. This trend is expected as we have defined $p$ as the probability of transmission failure and a channel with a smaller $p$ will have a lower chance of transmission failure which leads to a higher throughput. For the same value of $p$, larger $\alpha$ implies a higher correlation in the channel quality. Since pushback takes advantage of the correlation in channel quality to schedule transmissions, we can see that the throughput performance of a channel with higher correlation will be better with transmission pushback than a channel with lower correlation.

We tested the validity of our Markov model using an experimental setup in the Kansei testbed [47] by setting up a wireless link between two sensor motes. This link was encircled with seven sensor motes spread around the perimeter of a 20 $m^2$ area. We measured the packet success rate between two motes, while the rest of the motes acted as sources of interference. For this experiment, the wireless link transmitted 36 byte packets at 100 ms interval. We programmed the nodes causing interference to transmit bursts of packets once every second. The burst size was selected according to a uniform distribution between [0,15]. We estimated the two parameters, $\alpha$ and $p$ using the ACK-level data of the entire trace of 30 mins. Note that the purpose of this experiment was to validate the model. In our actual algorithm, the estimation of the two parameters is much simpler and does not depend on a long trace of data. The

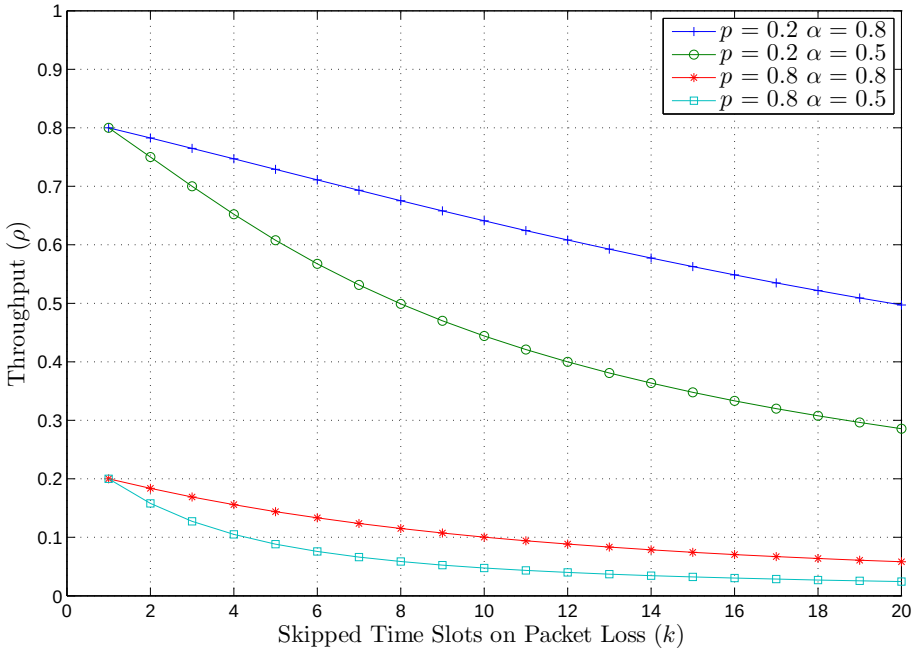



Fig. 4.4: Theoretical throughput ($\rho$) achieved by deferring transmission on a packet loss for different values of $p$ and $\alpha$.

theoretical packet success rate based on the estimated $\alpha$ and its actual experimental value are plotted in Fig. 4.5 as a function of $k$. This plot gives credence to the Markov model.

## 4.3 The Pushback Algorithm

The objective of the pushback algorithm is to estimate the period for which the channel will remain in a poor state and defer retransmissions accordingly in order to conserve energy. In addition, the algorithm must provide similar throughput as CSMA for a reduced number of transmission attempts.

The proposed pushback algorithm is based on CSMA. If a transmission is successful, the next transmission is scheduled by CSMA. However, in case of a failed

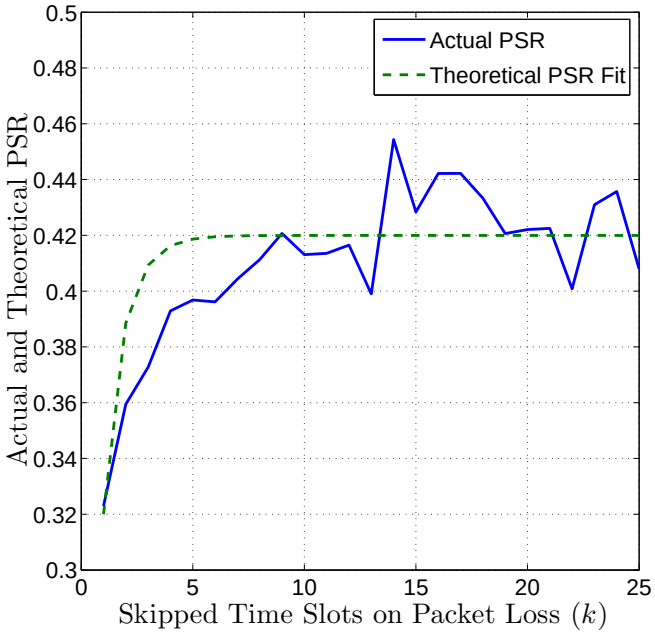



Fig. 4.5: Comparison of the actual PSR gain achieved by our pushback algorithm and the theoretical fit using Eq. (4.9).

transmission, the next transmission is pushed back by $k$ slots after which CSMA takes effect. Note that a pushback "slot" is a packet slot, i.e., the time it takes to transmit a packet. Therefore it is different from the contention slot of CSMA. Also, even though the value of $k$ is generated deterministically, the randomness of the back-off period (in contention slots) of CSMA avoids persistent pushbacks caused by local synchronization of nodes due to similar channel conditions.

When nodes boot up, an initial value $k_{\text{init}}$ is assigned to $k$, and then $k$ is recalculated periodically each time the $m$-th (predefined) transmission failure happens using a simple mechanism based on the estimates of the channel parameters and throughput constraint. Our computation is both practical to implement and is also shown to perform well using simulations in [45].

The proposed mechanism for computing $k$ is based on the formulation presented in Section 4.2. First, $k$ is set to an initial value $k_{\text{init}}$, and based on the ACK-level observations of success and failure in the recent past, a maximum likelihood (ML) estimation of parameters $\alpha$ and $p$ is made. In fact, estimating the transition probabilities of the Markov chain (parameters $x$ and $y$ in Fig. 4.3) is sufficient for the desired ML estimation. Indeed, given the ML estimates $\hat{x}$ and $\hat{y}$ (of $x$ and $y$ respectively), we show next that the solution $(\hat{\alpha}, \hat{p})$ to the system of Eqs. (4.7) and (4.8) gives the ML estimates of $\alpha$ and $p$ which characterize the channel.

**Lemma 4.2.** *For a HMM channel with $n \gg 1$ packet transmissions, the maximum likelihood estimates $\hat{\alpha}_{ML}$ and $\hat{p}_{ML}$ of $\alpha$ and $p$ are given by,*

$$1 - \frac{n_{ss}}{n_s} = \hat{p}_{ML}(1 - \hat{\alpha}_{ML}) \tag{4.12}$$

$$\frac{n_{ff}}{n_f} = \hat{p}_{ML} + (1 - \hat{p}_{ML})\hat{\alpha}_{ML}^{k}. \tag{4.13}$$

*Here $n_s$ and $n_f$ are the number of successful and unsuccessful packet transmissions, $n_{ss}$ is the number of consecutive sucessful packet transmission and $n_{ff}$ is the number of consecutive unsuccessful packet transmissions.*

*Proof.* Consider the ML Estimation of the Markov channel parameters, $\alpha, p$, based on the binary error realization sequence. We assume that the initial state of the chain is picked according to its steady state distribution.

Suppose a transmitter transmits $n \gg 1$ packets and observes the associated ACK sequence. Let the number of successful transmissions be $N_s$ (i.e., the number of failed transmissions is $N_f = n - N_s$), the number of consecutive successful transmissions be $N_{ss}$ and the number of consecutive failed transmissions be $N_{ff}$. First we find the estimates of the Markov chain transition probabilities $x, y$ based on the observations $N_s = n_s,\ N_{ss} = n_{ss}$ and $N_{ff} = n_{ff}$. For notational simplicity, we drop the random variables and use $\mathrm{P}(n_{ss})$ instead of $\mathrm{P}(N_{ss} = n_{ss})$ for instance. The ML estimate for the pair

$$
\begin{aligned}
(\hat{x}_{\mathrm{ML}}, \hat{y}_{\mathrm{ML}}) &= \underset{(x,y)\in(0,1)}{\arg\max}\, \mathrm{P}\left(n_s, n_{ss}, n_{ff} \mid x, y\right) \\
&= \underset{(x,y)\in(0,1)}{\arg\max}\, \mathrm{P}\left(n_{ss}, n_{ff} \mid n_s, x, y\right) \mathrm{P}\left(n_s | x, y\right) \\
&= \underset{(x,y)\in(0,1)}{\arg\max} \Bigg\{ \underbrace{\binom{n_s}{n_{ss}} (1-x)^{n_{ss}} x^{n_s - n_{ss}}}_{\text{(I)}} \\
&\qquad \times \underbrace{\binom{n-n_s}{n_{ff}} (1-x)^{n_{ss}} y^{n_{ff}} (1-y)^{n-n_s-n_{ff}}}_{\text{(II)}} \times \underbrace{\mathrm{P}\left(n_s \mid x, y\right)}_{\text{(III)}} \Bigg\}.
\end{aligned}
\tag{4.14}
$$

The value of $x \in (0,1)$ that maximizes Term (I) is $x_m = 1 - \frac{n_{ss}}{n_s}$ and the value of $y$ that maximizes Term (II) is $y_m = \frac{n_{ff}}{n-n_s}$. If we show

$$(x_m, y_m) = \arg\max_{(x,y)} \mathrm{P}\left(n_s \mid x, y\right), \tag{4.15}$$

then $(\hat{x}_{\mathrm{ML}}, \hat{y}_{\mathrm{ML}}) = (x_m, y_m)$. For $n \gg 1$, $\mathrm{P}\left(N_s = n_s \mid x, y\right)$ is maximized for $n_s = n\pi_s = n\frac{1-y}{x+1-y}$. Thus, if we verify

$$\frac{n_s}{n} = \frac{1-y_m}{x_m + 1 - y_m},$$

then (4.15) is proved. Note that the number of transitions from state '$S$' to state '$F$' is $n_{sf} = n_s - n_{ss}$ and the number of transitions from state '$F$' to state '$S$' is $n_{fs} = n_f - n_{ff}$. Since $n_{sf} = n_{fs} \mp 1$, $\frac{n_{sf}}{n_{fs}} \to 1$ as $n \to \infty$ ($x, y \in (0,1)$, therefore $n_{sf}, n_{fs} \to \infty$). Consequently

$$\begin{aligned}\frac{1-y_m}{x_m+1-y_m} &= \frac{1-\frac{n_{ff}}{n-n_s}}{1-\frac{n_{ss}}{n_s}+1-\frac{n_f f}{n-n_s}} = \frac{n_{fs}/n_f}{n_{sf}/n_s + n_{fs}/n_f} \\ &= \frac{1/n_f}{1/n_s + 1/n_f} = \frac{n_s}{n}\end{aligned}$$

Since $p, \alpha$ is deterministic given $x, y$, the ML estimates $\hat{p}_{\mathrm{ML}}, \hat{\alpha}_{\mathrm{ML}}$ can be found by solving the system of equations

$$\begin{aligned}1 - \frac{n_{ss}}{n_s} &= \hat{p}_{\mathrm{ML}}(1 - \hat{\alpha}_{\mathrm{ML}}) \\ \frac{n_{ff}}{n_f} &= \hat{p}_{\mathrm{ML}} + (1 - \hat{p}_{\mathrm{ML}})\hat{\alpha}_{\mathrm{ML}}^k.\end{aligned}$$

□

In addition, to ensure that a node can sustain the incoming rate of packets, the computation of $k$ must take the incoming packet rate into account. We use the running average of the incoming packet rate, $\rho_{\mathrm{new}} = \gamma/t_e + (1-\gamma)\rho_{\mathrm{old}}$, to represent

| **Table** | **Eqs.** | **Purpose** |
|---|---|---|
| $\mathrm{T}_\alpha(x, y, k)$ | Eqs. (4.7), (4.8) | To compute $\alpha$ for given $x$, $y$ and $k$. |
| $\mathrm{T}_k(p, \alpha, \rho)$ | Eq. (4.11) | To compute $k$ for given $p$, $\alpha$ and $\rho$. |

Table 4.2: Lookup tables used in the pushback algorithm.

the throughput constraint. Here $\rho_{\text{new}}$ ($\rho_{\text{old}}$) is the new (previous) estimate of the throughput requirement, $t_e$ is the time elapsed since the last packet arrival and $\gamma$ is smoothing factor. Using $\rho_{\text{new}}$ (throughput constraint), $\hat{\alpha}$ and $\hat{p}$, Eq. (4.11) can be used to compute the new value of $k$.

To avoid the complexity of direct computation of $k$, we propose the use of look-up tables. The first table $T_{\hat{\alpha}}(x, y, k)$ contains the values of $\hat{\alpha}$ corresponding to $k$ and discretized $x$ and $y$. The second table $T_k(\hat{p}, \hat{\alpha}, \rho)$ contains the values of $\rho(k)$ corresponding to $k$ and discretized $\alpha$ and $p$. A brief description of the various tables used is given in Table 4.2. These tables will not change during the operation of the node, so they can be computed offline and uploaded. The available storage spaces on the nodes will determine the size of the tables. From our experimental experience, it should suffice to have a $10 \times 20 \times 20$ table. For instance, this table can have 10 values of $k$ (2 to 11), 20 values of $p$ (0 to 0.95 in increments of 0.05) and 20 values of $\alpha$ (0 to 0.95 in increments of 0.05). These numbers could be stored as integers between 0 and 100. Hence, the two tables would take 8K bytes.

In summary, upon the $m$-th transmission failure, function **Pushback()**, shown in Table 4.3, is called. In this algorithm, The ML estimates $\hat{x}$ and $\hat{y}$ are calculated in lines 3 and 4. A table lookup is employed to find the value of $\hat{\alpha}$ corresponding to $\hat{x}$, $\hat{y}$ and $k$, and then $\hat{p}$ can be calculated according to Eq. (4.7). Finally the pushback

**Input:** Throughput constraint $\rho$.

**Output:** Pushback duration $k$.

**Given:** A three-dimensional lookup table $\mathrm{T}_{\hat{\alpha}}(\hat{x}, \hat{y}, k)$ and $\mathrm{T}_{\hat{p}}(\hat{x}, \hat{y}, k)$ containing values of $\hat{\alpha}$ and $\hat{p}$ for various values of $\hat{x}$, $\hat{y}$ and $k$, failure count $m$, a three-dimensional lookup table $\mathrm{T}_k(\hat{p}, \hat{alpha}, \rho)$ containing values of the pushback period $k$ for various values of $\hat{p}$, $\hat{\alpha}$ and $\rho$.

**Pushback()**

1: **if** $(failureCount = m)$ **then**
2: $failureCount \leftarrow 0$
3: $\hat{x} \leftarrow \frac{\text{Number of S}\rightarrow\text{F transitions}}{\text{Total number of stays in S states}}$
4: $\hat{y} \leftarrow \frac{\text{Number of F}\rightarrow\text{F transitions}}{\text{Total number of stays in F states}}$
5: $\hat{\alpha} \leftarrow \mathrm{T}_{\hat{\alpha}}(\hat{x}, \hat{y}, k)$
6: $\hat{p} \leftarrow \hat{x}/(1 - \hat{\alpha})$
7: $k \leftarrow \mathrm{T}_k(\hat{p}, \hat{\alpha}, \rho)$
8: **end if**Delay the retransmission for $k$ slots

Table 4.3: The Pushback Algorithm.

period $k$ is estimated using another table lookup with the appropriate values of $\hat{p}$, $\hat{\alpha}$ and $\rho$.

### 4.3.1 Remedial Mechanisms

The pushback algorithm above can work well if the real packet loss pattern is captured well by our channel model introduced earlier and the transition probabilities are accurately measured. However, either of them may deviate from reality, in which case the throughput may not be maintained if $k$ is chosen too aggressively. Hence, we introduce two remedial mechanisms to solve such problems.

