12-2012

Control Design and Filtering for Wireless Networked Systems

Xiao Ma
xma4@utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Xiao Ma entitled "Control Design and Filtering for Wireless Networked Systems." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

Seddik M. Djouadi, Major Professor

We have read this dissertation and recommend its acceptance:

Douglas Birdwell, Husheng Li, Vasileios Maroulas, James Nutaro

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Control Design and Filtering for Wireless Networked Systems

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Xiao Ma

December 2012

To my mother and father

Yingjue Wang and Kening Ma

Acknowledgements

In the moment of completing my dissertation, I have so many people to thank that it could be another Ph.D dissertation. Five years' work does not only include my own effort but also other people's help.

My debt of gratitude must go to my advisor, Dr. Seddik Djouadi. His professionalism and expertise taught me to survive. Moreover, without his supervision, support, and encouragement, I could not finish this whole project. I also want to thank him for his humor and kindness which warmed my heart throughout the difficulties that I faced during this pursuit.

Special thanks to my committee, Dr. Husheng Li, Dr. Jie Xiong, Dr. Douglas Birdwell, Dr. Vasileios Maroulas, and Dr. James Nutaro for their support, guidance and helpful comments, criticism and suggestions. They provided me many ideas and new thoughts to accomplish the dissertation.

I would like to thank my parents, Kening Ma and Yingjue Wang. They gave me the confidence to come here to pursue the Ph.D degree. I owe them everything and wish I could show them just how much I love and appreciate them. I also wish to thank my wife, Xin Jia, who accompanied with me for the past three years of my pursuit. I will spend the rest of my life showing her my appreciation, so I just give her a heartfelt thanks here.

At last, I would like to deeply appreciate Oak Ridge National Laboratory (ORNL) for their support of my work during the past five years. Specially, thanks to Teja

Kuruganti, Mohammed Olama, and Stephen Smith for their valuable discussions and suggestions.

Knowledge changes fates

–Ka-Shing Li

Abstract

This dissertation is concerned with estimation and control over wireless networked systems. Several problems are addressed, including estimator design over packet loss links, control and estimation over cognitive radio systems, modeling and prediction of wireless sensor networks (WSNs), and localization with the Theater Positioning System (TPS).

The first problem addressed is the state estimation of a discrete-time system through a packet loss link modeled by a Bernoulli random variable. The optimal filter is derived by employing exact hybrid filtering. The performance of the optimal filter is illustrated by numerical simulations.

Next, we consider the problem of estimation and control over cognitive radio (CR) systems. A two-switch model is first used to model this link. The linear optimal estimator and controller are derived over a single CR link. Also discussed here is estimation and control of the closed-loop system over two CR links.

Furthermore, a more practical semi-Markov model for the CR system is proposed. Two cases are considered, where one assumes that acknowledgement of the information arrival is not available while the other assumes it is available. In the former, a suboptimal estimator is proposed and, in the latter, sufficient conditions are derived for the stability of a peak covariance process. Then, a controller design for the semi-Markov model is developed using linear matrix inequalities (LMIs).

Additionally, the third problem addressed is modeling, identification, and prediction of the link quality of WSNs, such as the packet reception rate (PRR)

and received signal strength indicator (RSSI). The state-space model is applied for this purpose. The prediction error minimization method (PEM) is employed for estimating parameters in the proposed model. The method employed is demonstrated through real measurements sampled by wireless motes.

The last problem analyzed is localization using a new navigation system, TPS. In this study, we focus on users' position estimation with the TPS when a GPS signal is not available. Several models are proposed to model transmission delays utilizing previous GPS signals. Last, a navigation scheme is provided for the TPS to improve its localization accuracy when the GPS signal is unavailable.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background and Overview

1.1.1 Packet Losses Links

Networks are applied in many modern technologies and extend the range of information communications to huge distances as well as resulting in significant financial savings. In recent years, the combination of network and control has been a major topic in the control community and is developing rapidly where systems are distributed in different places. However, due to network constraints, e.g., packet loss, transmission delay, and bandwidth, traditional control algorithms do not provide satisfactory performance. Thus, algorithms must be developed. In recent decades, the packet loss problem has received a great deal of attention. Different kinds of state estimators over packet loss links have been developed.

In this dissertation, we derived the optimal filter explicitly over a packet loss link. For example, consider a Bernoulli lossy link from the output of the plant to the estimator. We modeled the Bernoulli packet loss indicator as a Markov process, and then the system can be considered as a special case of the problem in [14] where the exact hybrid filter can be applied. In this way, we obtained an explicit solution and thus provided the optimal filter to this problem. Also, there are many studies of

state estimation on Markov jump linear systems (MJLS) [15]∼[21]. However, most of them computed only suboptimal estimators or linear estimators.

The main idea of exact hybrid filter is the change of measure method, which transfers the problem from the current probability space where the optimal filter is difficult to derive to another probability space where the optimal filter derivation is tractable.

1.1.2 Control and Estimation over Cognitve Radio System

The rapid development of communication and networking has extended the areas of traditional science. These remote techniques are employed everywhere to facilitate user application in different areas. However, the wide use of various technologies, such as radio, satellite, and phone service, also increases the need of bandwidth used for the transmission. Most of the current spectrum has been licensed to different users to ensure the coexistence of diverse wireless systems [32]. Thus an important question: *How can bandwidth be saved without affecting the performance too much*?

The Federal Communications Commission's (FCC) frequency allocation chart [33] shows that although the majority of frequency spectrum has been assigned to different users, large portions of spectrum are frequently unused [34]. To increase spectrum use, **cognitive radio architecture** [35] [36] is proposed as a communication system to sense available spectrum, search for unutilized spectrum, and communicate over the unused spectrum with minimal disturbance to primary users. In the CR system, each secondary user is able to sense the licensed spectrum band and detect unused spectrum holes. If a frequency channel is not being used by primary users, secondary users can access it for communications. Due to sparse activities of primary users, CR can provide a large amount of spectrum for communications. With a CR system, the question above is answered as bandwidth, thus, money can be saved for the transmission.

An interesting application of CR is in control engineering, which, however, raises new issues. For instance, when the user wants to do remote control without having any authorized bandwidth or without enough funds to purchase large bandwidth. CR can be employed to help the user reach his target; however, CR suffers from interruptions from primary users since secondary users must leave the licensed channel when primary users emerge. Hence, the CR-based communication link may not be reliable, which can cause significant impact on the system state estimation and control since observations from sensor may not be able to reach the controller in a timely fashion.

In Chapter 3, the CR system is modeled by a two-switch model with distributed and dynamic spectral activity introduced in [32]. This model employs two sensors located at the transmitter and the receiver of secondary users, respectively, where they can sense whether the channel is free for secondary users to transmit. The switching variables are assumed to be Bernoulli variables. The advantage of the two-switch model is that it can avoid disturbing the primary users and thus preserves the benefit of primary users. The details of this model are provided in Chapter 3.

On the other hand, the two-switch model suffers from two shortcomings: (a) The influence of the secondary receiver is not so obvious unless the receiver is very close to primary users, (b) The sensor located at the secondary receiver increases the cost. Thus, it is more practical to not put a sensor at the secondary receiver, and CR system is reduced to a model with only one switching variable, the same as packet loss model considered in 1.1.1. Moreover, as shown through theory and experiments [39], a semi-Markov process captures the stochastic behavior of each channel in the CR system more accurately. Based on these facts, a semi-Markov model is proposed, where measurements are governed by N independent semi-Markov processes, where N represents the number of channels that can be sensed in CR system. State estimation over the semi-Markov model is addressed. In particular, two different cases are considered: one case assumes acknowledgement of information arrival is not available at the estimator while the other assumes it is available. In

the first case, sufficient conditions are derived for the stability of the peak covariance process which is an estimate of filtering deterioration caused by packet losses, and in the second case, an estimator is proposed based on Interacting Multiple Method (IMM). In Chapter 5, we will discuss the controller design over such a CR link.

1.1.3 Link Quality Prediction for Wireless Sensor Networks

One key component of the emerging networked embedded systems, such as wireless sensor networks (WSN), is efficient and reliable data collection and aggregation. This task is complicated by various factors, and of the most notable is the extremely unreliable nature of wireless links through which data are collected, causing several problems such as congestion and packet losses. Inappropriate selection and use of wireless links will cause tremendous energy cost, shortened system lifetimes, and degrade performance.

So far, the basic structure for the collection is usually considered to be a multi-hop tree topology [119]: Each node is connected to the root through multiple hops, forming a tree structure. Routing protocols establish the routing tree based on the wireless link quality to reduce the end-to-end path cost, thus, decreasing the cost of sending a packet to the root.

However, even though enormous efforts have been invested in choosing the best link to deliver packets, reports from the field are far from satisfactory. In a well known study monitoring volcano activities, the data yielded was estimated to be only between 20% and 80% [122]. Other experiments yielded similar results [121].

We observed that data losses were inevitable as long as the size of internal buffers of intermediate nodes were limited in the presence of poor link quality. This is because the sender could easily fill up the buffer given that there is not sufficient time to reliably transmit all packets. One of the key reasons for this problem is the lack of the feedback of the link-layer information to the upper-layer applications and protocols. For example, in a sensor network developed to gather the real-time information of

passing vehicles (e.g., the VigilNet project [120]), whenever the target appears, the traffic volume surges, causing internal packet queues to grow dramatically. At the same time, simultaneous transmissions along the tree cause interference, decreasing the link quality of pre-established trees. On the other hand, the application tries to guarantee reliable packet delivery by retransmissions, only to cause cascading effects that further reduce available bandwidth, a problem that finally leads to packet losses when internal queues become full.

While packet losses may not be a big problem for applications that are tolerant to them due to the redundancy in sensor data, for those applications that require high degree of fidelity in data records, the problem will be extensive. For example, for a smart camera sensor network that transmits one single image using multiple encoded packets, losing any packet will lead to failures in reconstructing the complete original picture. Another problem with lost packets is wasted energy, such as a path of N hops where a packet is lost in the Nth hop. The energy spent on the transmission and the retransmission of these $N-1$ hops will be wasted. Therefore, losing packets that have traveled for long distances is especially cost inefficient. For these reasons, losing packets poses serious challenges for the cost effectiveness of WSNs.

In this dissertation, we propose a new idea: We use a state-space model to predict the link quality and provide these estimates as a system-level service to application developers. This idea is based on the premise that to achieve the best performance, the application-layer behavior should be aware of the networking-layer conditions, e.g. in the collection protocol, and adjust its behavior accordingly, to achieve balanced performance with the link quality. The resulting integrated framework is what we call LIPS, or ***Li****nk* ***P****redictions as a* ***S****ervice*, and represents an integrated solution.

1.1.4 Localization in the Theater Positioning System

The last topic discussed in this dissertation is localization, using what we call the Theater Positioning System (TPS). The Global Positioning System (GPS) has been

widely employed for both military and civil purposes. However, the nearly exclusive dependence on the GPS satellite constellation for accurate position information becomes a major operational concern for deploying U. S. military and law-enforcement personnel. Such concern comes from the comparatively weakness of the GPS, e.g. it may suffer from multipath and RF interference (intentional and unintentional), or even invasion from an adversary, which results in inaccurate localization. A backup to the GPS when GPS signals are out of reception is inertial navigation systems (INS) [99, 100]. These units can be viewed as short-term backups to GPS but are in general too costly, inaccurate, and/or power-hungry to be deployed except in a few specialized applications. Thus, a much more robust, inexpensive, and reliable GPS augmentation technique is badly needed for dismounted personnel and most platforms.

TPS is less expensive and offers far more consistent coverage than with the GPS alone was developed in [89]. It operates at 90-110 kHz ground-wave radio-frequency (RF) and can be used with or without the GPS. Like the GPS, this enhanced low frequency (LF) component of the system uses spread-spectrum transmission to improve the accuracy, exhibits a large processing gain for greater interference immunity, and thus has a significant advantage over conventional LORAN-C radio-navigation systems. It can be considered to be a navigation system that uses terrestrial signal transmitters and a much lower RF compared with the GPS. The low frequency property improves the capability of TPS signals penetration through obstacles such as buildings, canyons, and forrests, where the GPS may be disabled due to signal blocking. For example, my GPS did not work well or even provide close navigation in downtown Chicago due to skyscrapers and bridges. The independence of the TPS allows it to provide localization information for users when GPS signals are denied.

The predecessor of the TPS component of the navigation system is LORAN, which was the original wide-area radio-navigation system that preceded the GPS. Its relatively long wavelength (3000 meters) provides wide geographic coverage via ground-wave propagation without the need for satellites. Furthermore, its

long wavelength provides significant immunity from false locations due to local multipaths. However, conventional LORAN is limited by a lack of resolution and is highly susceptible to interference. Incorporating spread-spectrum signals at LORAN frequencies provides more precise location and better interference immunity [89].

In the usual operating mode, GPS serves as the principal positioning source. Continuity of their fixes are assured since, during the normal TPS tracking process, the TPS and GPS position data are continually compared. As long as the recent and current GPS signal quality is good, the displayed TPS fix will be automatically adjusted to overlay GPS values; this is generally done to provide an ongoing in-situ calibration of TPS signal propagation delay figures and thus "drag" the TPS fix to match the GPS. If the GPS suddenly fails to provide a clean or continuous fix, the TPS value will track the last good GPS coordinates. Once the GPS signal integrity is restored for at least a few seconds and a new lock is satisfactorily obtained, the system will smoothly revert to the GPS fix and return to the normal operation. In the event that the GPS is jammed or otherwise unavailable for an extended period, the TPS will be employed in a standalone mode to derive the unit's fix, with a caution to the user that the fix accuracy may be reduced. Another specific advantage of the TPS concept lies in the use of the TPS as an antispoofing detector for the GPS. For instance, if the TPS (presumed stable) and GPS planar fix do not essentially coincide (i.e., where the GPS solution differs considerably from the TPS fix), this could be an indicator of GPS receiver problems or of the presence of a spoofing signal. However, the fix accuracy is reduced when the TPS works alone [89]. Thus, compensation methods need to be developed for the purpose of improving the accuracy.

1.2 Literature Review

1.2.1 Optimal Filtering over Packet Loss Links

Many studies have been conducted, in particular, on the limitations of networks such as packets loss and communication delays [1]∼[9]. Other research has focused on communication properties such as bandwidth limitation and quantization levels in [10] and [11]. The study of networking and communications, together with control systems, is a major direction for modern control now and in the near future.

Several investigations have been performed on the combination of network and control systems, and some optimal control and estimation algorithms have been derived under certain situations. An optimal LQG controller taking the factors of bounded delays between the sensor and the controller, and the controller and the actuator is proposed [1], but packet losses were not taken into consideration. Other studies [2] and [3] considered uncertain observations. More recently, a suboptimal estimator was provided [4]. Another study [5] proposed a Kalman Filter with intermittent observations under a Bernoulli distribution. In [6], an optimal LQG controller with packet losses both between the sensor and the controller, and between the controller and the actuator was developed. There the authors considered both the TCP protocol and the UDP protocol between the controller and actuator and assumed available information of packet arrival from the sensor to the estimator. Then, [7] derived stability conditions for Kalman filtering with Markovian packet losses. Moreover, another work [9] proposed an optimal LQG algorithm by positioning the encoder and the decoder at the transmitter and the receiver, respectively. All these papers tackled problems on the packet dropouts model with time stamp. Furthermore, [12] considered the optimal estimator of the packet loss model assuming no packet arrival information; however, it is only the linear optimal estimator of the problem as it assumed that the optimal state estimate was a linear function of the measurement.

Later, the work [13] provided a way to compute the optimal state estimate without any assumption of the form of the estimator.

1.2.2 Estimation and Control over CR systems

Significant research has been performed in control and estimation over communication links under constraints such as packet losses, transmission delays, and bandwidth constraints [1]∼[11], but minimal research has been initiated regarding CR architecture. The state estimation of the system over a CR system was first considered by [37], where the CR link was modeled by a two-switch model with distributed and dynamic spectral activity [32]. The switching variables were assumed to be Bernoulli variables. Control and estimation of the closed-loop system over the same CR links were then discussed by [38].

As the CR system can be modeled as a semi-Markov model, the problem can be further formulated as a semi-Markov jump linear system (SMJLS) problem. Control and estimation of SMJLSs can be extended from those for Markov jump linear systems (MJLSs). There are many studies of the estimator design of MJLSs, e.g., particle filters [15, 16], exact hybrid filter [14], linear mean square estimator [17], and others [19]∼[24].

Control design over MJLSs has been developed comprehensively over the past several decades. There are some studies focused on the system governed by one Markov process, e.g., [60]∼[70]. Those works computed the optimal control expressions using standard dynamic programming method and derived different stability conditions, e.g., mean-square stability, stochastically stability, exponential mean square stability, and almost sure stability. The first three stability notions have been shown to be equivalent and concluded as second-moment stability (SMS) by [66]. Moreover, it showed that the first three were also sufficient but not necessary conditions for the fourth stability notion. For SMJLSs, several works have been conducted on the control design for continuous-time systems [72, 73, 74]. For example,

the optimal control of SMJLSs was computed by [72], but it is difficult to compute the optimal solution explicitly due to coupled Riccati equations. Later, a suboptimal but tractable control solution was developed in [73] by applying techniques that had been used in MJLSs [69]. More recently, a robust state feedback controller was developed in [74]. In additional, [75] considered the control design over a discrete-time SMJLS; the cost function employed did not depend on the sojourn time.

With the increasing popularity of NCSs, the theorems developed in the control design of MJLSs were also widely applied in NCSs by assuming the network or communication factors to be Markovian random variables (see [76] for a survey in NCSs). For example, an H_∞ approach to Markovian packet loss links was developed in [71] by employing the stability conditions derived for MJLSs.

1.2.3 Link Quality Prediction in WSNs

Previous works on link quality prediction have been undertaken in several papers [82, 83, 85]. In [82, 83], the AR model was applied to model the behaviour of the quality metrics, e.g., signal to noise ratio (SNR) in [82], and PRR in [83]. Moreover, three different methods, Bayes classifier, logistic regression, and artificial neural networks were employed for modeling metrics [84]. The modeling step implemented the model obtained from offline training and selection. From a experimental point of view, they showed that the logistic regression performed the best among the three methods. Additionally, a link quality estimator was proposed by [86] which used packet loss rate (PLR), round trip time (RTT), available bandwidth (ABW), and feedback information to track the channel error probability and the collision probability of the receiver using unscented Kalman filter. However, it assumed a known model to avoid the procedure of the parameter estimation. Similarly, the relationship model between the chip correlation indicator (CCI) and the PRR was considered under the case of perceive packet loss [84]. It used Kalman filter to extract the CCI from the noise background through a known discrete-time state-space model.

1.2.4 Navigation with the TPS

The concept and detail configuration of the TPS were proposed by [89]. Most GPS techniques can be employed directly [90]. There are also some methods that have been already applied in solving GPS pseudorange equations, e.g., Newton-Raphson [90, 91], Kalman filter [92, 93, 94] or particle filter [95].

1.3 Contributions

Contributions of the research undertaken here are summarized as follows:

1. Derivation of the optimal filter over Bernoulli i.i.d packet losses link.

The derivation of the optimal filter provided in this dissertation employs exact hybrid filter and shows the optimal filter is a non-linear function of the measurement.

2. Derivation of a linear optimal estimator and a linear optimal controller over a single CR link, which is represented by the two-switch model.

The CR system is first modeled by a two-switch model proposed in the communication community. The linear optimal estimator is derived by placing a CR link between the sensor and the estimator, and the linear optimal controller is also derived with a CR link between the controller and the actuator.

3. Derivation of the estimator and the controller of the closed-loop system over double CR links with stability conditions.

The linear optimal estimator is obtained, and it is demonstrated that the optimal controller is not a linear function of the state estimate. It is also proven that the separation principle does not hold.

4. Derivation of sufficient stability conditions for the case with acknowledgement of packet arrivals.

Sufficient stability conditions of the peak covariance process of the state estimator when acknowledgement of packet arrivals is known are derived in terms of statistics of the semi-Markov process.

5. Design of the state estimator without acknowledgement of packet arrivals.

The switching variable is governed by N semi-Markov processes and is further modeled by a process with $2N$ states. The time varying transition probability matrix is computed, and the IMM algorithm is employed to complete the estimation.

6. Prediction of link quality metrics in WSNs.

The state-space model is proposed to model, identify and predict RSSI, LQI, and PRR in WSNs and demonstrated with real measurements from WSNs..

7. Development of a navigation scheme for the TPS when the GPS signal is not available.

Three different models are employed and predicted the delays caused during the TPS transmission by utilizing past GPS signals to improve the localization accuracy of the TPS.

1.4 Thesis Organization

The thesis is organized as follows:

Chapter 2 considers the filtering over wireless communication channels subject to packet losses. The packet losses are assumed to follow a Bernoulli distribution. It is interpreted as a special case of a Markov process for which hybrid filtering theory is shown to provide an exact solution. The optimal filter is derived and shown to be a non-linear function of the measurement. Illustrative examples compare the performance of the linear optimal estimator and the optimal filter and show that the latter offers superior performance.

Chapter 3 considers control and estimation via the two-switch model, which represents the CR system. The linear optimal estimator and the linear optimal controller are derived through a single CR link. Attention is then turned to control and estimation of the closed-loop system over double cognitive radio links and it is demonstrated that the optimal controller is nonlinear in the state estimate. As a result, it is also shown that the separation principle does not hold. Several stability

conditions are also discussed. Numerical examples are provided to illustrate the results.

Chapter 4 proposes to communicate through a CR link represented by a semi-Markov model between the sensor and the estimator based on results in [39]. In this way, the link is governed by multiple semi-Markov processes, each of which can capture the stochastic behavior of each channel to be sensed. Two different cases are considered, where one assumes acknowledgement of the information arrival is not available at the estimator, while the other assumes it is available. In the first case, sufficient conditions are derived for the stability of the peak covariance process, and in the second case, the state estimator is proposed based on IMM.

Chapter 5 first computes the optimal controller of the discrete-time system over the semi-Markov based CR link between the controller and the actuator. However, the optimal solution is untractable; thus, a suboptimal controller that is tractable and based on linear matrix inequalities (LMIs) is derived.

Chapter 6 considers the link quality prediction of WSNs. The link quality metrics employed are RSSI, LQI, and PRR. State-space model is proposed to model these metrics using past measurements; the model built is then used to predict future quality metrics to seek the best transmission channel. Real experimental data is used to demonstrate the proposed method.

Chapter 7 addresses the accuracy problem of the TPS when it works alone. The transmission delay, which is one of the main reasons that degrade the accuracy, is modeled by three different models by using past GPS signals. Then, they are employed to compute the delays. A navigation scheme based on these methods is also proposed for the TPS when it works alone.

Chapter 2

Optimal Filtering over Packet Losses Link

Control theories currently enjoy very broad cooperation with communication links. The uncertainties, such as packet losses, delays, capacity constraints, and bandwidth limit, introduced by transmission links require the existing control algorithms to adjust these uncertainties when control systems are connected through these links. This chapter considers the optimal filtering over a Bernoulli i.i.d packet loss link, where the arrival information is unknown to the estimator. Parts of this work have been published in [128].

2.1 Linear Optimal Estimator

[12] first considered this problem by assuming the state estimate is a linear function of the measurement. In this section, we formulate the problem and give this linear optimal estimator.

2.1.1 Problem Formulation

For the following linear discrete-time system:

$$\begin{aligned} x_{k+1} &= Ax_k + \upsilon_k \\ y_k &= C_k x_k + \omega_k \end{aligned} \tag{2.1}$$

where $x_k \in \mathbb{R}^n$ is the state at time k, $y_k \in \mathbb{R}^l$ is the observation received at the estimator, $\upsilon_k \in \mathbb{R}^n$ is the Gaussian white noise with zero mean, $\omega_k \in \mathbb{R}^l$ is another Gaussian white noise with zero mean and independent of υ_k. Let V and W denote the covariance matrices of υ_k and ω_k, respectively. The matrices A and B are deterministic parameters of the system, while $C_k = \gamma_k C$ is the stochastic parameter taking the factor of packet losses into account. We assume there are packet losses between the sensor and the estimator and use γ_k to represent packet arrival at time k, where γ_k is a Bernoulli random variable with probability $\mathbb{P}(\gamma_k = 1) = p$ and $\mathbb{P}(\gamma_k = 0) = 1 - p$. We assume that the state, the noise and γ_k are independent of each other. Moreover, We assume **the estimator does not have knowledge of packet arrivals**, which means that the information set $I_k = \{y_k\}_k$ and the estimator does not know γ_k. The arrival probability p is used to design the optimal estimator.

The estimator minimizes the following cost function:

$$J_k = \mathbb{E}((x(k) - \hat{x}(k))^T (x(k) - \hat{x}(k)) | I_k) \tag{2.2}$$

where $\hat{x}(k)$ is the optimal state estimate.

We first give the state estimate and the error covariance prior to received measurements at the current time as follows:

$$\begin{aligned} \hat{x}_{k|k-1} &= A\hat{x}_{k-1|k-1} \\ P_{k|k-1} &= AP_{k-1|k-1}A^T + V \end{aligned} \tag{2.3}$$

where $\hat{x}_{k-1|k-1}$ is the posterior state estimate at time $k-1$, $\hat{x}_{k|k-1}$ is the prior state estimate at time k. $P_{k-1|k-1}$ and $P_{k|k-1}$ are corresponding covariances.

Assume that the optimal state estimate after received measurements can be written in the following form:

$$\begin{aligned} \hat{x}_{k|k} &= \mathbb{E}\{\hat{x}_{k|k-1} + K_k(y_k - C_k\hat{x}_{k|k-1})\} \\ &= \hat{x}_{k|k-1} + K_k(y_k - pC\hat{x}_{k|k-1}) \end{aligned} \tag{2.4}$$

Then, the linear optimal estimator has been computed as follows [12]:

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(y_k - pC\hat{x}_{k|k-1}) \\ W^{'} &= W + (p - p^2)CX_k\,C^T \\ P_{k|k} &= (I - K_kpC)P_{k|k-1}(I - K_kpC)^T + K_kW^{'}K_k^T \\ K_k &= P_{k|k-1}pC^T(pCP_{k|k-1}pC^T + W^{'})^{-1} \end{aligned} \tag{2.5}$$

If we go back to (2.1) and observe that the measurement is not Gaussian as it is the summation of a Bernoulli process and a Gaussian process, it may be not appropriate to assume the state estimate to be a linear function of the measurement because the state is Gaussian. To obtain the optimal filter, the next subsection employs the exact hybrid filter theory [14] and derives the optimal state estimate and error covariance.

2.2 Optimal Filter

In this section, we examine the problem by formulating it into a Markov jump linear system and derive the optimal filter.

2.2.1 Problem Reformulation

In this subsection, the problem in (2.1) is reformulated for convenience of the application of the exact hybrid filter. Note the reformulated one below is the same as (2.1).

We start with some mathematical preliminaries. Let $(\Omega, \Gamma, \mathbb{P})$ be a probability space upon which $\upsilon_k \in \mathbb{R}$ and $\omega_k \in \mathbb{R}$ are independent Gaussian white sequences with zero mean and unit variance. For convenience, we assume a scalar system, however we will indicate how to extend the result to the general case later on. Consider the following linear discrete-time system:

$$\begin{aligned} x_{k+1} &= Ax_k + B\upsilon_k \\ y_k &= h(Z_k)x_k + \omega_k \end{aligned} \tag{2.6}$$

where $x_k \in \mathbb{R}$ is the state of the system at time k, $y_k \in \mathbb{R}$ is the measurement received at the estimator; The scalars $A \in \mathbb{R}$ and $B \in \mathbb{R}$ are system parameters. x_0 is the initial value of processes $\{x_k\}_{k\in\mathbb{N}}$. Let $\{\Gamma_k\}_{k\in\mathbb{N}}$ be the complete filtration generated by $\{x_0, ...x_k, Z_0, ...Z_k, \omega_0, ...\omega_{k-1}\}$ and $\{\Sigma_k\}_{k\in\mathbb{N}}$ be the complete filtration generated by $\{Z_0, ...Z_k\}$. $\{Z_k\}_{k\in\mathbb{N}}$ is assumed to be a Σ_k-Markov process: $Z_{k+1} = \Pi Z_k + M_k$ with a state-space $S = \{e_1, e_2\}$ where e_1, e_2 are 2-canonical unit vectors, and Z_0 is uniformly distributed and independent of other processes. M_k is an Σ_k-martingale increment and $\Pi = (\pi_{ji})$ is a 2×2 matrix with $i, j = 1, 2$ and $\pi_{ji} = \mathbb{P}(Z_k = e_j | Z_{k-1} = e_i)$. Here we consider an i.i.d Bernoulli packet loss model, we have the relationship: $\pi_{ji} = \pi_{ii}$ and the transition probability matrix is

$$\begin{bmatrix} 1-p & p \\ 1-p & p \end{bmatrix}$$

in this case. In our model, we set $h(Z_k) = 0$ when $Z_k = e_1$ represents the packet loss and $h(Z_k) = C$ (C is a scalar) when $Z_k = e_2$ represents the packet arrival at that

time. The complete filtration generated by $\{y_0, ..., y_k\}$ is labeled by $\{I_k\}_{k\in\mathbb{N}}$. The optimal state estimate is defined as the minimizer of the following cost function:

$$J_k = \mathbb{E}((x_k - \hat{x}_k)^2 | I_k) \tag{2.7}$$

Note that from equation (2.6) we see that the measurement is the summation of a Markov process and a Gaussian process which renders measurements not Gaussian.

2.2.2 Optimal Filter over the Bernoulli Packet Loss Link

In this subsection, we rely on hybrid filtering theory [14] to derive the optimal filter. The main idea is to use a change of measure method, which transforms the filtering problem from the original probability space to a new probability space where the measurement and the state are both Gaussian. The optimal filter under the original probability space is then obtained by using the conditional Bayes' theorem.

Initially, assume that all processes are defined on a new probability space $(\Omega, \Gamma, \mathbb{Q})$, and under $\mathbb{Q}$:

1) $\{x_k\}_{k\in\mathbb{N}}$ is a sequence of independent and identically distributed real Gaussian random variables $N(0, 1)$ with density function Φ;

2) $\{y_k\}_{k\in\mathbb{N}}$ is a sequence of independent and identically distributed real Gaussian random variables $N(0, 1)$ with density function Φ;

Under the new probability measure $\mathbb{Q}$, it is able to compute the optimal estimate as both x_k and y_k are Gaussian. Then, the optimal filter with respect to $\mathbb{P}$ can be obtained.

For $k = 0, 1, 2, ...$, let $\bar{\lambda}_0 := \frac{\Phi(y_0 - h(Z_0)x_0)}{\Phi(y_0)}$ and define:

$$\bar{\lambda}_{k+1} := \frac{\Phi(y_{k+1} - h(Z_{k+1})x_{k+1})}{\Phi(y_{k+1})} \frac{\Phi(\frac{x_{k+1} - Ax_k}{B})}{B\Phi(x_{k+1})} \tag{2.8}$$

$$\bar{\Lambda}_n = \prod_{k=1}^{n} \bar{\lambda}_k \tag{2.9}$$

Then the process $\{\bar{\Lambda}_k\}_{k\in\mathbb{N}}$ is an $\mathbb{Q}$-martingale with respect to the filtration Γ_k, $k \in \mathbb{N}$ [22].

Define the probability measure $\mathbb{P}$ on Ω that is absolutely continuous with respect to $\mathbb{Q}$ and the Radon-Nikodym derivative on (Ω, Γ) with restriction to Γ_k is given by : $d\mathbb{P}/d\mathbb{Q} = \bar{\Lambda}_k$. Then, on $\{\Omega, \Gamma_k\}$ and under $\mathbb{P}$, $\{\upsilon_k\}_k$, $\{\omega_k\}_k$, are i.i.d. standard Gaussian random processes, such that

$$\begin{aligned} \upsilon_k &= \frac{x_{k+1} - Ax_k}{B} \\ \omega_k &= y_k - h(Z_k)x_k \end{aligned} \tag{2.10}$$

Let $g : \mathbb{R} \to \mathbb{R}$ be a "test function". From the Bayes' theorem, for Γ-adapted sequence $\{\langle Z_k, e_i\rangle g(x_k)\}_{k\in\mathbb{N}}$:

$$\mathbb{E}[\langle Z_k, e_i\rangle g(x_k)|I_k] = \frac{\bar{\mathbb{E}}[\bar{\Lambda}_k\langle Z_k, e_i\rangle g(x_k)|I_k]}{\bar{\mathbb{E}}[\bar{\Lambda}_k|I_k]}$$

where $\bar{\mathbb{E}}(\bullet)$ denotes the expectation with respect to probability measure $\mathbb{Q}$. Then, define $q_k^i(x), p_k^i(x)$ as the unnormalized conditional density and normalized density as follows [14]:

$$\bar{\mathbb{E}}[\bar{\Lambda}_k\langle Z_k, e_i\rangle g(x_k)|I_k] = \int_{\mathbb{R}} g(\eta) q_k^i(\eta) d\eta \tag{2.11}$$

$$\mathbb{E}[\langle Z_k, e_i\rangle g(x_k)|I_k] = \int_{\mathbb{R}} g(\eta) p_k^i(\eta) d\eta \tag{2.12}$$

Then still from a version of Bayes' theorem, we have

$$p_k^i(\eta) = \frac{q_k^i(\eta)}{\sum_{j=1}^{2}\int_{\mathbb{R}} q_k^j(\zeta)d\zeta} \tag{2.13}$$

A recurrence relationship is obtained between the unnormalized probability densities $q^i_{k+1}(x)$ and $q^i_k(x)$:

$$q^i_{k+1}(x) = \frac{\Phi(y_{k+1} - h(Z_{k+1})x_{k+1})}{\Phi(y_{k+1})} \sum_{j=1}^{2} \frac{\pi_{ji}}{B} \int_{\mathbb{R}} \Phi(\frac{x - A\zeta}{B}) q^j_k(\zeta) d\zeta \tag{2.14}$$

Then the unnormalized probability density is computed as follows:

Lemma 2.0.1. *Suppose $q^i_0(x)$ is Gaussian for $i = 1, 2$. Then $q^i_k(x)$ is a finite sum of Gaussian densities and*

$$q^i_k(x) = \sum_{l=1}^{2^k} A_k(i,l) \exp[-\alpha_k(i,l)x^2 + \beta_k(i,l)x]$$

where

$$\begin{aligned}
A_k(i,l) &= \tfrac{\pi_{qr} A_{k-1}(q,r)}{B\sqrt{2D_{k-1}(i,l)}} \exp[\tfrac{1}{2}(\tfrac{\beta_{k-1}(q,r)^2}{2D_{k-1}(i,l)})] \\
D_{k-1}(i,l) &= \tfrac{(A)^2}{2B^2} + \alpha_{k-1}(q,r), \\
\beta_k(1,l) &= \tfrac{A\beta_{k-1}(q,r)}{2B^2 D_{k-1}(1,l)}, \\
\beta_k(2,l) &= Cy_k + \tfrac{A\beta_{k-1}(q,r)}{2B^2 D_{k-1}(2,l)} \\
\alpha_k(1,l) &= \tfrac{1}{2B^2} - \tfrac{A^2}{4B^4 D_{k-1}(1,l)} \\
\alpha_k(2,l) &= \tfrac{C^2}{2} + \tfrac{1}{2B^2} - \tfrac{A^2}{4B^4 D_{k-1}(2,l)}
\end{aligned}$$

The integers q, r are defined by the following equation:

$$(q,r) = \begin{cases} (\tilde{q}, 2^{k-1}), \ if \ \tilde{r} = 0 \\ (\tilde{q}+1, \tilde{r}), \ otherwise \end{cases}$$

with the integers $\tilde{q}, \tilde{r}$ being the quotient and rest of the division of l by 2^{k-1}.

Proof. The proof of this lemma follows from arguments in [14] directly, and is omitted here. □

The next lemma computes the optimal state estimate and error covariance through (2.13) and the results are in terms of $A_k(i,l)$, $\alpha_k(i,l)$ and $\beta_k(i,l)$.

Lemma 2.0.2. *The optimal state estimate $\hat{x}_k$ and error covariance P_k are:*

$$\begin{aligned}\hat{x}_k &= \frac{1}{S_k}\sum_{i=1}^{2}\sum_{l=1}^{2^k} A_k(i,l)\exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)})\frac{\beta_k(i,l)}{2\alpha_k(i,l)}\sqrt{\frac{2\pi}{2\alpha_k(i,l)}} \\ P_k &= \frac{1}{S_k}\sum_{i=1}^{2}\sum_{l=1}^{2^k} A_k(i,l)\exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)})\sqrt{\frac{2\pi}{2\alpha_k(i,l)}}\frac{2\alpha_k(i,l)+\beta_k(i,l)^2}{4\alpha_k(i,l)^2} - \hat{x}_k^2\end{aligned} \tag{2.15}$$

where $S_k = \sum_{i=1}^{2}\int_{\mathbb{R}} q_k^i(\zeta)d\zeta$ is computed below and both of $\hat{x}_k$ and P_k are nonlinear functions of measurements.

Proof. As $q_k^i(x) = \sum_{l=1}^{2^k} A_k(i,l)\exp[-\alpha_k(i,l)x^2 + \beta_k(i,l)x]$, we have:

$$\begin{aligned}&S_k = \sum_{i=1}^{2}\int_{\mathbb{R}} q_k^i(\zeta)d\zeta = \sum_{i=1}^{2}\sum_{l=1}^{2^k} A_k(i,l)\exp(\tfrac{\beta_k(i,l)^2}{4\alpha_k(i,l)})\int_{\mathbb{R}}\exp[-\tfrac{1}{2\frac{1}{2\alpha_k(i,l)}}(\zeta - \tfrac{\beta_k(i,l)}{2\alpha_k(i,l)})^2]d\zeta \\ &= \sum_{i=1}^{2}\sum_{l=1}^{2^k} A_k(i,l)\exp(\tfrac{\beta_k(i,l)^2}{4\alpha_k(i,l)})\sqrt{\tfrac{2\pi}{2\alpha_k(i,l)}}\end{aligned}$$

Then, the normalized density can be computed as:

$$p_k^i(x) = \frac{\sum_{l=1}^{2^k} A_k(i,l)\exp[-\alpha_k(i,l)x^2 + \beta_k(i,l)x]}{S_k}$$

Thus, the optimal state estimate is computed as:

$$\begin{aligned}&\hat{x}_k = \mathbb{E}\{x_k|I_k\} = \sum_{i=1}^{2}\int_{\mathbb{R}} x_k^i p_k^i(x_k^i)dx_k^i \\ &= \tfrac{1}{S_k}\sum_{i=1}^{2}\sum_{l=1}^{2^k} A_k(i,l)\int_{\mathbb{R}} x_k^i\exp[-\alpha_k(i,l)(x_k^i)^2 + \beta_k(i,l)x_k^i]dx_k^i \\ &= \tfrac{1}{S_k}\sum_{i=1}^{2}\sum_{l=1}^{2^k} A_k(i,l)\exp(\tfrac{\beta_k(i,l)^2}{4\alpha_k(i,l)})\int_{\mathbb{R}} x_k^i\exp[-\tfrac{1}{2\frac{1}{2\alpha_k(i,l)}}(x_k^i - \tfrac{\beta_k(i,l)}{2\alpha_k(i,l)})^2]dx_k^i \\ &= \tfrac{1}{S_k}\sum_{i=1}^{2}\sum_{l=1}^{2^k} A_k(i,l)\exp(\tfrac{\beta_k(i,l)^2}{4\alpha_k(i,l)})\tfrac{\beta_k(i,l)}{2\alpha_k(i,l)}\sqrt{\tfrac{2\pi}{2\alpha_k(i,l)}}\end{aligned}$$

Similarly for covariance , we have:

$$P_k = \mathbb{E}\{(x_k - \hat{x}_k)^2 | I_k\} = \mathbb{E}\{x_k^2 | I_k\} - \hat{x}_k^2$$

where $E\{x_k^2|I_k\}$ is computed as:

$$\begin{aligned}
&\mathbb{E}\{x_k^2|I_k\} = \sum_{i=1}^{2} \int_{\mathbb{R}} (x_k^i)^2 p_k^i(x_k^i) dx_k^i \\
&= \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \int_{\mathbb{R}} (x_k^i)^2 \exp[-\frac{1}{2\frac{1}{2\alpha_k(i,l)}} (x_k^i - \frac{\beta_k(i,l)}{2\alpha_k(i,l)})^2] dx_k^i \\
&= \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \int_{\mathbb{R}} (x_k^i - \frac{\beta_k(i,l)}{2\alpha_k(i,l)})^2 \exp[-\frac{1}{2\frac{1}{2\alpha_k(i,l)}} (x_k^i - \frac{\beta_k(i,l)}{2\alpha_k(i,l)})^2] dx_k^i \\
&+ \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \int_{\mathbb{R}} \frac{\beta_k(i,l)}{\alpha_k(i,l)} x_k^i \exp[-\frac{1}{2\frac{1}{2\alpha_k(i,l)}} (x_k^i - \frac{\beta_k(i,l)}{2\alpha_k(i,l)})^2] dx_k^i \\
&- \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \int_{\mathbb{R}} (\frac{\beta_k(i,l)}{2\alpha_k(i,l)})^2 \exp[-\frac{1}{2\frac{1}{2\alpha_k(i,l)}} (x_k^i - \frac{\beta_k(i,l)}{2\alpha_k(i,l)})^2] dx_k^i \\
&= \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \sqrt{\frac{2\pi}{2\alpha_k(i,l)}} \frac{1}{2\alpha_k(i,l)} \\
&+ \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \sqrt{\frac{2\pi}{2\alpha_k(i,l)}} \frac{\beta_k(i,l)}{\alpha_k(i,l)} \frac{\beta_k(i,l)}{2\alpha_k(i,l)} \\
&- \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \sqrt{\frac{2\pi}{2\alpha_k(i,l)}} (\frac{\beta_k(i,l)}{2\alpha_k(i,l)})^2 \\
&= \frac{1}{S_k} \sum_{i=1}^{2} \sum_{l=1}^{2^k} A_k(i,l) \exp(\frac{\beta_k(i,l)^2}{4\alpha_k(i,l)}) \sqrt{\frac{2\pi}{2\alpha_k(i,l)}} \frac{2\alpha_k(i,l) + \beta_k(i,l)^2}{4\alpha_k(i,l)^2}
\end{aligned}$$

□

Extension to General Case: One can easily extend the result to matrix case as in [22] and [23]. Assume (2.6) to be a vector version problem where parameters A, B and C are matrices of compatible dimensions, then $q_k^i(x)$ can be represented by:

$$q_k^i(x) = \sum_{l=1}^{2^k} A_k(i,l) \exp[-x^T \alpha_k(i,l) x + \beta_k^T(i,l) x]$$

where ^{T} denotes the transpose, and

$$
\begin{aligned}
A_k(i,l) &= \frac{\pi_{qr} A_{k-1}(q,r)}{|B|\sqrt{2|D_{k-1}(i,l)|}} \exp[\tfrac{1}{2}(\beta_{k-1}^T(q,r)(2D_{k-1}(i,l))^{-1}\beta_{k-1}(q,r))] \\
D_{k-1}(i,l) &= A^T(2BB')^{-1}A + \alpha_{k-1}(q,r), \\
\beta_k(1,l) &= (2BB^T)^{-1}A(D_{k-1}(1,l)^{-1})\beta_{k-1}(q,r), \\
\beta_k(2,l) &= C'y_k + (2BB^T)^{-1}A(D_{k-1}(2,l))^{-1}\beta_{k-1}(q,r) \\
\alpha_k(1,l) &= (2BB^T)^{-1} - (2BB^T)^{-1}A(D_{k-1}(1,l))^{-1}A^T(2BB^T)^{-1} \\
\alpha_k(2,l) &= \frac{C^TC}{2} + (2BB^T)^{-1} - (2BB^T)^{-1}A(D_{k-1}(2,l))^{-1}A^T(2BB')^{-1}
\end{aligned}
$$

The state estimate and the covariance for the vector case can be derived similarly as in Lemma 2.0.2.

Comparing the result obtained here and from [13], we observe that: In [13], the derivation involved the Kalman filter and Bayesian theorem, while in our work, we employed the change of measure method and Bayesian theorem. The difference in the derivation methods results in totally different forms of the solution, where in this work the optimal estimate and error covariance are represented explicitly by several recursive characteristics while in [13], the optimal state estimate is written as a function of the error covariance and previous estimate. Obviously, the solution provided in this work is much more explicit. This form of the solution may provide another insight into deriving the suboptimal estimator to this problem.

Note that from the expression (2.15), we can conclude that the optimal filter is not a linear function but rather an exponential function of measurements. As a result, the state estimate and the covariance require the calculation of a summation from 1 to 2^k where k is time. Since k is increasing, the computation of this optimal filter require exponentially increasing memory (as also mentioned in [13]). This is the price that one has to pay to obtain the optimal filter. However, two important features can be deduced from the optimal filter. The first feature is that the optimal filter can be used as a benchmark for suboptimal filters to determine how far they are from

optimality. The second, and more important feature, is that the optimal filter shows superior performance for unstable systems compared with the linear optimal filter.

2.3 Numerical Examples

In this section, we compare the performance of the linear optimal estimator and the optimal filter.

2.3.1 Comparison between the Optimal Filter and the Linear Optimal Estimator

To illustrate the performance, we consider a linear system with packet arrival probability $p = 0.8$, $A = 0.9$, $B = 0.01$, $h(e_1) = 0, h(e_2) = 1$, and

$$\prod = \begin{pmatrix} 0.20 & 0.80 \\ 0.20 & 0.80 \end{pmatrix}$$

Simulations are performed for 30 runs from $k = 0$ to $k = 15$. There are packet losses at $k = 4, 6, 13$. The performance of our estimator and the linear optimal estimator are compared. It is seen from fig 2.1 that both filters estimate the true system state very well while the root mean square error (RMSE) for the optimal filter is a bit smaller.

To further compare the performance, the system parameter A is increased from 0.9 to 1.14, that is, now the system is unstable. The result of RMSEs is shown in fig 2.2. The figure clearly shows that the performance of the optimal filter is much better since the performance of unstable systems degrades rapidly as the linear optimal estimator diverges [26].

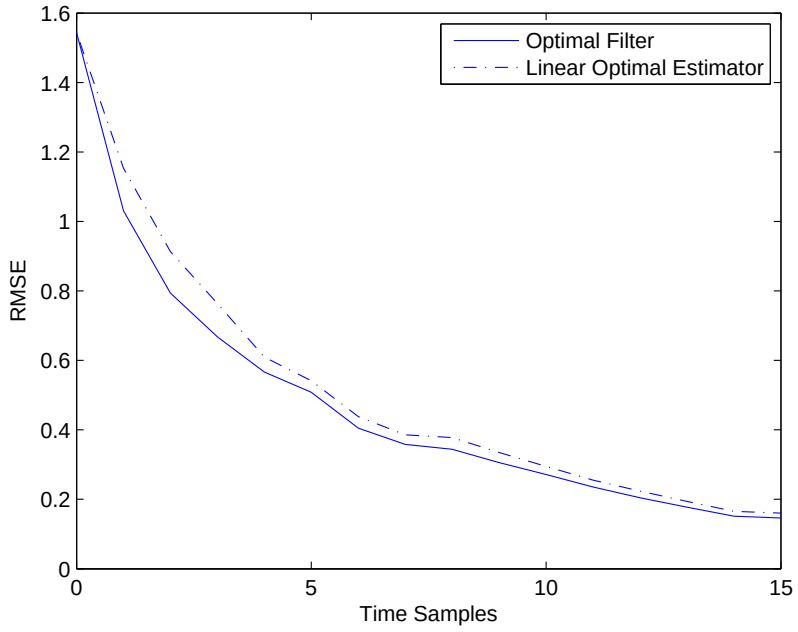

Figure 2.1: Root mean square error between the optimal filter (solid curve) and the linear optimal estimator (dash curve) when $A = 0.9$.

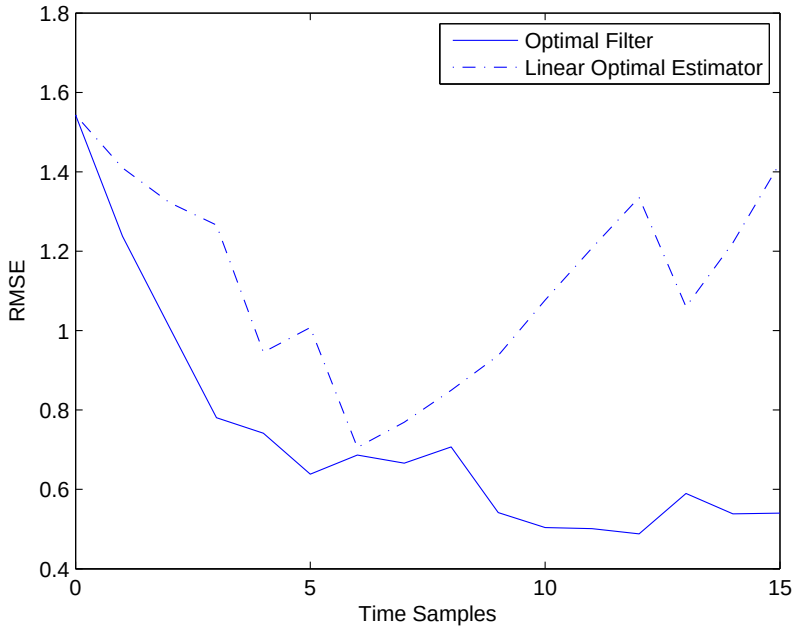

Figure 2.2: Root mean square error between the optimal filter (solid curve) and the linear optimal estimator (dash curve) when $A = 1.14$.

Chapter 3

Estimation and Control over Cognitive Radio Systems Based on a Two-Switch Model

The rapid development of various technologies such as radio, satellite, and phone service, has increased the need for wider channel frequency bandwidth. The current bandwidth spectrum has been licensed to different users to ensure the coexistence of diverse wireless systems [32]. However, the FCC's frequency allocation chart [33] shows that the majority of frequency bandwidth has been assigned to different users and that large portions of the spectrum are frequently unused [34]. CR [35] is proposed for the purpose of the efficient spectrum use.

As noted in Chapter 1, a great deal of research has been performed in the area of control and estimation over communication links under constraints, but limited research has been performed regarding the CR architecture. In this study, we provide comprehensive work in estimation and control over this CR system. As far as we know, our work with the combination of CR and control theory is the first in this area. This chapter employs a two-switch model proposed from the communication community [32] as the model of the CR system. This model avoids the disturbance

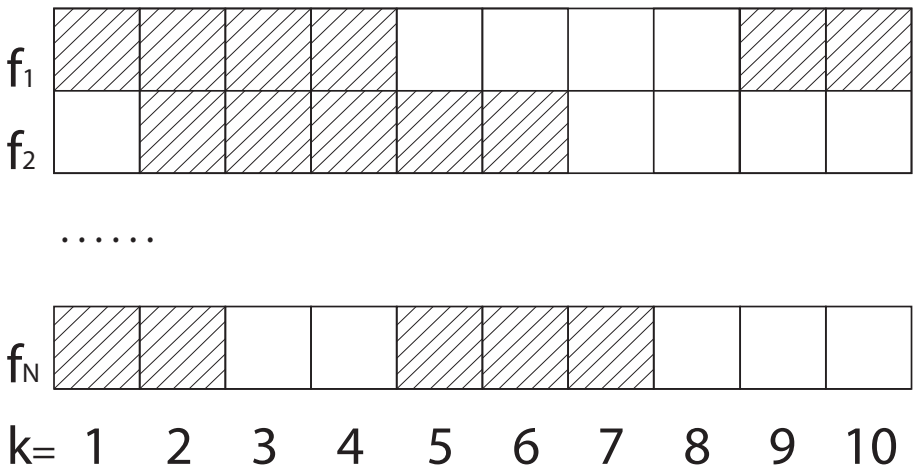

Figure 3.1: Channels in CR system.

from secondary users completely and thus protects the benefit of primary users. Parts of this work have been published in [38] and [37].

3.1 The Two-Switch Model

In this section, we introduce the two-switch model used to model the CR system throughout this chapter.

The general idea of the CR system can be interpreted by fig 3.1. Assume there are $\boldsymbol{N}$ independent licensed channels that can be sensed named as f_1, f_2,..., f_N; each channel is divided into parts by vertical lines and each part represents that channel in one time slot; the marked slot represents that the channel is utilized by PUs and the SUs can not use it at that time while the blank one means that it is free to be used by SUs.

In CR systems, PUs represent the users that occupy the spectrum as they pay or as they are assigned to it. SUs take advantage of inactivity periods of PUs to transmit information through the available channel. SUs have to avoid transmitting to minimize interference with PUs. The model considered in this chapter is proposed in the communication community [32].

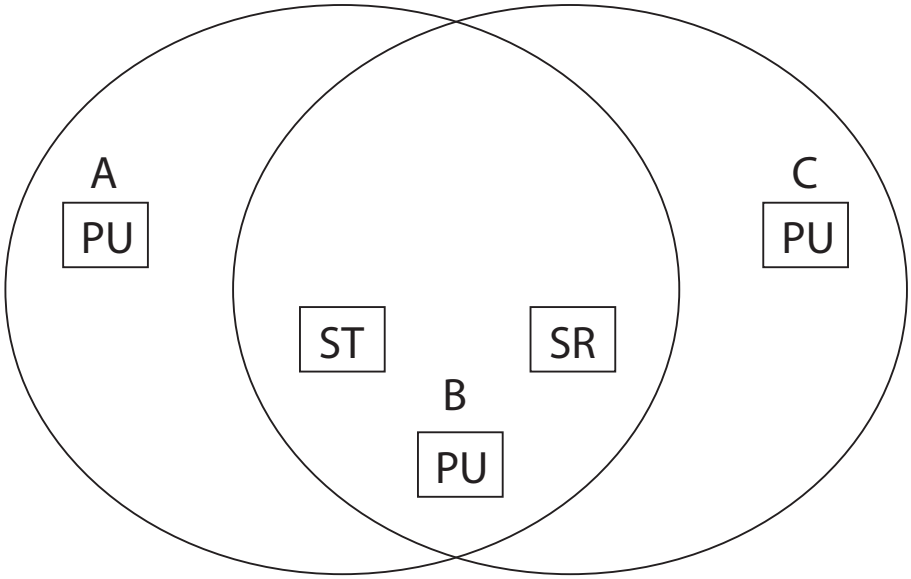

Figure 3.2: Conceptual model of a cognitive radio system with secondary transmitter ST and receiver SR [32].

First, consider the CR link shown in fig 3.2. We assume one secondary transmitter (ST) and one secondary receiver (SR) in the presence of several PUs, e.g. 3 PUs A, B, and C (assume 3 PUs only for convenience). The circles represent sensing regions where the ST and SR can detect activities of PUs. In fig 3.2, for example, the ST can only sense whether A or B is active, and then reports that the spectrum as available for the transmission when both A and B are inactive. Similarly, the SR does the same to B and C.

Due to the independence of each channel, we can consider this as a problem where only one channel in the CR system and design the estimator and controller over it.

The conceptual model in fig 3.2 produces the two-switch mathematical model shown in fig 3.3. Here, we use s_t and s_r to denote sensing variables of the ST and the SR. Let $s_t = 0$ if the ST senses active PUs and $s_t = 1$ if no active PUs. $s_r = 0$ if the SR senses active PUs and $s_r = 1$ if no active PUs. We also assume that PUs are independent with each other. Assume that s_t and s_r are i.i.d Bernoulli variables.

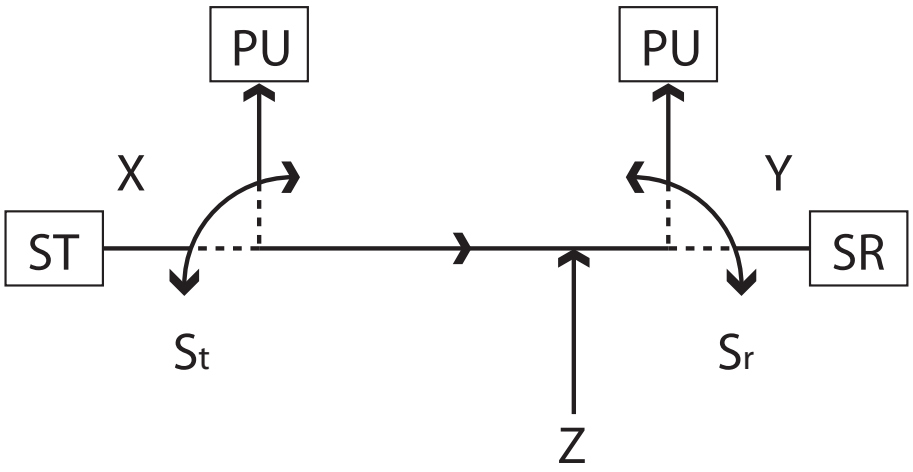

Figure 3.3: Mathematical model of two-switch model [32].

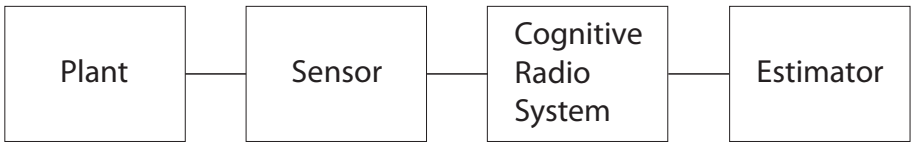

Figure 3.4: Estimation over cognitive radio system.

The switch state s_t is known only to the transmitter, while s_r is known only to the receiver. The correlation which exists between them as can be seen from the fig 3.2. They both depend on PUs that exist in the intersecting region of both sensing regions. The mathematical model can be written as $Y = s_r(s_t X + Z)$, where Y is the received signal at the receiver, and X and Z are the transmitted signal and the noise, respectively.

3.2 Linear Optimal Estimator

In this section, we derive the linear optimal estimator through a single CR system.

3.2.1 Problem Formulation

First, we consider estimation over a single CR link between the sensor and the estimator as shown in fig 3.4. For the following discrete-time system:

$$\begin{aligned} x_k &= Ax_{k-1} + \upsilon_k \\ y_k &= s_r^k(s_t^k C x_k + \omega_k) \end{aligned} \tag{3.1}$$

where $x_k \in \mathbb{R}^n$ is the state at time k, $y_k \in \mathbb{R}^l$ is the observation received at the receiver, x_0 is the initial value of the processes $\{x_k\}_{k\in\mathbb{N}}$. $\upsilon_k \in \mathbb{R}^n$ and $\omega_k \in \mathbb{R}^l$ are independent Gaussian white sequences with zero mean and positive definite covariance matrices V and W. The matrices A and C are system matrices, and (A, C) is observable. s_t^k and s_r^k are switching variables of the ST and the SR at time k, respectively. The CR system is located between the sensor and the estimator. We assume that s_t^k and s_r^k are two i.i.d Bernoulli variables with the probability $\mathbb{P}\{s_t^k = 1\} = \lambda$ and $\mathbb{P}\{s_r^k = 1\} = q$. Note that the two Bernoulli variables are assumed to be independent with the states and noises, but may depend on each other due to the intersection of sensing regions. The switching variable s_t^k is not known while s_r^k is known at the estimator. Denote by $\{I_k\}_{k\in N}$ the complete filtration (σ-algebra) generated by $\{y_0, ..., y_k, s_r^1, ..., s_r^k\}$.

The optimal estimation problem can be posed as the minimization of the cost function:

$$J_k = \mathbb{E}\{(x_k - \hat{x}_{k|k})^T(x_k - \hat{x}_{k|k})|I_k\}$$

3.2.2 Linear Optimal Estimator

In this section, we derive the linear optimal estimator by assuming the state estimate is a linear function of the measurement.

Theorem 3.1. *The linear estimator that minimizes the cost function (3.1) is given by:*

$$\begin{aligned}
\hat{x}_{k|k-1} &= A\hat{x}_{k-1|k-1} && (3.2)\\
P_{k|k-1} &= AP_{k-1|k-1}A^T + V && (3.3)\\
\hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(y_k - s_r^k pC\hat{x}_{k|k-1}) && (3.4)\\
P_{k|k} &= P_{k|k-1} - s_r^k pK_kCP_{k|k-1} && (3.5)\\
K_k &= P_{k|k-1}pC^T(pCP_{k|k-1}pC^T + W')^{-1} && (3.6)\\
W' &= W + (p-p^2)CX_kC^T && (3.7)\\
X_{k+1} &= AX_kA^T + V && (3.8)
\end{aligned}$$

where $p = \mathbb{P}(s_t^k = 1 | s_r^k = 1)$.

Proof. : The prediction step is given by:

$$\hat{x}_{k|k-1} = \mathbb{E}\{x_k | I_{k-1}\} = A\hat{x}_{k-1|k-1}$$

$$P_{k|k-1} = \mathbb{E}\{(x_k - \hat{x}_{k|k-1})(x_k - \hat{x}_{k|k-1})^T | I_{k-1}\} = AP_{k-1|k-1}A^T + V$$

where $\hat{x}_{k|k-1}$ is the a priori state estimate at time k and $\hat{x}_{k-1|k-1}$ is the a posterior state estimate at time $k-1$, $P_{k|k-1}$ is the covariance of the estimation error of $x_k - \hat{x}_{k|k-1}$; $P_{k-1|k-1}$ is the covariance of the estimation error of $x_k - \hat{x}_{k-1|k-1}$.

Assume that the state is a linear function of the measurement:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + \mathbb{E}\{K_k(y_k - s_r^k s_t^k C\hat{x}_{k|k-1}) | I_k\} \tag{3.9}$$

where K_k is the linear optimal estimator gain matrix at time k and $y_k - s_r^k s_t^k C\hat{x}_{k|k-1}$ is the innovation process.

In (3.9) while disregarding the term $s_r^k s_t^k C\hat{x}_{k|k-1}$, other terms do not depend on s_t^k and s_r^k. Thus, (3.9) becomes:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - s_r^k \mathbb{E}\{s_t^k | s_r^k\} C\hat{x}_{k|k-1})$$

since s_r^k is measurable with respect to I_k and s_t^k only depends on s_r^k.

The mean estimator error then is computed as:

$$\begin{aligned}
&\mathbb{E}\{\varepsilon_{x,k|k}|I_k\} = \mathbb{E}\{x_k - \hat{x}_{k|k}|I_k\} = \mathbb{E}\{x_k - \hat{x}_{k|k-1} - K_k(y_k - s_r^k p_t C\hat{x}_{k|k-1})|I_k\} \\
&= (I - s_r^k K_k p_t C)E\{\varepsilon_{x,k|k-1}|I_k\} - s_r^k K_k \mathbb{E}\{\omega_k^{'}|I_k\}
\end{aligned}$$

where $p_t =: \mathbb{E}\{s_t^k|s_r^k\}$; $\omega_k^{'} := \omega_k + (s_t^k - p_t)Cx_k$ is viewed as the new measurement noise. Then, by independence of the state, the noise and s_t^k, we have:

$$\mathbb{E}\{\omega_k^{'}|I_k\} = \mathbb{E}\{\omega_k + (s_t^k - p_t)Cx_k|I_k\} = \mathbb{E}\{\omega_k\} + \mathbb{E}\{s_t^k - p_t|I_k\}\mathbb{E}\{Cx_k|I_k\} = 0$$

Also we have $\mathbb{E}\{\omega_k^{'}\upsilon_k^T\} = 0$. Then the estimation error covariance $P_{k|k}$ at time k is:

$$\begin{aligned}
&P_{k|k} = \mathbb{E}\{\varepsilon_{x,k|k}{\varepsilon^T}_{x,k|k}|I_k\} \\
&= (I - s_r^k K_k p_t C)\mathbb{E}\{\varepsilon_{x,k|k-1}{\varepsilon^T}_{x,k-1}|I_k\}(I - s_r^k K_k p_t C)^T - s_r^k K_k \mathbb{E}\{\omega_k^{'}\varepsilon_{x,k|k-1}^T|I_k\} \\
&\times(I - s_r^k K_k p_t C)^T - (I - s_r^k K_k p_t C)\mathbb{E}\{\varepsilon_{x,k|k-1}\omega_k^{'T}|I_k\}s_r^k K_k^T + s_r^k K_k \mathbb{E}\{\omega_k^{'}\omega_k^{'T}|I_k\}s_r^k K_k^T
\end{aligned}$$

Note that $\varepsilon_{x,k|k-1}$ is the estimation error at time k before receiving the measurement, $\omega_k^{'}$ is combined with the measurement noise ω_k at time k. Thus, $\omega_k^{'}$ is independent of $\varepsilon_{x,k|k-1}$. Therefore, $\mathbb{E}\{\omega_k^{'}\varepsilon_{x,k|k-1}^T\} = \mathbb{E}\{\varepsilon_{x,k|k-1}\omega_k^{'T}\} = 0$ and we have

$$P_{k|k} = (I - s_r^k K_k p_t C)P_{k|k-1}(I - s_r^k K_k p_t C)^T + s_r^k K_k W^{'} K_k^T \tag{3.10}$$

where $W^{'} = \mathbb{E}\{\omega_k^{'}\omega_k^{'T}|I_k\}$ is the variance of $w^{'}$ and is determined by

$$\begin{aligned} W^{'} &= \mathbb{E}\{\omega_k^{'}\omega_k^{'T}|I_k\} = \mathbb{E}\{(\omega_k + (s_t^k - p_t)Cx_k)(\omega_k + (s_t^k - p_t)Cx_k)^T|I_k\} \\ &= W + (\mathbb{E}\{(s_t^k)^2|s_r^k\} - (p_t)^2)CX_kC^T \end{aligned}$$

where $X_k = \mathbb{E}\{x_k x_k^T|I_k\}$.