#### Measuring Actual Pushback Amount

In our pushback algorithm, the delay amount, $k$ slots, is calculated according to the state transition probabilities and the throughput constraint. However, after delaying $k$ slots, nodes may need to delay their retransmissions further due to contentions from other senders. This could lead to the loss in throughput since the delaying amount is longer than expected by the model. Hence, the running average of the difference between the calculated delay amount and the actual delay amount is maintained, and subtracted from the newly calculated $k$.

#### Controlling the Pushback Amount at the Interface Queue

Once our channel model deviates from the actual channel, simply adjusting the $k$ as in Section 4.3.1 may still not work very well. To cope with such situations, we let the interface queue impose a pushback control policy to speed up the packet forwarding once the queue is excessively built up. This policy simply commands the

pushback algorithm to fall back to CSMA (using $k = 1$) if the queue length is above a certain threshold. In our evaluations, this value is set to half of the queue capacity.

# CHAPTER 5

# ENERGY MANAGEMENT IN SENSOR NETWORKS WITH REPLENISHMENT

## 5.1 Introduction

One of the major requirements for sensor networks is the ability of operate unattended for long durations. However, the lack of easy access to a continuous power source in most scenarios and the limited lifetime of batteries have hindered the deployment of these networks. With new and exciting developments in the areas of renewable sources of energy [4, 48, 49, 50, 51, 52, 53] it is feasible for sensor networks to operate unsupervised for extended periods. These renewable sources of energy could be attached to sensing nodes and would typically provide energy replenishment at a rate that could be variable and dependent on the surroundings. For example, self-powered sensors have been developed that rely on harvesting strain and vibration energies from their working environment [49], as well as other types of energy sources including solar cells [4, 50, 51].

In this chapter, we analyze the *limits* of the *performance* of networks comprised of sensor nodes with limited energy, being replenished at a variable rate. We provide a simple localized *energy management scheme* that achieves a performance, close to

the optimal scheme that has access to an unlimited energy reservoir. Indeed, we show that, if the performance can be measured by a general utility function of the energy, it is possible to observe a quadratic decay for the probability of complete battery discharge, and at the same time achieve a $\Theta\left(\frac{(\log M)^2}{M^2}\right)$ convergence to the optimal utility[5]. Here $M$ is the total capacity of the energy source. Based on the insights developed, we address the problem of energy management in the presence of a finite data buffer. We modify our basic energy management scheme to achieve a $\Theta\left(\frac{(\log K)^2}{K^2}\right)$ convergence to the maximum utility achieved by a scheme that has access to an infinite data and energy buffers. Here $K$ is the data buffer size. In addition, this scheme achieves an exponential decay with $M$ for the battery discharge probability and a quadratic decay with $K$ for the buffer overflow probability. To evaluate these decay rates, the main tools we use are the *large deviations theory* and *diffusion approximations*.

The added dimension of renewable energy makes the problem of energy management in sensor networks substantially different from its non-replenishment counterpart. For nodes with replenishment, conservative energy expenditure may lead to missed recharging opportunities due to battery capacity limitations. On the other hand, aggressive usage of energy may cause battery outages that leads to lack of coverage or connectivity for certain time periods. Thus, new techniques must be developed to balance these seemingly contradictory goals to maximize performance. Here, our *main goal* will be to identify the performance limits of sensor nodes with energy replenishment and provide guidelines to approach these limits.

[5]The following notations will be used to compare rates of convergence: $a_n = \mathrm{O}(b_n)$ if $a_n$ goes to zero at least as fast as $b_n$; $a_n = \mathrm{o}(b_n)$ if $a_n$ goes to zero strictly faster than $b_n$; $a_n = \Theta(b_n)$ if $a_n$ and $b_n$ go to zero at the same rate; $a_n = \Omega(b_n)$ if $a_n$ goes to zero no faster than $b_n$.

Many fundamental wireless communication and networking problems can be stated as *utility maximization* problems, subject to energy constraints. The utility function can be the throughput (e.g., in energy efficient routing), the probability of detection of an intruder (e.g., in coverage) or the network lifetime (e.g., in sleep-wake scheduling) or the achievable rate of reliable transmission in basic wireless communication. These problems have been mainly addressed for stations with unlimited and/or non-replenishing energy sources. Here, we focus on basic guidelines to formulate energy aware schemes that exploit the replenishment energy to maximize the utility of a node. This problem has two aspects: (a) development of an optimal or near-optimal energy management scheme, and (b) evaluation of the performance metrics (e.g., probability of battery discharge, probability of buffer overflow) for this scheme. Development of an optimal scheme typically requires stochastic optimization techniques involving high computational overheads that might be unsuitable for sensor nodes. Consequently, we will focus our attention on simple localized solutions that achieve near-optimal or asymptotically optimal performance. Evaluation of performance metrics can be quite a hard problem for all but the simplest models. In this work, we formulate an analytical framework to model the battery and buffer dynamics for our energy management policies. We use tools from large deviations theory and diffusion approximation to find closed-form expressions for these metrics. These techniques allow us to analyze a large variety of system models, as we need to make only mild assumptions on the battery charging and data arrival processes.

There have been recent works that have studied different problems in networks with energy replenishment. Kar, et. al., [54] have proposed an activation scheme for rechargeable sensors that maximizes the network-level utility of sensing networks. The

utility function in [54] depends on the number of active sensors. Gatzianas, et. al., [55] use back pressure policies to maximize the network flow of information in networks with energy replenishment. While [54] and [55] look at the total system utility, we will focus on the analyzing node-level performance leading to localized energy management schemes. Liu, et. al., [56] have derived a battery control scheme similar to the one described in this work. In addition to providing stronger convergence results than those in [56], we also consider the effect of a finite data buffer in this thesis. Kansal, et. al., [57] introduce the concept of energy neutral operation which can be defined as the operation mode wherein the energy consumed by a node is less than or equal to the energy harvested. Vigorito, et. al., [58] extend the idea of energy neutral operation to propose an algorithm that attempts to keep the battery state close to a fixed level and at the same time stabilizes the duty cycle in order to maximize system performance. Sharma, et. al., [59] have proposed energy management scheme for energy harvesting nodes that is throughput optimal. However, in all these works [57, 58, 59], the authors do not analyze battery discharge or queue overflow probabilities for their energy management schemes.

The outline of this chapter is as follows. We first state the general form of the utility maximization problem in Section 5.2 and show ways to achieve the maximum achievable utility with replenishing sources. In Section 5.3 we add a finite buffer to the problem and study energy management schemes that achieve optimal utility asymptotically while keeping the probabilities of battery discharge and buffer overflow low. Finally, we numerically evaluate the performance of our energy management schemes in Section 5.4.

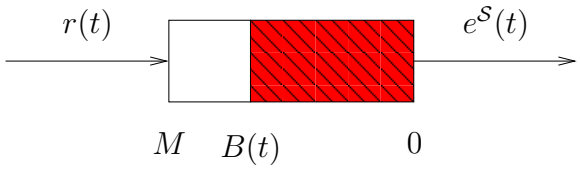



Fig. 5.1: Energy source with a replenishment rate $r(t)$.

## 5.2 Achieving Maximum Utility With a Finite-Battery Constraint

Fig. 5.1 shows the energy source (or the battery) of a node. The total capacity of this battery is $M$ units of energy. We denote the total available energy in the battery as $B(t)$, where $t$ is the discrete time index. The battery replenishes at a rate $r(t)$. The replenishment process $\{r(t), t \geq 1\}$ is assumed to be an ergodic stochastic process with a long term mean $\lim_{\tau\to\infty} \mathrm{E}\left[r(\tau)\right] = \mu$. A power allocation policy $\mathcal{S}$ draws energy from this battery at a rate $e^{\mathcal{S}}(t)$ to achieve certain tasks. The success of the node in achieving these tasks is measured in terms of a concave and non-decreasing utility function[6] $U(e^{\mathcal{S}}(t))$ of the consumed energy $e^{\mathcal{S}}(t)$. We define the time average utility,

$$\bar{U}^{\mathcal{S}}(\tau) = \frac{1}{\tau}\sum_{t=1}^{\tau} U(e^{\mathcal{S}}(t)). \tag{5.1}$$

We consider the optimization problem in which a node tries to maximize its long-term average utility, $\bar{U}^{\mathcal{S}} = \lim_{\tau\to\infty} \bar{U}^{\mathcal{S}}(\tau)$, subject to battery constraints:

$$\begin{aligned} \max_{\{e^{\mathcal{S}}(t),\ t\geq 1\}} \quad & \bar{U}^{\mathcal{S}} \\ \text{subject to} \quad & B(t) = \max\left\{0, \min\{M, B(t-1) + r(t) - e^{\mathcal{S}}(t-1)\}\right\} \\ \text{and} \quad & e^{\mathcal{S}}(t) \leq B(t). \end{aligned} \tag{5.2}$$

[6]Note that, in many practical scenarios, it is reasonable to assume that the utility function is non-decreasing and there is diminishing returns for increasing power. Equivalently, one can also consider a convex and non-increasing cost function.

This optimization problem can be solved using Markov decision process (MDP) techniques. Since solving MDPs is computationally intensive, these methods are not suited for resource-limited sensor nodes. Consequently, we seek schemes that are easy to implement and achieve close to optimal performance. Note that, with an unlimited energy reservoir (i.e., $M = \infty$) and average energy replenishment rate $\mu$, the maximum average utility that a node can achieve is $U(\mu)$. This can be achieved by simply choosing $e^{\mathcal{S}}(t) = \mu$ for all $t$, and a higher utility cannot be achieved due to Jensen's inequality. However, if $M < \infty$, achieving $\bar{U}^{\mathcal{S}} = U(\mu)$ is not possible. Indeed, if one uses $e^{\mathcal{S}}(t) = \mu$, for all $t \geq 1$, $B(t)$ will occasionally get discharged completely, due to the variability of $r(t)$. At such instances, $e^{\mathcal{S}}(t)$ has to be set to 0 leading to a service outage, which is highly undesirable. One may think that as $M$ increases, it is straightforward to achieve an average utility close to the upper bound $U(\mu)$. However, this statement is not true in general as we will show next.

### 5.2.1 Basic Limitations of Power Allocation Schemes

In this section we show that there is a trade-off between achieving maximum utility and keeping the discharge probability low. First, we make some weak assumptions on the replenishment process $r(t)$, which we will be using throughout this chapter. In particular, we assume that the asymptotic semi-invariant log moment generating function,

$$\bar{\Lambda}_r(s) = \lim_{\tau \to \infty} \frac{1}{\tau} \log \mathrm{E}\left[\exp\left(s \sum_{t=1}^{\tau} r(t)\right)\right], \tag{5.3}$$

of $r(t)$ exists for $s \in (-\infty, s_{\max})$, for some $s_{\max} > 0$. We also assume that the asymptotic variance $\bar{\sigma}_r^2 \triangleq \lim_{\tau \to \infty} \frac{1}{\tau} \operatorname{var} \sum_{t=1}^{\tau} r(t)$ of $r(t)$ exists. Note however that, in practice, the recharging process is not necessarily stationary. The statistics of $r(t)$

has significant variations due to clouds and the solar power at different times of the day. Hence, these assumptions should be interpreted as follows: The battery size $M$ is large enough for the variations in $r(t)$ to average out nicely over the time scale that $B(t)$ changes significantly and that there are no long-range dependencies in $r(t)$.

**Theorem 5.1.** *Consider any concave and non-decreasing utility function $U(\cdot)$. If an ergodic energy management scheme $\mathcal{S}$ has a discharge probability $p^{\mathcal{S}}_{discharge}(M) = \Theta(\exp(-\alpha_c M))$ for some constant $\alpha_c > 0$, then the time average utility for policy $\mathcal{S}$, $\bar{U}^{\mathcal{S}}$, satisfies $U(\mu) - \bar{U}^{\mathcal{S}} = \Omega(1)$.*

*Proof.* Consider an energy management scheme $\mathcal{S}$ that uses $e^{\mathcal{S}}(t)$ units of energy in the time slot $t$ with $\mathrm{E}\left[e^{\mathcal{S}}(t)\right] < \mathrm{E}\left[r(t)\right]$[7]. Note that Scheme $\mathcal{S}$ can be deterministic or randomized, with the only restriction being that the asymptotic semi-invariant log moment generating function $\bar{\Lambda}_{d^{\mathcal{S}}}(s)$ of the net battery drift $d^{\mathcal{S}}(t) \triangleq e^{\mathcal{S}}(t) - r(t)$ should exist. Let $s^*_{d^{\mathcal{S}}}$ be the positive root[8] of $\bar{\Lambda}_{d^{\mathcal{S}}}(s)$, i.e., $\bar{\Lambda}_{d^{\mathcal{S}}}(s^*_{d^{\mathcal{S}}}) = 0$.

Before we prove Theorem 5.1, we state the following lemma. This lemma is adapted from [60]. The detailed proof is provided in Appendix A.1.

**Lemma 5.1.** *The probability of battery discharge under Scheme $\mathcal{S}$ with battery size $M$ follows $p^{\mathcal{S}}_{discharge}(M) = \Theta(\exp(-s^*_{d^{\mathcal{S}}} M))$, where $s^*_{d^{\mathcal{S}}}$ is the positive root of $\bar{\Lambda}_{d^{\mathcal{S}}}(s)$.*

Since we have assumed that $\{e^{\mathcal{S}}(t), t > 0\}$ is an ergodic process, the time-average utility achieved by Scheme $\mathcal{S}$ is given by $\bar{U}^{\mathcal{S}} = \mathrm{E}\left[U(e^{\mathcal{S}}(t))\right]$. The difference between

[7] If the scheme $\mathcal{S}$ does not satisfy this condition, the battery will not operate under energy neutral conditions and it will discharge with probability 1.

[8] Note that $\bar{\Lambda}_{d^{\mathcal{S}}}(0) = 0$ and $\left.\frac{\partial \bar{\Lambda}_{d^{\mathcal{S}}}(s)}{\partial s}\right|_{s=0} = \lim_{\tau\to\infty} \mathrm{E}\left[d^{\mathcal{S}}(\tau)\right] < 0$, consequently $s^*_{d^{\mathcal{S}}} > 0$ will exist.

the utilities is given by,

$$\begin{aligned} U(\mu) - \bar{U}^{\mathcal{S}} &= U(\mu) - \mathrm{E}\left[U(e^{\mathcal{S}}(t))\right] \\ &\geq U(\mu) - U(\mathrm{E}\left[e^{\mathcal{S}}(t)\right]) \qquad (5.4) \\ &= \Omega(1). \qquad (5.5) \end{aligned}$$

Where Eq. (5.4) follows from Jensen's inequality and Eq. (5.5) follows from the fact that $\mu > \mathrm{E}\left[e^{\mathcal{S}}(t)\right]$ and $U(\cdot)$ is a non-decreasing function. By choosing $\alpha_c = s^*$ in Lemma 5.1, we get the required scaling law $p^{\mathcal{S}}_{\text{discharge}} = \Theta(\exp(-\alpha_c M))$. This completes the proof. □

Theorem 5.1 states that an energy management scheme cannot achieve exponential decay in discharge probability and convergence (even asymptotically) to the maximum average utility function simultaneously. In Lemma 5.1 we have applied the *large deviations* technique to the net drift of the battery process to find $p^{\mathcal{S}}_{\text{discharge}}(M)$. Jensen's inequality is then used to lower bound the difference between $U(\mu)$ and $\bar{U}^{\mathcal{S}}$. The question we answer next is, "how close can the average utility $\bar{U}^{\mathcal{S}}$ get to the upper bound $U(\mu)$ asymptotically, as $M \to \infty$, while keeping the battery discharge probability low?"

### 5.2.2 An Asymptotically Optimal Power Allocation Scheme

We can infer that by choosing a battery drift that goes to zero with increasing battery size, one might achieve a long-term average utility that is close to $U(\mu)$ as $M$ increases. However, smaller drift away from the empty battery state implies a higher discharge probability. In the following theorem, we show that one can achieve a quadratic decay for the battery discharge probability, and at the same time achieve a utility that approaches the maximum long-term utility value as $(\log M)^2/M^2$.