Following [26] we obtain $X_{k+1} = AX_kA^T + V$, and $X_0 = x_0x_0^T + P_0$ to make $\{X_k\}$ a known sequence.

The optimality criterion is set to minimize the cost function J_k. Note $J_k = Trace(P_{k|k})$ [27]. Differentiating J_k with respect to (w.r.t) K_k yields

$$\begin{aligned} \frac{\partial J_k}{\partial K_k} &= \frac{\partial Trace((I - s_r^k K_k p_t C)P_{k|k-1}(I - s_r^k K_k p_t C)^T + s_r^k K_k W^{'} K_k^T)}{\partial K_k} \\ &= 2(I - s_r^k K_k p_t C)P_{k|k-1}(-s_r^k p_t C^T) + 2s_r^k K_k W^{'} \end{aligned} \tag{3.11}$$

Letting (3.11) be equal to 0, and solving for K_k results:

$$K_k = P_{k|k-1}p_tC^T(p_tCP_{k|k-1}p_tC^T + W^{'})^{-1}$$

Plug K_k back to (3.10)

$$P_{k|k} = P_{k|k-1} - s_r^k p_t K_k C P_{k|k-1} \tag{3.12}$$

Next p_t is computed. As $s_t^k \in \{0, 1\}$, $p_t = 1 \times \mathbb{P}(s_t^k = 1|s_r^k) + 0 \times \mathbb{P}(s_t^k = 0|s_r^k) = \mathbb{P}(s_t^k = 1|s_r^k)$, which includes are two cases: $s_r^k = 0$ and $s_r^k = 1$. Note when $s_r^k = 0$, the receiver is closed, so $y_k = 0$. Then, the second term on the right hand side in both (3.9) and (3.12) vanishes, which means p_t does not affect the estimation algorithm when $s_r^k = 0$. Thus, we only need to compute $p_t = \mathbb{P}(s_t^k = 1|s_r^k = 1)$ and (3.9) can be represented as:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - s_r^k \mathbb{P}(s_t^k = 1|s_r^k = 1)C\hat{x}_{k|k-1})$$

which includes both cases.

Similarly, in $W^{'}$, we have $\mathbb{E}\{(s_t^k)^2|s_r^k\} = p(s_t^k = 1|s_r^k)$. Using the same argument as above, we can write:

$$\mathbb{E}\{(s_t^k)^2|s_r^k\} - (p_t)^2 = \mathbb{P}(s_t^k = 1|s_r^k = 1) - \mathbb{P}(s_t^k = 1|s_r^k = 1)^2$$

including both cases. For convenience, we denote $p = \mathbb{P}(s_t^k = 1|s_r^k = 1)$ and finish the proof. □

In the next lemma, we compute p in the linear optimal estimator.

Lemma 3.1.1. *Assume in the two-switch model, there are n independent PUs $\{u_1, ...u_n\}$ in the sensing region of ST only (not in the sensing region of the SR), and another m independent PUs $\{u_{n+1}, ...u_{n+m}\}$ in the intersection of sensing regions of both ST and SR, and another o independent PUs $\{u_{n+m+1}, ...u_{n+m+o}\}$ in the sensing region of the SR only. Let the sequence $\{p_1, ..., p_{n+m+o}\}$ denote probabilities that PUs are inactive respectively. Then,*

$$p = \mathbb{P}(s_t^k = 1|s_r^k = 1) = \prod_{i=1}^{n} p_i \tag{3.13}$$

(3.13)) means that the linear optimal estimator depends on probabilities of inactive PUs that only exist in the sensing region of the ST.

Proof. Note that: $\mathbb{P}(s_t^k = 1) = \prod_{i=1}^{n+m} p_i$, $\mathbb{P}(s_r^k = 1) = \prod_{i=n+1}^{n+m+o} p_i$ Then,

$$\mathbb{P}(s_t^k = 1|s_r^k = 1) = \frac{\mathbb{P}(s_t^k = 1, s_r^k = 1)}{\mathbb{P}(s_r^k = 1)} = \frac{\prod_{i=1}^{n+m+o} p_i}{\prod_{i=n+1}^{n+m+o} p_i} = \prod_{i=1}^{n} p_i$$

□

Note that (i) A must be stable to guarantee the convergence of the estimator due to the term X_k in (3.8); (ii) $P_{k+1|k}$ is a random variable that depends on s_r^k.

3.2.3 Example: Application to An Inverted Pendulum-Cart System

To illustrate the performance of the linear optimal estimator, an application to estimate states of an inverted pendulum-cart system via the CR system is performed.

Parameters of the system, a stable inverted pendulum-cart system, are given by:

$$A = \begin{bmatrix} 1.0000 & -0.0002 & 0.0010 & -0.0000 \\ 0.0000 & 0.9996 & 0.0001 & 0.0010 \\ 0.0315 & -0.3901 & 1.0518 & 0.0417 \\ 0.0726 & -0.8763 & 0.1193 & 0.9038 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, W = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}, V = \begin{bmatrix} 0.0100 & 0.0090 & 0.0020 & 0.0050 \\ 0.0060 & 0.0100 & 0.0080 & 0.0060 \\ 0.0040 & 0.0080 & 0.0030 & 0.0070 \\ 0.0090 & 0.0040 & 0.0050 & 0.0100 \end{bmatrix}$$

Here $x = [s;\ \theta;\ v;\ \omega]^T$ is the state vector, with s the position of the cart; θ the angle of the pendulum with the vertical line; v the velocity of the cart; and ω the angular velocity of the pendulum.

The output signals are given by the position and the angle of the inverted pendulum. The position should act as the reference signal at approximately 1m and the angle should be around 0. The two-switch model CR system as in fig 3.2 is considered. Assume that three PUs are detected in sensing regions, and $p_1 = p_2 = p_3 = 0.8$. The estimates of the position and the angle from the proposed estimator is shown in fig 3.5. From the figure, it is obvious that the estimated states converge to real ones.

3.2.4 Multi-channels Case

In the previous section, we discuss only one channel sensed in the CR system. Due to the independence of each channel, it can be easily extended from single channel case to multi-channels. The term needed to compute in the estimation algorithm is

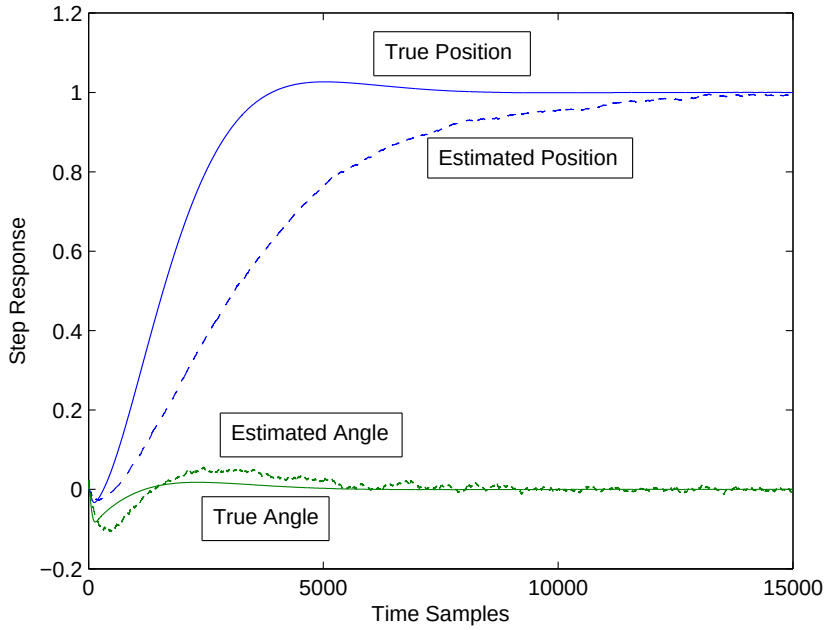

Figure 3.5: State estimates of the position and the angle.

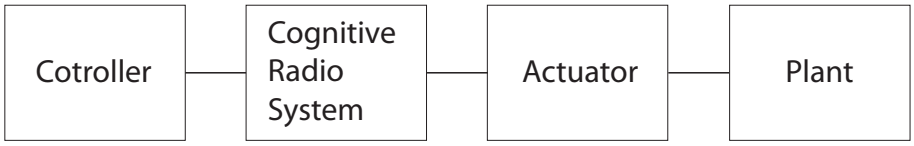

Figure 3.6: Control over cognitive radio system.

$\mathbb{E}\{s_t^k|s_r^k\} = p_t$, and based on the same arguments in section 3.2.2, we have $\mathbb{E}\{s_t^k|s_r^k\} = \mathbb{P}\{s_{t_i}^k|s_{r_i}^k\}$, where $s_{t_i}^k$ and $s_{r_i}^k$ are switching states of the ST and the SR under the ith channel.

3.3 Controller Design Through the CR System

In this section, the linear optimal controller over a single CR link between the controller and the actuator shown in fig 3.6 is derived.

The system is modeled as:

$$x_{k+1} = Ax_k + Bs_r^k(s_t^k u_k + \upsilon_k) \tag{3.14}$$

The linear state feedback controller is computed as the minimizer of the quadratic cost function:

$$V_N = \frac{1}{N}\mathbb{E}\{x_N^T Q_N x_N + \sum_{k=0}^{N-1}(x_k^T Q_k x_k + u_k^T R_k u_k)\}$$

where Q_k and R_k are respectively positive semi-definite and positive definite. The linear controller is:

$$u_k = u(x_k) = L_k x_k$$

Similarly as in [4] using dynamic programming, we get the gain matrix L_k:

$$L_k = -\Psi_k \mathbb{E}\{s_r^k s_t^k | U_k\} B^T H_{k+1} A$$

and

$$\Psi_k = (R_k + \mathbb{E}\{s_r^k s_t^k B^T H_{k+1} B s_r^k s_t^k | U_k\})^{-1}$$

where the matrices H_k are given by the recursive equation:

$$H_N = Q_N$$

$$H_k = A^T H_{k+1} A - A^T H_{k+1} B \mathbb{E}\{s_r^k s_t^k | U_k\} \Psi_k \mathbb{E}\{s_r^k s_t^k | U_k\} B^T H_{k+1} A + Q_k \tag{3.15}$$

L_k is a function of s_r^k and s_t^k. Note that in (3.14) s_t^k is known and the information set $\{U_k\}_{k\in N}$ is the complete filtration generated by $\{x_0, ..., x_k, s_t^1, ..., s_t^k\}$. Therefore, when $s_t^k = 0$, it is obvious that $u_k = 0$ as $L_k = 0$. Hence to compute the control signal, we only need to consider $s_t^k = 1$. The problem reduces to:

$$x_{k+1} = Ax_k + Bs_r^k(u_k + \upsilon_k) \tag{3.16}$$

The equation above is similar to a packet loss model without arrival information. The linear optimal controller based on (3.16) can be derived similarly as [29]. The Riccati-like equation (3.15) with $s_t^k = 1$ has the following form:

$$H_{k+1} = A^T H_k A - \alpha^2 A^T H_k B(R + \alpha B^T H_k B)^{-1} B^T H_k A + Q_k$$

where $\alpha = \mathbb{P}(s_r^k = 1 | s_t^k = 1)$ and is computed similarly to the derivation in Lemma 3.1.1. The stability conditions are discussed in [29].

Next, we present an example to show the performance of the linear optimal controller under the CR link. Consider fig 3.2 again as an example, and the instable inverted pendulum-cart system has parameters: $A = \begin{bmatrix} 1.0000 & 0.0000 & 0.0010 & -0.0000 \\ 0.0000 & 1.0000 & -0.0000 & 0.0010 \\ 0.0000 & 0.0022 & 0.9842 & -0.0000 \\ 0.0000 & 0.0278 & -0.0363 & 0.9999 \end{bmatrix}$, B $=[0.0000 \ \ , 0.0000 \ \ , 0.0023 \ \ , 0.0052]^T$, $p_1 = 0.9, p_2 = 0.8$, $p_3 = 0.5$ and the LQR gain for a deterministic system $x_{k+1} = Ax_k + Bu_k$ where $u_k = -Gx_k$ is:

$$G = [-13.9382173.6752 - 29.903018.4750]$$

The infinite horizon linear optimal controller gain computed by the proposed algorithm is:

$$K = [-19.6887, 217.4028, -56.7957, 32.8859]^T$$

Fig 3.7 and fig 3.8 compare the LQR gain and the new controller gain using the step response of the system. As can be seen from fig 3.7, the position controlled by the new controller converges at 1m while it diverges for the standard LQR controller. Also in fig 3.8 the new controller stabilizes the system by forcing the angle to be near 0, while the standard LQR controller makes the angle oscillates away from 0.

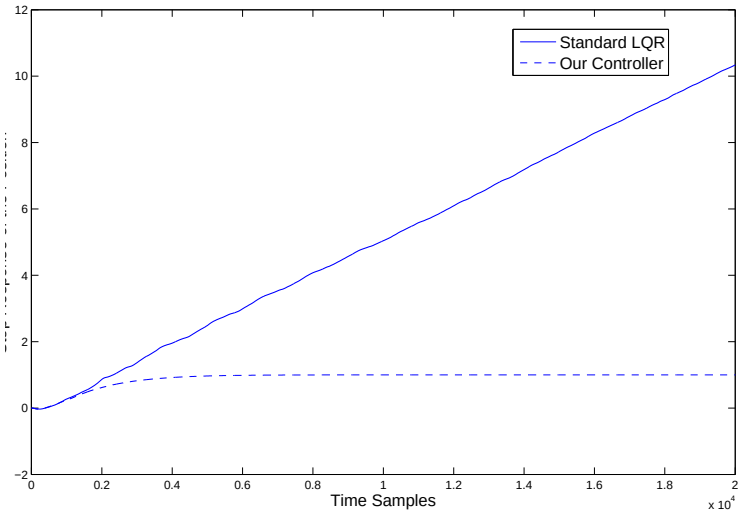

Figure 3.7: Step response of the position with comparison.

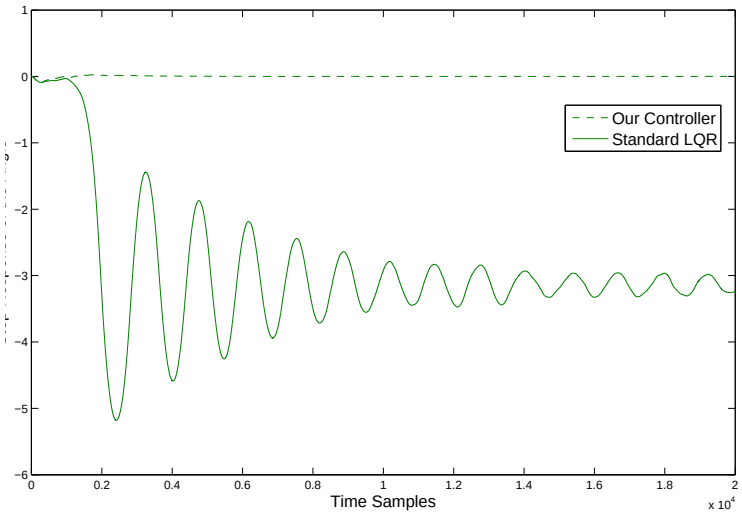

Figure 3.8: Step response of the angle with comparison.

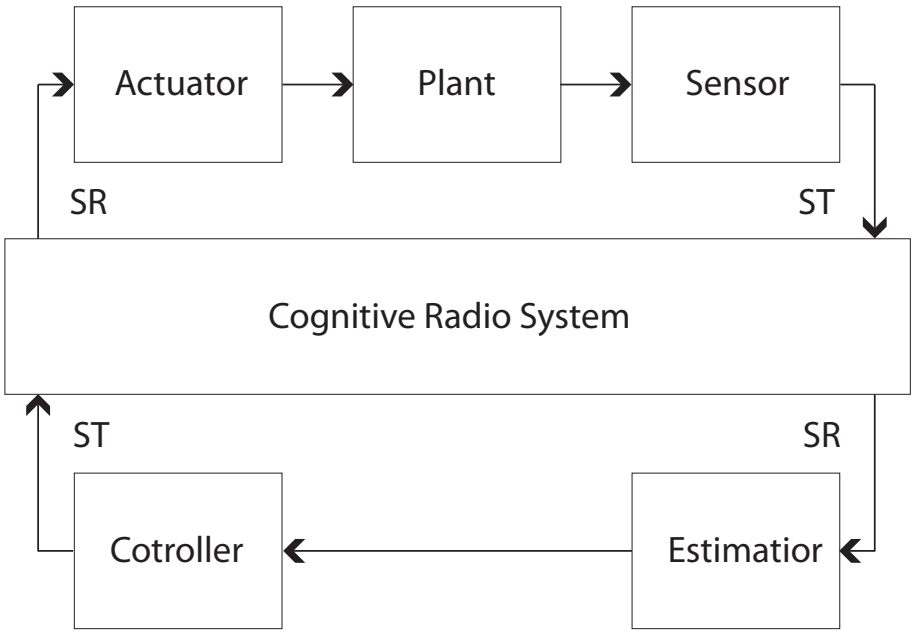

Figure 3.9: Closed-Loop system.

3.4 Estimation and Control Through Cognitive Radio

3.4.1 Estimation

In this section, we consider estimation and control of the closed-loop system when CR links exist between both the sensor to the estimator and the controller to the actuator, as shown in fig 3.9. There are two STs, located at the sensor and the controller ends, respectively, similarly two SRs at estimator and actuator ends. Observe that the sensing variables are the same for the receiver at the estimator and the transmitter at the controller, thus for convenience we use s_r^k to denote sensing variables of the transmitter at the controller and receiver at the estimator. Similarly, we use s_t^k for the receiver at the actuator and the transmitter at the sensor.

The system described in fig 3.9 becomes:

$$\begin{aligned} x_{k+1} &= Ax_k + Bs_t^k(s_r^k u_k + \upsilon_k) \\ y_k &= s_r^k(s_t^k C x_k + \omega_k) \end{aligned} \tag{3.17}$$

The linear optimal estimator for this system is to minimize the cost function defined in section 3.2.1 by assuming that the state estimate is a linear combination of measurements.

The a priori state estimate can be computed similarly as follows:

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k} + ps_r^k B u_k \\ P_{k+1|k} &= AP_{k|k}A^T + p(1-p)s_r^k B u_k u_k^T B^T + p_d B V B^T \end{aligned} \tag{3.18}$$

where $p_d = p$ when $s_r^k = 1$ and $p_d = \mathbb{P}(s_t^k = 1 | s_r^k = 0)$ when $s_r^k = 0$, the latter probability can be computed similar as lemma 3.1.1 and $\mathbb{P}(s_t^k = 1 | s_r^k = 0) = (\prod_{i=1}^{n+m} p_i(1 - \prod_{n+m+1}^{n+m+o} p_i))/(1 - \prod_{n+1}^{n+m+o} p_i)$.

After receiving the measurement we can obtain the a posterior state estimate:

$$\begin{aligned} \hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K_{k+1}(y_{k+1} - s_r^{k+1} pC\hat{x}_{k+1|k}) \\ W_{k+1}{}' &= W + (p - (p)^2)CX_{k+1}C^T \\ P_{k+1|k+1} &= P_{k+1|k} - s_r^{k+1} pK_{k+1}CP_{k+1|k} \\ K_{k+1} &= P_{k+1|k}pC^T(pCP_{k+1|k}pC^T + W_{k+1}{}')^{-1} \\ X_{k+1} &= \mathbb{E}\{x_{k+1}x_{k+1}^T\} \end{aligned} \tag{3.19}$$

After additional computations, we have:

$$X_{k+1} = \mathbb{E}\{x_{k+1}x_{k+1}^T | I_k\} = \mathbb{E}\{(Ax_k + Bs_t^k(s_r^k u_k + \upsilon_k))(Ax_k + Bs_t^k(s_r^k u_k + \upsilon_k))^T | I_k\}$$
$$= (A\hat{x}_{k|k} + ps_r^k B u_k)(A\hat{x}_{k|k} + ps_r^k B u_k)^T + P_{k+1|k}$$

due to $\mathbb{E}\{e\hat{x}^T\} = 0$ and $\mathbb{E}\{eu^T\} = \mathbb{E}\{e\}u^T = 0$.

Note that the error covariance in (3.19) can be written in the following form:

$$P_{k+1} = AP_kA^T - s_r^k pAK_kCP_kA^T + p(1-p)s_r^k Bu_k u_k^T B^T + p_d BVB^T \tag{3.20}$$

3.4.2 Control

In this section, we discuss control design of the closed-loop system. We can see above that the error covariance is a function of the control input, which implies that the separation principle does not hold. We give an example to illustrate that it is indeed the case. Assume a SISO system with $A = 1$, $B = 1$, $C = 1$, $W = 1$ and $V = 0$. Consider the value function defined as:

$$\begin{aligned} V_N(x_N) &= \mathbb{E}\{{x_N}^T Q_N x_N | I_N\} \\ V_k(x_k) &= \min_{u_k} \mathbb{E}\{{x_k}^T Q_k x_k + s_t^k s_r^k {u_k}^T R_k u_k + V_{k+1}(x_{k+1}) | I_k\} \end{aligned} \tag{3.21}$$

Also assume $Q_N = Q_k = 1$ and $R = 0$.

When $k = N$, $V_N(x_N) = \mathbb{E}\{{x^2}_N | I_N\}$. When $k = N-1$,

$$\begin{aligned} &V_{N-1}(x_{N-1}) = \min_{u_{N-1}} \mathbb{E}\{({x_{N-1}}^2 + V_N(x_N)) | I_{N-1}\} \\ &= \min_{u_{N-1}} \mathbb{E}\{(2{x_{N-1}}^2 + 2s_t^{N-1} s_r^{N-1} u_{N-1} x_{N-1} + s_t^{N-1} s_r^{N-1} {u_{N-1}}^2) | I_{N-1}\} \\ &= \mathbb{E}\{(2{x_{N-1}}^2) | I_{N-1}\} + \min_{u_{N-1}} \{2ps_r^{N-1} u_{N-1} \hat{x}_{N-1|N-1} + ps_r^{N-1} {u_{N-1}}^2\} \end{aligned}$$

To compute the control action for step $k = N-1$ we need to differentiate the above equation on both sides with respect to u_{N-1}: $\frac{\partial V_{N-1}}{\partial u_{N-1}} = 0 \Rightarrow u^*_{N-1} = -\hat{x}_{N-1|N-1}$ and plug back to above equation we get:

$$\begin{aligned} V_{N-1}(x_{N-1}) &= \mathbb{E}\{(2{x_{N-1}}^2) | I_{N-1}\} - s_r^{N-1} p \hat{x^2}_{N-1|N-1} \\ &= \mathbb{E}\{(2 - s_r^{N-1} p){x_{N-1}}^2 | I_{N-1}\} + s_r^{N-1} p P_{N-1|N-1} \end{aligned} \tag{3.22}$$

When $k = N-2$, we have:

$$V_{N-2}(x_{N-2}) = \min_{u_{N-2}} \mathbb{E}\{({x_{N-2}}^2 + V_{N-1}(x_{N-1}))|I_{N-2}\} = \mathbb{E}\{(3 - s_r^{N-1}p){x_{N-2}}^2|I_{N-2}\}$$
$$+pq(1-q)P_{N-2|N-2} + \frac{q^2}{p} + \min_{u_{N-2}}\{2ps_r^{N-2}(2-pq)\hat{x}_{N-2|N-2} + ps_r^{N-2}(2-pq){u_{N-2}}^2$$
$$+p^2q(1-p)(1-q)s_r^{N-2}{u_{N-2}}^2 - q^2(1-p)X_{N-1} + \frac{2q^2W'(pP_{N-2|N-2}+p^2(1-p)s_r^{N-2}{u_{N-2}}^2)}{D}$$
$$+\frac{q^2W'}{pD}\}$$

where $W' = W + p(1-p)X_{N-1}$, $D = p^2(P_{N-2|N-2} + p(1-p)s_r^{N-2}{u_{N-2}}^2) + W'$, $X_{N-1} = (\hat{x}_{N-2|N-2} + ps_r^{N-2}u_{N-2})(\hat{x}_{N-2|N-2} + ps_r^{N-2}u_{N-2})^T + P_{N-1|N-2}$.

It is obvious from the minimization of the cost function V_{N-2}, when $p \neq 1$, the optimal control action u^*_{N-2} is a nonlinear function of the state estimate $\hat{x}_{N-2|N-2}$. Moreover, it is a function of the error covariance $P_{N-2|N-2}$ which means the separation principle does not hold.

The only case where the optimal controller is a linear gain of the state estimate is when $p = 1$, which means that no PU exist in the transmitter sensing region. Using fig 3.2 as an example, when the PU A does not exist or exists in the intersection of both sensing regions, there is an optimal controller that is a linear function of the state estimate. This provides us with an interesting insight: In order to obtain the optimal controller in the linear function of the state estimate, the receiver should be located at a position where all PUs are covered by its sensing region.

3.4.3 Some Discussion of the Closed-Loop System Stability

As seen above, the optimal controller depends on the estimation error covariance and is in fact a nonlinear function of the state estimate. Therefore it is not obvious to study in details the stability of the closed-loop system without an explicit expression of the controller. To simplify the problem, we assume a suboptimal controller that is a linear function of the state estimate, such as $u_k = -F\hat{x}_{k|k-1}$, where F is a constant matrix that is chosen such that u_k stabilizes the original system $x_{k+1} = Ax_k + Bu_k$. We are going to derive stability conditions of the closed-loop system through this

linear controller. Note in this case the error covariance is still a function of the control input. State equations of the closed-loop system are derived as:

$$\begin{aligned} \hat{x}_{k+1} &= (A - ps_r^k BF - pAK_k s_r^k C)\hat{x}_k + AK_k s_r^k s_t^k C x_k + AK_k s_r^k \omega_k \\ x_{k+1} &= Ax_k - s_r^k s_t^k BF\hat{x}_k + Bs_t^k \upsilon_k \end{aligned} \tag{3.23}$$

where $\hat{x}_{k+1} := \hat{x}_{k+1|k}$.

Define $e_{k+1} := \varepsilon_{x,k+1|k} = x_{k+1} - \hat{x}_{k+1|k}$. Subtracting the equations above and incorporating them in the closed-loop system, we have:

$$\begin{aligned} \begin{bmatrix} e_{k+1} \\ \hat{x}_{k+1} \end{bmatrix} &= \begin{bmatrix} A - AK_k s_r^k s_t^k C & (AK_k C + BF)s_r^k(p - s_t^k) \\ AK_k s_r^k s_t^k C & A - ps_r^k BF - AK_k C s_r^k (p - s_t^k) \end{bmatrix} \times \begin{bmatrix} e_k \\ \hat{x}_k \end{bmatrix} \\ &+ \begin{bmatrix} Bs_t^k & -AK_k s_r^k \\ 0 & AK_k s_r^k \end{bmatrix} \begin{bmatrix} \upsilon_k \\ \omega_k \end{bmatrix} \end{aligned} \tag{3.24}$$

The conditions for the mean stability [26] of the closed-loop system are given in the following Theorem.

Theorem 3.2. *The closed-loop system equation (3.24) is m-stable (mean stable) if the following conditions are satisfied:*
(i) $|\rho(A - pqBF)| < 1$*;*
(ii) $|\rho(A - pqA\tilde{K}_k C)| < 1, \forall N$*, for all* $k \geq N$*.*
where $\rho(Z)$ *represent the spectral radius of the matrix* Z *and* $\tilde{K}_k = \mathbb{E}\{K_k\}$*,where* K_k *is a function of* $\{s_r^1, ..., s_r^{k-1}\}$ *computed in section 3.2.*

Proof. By the definition of the mean stability [26], taking the expectation of both sides of (3.24) and we get:

$$\mathbb{E}\left\{ \begin{bmatrix} e_{k+1} \\ \hat{x}_{k+1} \end{bmatrix} \right\} = \begin{bmatrix} A - pqA\tilde{K}_k C & 0 \\ pqA\tilde{K}_k C & A - pqBF \end{bmatrix} \mathbb{E}\left\{ \begin{bmatrix} e_k \\ \hat{x}_k \end{bmatrix} \right\} \tag{3.25}$$

where $\tilde{K}_k$ comes from $\mathbb{E}\{K_k s_r^k s_t^k\} = pq\mathbb{E}\{K_k\} = pq\tilde{K}_k$.

From (3.25) we get so that the m-stability conditions (i) and (ii) of Theorem 3.2. □

Remark 1. *: Condition (i) can be used as a necessary condition for the stability of F when p, q are known. It provides a way to update the suboptimal linear controller for known p, q, with a new gain $\tilde{F}$ which stabilizes $x_{k+1} = Ax_k + pqBu_k$. We show in examples that this new gain will improve the performance of the closed-loop system. In condition (ii), $\{\tilde{K}_k\}$ is a deterministic time varying sequence. It can be computed as follows:*

$$\begin{aligned}\mathbb{E}\{K_k\} &= q\mathbb{E}\{K_k|\, s_r^{k-1} = 1\} + (1-q)\mathbb{E}\{K_k|\, s_r^{k-1} = 0\} \\ &= q\mathbb{E}\{\tilde{K}_k^1\} + (1-q)\mathbb{E}\{\tilde{K}_k^0\}\end{aligned}$$

where $\tilde{K}_k^1$ and $\tilde{K}_k^0$ are functions of $\{s_r^0, s_r^1, ..., s_r^{k-2}\}$ and can be computed by plugging $s_r^{k-1} = 1$ and $s_r^{k-1} = 0$ back into the estimator equations (3.8). Similarly, $\mathbb{E}\{\tilde{K}_k^1\} = q\mathbb{E}\{\tilde{K}_k^1|s_r^{k-2} = 1\ \} + (1-q) \times \mathbb{E}\{\tilde{K}_k^1|s_r^{k-2} = 0\}$, $\mathbb{E}\{\tilde{K}_k^0\} = q\mathbb{E}\{\tilde{K}_k^0|\, s_r^{k-2} = 1\} + (1-q) \times \mathbb{E}\{\tilde{K}_k^0|s_r^{k-2} = 0\}$, and so on. Thus, through the same deduction from s_r^{k-1} to s_r^0 and apply equations (3.8), $\tilde{K}_k$ is obtained.

Similarly, we can obtain the mean square stability in the next theorem.

Theorem 3.3. *The closed-loop system equation (3.24) is ms-stable (mean square stable) if and only if $|\rho(\Phi_k)| < 1$, $\exists N$, for all $k \geq N$, where*

$$\Phi_k = \begin{bmatrix} \Phi_k^1 & \Phi_k^2 \\ \Phi_k^3 & \Phi_k^4 \end{bmatrix} \tag{3.26}$$

and $\Phi_k^i, i = 1, 2, 3, 4$ are given in the Appendix A.2.1.

Proof. Let

$$G_k = \begin{bmatrix} A - AK_k s_r^k s_t^k C & (AK_k C + BF)s_r^k(p - s_t^k) \\ AK_k s_r^k s_t^k C & A - ps_r^k BF - AK_k C s_r^k(p - s_t^k) \end{bmatrix}$$

Then, the ms-stability follows from $|\rho(\Phi_k)| = |\rho(\overline{G_k \otimes G_k})|$ by [26]. □

Next we turn to a special but simplified case. This is the case when $p = 1$, (3.17) becomes:

$$\begin{aligned} x_{k+1} &= Ax_k + Bs_r^k u_k + Bs_i^k \upsilon_k \\ y_k &= s_r^k(Cx_k + \omega_k) \end{aligned} \tag{3.27}$$

where s_i^k represents whether PUs in the intersection region of both sensing regions are active or not. The problem then becomes a packet loss problem that has been considered in [6, 30]. However, the calculation of the optimal controller needs the exact value of q which is difficult to predict in CR systems as it is governed by the PUs' behavior. Next, we will give sufficient conditions of the peak covariance process which can be viewed as an estimate of filtering deterioration caused by disruptions from PUs. First, we introduce the following definition.

Definition 1. *Assume that (A, B) is controllable, and (A, C) is observabile, the observability and controllability index are the smallest integer I_0 and I_1 such that $[C', A'C', ..., (A^{I_0-1})'C']$ and $[B, AB, ..., (A^{I_1-1})B]$ have rank n, respectively.*

When $p = 1$, (3.24) becomes:

$$\begin{bmatrix} e_{k+1} \\ \hat{x}_{k+1} \end{bmatrix} = \begin{bmatrix} A - AK_k s_r^k C & 0 \\ AK_k s_r^k C & A - s_r^k BF \end{bmatrix} \begin{bmatrix} e_k \\ \hat{x}_k \end{bmatrix} + \begin{bmatrix} Bs_i^k & -AK_k s_r^k \\ 0 & AK_k s_r^k \end{bmatrix} \begin{bmatrix} \upsilon_k \\ \omega_k \end{bmatrix}$$

Let $L_{k+1} = \mathbb{E}\left\{ \begin{bmatrix} e_{k+1} \\ \hat{x}_{k+1} \end{bmatrix} \begin{bmatrix} e_{k+1} \\ \hat{x}_{k+1} \end{bmatrix}^T | I_k \right\}$. Assume the initial condition $s_r^1 = 1$. The following two stopping times are introduced [7]:

$$\alpha_1 = inf\{k : k > 1, s_r^k = 0\}.$$
$$\beta_1 = inf\{k : k > \alpha_1, s_r^k = 1\}.$$

Thus, α_1 is the first time when primary users occur and β_1 is the first time the channel becomes idle again. The above procedure then generates two sequences:

$$\alpha_1,\ \alpha_2, ...,\ \alpha_n, ...$$
$$\beta_1,\ \beta_2, ...,\ \beta_n, ...$$

where for $j > 1$:

$$\alpha_j = inf\{k : k > \beta_{j-1}, s_r^k = 0\}.$$
$$\beta_j = inf\{k : k > \alpha_j, s_r^k = 1\}.$$

Denote $L_n^p = L_{\beta_n}$, and $\{L_n^p\}_{n\geq 1}$ is called as the peak covariance process (also a subsequence process) of $\{L_k\}_{k\geq 1}$ [7]. The peak covariance is computed at the last time instant of a consecutive $s_r^k = 0$. The stability analysis of it is important and useful for analyzing the system performance in that it provides an insight that due to successive packet losses, how "bad" the covariance process might be.

Definition 2. *[7] We say the peak covariance sequence $\{L_n^p\}_{n\geq 1}$ is stable if*

$$\sup_{n\geq 1} \mathbb{E} \parallel L_n^p \parallel < \infty$$

Accordingly, we say the system satisfies peak covariance stability.

Lemma 3.3.1. *$\{L_n^p\}_{n\geq 1}$ is stable if the following two conditions hold:*
(i) $q \geq 1 - \frac{1}{\max_i |\lambda_i(A)|^2}$
(ii) $(1-q)qd_1^{(1)}[1 + \sum_{i=1}^{I-1} d_i^{(1)} q^i] \sum_{j=1}^{\infty} \parallel A^j \parallel^2 (1-q)^{j-1} < 1$
where λ_A is an eigenvalue of the largest magnitude for matrix A, and $I = \max\{I_0, I_1\}$ and $d_i^{(1)}$ is a positive constant given in the proof in Appendix A.1.1.

Proof. See Appendix A.1.1. □

Remark 2. *: Lemma 3.3.1 gives sufficient conditions for a linear gain to stabilize the system $x_{k+1} = Ax_k + Bu_k$ in the case the optimal controller cannot be obtained (since the exact value of q is needed to compute the optimal controller).*

3.4.4 Numerical Examples

In this section, we perform simulations to show improved performance of the closed-loop system through CR links and test stability conditions.

We still consider the model of the CR system as shown in fig 3.3, and instable inverted pendulum-cart system parameters are:

$$A = \begin{bmatrix} 1.0000 & 0.0000 & 0.0010 & -0.0000 \\ 0.0000 & 1.0000 & -0.0000 & 0.0010 \\ 0.0000 & 0.0022 & 0.9842 & -0.0000 \\ 0.0000 & 0.0278 & -0.0363 & 0.9999 \end{bmatrix}, \text{B} = [0.0000 \ , 0.0000 \ , 0.0023 \ , 0.0052]^T,$$

First, assume $p_1 = p_2 = p_3 = 0.8$. The controller is an LQR controller to the linear deterministic system: $x_{k+1} = Ax_k + Bu_k$ where $u_k = -Fx_k$:

$$F = [-13.9382 \ \ 173.6752 \ \ -29.9030 \ \ 18.4750]$$

We can see the step response is satisfactory in fig 3.10. By fixing F and changing $p_1 = 0.5$, we can see the step response diverges in fig 3.11.

Next, we set $p_1 = 0.5, p_2 = p_3 = 0.8$, but designing an LQR controller for the system $x_{k+1} = Ax_k + pqBu_k$ as suggested, and running the step response for the closed-loop system, produces fig 3.12. This improved design shows that the step response of the system is stable.

Lastly, we would like to prove Lemma 3.3.1. Set $p_1 = 1, p_2 = 0.7, p_3 = 0.8$. Here we have $I = I_1 = I_0 = 4$ and $||F^i(P)|| \leq ||A^i A^{iT}||||P||$ and $||G^i(M)|| \leq ||A^i A^{iT}||||M||$. Thus, we take $d_1^{(1)} = 1.0395$, $d_2^{(1)} = 1.0805$, $d_3^{(1)} = 1.1230$. After some computations, the left-hand side of condition (ii) in Lemma 3.3.1 is approximately $0.9997 \leq 1$ and

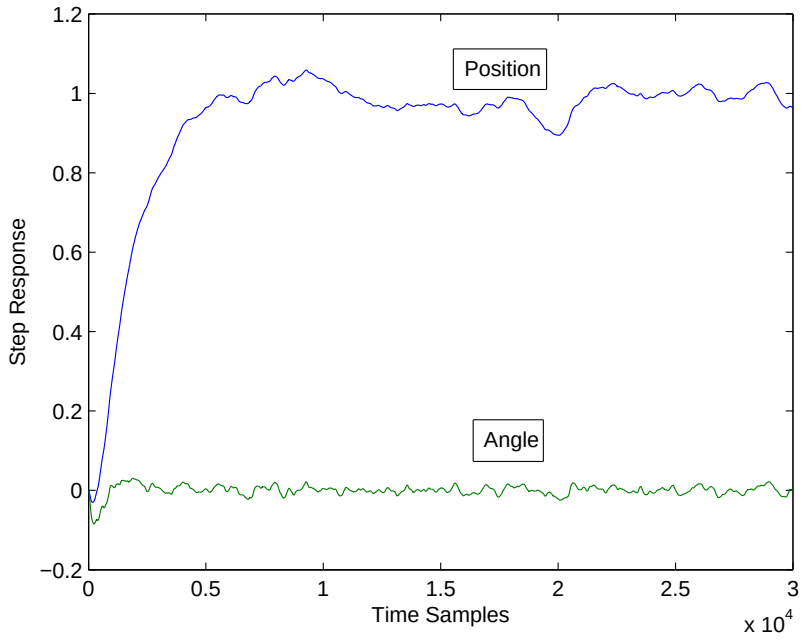

Figure 3.10: Step response of the closed-loop system when stability conditions are satisfied.

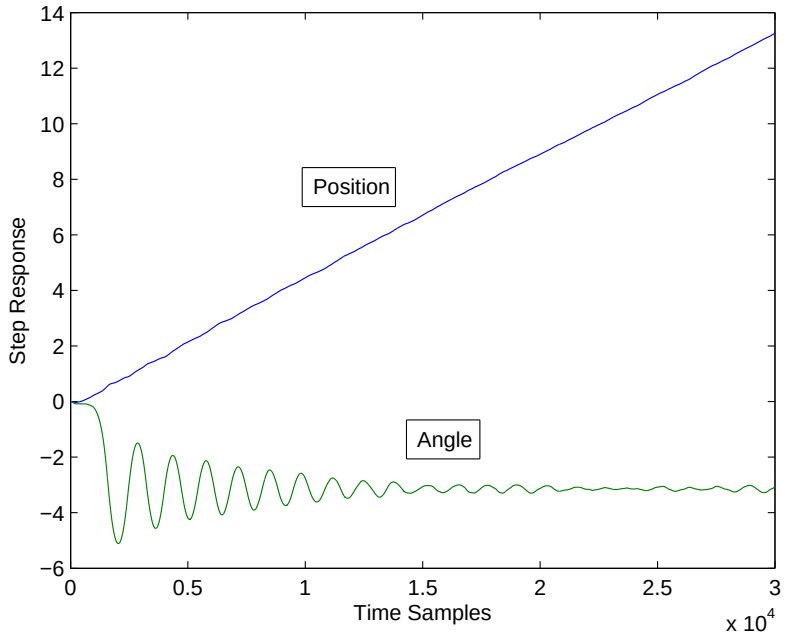

Figure 3.11: Step response with more activity of primary users.

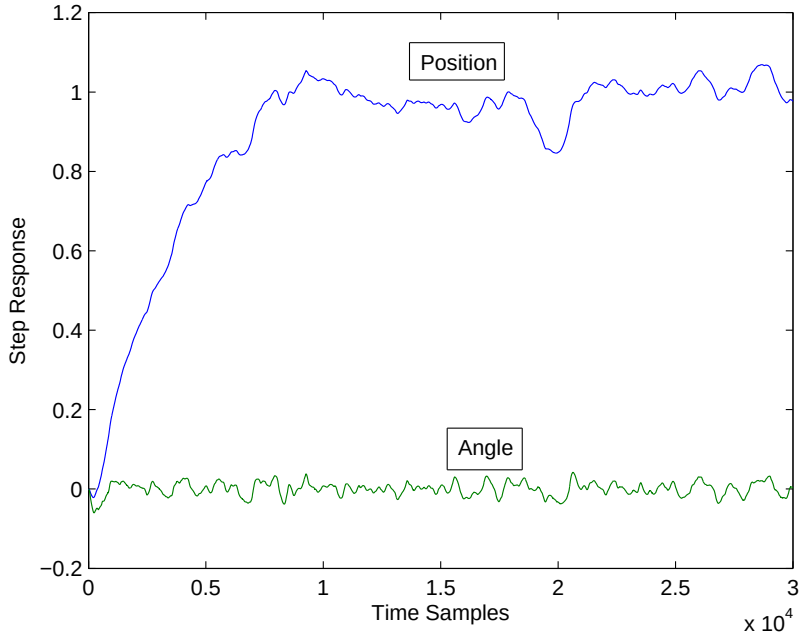

Figure 3.12: Step response for a better controller gain.

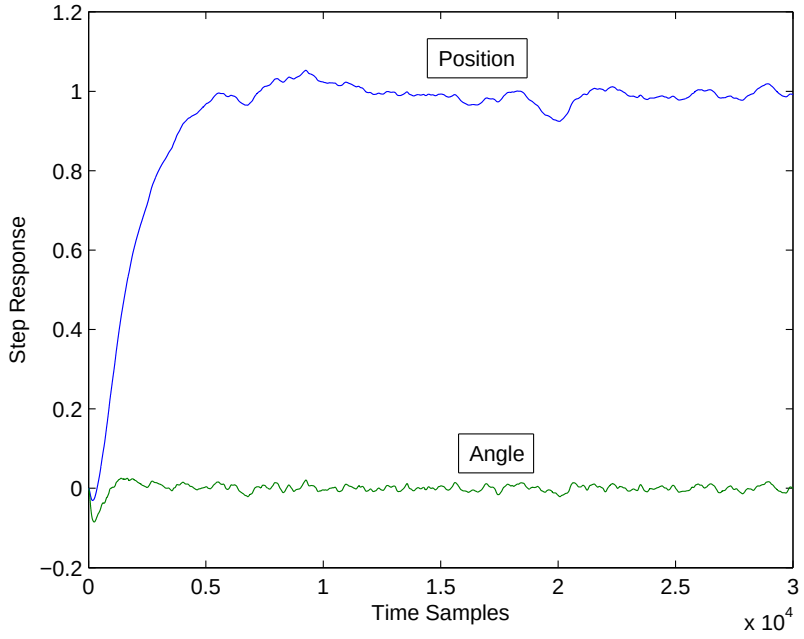

Figure 3.13: Step response when $p = 1$.

the condition is satisfied. The result is depicted in fig 3.13. Note that in this case the optimal controller exists and is linear in the state estimate.

Chapter 4

Estimation over Cognitive Radio Links Modeled by Semi-Markov Processes

The previous chapter addressed estimation and control over CR links represented by a two-switch model. However, it can be simplified into one sensing variable due to the following two reasons: First, the effect of the SR on PUs is negligible unless their locations are very close; second, it is costly to use sensors to sense CR channels. Thus, in communications, people are more willing to omit the influence of the SR in the CR system and the two-switch model becomes a single switching state model. What's more, as has been shown in [39], a semi-Markov process captures the stochastic behavior of each channel in the CR system more accurately. This study uses a semi-Markov model to capture the behavior of CR links. The problem model in this chapter thus coincides with the model of the state estimation through the packet loss link, which has been previously examined [3]∼[9]. However, other studies have modeled the packet loss indicator by either a Bernoulli or Markov process, while here the lossy indicator in CR system is modeled by semi-Markov processes, which has not been addressed before, and results in this chapter are more general as both Bernoulli and

Markov processes are special cases of semi-Markov processes. Parts of this work have been published in [58, 59, 127].

4.1 System Model

4.1.1 Cognitive Radio Model

Consider an example of the CR system as seen below in fig 4.1. Reference [39]

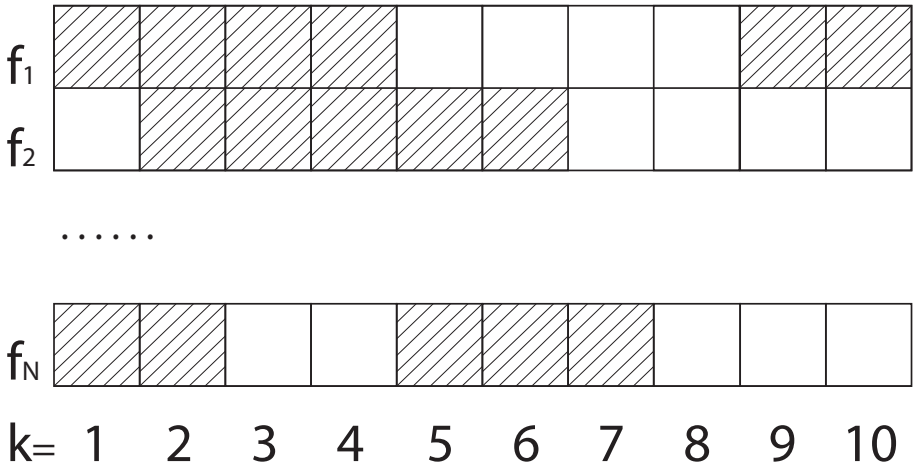

Figure 4.1: Channels status in cognitive radio.

shows that each channel is governed by a semi-Markov process: In each channel, there are two states (busy and idle). The times that the channel stays in one state are i.i.d random variables following some probability distribution functions. The CR structure considered in [37] [38] employs i.i.d Bernoulli variables to represent the switch between idle and busy states. In fact, the Bernoulli distribution is a special case of the Markov process and thus a special case of the semi-Markov process. In this work, a homogeneous semi-Markov process is used to model each channel.

Assume the sensor in CR transmitter senses only one channel at each time step (this avoids costly and complicated sensors which can sense multiple channels). The sensor first chooses one channel to sense according to some sensing policy; if

the channel is idle, the signal is transmitted through the channel; otherwise, stops transmission to avoid the collision.

Denote the signal sent at time k as y_k, then the received signal $\tilde{y}_k$ can be written as:

$$\tilde{y}_k = \gamma_k y_k + \omega_k \tag{4.1}$$

where γ_k is governed by N semi-Markov processes each of which represents the behavior of one channel. $\gamma_k = 1$ if an unutilized channel is sensed and used to transmit the signal, and $\gamma_k = 0$ if a busy channel is sensed and no information is delivered. Let ω_k denote the Gaussian white noise with zero mean and variance R.

4.1.2 Problem Formulation

For the discrete-time linear system:

$$x_{k+1} = Ax_k + v_k \tag{4.2}$$

$$y_k = Cx_k$$

where $x_k \in \mathbb{R}^{d\times 1}$ is the state vector at time k, $A \in \mathbb{R}^{d\times d}, C \in \mathbb{R}^{m\times d}$ are system parameters and A is assumed to be unstable, i.e., at least one eigenvalue is in the right half plane, (A, C) is observable, v_k represents Gaussian white noise with zero mean and variance Q, $y_k \in \mathbb{R}^{m\times 1}$ represents the system output at time k. The measurements received through the CR system is defined as:

$$\tilde{y}_k = \gamma_k Cx_k + \omega_k \tag{4.3}$$

Let γ_k^l denote the status of the lth channel at time k and $\{\gamma_k^l\}_{k\geq 1}$ is the lth semi-Markov process where $\gamma_k^l = 1$ means that the lth channel is idle at time k otherwise it is busy.

Two different cases will be addressed in the state estimation problem. In the first case, γ_k is assumed to be known at each time step at the estimator (e.g., by using a common control channel to synchronize the frequency channel or adding a time stamp to inform failure of transmission). In the other case, γ_k is assumed as unknown (e.g., no time stamp or no reliable common control channel).

4.2 Preliminary on Semi-Markov Process

In this section, we introduce some preliminaries of the discrete-time semi-Markov process that will be useful in the next section.

A semi-Markov chain is characterized by an imbedded Markov chain and a set of sojourn time probability distribution functions. When the process enters state i, the next state j, is chosen based on the imbedded Markovian transition probability, and the time after which the jump takes place is obtained from the sojourn time (waiting time) distribution function.

The associated homogeneous semi-Markov kernel Q is defined by [53]:

$$Q_{ij}(\tau) = \mathbb{P}\{\gamma_{n+1} = j, k_{n+1} - k_n \leq \tau \mid \gamma_n = i\}, \tag{4.4}$$

where k_{n+1} denotes the time instant for the $n+1$th jump (a jump denotes a state switch) of the semi-Markov process; γ_n denote the state after the nth jump, and $i, j = \{0, 1\}, i \neq j$. The transition probability of the imbedded Markov chain satisfies [54],

$$p_{ij} := \lim_{\tau \to \infty} Q_{ij}(\tau) = \mathbb{P}\{\gamma_{n+1} = j \mid \gamma_n = i\}, \tag{4.5}$$

where $T = [p_{ij}]$ is the transition probability matrix of the imbedded Markov chain. In this work, for each semi-Markov process, there are only two states corresponding to the busy and idle status of each channel, respectively. Thus, it is obvious that $p_{01} = p_{10} = 1$ in the practical point of view (one state will always switch to the other one).

Now define the following conditional distribution function:

$$S_{ij}(\tau) := \mathbb{P}\{k_{n+1} - k_n = \tau \mid \gamma_{n+1} = j, \gamma_n = i\},\ \tau = 1, 2, 3, ... \tag{4.6}$$

It is easy to see that $\sum_{\tau=1}^{\infty} S_{ij}(\tau) = 1$ for both $i, j = \{0, 1\}, i \neq j$ [55]. Moreover, since there are only two states, we have

$$S_{ij}(\tau) = \mathbb{P}\{k_{n+1} - k_n = \tau \mid \gamma_n = i\}$$

The conditional transition probability from 0 to 1 at time k is computed as [55, 56]:

$$p_{01}(k) \quad = \quad \mathbb{P}\{\gamma_{k+1} = 1 \mid \gamma_k = 0\} = \mathbb{P}\{k_{n+1} - k_n \leq k \mid \gamma_n = 0\} = \sum_{\tau=1}^{k} S_{01}(\tau) \tag{4.7}$$

Similarly, we have

$$\begin{aligned} p_{10}(k) &= \sum_{\tau=1}^{k} S_{10}(\tau), \\ p_{00}(k) &= 1 - p_{01}(k), \\ p_{11}(k) &= 1 - p_{10}(k) \end{aligned} \tag{4.8}$$

Note that time k is different from the sojourn time τ. These conditional probabilities are useful in state estimation when the second case when γ_k is unknown at the estimator. Denote by $S_{ij}^{l}(\tau)$ the probability distribution of the sojourn time (S_{ij} above) of the lth channel (i.e., the lth semi-Markov process). In a practical situation, statistical properties of each channel can be obtained over a period of time.

4.3 State Estimation over Cognitive Radio System

In this section, state estimation algorithms over CR are derived for both cases described in Section 4.1.

4.3.1 State Estimation when γ_k is known at the receiver

The Optimal Filter

The optimal state estimator for system (4.2) and (4.3) when γ_k is known at the receiver becomes a standard state estimation of a linear time varying system subject to Gaussian white noise. The optimal estimator is the standard Kalman Filter given as follows [5]:

A Priori state estimate and error covariance:

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} \tag{4.9}$$

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q \tag{4.10}$$

A Posteriori state estimate and error covariance:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + \gamma_k K_k(\tilde{y}_k - C\hat{x}_{k|k-1}) \tag{4.11}$$

$$K_k = P_{k|k-1}C^T(CP_{k|k-1}C^T + R)^{-1} \tag{4.12}$$

$$P_{k|k} = P_{k|k-1} - \gamma_k K_k P_{k|k-1} \tag{4.13}$$

where $\hat{x}_{k|k-1}$ is the a prior state estimate at time k; $\hat{x}_{k|k}$ is the a posterior state estimate at time k; $P_{k|k-1}$ is the error covariance of $x_k - \hat{x}_{k|k-1}$;$P_{k|k}$ is the error covariance of $x_k - \hat{x}_{k|k}$; K_k is the Kalman gain.

To characterize the prediction error covariance, one can easily derive the following Riccati equation:

$$P_{k+1} = AP_kA^T + Q - \gamma_k AP_kC^T(CP_{k|k-1}C^T + R)^{-1}CP_kA^T \tag{4.14}$$

where $P_{k+1} := P_{k+1|k}$. Assume without loss of generality that the initial condition of (4.14) is $P_1 = P_{1|0}$ and $\gamma_1 = 1$.

The process γ_k will experience a consecutive sequence of 1's followed by a consecutive sequence of 0's. Thus, starting from a nonnegative definite real matrix

P_1, when $\gamma_k = 1$, $P_{k+1} = AP_kA^T + Q - AP_kC^T(CP_{k|k-1}C^T + R)^{-1}CP_kA^T$ converges according to the Kalman Filtering theory; when $\gamma_k = 0$, $P_{k+1} = AP_kA^T + Q$ diverges as A is unstable. So the covariance will go through a "stable process" (when $\gamma_k = 1$) and then a "unstable process" (when $\gamma_k = 0$). To better illustrate the stability of the covariance, we employ the concept of the peak covariance process introduced in [7]. In the previous chapter, we have introduced this notation for the stability study of the Bernoulli variable. Here, we present it again for this semi-Markov model based problem.

Let β_c denote the time of the cth jump of γ_k from 0 to 1 (see section 4.3.1 for more details). Labeling a subsequence of the covariance process P_c by the sequence of times β_c, denote

$$M_c = P_{\beta_c}$$

M_c denotes the value of the covariance $P_{\beta_c} = P_{\beta_c|\beta_c-1}$ computed by $P_{k+1} = AP_kA^T+Q$ at $k = \beta_c-1$ and $\{M_c\}_{c\geq 1}$ is called the peak covariance process. The peak covariance process thus consists of a sequence of covariances which are computed at $k = \beta_c - 1$ before γ_k jumping into the state $\gamma_{\beta_c} = 1$.

Definition 3. *[7] We say the peak covariance sequence $\{M_c\}$ is stable if $\sup_{c\geq 1}\mathbb{E} \parallel M_c \parallel < \infty$. Accordingly, we say the system satisfies peak covariance stability.*

Consider a series of systems:

$$\begin{aligned} x_{k+1} &= Ax_k + v_k \\ \tilde{y}_k &= \gamma_k^l Cx_k + \omega_k \end{aligned} \tag{4.15}$$

where $l = 1, ..., N$. Note γ_k in (4.3) is replaced by γ_k^l (defined in section II.B) in (4.15) and the original problem (4.2), (4.3) has been divided into N independent problems, each of which is a packet loss problem governed by a semi-Markov process. Optimal filters for these systems can be derived similarly through (4.10)$\sim$(4.13). Let $\{P_k^l\}_{k\geq 1}$

denote the covariance process of each optimal filter and $\{M_c^l\}_{c\geq 1}$ denote the peak covariance process of the lth system. The following assumption is made:

Assumption 1. *Assume there is at least one channel, h of the N channels satisfying:*

$$\sup_{c\geq 1} \mathbb{E} \parallel M_c \parallel \leq \sup_{c\geq 1} \mathbb{E} \parallel M_c^h \parallel$$

This assumption is reasonable since the sensor is employed in the CR system to help SUs to search for available channels to transmit. If no channel satisfies Assumption 1, then the peak covariance M_c is "worse" than the peak covariance M_c^l for each channel, which makes the sensor useless.

The following lemma is useful for the derivation of stability conditions of the optimal filter.

Lemma 4.0.2. *Under assumption 1, the peak covariance process $\{M_c\}_{c\geq 1}$ of the optimal filter of the original system (4.2, 4.3) is **stable** if $\{M_c^l\}_{c\geq 1}$ is **stable** for each l.*

Proof. From the statement of the lemma, $\{M_c^l\}_{c\geq 1}$ is stable for each l of N, thus we have

$$\sup_{c\geq 1} \mathbb{E} \parallel M_c^l \parallel < \infty$$

which further leads to $\sup_{c\geq 1} \mathbb{E} \parallel M_c \parallel < \infty$. □

The argument for each l is necessary as in practice, the information about which channel satisfies the assumption 1 is unknown.

Stability Analysis

Based on Lemma 4.0.2 and due to the independence of each system in (4.15), the stability problem for the optimal filter over the CR system reduces to the stability problem for each system in (4.15). In this section, we analyze the stability of (4.15). By *suppressing the superscript l* in (4.15), the packet indicator γ_k now represents one

semi-Markov process instead of the N semi-Markov processes in the original problem, and $\{M_c\}_{c\geq 1}$ (after *suppressing the superscript* l) represents the peak covariance of the lth optimal filter.

For the given initial condition $\gamma_1 = 1$, the following two stopping times are introduced [7]:

$$\tau_1 = \inf\{k : k > 1, \gamma_k = 0\}.$$
$$\beta_1 = \inf\{k : k > \tau_1, \gamma_k = 1\}.$$

For convenience, define $\beta_0 = \gamma_1 = 1$. Thus, τ_1 is the first time when primary users occur. The above procedure generates two sequences $\{\tau_i, i \geq 1\}$ and $\{\beta_i, i \geq 1\}$, where for $i > 1$: $\tau_i = \inf\{k : k > \beta_{i-1}, \gamma_k = 0\}$ and $\beta_i = \inf\{k : k > \tau_i, \gamma_k = 1\}$. Both sequences have finite values for each of their entries [7].

Define: $\tau_i^* = \tau_i - \beta_{i-1}$ and $\beta_i^* = \beta_i - \tau_i$, where $\beta_0 = 1$. Here τ_i^* and β_i^* denote sojourn times at state 1 and state 0, respectively.

Lemma 4.0.3. *The following hold*

(i) The random variables $\{\tau_i^*, i \geq 1\}$ *are i.i.d., and* $\mathbb{P}(\tau_i^* = \xi) = S_{10}(\xi), \xi \geq 1$.

(ii) The random variables $\{\beta_i^*, i \geq 1\}$ *are i.i.d., and* $\mathbb{P}(\beta_i^* = \xi) = S_{01}(\xi), \xi \geq 1$.

(iii) The random variables $\{\tau_i^*, \beta_i^*, i \geq 1\}$ *are independent of each other.*

Proof. We only give the proof of (i), the proof of (ii) and (iii) can be obtained similarly. By the homogeneity of the semi-Markov process, the sojourn time $\{\tau_i^*, i \geq 1\}$ are i.i.d.

By definition:

$$\begin{aligned}
\mathbb{P}(\tau_i^* = \xi) &= \mathbb{P}(\gamma_{\beta_{i-1}+1} = 1, ..., \gamma_{\beta_{i-1}+\xi-1} = 1, \gamma_{\tau_i} = 0 | \gamma_{\beta_{i-1}} = 1) \\
&= \mathbb{P}(k_{n+1} - k_n = \xi, \gamma_{n+1} = 0 | \gamma_n = 1) \\
&= \mathbb{P}(k_{n+1} - k_n = \xi | \gamma_n = 1) \\
&= S_{10}(\xi) \qquad (4.16)
\end{aligned}$$

□

Definition 4 and Lemma 4.0.4 stated below from [7] are useful in deriving the main theorem.

Let S^d denote the set of all $d \times d$ nonnegative definite real matrices. Define the map $F(\cdot)$: $S^d \to S^d$ by

$$F(P) = APA^T + Q - APC^T(CPC^T + R)^{-1}CPA^T$$

where $P \in S^d$. It is obvious that for any $P \in S^d$, $F(P) \geq F(0) = Q$, and, therefore, $F(P) \in S^d$.

Definition 4. *For the observable linear system $[A, C]$, the observability index is the smallest integer I_0 such that $[C^T, A^TC^T, ..., (A^{I_0-1})^TC^T]$ has rank d.*

Define $S_0^d := \{P : 0 \leq P \leq A\tilde{P}A^T + Q, \text{ for some } \tilde{P} \geq 0\}$. Note that S_0^d is a convex subset of S^d.

Lemma 4.0.4. *For the map $F(P)$ defined above, there exists a constant $K > 0$ such that:*
(i) For any $\bar{P} \in S_0^d$, $F^{\xi}(\bar{P}) \leq KI$ for all $\xi \geq I_0$;
(ii) For any $\bar{P} \in S^d$, $F^{\xi+1}(\bar{P}) \leq KI$ for all $\xi \geq I_0$;
(iii) For $1 \leq i \leq (I_0 - 1) \vee 1$, where $(I_0 - 1) \vee 1 = max\{(I_0 - 1), 1\}$, there exist positive constants $d_i^{(0)}$ and $d_i^{(1)}$ satisfy the following inequality:

$$\| F^i(P) \| \leq d_i^{(1)} \| P \| + d_i^{(0)}, \quad \forall P \in S_0^d \tag{4.17}$$

where I is the $d \times d$ identity matrix; $\| \cdot \|$ denotes the matrix induced norm for matrices. For $I_0 = 1$, $d_1^{(1)} = 0$ and $d_i^{(0)} > 0$.

The following theorem gives sufficient conditions of the peak covariance.

Theorem 4.1. *The peak covariance process* $\{M_c\}_{c\geq 1}$ *is stable if the following three conditions hold:*

$$(i) \qquad \limsup_{\xi\to\infty}(1-\frac{S_{01}(\xi+1)}{1-\sum_{j=1}^{j=\xi}S_{01}(j)})<\frac{1}{|\lambda_A|^2}$$

$$(ii) \qquad \limsup_{\xi\to\infty}(\frac{S_{01}(\xi+1)}{S_{01}(\xi)})<\frac{1}{|\lambda_A|^2}$$

$$(iii) \qquad d_1^{(1)}[S_{10}(1)+\sum_{i=1}^{I_0-1}d_i^{(1)}S_{10}(i+1)]\sum_{j=1}^{\infty}\parallel A^j\parallel^2 S_{01}(j)<1$$

where λ_A *is an eigenvalue of the largest magnitude for matrix* A*. Moreover, if* C *is invertible, then condition (iii) above vanishes and the peak covariance stability holds under condition (i) and (ii).*

Proof. The expectation of $\parallel P_{\beta_{c+1}+1}\parallel$ conditioned on $P_{\beta_c+1}=P\geq 0$ is computed first:

$$\begin{aligned}
\mathbb{E}[\parallel P_{\beta_{c+1}+1}\parallel\mid P_{\beta_c+1}=P] &= \sum_{j=1}^{\infty}\sum_{i=1}^{\infty}\mathbb{E}[\parallel P_{\beta_{c+1}+1}\parallel\times 1_{\tau_{c+1}-\beta_c=i,\beta_{c+1}-\tau_{c+1}=j}|P_{\beta_c+1}=P]\\
&\leq \sum_{j=1}^{\infty}\sum_{i=1}^{\infty}d_1^{(1)}\parallel A^jF^{i-1}(P)(A^T)^j+A^{j-1}Q(A^T)^{j-1}\\
&\quad +\cdots+AQA^T+Q\parallel\times S_{10}(i)S_{01}(j)+d_1^{(0)}\\
&\leq \sum_{j=1}^{\infty}\sum_{i=1}^{\infty}d_1^{(1)}\parallel A^{j-1}Q(A^T)^{j-1}+\cdots+AQA^T\\
&\quad +Q\parallel\times S_{10}(i)S_{01}(j)+\sum_{j=1}^{\infty}\sum_{i=I_0+1}^{\infty}d_1^{(1)}\\
&\quad \parallel A^jF^{i-1}(P)(A^T)^j\parallel\times S_{10}(i)S_{01}(j)\\
&\quad +\sum_{j=1}^{\infty}\sum_{i=1}^{I_0}d_1^{(1)}\parallel A^jF^{i-1}(P)(A^T)^j\parallel\times S_{10}(i)S_{01}(j)+d_1^{(0)}\\
&= \Gamma_1+\Gamma_2+\Gamma_3+d_1^{(0)} \qquad (4.18)
\end{aligned}$$

where $1_{(\cdot)}$ denotes the characteristic function.