**Theorem 5.2.** *Consider any concave and non-decreasing utility function $U(e(t))$ such that $\left|\frac{\partial^2 U(e)}{\partial e^2}\right| < \infty$ for all $e > 0$. Given any $\beta \geq 2$, there exists a power allocation scheme $\mathcal{B}$ such that the battery discharge probability $p^{\mathcal{B}}_{discharge}(M) = O(M^{-\beta})$ and $U(\mu) - \bar{U}^{\mathcal{B}} = \Theta\left(\frac{(\log M)^2}{M^2}\right)$.*

*Proof.* Recall that the scheme $\mathcal{B}$ is described as,

$$e^{\mathcal{B}}(t) = \begin{cases} \mu - \delta^-, & B(t) \leq M/2 \\ \mu + \delta^+, & B(t) > M/2 \end{cases}, \tag{5.6}$$

for some pair $\delta^-, \delta^+$, that will be chosen later.

Depending on whether the battery state is less than (or more than) half full, the expected drift of the battery state becomes positive (or negative). Given $B(t) \leq M/2$, the asymptotic semi-invariant log-moment generating function of the battery state drift, $d^- \triangleq r(t) - (\mu - \delta^-)$, is

$$\begin{aligned} \bar{\Lambda}_{d^-}(s) &= \lim_{\tau \to \infty} \frac{1}{\tau} \log \mathrm{E}\left[\exp\left(s \sum_{t=1}^{\tau} d^-(t)\right)\right] \\ &= \bar{\Lambda}_r(s) - s(\mu - \delta^-). \end{aligned} \tag{5.7}$$

Where $\bar{\Lambda}_r(s)$ is given by Eq. (5.3). Let $s^*_{d^-}$ be the negative root[9] of $\bar{\Lambda}_{d^-}(s)$, i.e., $\bar{\Lambda}_{d^-}(s^*_{d^-}) = \bar{\Lambda}_r(s^*_{d^-}) - s^*_{d^-}(\mu - \delta^-) = 0$. Also as $\delta^- \to 0$, $s^*_{d^-} \to 0$.

Before we prove Theorem 5.2, we state and prove the following lemmas. Lemma 5.2 gives the rate of decay of the probability of battery discharge with respect to the battery size $M$ for the Scheme $\mathcal{B}$. This lemma is a minor modification of Lemma 5.1, however we provide the entire proof in Appendix A.2 for completeness. Lemma 5.3 expresses the rate decay exponent $s^*_{d^-}$ for scheme $\mathcal{B}$ in terms of the asymptotic variance of energy replenishment process $r(t)$.

[9]Note that $\bar{\Lambda}_{d^-}(0) = 0$ and $\left.\frac{\partial \bar{\Lambda}_{d^-}(s)}{\partial s}\right|_{s=0} = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathrm{E}\left[d^-(t)\right] = \delta^- > 0$ for $\delta^- > 0$, consequently $s^*_{d^-} < 0$ will exist.

**Lemma 5.2.** *The probability of battery discharge under Scheme $\mathcal{B}$ with battery size $M$ follows,*

$$p^{\mathcal{B}}_{discharge}(M) = \Theta\left(\exp\left(\frac{s^*_{d^-}M}{2}\right)\right), \tag{5.8}$$

*where $s^*_{d^-}$ is the negative root of $\bar{\Lambda}_{d^-}(s)$.*

**Lemma 5.3.** *The asymptotic variance of $r(t)$, $\bar{\sigma}^2_r \triangleq \lim_{T\to\infty}\frac{1}{T}\operatorname{var}\sum_{t=1}^{T} r(t)$ satisfies,*

$$\left.\frac{\partial s^*_{d^-}}{\partial \delta^-}\right|_{\delta^-=0} = -\frac{2}{\bar{\sigma}^2_r} \tag{5.9}$$

*Proof.* First, we define $\bar{\Lambda}^{(n)}_{d^-}(0) = \left.\frac{\partial^n \bar{\Lambda}_{d^-}(s)}{\partial s^n}\right|_{s=0}$. The Taylor series expansion of $\bar{\Lambda}_{d^-}(s^*_{d^-})$ about $s = 0$ gives,

$$\begin{aligned}
0 = \bar{\Lambda}_{d^-}(s^*) &= \sum_{n=0}^{\infty} \bar{\Lambda}^{(n)}_{d^-}(0)\frac{(s^*_{d^-})^n}{n!} \\
&= \underbrace{\bar{\Lambda}_{d^-}(0)}_{=0} + \bar{\Lambda}^{(1)}_{d^-}(0)s^*_{d^-} + \bar{\Lambda}^{(2)}_{d^-}(0)\frac{(s^*_{d^-})^2}{2!} + \cdots \\
&= \sum_{n=1}^{\infty} \bar{\Lambda}^{(n)}_{r}(0)\frac{(s^*_{d^-})^n}{n!} - (\mu - \delta^-)s^*_{d^-} \\
&= \mu s^*_{d^-} + \sum_{n=2}^{\infty} \bar{\Lambda}^{(n)}_{r}(0)\frac{(s^*_{d^-})^n}{n!} - (\mu - \delta)s^*_{d^-}.
\end{aligned}$$

Rearranging the terms, we have

$$\sum_{n=2}^{\infty} \bar{\Lambda}^{(n)}_{r}(0)\frac{(s^*_{d^-})^{n-1}}{n!} = -\delta^-. \tag{5.10}$$

Differentiating with respect to $\delta^-$, we have,

$$\frac{\partial s^*_{d^-}}{\partial \delta^-}\sum_{n=2}^{\infty} \bar{\Lambda}^{(n)}_{r}(0)\frac{(n-1)(s^*_{d^-})^{n-2}}{n!} = -1.$$

As $\delta^- \to 0$, $s^*_{d^-} \to 0$ the above expression reduces to,

$$\left.\frac{\partial s^*_{d^-}}{\partial \delta^-}\right|_{\delta^-=0} \bar{\Lambda}^{(2)}_{r}(0)\frac{1}{2} = -1. \tag{5.11}$$

Since $\bar{\Lambda}_r^{(2)}(0) = \bar{\sigma}_r^2$, Eq. (5.11) becomes,

$$\left.\frac{\partial s_{d^-}^*}{\partial \delta^-}\right|_{\delta^-=0} = -\frac{2}{\bar{\sigma}_r^2}.$$

Which implies,

$$\frac{\partial s_{d^-}^*}{\partial \delta^-} = -\frac{2}{\bar{\sigma}_r^2} + \mathrm{o}(\delta^-).$$

Integrating this expression, we have,

$$s_{d^-}^* = -\frac{2}{\bar{\sigma}_r^2}\delta^- + \mathrm{o}(\delta^-). \tag{5.12}$$

□

Substituting the expression for $s_{d^-}^*$ from Lemma 5.3 in Eq. (5.8), we have,

$$p_{\text{discharge}}^{\mathcal{B}}(M) = \mathrm{O}\left(\exp\left(\left(-\frac{2}{\bar{\sigma}_r^2}\delta^- + \mathrm{o}\left(\delta^-\right)\right)\frac{M}{2}\right)\right).$$

By choosing $\delta^- = \alpha\frac{\ln M}{M}$ and $\alpha = \beta\bar{\sigma}_r^2$ we have $p_{\text{discharge}}^{\mathcal{B}}(M) = \mathrm{O}(M^{-\beta})$.

Next we show that with $\delta^+ = \alpha\frac{\ln M}{M}$ the scheme achieves an average utility $\bar{U}^{\mathcal{B}}$ such that $U(\mu) - \bar{U}^{\mathcal{B}} = \Theta\left(\frac{(\ln M)^2}{M^2}\right)$. The instantaneous utility $U(e^{\mathcal{S}}(t))$ is zero with an $\mathrm{O}(M^{-\beta})$ probability. For the remaining time, the utility alternates between $U^+$ and $U^-$ as illustrated in Fig. 5.2. The Taylor series expansion of the utility functions about $\mu$ will be,

$$U^+ = U(\mu) + U^{(1)}(\mu)\delta^+ + U^{(2)}(\mu)(\delta^+)^2 + \mathrm{o}((\delta^+)^2),$$

and,

$$U^- = U(\mu) - U^{(1)}(\mu)\delta^- + U^{(2)}(\mu)(\delta^-)^2 + \mathrm{o}((\delta^-)^2).$$

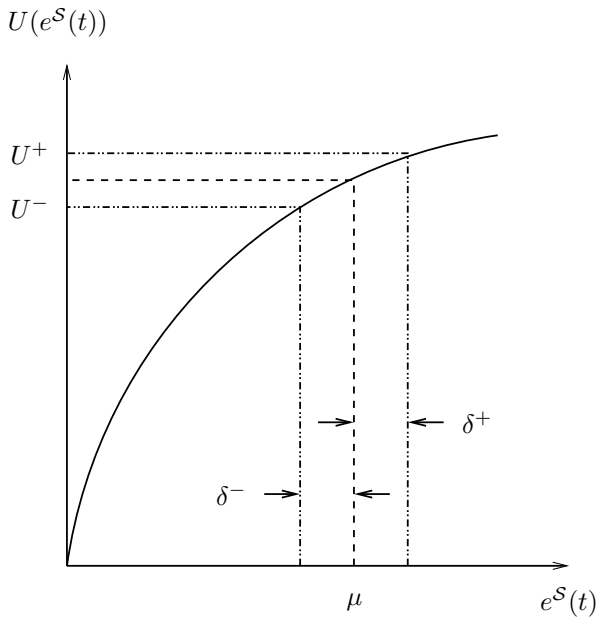



Fig. 5.2: With scheme $\mathcal{B}$, utility alternates between $U^+$ and $U^-$

We define $\rho^+$ as the fraction of time that $B(t) > M/2$ and $\rho^- = 1-\rho^+$ as the fraction of time that $B(t) \leq M/2$. The average utility $\bar{U}^{\mathcal{S}}$ can be written as,

$$\begin{aligned}\bar{U}^{\mathcal{S}} &= (\rho^+ U^+ + \rho^- U^-)(1 - p^{\mathcal{S}}_{\text{discharge}}(M)) \\ &= U(\mu) + U^{(1)}(\mu)(\rho^+\delta^+ - \rho^-\delta^-) + \Theta\left(\frac{(\ln M)^2}{M^2}\right), \end{aligned} \tag{5.13}$$

where Eq. (5.13) follows from the fact that $\delta^-, \delta^+ = \alpha\frac{\ln M}{M}$ and $p^{\mathcal{B}}_{\text{discharge}}(M) = \mathrm{O}(M^{-\beta})$ where $\beta \geq 2$.

From conservation of energy,

$$\rho^+(\mu+\delta^+) + \rho^-(1 - p^{\mathcal{B}}_{\text{discharge}}(M))(\mu - \delta^-) = \mu(1 - p^{\mathcal{B}}_{\text{overflow}}(M)), \tag{5.14}$$

where $p^{\mathcal{B}}_{\text{overflow}}(M)$ is the probability of the battery being full under the power allocation scheme $\mathcal{B}$. By a straightforward extension of Lemmas 5.2 and 5.3, it can be

shown that $p^{\mathcal{B}}_{\text{overflow}}(M) = \Theta\left(M^{-\beta}\right)$. We can simplify Eq. (5.14) as,

$$\begin{aligned}\rho^+\delta^+ - \rho^-\delta^- &= (\rho^-(\mu - \delta^-) - \mu)\Theta\left(M^{-\beta}\right) \\ &= \Theta\left(M^{-\beta}\right). \end{aligned} \tag{5.15}$$

By substituting Eq. (5.15) in the first-order term of Eq. (5.13), we observe that the scheme $\mathcal{B}$ achieves $U(\mu) - \bar{U}^{\mathcal{S}} = \Theta\left(\frac{(\ln M)^2}{M^2}\right)$. This concludes the proof. □

Here, we illustrated that with simple localized solutions, it is possible to achieve good scaling laws for the performance of a given task if the replenishment process behaves "nicely." To illustrate the theorem we consider a specific example.

**Example 5.1 Capacity of an AWGN Channel**

We study the basic limitations of point to point communication with finite but replenishing energy sources. For simplicity, we consider the additive white Gaussian noise (AWGN) channel. At time $t$, the transmitter transmits a complex valued block (packet) $X(t)$ and the receiver receives $Y(t)$. We have,

$$Y(t) = hX(t) + W(t), \tag{5.16}$$

where the channel gain $h$ is a complex constant and $W(t)$ is additive white (complex) Gaussian noise with two sided power spectral density $N_0/2$. We define the channel SNR as $\gamma = \mathrm{E}\left[|h|^2\right]/N_0$. The maximum amount of data that could be reliably communicated [15] over this channel with an amount of energy $e(t)$ at time $t$ is,

$$C(e(t)) = \log_2\left(1 + e(t)\gamma\right) \text{ bits/channel use}, \tag{5.17}$$

assuming the block size is long enough so that sufficient averaging of additive noise is possible. Thus, the rate at which reliable communication can be achieved is a

concave non-decreasing function of the transmit power and it can be viewed as our utility function. Consequently, using a constant power $\mu$, the maximum utility of $\bar{C} = C(\mu)$ can be achieved, which is the famous AWGN channel capacity result.

Now, we generalize the AWGN capacity result to the case with finite energy sources. Suppose that we want to transmit the maximum amount of data over the AWGN channel, using a battery of energy capacity $M$ and a replenishment rate $r(t)$. We assume that each time slot is long enough for sufficiently long code blocks to be formed. We substitute $U(\cdot)$ with $C(\cdot)$ in Eq. (5.1) to get the relevant optimization problem. With an unlimited energy source ($M = \infty$) of limited average power $\mu$, the maximum achievable long term average rate, i.e., the channel capacity is $C(\mu) = \log_2(1 + \mu\gamma)$ bits/channel use. By using the energy management scheme $\mathcal{B}$ given in Eq. (5.6), an average rate $\bar{C}^{\mathcal{B}}$ can be achieved such that $C(\mu) - \bar{C}^{\mathcal{B}} = \Theta\left(\frac{(\log M)^2}{M^2}\right)$ while the battery discharge probability follows $p^{\mathcal{B}}_{\text{discharge}}(M) = \Theta(M^{-\beta})$ for some $\beta \geq 2$.

## 5.3 Achieving Maximum Utility with Finite Buffer and Battery Constraints

So far, we have shown how to maximize a concave non-decreasing utility function subject to battery constraints. At every point in time, one should choose a power level as close to the replenishment rate as the battery constraints allow and this way one can asymptotically achieve a performance very close to that with unlimited energy sources. The main limitation of this approach is that it may not be feasible for some applications in practice. For instance in many sensor network applications, data is stored in finite buffers for transmission. Since scheme $\mathcal{B}$ does not adapt to the buffer state, this may lead to data buffer overflows. To overcome these limitations, in this

section, we investigate energy management schemes with finite buffer and battery constraints.

In this section, we extend the problem introduced in Section 5.2 to the case when data packets arrive at a node and are kept in a finite buffer before transmission. Hence, the task is to transmit packets arriving at the data buffer without losing them due to data buffer overflow and/or battery discharge. We define $Q(t)$ as the data queue state at time $t$, and the data buffer size is $K < \infty$. The data arrival process $a(t)$, represents the amount of data (in bits) arriving at the data buffer in the time slot $t$. The process $\{a(t), t \geq 1\}$ is a stochastic process independent of the energy replenishment process $\{r(t), t \geq 1\}$ and $\lim_{\tau\to\infty} \mathrm{E}\left[a(\tau)\right] = \lambda$. We assume that the process $a(t)$ has a finite asymptotic variance $\bar{\sigma}_a^2 = \lim_{\tau\to\infty} \frac{1}{\tau} \operatorname{var} \sum_{t=1}^{\tau} a(t)$. The energy replenishment model is the same as used previously. The rate-power function for the wireless channel is given by $C(\cdot)$. We also assume that $\mu < C(\lambda)$ which is a necessary condition for system stability [59]. Without this condition, there exists no joint energy and data buffer control policy, which can keep the battery discharge and data buffer overflow probabilities low at the same time.

The objective of an efficient energy management scheme in this case is to maximize the average utility function of the data transmitted subject to battery and data buffer constraints:

$$\begin{aligned} \max_{e(t),\ t\geq 1} \quad & \lim_{\tau\to\infty} \frac{1}{\tau} \sum_{t=1}^{\tau} U_D(C(e(t))) \\ \text{subject to} \quad & B(t) = \max\{0, \min\{M, B(t-1) + r(t) - e(t-1)\}\}, \\ & Q(t) = \max\{0, \min\{K, Q(t-1) + a(t) - C(e(t-1))\}\}, \\ & e(t) \leq B(t) \quad \text{and} \quad C(e(t)) \leq Q(t). \end{aligned} \tag{5.18}$$

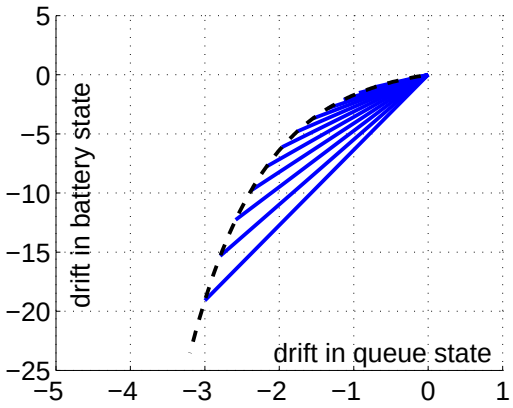



Fig. 5.3: Possible drift directions for $(Q(t), B(t))$ for an AWGN channel of channel SNR 0 dB.

Here $U_D(C(e))$ is a non-decreasing concave utility gained by transmitting $C(e)$ bits.

### 5.3.1 An Asymptotically Optimal Energy Management Scheme

Intuitively, an optimal or near-optimal solution for the optimization problem (5.18) should jointly control the data queue state and the battery state to avoid energy outage and data overflow while maximizing the utility. However, the main complexity in such an approach stems from the fact that the drifts of $Q(t)$ and $B(t)$ are dependent. In Fig. 5.3 we illustrate the connection between the service rate and the energy consumed at a time slot for an AWGN channel with SNR 0 dB. For instance, to provide 3 units of service, the node needs to consume $\sim 18$ units of energy.

With this dependence, the parameter of importance becomes the relative "size" of the data buffer with respect to the battery. In this chapter, we assume a *large battery regime*, where the time scale at which changes occur in $B(t)$ is much larger than the time scale at which $Q(t)$ varies. Namely, within the duration that some change occurs in $B(t)$, $Q(t)$ may fluctuate significantly. Technically, for an AWGN

channel with an SNR $\gamma$, this assumption is equivalent to $M \gg \frac{1}{\gamma}(2^{\lambda}-1)K$, i.e., the total amount of energy in the battery is much larger than that required to serve a full data buffer worth of packets. In such a case, we should give priority to adjust the queue state. In other words we need to control $Q(t)$ instead of $B(t)$, i.e., choose $e(t)$ such that the drift of $Q(t)$ is always toward a desired queue state. With these observations, we state the following theorem.

**Theorem 5.3.** *Consider any non-decreasing concave utility function $U_D(\cdot)$ such that $\left|\frac{\partial^2 U_D(e)}{\partial e^2}\right| < \infty$ for all $e > 0$. For any $\lambda < C(\mu)$, given any $\beta_Q \geq 2$, there exists an energy management scheme $\mathcal{Q}$ that achieves a buffer overflow probability $p^{\mathcal{Q}}_{overflow}(K) = O(K^{-\beta_Q})$, battery discharge probability $p^{\mathcal{Q}}_{discharge}(M) = O(\exp(-\alpha_Q M))$ for some $\alpha_Q > 0$ and $U_D(\lambda) - \bar{U}^{\mathcal{Q}} = \Theta\left(\frac{(\log K)^2}{K^2}\right)$ under the large battery regime.*

Theorem 5.3 states that it is possible to have an exponential decay (with $M$) for the battery discharge probability and a quadratic decay (with $K$) for the buffer overflow probability and at the same time achieve a time average utility that approaches the maximum long term value (with an infinite buffer and battery capacities) as $(\log K)^2/K^2$. To preserve continuity we provide an outline for the proof, a full version of which can be found in Appendix A.3. The proof for this theorem is also constructive as we first present the scheme $\mathcal{Q}$, and then derive the performance metrics for this scheme.

Consider the energy management scheme $\mathcal{Q}$, where

$$e^{\mathcal{Q}}(t) = \begin{cases} \mu - \delta_1^{(r)}, & Q(t) \geq K/2 \\ \mu - \delta_2^{(r)}, & Q(t) < K/2 \end{cases}, \tag{5.19}$$

and the drifts $\delta_1^{(r)}$ and $\delta_2^{(r)}$ are chosen to satisfy the relationship $C(\mu - \delta_1^{(r)}) - \lambda = \lambda - C(\mu - \delta_2^{(r)}) = \beta_Q \bar{\sigma}_a^2 \frac{\log K}{K}$. From Fig. 5.4, we note that this choice of energy drifts

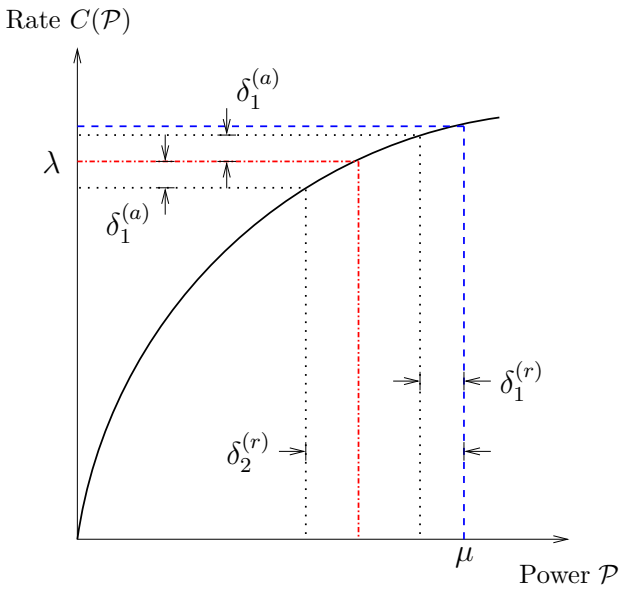



Fig. 5.4: Relation between $\delta_1^{(a)}$, $\delta_1^{(r)}$ and $\delta_2^{(r)}$.

correspond to a queue drift of $|\delta_1^{(a)}| = \beta_Q \bar{\sigma}_a^2 \frac{\ln K}{K}$, toward the state $K/2$, regardless of the queue state $Q(t)$. The queue and battery drifts with scheme $\mathcal{Q}$ are illustrated in Fig. 5.5. We observe that even though the scheme $\mathcal{Q}$ regulates the data queue to a desired state (i.e., $K/2$), the battery is always regulated towards full state (i.e., $M$). State equation for $Q(t)$ is given by,

$$Q(t+1) = \begin{cases} \min\{K, Q(t) + a(t) - \lambda - \delta_1^{(a)}\}, & Q(t) \geq K/2 \\ \max\{0, Q(t) + a(t) - \lambda + \delta_1^{(a)}\}, & Q(t) < K/2 \end{cases}. \tag{5.20}$$

While $p^{\mathcal{Q}}_{\text{overflow}}(K)$ can be found using a method similar to the one used in Theorem 5.2 to calculate $p^{\mathcal{B}}_{\text{discharge}}(M)$, calculation of $p^{\mathcal{Q}}_{\text{discharge}}(M)$ needs careful analysis. More specifically, the battery drift in a particular time slot depends on the data queue state which makes the application of large deviation techniques for calculating

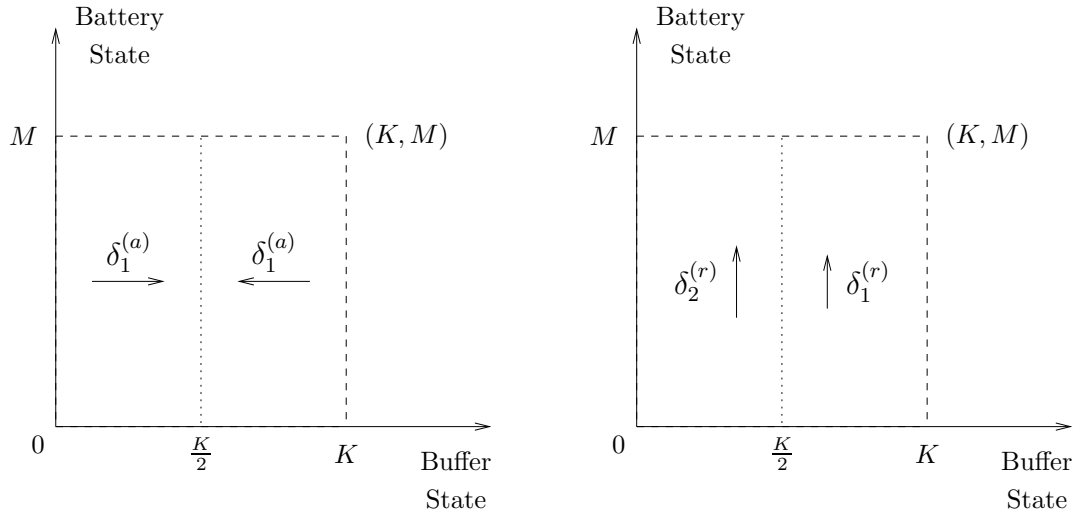



Fig. 5.5: Data queue and battery drifts for scheme $\mathcal{Q}$.

$p^{\mathcal{Q}}_{\text{discharge}}(M)$ difficult. To show these order results, we use a sequence of arrival rates such that $\lambda \uparrow C(\mu)$. In this limit, both the battery and the data queues operate in the *heavy traffic* limit. Consequently, we can apply the *diffusion approximation* to analyze the joint queue and battery process and get a 2-dimensional *reflected Brownian motion* (RBM) [61]. Probability of battery discharge or buffer overflow can be found as the probability of reflection at the appropriate boundary. Closed-form analysis of the stationary distribution for RBMs is not possible except in some special cases given in [62]. Since our model does not fall in that category, the steady state reflection probabilities cannot be derived in closed form. In order to find an analytical solution for the performance metrics, we approximate the system model using a 2-dimensional Brownian motion *without* reflections. The data buffer approximation $\mathbf{Q}(t)$ is given by [61],

$$Q(t) \overset{d}{\approx} \mathbf{Q}(t) = \begin{cases} BM(-\delta_1^{(a)}, \bar{\sigma}_a^2), & \mathbf{Q}(t) \geq K/2 \\ BM(\delta_1^{(a)}, \bar{\sigma}_a^2), & \mathbf{Q}(t) < K/2 \end{cases}, \tag{5.21}$$

and the battery diffusion approximation $\mathbf{B}(t)$ is given by,

$$B(t) \overset{d}{\approx} \mathbf{B}(t) = \begin{cases} BM(\delta_1^{(r)}, \bar{\sigma}_r^2), & \mathbf{Q}(t) \geq K/2 \\ BM(\delta_2^{(r)}, \bar{\sigma}_r^2), & \mathbf{Q}(t) < K/2 \end{cases}. \tag{5.22}$$

Here '$\overset{d}{\approx}$' denotes approximately equal in distribution and $BM(m, \bar{\sigma}^2)$ is a Brownian motion process with mean $mt$ and variance $\bar{\sigma}^2 t$. With the heavy traffic limit, these probabilistic equalities are precise. Referring to Figure 5.5, the buffer overflow probability is given by $p^{\mathcal{Q}}_{\text{overflow}}(K) = \lim_{\tau\to\infty} P(\mathbf{Q}(\tau) > K)$. The battery discharge probability is given by $p^{\mathcal{Q}}_{\text{discharge}}(M) = \lim_{\tau\to\infty} P(\mathbf{B}(\tau) < 0)$. The following lemmas give the scaling laws for these probabilities.

**Lemma 5.4.** *For the energy management scheme $\mathcal{Q}$, given any $\beta_Q \geq 2$, the buffer overflow probability for a buffer of size $K$ is given by $p^{\mathcal{Q}}_{overflow}(K) = O(K^{-\beta_Q})$.*

We provide a brief sketch of the proof, details of which can be found in Appendix A.3. If we assume the starting state of the Brownian motion to be $\mathbf{Q}(0) = K/2$. Note that due to the strong Markovian property of a Brownian motion [63], the instants $\{T_n,\ n = 1, 2, \ldots\}$ at which the system returns to state $K/2$ (i.e., $\mathbf{Q}(T_i) = K/2$) is probabilistically equal to the starting state. Hence we can study these renewal epochs[10] to obtain steady state properties for the data queue process. Now, consider the random variables $T_u \triangleq \arg\min_{t>0}\{\mathbf{Q}(t) = K/2, \mathbf{Q}(0) = K/2, \mathbf{Q}(0^+) > K/2\}$ and $T_l \triangleq \arg\min_{t>0}\{\mathbf{Q}(t) = K/2, \mathbf{Q}(0) = K/2, \mathbf{Q}(0^+) < K/2\}$. From Eq. (5.21), we note that the random variables $T_u$ and $T_l$ will have the same distribution. Furthermore, once the process $\mathbf{Q}(t) = K/2$, it can go above or below $K/2$ with equal probability. Consequently, we need to study the renewals associated with $\mathbf{Q}_u(t) \triangleq$

[10]If we assume the starting state to be $\mathbf{Q}(0) \neq 0$, we can simply consider the process to be a delayed renewal process. The steady state properties in the resulting analysis will not change.

$|\mathbf{Q}(t) - K/2| + K/2$ and can find $\mathrm{P}(\mathbf{Q}(t) > K) = \frac{1}{2}\mathrm{P}(\mathbf{Q}_u(t) > K)$. If we define a unit reward (i.e., $R(t) = 1$) for every time $t$ that the process $\mathbf{Q}_u(t) > K$ then from renewal-reward theory [64],

$$\lim_{t\to\infty} P(\mathbf{Q}_u(t) > K) = \lim_{t\to\infty} \mathrm{E}[R(t)] = \frac{\mathrm{E}[R_n]}{\mathrm{E}[T_u]}. \tag{5.23}$$

Where $\mathrm{E}[R_n]$ is the expected award accumulated in one renewal period, and $\mathrm{E}[T_u]$ is the expected length of the renewal period. Evaluating Eq. (5.23), and noting that the stochastic process will be in the upper half of the buffer with probability 1/2, we have,

$$\lim_{K\to\infty} \frac{1}{K} \log(p^{\mathcal{Q}}_{\text{overflow}}(K)) = -\frac{\delta_1^{(a)}}{\bar{\sigma}_a^2} \tag{5.24}$$

By substituting $\delta_1^{(a)} = \beta_Q \bar{\sigma}_a^2 \frac{\log K}{K}$, we have $p^{\mathcal{Q}}_{\text{overflow}}(K) = \mathrm{O}(K^{-\beta_Q})$.

**Lemma 5.5.** *For the energy management scheme $\mathcal{Q}$, the battery discharge probability for a battery of size $M$ is given by* $\lim_{M\to\infty} \frac{1}{M} \log(p^{\mathcal{Q}}_{discharge}(M)) = -\frac{2(\mu - C^{-1}(\lambda))}{\bar{\sigma}_r^2}$*, where $C^{-1}(\cdot)$ is the inverse of the rate-power function $C(\cdot)$.*

The detailed proof for this Lemma is in Appendix A.3, we provide a brief outline here. First we define,

$$p_A(x) = P(\mathbf{B}(t) \text{ hits } 0 \text{ before } +A | \mathbf{B}(0) = x),$$

i.e., probability of discharge starting with $x$ amount of charge in the battery. The differential equation for $p_A(x)$ can be written as,

$$-\left(\frac{\delta_1^{(r)} + \delta_2^{(r)}}{\bar{\sigma}_r^2}\right) p_A^{(1)}(x) + p_A^{(2)}(x) = 0, \tag{5.25}$$

where $p_A^{(n)}(x) = \frac{\partial^n p_A(x)}{\partial x^n}$. By applying the appropriate boundary conditions, we can solve Eq. (5.25) for $p_A(x)$. Discharge probability can then be found by passing the limit $\lim_{A\to\infty} p_A(M)$,

$$p^{\mathcal{Q}}_{\text{discharge}}(M) = \lim_{A\to\infty} p_A(M) = \exp\left(-\frac{(\delta_1^{(r)} + \delta_2^{(r)})M}{\bar{\sigma}_r^2}\right). \tag{5.26}$$

For a rate-power function $C(\cdot)$, we have $\delta_1^{(a)}+\delta_2^{(a)} = 2(\mu - C^{-1}(\lambda)) + \mathrm{o}(\delta_1^{(a)})$. Substituting this expression in Eq. (5.26), we get the required result. Proof for the convergence of the time average utility is similar to that for Theorem 5.2.