Then, we have:

$$\begin{aligned}\Gamma_1 &= \sum_{j=1}^{\infty} d_1^{(1)} \sum_{i=1}^{\infty} S_{10}(i) \parallel \sum_{\xi=0}^{j-1} A^{\xi} Q (A^T)^{\xi} \parallel S_{01}(j) \leq \sum_{j=1}^{\infty} d_1^{(1)} \sum_{\xi=0}^{j-1} \parallel A^{\xi} \parallel^2 \parallel Q \parallel S_{01}(j) \\ &= d_1^{(1)} \parallel Q \parallel \sum_{\xi=0}^{\infty} \parallel A^{\xi} \parallel^2 \sum_{j=\xi+1}^{\infty} S_{01}(j) < \infty \end{aligned} \tag{4.19}$$

where by positive series property, the series converges if:

$$\limsup_{\xi \to \infty} \frac{\parallel A^{\xi+1} \parallel^2 \sum_{j=\xi+2}^{\infty} S_{01}(j)}{\parallel A^{\xi} \parallel^2 \sum_{j=\xi+1}^{\infty} S_{01}(j)} < 1 \tag{4.20}$$

Thus we have condition (i) from (4.20) by the fact that $\sum_{j=1}^{\infty} S_{01}(j) = 1$.

Similarly,

$$\Gamma_2 \leq K d_1^{(1)} \sum_{i=I_0+1}^{\infty} S_{10}(i) \sum_{j=1}^{\infty} \parallel A^j \parallel^2 S_{01}(j) < \infty \tag{4.21}$$

where the positive series converges if:

$$\limsup_{j \to \infty} \frac{\parallel A^{j+1} \parallel^2 S_{01}(j+1)}{\parallel A^j \parallel^2 S_{01}(j)} \leq |\lambda_A^2| \limsup_{j \to \infty} \frac{S_{01}(j+1)}{S_{01}(j)} < 1 \tag{4.22}$$

Thus, condition (ii) is obtained from (4.22). Finally, we have:

$$\begin{aligned}\Gamma_3 &\leq \sum_{j=1}^{\infty} d_1^{(1)} \parallel A^j \parallel^2 S_{01}(j) [S_{10}(1) \parallel P \parallel + \sum_{i=1}^{I_0-1} (d_i^{(1)} \parallel P \parallel + d_i^{(0)}) S_{10}(i+1)] \\ &= \{[S_{10}(1) + \sum_{i=1}^{I_0-1} (d_i^{(1)} S_{10}(i+1))] \parallel P \parallel + \sum_{i=1}^{I_0-1} d_i^{(0)} \times S_{10}(i+1)\} \\ &\quad \times d_1^{(1)} \sum_{j=1}^{\infty} \parallel A^j \parallel^2 S_{01}(j) \\ &= C_0 \parallel P \parallel + C_1 \end{aligned} \tag{4.23}$$

where C_1 is a positive finite constant.

Then, by (4.19), (4.22) and (4.23), (4.18) can be written as:

$$\mathbb{E}[\parallel P_{\beta_{c+1}+1} \parallel\mid P_{\beta_c+1} = P] \leq C_0 \parallel P \parallel + C_2 \tag{4.24}$$

To guarantee stability, let

$$C_0 \;=\; [S_{10}(1) + \sum_{i=1}^{I_0-1}(d_i^{(1)} S_{10}(i+1))] \times d_1^{(1)} \sum_{j=1}^{\infty} \parallel A^j \parallel^2 S_{01}(j) < 1$$

(4.24) implies:

$$\mathbb{E}[\parallel P_{\beta_{c+1}+1} \parallel\mid P_{\beta_c+1}] \leq C_0 \parallel P_{\beta_c+1} \parallel + C_2 \tag{4.25}$$

which leads to

$$\mathbb{E}[\parallel P_{\beta_{c+1}+1} \parallel] \leq C_0 E[\parallel P_{\beta_c+1} \parallel] + C_2 \tag{4.26}$$

which means $\limsup_c \mathbb{E}[\parallel P_{\beta_{c+1}+1} \parallel] < \infty$.

Similarly, we estimate $\mathbb{E}[\parallel P_{\beta_{c+1}} \parallel]$ starting with P_{β_c+1}:

$$\begin{aligned}
\mathbb{E}[\parallel P_{\beta_{c+1}} \parallel | P_{\beta_c+1}, \beta_c] \;&=\; \sum_{j=1}^{\infty}\sum_{i=1}^{\infty} \parallel A^j F^{i-1}(P_{\beta_c+1})(A^T)^j + A^{j-1}Q(A^T)^{j-1} \\
&\quad + \cdots + AQA^T + Q \parallel \times S_{10}(i) S_{01}(j) \\
&\leq \sum_{j=1}^{\infty}\sum_{i=1}^{\infty} (\parallel A^j F^{i-1}(P_{\beta_c+1})(A^T)^j \parallel + \parallel A^{j-1}Q(A^T)^{j-1} \\
&\quad + \cdots + AQA^T + Q \parallel) \times S_{10}(i) S_{01}(j) \\
&= \sum_{j=1}^{\infty}\sum_{i=1}^{\infty} \parallel A^j F^{i-1}(P_{\beta_c+1})(A^T)^j \parallel S_{10}(i) S_{01}(j) + O(1) \\
&= \sum_{i=1}^{\infty} \parallel F^{i-1}(P_{\beta_c+1}) \parallel S_{10}(i) + O(1) \\
&= \sum_{i=1}^{I_0} \parallel F^{i-1}(P_{\beta_c+1}) \parallel S_{10}(i) + O(1) \leq K_1 \parallel P_{\beta_c+1} \parallel + K_2
\end{aligned}$$

where K_1, K_2 are positive constants. Above, the second equality is from condition (i), the third comes from condition (ii), the fourth is from Lemma 4.0.4 and the last inequality is from (4.17). Therefore, it easily follows that $\sup_{c\geq 1}\mathbb{E}[\| P_{\beta_{c+1}} \|] < \infty$ and the stability of the peak covariance process is obtained.

Note the left hand side of condition (iii) indicate $\sum_{j=1}^{\infty} \| A^j \|^2 S_{01}(j) < \infty$, which also leads to condition (ii).

When C is invertible, then $I_0 = 1$, which means $d_1^{(1)} = 0$. Condition (iii) vanishes. □

Next, we provide another theorem for peak covariance stability of each channel, which can substitute for Theorem 4.1.

Theorem 4.2. *The peak covariance process $\{M_c\}_{c\geq 1}$ is stable if condition (i), (ii) in Theorem 4.1 and the following inequality*

$$\sum_{j=1}^{\infty} \| A^j \|^2 S_{01}(j) \sum_{i=1}^{I_0-1} \| A^i \|^2 S_{10}(i) < 1 \tag{4.27}$$

hold.

Proof. The proof follows the argument in the proof of Theorem 4.1 and Theorem 3.1 in [8]. We only state some key points in the following. First, from the above description, we have

$$P_{\beta_{c+1}} = A^{\beta^*_{c+1}} P_{\tau_{c+1}} (A^{\beta^*_{c+1}})^T + \sum_{c=0}^{\beta^*_{c+1}-1} A^c Q (A^i)^T$$

Then, after some calculations, we have

$$\mathbb{E}[\| P_{\beta_{c+1}} \|] \leq \sum_{j=1}^{\infty} \| A^j \|^2 S_{01}(j) \mathbb{E}[\| P_{\tau_{c+1}} \|] + \sum_{j=1}^{\infty} \sum_{c=0}^{j-1} \| A^c \|^2 \| Q \| S_{01}(j) \tag{4.28}$$

where the first and the second term on the right hand side of the above inequality leads to condition (ii) and (i), respectively, by the convergence of positive series, as in the proof of theorem 4.1.

Following the statements given by Theorem 3.1 in [8], we obtain

$$\mathbb{E}[\| P_{\tau_{c+1}} \|] \leq \sum_{i=1}^{I_0-1} \| A^i \|^2 S_{10}(i)\mathbb{E}[\| P_{\beta_c} \|] + L_1 \tag{4.29}$$

where L_1 is a positive constant. Substituting (4.29) into (4.28), we obtain

$$\mathbb{E}[\| P_{\beta_{c+1}} \|] \leq \sum_{j=1}^{\infty} \| A^j \|^2 S_{01}(j) \sum_{i=1}^{I_0-1} \| A^i \|^2 S_{10}(i)\mathbb{E}[\| P_{\beta_c} \|] + L \tag{4.30}$$

where L is a positive constant. Thus, we reach condition (ii) from the above inequality with the initial condition $\mathbb{E}[\| P_{\beta_0} \|] = \mathbb{E}[\| P_1 \|] = \| P_1 \|$. □

A necessary condition of M_c is given below.

Theorem 4.3. *M_c is stable only if the positive series $\sum_{i=1}^{N} \rho(A)^{2i} \sum_{j=i}^{\infty} S_{01}(j)$ converges.*

Proof. Since the spectral norm equals to the matrix norm induced by the Euclidian norm, we have

$$\| P_{\beta_{c+1}} \| = \sup_{\|x\|=1} x P_{\beta_{c+1}} x^T$$

from which we have

$$\| P_{\beta_{c+1}} \| \geq x P_{\beta_{c+1}} x^T$$

Furthermore, since $P_{\alpha_{c+1}} \geq Q$

$$P_{\beta_{c+1}} = A^{\beta^*_{c+1}} P_{\alpha_{c+1}} (A^{\beta^*_{c+1}})^T + \sum_{i=0}^{\beta^*_{c+1}-1} A^i Q (A^i)^T \geq \sum_{i=0}^{\beta^*_{c+1}} A^i Q (A^i)^T$$

Then,

$$xP_{\beta_{c+1}}x^T \geq \sum_{i=0}^{\beta^*_{c+1}} xA^iQ(A^i)^Tx^T = \sum_{i=0}^{\beta^*_{c+1}} \rho(A)^{2i}xQx^T$$

which implies

$$\begin{aligned}
\mathbb{E}[\parallel P_{\beta_{c+1}} \parallel] &\geq \mathbb{E}[\sum_{i=0}^{\beta^*_{c+1}} \rho(A)^{2i}]xQx^T \parallel x \parallel^{-2} \\
&= xQx^T \parallel x \parallel^{-2} \sum_{j=1}^{\infty}\sum_{i=0}^{j} \rho(A)^{2i}S_{01}(j) \\
&= xQx^T \parallel x \parallel^{-2} (\sum_{i=1}^{\infty} \rho(A)^{2i} \sum_{j=i}^{\infty} S_{01}(j) + \sum_{j=1}^{\infty} S_{01}(j)) \\
&= xQx^T \parallel x \parallel^{-2} (\sum_{i=1}^{\infty} \rho(A)^{2i} \sum_{j=i}^{\infty} S_{01}(j) + 1)
\end{aligned}$$

Thus, $\mathbb{E}[\parallel P_{\beta_{c+1}} \parallel] \leq \infty$ indicates the the convergence of the series in the statement. □

The next theorem gives a sufficient condition of the peak covariance of the original system which is a direct result of lemma 4.0.2.

Theorem 4.4. *The peak covariance process of the original system (4.2), (4.3) is stable if each of the channels sensed in the CR system can be represented by a semi-Markov process that satisfies Theorem 4.1.*

Remark 3. *When C is invertible, condition (iii) vanishes in theorem 4.1, thus an appropriate choice of $S_{01}(\xi)$ will stabilize the covariance process, and provide a way to design CR channels to guarantee stability.*

Remark 4. *If γ_k is a Markov process, conditions (i) and (ii) coincide and Theorem 4.1 becomes theorem 6 in [7].*

Remark 5. *In this remark,* the notation M_c represents the peak covariance of the original problem (4.2) and (4.3). *Theorem 4.4 states that if each channel in the CR*

system has the semi-Markov statistic satisfying Theorem 4.1, then the peak covariance process of the overall system (4.2) and (4.3) is stable. However, it introduces some conservatism as each channel has to guarantee a stable peak covariance, which may not be necessary as a stable state estimator may occur with only a few channels with stable peak covariances. Theorem 4.4 is actually a direct result of Assumption 1 (which leads to Lemma 4.0.2 as discussed above), which depends on the sensing strategy of the CR system. In the case that some channels have unstable peak covariances, assumption 1 is useless. We claim that without this assumption, the original problem can not be divided into N independent problems, each of which addresses only one semi-Markov process, instead of the combination of N semi-Markov processes in the original problem. To solve the original problem directly without that assumption is difficult as it involves γ_k which is a heterogenous Markov process. It will be our future work.

However, if we know which channels have stable peak covariances (from their statistics), it is possible to design a sensing strategy such that the following inequality is satisfied:

$$\sup_{c\geq 1} \mathbb{E} \parallel M_c \parallel \leq \sup_{c\geq 1} \mathbb{E} \parallel M_c^s \parallel < \infty$$

for some channel s with a stable peak covariance. Then, M_c is stable.

Remark 6. *In general, it is hard to determine between Theorem 4.1 and Theorem 4.2 which is more conservative. However, for some cases, Theorem 4.2 is a weaker sufficient condition than Theorem 4.1. For example, let $I_0 = 2$ and choose $d_1^{(1)} = \parallel AA^T \parallel$. Then, Theorem 4.2 is less conservative then Theorem 4.1 as the left hand side of (4.27) becomes $\sum_{j=1}^{\infty} \parallel A^j \parallel^2 S_{01}(j) \parallel A \parallel^2 S_{10}(1)$, while condition (iii) in Theorem 4.1 becomes $\sum_{j=1}^{\infty} \parallel A^j \parallel^2 S_{01}(j) \parallel A \parallel^2 [S_{10}(1) + d_1^{(1)} S_{10}(2)]$. It is obvious that the latter is larger than the former which means Theorem 4.2 is less conservative.*

4.3.2 State Estimation when γ_k is unknown

When γ_k is unknown, it acts as a stochastic parameter. Here, we remove the assumption that A is unstable, and assume that the estimator (the receiver) knows the statistics of each channel (corresponding to each semi-Markov process), that is, $\{S_{ij}^l(\tau), l = 1, ..., N, i, j = \{0, 1\}, i \neq j\}$ are known at the estimator.

The estimation algorithm below involves conditional probabilities of γ_k, therefore, we need to specify the sensing policy explicitly. The sensing policies varying according to the communication systems. The policy used here is: The sensor in the CR system first chooses one channel to sense randomly; if it is idle, the sensor transmits the signal through it and at the next time step, still chooses that channel to sense; otherwise, stops transmission (no signal transmitted at this time) to avoid the collision, and chooses another channel randomly with equal probability at the following time step. Note for a different policy, the probability analysis discussed below should be adjusted correspondingly.

Probability Analysis

For each channel, there are two states (busy or idle, corresponding to $\gamma_k^l = 0$ or 1). Thus γ_k can be represented by a process with $2N$ states each of which can be described as 'choosing the lth channel to sense and $\gamma_k^l = 1$ or 0'. The transition matrix of this process is thus $2N$ by $2N$. Next, we compute conditional probabilities in the transition matrix that will be useful in the state estimation. Assume $Y_k = \{\tilde{y}_1, ..., \tilde{y}_k\}$ is the information set including observations from time 1 to k which are available at k, then, conditional probabilities we need to compute are summarized:

- $\mathbb{P}(\textit{choose } l, \gamma_{k+1}^l = 1 | \textit{choose } l, \gamma_k^l = 0, Y_k)$,
- $\mathbb{P}(\textit{choose } l, \gamma_{k+1}^l = 0 | \textit{choose } l, \gamma_k^l = 1, Y_k)$,
- $\mathbb{P}(\textit{choose } r, \gamma_{k+1}^r = 1 | \textit{choose } l, \gamma_k^l = 0, Y_k)$,
- $\mathbb{P}(\textit{choose } r, \gamma_{k+1}^r = 0 | \textit{choose } l, \gamma_k^l = 1, Y_k)$.

Other probabilities can be obtained from the above terms. For notation convenience, 'choosing r' is shortened by 'r' in the following statements.

These terms can be approximated following [56], however, the calculation requires the knowledge of statistics of semi-Markov processes given the set Y_k, e.g. $\mathbb{P}\{k_{n+1} - k_n = \tau \mid \gamma_{k+1} = \gamma_{n+1} = j, \gamma_k = \gamma_n = i, Y_k\}$, which can not be observed until measurements are received. Moreover, their computation is complicated and costly. In this work, we use the conditional probabilities of $\{\gamma_k\}_{k\geq 0}$ to approximate the above terms, e.g. approximate $\mathbb{P}(l, \gamma_{k+1}^l = 1 | l, \gamma_k^l = 0, Y_k)$ by $\mathbb{P}(l, \gamma_{k+1}^l = 1 | l, \gamma_k^l = 0)$. Also, assume for simplicity that the channels' states at $k+1$ are independent of the sensor's the choice at k, and note that the choice of the sensor at time k depends on the choice at $k-1$ and the chosen channel's status according to the sensing policy. Thus, we obtain

$$\begin{aligned}
\mathbb{P}(l, \gamma_{k+1}^l = 1 | l, \gamma_k^l = 0) &= 0, \\
\mathbb{P}(l, \gamma_{k+1}^l = 0 | l, \gamma_k^l = 1) &= \mathbb{P}(\gamma_{k+1}^l = 0 | l, \gamma_k^l = 1) = \mathbb{P}(\gamma_{k+1}^l = 0 | \gamma_k^l = 1) = p_{10}^l(k), \\
\mathbb{P}(r, \gamma_{k+1}^r = 1 | l, \gamma_k^l = 0) &= \frac{1}{N-1}\mathbb{P}(\gamma_{k+1}^r = 1) = \frac{1}{N-1}\sum_{j=0}^{1}\mathbb{P}(\gamma_{k+1}^r = 1, \gamma_k^r = j) \\
&= \frac{1}{N-1}\sum_{j=0}^{1} p_{j1}^r(k)\mathbb{P}(\gamma_k^r = j) \\
\mathbb{P}(r, \gamma_{k+1}^r = 0 | l, \gamma_k^l = 1) &= 0,
\end{aligned} \tag{4.31}$$

The first and the fourth equalities in (4.31) are obvious from the sensing policy, the second is the transition probability for the lth channel and the third is based on the sensing policy and the independence between the sensor and the channel. Without the assumption that channels' states are independent of the sensor's choice, there exists a complicated and recursive probability calculation which will be addressed in future work.

To better illustrate the arguments above, we take $N = 2$ as an example:

- Choose 1, $\gamma_k^1 = 0$: case 1;

- Choose 2, $\gamma_k^2 = 0$: case 2;
- Choose 1, $\gamma_k^1 = 1$: case 3;
- Choose 2, $\gamma_k^2 = 1$: case 4.

Then we can build the transition matrix as

$$\begin{pmatrix} 0 & \sum_{j=0}^{1} p_{j0}^{2}(t)\mathbb{P}(\gamma_k^2 = j) & 0 & \sum_{j=0}^{1} p_{j1}^{2}(k)\mathbb{P}(\gamma_k^2 = j) \\ \sum_{j=0}^{1} p_{j0}^{1}(k)\mathbb{P}(\gamma_k^1 = j) & 0 & \sum_{j=0}^{1} p_{j1}^{1}(k)\mathbb{P}(\gamma_k^1 = j) & 0 \\ p_{10}^{1}(k) & 0 & p_{11}^{1}(k) & 0 \\ 0 & p_{10}^{2}(k) & 0 & p_{11}^{2}(k) \end{pmatrix}$$

As a consequence, the process $\{\gamma_k\}_{k\geq 0}$ is modeled as a heterogeneous Markov process and any state estimation algorithm for MJLSs can be applied. In the sequel, we employ an efficient algorithm – the Interacting Multiple Model (IMM) Algorithm [57] for the state estimation.

IMM Algorithm

In this section, IMM algorithm is used to estimate the state of the system over the CR structure. The algorithm can be found in [57] and summarized as follows:

(1) Given initial values of $\{\mathbb{P}(\gamma_1^l = i), l = 1, ..., N; i = 0 \ or \ 1\}$, we start with $2N$ weights $\hat{p}_{2l-1}(k)$ and $\hat{p}_{2l}(k)$ where $\hat{p}_{2l-1}(k) = \mathbb{P}(l, \gamma_k^l = 0|Y_k)$ and $\hat{p}_{2l}(k) = \mathbb{P}(l, \gamma_k^l = 1|Y_k)$; $2N$ means $\hat{x}_{2l-1}(k)$ and $\hat{x}_{2l}(k)$; $2N$ associated covariances $\hat{V}_{2l-1}(k)$ and $\hat{V}_{2l}(k)$; then we compute the mixed initial condition for the filter matched to each state $q, q = 1, \cdots, 2N$, according to the following equations:

$$\begin{aligned} \bar{p}_q(k+1) &= \sum_{i=1}^{2N} w_{iq}(k)\hat{p}_i(k), \\ \hat{x}^q(k) &= \sum_{i=1}^{2N} w_{iq}(k)\hat{p}_i(k)\hat{x}_i(k)/\bar{p}_q(k+1), \\ \hat{V}^q(k) &= \sum_{i=1}^{2N} w_{iq}(k)\big(\hat{V}_i(k) + (\hat{x}_i(k) - \hat{x}^q(k))(\hat{x}_i(k) - \hat{x}^q(k))^T\big)/\bar{p}_q(k+1). \end{aligned}$$

where $w_{iq}(k)$ denotes the transition probability of the process γ_k calculated in (4.31).

(2) Then, after obtaining the above values of intermediate variables, use them as inputs to $2N$ Kalman Filters matched to each state q. From this we get $\bar{x}_q(k+1)$, $\bar{V}_q(k+1)$ after the time update and $\hat{x}_q(k+1)$, $\hat{V}_q(k+1)$ after the measurement update.

(3) The weights $\bar{p}_q(k+1)$ are updated from the innovation process of the Kalman Filter as:

$$\hat{p}_q(k+1) \;=\; c\bar{p}_q(k+1)\|W_q(k+1)\|^{-1/2} \times exp\{-1/2\zeta_q^T(k+1)W_q^{-1}(k+1)\zeta_q(k+1)\},$$

with c denotes a normalizing constant and

$$\begin{aligned} \zeta_q(k+1): &= y_{k+1} - \gamma_k^q C\bar{x}_q(k+1), \\ W_q(k+1): &= \gamma_k^q C\bar{V}_q(k+1)C^T\gamma_k^{qT} + R. \end{aligned}$$

(4) We can update the state estimate and the covariance according to

$$\hat{x}(k+1) = \sum_{q=1}^{2N} \hat{p}_q(k+1)\hat{x}_q(k+1),$$

$$\hat{V}(k+1) = \sum_{q=1}^{2N} \hat{p}_q(k+1)(\hat{V}_q(k+1) + (\hat{x}_q(k+1) - \hat{x}(k+1))(\hat{x}_q(k+1) - \hat{x}(k+1))^T).$$

In the next section, examples are provided to illustrate the algorithms developed.

4.4 Numerical Simulations

4.4.1 When γ_k is known

We give the following example to illustrate the performance of Theorem 4.1. For simplicity and without loss of generality, assume $N = 1$ since each channel is independent. The parameters of the system are given by:

$$A = \begin{bmatrix} 1.1 & 0.1 \\ 0 & 1.2 \end{bmatrix}, C = [1\ 1], V = I_{2\times 2}, W = 1$$

The channel is characterized by a semi-Markov process with the transition probability matrix $T = [p_{ij}]$ and the sojourn time conditional distribution function $S_{ij}(\tau)$:

$$T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, S_{01}(\tau) = s_0 \exp(-|\tau|), S_{10}(\tau) = s_1 \exp(-|\tau - 3|),$$

with s_i satisfying that $\sum_{\tau=1}^{\infty} S_{ij}(\tau) = 1$.

It is easy to see with the above information that the left hand side of condition (i) and (ii) are both $e^{-1} = 0.3679$ and $|\lambda_A| = 1.2$, thus condition (i) and (ii) are satisfied.

We also have $\| F(P) \| \leq \| AA' \| \| P \|$ and since AA' has two eigenvalues $\lambda_1 = 1.1672$ and $\lambda_2 = 1.4927$. Thus choose $d_1^{(1)} = 1.4928$. Numerical calculation yields $\sum_{j=1}^{\infty} \| A^j \|^2 S_{01}(j) \leq 2.1$, and $S_{10}(1) = 0.0649, S_{10}(2) = 0.17643$ gives $S_{10}(1) + d_1^{(1)} S_{10}(2) = 0.3283$. The left hand side of condition (iii) is computed as $d_1^{(1)}[S_{10}(1) + d_1^{(1)} S_{10}(2)] \sum_{j=1}^{\infty} \| A^j \|^2 S_{01}(j) < 0.9836 < 1$. Thus, conditions in Theorem 4.1 are all satisfied. Similarly, we can compute that $\sum_{j=1}^{\infty} \| A^j \|^2 S_{01}(j) \| A \|^2 S_{10}(1) < 0.2035$, which shows that conditions in Theorem 4.2 are satisfied and less conservative as $I_0 = 2$. $P_{11}(k)$ and $P_{12}(k)$ are two entries of the covariance matrix P_k, fig 4.2 and fig 4.3 show that they are bounded. Similarly, the other two entries $P_{21}(k)$ and $P_{22}(k)$ are also bounded.

4.4.2 When γ_k is unknown

For simplicity, in this example, we assume a scalar system with parameters: $A = 0.95$, $C = 1$, $V = 0.0001$, $W = 1$. Let $\mathbb{P}(\gamma_1^1 = 1) = 0.7$, $\mathbb{P}(\gamma_1^1 = 0) = 0.3$, $\mathbb{P}(\gamma_1^2 = 1) = 0.8$, and $\mathbb{P}(\gamma_1^2 = 0) = 0.2$.

We assume that there are two channels ($N = 2$) in the CR system. Each channel is characterized by a semi-Markov process with imbedded transition probability matrice $T^1 = [p_{ij}^1]$,$T^2 = [p_{ij}^2]$, sojourn time conditional distribution functions $S_{ij}^1(\tau)$ and $S_{ij}^2(\tau)$:

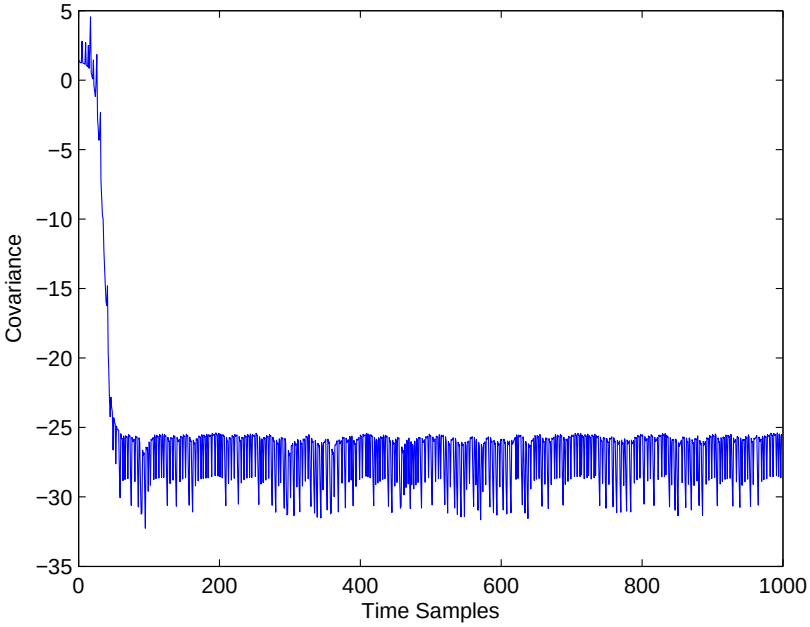

Figure 4.2: $P_{11}(k)$ of the error covariance.

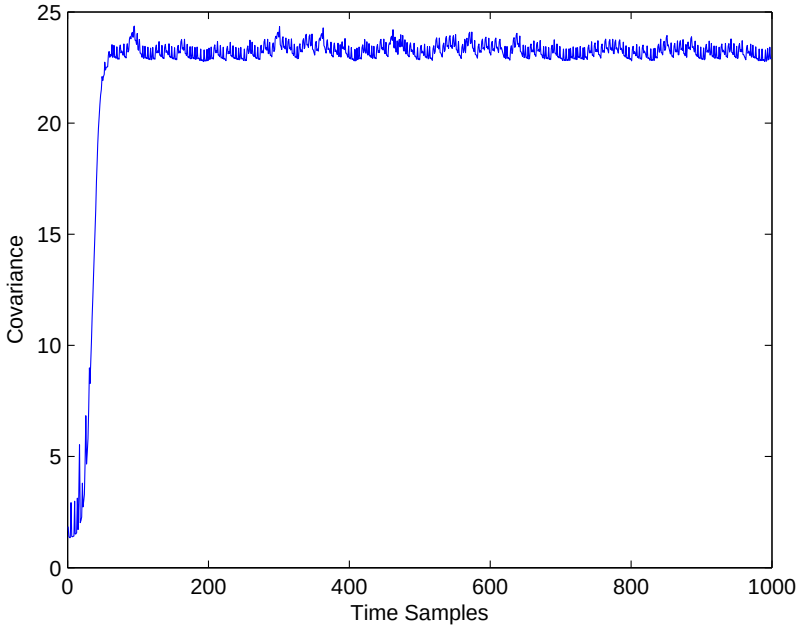

Figure 4.3: $P_{12}(k)$ of the error covariance.

$$T^1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} T^2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

$$S^1_{01}(\tau) = s^1_0 \exp(-|\tau|),\ S^1_{10}(\tau) = s^1_1 \exp(-|\tau - 4|)$$
$$S^2_{01}(\tau) = s^2_0 \exp(-|\tau|),\ S^2_{10}(\tau) = s^2_1 \exp(-|\tau - 3|)$$

with s^j_i satisfying that $\sum_{\tau=1}^{\infty} S^1_{ij}(\tau) = \sum_{\tau=0}^{\infty} S^2_{ij}(\tau) = 1$.

We conducted 50 Monte-Carlo simulations for 60 samples and computed the average of them. Fig 4.4 compares the true and estimated states using the proposed algorithm. The RMSE is plotted in fig 4.5. These figures show excellent agreement with the algorithm developed.

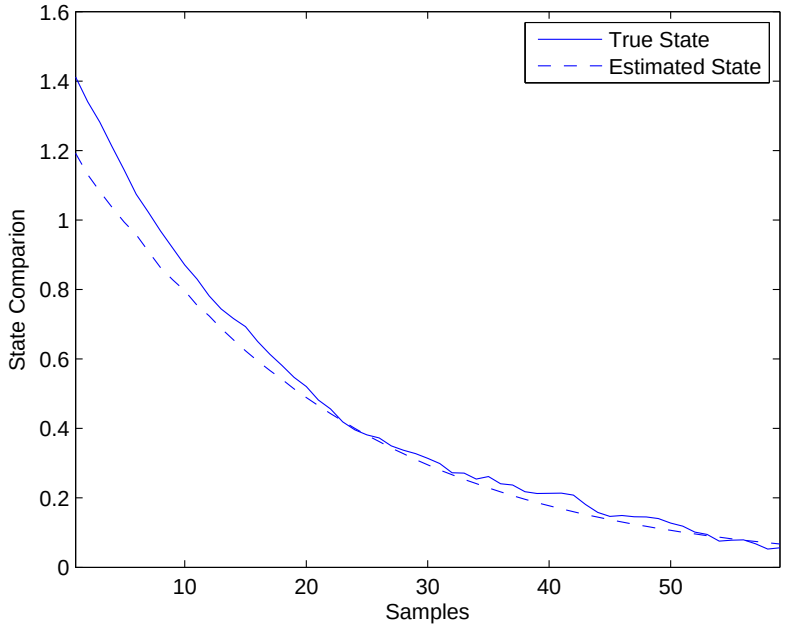

Figure 4.4: Comparison between the true states and the estimated ones.