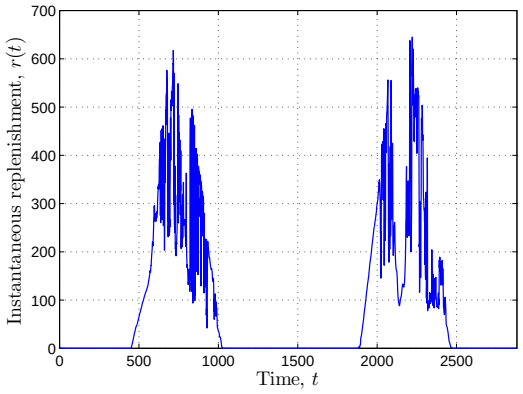
## 5.4 Performance Evaluation

We conduct simulations to evaluate the performance of our energy management schemes. We construct the energy replenishment process $r(t)$ using the real solar radiation measurements collected at the Solar Radiation Research Laboratory [65]. The data set used is the global horizontal radiation or the total solar radiation using a Precision Spectral Pyranometer. We use data from January 1999 to July 2010 collected at 1 minute intervals. We assume that the energy replenishment process is proportional to the total solar radiation. To compare our scheme, we use the Throughput-Optimal (TO) policy given in Eq. (4) of [59]. The TO policy is given by,

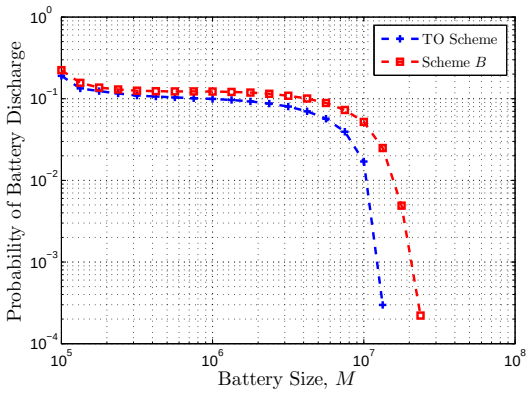
$$e^{TO}(t) = \min\{B(t), \mu - \epsilon\}, \tag{5.27}$$

where $\epsilon$ is a constant such that $C(\mu - \epsilon) > \lambda$. Note that the TO policy will follow the scaling laws given by Theorem 5.1.

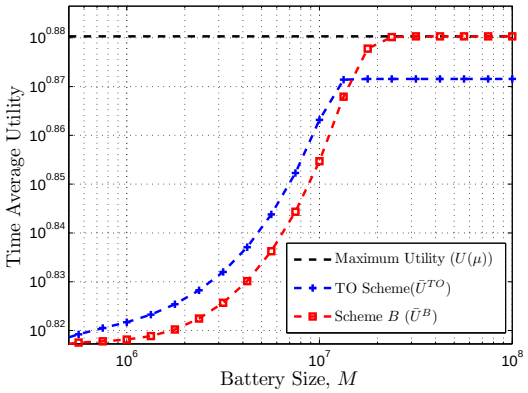
In Fig. 5.6, we revisit the power allocation scheme $\mathcal{B}$ discussed in Example 5.1. The communication channel is AWGN and we choose $\beta = 2$. Fig. 5.6(a) shows a sample of the replenishment process $r(t)$ over a 48 hour period. Note that, in all




(a) Sample of the replenishment process $r(t)$ over a 48 hour period.




(b) Battery discharge probability scaling with battery size $M$.

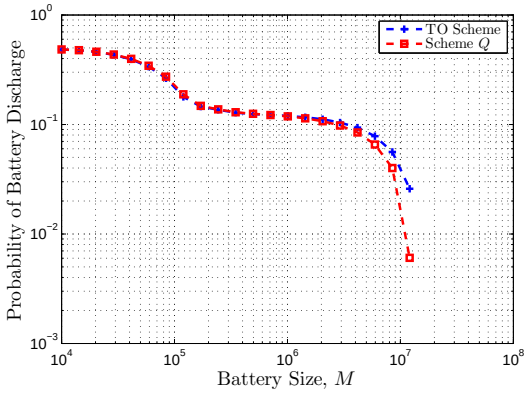



(c) Time average utility scaling with battery size $M$.

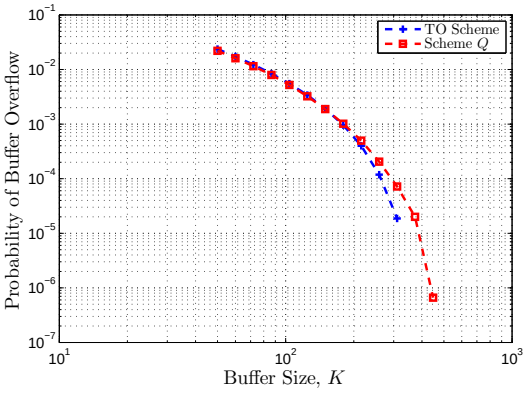
Fig. 5.6: Performance evaluation for the AWGN channel example.

numerical evaluations, we normalize the energy units by the channel noise power $N_0$. From Theorems 5.1 and 5.2 we know that the TO policy should achieve an exponential decay for discharge probability compared to the quadratic decay for scheme $\mathcal{B}$. On the other hand, the TO policy can not achieve the maximum utility while scheme $\mathcal{B}$ should achieve maximum utility as $(\log M)^2/M^2$. Fig. 5.6(b) plots the battery discharge probability as a function of the battery size $M$. As expected, the TO policy performs better than scheme $\mathcal{B}$. However, the advantage of using policy $\mathcal{B}$ is evident in Fig. 5.6(c), which compares the time average utilities achieved by each scheme. It can be seen that, for the choice of parameters and data used in this simulation, scheme $\mathcal{B}$ achieves the maximum utility $U(\mu)$ for a battery size of $10^7$ energy units, whereas the TO scheme does not achieve the maximum utility even asymptotically.

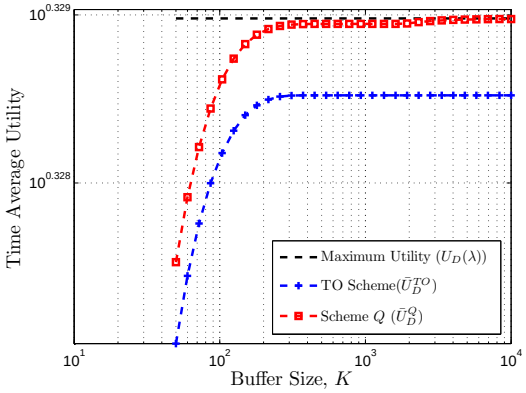
Fig. 5.7 compares the performance of power allocation schemes when both battery and buffer constraints are present. We simulate the data arrival process by generating a Poisson random variable with mean $\lambda = 7.44$ bits in every time slot. The mean of the energy replenishment process $\mu = 191.78$ energy units per time slot and we choose $\beta_Q = 2$. In Fig. 5.7(a), we fix the buffer length to $10^5$ bits and plot the battery discharge probability as a function of the battery size $M$. The discharge probabilities for both schemes should decay exponentially for both schemes. However, the decay exponent for scheme $\mathcal{Q}$ should be larger than the decay exponent for the TO scheme. As expected, the discharge probability for scheme $\mathcal{Q}$ decays faster than that of the TO scheme. In Fig. 5.7(b), we plot the buffer overflow probability as a function of the buffer size while keeping the battery size fixed at $10^7$ energy units. Here we expect the overflow probability for the TO scheme to have an exponential decay compared to a quadratic decay for scheme $\mathcal{Q}$. However, we observe that the overflow probabilities




(a) Battery discharge probability scaling with battery size $M$.




(b) Buffer overflow probability scaling with buffer size $K$.

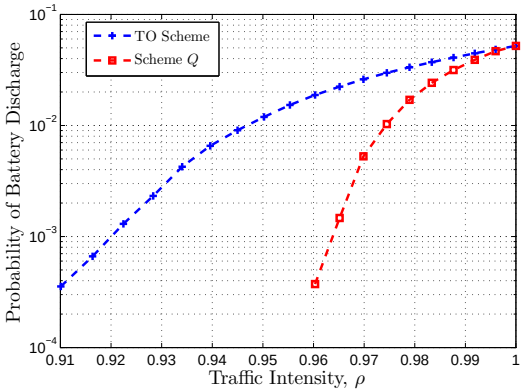



(c) Time average utility scaling with buffer size $K$.

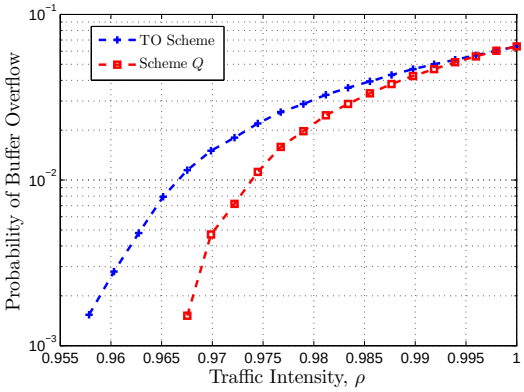
Fig. 5.7: Performance evaluation for energy management schemes under buffer and battery constraints.

for both the schemes decay at approximately the same rate. This anomaly can be explained by noting that the discharge probability of the TO scheme is higher than that of scheme $\mathcal{Q}$. Consequently, there are more service outages for the TO scheme. This offsets the advantage that the TO scheme (with an infinite battery) has in terms of a better buffer overflow probability decay rate. Fig. 5.7(c) compares the convergence of the time average utilities to the maximum utility function for the two schemes. We observe that scheme $\mathcal{Q}$ converges to the maximum utility for moderately large buffer sizes ($\sim 10^3$ bits). On the other hand, TO scheme does not achieve the optimal utility even asymptotically.

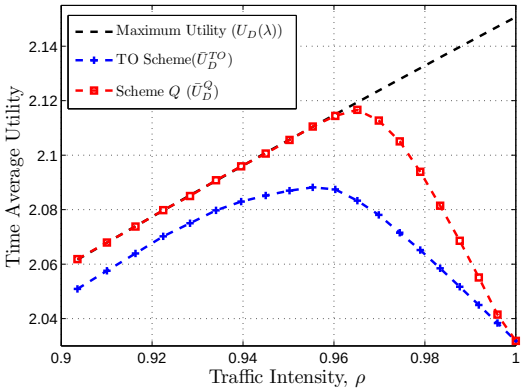
Fig. 5.8 compares the performance of power allocation schemes with increasing traffic intensity. We define traffic intensity as $\rho \triangleq \frac{\lambda}{C(\mu)} = \frac{\lambda}{\log_2(1+\gamma\mu)}$. We fix the buffer length at $10^5$ bits and battery capacity is set at $10^7$ energy units. In Fig. 5.8(a) we observe that the discharge probability increases with traffic intensity. For values of $\rho < 0.97$, scheme $\mathcal{Q}$ performs almost 10 times better than the TO scheme in terms of the discharge probability. For traffic intensities close to unity the scheme $\mathcal{Q}$ degenerates to the TO scheme and their performances converge. Fig. 5.8(b) shows that the buffer overflow probabilities for both schemes also increases with increasing traffic intensity. For values of $\rho < 0.97$, the overflow probability for scheme $\mathcal{Q}$ is almost an order of magnitude lower than that for the TO scheme. Again, this can be explained by the higher discharge probability for the TO scheme leading to severe performance degradation. Finally, we observe in Fig. 5.8(c) that the time average utility decreases with increasing $\rho$. This is a direct consequence of increasing battery discharge and buffer overflow probabilities leading to sub-optimal performance in both




(a) Battery discharge probability scaling with traffic intensity $\rho$.




(b) Buffer overflow probability scaling with traffic intensity $\rho$.




(c) Time average utility scaling with traffic intensity $\rho$.

Fig. 5.8: Performance evaluation of energy management schemes under buffer and battery constraints with increasing traffic intensities.

schemes. Once again, we observe that with $\rho \approx 1$, the performance of the two energy management schemes converge.

# CHAPTER 6

# CONCLUSIONS

## 6.1 Summary of Our Work

An important consideration in the effective performance and deployment of sensor network is the design of efficient energy management schemes. For instance, sensor network applications may require deployment in remote area. Hence, it becomes critical to ensure that the networks are capable of operating unattended for long durations of time. Since battery lifetime limitation plays a major role in determining the sensor node/network lifetime, we focused on developing efficient management of energy resources for sensor networks in this thesis.

We derived the capacity-cost function for energy-efficient transmission schedulers over finite state Markov channels with imperfect feedback. The capacity-cost function found here constitute an upper bound for the achievable throughput of transmission schedulers over finite state Markov channels. Our analysis provides insights into the fundamental value of using ACK/NAK information while scheduling transmissions. For instance, in one example, the maximum power penalty (relative to perfect feedback) of no feedback is 7 dB. The use of ARQ feedback, can potentially reduce this power penalty to less than 1 dB.

With these insights, we introduced a channel aware transmission scheduler to minimize energy expenditure for communication tasks in sensor networks. Our transmission scheduler meets the deadline constraint for all packets waiting in the transmission queue with high probability while optimizing the total energy spent on transmission. To achieve this, we used finite-horizon DP and utilized different types of CSI in the optimization process. We showed that the performance achieved by a DP-based scheduler is close to the fundamental limits for a wide range of channel parameters. Indeed, for some cases, the ACK/NAK scheduler achieves performance that is within 80% of the theoretical bound. Comparisons between the performance of different grades of CSI show that the difference between performance of causal perfect CSI and ACK/NAK schemes for a wide range of channel parameters is not significant. On the other hand, blind transmission scheme in which the ACK/NAK feedback is not exploited to estimate the channel state incurs a significant power penalty.

We also developed a practical channel aware energy management scheme for sensor networks called the pushback algorithm. Here, we used channel estimation to track channel variability and schedule retransmissions accordingly. For a given throughput requirement, the pushback mechanism optimizes the packet success rate leading to improving energy efficiency in a sensor network. Using a simple packet loss model, we developed an implementation of this mechanism that incurred a low computational overhead on the sensor nodes. This algorithm is easy to implement over existing MAC and link layer protocols. The combination of these features make the pushback algorithm highly suitable for energy constrained wireless networks.

Finally, we studied the basic limitations of energy management schemes in energy replenishing sensor networks. We showed that it is possible to observe a quadratic

decay for the discharge probability with increased battery size $M$, and at the same time achieve $\Theta((\log M)^2/M^2)$ convergence to the maximum achievable utility using a simple energy management scheme. The strength of this result can be seen by noting that it is not possible to simultaneously observe an exponential decay for the discharge probability and achieve maximum utility. With the insights drawn, we addressed the problem of energy management with buffer and battery constraints. We showed that, in addition to achieving $\Theta((\log K)^2/K^2)$ convergence, with buffer size $K$, to the optimum utility, it is possible to achieve a quadratic decay for the buffer overflow probability and exponential decay for the battery discharge probability using a simple energy management scheme. To analyze the buffer and battery processes we used large deviations theory and diffusion approximations. The main advantage of using these tools in our work is that it allows analytical tractability while keeping the system model fairly general. We compared our energy management schemes with the Throughput Optimal (TO) scheme, and demonstrated that our scheme can perform upto an order of magnitude better in terms of outage probabilities while achieving the maximum utility asymptotically.

## 6.2 Future Research Directions

In this thesis we have used channel quality prediction to appropriately schedule transmissions. However, channel quality prediction can be used to adjust other parameters such as physical layer data rate, transmission power, and carrier- sense threshold, some of which are inter-related. The existing framework of the dynamic programming algorithm can be modified to incorporate different control actions corresponding to these transmission parameters.

Our transmission scheduler has been designed for a point-to-point link. However, the dynamic programming algorithm can also be applied to networks for multi-user scheduling. One natural extension would be the multi-user downlink case. For a general multi-user case, as long as the decision-making is centralized, some variation of the dynamic programming approach can be applied. However for the distributed decision-making case, a more sophisticated approach employing elements of game theory might be required. All these scenarios would be interesting research topics for the future.