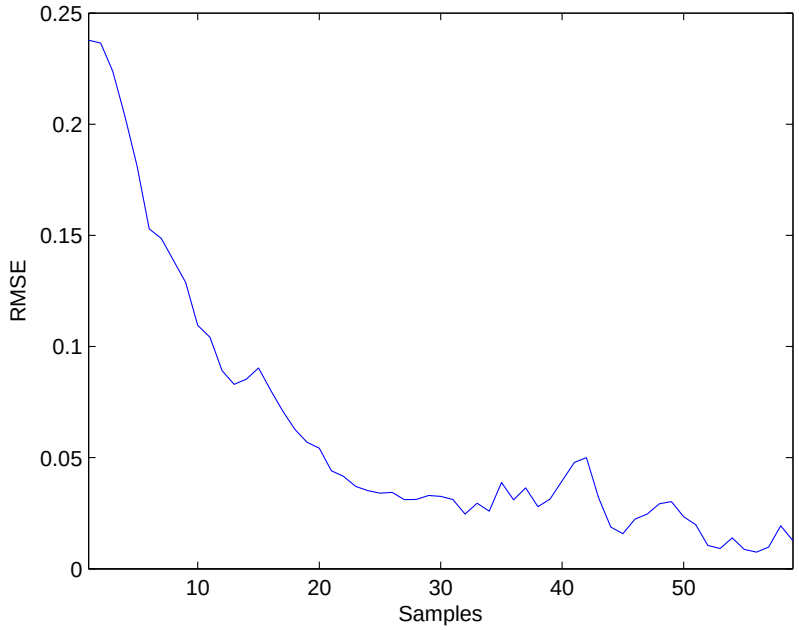

Figure 4.5: RMSE of the proposed estimator.

4.5 Applications

In this section, we provide some applications where this work may apply. Furthermore, we will see that the work is not only suitable for control systems but also for the modeling and identification of CR communication channels.

CR systems in Large-Scale Systems

CR systems have a promising prospective future in large-scale systems due to high demand of bandwidth during the interaction and communication between each subsystem. The accuracy of state estimation of the system over CR links becomes an important issue due to the tradeoff of the benefit of a large amount of free bandwidth and intermittent interruptions from PUs. More specifically, consider the problem of the localization of mobile stations (MS) in wireless networks [49], where high wireless communications are needed and thus is very suitable to the application of CR systems.

The general form of a MS together with a CR link can be written as follows by combining the argument in section 4.1 and [49]:

$$\begin{aligned} x_{k+1} &= f(x_k) + v_k \\ y_k &= \gamma_t g(x_k) + \omega_k \end{aligned} \tag{4.32}$$

where x_k is the state vector including user's Cartesian coordinates and velocities of the MS in the X and Y directions; $f(\cdot)$ and $h(\cdot)$ are known vector functions. To estimate the state accurately, the methods developed in this chapter can be applied by substituting Kalman filter by the extended Kalman filter for the nonlinearity. In this example, let

$$\begin{aligned} A_{k-1} &= \frac{\partial f}{\partial x} \mid_{\hat{x}_{k-1|k-1}} \\ C_k &= \frac{\partial g}{\partial x} \mid_{\hat{x}_{k|k-1}} \end{aligned} \tag{4.33}$$

Replace A and C in (4.10)$\sim$(4.13) by A_{k-1} and C_k, then the estimator is obtained when γ_k is known. Similarly, substituting A and C in the Kalman Filter in step (2) and (3) in the IMM algorithm with A_k and C_{k+1}, the state estimator when γ_k is unknown can be obtained.

Modeling of CR Communication Links

Another application is channel state estimation. Traditionally, wireless communication channels can be represented by time varying discrete-time state-space model comprised of the state equation and the measurement equation [50][101]:

$$\begin{aligned} x_{k+1} &= A_k x_k + v_k \\ y_k &= C_k x_k + \omega_k \end{aligned} \tag{4.34}$$

It directly follows that under CR systems, the measurement received from the channel can be written as $y_k = \gamma_k C_k x_k + \omega_k$. Channel state estimation for this example can be performed by plugging A_k and C_k into the the technique developed in this chapter. The estimated states are used to calculate link gains for power control algorithms developed in [47, 48]. For example, signal attenuation coefficients between the mobile j and the base station i can be represented in terms of the state estimate by

$$F_k^{ij} = e^{G\hat{x}_{k|k}}, \quad G = -\frac{ln(10)}{20}$$

Then, power control schemes can be developed based on this channel information (see section 3 in [48] for more details). However, the parameter estimation algorithm for the channel [101] needs to be updated correspondingly as measurements are no longer Gaussian. This will be addressed in future research.

Encoder and Decoder Design in Control/Communication Systems

There are extensive works on analysis and design problems involving control of deterministic and stochastic systems over communication channels with limited channel capacity [51], and on applications in which communication data rates are limited and the feedback is available from the output of the channel to the input of the channel as shown in fig 4.6 [52]. Take the CR model considered in this work as the feedback communication channel, i.e., the encoder transmits the message through a CR link to the decoder.

Assume γ_k is known to both the encoder and decoder, then, the encoder and the decoder can be designed as the innovation process $y_k - C\hat{x}_{k|k-1}$ and the state estimator which generates $\hat{x}_{k|k}$ through (4.10)$\sim$(4.13), respectively [52].

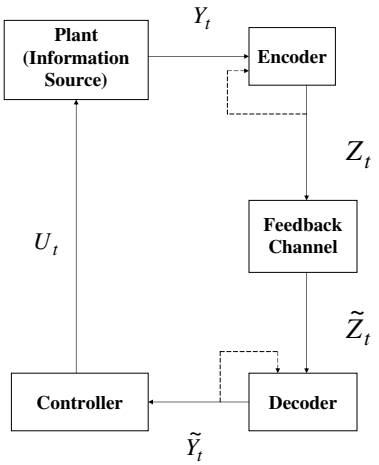

Figure 4.6: Control/communication system

Chapter 5

Control over Cognitive Radio Links Modeled by Semi-Markov Processes

The previous chapter discussed how to obtain the state estimator over semi-Markov model based CR systems. In this chapter, we focus on control design of a discrete-time system with a semi-Markov based CR link connected between the controller and the actuator. The problem is formulated as a SMJLS problem, where the techniques from MJLS can be applied. For convenience, we will only consider one channel in the CR system, that is, the CR link is modeled by a semi-Markov process. In the sequel, we first formulate the problem; we then compute the optimal control for the discrete-time SMJLS with a cost function depend on the sojourn time. Later, we will design a suboptimal controller tractable through LMIs. Simulation results are provided to demonstrate the controller computed.

5.1 Problem Formulation

The discrete-time model considered for control design over the CR link can be represented as follows:

$$x_{k+1} = Ax_k + \iota_k Bu_k \tag{5.1}$$

where ι_k represents the packet loss indicator of the CR link. Simply consider one channel in the CR system, thus ι_k is a semi-Markov process with property discussed in section 4.2. The aim of control design is to compute the control signal to stabilize (5.1). Because (5.1) is a SMJLS, in the sequel, we first discuss control design for the general SMJLS and then apply it to (5.1).

Consider the general SMJLS described by

$$x_{k+1} = A(r_k)x_k + B(r_k)u_k \tag{5.2}$$

where r_k is a finite state semi-Markov random process with state-space $\{1, ..., r\}$. Denote the residence time in some mode until the current time k by $\varrho(k)$. Note the difference between the residence time and the sojourn time is that the former means the time already spent on one state, while the latter means the time spent on one state before jumping to another state.

The cost criterion is given by

$$J(u) = \mathbb{E}\left[\sum_{l=k}^{N-1} x_l^{'} Q_{r_l} x_l + u_l^{'} R_{r_l} u_l\right] \tag{5.3}$$

where Q_{r_l}, R_{r_l} are positive-semidefinite matrices. The optimal controller is to compute some $u_k^*,\ k = 0, 1, ..., N-1$ that minimizes $V(k, x, i, \varrho)$ where

$$\begin{aligned} V(k, x, i, \varrho) &= \min_{u_k} \mathbb{E}\left[\sum_{l=k}^{N-1} x_l^{'} Q_{r_l} x_l + u_l^{'} R_{r_l} u_l | x_k = x, r_k = i, \varrho(k) = \varrho\right] \\ V(N, x, i, \varrho) &= x_N^{'} Q(r_N) x_N \end{aligned} \tag{5.4}$$

However, as shown in the next section, the optimal solution needs to solve a batch of coupled Riccati equations which are difficult to compute explicitly. Thus, we also provide a tractable method by using LMIs in section 5.3.

5.2 Optimal Controller Design

Control design of the discrete-time SMJLS can employ dynamic programming similar as that for the continuous-time SMJLS [72]. Moreover, [73] shows that the design for both discrete-time and contnuous-time are unified through a δ-operator approach. Although there is some similarity between them, we still derive the optimal controller for the discrete-time SMJLS.

Start from

$$\begin{aligned} V(k,x,i,\varrho) &= \min_{u_k} \mathbb{E}[x_k^{'} Q_{r_k} x_k + u_k^{'} R_{r_k} u_k + V(k+1, x_{k+1}, r_{k+1}, \varrho(k+1)) \\ &\quad |x_k = x, r_k = i, \varrho(k) = \varrho] \end{aligned} \tag{5.5}$$

As a trial solution to (5.5), let

$$V(k,x,i,\varrho) = x_k^{'} K(k,i,\varrho) x_k \tag{5.6}$$

Substituting in (5.5) yields

$$\begin{aligned} &x_k^{'} K(k,i,\varrho) x_k = \min_{u_k} [x_k^{'} Q_{r_k} x_k + u_k^{'} R_{r_k} u_k + (A(i)x_k + B(i)u_k)' \\ &\mathbb{E}\{K(k+1, r_{k+1}, \varrho(k+1)) | x_k = x, r_k = i, \varrho(k) = \varrho\}(A(i)x_k + B(i)u_k)] \end{aligned} \tag{5.7}$$

where

$$\begin{aligned}
& \mathbb{E}\{K(k+1, r_{k+1}, \varrho(k+1))|x_k = x, r_k = i, \varrho(k) = \varrho\} \\
= & \sum_{j=1}^{r} K_j(k+1, \varrho(k+1)) p(r_{k+1} = j | r_k = i, \varrho(k) = \varrho) \\
= & \sum_{j=1, j\neq i}^{r} K_j(k+1, \varrho(k+1) = 1) p(r_{k+1} = j | r_k = i, \varrho(k) = \varrho) \\
& + K_i(k+1, \varrho(k+1) = \varrho + 1) p(r_{k+1} = i | r_k = i, \varrho(k) = \varrho) \qquad (5.8)
\end{aligned}$$

and $K_j(k+1, \varrho(k+1)) := K(k+1, j, \varrho(k+1))$.

Denote by $\bar{p}_{ij}(\varrho) := p(r_{k+1} = j | r_k = i, \varrho(k) = \varrho)$ and $\bar{p}_{ii}(\varrho) := p(r_{k+1} = i | r_k = i, \varrho(k) = \varrho)$, we have

$$\bar{p}_{ij}(\varrho) = \frac{p_{ij} S_{ij}(\varrho)}{T_i(\varrho)}$$

$$\bar{p}_{ii}(\varrho) = 1 - \sum_{j=1, j\neq i}^{r} \frac{p_{ij} S_{ij}(\varrho)}{T_i(\varrho)}$$

where $T_i(\varrho) = p(\tau(k) = \varrho | r_k = i)$ denotes the probability of a residence time ϱ on state i until k (assumed homogeneous to be independent with k).

Substituting (5.8) back to (5.7), we obtain

$$\begin{aligned}
& x_k^{'} K_i(k, \varrho) x_k = \min_{u_k} [x_k^{'} Q_{r_k} x_k + u_k^{'} R_{r_k} u_k + (A(i)x_k + B(i)u_k)' \times \\
& \Big(\sum_{j=1, j\neq i}^{r} K_j(k+1, 1) \bar{p}_{ij}(\varrho) + K_i(k+1, \varrho+1) \bar{p}_{ii}(\varrho) \Big) \times (A(i)x_k + B(i)u_k)] \qquad (5.9)
\end{aligned}$$

Differentiating (5.9) with respect to u_k yields the optimal control

$$\begin{aligned}
& u_k^*(x, i, \varrho) \\
= & -\left[R_i + B(i)' \Big(\sum_{j=1, j\neq i}^{r} K_j(k+1, 1) \bar{p}_{ij}(\varrho) + K_i(k+1, \varrho+1) \bar{p}_{ii}(\varrho) \Big) B(i) \right]^{-1} \\
& \times B(i)' \Big(\sum_{j=1, j\neq i}^{r} K_j(k+1, 1) \bar{p}_{ij}(\varrho) + K_i(k+1, \varrho+1) \bar{p}_{ii}(\varrho) \Big) A(i) x_k \qquad (5.10)
\end{aligned}$$

Substituting (5.10) into (5.9) and after some manipulations, we can obtain the expression for $K_i(k,\varrho)$:

$$\begin{aligned} K_i(k,\varrho) &= \left[A(i)'(\sum_{j=1,j\neq i}^{r} K_j(k+1,1)\bar{p}_{ij}(\varrho)+K_i(k+1,\varrho+1)\bar{p}_{ii}(\varrho))A(i)+Q_i\right] \\ &\quad -A(i)'(\sum_{j=1,j\neq i}^{r} K_j(k+1,1)\bar{p}_{ij}+K_i(k+1,\varrho+1)\bar{p}_{ii}(\varrho))'B(i) \\ &\quad \times\left[R_i+B(i)'(\sum_{j=1,j\neq i}^{r} K_j(k+1,1)\bar{p}_{ij}(\varrho)+K_i(k+1,\varrho+1)\bar{p}_{ii}(\varrho))B(i)\right] \\ &\quad \times B(i)'(\sum_{j=1,j\neq i}^{r} K_j(k+1,1)\bar{p}_{ij}(\varrho)+K_i(k+1,\varrho+1)\bar{p}_{ii}(\varrho))A(i) \end{aligned} \tag{5.11}$$

When $k\to\infty$, we have the steady-state solution

$$\begin{aligned} K_i^{\infty}(\varrho) &= \left[A(i)'(\sum_{j=1,j\neq i}^{r} K_j^{\infty}(1)\bar{p}_{ij}(\varrho)+K_i^{\infty}(\varrho+1)\bar{p}_{ii}(\varrho))A(i)+Q_i\right] \\ &\quad -A(i)'(\sum_{j=1,j\neq i}^{r} K_j^{\infty}(1)\bar{p}_{ij}(\varrho)+K_i^{\infty}(\varrho+1)\bar{p}_{ii}(\varrho))'B(i) \\ &\quad \times\left[R_i+B(i)'(\sum_{j=1,j\neq i}^{r} K_j^{\infty}(1)\bar{p}_{ij}(\varrho)+K_i^{\infty}(\varrho+1)\bar{p}_{ii}(\varrho))B(i)\right] \\ &\quad \times B(i)'(\sum_{j=1,j\neq i}^{r} K_j^{\infty}(1)\bar{p}_{ij}(\varrho)+K_i^{\infty}(\varrho+1)\bar{p}_{ii}(\varrho))A(i) \end{aligned} \tag{5.12}$$

This procedure derive the optimal controller for general SMJLSs, for the problem of control over the CR link (5.1), applying equations above, we have the optimal control

$$\begin{aligned} u_{0,k}^{*}(\varrho) &= 0 \\ u_{1,k}^{*}(\varrho) &= -\left[R_1+B'(K_0(k+1,1)\bar{p}_{10}(\varrho)+K_1(k+1,\varrho+1)\bar{p}_{11}(\varrho))B\right]^{-1} \\ &\quad \times B'(K_0(k+1,1)\bar{p}_{10}(\varrho)+K_1(k+1,\varrho+1)\bar{p}_{11}(\varrho))Ax_k \end{aligned} \tag{5.13}$$

with

$$\begin{aligned}
K_0(k,\varrho) &= [A'(K_1(k+1,1)\bar{p}_{01}(\varrho) + K_0(k+1,\varrho+1)\bar{p}_{00}(\varrho))A + Q_0] \\
K_1(k,\varrho) &= [A'(K_0(k+1,1)\bar{p}_{10}(\varrho) + K_1(k+1,\varrho+1)\bar{p}_{11}(\varrho))A + Q_1] \\
&\quad -A'(K_0(k+1,1)\bar{p}_{10}(\varrho) + K_1(k+1,\varrho+1)\bar{p}_{11}(\varrho))'B \\
&\quad \times [R_1 + B'(K_0(k+1,1)\bar{p}_{10}(\varrho) + K_1(k+1,\varrho+1)\bar{p}_{11}(\varrho))B] \\
&\quad \times B'(K_0(k+1,1)\bar{p}_{10}(\varrho) + K_1(k+1,\varrho+1)\bar{p}_{11}(\varrho))A \qquad (5.14)
\end{aligned}$$

When $k \to \infty$, we have the steady-state solution

$$\begin{aligned}
K_0^{\infty}(\varrho) &= [A'(K_1^{\infty}(1)\bar{p}_{01}(\varrho) + K_0^{\infty}(\varrho+1)\bar{p}_{00}(\varrho))A + Q_0] \\
K_1^{\infty}(\varrho) &= [A'(K_0^{\infty}(1)\bar{p}_{10}(\varrho) + K_1^{\infty}(\varrho+1)\bar{p}_{11}(\varrho))A + Q_1] \\
&\quad -A'(K_0^{\infty}(1)\bar{p}_{10}(\varrho) + K_1^{\infty}(\varrho+1)\bar{p}_{11}(\varrho))'B \\
&\quad \times [R_1 + B'(K_0^{\infty}(1)\bar{p}_{10}(\varrho) + K_1^{\infty}(\varrho+1)\bar{p}_{11}(\varrho))B] \\
&\quad \times B'(K_0^{\infty}(1)\bar{p}_{10}(\varrho) + K_1^{\infty}(\varrho+1)\bar{p}_{11}(\varrho))A \qquad (5.15)
\end{aligned}$$

where $\bar{p}_{01}(\varrho) = \frac{S_{01}(\varrho)}{T_0(\varrho)}$ and $\bar{p}_{10}(\varrho) = \frac{S_{10}(\varrho)}{T_1(\varrho)}$. When $\varrho \to \infty$, $T_1(\infty) = T_0(\infty) = 0$ and $\bar{p}_{01}(\infty) = \bar{p}_{10}(\infty) = 1$.

It is difficult to compute the optimal solution from these expressions because of coupled Riccati equations and $\varrho \in (0, \infty)$. Thus, in the next section, we provide a suboptimal but tractable solution.

5.3 A Suboptimal Controller

To obtain an explicit controller, we assume that when $\varrho \geq T_s$, $T_1(T_s) = T_0(T_s) = 0$ and $\bar{p}_{01}(T_s) = \bar{p}_{10}(T_s) = 1$. This assumption is reasonable as the process can not stay in the same state (0 or 1) forever, thus after a finite time T_s, the probability to transfer from one state to the other one are 1. In this sense, we only need to deal

with finite variables. Before deriving the suboptimal controller, we first discuss the stochastic stability.

5.3.1 Stochastic Stability

In this section, we analyze stochastic stability in the mean square sense for the SMJLS. The definition of the stability is given as follows [77]:

Definition 5. *System (5.2) with $u_k \equiv 0$ is said to be stochastically stable (SS) if*

$$\mathbb{E}\left\{\sum_{k=0}^{\infty} \| x_k \|^2\right\} < \infty$$

for any finite initial condition x_0 and r_0.

From the definition above, we have the following proposition.

Proposition 1. *System (5.2) is SS if there exist matrices $G_i^{\varrho} = (G_i^{\varrho})' > 0, \varrho = 1, ..., T_s$, such that*

$$A(i)' \left[\sum_{j=1, j\neq i}^{r} G_j^1 \bar{p}_{ij}(\varrho) + G_i^{\varrho+1} \bar{p}_{ii}(\varrho)\right] A(i) - G_i^{\varrho} < 0$$

Proof. The proof follows dynamic programming similar as the SMS condition derived for MJLSs in [62]. □

Back to our special case where there are only two states, Proposition 1 becomes:

Proposition 2. *System (5.2) is SS if there exist matrices $G_i^{\varrho} = (G_i^{\varrho})' > 0, \varrho = 1, ..., T_s, i = 0, 1$, such that*

$$A' \left[G_j^1 \bar{p}_{ij}(\varrho) + G_i^{\varrho+1} \bar{p}_{ii}(\varrho)\right] A - G_i^{\varrho} < 0$$

In the following proposition, we show an equivalent testable condition to Proposition 2 in terms of a finite LMI feasibility problem utilizing a similar derivation as in [78].

Proposition 3. *Proposition 2 is true if and only if there exist matrices* $S_i^{\varrho} = (S_i^{\varrho})' > 0$ *and matrices* L_i^{ϱ} *satisfying the following LMIs:*

$$\begin{bmatrix} L_i^{\varrho} + (L_i^{\varrho})' - S_i^{\varrho} & (L_i^{\varrho})' A' \Pi_i^{\varrho} \\ (\Pi_i^{\varrho})' A L_i^{\varrho} & \bar{S} \end{bmatrix} > 0 \tag{5.16}$$

$\forall i = 0, 1, \ \varrho = 1, ..., T_s$

where $\Pi_i^{\varrho} = \begin{bmatrix} \sqrt{p_{i0}(\varrho)}\mathbb{I} & \sqrt{p_{i1}(\varrho)}\mathbb{I} \end{bmatrix}$

and

$\bar{S} = \begin{bmatrix} S_0^{\varrho+1} & \mathbf{0} \\ \mathbf{0} & S_1^1 \end{bmatrix}, if \ i = 0 \ or \ \bar{S} = \begin{bmatrix} S_0^1 & \mathbf{0} \\ \mathbf{0} & S_1^{\varrho+1} \end{bmatrix}, if \ i = 1$

Proof. $(\Rightarrow)$

The system is SS if and only if there exist matrices $\{G_i^{\varrho}\} > 0$, such that:

$$A' \left[G_j^1 \bar{p}_{ij}(\varrho) + G_i^{\varrho+1} \bar{p}_{ii}(\varrho) \right] A - G_i^{\varrho} < 0 \tag{5.17}$$

which can be written as

$$\begin{aligned} & G_i^{\varrho} - A' \left[G_j^1 \bar{p}_{ij}(\varrho) + G_i^{\varrho+1} \bar{p}_{ii}(\varrho) \right] A > 0 \\ \Rightarrow \\ & G_i^{\varrho} - A' \left[\sum_{j=0}^{1} \bar{p}_{ij}(\varrho) G_j^m \right] A > 0 \end{aligned}$$

where $m = 1$, if $i \neq j$, and $m = \varrho + 1$, if $i = j$.

Let $S_i^{\varrho} = (G_i^{\varrho})^{-1}$, and $S_j^m = (G_j^m)^{-1}$, then

$$(S_i^{\varrho})^{-1} - A' \left[\sum_{j=0}^{1} \bar{p}_{ij}(\varrho) (S_j^m)^{-1} \right] A > 0$$

Using the Schur complement [79]

$$\begin{bmatrix} (S_i^\varrho)^{-1} & A'\Pi_i^\varrho \\ (\Pi_i^\varrho)'A & \bar{S} \end{bmatrix} > 0 \tag{5.18}$$

which is equivalent to

$$\bar{S} - (\Pi_i^\varrho)^{-1} A S_i^\varrho A' \Pi_i^\varrho = \Upsilon_i^\varrho > 0 \tag{5.19}$$

Let $L_i^\varrho = S_i^\varrho + \alpha_i^\varrho \mathbb{I}$, with α_i^ϱ positive scalars. There exist $\alpha_i^\varrho > 0$, such that

$$(\alpha_i^\varrho)^{-2}(S_i^\varrho + 2\alpha_i^\varrho \mathbb{I}) > A'\Pi_i^\varrho (\Upsilon_i^\varrho)^{-1} (\Pi_i^\varrho)' A$$

Then, using the Schur complement again, we have

$$\begin{bmatrix} S_i^\varrho + 2\alpha_i^\varrho \mathbb{I} & -\alpha_i^\varrho A'\Pi_i^\varrho \\ -\alpha_i^\varrho (\Pi_i^\varrho)'A & \Upsilon_i^\varrho \end{bmatrix} > 0 \tag{5.20}$$

This can be written as

$$\begin{bmatrix} L_i^\varrho + (L_i^\varrho)' - S_i^\varrho & (S_i^\varrho - L_i^\varrho)A'\Pi_i^\varrho \\ (S_i^\varrho - L_i^\varrho)(\Pi_i^\varrho)'A & \Upsilon_i^\varrho \end{bmatrix} > 0 \tag{5.21}$$

which can be further converted to

$$\begin{bmatrix} L_i^\varrho + (L_i^\varrho)' - S_i^\varrho & (L_i^\varrho)'A'\Pi_i^\varrho \\ (\Pi_i^\varrho)'AL_i^\varrho & \bar{S} \end{bmatrix} > 0 \tag{5.22}$$

($\Leftarrow$)

We have

$$L_i^\varrho + (L_i^\varrho)^{-1} - S_i^\varrho > 0$$

We also have

$$(S_i^\varrho - L_i^\varrho)'(S_i^\varrho)^{-1}(S_i^\varrho - L_i^\varrho) \geq 0$$

or equivalently

$$(L_i^\varrho)'(S_i^\varrho)^{-1}(L_i^\varrho) \geq L_i^\varrho + (L_i^\varrho)^{-1} - S_i^\varrho$$

Then, the inequality becomes

$$\begin{bmatrix} (L_i^\varrho)'(S_i^\varrho)^{-1}(L_i^\varrho) & (L_i^\varrho)'A'\Pi_i^\varrho \\ (\Pi_i^\varrho)'AL_i^\varrho & \bar{S} \end{bmatrix} > 0 \tag{5.23}$$

also implying

$$\begin{bmatrix} (L_i^\varrho)' & \mathbf{0} \\ \mathbf{0} & (\bar{S}) \end{bmatrix} \begin{bmatrix} (S_i^\varrho)^{-1} & A'\Pi_i^\varrho(\bar{S})^{-1} \\ (\bar{S})^{-1}(\Pi_i^\varrho)'A & (\bar{S})^{-1} \end{bmatrix} \begin{bmatrix} L_i^\varrho & \mathbf{0} \\ \mathbf{0} & (\bar{S}) \end{bmatrix} > 0 \tag{5.24}$$

Let $G_i^\varrho = (S_i^\varrho)^{-1}$, then

$$\begin{bmatrix} G_i^\varrho & A'\Pi_i^\varrho\bar{G} \\ \bar{G}(\Pi_i^\varrho)'A & \bar{G} \end{bmatrix} > 0 \tag{5.25}$$

where $\bar{G} = (\bar{S})^{-1}$. By the Schur complement, we obtain

$$A'\left[G_j^1\bar{p}_{ij}(\varrho) + G_i^{\varrho+1}\bar{p}_{ii}(\varrho)\right]A - G_i^\varrho < 0$$

This concludes the proof. □

5.3.2 Control Design

We are looking for a state feedback controller in the form

$$u_k = K_i^\varrho x_k \tag{5.26}$$

where the control law K_i^ϱ depends on both the mode $i = 0, 1$ and $\varrho = 1, ..., T_s$. The following proposition comes out directly from the previous result.

Proposition 4. *The system is stochastically stabilized by (5.26) if there exist matrices* $S_i^\varrho = (S_i^\varrho)' > 0$, L_i^ϱ *and* H_i^ϱ, $\forall i = 0, 1$, $\varrho = 1, ..., T_s - 1$, *such that the following LMIs*

are feasible:

$$\begin{bmatrix} L_i^{\varrho} + (L_i^{\varrho})' - S_i^{\varrho} & ((L_i^{\varrho})'A' + (H_i^{\varrho})'B(i)'))\Pi_i^{\varrho} \\ (\Pi_i^{\varrho})'(AL_i^{\varrho} + B(i)H_i^{\varrho}) & \bar{S} \end{bmatrix} > 0 \tag{5.27}$$

where $B(i) = 0$, *if* $i = 0$, *and* $B(i) = B$, *if* $i = 1$. *The control law can then be calculated through [78]*

$$K_i^{\varrho} = H_i^{\varrho}(L_i^{\varrho})^{-1} \tag{5.28}$$

5.4 Simulation Results

In this section, we provide a numerical example to demonstrate the proposed control design method. Consider the given system parameters:

$A = \begin{bmatrix} 1.1 & 0.1 \\ 0 & 1.2 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 0.7 \end{bmatrix}$

For the semi-Markov CR link, assume transition probabilities: $\bar{p}_{01}(1) = 0.4, \bar{p}_{01}(2) = 0.6, \bar{p}_{01}(3) = 0.75, \bar{p}_{01}(4) = 0.95$, $\bar{p}_{01}(\varrho_0) = 1, \forall \varrho_0 \geq 5$, and $\bar{p}_{10}(1) = 0.3, \bar{p}_{10}(2) = 0.25, \bar{p}_{10}(3) = 0.45, \bar{p}_{10}(4) = 0.6, \bar{p}_{10}(5) = 0.9$, $\bar{p}_{10}(\varrho_1) = 1, \forall \varrho_1 \geq 6$.

The initial state $x_0 = \begin{bmatrix} 3.8 \\ 6.4 \end{bmatrix}$, and applying the technique in the previous section, we obtained the following control law:

$$K_1^1 = [4.70 \ \ -8.61]$$
$$K_1^2 = [4.73 \ \ -8.66]$$
$$K_1^3 = [4.15 \ \ -7.81]$$
$$K_1^4 = [3.95 \ \ -7.51]$$
$$K_1^5 = [4.19 \ \ -7.86]$$

The states of the closed-loop system with designed control input are plotted in fig 5.1 and fig 5.2. We can observe that they converge to 0 as time evolves.

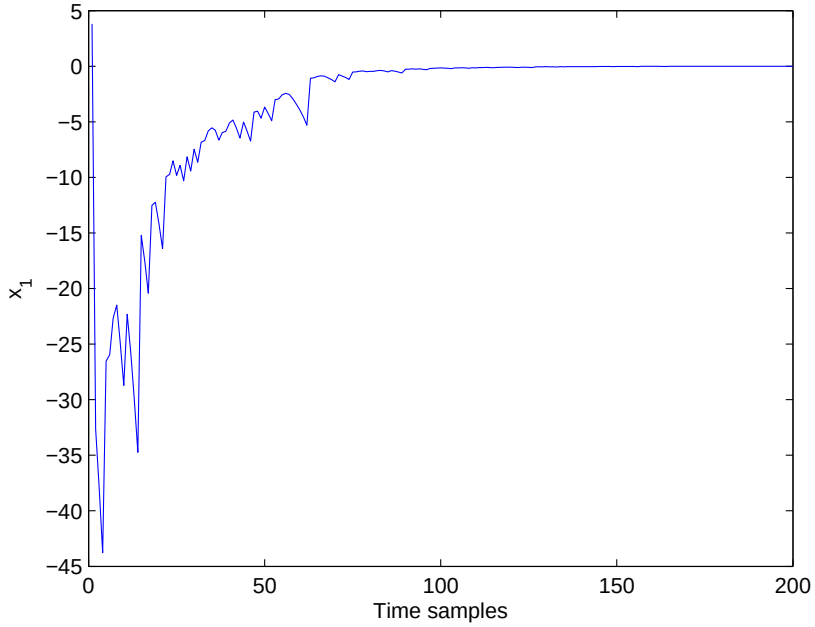

Figure 5.1: Trajectory of x_1.