We have focused on the application of the concept of pushbacks in the context of sensor networks. However, pushbacks, when applied to other networking scenarios such as ad hoc networks and mesh networks, can be used to optimize other parameters such as the number of transmissions. In these networks, reduced interference due to reduction in number of transmission is expected to result in increased throughput.

Future directions of research in energy replenishing networks include the design of optimal or near-optimal cross-layer distributed energy management solutions in the presence of channel fading and multi-user interference.

# APPENDIX A

# PROOFS FOR CHAPTER 5

## A.1 Proof of Lemma 5.1

This lemma gives the rate of decay of the probability of complete discharge with respect to the battery size $M$. To prove this result, we first find an *upper bound* for the required probability. We fix a constant $A > 0$ and decompose the time line into intervals, such that each interval is of length $\lceil \frac{M}{A} \rceil$ and the $i$th interval ends at time slot $t_i = i\lceil \frac{M}{A} \rceil$. Next, define $E_i$ as the event that the battery is empty at the end of time slot 0 and the last time the battery was full (i.e., $M$) is some time during the interval $-i = \left[-(i+1)\lceil \frac{M}{A} \rceil + 1, -i\lceil \frac{M}{A} \rceil\right]$. The event of an empty battery at time slot 0 can be decomposed as a union of events $E_i$,

$$p^{\mathcal{S}}_{\text{discharge}}(M) = \sum_{i=0}^{\infty} \mathrm{P}\left(E_i\right) \tag{A.1}$$

A necessary condition for event $E_i$ to occur is,

$$\sum_{t=-(i+1)\lceil \frac{M}{A} \rceil + 1}^{0} \left(e^{\mathcal{S}}(t) - r(t)\right) > M. \tag{A.2}$$

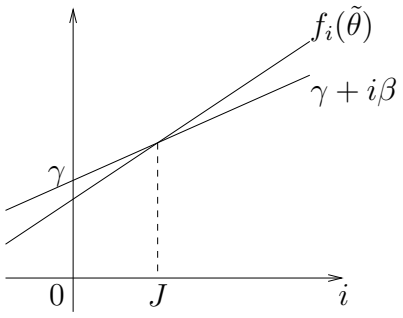



Fig. A.1: A geometric proof for the existence of $K$ and $\delta > 0$ such that for every $i > K$, $f_i(\tilde{\theta}) > \gamma + i\delta$.

Using Chernoff's bound, for any $\theta_i \geq 0$,

$$\begin{aligned}
&\mathrm{P}\left(\sum_{t=-(i+1)\lceil\frac{M}{A}\rceil+1}^{0} d^{\mathcal{S}}(t) > M\right) \\
&\leq \mathrm{E}\left[\exp\left(\theta_i \sum_{t=-(i+1)\lceil\frac{M}{A}\rceil+1}^{0} d^{\mathcal{S}}(t)\right)\right] \exp\left(-\theta_i M\right) \\
&= \exp\left(-M\left(\theta_i - \frac{i+1}{A}\bar{\Lambda}_{d^{\mathcal{S}}}(\theta_i) + \epsilon_i(M,\theta_i)\right)\right), \qquad \text{(A.3)}
\end{aligned}$$

where $\epsilon_i(M,\theta_i) \to 0$ as $M \to \infty$.

In order to find the tightest bound for each $i$, we choose $\theta_i^* \geq 0$ to maximize $f_i(\theta) \triangleq \theta - \frac{i+1}{A}\bar{\Lambda}_{d^{\mathcal{S}}}(\theta)$, and define $\gamma = \inf_{i\geq 0}\sup_{\theta\geq 0} f_i(\theta)$. We can rewrite $f_i(\theta)$ as,

$$f_i(\theta) = \theta - \frac{\bar{\Lambda}_{d^{\mathcal{S}}}(\theta)}{A} - i\frac{\bar{\Lambda}_{d^{\mathcal{S}}}(\theta)}{A}.$$

Since $\lim_{\tau\to\infty} \mathrm{E}\left[d^{\mathcal{S}}(\tau)\right] < 0$, the function $\bar{\Lambda}_{d^{\mathcal{S}}}(\theta)$ has a negative slope at $\theta = 0$. Hence, we can choose some $\tilde{\theta} > 0$, such that $\bar{\Lambda}_{d^{\mathcal{S}}}(\tilde{\theta}) < 0$. This implies that there exists a $J$ and a $\beta > 0$ such that for every $i > J$ (see Fig. A.1 for a graphical proof),

$$f_i(\tilde{\theta}) > \gamma + i\beta. \qquad \text{(A.4)}$$

Returning to Eq. (A.1),

$$
\begin{aligned}
p^{\mathcal{S}}_{\text{discharge}}(M) &= \sum_{i=0}^{\infty} \mathrm{P}\left(E_i\right) \\
&\leq \sum_{i=0}^{\infty} \mathrm{P}\left(\sum_{t=-(i+1)\lceil \frac{M}{A} \rceil + 1}^{0} d^{\mathcal{S}}(t) > M\right) \\
&\leq \sum_{i=0}^{J} \exp\left(-M\left(f_i(\theta_i^*) + \epsilon_i(M, \theta_i^*)\right)\right) \\
&\quad + \sum_{i=J+1}^{\infty} \exp\left(-M\left(f_i(\tilde{\theta}) + \epsilon_i(M, \tilde{\theta})\right)\right) \\
&\leq \sum_{i=0}^{J} \exp\left(-M\left(\gamma + \min_{0 \leq i \leq J} \epsilon(M, \theta_i^*)\right)\right) \\
&\quad + \sum_{i=J+1}^{\infty} \exp\left(-M\left(\gamma + i\beta + \inf_{i>J} \epsilon_i(M, \tilde{\theta})\right)\right) \\
&= \exp\left(-M\gamma\right)\left[(J+1)\exp\left(\min_{0 \leq i \leq J} \epsilon(M, \theta_i^*)\right)\right. \\
&\quad \left. + \frac{\exp\left(-M\left((J+1)\beta + \inf_{i>J} \epsilon_i(M, \tilde{\theta})\right)\right)}{1 - \exp\left(-\beta M\right)}\right].
\end{aligned} \tag{A.5}
$$

As $M \to \infty$,

$$
\limsup_{M \to \infty} \frac{1}{M} \log p^{\mathcal{S}}_{\text{discharge}}(M) \leq -\gamma \tag{A.6}
$$

Since this inequality holds for any $A > 0$, we let $A \to \infty$ and get,

$$
\begin{aligned}
\limsup_{M \to \infty} \frac{1}{M} \log p^{\mathcal{S}}_{\text{discharge}}(M) &\leq -\inf_{i \geq 0} \sup_{\theta \geq 0} \left[\theta - \frac{i}{A} \bar{\Lambda}_{d^{\mathcal{S}}}(\theta)\right] \\
&= -\inf_{T \geq 0} \sup_{\theta \geq 0} \left[\theta - T\bar{\Lambda}_{d^{\mathcal{S}}}(\theta)\right] \\
&= -\inf_{T \geq 0} T \sup_{\theta \geq 0} \left[\frac{\theta}{T} - \bar{\Lambda}_{d^{\mathcal{S}}}(\theta)\right].
\end{aligned} \tag{A.7}
$$

Next, we find the *lower bound*. For some $T \geq 0$, a sufficient condition for the battery to be empty at some time slot in the interval $[-\lceil TM \rceil, 0]$ is that,

$$\sum_{t=-\lceil TM \rceil+1}^{0} (e^{\mathcal{S}}(t) - r(t)) > M. \tag{A.8}$$

We can lower bound $p^{\mathcal{S}}_{\text{discharge}}(M)$ using the union bound,

$$\begin{aligned} & p^{\mathcal{S}}_{\text{discharge}}(M)\lceil TM \rceil \\ & \geq \mathrm{P}\left(\text{battery is empty in some slot during } [-\lceil TM \rceil, 0]\right). \end{aligned} \tag{A.9}$$

We define, $Z_{M,T} \triangleq \frac{1}{TM} \sum_{t=-\lceil TM \rceil+1}^{0} d^{\mathcal{S}}(t)$. Consequently,

$$\mathrm{P}\left( \sum_{t=-\lceil TM \rceil+1}^{0} d^{\mathcal{S}}(t) > M \right) = \mathrm{P}\left( Z_{M,T} > \frac{1}{T} \right).$$

Now, $\lim_{M\to\infty} \mathrm{E}\left[Z_{M,T}\right] < 0 < \frac{1}{T}$ for all $T > 0$. Applying the Gärtner-Ellis Theorem, we get,

$$\begin{aligned} \lim_{M\to\infty} \frac{1}{M} \log \mathrm{P}\left( Z_{M,T} > \frac{1}{T} \right) &= -\sup_{s\geq 0} \left[ \frac{s}{T} - \bar{\Lambda}_Z(s) \right] \\ &= -T \sup_{s\geq 0} \left[ \frac{s}{T^2} - \bar{\Lambda}_{d^{\mathcal{S}}}\left(\frac{s}{T}\right) \right] \\ &= -T \sup_{\theta\geq 0} \left[ \frac{\theta}{T} - \bar{\Lambda}_{d^{\mathcal{S}}}(\theta) \right], \end{aligned} \tag{A.10}$$

where $\bar{\Lambda}_Z(s)$ is the asymptotic semi-invariant log-moment generating function of $Z_{M,T}$. By noting that Eq. (A.10) holds for all $T \geq 0$ and combining it with Eq. (A.9), we have,

$$\liminf_{M\to\infty} \frac{1}{M} \log p^{\mathcal{S}}_{\text{discharge}}(M) \geq -\inf_{T\geq 0} T \sup_{\theta\geq 0} \left[ \frac{\theta}{T} - \bar{\Lambda}_{d^{\mathcal{S}}}(\theta) \right]. \tag{A.11}$$

From Eqs. (A.7) and (A.11) we have,

$$\begin{aligned} \lim_{M\to\infty} \frac{1}{M} \log p^{\mathcal{S}}_{\text{discharge}}(M) &= -\inf_{T\geq 0} T \sup_{\theta\geq 0} \left[ \frac{\theta}{T} - \bar{\Lambda}_{d^{\mathcal{S}}}(\theta) \right] \\ &= -s^*_{d^{\mathcal{S}}}. \end{aligned} \tag{A.12}$$

This gives us $p^{\mathcal{S}}_{\text{discharge}}(M) = \Theta(\exp(-s^*_{d^{\mathcal{S}}} M))$.

## A.2 Proof of Lemma 5.2

Fix a constant $A > 0$ and decompose the time line into intervals, such that each interval is of length $\lceil \frac{M}{2A} \rceil$ and the $i$th interval ends at time slot $t_i = i\lceil \frac{M}{2A} \rceil$. Assume that the system is in steady state. We define $E_i$ as the event that the battery is empty at the end of time slot 0 and the last time the battery was half full (i.e., $M/2$) is some time during the interval $-i = \left[-(i+1)\lceil \frac{M}{2A} \rceil + 1, -i\lceil \frac{M}{2A} \rceil\right]$. The event of an empty battery at time slot 0 can be decomposed as a union of events $E_i$,

$$p^{\mathcal{B}}_{\text{discharge}}(M) = \sum_{i=0}^{\infty} \mathrm{P}\left(E_i\right) \tag{A.13}$$

A necessary condition for event $E_i$ to occur is,

$$\sum_{t=-(i+1)\lceil \frac{M}{2A} \rceil + 1}^{0} \left(e^{\mathcal{B}}(t) - r(t)\right) > \frac{M}{2} \tag{A.14}$$

Using Chernoff's bound, for any $\theta_i \geq 0$,

$$\begin{aligned}
&\mathrm{P}\left(\sum_{t=-(i+1)\lceil \frac{M}{2A} \rceil + 1}^{0} (e^{\mathcal{B}}(t) - r(t)) > \frac{M}{2}\right) \\
&\leq \mathrm{E}\left[\exp\left(\theta_i \sum_{t=-(i+1)\lceil \frac{M}{2A} \rceil + 1}^{0} (e^{\mathcal{B}}(t) - r(t))\right)\right] \exp\left(-\theta_i \frac{M}{2}\right) \\
&= \mathrm{E}\left[\exp\left(-\theta_i \sum_{t=-(i+1)\lceil \frac{M}{2A} \rceil + 1}^{0} r(t)\right)\right] \\
&\quad \times \exp\left(\theta_i (i+1)\left\lceil \frac{M}{2A} \right\rceil (\mu - \delta^-)\right) \exp\left(-\theta_i \frac{M}{2}\right) \\
&= \exp\Bigg( - \frac{M}{2}\Bigg[\theta_i \left(1 - \frac{i+1}{A}(\mu - \delta^-)\right) - \frac{i+1}{A}\bar{\Lambda}_r(-\theta_i) \\
&\qquad + \epsilon_i(M, \theta_i)\Bigg]\Bigg),
\end{aligned} \tag{A.15}$$

where $\epsilon_i(M, \theta_i) \to 0$ as $M \to \infty$.

In order to find the tightest bound for each $i$, we choose $\theta_i^* \geq 0$ to maximize,

$$f_i(\theta) \triangleq \theta \left(1 - \frac{i+1}{A}(\mu - \delta^-)\right) - \frac{i+1}{A}\bar{\Lambda}_r(-\theta), \tag{A.16}$$

and let $\gamma = \inf_{i\geq 0} \sup_{\theta \geq 0} f_i(\theta)$. We can rewrite $f_i(\theta)$ as,

$$f_i(\theta) = \theta - \frac{\mu - \delta^-}{A}\theta - \frac{\bar{\Lambda}_r(-\theta)}{A} - i\left(\frac{(\mu - \delta^-)\theta + \bar{\Lambda}_r(-\theta)}{A}\right)$$

Since $\lim_{\tau\to\infty} \frac{1}{\tau}\sum_{t=1}^{\tau} \mathrm{E}\left[r(t)\right] = \mu > \mu - \delta^-$, the function $(\mu - \delta^-)\theta + \bar{\Lambda}_r(-\theta)$ has a negative slope at $\theta = 0$. Hence, we can choose some $\tilde{\theta} > 0$, such that $(\mu - \delta^-)\tilde{\theta} + \bar{\Lambda}_r(-\tilde{\theta}) < 0$. This implies that there exists a $J$ and a $\beta > 0$ such that for every $i > J$,

$$f_i(\tilde{\theta}) > \gamma + i\beta. \tag{A.17}$$

Returning to Eq. (A.13),

$$\begin{aligned}
p^{\mathcal{B}}_{\text{discharge}}(M) &= \sum_{i=0}^{\infty} \mathrm{P}\left(E_i\right) \\
&\leq \sum_{i=0}^{\infty} \mathrm{P}\left(\sum_{k=-(i+1)\lceil\frac{M}{2A}\rceil+1}^{0} (e^{\mathcal{B}}(k) - r(k)) > \frac{M}{2}\right) \\
&\leq \sum_{i=0}^{J} \exp\left(-\frac{M}{2}\left[f_i(\theta_i^*) + \epsilon_i(M,\theta_i^*)\right]\right) \\
&\quad + \sum_{i=J+1}^{\infty} \exp\left(-\frac{M}{2}\left[f_i(\tilde{\theta}) + \epsilon_i(M,\tilde{\theta})\right]\right) \\
&\leq \sum_{i=0}^{J} \exp\left(-\frac{M}{2}\left[\gamma + \min_{0\leq i\leq J} \epsilon(M,\theta_i^*)\right]\right) \\
&\quad + \sum_{i=J+1}^{\infty} \exp\left(-\frac{M}{2}\left[\gamma + i\beta + \inf_{i>J} \epsilon_i(M,\tilde{\theta})\right]\right) \\
&= \exp\left(-\frac{M}{2}\gamma\right)\left[(J+1)\exp\left(\min_{0\leq i\leq J} \epsilon(M,\theta_i^*)\right)\right. \\
&\quad \left. + \frac{\exp\left(-\frac{M}{2}\left((J+1)\beta + \inf_{i>J} \epsilon_i(M,\tilde{\theta})\right)\right)}{1 - \exp\left(-\beta\frac{M}{2}\right)}\right].
\end{aligned} \tag{A.18}$$

As $M \to \infty$,

$$\limsup_{M\to\infty} \frac{2}{M} \log p^{\mathcal{B}}_{\text{discharge}}(M) \leq -\gamma$$

Since this inequality holds for any $A > 0$, we let $A \to \infty$ and get,

$$\begin{aligned}
&\limsup_{M\to\infty} \frac{2}{M} \log p^{\mathcal{B}}_{\text{discharge}}(M) \\
&\quad \leq -\inf_{i\geq 0} \sup_{\theta\geq 0} \left[\theta\left(1 - \frac{i}{A}(\mu-\delta)\right) - \frac{i}{A}\bar{\Lambda}_r(-\theta)\right] \\
&\quad = -\inf_{T\geq 0} \sup_{\theta\geq 0} \left[\theta\left(1 - T(\mu-\delta)\right) - T\bar{\Lambda}_r(-\theta)\right] \\
&\quad = -\inf_{T\geq 0} T \sup_{\theta\geq 0} \left[-\theta\left(\mu - \delta - \frac{1}{T}\right) - \bar{\Lambda}_r(-\theta)\right] \qquad \text{(A.19)}
\end{aligned}$$

Next, we find the lower bound. For some $T \geq 0$, a sufficient condition for the battery to be empty at some time slot in the interval $[-\lceil TM/2\rceil, 0]$ is that,

$$\sum_{t=-\lceil \frac{TM}{2}\rceil+1}^{0} (e^{\mathcal{B}}(t) - r(t)) > M. \qquad \text{(A.20)}$$

We can lower bound $p^{\mathcal{B}}_{\text{discharge}}(M)$ using the union bound,

$$\begin{aligned}
&p^{\mathcal{B}}_{\text{discharge}}(M) \left\lceil \frac{TM}{2} \right\rceil \\
&\geq \mathrm{P}\left(\text{battery is empty in some slot during } \left[-\left\lceil \frac{TM}{2}\right\rceil, 0\right]\right). \qquad \text{(A.21)}
\end{aligned}$$

We can write,

$$\begin{aligned}
&\mathrm{P}\left(\sum_{t=-\lceil \frac{TM}{2}\rceil+1}^{0} (e^{\mathcal{B}}(t) - r(t)) > \frac{M}{2}\right) \\
&\qquad = \mathrm{P}\left(\sum_{t=-\lceil \frac{TM}{2}\rceil+1}^{0} (\mu - \delta^- - r(t)) > \frac{M}{2}\right). \qquad \text{(A.22)}
\end{aligned}$$

We define, $Z_{M,T} \triangleq \frac{2}{TM}\sum_{k=-\lceil \frac{TM}{2}\rceil+1}^{0} (\mu - \delta^- - r(k))$. Consequently,

$$\mathrm{P}\left(\sum_{t=-\lceil \frac{TM}{2}\rceil+1}^{0} (\mu - \delta^- - r(t)) > \frac{M}{2}\right) = \mathrm{P}\left(Z_{M,T} > \frac{1}{T}\right).$$

Now, $\lim_{M\to\infty} \mathrm{E}\left[Z_{M,T}\right] = -\delta^- < 0 < \frac{1}{T}$ for all $T > 0$. Applying the Gärtner-Ellis Theorem, we get,

$$\begin{aligned}
&\lim_{M\to\infty} \frac{2}{M} \log \mathrm{P}\left(Z_{M,T} > \frac{1}{T}\right) \\
&\quad = -\sup_{s\geq 0}\left[\frac{1}{T}s - s\left(\mu-\delta\right) + T\bar{\Lambda}_r\left(-\frac{s}{T}\right)\right] \\
&\quad = -T\sup_{s\geq 0}\left[-\frac{s}{T}\left(\mu-\delta-\frac{1}{T}\right) - \bar{\Lambda}_r\left(-\frac{s}{T}\right)\right] \\
&\quad = -T\sup_{\theta\geq 0}\left[-\theta\left(\mu-\delta-\frac{1}{T}\right) - \bar{\Lambda}_r(-\theta)\right]. \qquad \text{(A.23)}
\end{aligned}$$

Combining Eqs. (A.21) and (A.23), we have,

$$\begin{aligned}
&\liminf_{M\to\infty} \frac{2}{M} \log p^{\mathcal{B}}_{\text{discharge}}(M) \\
&\quad \geq -\inf_{T\geq 0} T\sup_{\theta\geq 0}\left[-\theta\left(\mu-\delta-\frac{1}{T}\right) - \bar{\Lambda}_r(-\theta)\right]. \qquad \text{(A.24)}
\end{aligned}$$

From Eqs. (A.19) and (A.24) we have,

$$\begin{aligned}
&\lim_{M\to\infty} \frac{2}{M} \log p^{\mathcal{B}}_{\text{discharge}}(M) \\
&\quad = -\inf_{T\geq 0} T\sup_{\theta\geq 0}\left[-\theta\left(\mu-\delta-\frac{1}{T}\right) - \bar{\Lambda}_r(-\theta)\right] \\
&\quad = s^*_{d^-}. \qquad \text{(A.25)}
\end{aligned}$$

This gives us $p^{\mathcal{B}}_{\text{discharge}}(M) = \Theta\left(\exp\left(s^*_{d^-}\frac{M}{2}\right)\right)$.

## A.3 Proof of Theorem 5.3

As discussed in Section 5.3.1, the buffer overflow probability is given by $p^{\mathcal{Q}}_{\text{overflow}}(K) = \lim_{\tau\to\infty} P(\mathbf{Q}(\tau) > K)$ and the battery underflow probability is given by $p^{\mathcal{Q}}_{\text{discharge}}(M) = \lim_{\tau\to\infty} P(\mathbf{B}(\tau) < 0)$. Under the heavy traffic condition, taking the diffusion limit of the queue process, we get the data queue approximation,

$$\mathbf{Q}(t) = \begin{cases} BM(-\delta_1^{(a)}, \bar{\sigma}_a^2), & \mathbf{Q}(t) \geq K/2 \\ BM(\delta_1^{(a)}, \bar{\sigma}_a^2), & \mathbf{Q}(t) < K/2 \end{cases}, \qquad \text{(A.26)}$$

and the battery approximation,

$$\mathbf{B}(t) = \begin{cases} BM(\delta_1^{(r)}, \bar{\sigma}_r^2), & \mathbf{Q}(t) \geq K/2 \\ BM(\delta_2^{(r)}, \bar{\sigma}_r^2), & \mathbf{Q}(t) < K/2 \end{cases}. \tag{A.27}$$

Here $BM(m, \bar{\sigma}^2)$ is a Brownian motion process with mean $mt$ and variance $\bar{\sigma}^2 t$. For completeness, we re-state the lemmas used for the proof of Theorem 5.3.

**Lemma 5.4.** *For the energy management scheme $\mathcal{Q}$, given any $\beta_Q \geq 2$, the buffer overflow probability for a buffer of size $K$ is given by $p^{\mathcal{Q}}_{overflow}(K) = O(K^{-\beta_Q})$.*

*Proof.* We focus on the component Brownian motions of the process $\mathbf{Q}(t)$ given in Eq. (A.26). Here we note that the only difference between the two component processes is that their drifts are in opposite directions. If we assume the starting state of the queue process to be $\mathbf{Q}(0) = K/2$. Note that due to the strong Markovian property of a Brownian motion [63], the instants[11] $\{T_n, n > 0\}$ at which the system returns to state $\mathbf{Q}(T_n) = K/2$ is probabilistically equal to the starting state $\mathbf{Q}(0)$. Hence we can study these renewal epochs[12] $\{T_n,\ n = 1, 2, \cdots\}$ to obtain steady state properties for the data queue process. Now, consider the random variables $T_u \triangleq \arg\min_{t>0}\{\mathbf{Q}(t) = K/2, \mathbf{Q}(0) = K/2, \mathbf{Q}(0^+) > K/2\}$ and $T_l \triangleq \arg\min_{t>0}\{\mathbf{Q}(t) = K/2, \mathbf{Q}(0) = K/2, \mathbf{Q}(0^+) < K/2\}$. From Eq. (A.26), we note that the random variables $T_u$ and $T_l$ will have the same distribution. Furthermore, at renewal epochs $t = T_n$, the process can go above or below $K/2$ with equal probability. Consequently, we need to study the renewals associated with $\mathbf{Q}_u(t) \triangleq |\mathbf{Q}(t) - K/2| + K/2$. Once the process reaches $\mathbf{Q}(t) = K/2$, it can go above or below $\mathbf{Q}(t)$ with equal probability. Consequently, we need to study only

[11] By definition, we let $T_0 = 0$

[12] If we assume the starting state to be $\mathbf{Q}(0) \neq 0$, we can simply consider the process to be a delayed renewal process. The steady state properties in the resulting analysis will not change.

$\mathbf{Q}_u(t)$ and can find $\mathrm{P}\left(\mathbf{Q}(t) > K\right) = \frac{1}{2}\mathrm{P}\left(\mathbf{Q}_u(t) > K\right)$. If we define a unit reward (i.e., $R(t) = 1$) for every time $t$ that the process $\mathbf{Q}_u(t) > K$ then,

$$\lim_{t\to\infty} P(\mathbf{Q}_u(t) > K) = \lim_{t\to\infty} \mathrm{E}\left[R(t)\right]. \tag{A.28}$$

From renewal-reward theory [64] we can write,

$$\lim_{t\to\infty} \mathrm{E}\left[R(t)\right] = \frac{\mathrm{E}\left[R_n\right]}{\mathrm{E}\left[X\right]}, \tag{A.29}$$

where $\mathrm{E}\left[R_n\right]$ is the expected award accumulated in one renewal period, and $\mathrm{E}\left[X\right]$ is the expected length of the renewal period. To get a non-trivial expression for $\mathrm{E}\left[X\right]$, we define $\mathrm{E}\left[X\right]$ as the expected time for process $\mathbf{Q}(t)$ to return to $K/2$ given that it starts at $K/2 + \epsilon$ and then pass the limit $\epsilon \downarrow 0$. The expression for $\mathrm{E}\left[X\right]$ is given by [66],

$$\mathrm{E}\left[X\right] = \lim_{\epsilon\downarrow 0} \frac{\epsilon}{\delta_1^{(a)}}. \tag{A.30}$$

Starting at $K/2 + \epsilon$, the probability of reaching $K$ before $K/2$ will give the expected reward accumulated in one renewal period. Applying the expression for this probability from [66],

$$\begin{aligned}
\mathrm{E}\left[R_n\right] &= \lim_{\epsilon\downarrow 0} \frac{\exp\left(\frac{2\delta_1^{(a)}}{\bar{\sigma}_a^2}\epsilon\right) - 1}{\exp\left(\frac{2\delta_1^{(a)}}{\bar{\sigma}_a^2}\frac{K}{2}\right) - 1} \\
&= \lim_{\epsilon\downarrow 0} \frac{\frac{2\delta_1^{(a)}}{\bar{\sigma}_a^2}\epsilon + o(\epsilon)}{\exp\left(\frac{2\delta_1^{(a)}}{\bar{\sigma}_a^2}\frac{K}{2}\right) - 1}.
\end{aligned} \tag{A.31}$$

Plugging Eqs. (A.30) and (A.31) in Eq. (A.29) we have,

$$\lim_{t\to\infty} \mathrm{E}\left[R(t)\right] = \lim_{\epsilon\downarrow 0} \frac{\left(\frac{2\delta_1^{(a)}}{\bar{\sigma}_a^2} + \frac{o(\epsilon)}{\epsilon}\right)\delta_1^{(a)}}{\exp\left(\frac{2\delta_1^{(a)}}{\bar{\sigma}_a^2}\frac{K}{2}\right) - 1}. \tag{A.32}$$

Evaluating the limit $\epsilon \downarrow 0$, and noting that the stochastic process will be in the upper half of the buffer with probability 1/2, for large $K$ we have,

$$p^{\mathcal{Q}}_{\text{overflow}}(K) = \frac{{\delta_1^{(a)}}^2}{\bar{\sigma}_a^2} \exp\left(-\frac{\delta_1^{(a)} K}{\bar{\sigma}_a^2}\right)$$
$$= \mathrm{O}\left(\exp\left(-\frac{\delta_1^{(a)} K}{\bar{\sigma}_a^2}\right)\right). \tag{A.33}$$

By choosing $\delta_1^{(a)} = \beta_D \bar{\sigma}_a^2 \frac{\ln K}{K}$, we have $p^{\mathcal{Q}}_{\text{overflow}}(K) = \mathrm{O}(K^{-\beta_D})$. □

**Lemma 5.5.** *For the energy management scheme $\mathcal{Q}$, the battery discharge probability for a battery of size $M$ is given by* $\lim_{M\to\infty} \frac{1}{M} \log(p^{\mathcal{Q}}_{discharge}(M)) = -\frac{2(\mu - C^{-1}(\lambda))}{\bar{\sigma}_r^2}$, *where $C^{-1}(\cdot)$ is the inverse of the rate-power function $C(\cdot)$.*

*Proof.* Next, we focus our attention on $p^{\mathcal{Q}}_{\text{discharge}}(M)$ or $\lim_{t\to\infty} P(\mathbf{B}(t) < 0)$. First we define,

$$p_A(x) = P(\mathbf{B}(t) \text{ hits 0 before } +A | \mathbf{B}(0) = x),$$

i.e., probability of discharge starting with $x$ amount of charge in the battery. Discharge probability would be given by $\lim_{A\to\infty} p_A(x)$. If we define $Y = \mathbf{B}(h) - \mathbf{B}(0)$, we can write,

$$p_A(x) = \mathrm{E}_Y\left[p_A(x+Y)\right] + o(h),$$

where $o(h)$ refers to the probability that the process would have hit one of the barriers by time $h$. Using the Taylor series expansion of $p_A(x+Y)$ about $x$, we get,

$$p_A(x) = \mathrm{E}_Y\left[p_A(x) + p_A^{(1)}(x)Y + p_A^{(2)}(x)\frac{Y^2}{2} + \cdots\right] + o(h).$$

Under the diffusion approximation, we observe that $Y \sim \mathcal{N}(\delta_1^{(r)}h, \bar{\sigma}_r^2 h)$ with probability 1/2 and $Y \sim \mathcal{N}(\delta_2^{(r)}h, \bar{\sigma}_r^2 h)$ with probability 1/2. So we can write Eq. (A.34)

as,

$$p_A(x) = p_A(x) + \frac{1}{2}p_A^{(1)}(x)(\delta_1^{(r)} + \delta_2^{(r)}) + p_A^{(2)}(x)\frac{{\delta_1^{(r)}}^2h^2 + {\delta_2^{(r)}}^2h^2 + 2\bar{\sigma}_r^2h}{4} + o(h), \tag{A.34}$$

which can be rewritten as the differential equation (passing the limit $h \to 0$),

$$-\left(\frac{\delta_1^{(r)} + \delta_2^{(r)}}{\bar{\sigma}_r^2}\right)p_A^{(1)}(x) + p_A^{(2)}(x) = 0. \tag{A.35}$$

Integrating the differential equation (A.35) twice we have,

$$p_A(x) = C_1 + C_2\exp\left(-\frac{\delta_1^{(r)} + \delta_2^{(r)}}{\bar{\sigma}_r^2}x\right). \tag{A.36}$$

Solving for the constants $C_1$ and $C_2$ using the initial conditions, $p_A(0) = 1$ and $P_A(A) = 0$, we get,

$$p_A(x) = \frac{\exp\left(-\frac{\delta_1^{(r)}+\delta_2^{(r)}}{\bar{\sigma}_r^2}x\right) - \exp\left(-\frac{\delta_1^{(r)}+\delta_2^{(r)}}{\bar{\sigma}_r^2}A\right)}{1 - \exp\left(-\frac{\delta_1^{(r)}+\delta_2^{(r)}}{\bar{\sigma}_r^2}A\right)}. \tag{A.37}$$

To find the discharge probability, we set $x = M/2$ and pass the limit $A \to \infty$,

$$p_{\text{discharge}}^{\mathcal{Q}}(M) = \lim_{A\to\infty} p_A(M) = \exp\left(-\frac{(\delta_1^{(\mathrm{r})} + \delta_2^{(\mathrm{r})})M}{\bar{\sigma}_r^2}\right). \tag{A.38}$$

For a rate-power function $C(\cdot)$, we have $\delta_1^{(a)} + \delta_2^{(a)} = 2(\mu - C^{-1}(\lambda)) + \mathrm{o}(\delta_1^{(a)})$. Substituting this expression in Eq. (A.38), we get the required result. □

To illustrate Lemma 5.5, we consider the AWGN channel capacity given in Eq. (5.17). If we assume $\gamma = 1$, $\delta_1^{(r)} = \mu - \frac{\exp\left(\left(\lambda+\delta_1^{(a)}\right)\ln 2\right)-1}{\gamma}$ and $\delta_2^{(r)} = \mu - \frac{\exp\left(\left(\lambda-\delta_1^{(a)}\right)\ln 2\right)-1}{\gamma}$. We can use the power series expansion of the exponential function to get,

$$\delta_1^{(r)} = \mu - \frac{1}{\gamma}\left((\lambda + \delta_1^{(a)})\ln 2 + \frac{(\lambda + \delta_1^{(a)})^2(\ln 2)^2}{2} + \cdots\right),$$

and,

$$\delta_2^{(r)} = \mu - \frac{1}{\gamma}\left((\lambda - \delta_1^{(a)})\ln 2 + \frac{(\lambda - \delta_1^{(a)})^2(\ln 2)^2}{2} + \cdots\right).$$

Combining these expressions, we get,

$$\delta_1^{(r)} + \delta_2^{(r)} = 2\mu - \frac{1}{\gamma}\left(\lambda \ln 2 + \Theta(1) + \mathrm{o}(\delta_1^{(a)})\right) = \Theta(1). \tag{A.39}$$

Substituting Eq. (A.39) in Eq. (A.38), we get $p^{\mathcal{Q}}_{\text{discharge}}(M) = \mathrm{O}(\exp(-\alpha_Q M))$.