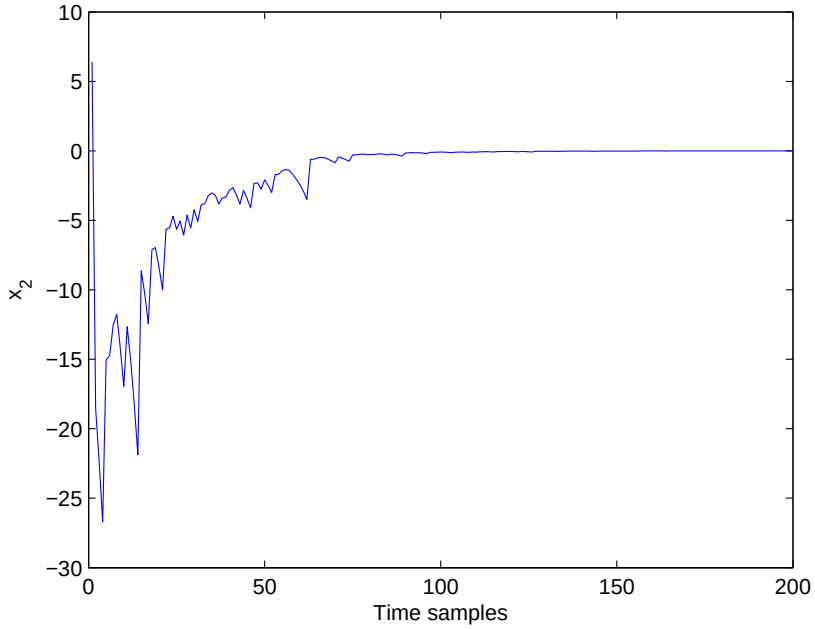

Figure 5.2: Trajectory of x_2.

Chapter 6

LIPS: Link Prediction as a Service for Adaptive Data Aggregation in Wireless Sensor Networks

In previous chapters, we designed control and estimation algorithms for discrete-time linear systems over packet loss and CR links. They are modeled as noise-corrupted communication channels, e.g., the received signal is the summation of the transmitted signal and the Gaussian noise. In this chapter, we shift our attention from control techniques development to the performance of channels, e.g., determining a better channel for the transmission. Many studies have focused on modeling, identification, and estimation of wireless communication channels using a state-space model [46, 47, 48, 45]. We propose a novel idea: the state-space model can be used to predict the link quality, and provide these estimates as a system level service to application developers. This idea is based on the premise that to achieve the best performance, the application-layer behavior should be aware of networking-layer conditions, e.g., in the collection protocol, and adjust its behavior to achieve balanced performance with the link quality. The resulting integrated framework is what we have designated LIPS, or ***Li****nk* ***P****redictions as a* ***S****ervice*, that represents an integrated solution.

First, LIPS presents a state-space based link prediction for selecting the best path. Most approaches for estimating the link quality are currently based on metrics such as packet reception rate (PRR), link quality indicator (LQI), and received signal strength indicator (RSSI) [81, 80]. However, existing approaches have been shown to be limited in their ability to predict the future. This is because using historical data implicitly makes the assumption that future measurements may stay similar, an assumption that is often invalidated by frequent variations of wireless links. Therefore, in this chapter, we tackle this problem by trying to predict the expected link quality using the state-space model. We further integrate such predictions as the foundation for upper-layer protocol adaptations.

Second, in response to link quality changes, LIPS presents a queue management architecture based on modifying the OS kernel to support elastic applications. Specifically, we observe the following trade-off: If the link quality becomes worse, the queues of intermediate nodes will increase due to an increased number of retransmissions. Therefore, we argue that the length of the queues, and especially their changing trends, reflects the link quality and provides additional information to applications. To this end, we provide a suite of APIs for user applications for managing queue operations.

Finally, we demonstrate one case study where the application layer reduces its data rate and performs more aggressive data aggregation. This case study demonstrates that applications can indeed use our APIs to adjust themselves to the link layer realities and validate the feasibility of our approach.

To the best of our knowledge, this is the first integrated framework that aims to improve application-layer data collection services through a co-design of link layer prediction, queue management, and API support. Furthermore, our use of state-space models to predict the link quality is the first as far as we know. Finally, the overall design is effective, based on our preliminary experimental results. Parts of this work have been published in [126].

6.1 The State-Space Model and Parameters Estimation

6.1.1 State Space Model

In this section, we describe the state-space model for predicting RSSI and PRR readings. The state equation can be written as a stochastic difference equation (SDE):

$$x(k+1) = A(k)x(k) + B(k)w(k) \tag{6.1}$$

where $x(k+1) \in \mathbb{R}^{n\times 1}$ is the state vector of the time series $\{x(k)\}_k$ determined by the previous state $x(k)$ and the noise term $w(k) \in \mathbb{R}^{m\times 1}$ introduced at each k; $A(k) \in \mathbb{R}^{n\times n}$ and $B(k) \in \mathbb{R}^{n\times m}$ are coefficients that affect amplitudes of $x(k)$ and $w(k)$ at each k.

The measurement equation can be written as:

$$y(k) = C(k)x(k) + D(k)v(k) \tag{6.2}$$

where $y(k) \in \mathbb{R}^{l\times 1}$ is the measurement generated by $x(k)$ and the noise term $v(k) \in \mathbb{R}^{m\times 1}$; $C(k) \in \mathbb{R}^{l\times n}$ and $D(k) \in \mathbb{R}^{l\times m}$ are the corresponding coefficients. Note $x(k)$ characterizes the variety and the evolution of the series $\{x(k)\}_k$. With time increasing, $x(k)$ is involving through (6.1) and then affects $y(k)$ through (6.2). This property makes it very suitable to model random measurements as the time variety of measurements is transplanted to that of states. Also note $w(k)$ and $v(k)$ can represent small perturbations or uncertainties which increase the flexibility of the model.

Due to theses special characteristics, we propose to use the state-space model to track, model and predict stochastic behaviors of RSSI/PRR. Specifically, $y(k)$ in (6.2) is used to denote the value of RSSI/PRR at time k. For a better illustration, we state the procedure on how to predict future measurements of RSSI/PRR as follows as well

as in fig 6.1:

Prediction: *Given a batch of RSSI/PRR measurements* $\{y(k)\}_{k=0}^{K}$*, then the prediction of future measurements can be divided into three steps:*

- *First, measurements are characterized and governed by the state-space model (6.3);*
- *Second, a parameter estimation algorithm (section 6.1.2) is employed to compute the parameters (e.g.* $A(k), B(k), C(k), D(k)$*, and* $x(0)$*) in the model;*
- *Last, future measurements can be predicted explicitly by Lemma 6.1.2.*

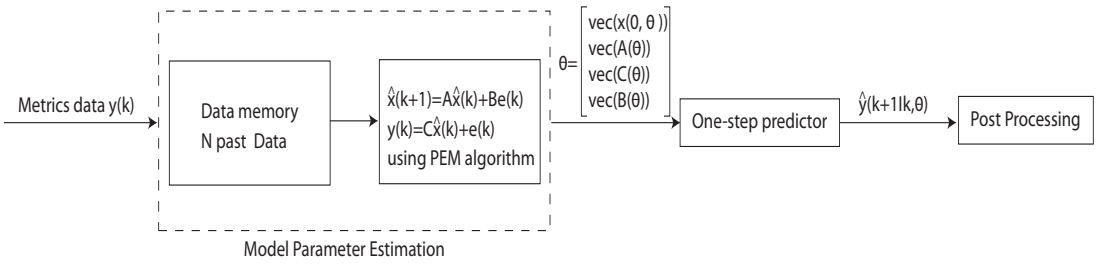

Figure 6.1: The one-step-ahead prediction process of RSSI/PRR measurements. Once new measurements are provided, this process is repeated.

First we introduce the following definition about the multi-step-ahead prediction.

Definition 6. *The* N_p *multi-step-ahead prediction of* $y(k)$ *is a prediction at the time instant* $k + N_p$ *making use of measurements* $y(l), l \leq k$*. It is denoted by*

$$\hat{y}(k + N_p|k, \theta)$$

where θ denotes the parameters of the model and will be described in details in section 6.1.2.

The state-space predictor used to model RSSI/PRR measurements can be written as

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Be(k), \;\; \hat{x}(0) = x_0 \\ y(k) &= C\hat{x}(k) + e(k) \end{aligned} \tag{6.3}$$

where $y(k)$ is the RSSI/PRR measurement; $\hat{x}(k)$ is the state variable characterizing the link property; A, B, and C are coefficients of the model; $x(0)$ denotes the initial state condition; $e(k) = y(k) - C\hat{x}(k)$ is the error between the measured value and the predicted value $C\hat{x}(k)$ at k. A, B, C and initial state condition $\hat{x}(0)$ in (6.3) are unknown and need to be estimated through RSSI/PRR measurements $y(1), y(2), ..., y(k)$ and then used to predict future measurements $\hat{y}(k + N_p|k, \theta)$. Because RSSI/PRR measurements are scalars, thus in (6.3), $y(k) \in \mathbb{R}^{1\times 1}$, $e(k) \in \mathbb{R}^{1\times 1}$ and $B \in \mathbb{R}^{n\times 1}$.

In the next section, we are going to introduce the algorithm on how to estimate the parameters of the model and predict future link quality metrics.

6.1.2 Prediction Error Minimization (PEM) Algorithm

There are several parameter estimation algorithms, e.g. expectation and maximization [106], which yields the maximum likelihood parameter estimate. In this section, we introduce a parameter estimation algorithm – Prediction Error Minimization Algorithm [114]. In the sequel, we will use x to denote $\hat{x}$ in (6.3) for the sake of convenience.

We first need to parameterize the model (6.3). The aim of parametrization is to unify the parameters of the model into a vector variable to facilitate the parameter estimation. Assume entries of parameters A, B, C, and $x(0)$ depend on a parameter

vector θ, then (6.3) can be written as:

$$\begin{aligned} x(k+1|k,\theta) &= A(\theta)x(k|k-1,\theta) + B(\theta)e(k) \\ y(k) &= C(\theta)x(k|k-1,\theta) + e(k) \end{aligned} \tag{6.4}$$

and $x(0)$ is parameterized by $x(0,\theta)$.

It is obvious that the dimensions of A, B, C, $x(k)$, $y(k)$ and $e(k)$ are $n \times n$, $n \times 1$, $1 \times n$, $n \times 1$, 1×1 and 1×1, respectively. Then, the parameter vector θ including all entries of matrices A, B and C as well as the initial state conditions $x(0)$ can be written as

$$\theta = \begin{bmatrix} vec(x(0,\theta)) \\ vec(A(\theta)) \\ vec(C(\theta)) \\ vec(B(\theta)) \end{bmatrix}$$

where $vec(M)$ is the operator vectorizing matrix M by stacking its columns. Thus, the dimension of θ is $q \times 1$, where $q = n + n \times n + 1 \times n + n \times 1 = 3n + n^2$. We also provide an example to explain the parametrization below.

Example 1. *Consider the model:*

$$\begin{aligned} x(k+1) &= Ax(k) + Be(k), x(0) = x_0 \\ y(k) &= Cx(k) + e(k) \end{aligned} \tag{6.5}$$

where

$$A = \begin{bmatrix} 0.5 & 1.2 \\ 1.1 & 0.6 \end{bmatrix}, B = \begin{bmatrix} 0.7 \\ 0.9 \end{bmatrix}, C = [1\ 0], x_0 = [12]^T$$

If the model is parameterized with all entries of the parameter matrices, then the following parametric model is obtained:

$$\begin{aligned} x(k+1|k,\theta) &= \begin{bmatrix} \theta(3)\ \theta(4) \\ \theta(5)\ \theta(6) \end{bmatrix} x(k|k-1,\theta) + \begin{bmatrix} \theta(9) \\ \theta(10) \end{bmatrix} e(k), \\ x(0,\theta) &= [\theta(1)\ \theta(2)]^T \\ y(k) &= [\theta(7)\ \theta(8)]x(k|k-1,\theta) + e(k) \end{aligned} \tag{6.6}$$

where the parameter vector $\theta = [\theta(1), ..., \theta(10)]^T$.

After parametrization, we must estimate θ.

As it stands, PEM estimates parameters by minimizing the prediction error. Here, we will concentrate on the case $N_p = 1$ where only the one-step-ahead prediction is involved. Thus, given a finite number of measurements N, PEM estimates θ by minimizing a least square cost function with respect to θ:

$$J_N(\theta) = \frac{1}{N} \sum_{k=0}^{N-1} \parallel y(k) - \hat{y}(k|k-1,\theta) \parallel_2^2 \tag{6.7}$$

where $\hat{y}(k|k-1,\theta)$ is the one-step-ahead prediction. A more specific form of $J_N(\theta)$ is given in the following theorem.

Theorem 6.1. *The functional* $J_N(\theta)$ *can be written as*

$$J_N(\theta) = \frac{1}{N} \sum_{k=0}^{N-1} \parallel y(k) - \phi(k,\theta)[x(0,\theta), B(\theta)]^T \parallel_2^2$$

where the matrix $\phi(k,\theta)$ *with dimension* $1 \times 2n$ *is explicitly given as*

$$\phi(k,\theta) = [C(\theta)(A(\theta) - B(\theta)C(\theta))^k \sum_{\tau=0}^{k-1} y^T(\tau) \otimes C(\theta)(A(\theta) - B(\theta)C(\theta))^{k-1-\tau}]$$

where $\otimes$ *is Kronecker product. That means*

$$\hat{y}(k|k-1,\theta) = C(\theta)(A(\theta) - B(\theta)C(\theta))^k x(0,\theta)$$

$$+ \sum_{\tau=0}^{k-1} C(\theta)(A(\theta) - B(\theta)C(\theta))^{k-1-\tau} B(\theta) y(\tau) \tag{6.8}$$

Proof. Follow the proof of Theorem 8.1 on page 262∼263 in [115] and set input $u(k)$ equal to 0. □

In order to compute θ, we employ the Gauss-Newton method [115] to numerically minimize the cost function. Define the error vector $E_N(\theta) = [\epsilon(0,\theta), \epsilon(1,\theta), ..., \epsilon(N-1,\theta)]^T$ where $\epsilon(k,\theta) = y(k) - \hat{y}(k|k-1,\theta)$. Note the difference between $\epsilon(k,\theta)$ and $e(k)$ is that the former is a function of θ. Then the cost function $J_N(\theta)$ can be written as

$$J_N(\theta) = \frac{1}{N} \sum_{k=0}^{N-1} \parallel y(k) - \hat{y}(k|k-1,\theta) \parallel_2^2 = \frac{1}{N} E_N^T(\theta) E_N(\theta) \tag{6.9}$$

Also, define the derivative of $E_N(\theta)$ with the notation

$$\Psi_N(\theta) = \frac{\partial E_N(\theta)}{\partial \theta^T}$$

Then the Jacobian and Hessian of $J_N(\theta)$ (first derivative and second derivative of $J_N(\theta)$) can be expressed as [115]

$$J'_N(\theta) = \frac{\partial J_N(\theta)}{\partial \theta} = \frac{2}{N} \Psi_N^T(\theta) E_N(\theta) \tag{6.10}$$

$$J"_N(\theta) = \frac{\partial^2 J_N(\theta)}{\partial \theta \partial \theta^T} = \frac{2}{N} \frac{\partial^2 E_N^T(\theta)}{\partial \theta^T \theta} (I_p \otimes E_N(\theta)) + \frac{2}{N} \Psi_N^T(\theta) \Psi_N(\theta) \tag{6.11}$$

where I_p is $p \times p$ identity matrix.

The Gauss-Newton method approximates the Hessian by the matrix $H_N(\theta) = \frac{2}{N} \Psi_N^T(\theta) \Psi_N(\theta)$, where the first term is neglected and thus saves high computation

cost. When $H_N(\theta)$ is invertible, the method updates θ by

$$\theta_{i+1} = \theta_i - H_N(\theta_i)^{-1} J'_N(\theta_i) \tag{6.12}$$

where the index i denotes the ith iteration. The derivation of the update equation can be found in any optimization book, e.g. [115]. Once the cost function $J_N(\theta_i)$ reaches a tolerable threshold, the iteration can be stopped at $\theta = \theta_i$.

Remark: The matrix $H_N(\theta)$ may be singular. One possible way to solve this is through regularization, where a penalty term is added to the cost function to address the singularity. Instead of minimizing $J_N(\theta)$, the problem becomes $min_\theta J_N(\theta) + \lambda \parallel \theta \parallel_2^2$ and the update equation (6.12) is rewritten as

$$\theta_{i+1} = \theta_i - (H_N(\theta_i) + \lambda I_p)^{-1} J'_N(\theta_i) \tag{6.13}$$

where $\lambda > 0$ and $H_N(\theta_i) + \lambda I_p$ is made non-singularity.

Note that the Gauss-Newton method is one of the optimization algorithms to minimize a function, other algorithms such as *steepest descent method*, and *gradient projection* are also applicable to our problem.

For convenience, we use $n = 1$ dimension state-space model to model PRR/RSSI. Next, we are going to derive explicit equations of $\Psi_N(\theta)$ in this case. For the case when $n > 1$, the equations can be derived similarly.

Lemma 6.1.1. *If $n = 1$, then we have*

$$\Psi_N(0, \theta) = [C(\theta),\ 0,\ x(0, \theta),\ 0]$$

$$\Psi_N(1, \theta) = [C(\theta)(A(\theta) - B(\theta)C(\theta)),\quad C(\theta)x(0, \theta),$$

$$(A(\theta) - 2B(\theta)C(\theta))x(0, \theta) + B(\theta)y(0),\quad -C(\theta)^2 x(0, \theta) + C(\theta)y(0)]$$

And for $1 < k < N - 1$, $\Psi_N(k, \theta)$ is computed in (6.14),

Proof. From the definition of $E_N(\theta)$ and $\Psi_N(k,\theta)$, we have

$$\Psi_N(k,\theta) = \frac{\partial E_N(k,\theta)}{\partial \theta^T}$$

$$= [\frac{\partial E_N(k,\theta)}{\partial x(0,\theta)}, \frac{\partial E_N(k,\theta)}{\partial A(\theta)}, \frac{\partial E_N(k,\theta)}{\partial C(\theta)}, \frac{\partial E_N(k,\theta)}{\partial B(\theta)}]$$

By computing each derivative term above, $\Psi_N(k,\theta)$ in the lemma can be obtained.

□

$$\Psi_N(k,\theta) = \begin{bmatrix} C(\theta)(A(\theta) - B(\theta)C(\theta))^k \\ \\ kC(\theta)(A(\theta) - B(\theta)C(\theta))^{k-1}x(0,\theta) + \sum_{\tau=0}^{k-1}(k-\tau-1) \\ \times C(\theta)(A(\theta) - B(\theta)C(\theta))^{k-\tau-2}B(\theta)y(\tau), \\ \\ (A(\theta) - B(\theta)C(\theta))^k x(0,\theta) - kC(\theta)B(\theta)(A(\theta) - B(\theta)C(\theta))^{k-1}x(0,\theta) \\ + \sum_{\tau=0}^{k-1}((A(\theta) - B(\theta)C(\theta))^{k-\tau-1}B(\theta)y(\tau) - (k-\tau-1)C(\theta)B(\theta) \\ \times (A(\theta) - B(\theta)C(\theta))^{k-\tau-2}B(\theta)y(\tau)), \\ \\ -kC(\theta)^2(A(\theta) - B(\theta)C(\theta))^{k-1}x(0,\theta) + \sum_{\tau=0}^{k-1}(C(\theta) \\ \times (A(\theta) - B(\theta)C(\theta))^{k-\tau-1}y(\tau) - (k-\tau-1)C(\theta)^2 \\ \times (A(\theta) - B(\theta)C(\theta))^{k-\tau-2}B(\theta)y(\tau)) \end{bmatrix} \tag{6.14}$$

The next lemma provides a procedure to compute the multi-step-ahead prediction after the parameters have been estimated.

Lemma 6.1.2. *Given the model structure (6.4) and quantities $x(0,\theta)$, $A(\theta)$, $B(\theta)$, $C(\theta)$ and $\{y(l)\}_0^k$, then the one-step-ahead prediction is computed as:*

$$\begin{aligned} x(k+1|k,\theta) &= (A(\theta) - B(\theta)C(\theta))^{k+1}x(0,\theta) + \sum_{\tau=0}^{k}(A(\theta) - B(\theta)C(\theta))^{k-\tau}B(\theta)y(\tau) \\ \hat{y}(k+1|k,\theta) &= C(\theta)x(k+1|k,\theta) \end{aligned} \tag{6.15}$$

and based on the one-step-ahead prediction, the multi-step-ahead prediction where $N_p > 1$ *can be computed by:*

$$\begin{aligned} x(k+N_p|k,\theta) &= A(\theta)^k x(k+1|k,\theta) \\ \hat{y}(k+N_p|k,\theta) &= C(\theta)x(k+N_p|k,\theta) \end{aligned} \tag{6.16}$$

Proof. Follow Lemma 8.2 on page 260 in [115] and set input $u(k)$ equal to 0. □

6.2 Elastic Queue Management

Having described the mathematical foundation of LIPS, in this section, we describe the elastic queue management by modifying how the operating system handles incoming packets. We chose the LiteOS operating system [116], an in-house experimental operating system for our purpose. Given that the LiteOS system does not have integrated support for queueing, we modified its communication stack to incorporate dynamic memory management, and implement queueing through a doubly linked list data structure.

Fig 6.2 shows the communication stack that serves as the foundation for our queue management model. In this figure, both the receiving (on the left) and the sending (on the right) operations are illustrated. When the sender intends to deliver packets, it puts the destination address and the port number for the destination node into the packet header. The packet is then delivered to the MAC component, and broadcasted over the radio. When the packet is received by a neighbor, its CRC field is first checked for integrity. If this packet is sent to the current node, its port number is matched against each process that is listening to incoming packets. The thread that has a match in the port number is considered as the right thread for the incoming packet. The contents of the packet are then copied into the internal buffer that is provided by the thread, which is in turn "wakened up" to handle the incoming packet. Note that this communication stack is similar to the port-based socket model in Unix, and the

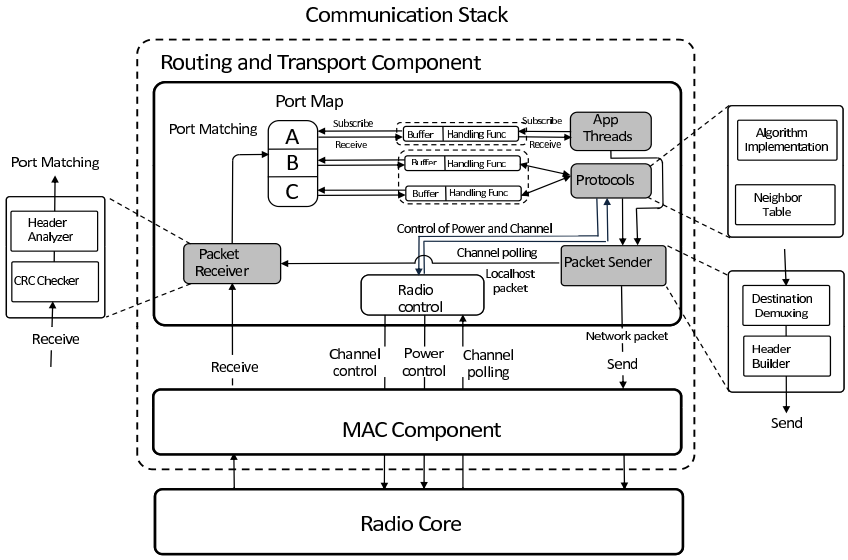

Figure 6.2: The Architecture of Communication Stack

listening thread implement a multi-hop spanning-tree based routing protocol (e.g., the **surge** example that originally distributed by TinyOS 1.x, and reimplemented in the LiteOS environment) that will continue to forward packets along the path.

Now we describe how we introduced the queueing model into this socket-like communication stack. Its implementation is closely integrated with the dynamic memory management module of the LiteOS operating system. Specifically, we exploited the free space between the end of global variables, or the **.bss** section, and the end of the growing stack, to implement a heap for memory allocation functions such as **malloc**. Whenever an incoming packet arrives, we allocate a chunk of memory whose size is the same as the size of a packet from the heap, and copy the contents of the packet to this chunk of memory. We assume that for each packet, it has been assigned a priority by the application layer. For example, a packet that contains aggregated results should have a higher priority compared to a packet that contains the initial raw data, since the former contains more condensed information. As another example, a packet that has an urgent deadline will have the highest priority

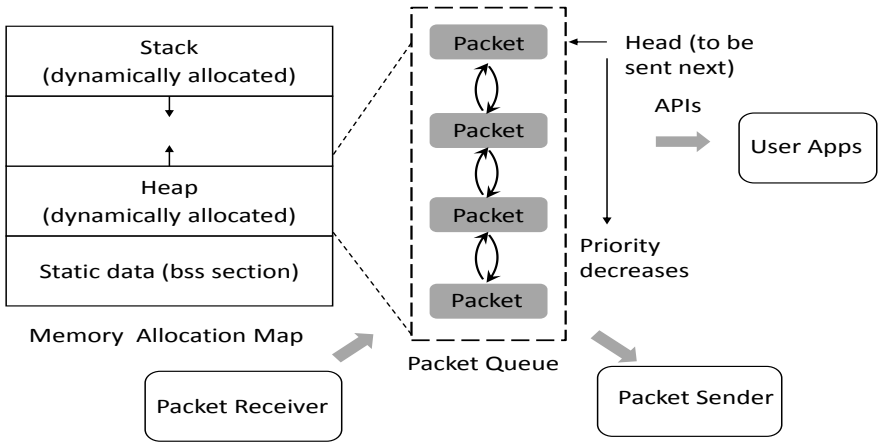

Figure 6.3: The Design of the Queueing Component

Link Prediction APIs	
setParameters	Set the link condition prediction parameters
getPRRPrediction	Get the link condition prediction for the next PRR reading
getRSSIPrediction	Get the link condition prediction for the next RSSI reading
Queue Management APIs	
getQueueLength	Get the current length of the queue
getQueueMax	Get the maximum length of the queue
Data Aggregation APIs	
getPacketFromQueue	Return a pointer to a packet in the queue
getFreePacketChunk	Return a pointer to a free slot for a new packet
releasePacketFromQueue	Release the packet as pointed by the pointer

Table 6.1: User Level APIs for Queue Management

to ensure that it gets transmitted first. Based on this priority, we order all packets in the doubly linked list into a queue, where those highest priority packets are always stored at the head of the queue. The design of the queue is shown in fig 6.3.

Observe that there are several advantages of the queue management model being implemented as a doubly linked list instead of an array. First, it allows in-place aggregation of packets. When two packets are aggregated together, we can allocate a free memory chunk to store the aggregation result, and release the earlier two packets. Then, the linked list is modified to insert the newly created packet in the

```
Thread AdaptiveSampling:
while (application is running)
{
   get predicted channel condition;
   get current queue length;
   get new sample reading;
   if (channel condition gets worse)
        sampling interval increases;
   else if (channel condition gets better)
        sampling interval decreases;
   if (queue length is more than a threshold)
      perform more aggressive packet aggregation;
   wait for sampling period;
}

Thread DataTransmission:
while (there is packet in the queue)
{
   send the highest priority packet to the next node;
}
```

Figure 6.4: The Design of the Adaptive Surge Case Study

right position. This eliminates the need to perform many copy operations to maintain the consistency of the doubly linked list. Second, by counting the total length of the queue, we can have an accurate estimate on the current congestion level. When the length of the queue grows beyond a certain threshold, the application layer can either decrease the rate of data generation, or perform aggressive data aggregation.

6.3 Application Adaptation

In this section, we describe the application layer adaptation. In particular, our design is based on the following premise: to achieve the best performance, the application layer behavior should be aware of networking layer conditions in the collection protocol of wireless sensor networks, and adjust its behavior accordingly. To achieve this goal, we develop a suite of APIs as services that include not only the link prediction, but also the queue management. In this section, we first describe our proposed API, followed by an case study to show how this API works in practice to regulate application behavior.

6.3.1 Summary of API

The APIs allow users to carry out a list of tasks such as getting the updated prediction of the link quality, reading the current size of the queue, among others. Table 6.1 shows a list of our proposed APIs for the management of the link quality and the queue as implemented on the LiteOS operating system in the form of a series of C functions.

These APIs are organized into three groups: link prediction, queue management detection, and data aggregation. The first group of APIs allows the user to set the prediction algorithm parameters, and reads the next PRR or RSSI prediction. Note that the interval between the current time and the next prediction is adjustable, depending on the user's needs. The second group of APIs allows the user to set the queue congestion level, where a maximum value is assumed to be the largest available queue size. The third group of APIs allows the user to manipulate packets in the queue, such as reading them, aggregating them, and releasing them, as needed.

6.3.2 Application Case Study

In this section, we design a modified version of the Surge example for the multi-hop transmission of packets, by taking into account current link layer realities. The skeleton of this application is shown in fig 6.4.

As shown in this figure, the modified design has two threads: the application thread that performs adaptive sampling of the sensors, and the packet processing thread that continuously transmits radio packets over to the next hop via the spanning tree. In the first thread, after each sample, the application checks if the radio condition is getting worse, or if the queue length is getting longer. In either case, the application adjusts its own behavior by performing aggressive packet aggregation, or modifying its own sampling periods. For our example, the application that transmits raw packets adopts aggregation functions, including **MAX**, **MIN**, **AVERAGE**, and **SUM** to

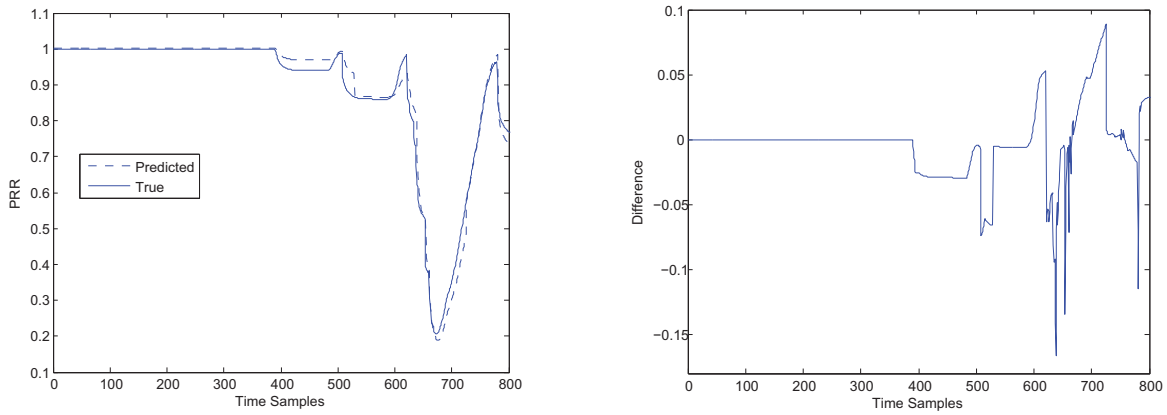

Figure 6.5: PRR Evaluations of Prediction for High-Frequency Transmission.

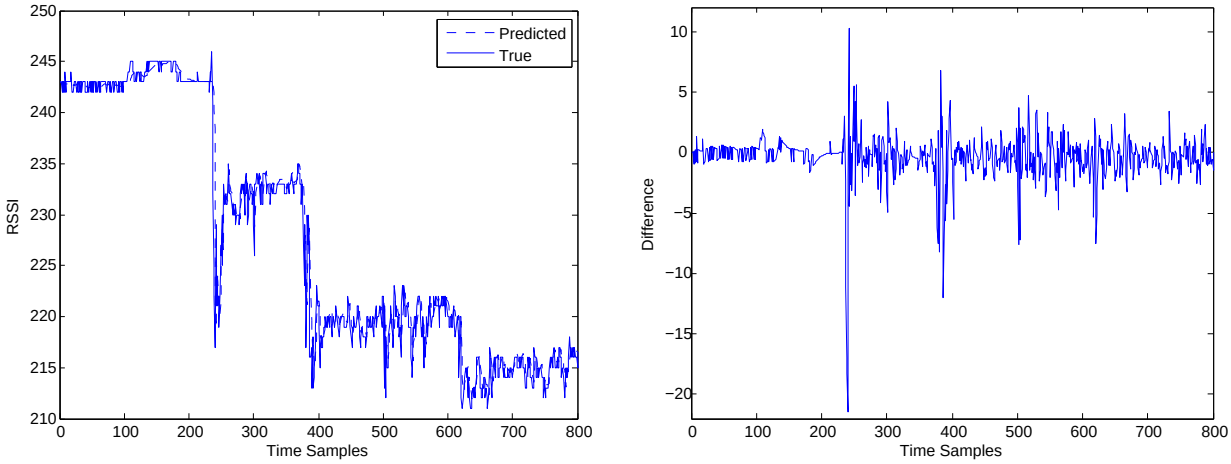

Figure 6.6: RSSI Evaluations of Prediction for High-Frequency Transmission.

profile the sensor readings and condense multiple packets into fewer, yet denser, representations.