Finally, we show that Scheme $\mathcal{Q}$ achieves an average utility $\bar{U}_D^{\mathcal{Q}}$ such that $U_D(\lambda) - \bar{U}_D^{\mathcal{Q}} = \Theta\left(\frac{(\ln K)^2}{K^2}\right)$. The instantaneous utility will be zero when the queue is empty or when the battery is discharged. Due to symmetry of the Brownian approximation, the empty queue probability will be the same as the queue overflow probability. Since $p^{\mathcal{Q}}_{\text{discharge}}(M) = \mathrm{O}(\exp(-M))$, under the large battery regime we can ignore the discharge term. The average utility $\bar{U}_D^{\mathcal{Q}}$ can be written as,

$$\begin{aligned}
\bar{U}_D^{\mathcal{Q}} &= \frac{1}{2}\left(U_D(\lambda + \delta_1^{(a)}) + U_D(\lambda - \delta_1^{(a)})\right)(1 - p^{\mathcal{Q}}_{\text{overflow}}(K)) \\
&= U_D(\lambda) + U_D^{(2)}(\lambda)(\delta_1^{(a)})^2 + \mathrm{o}\left((\delta_1^{(a)})^2\right) && \text{(A.40)} \\
&= U_D(\lambda) + \Theta\left(\frac{(\ln K)^2}{K^2}\right). && \text{(A.41)}
\end{aligned}$$

Where Eq. (A.40) follows from the fact that $p^{\mathcal{Q}}_{\text{overflow}}(K) = \mathrm{O}(K^{-\beta})$ for $\beta \geq 2$ and Eq. (A.41) comes from choosing $\delta_1^{(a)} = \beta_D \bar{\sigma}_a^2 \frac{\ln K}{K}$. This completes the proof of Theorem 5.3.

# BIBLIOGRAPHY


[1] M. Lukac, V. Naik, I. Stubailo, A. Husker, and D. Estrin, "In Vivo Characterization of a Wide Area 802.11b Wireless Seismic Array," Center for Embedded Network Sensing, Tech. Rep. 100, May 2007. [Online]. Available: http://repositories.cdlib.org/cens/wps/100

[2] C. Hartung, R. Han, C. Seielstad, and S. Holbrook, "FireWxNet: A Multi-Tiered Portable Wireless System for Monitoring Weather Conditions in Wildland Fire Environments," in *Proc. MobiSys'06*, June 2006, pp. 28–41.

[3] K. Martinez, R. Ong, and J. Hart, "Glacsweb: A Sensor Network for Hostile Environments," in *Proc. IEEE SECON'04*, Santa Clara, CA, October 2004, pp. 81– 87.

[4] J. M. Rabaey, M. J. Ammer, J. L. da Silva, D. Patel, and S. Roundy, "PicoRadio Supports Ad Hoc Ultra-Low Power Wireless Networking," *IEEE Computer*, vol. 33, no. 7, pp. 42–48, 2000.

[5] I. A. Essa, "Ubiquitous Sensing for Smart and Aware Environments," *IEEE Personal Commun. Mag.*, vol. 7, no. 5, pp. 47–49, Oct. 2000.

[6] C. Herring and S. Kaplan, "Component-Based Software Systems for Smart Environments," *IEEE Personal Commun. Mag.*, vol. 7, no. 5, pp. 60–61, Oct. 2000.

[7] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless Sensor Networks for Habitat Monitoring," in *Proc. ACM WSNA '02*. New York, NY, USA: ACM, 2002, pp. 88–97.

[8] C.-Y. Chong and S. P. Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges," *Proc. IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[9] U. Lee, E. Magistretti, M. Gerla, P. Bellavista, and A. Corradi, "Dissemination and Harvesting of Urban Data Using Vehicular Sensing Platforms," *IEEE Trans. Veh. Commun.*, vol. 58, no. 2, pp. 882–901, February 2009.

[10] F.-Y. Wang, D. Zeng, and L. Yang, "Smart Cars on Smart Roads: An IEEE Intelligent Transportation Systems Society Update," *IEEE Pervasive Computing*, vol. 5, no. 4, pp. 68 – 69, Oct. 2006.

[11] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, March 2002.

[12] R. J. McEliece, *The Theory of Information and Coding*, 2nd ed. Cambridge University Press, 2002.

[13] R. A. Berry and R. G. Gallager, "Communication Over Fading Channels With Delay Constraints," *IEEE Trans. Inform. Theory*, vol. 48, no. 5, pp. 1135–1149, May 2002.

[14] A. J. Goldsmith and P. P. Varaiya, "Capacity of Fading Channels with Channel Side Information," *IEEE Trans. Inform. Theory*, vol. 43, no. 6, pp. 1986–1992, November 1997.

[15] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 1st ed. John Wiley and Sons, 1999.

[16] H. Viswanathan, "Capacity of Markov Channels with Receiver CSI and Delayed Feedback," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 761–771, March 1999.

[17] S. Yüksel and S. Tatikonda, "Capacity of Markov Channels with Partial State Feedback," in *Proc. IEEE ISIT '07*, Nice, France, June 2007, pp. 1861–1865.

[18] S. Verdú, "On Channel Capacity per Unit Cost," *IEEE Trans. Inform. Theory*, vol. 36, no. 5, pp. 1019–1030, Sept. 1990.

[19] E. N. Gilbert, "Capacity of a Burst-Noise Channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1266, Sept. 1960.

[20] P. Sadeghi, R. A. Kennedy, P. B. Rapajic, and R. Shams, "Finite-State Markov Modeling of Fading Channels - A Survey of Principles and Applications," *IEEE Signal Process. Mag.*, vol. 25, no. 5, pp. 57–80, September 2008.

[21] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Scientific, 2003.

[22] M. Zorzi and R. R. Rao, "Error Control and Energy Consumption in Communications for Nomadic Computing," *IEEE Trans. Comput.*, vol. 46, no. 3, pp. 279–289, Mar. 1997.

[23] C.-F. Chiasserini and M. Meo, "Impact of ARQ Protocols on QoS in 3GPP Systems," *IEEE Trans. Veh. Technol.*, vol. 52, no. 1, pp. 205–215, Jan. 2003.

[24] L. A. Johnston and V. Krishnamurthy, "Opportunistic File Transfer over a Fading Channel: A POMDP Search Theory Formulation with Optimal Threshold Policies," *IEEE Trans. Wireless Commun.*, vol. 5, no. 2, pp. 394–405, February 2006.

[25] E. Uysal-Biyikoglu, B. Prabhakar, and A. E. Gamal, "Energy-Efficient Packet Transmission Over a Wireless Link," *IEEE/ACM Trans. Networking*, vol. 10, no. 4, pp. 487–499, August 2002.

[26] M. Zafer and E. Modiano, "Delay-Constrained Energy Efficient Data Transmission over a Wireless Fading Channel," in *Proc. Information Theory and Applications Workshop '07*, San Diego, USA, Jan. 2007.

[27] M. A. Haleem and R. Chandramouli, "Adaptive Downlink Scheduling and Rate Selection: A Cross-layer Design," *IEEE J. Select. Areas Commun.*, vol. 23, no. 6, pp. 1287– 1297, June 2005.

[28] C. K. Ho and J. Oostveen, "Rate Adaptation in Time Varying Channels using Acknowledgement Feedback," in *Proc. of IEEE VTC'06-Spring*, May 2006, pp. 1683–1687.

[29] A. K. Karmokar, D. V. Djonin, and V. K. Bhargava, "POMDP-Based Coding Rate Adaptation for Type-I Hybrid ARQ Systems over Fading Channels with Memory," *IEEE Trans. Wireless Commun.*, vol. 5, no. 12, pp. 3512–3523, Dec. 2006.

[30] D. V. Djonin, A. K. Karmokar, and V. K. Bhargava, "Joint Rate and Power Adaptation for Type-I Hybrid ARQ Systems Over Correlated Fading Channels Under Different Buffer-Cost Constraints," *IEEE Trans. Veh. Technol.*, vol. 57, no. 1, pp. 421–435, Jan. 2008.

[31] A. K. Karmokar, D. V. Djonin, and V. K. Bhargava, "Cross-Layer Rate and Power Adaptation Strategies for IR-HARQ Systems over Fading Channels with Memory: A SMDP-Based Approach," *IEEE Trans. Commun.*, vol. 56, no. 8, pp. 1352–1365, August 2008.

[32] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, February 1989.

[33] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 2005.

[34] A. Woo, T. Tong, and D. Culler, "Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks," in *Proc. ACM SenSys'03*, Oct. 2003, pp. 14–27.

[35] J. Zhao and R. Govindan, "Understanding Packet Delivery Performance in Dense Wireless Sensor Networks," in *Proc. ACM SenSys'03*, Oct. 2003, pp. 1–13.

[36] A. Willig, M. Kubisch, H. Christian, and A. Wolisz, "Measurements of a Wireless Link in an Industrial Environment Using an 802.11-Compilant Physical Layer," *IEEE Trans. Ind. Electron.*, vol. 49, no. 6, pp. 1265–1282, Dec. 2002.

[37] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A Short Preamble MAC Protocol for Duty-Cycled Wireless Sensor Networks," in *Proc. ACM SenSys'06*, Nov. 2006, pp. 307–320.

[38] S. Liu, K.-W. Fan, and P. Sinha, "CMAC: An Energy Efficient MAC Layer Protocol Using Convergent Packet Forwarding for Wireless Sensor Networks," in *Proc. IEEE SECON'07*, Jun. 2007.

[39] W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks," *IEEE/ACM Trans. Networking*, vol. 12, no. 3, pp. 493–506, Jun. 2004.

[40] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks," in *Proc. IEEE IPDPS'04*, Apr. 2004, pp. 224–231.

[41] C.-Y. Wan, S. B. Eisenman, and A. T. Campbell, "CODA: Congestion Detection and Avoidance in Sensor Networks," in *Proc. ACM SenSys'03*, Oct. 2003, pp. 266–279.

[42] B. Hull, K. Jamieson, and H. Balakrishnan, "Mitigating Congestion in Wireless Sensor Networks," in *Proc. ACM SenSys'04*, Nov. 2004, pp. 134–147.

[43] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis, "Interference-aware Fair Rate Control in Wireless Sensor Networks," in *Proc. ACM SIGCOMM'06*, Sept. 2006, pp. 63–74.

[44] C. Lim, H. Luo, and C.-H. Choi, "RAIN: A Reliable Wireless Network Architecture," in *Proc. IEEE ICNP'06*, Nov. 2006, pp. 228–237.

[45] S. Liu, R. Srivastava, C. E. Koksal, and P. Sinha, "Pushback: A Hidden Markov Model Based Scheme for Energy Efficient Data Transmission in Sensor Networks," *Ad Hoc Networks*, vol. 7, no. 5, pp. 973–986, July 2009.

[46] M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle Sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs," in *Proc. ACM SIGCOMM'05*, Nov. 2005, pp. 134–147.

[47] A. Arora, E. Ertin, R. Ramnath, W. Leal, and M. Nesterenko, "Kansei: A High-Fidelity Sensing Testbed," *IEEE Internet Computing*, vol. 10, no. 2, pp. 35–47, Mar. 2006.

[48] A. Kansal and M. B. Srivastava, "An Environmental Energy Harvesting Framework for Sensor Networks," in *Proc. ISLPED'03*, August 2003, pp. 481–486.

[49] S. Meninger, J. O. Mur-Miranda, R. Amirtharajah, A. Chandrakasan, and J. H. Lang, "Vibration-to-electric energy conversion," *IEEE Trans. VLSI Syst.*, vol. 9, no. 1, pp. 64–76, Feb. 2001.

[50] J. A. Paradiso and M. Feldmeier, "A Compact, Wireless, Self-Powered Pushbutton Controller," in *Proc. Ubicomp'01.* Springer-Verlag, October 2001, pp. 299–304.

[51] W. Weber, "Ambient Intelligence: Industrial Research on a Visionary Concept," in *Proc. ISLPED'03*, August 2003, pp. 247–251.

[52] J. Hsu, A. Kansal, J. Friedman, V. Raghunathan, and M. Srivastava, "Solar Energy Harvesting Wireless Embedded Systems," in *Proc. IEEE IPSN'05 Demo*, April 2005. [Online]. Available: http://www.ee.ucla.edu/~mbs/ipsn05/demo/18_JHsu.pdf

[53] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. B. Srivastava, "Design Considerations for Solar Energy Harvesting Wireless Embedded Systems," in *Proc. IEEE IPSN'05*, Los Angeles, USA, April 2005, pp. 457 – 462.

[54] K. Kar, A. Krishnamurthy, and N. Jaggi, "Dynamic Node Activation in Networks of Rechargeable Sensors," *IEEE/ACM Trans. Networking*, vol. 14, no. 1, pp. 15–26, Feb. 2006.

[55] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Control of wireless networks with rechargeable batteries," *IEEE Trans. Wireless Commun.*, vol. 9, no. 2, pp. 581–593, Feb. 2010.

[56] R.-S. Liu, P. Sinha, and C. E. Koksal, "Joint Enenrgy Management and Resource Allocation in Rechargeable Sensor Networks," in *Proc. IEEE INFOCOM'10*, San Diego, USA, March 2010.

[57] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power Management in Energy Harvesting Sensor Networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 4, p. 32, September 2007.

[58] C. M. Vigorito, D. Ganesan, and A. G. Barto, "Adaptive Control of Duty Cycling in Energy-Harvesting Wireless Sensor Networks," in *Proc. IEEE SECON'07*, San Diego, CA, June 2007, pp. 21–30.

[59] V. Sharma, U. Mukherji, V. Joseph, and S. Gupta, "Optimal energy management policies for energy harvesting sensor nodes," *IEEE Trans. Wireless Commun.*, vol. 9, no. 4, pp. 1326–1336, April 2010.

[60] D. N. C. Tse, "Variable-rate lossy compression and its effects on communication networks," Ph.D. dissertation, MIT, Cambridge, MA, September 1994. [Online]. Available: http://www.eecs.berkeley.edu/~dtse/pub.html

[61] W. Whitt, *Stochastic-Process Limits*, 1st ed. Springer, 2002.

[62] J. M. Harrison and R. J. Williams, "Multidimensional reflected brownian motions having exponential stationary distributions," *Ann. Prob.*, vol. 15, no. 1, pp. 115–137, January 1987.

[63] J. M. Harrison, *Brownian Motion and Stochastic Flow Systems*, 1st ed. John Wiley and Sons, 1985.

[64] R. G. Gallager, *Discrete Stochastic Processes*, 1st ed. Kluwer Academic Publishers, 1996.

[65] (2010) Measurement and Instrumentation Data Center. National Renewable Energy Laboratory. [Online]. Available: http://www.nrel.gov/midc/

[66] S. M. Ross, *Stochastic Processes*, 2nd ed. Wiley, 1995.