6.4 Evaluation

In this section, we present our evaluated results of LIPS, and the results obtained by implementing the state-space model in Matlab.

Type	Parameter
MCU	Ateml ATmega128
Radio	2.4 GHz IEEE 802.15.4 [88]
Date Rate	250 kbps
Program Flash Memory	128 KB
Configuration EEPROM	4 KB

Table 6.2: Summary table of hardware.

6.4.1 Evaluation of the State Space Algorithm

To demonstrate that the state space algorithm can predict the link quality with high accuracy, we carry out the following experiments. We use two nodes, one sender and one receiver. The sender repeatedly sends out packets with sequential numbers. The receiver receives the data, and logs RSSI and sequence numbers (used to calculate PRR). The runtime hardware is MicaZ [87] mote. In Table 6.2, we summarize the primary hardware and communication parameters of our hardware.

We compare the measured metrics with the predicted results. The results are shown in fig 6.5 and fig 6.6, for high frequency communication (once per 100ms), and fig 6.7 and fig 6.8, for low frequency communication (once per 500ms). Both prediction curves and error ratios are shown in these figures. We observe that the predicted results match the real measurements very well. Indeed, the error ratio is mostly within a bound of 5%. These results demonstrate the effectiveness and the accuracy of our proposed state-space algorithm.

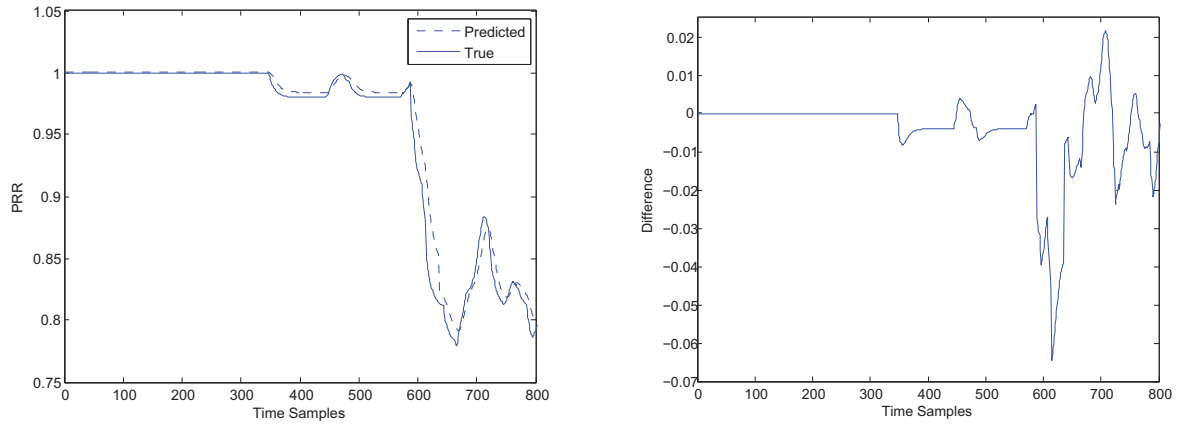

Figure 6.7: PRR Evaluations of Prediction for Low-Frequency Transmission.

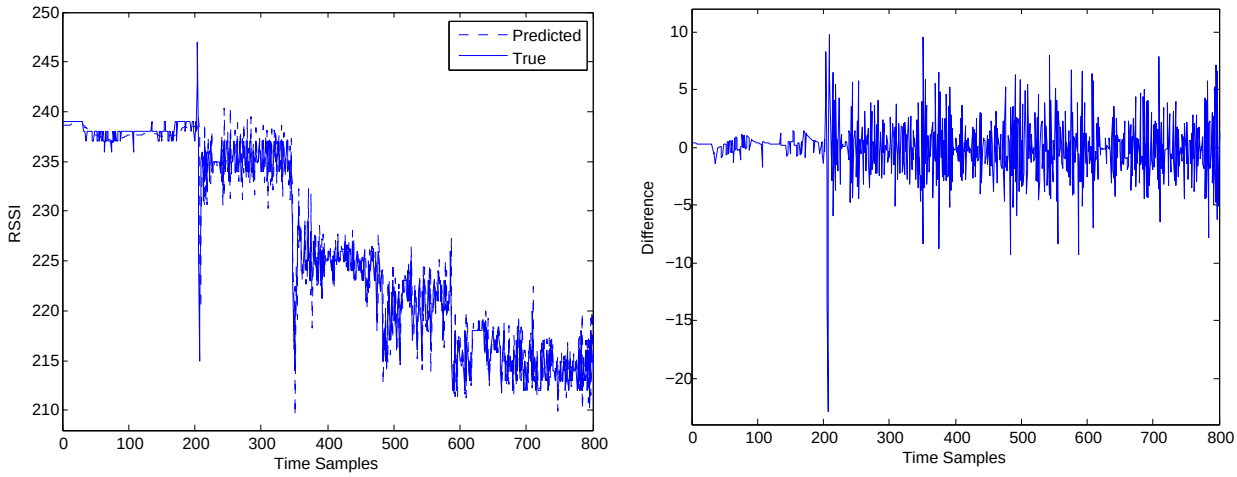

Figure 6.8: RSSI Evaluations of Prediction for Low-Frequency Transmission.

Chapter 7

Navigation in GPS-Denied Environments

The TPS can be employed as a backup to the GPS and can be quite stable in a "bad" environment. However, as mentioned previously, the fix accuracy is reduced when TPS works alone. Transmission errors (η_k^i in section 7.1.2) in the position estimation for TPS transmissions are caused by environmental factors, such as the earth's surface underlying the propagation path (e.g., water or land), as well as local variations in surface types (e.g., terrain, soil types, vegetation). Unlike errors in the GPS, such as ionospheric and tropospheric delays which have known models [90, 91], the errors in TPS are more difficult to capture and approximate by an exact model. The localization accuracy thus degrades when the TPS is used without special calibration factors (similar to those historically used in LORAN). There is a detailed description of the TPS in [89].

In this chapter, we build a dynamic model to capture the errors' stochastic characteristics, which are predictable, and thereby improving the accuracy of the TPS.

We also present an algorithm to solve pseudorange equations in both GPS and TPS. Most techniques presented in the literature have applied Newton-Raphson [90,

91], Kalman filter [92, 93, 94] or particle filter [95] methods to estimate the fixes. However, these algorithms require either the knowledge of noise statistics (Kalman filter and particle filter) or do not take noise components into consideration (Newton-Raphson). An algorithm based on the stochastic approximation is proposed here. The algorithm does not need any specific information of the noise variance but can still calculate the user's position efficiently. Moreover, the algorithm uses less computation than other methods. Parts of this work have been published in [125, 124].

7.1 Navigation Equations

In this chapter, we consider the navigation of the users on the earth's surface that is subject to environmental conditions such as urban areas, very tough terrain, or in tropical or heavily forested regions. The calculation of the distance between the user and TPS transmitters should accommodate the ground-wave propagation and great-circle path distances; this is done by adjusting the equivalent speed of the wave for the slower propagation along the earth surface; the curved-path distances may then be converted to equivalent chord distances to utilize normal rectilinear distance equations.

In the following subsections, we discuss the basic GPS pseudorange equation, the corrected great-circle distance equation, and an SA method is proposed to solve those pseudorange equations.

7.1.1 GPS Pseudorange Equation

The principle of the GPS navigation can be represented as follows [90, 91]: Each satellite is sending out signals with the following content: I am satellite X, my position is Y and this information was sent at time Z. These orbital data (ephemeris and almanac data) are stored by the GPS receiver for later calculations. For the determination of its position, the GPS receiver compares the time when the signal

was sent by the satellite with the time the signal was received. From this time difference, the distance between receiver and satellite can be calculated. If data from other satellites are taken into account, the present position can be calculated by trilateration (the determination of a distance from three points). This means that at least three satellites are required to determine the position of the GPS receiver on the earth's surface. The calculation of a position from 3 satellite signals is called a 2D-position fix (two-dimensional position determination); it is only two-dimensional because the receiver has to assume that it is located on the earth's surface. By using four or more satellites, an absolute position in a three-dimensional space can be determined. A 3D-position fix also gives the height above the earth surface as a result. The pseudorange of the ith transmitter at time k is given by the equation:

$$\rho_k^i = \rho_k^{T_i} + c(\zeta_k^i - \zeta_k^R) \tag{7.1}$$

where ρ_k^i is the pseudorange computed by the time difference between the receiver and the ith satellite and $\rho_k^{T_i}$ is the real range from the user to the ith GPS satellite at time k. The pseudorange contains two primary sources of errors. One error is introduced by the receiver's clock, which is denoted as ζ_k^R and called the receiver clock offset. This error remains the same in each pseudorange equation of each transmitter at time k. The other error is introduced in the transmission of GPS signals and denoted as ζ_k^i. This error can be modeled and approximated accurately [91], and thus is assumed known to the users. If we denote the ith satellite position by (X^i, Y^i, Z^i) relative to the center of the earth in Earth-Centered, Earth-Fixed (ECEF) coordinates, and the user's position by (X_k, Y_k, Z_k) in the same coordinates, then the distance between the ith satellite and the user can be written as the nonlinear expression:

$$\rho_k^{T_i} = \sqrt{(X_k - X^i)^2 + (Y_k - Y^i)^2 + (Z_k - Z^i)^2} \tag{7.2}$$

To solve user positions and the receiver clock offset, 4 satellites are needed to solve for $(X_k, Y_k, Z_k, \zeta_k^R)$ sufficiently.

7.1.2 TPS Great-Circle Distance

As mentioned above, the multilateration radiolocation algorithms for the TPS are generally similar to those used in the GPS except for the addition of great-circle corrections to accurately represent the lengths of the ground-wave propagation paths on the nearly spherical earth and (obviously) the deletion of the satellite almanac and ephemeris data. In most operational scenarios, TPS transmitters will be locked to the GPS time with very high-quality clocks. In addition, their locations will be pre-surveyed and known to fractions of a meter. TPS data streams will thus provide all the information needed by the receiver (except for onboard-stored local propagation-correction tables) to accurately compute its position. Due to the finite conductivity of the earth's surface, and local variations due to surface types (i.e., land or water), soil, moisture content, temperature, and (to a lesser extent) seasons, the average signal velocity must be reduced by very roughly 0.15%. In addition, the curved path on the earth's surface requires generic great-circle distance computations. As shown in fig 7.1, the true range transmitted is along the spherical earth instead of the chord between A (the user) and T (the transmitter) and should be estimated by the great-circle distance.

The TPS ground wave follows the great-circle distance between two points on the earth's surface (assumed spherical), which can be computed by the following formula, where δ^i and φ^i are latitude and longitude, respectively and r is the radius of the earth (approximately 6371 km on average), then the great-circle distance d is approximately:

$$\begin{aligned} d(\delta^1, \varphi^1, \delta^2, \varphi^2) = r \cos^{-1}[&\sin\delta^1 \sin\delta^2 \\ &+ \cos\delta^1 \cos\delta^2 \cos(\varphi^1 - \varphi^2)] \end{aligned} \tag{7.3}$$

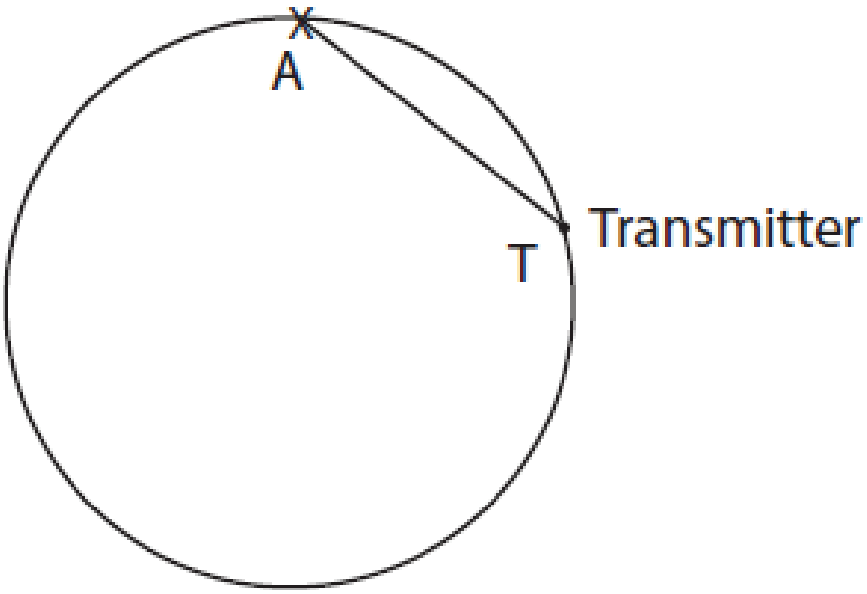

Figure 7.1: The users near the earth surface.

The great-circle distance equation is employed to calculate the distance of a near-spherical earth path between the user and land-based TPS transmitters. In this chapter, we consider only the navigation of the users near the surface, which means the height between the user and the earth surface is zero. For users at varying heights, the distances between the users and TPS transmitters do not quite follow the great-circle equations and should be calculated by taking the heights of the users into account.

Now assume there are M TPS transmitters. Then, the pseudorange equation at time k for the TPS can be written similarly as that of the GPS as follows:

$$d_k^i = d_k^{T_i} + c_T(\eta_k^i - \eta_k^R) \tag{7.4}$$

where d_k^i is the pseudorange between the user and the ith TPS transmitter, and $d_k^{T_i}$ is the true range between the user and the ith $(1 \leq i \leq M)$ TPS transmitter, which is approximated by the great-circle equation given above. η_k^i is the transmission error generated in the transmission of the TPS signal by the environment around the surface

and is what we need to model. η_R^k is the receiver clock offset, equivalent to ζ_k^R in the GPS pseudorange equation. η_k^i is unfortunately difficult to model accurately due to its characteristic irregularity. When GPS signals are available, we can calculate η_k^i, but when GPS signals are absent, we do not have enough information to do so. This motivates the model proposed here.

7.2 Stochastic Approximation Method

To solve pseudorange equations explicitly, numerous algorithms have already been proposed in the literature such as the Kalman Filter, Newton-Raphson method, particle filter, and the likes [90, 91, 92, 93, 94, 95]. However, most of them require either the variance of the noise (Kalman Filter and particle filter) is known, or do not consider the effect of the noise (e.g. Newton-Raphson). In this section, a stochastic approximation algorithm in [96] is employed to compute the fixes explicitly. This method trains the Kalman gain matrix to its correct, steady-state form, when plant noise and observation noise covariance matrices are unknown. Following [96] the SA algorithm is discussed next. Consider the discrete-time system [97]:

$$x_{k+1} = Ax_k + w_k \tag{7.5}$$

$$y_k = d(x_k) + v_k \tag{7.6}$$

where x_k is the state vector containing the longitude, latitude (or X, Y, Z fixes in ECEF coordinates) of the user at time k, velocities, and clock offsets; A is the corresponding system matrix; $d(x_k)$ is the pseudorange vector containing all pseudoranges (as in (7.1) and (7.4)) sampled for each TPS transmitter and is a function of x_k; and w_k and v_k are noise terms representing uncertainties of the system with variance W and V.

It is well known from Kalman filtering theory that the a posterior estimate of x_k is given by:

$$\hat{x}_{k+1} = A\hat{x}_k + K_{k+1}(y_{k+1} - d(A\hat{x}_k)) \tag{7.7}$$

where $K_{k+1} = P_{k+1|k}D_{k+1}^T(D_{k+1}P_{k+1|k}D_{k+1}^T+V)^{-1}$ is the Kalman gain, where $P_{k+1|k} = A_{k+1}(I - K_kD_k)P_{k|k-1}A_{k+1}^T + W$ and $D_{k+1} = \frac{\partial d}{\partial x}|_{A\hat{x}_{k+1|k}}$

Instead of the traditional Kalman gain, the stochastic approximation procedure provides a recursive gain adaptation algorithm in the following form:

$$K_{k+1} = K_k + \mu_k\theta(K_k) \tag{7.8}$$

where μ_k is a decreasing sequence of real numbers and $\theta(K_k)$ is an unspecified stochastic vector that depends on K_k. One choice for $\theta(K_k)$ is $\theta(K_k) = A\hat{x}_{k+1}{v_{k+1}}^T$, and under certain conditions on μ_k in [96], K_{k+1} converges to the optimal Kalman gain.

The advantages of this SA algorithm over other algorithms are summarized as follows:

- It does not assume knowledge of noise covariance matrices;
- The computation of its Kalman gain does not require the calculation of the estimation covariance, which can reduce the computation cost significantly over that of the Kalman filter.
- Unlike Newton-Raphson, which needs N equations to solve for N unknowns, SA can estimate x_k accurately with a number of measurements smaller than the number of variables contained in the state x_k (partially observed).

7.3 State Space Model

In this section, we model the errors produced during the transmission of the TPS via a dynamic state-space model which is based on SDEs. SDEs have been widely used to

model control systems and communication channels. For example, a mobile-to-mobile communication channels can be modeled as [101]:

$$\begin{aligned} x_{k+1} &= F_k x_k + G_k w_k \\ y_k &= H_k x_k + N_k v_k \end{aligned} \tag{7.9}$$

where x_k is the state; y_k is the measurement sampled at the output of the channel; w_k and v_k are noises; and F_k, G_k, H_k and N_k are parameters of the channel.

Other related examples can be found in [102, 103, 104, 105], where great flexibility and utility offered by state-space models are employed extensively in a number of different areas of statistics. For example, the Bureau of Labor Statistics (BLS) in the U.S. used state-space models for the estimation of all the monthly employment and unemployment estimates for the 50 states and the District of Columbia. There, state-space models were fitted independently between the internal states and used to build a model of the true population values with an accompanying model for the sampling errors [102]. The authors in [103] used the state-space approach to model, estimate, and predict short-term electric power consumption, which can provide an insight into the future power demand and thus make decisions on the power generation, e.g., whether to activate reserve power generators or decrease generator outputs. [104] generalized the linear discrete-time state-space model from a single-dimensional time to two-dimensional space and then used it as a model for the linear image processing. Further, [105] developed a non-Gaussian state-space model for the censored data. These applications suggest its successful use in modeling and predicting TPS transmission errors.

The time-varying property of parameters in (7.9) adapts dynamically to the variety of states. The noises w_k and v_k can also capture the range uncertainties introduced during the transmission. Due to these special characteristics, we propose to use the state-space model to track, estimate and predict transmission errors (η_k^i) in TPS

transmissions. Consider the following time-invariant state-space predictor:

$$\begin{aligned} x_{k+1} &= Fx_k + Ge_k \\ \eta_k &= Hx_k + e_k \end{aligned} \tag{7.10}$$

where η_k is the transmission error computed for each transmitter, and $e_k = \eta_k - Hx_k$ is the error between the measured error and the error predicted by the model. To estimate F, G, H, and x_k, the PEM method [107] is employed. PEM estimates the parameters by minimizing a least-square cost function:

$$V_N = \sum_{k=1}^{N} e_k^T e_k \tag{7.11}$$

The details of the algorithm can be found in [107].

At each discrete-time instant k, the PEM algorithm estimates the parameters F, G and H recursively from measurements at k. With the estimated parameters and states of the model, the one step-ahead prediction of the transmission error can be computed by:

$$\hat{\eta}_{k+1} = \hat{H}_k(\hat{F}_k\hat{x}_k + \hat{G}_k(\eta_k - \hat{H}_k\hat{x}_k)) \tag{7.12}$$

where $\hat{\eta}_{k+1}$ denotes the predicted transmission error at $k+1$; $\hat{F}_k, \hat{G}_k$, $\hat{H}_k$ and $\hat{x}_k$ are the parameters and state estimated by PEM at time k; $\eta_k - \hat{H}_k\hat{x}_k$ is the prediction error at k; $\hat{F}_k\hat{x}_k + \hat{G}_k(\eta_k - \hat{H}_k\hat{x}_k)$ is from the state evolution equation in (7.10). The p-steps-ahead ($p > 1$) prediction of the transmission error can be computed by:

$$\hat{\eta}_{k+p} = \hat{H}_k \times \hat{F}_k^{p-1}(\hat{F}_k\hat{x}_k + \hat{G}_k(\eta_k - \hat{H}_k\hat{x}_k)) \tag{7.13}$$

where $\hat{F}_k^{p-1}(\hat{F}_k\hat{x}_k + \hat{G}_k(\eta_k - \hat{H}_k\hat{x}_k))$ computes the predicted state after p steps from k.

The state-space model introduced in this section employs the PEM algorithm which numerically estimates model parameters, however, such procedure may have a

high computational cost. In the next section, the AR model which is a special case of the state-space model is proposed. Parameters in the AR model can be estimated explicitly without involving numerical calculations.

7.4 Autoregressive (AR) Model

An AR model is a type of random process which is often used to model and predict various types of natural phenomena [108]. For instance, [110] used an infinite AR process to model stationary multi-dimensional multivariate time series. In [111], an AR model is used to represent shapes of boundaries detected in digitize binary images of objects. In [112], the authors employed a 2-D piecewise AR model in the image interpolation. [113] proposed a heterogenous AR model to describe the behavior of the volatility inherent in financial time series. These works motivate the application of the AR model in estimating TPS transmission errors. A typical AR process can be written as:

$$X_k = \sum_{i=1}^{p} \alpha_i X_{k-i} + \varepsilon_k \tag{7.14}$$

where X_k is a time series, $\alpha_1, ..., \alpha_p$ are parameters of the model, and ε_k represents white noise at time k. p is the order of the AR model. If we use X_k to represent the transmission error η_k^i, then future errors can be predicted through (7.14). Before the prediction, parameters in (7.14) need to be estimated based on past errors computed from the available GPS data by minimizing the following quadratic cost function:

$$V_N = \frac{1}{N} \sum_{k=1}^{N} [X_k - \sum_{i=1}^{p} \alpha_i X_{k-i}]^2 \tag{7.15}$$

where N denotes the number of data available to estimate the parameters. For convenience, let $\theta = (\alpha_1, .., \alpha_p)^T$ and $\phi_k = (X_{k-1}, ..., X_{k-p})^T$. By minimizing the cost function V_N with respect to θ, the least square estimate of $\hat{\theta}_k$ can be calculated as

follows [114]:

$$\begin{aligned} \hat{\theta}_k &= \hat{\theta}_{k-1} + L_k[X_k - \hat{\theta}_{k-1}^T \Phi_k] \\ L_k &= \frac{M_{k-1}\Phi_k}{1 + \Phi_k^T M_{k-1}\Phi_k} \\ M_k &= M_{k-1} - \frac{M_{k-1}\Phi_k\Phi_k^T M_{k-1}}{1 + \Phi_k^T M_{k-1}\Phi_k} \end{aligned} \tag{7.16}$$

Remark 7. *The AR model can be viewed as a special case of the state-space model (actually, the state-space model can be transform to an autoregressive moving average (ARMA) process). The difference between them resides in the fact that the noise term in the state-space model is more complex, and thus requires a more complicated parameter estimation algorithm. Parameters in the AR model can be computed explicitly through (7.16) which is much simpler. However, the state-space model can provide more accurate results at the expense of more computational cost.*

7.5 Generalized Linear Model (GLM)

The proposed methods such as AR and state-space are actually linear models, in this section, we introduce an nonlinear model approach called generalized linear model for modeling purposes in the TPS transmission. As its name stands, it generalizes linear regression by allowing the linear model to be related to the response variable via a link function and by allowing the magnitude of the variance of each measurement to be a function of its predicted value [118].

In a GLM, the responses Y are assumed to be generated from a particular distribution in the exponential family, including normal, binomial and poisson distributions, etc.. The mean, μ, of the distribution, depends on a function of the linear combination of independent variables X through the following expression [118]:

$$EY = \mu = g^{-1}(X\beta) \tag{7.17}$$

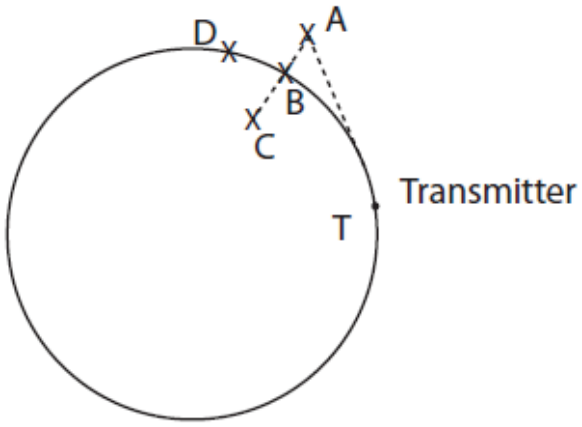

Figure 7.2: The users with varying heights.

where $X\beta$ is called the linear predictor, a linear combination of unknown parameters β; g is the link function. The unknown parameters, β, are typically estimated by maximum likelihood, maximum quasi-likelihood, or Bayesian techniques.

The GLM consists of three elements, including a probability distribution to describe the responses Y, a linear combination of X, $z = X\beta$, and a link function g such that $z = g(\mu)$. In this work, we consider the transmission errors η_k^i in the TPS transmission as responses Y, and stipulate X to be independent variables, i.e., transmission errors from 1 to k are denoted by Y, while X can be chosen as the standard Brownian motion $\{B_k\}_{k\geq 0}$, which is one of the most popular and fundamental stochastic processes. The relationship can then be set up based on X, Y, and β can be estimated. The estimated β are further used for the prediction of transmission errors when the TPS works alone. More details of the GLM can be retrieved from [118].

7.6 TPS Navigation Scheme Algorithm (NSA)

In this section, a navigation scheme for improving accuracy is introduced.

Assume both GPS and TPS data are available from time 1 to time n; also, from time n on, only the TPS data is accessable. The procedure to navigate only with the TPS at time $n+1$ is as follows:

1) Compute the user's position (X_k, Y_k, Z_k) and ζ_k^R for each $k = 1, ..., n$ using (7.1) and (7.2) through the SA algorithm. Then convert (X_k, Y_k, Z_k) to latitude and longitude. As the user is assumed to be near the earth's surface, the height of the user H is estimated as 0.

2) Plug the latitude and the longitude back into (7.3) to obtain $d_k^{T_i}$ for each $k = 1, ..., n$ and each transmitter.

3) As the receiver clock offset is constant in one single time slot, we can assume that $\eta_k^R = \zeta_k^R$. Then η_k^i can be computed in (7.4) for each k and each i (note d_k^i is measured by the TPS).

4) Build a statistical model for each $\{\eta_k^i\}_{k=1}^{k=n}$ and predict η_{n+1}^i for each transmitter using the models introduced in section 7.3, 7.4 and 7.5.

5) Plug η_{n+1}^i back into (7.4) and obtain the measurement equation. Together with the system equation (7.5), the state x_{n+1} can now be estimated accurately by the SA algorithm.

After the time $n+1$, continue to compute the user positions with the previous algorithm when the GPS signal is lost. Once the GPS becomes available again, update the model with new η_k^i values computed from the latest GPS measurement.

Remark 8. *The groundwave signal passes through a variety of different environments, which indicates that a fixed model can not capture the characteristics of the stochastic randomness during the transmission. In different environments, e.g. canyon, forest, etc., transmission errors introduced into the system are distinct greatly, while these models are well suited to such situation as it can be updated with new measurements. For example, once the signal transmission surrounding is changed, these models can be improved with new incoming GPS measurements for the new surrounding.*

Remark 9. *For the users with varying heights ($H \neq 0$), the distance equation to compute the true range should be updated. However, it is difficult to determine a unique equation in this case, as the user may be below the average earth altitude (e.g., a canyon) or on a hill, where the equations are different (see fig 7.2). In a practical situation, this range may be approximated by the great-circle equation. However, we maintain that once the distance equation is altered for varying heights, the navigation scheme proposed in this chapter is still applicable to the new distance equation.*

7.7 Numerical Example

In this section, we present an example to illustrate the performance of the navigation algorithm proposed in this chapter.

Assume N=3 TPS transmitters are located with latitude and longitude pairs: $(38.3127491°, 115.6442846°)$, $(39.2763475°, 116.0855268°)$, $(37.6413982°, 114.3172851°)$. The initial position of the user in ECEF coordinates is $(-2.172 \times 10^6,\ 4.390 \times 10^6,\ 4.074 \times 10^6)$.

The user is assumed to move along the earth's surface randomly. Thus, for convenience but without loss of generality, the distance equation can be written as (7.5) and (7.6), where $A = I_{4\times4}$, where $I_{4\times4}$ denotes the 4×4 identity matrix, the state vector $x_k^T = [\delta_k, \varphi_k, \eta_k^R]^T$ for the TPS and $x_k^T = [X_k, Y_k, Z_k, \zeta_k^R]^T$ for the GPS.

From time 1 to 50, when both GPS and TPS data are available, $\{x_k^G\}_{k=1}^{50}$ are computed by the SA algorithm and then $\{\eta_k^i\}_{k=1}^{50}$ can be obtained by following the NSA described in section 7.6. From times 51 to 150, the GPS signal is lost and only the TPS is available. A scalar state-space model is employed to model $\{\eta_k^i\}_{k=1}^{k=50}$, and then $\{\eta_k^i\}_{k=51}^{k=150}$ are predicted by this model using the algorithm proposed in section 7.3. Next, user positions are estimated by the SA algorithm from time 51 to 150. The differences between real fixes and estimated fixes of all coordinates are presented in fig 7.3 (shown in ECEF coordinates for the sake of comparison). It is obvious that the positions estimated by the proposed navigation scheme are close to the real ones since

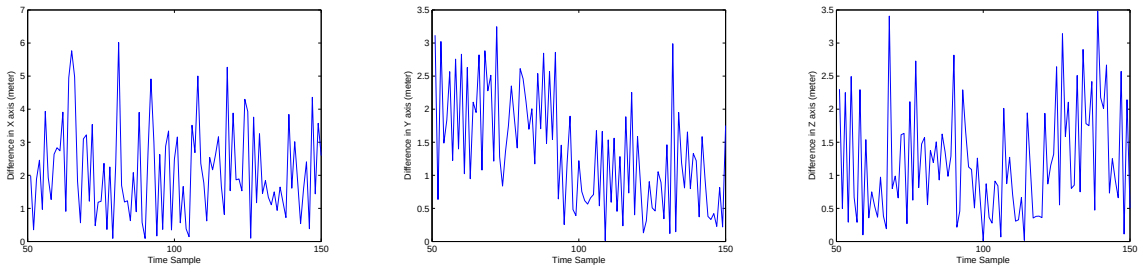

Figure 7.3: Position estimation errors in ECEF coordinates using state-space model.

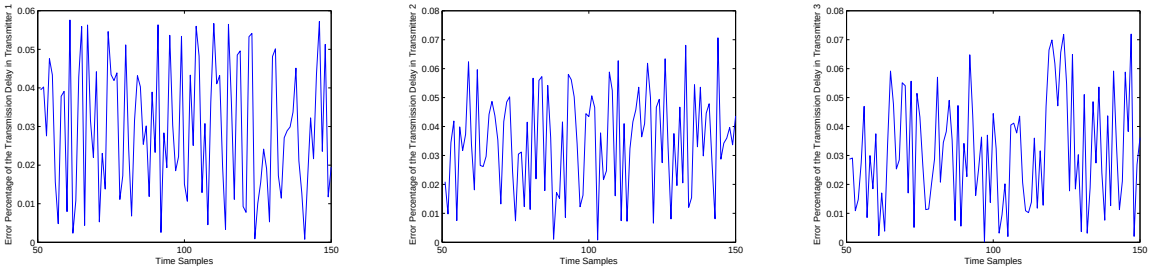

Figure 7.4: Error percentage of $\{\eta_i^k, i = 1, 2, 3\}$ predicted by the state-space model.

the differences between the estimated and the true fixes are small. Deviations from true positions are bounded by 6m, 3.5m and 3.5m on each axis, respectively. The percentage of the error between actual $\{\eta_k^i\}$ and predicted ones $\{\hat{\eta}_k^i\}$ $(|\eta_k^i - \hat{\eta}_k^i|/\eta_k^i)$ by the state-space model are also plotted in fig 7.4. These plots demonstrate that the proposed state-space model can predict transmission errors with small differences.

Similarly, an AR model with order 6 and a GLM are used in the navigation scheme to run the same simulation. In GLM, the independent variables X corresponding to the measurement y_k are generated by the normal distribution $N(0, k)$. The simulation results are shown in fig 7.5, fig 7.6, fig 7.7 and fig 7.8. Although the navigation schemes based on the AR model and GLM both generate close estimation fixes, the scheme based on the state-space model clearly offers a better localization performance for this data set. Usually the state-space method performs better than the AR model because the AR is a special case of the state-space model. However, it is difficult to determine which one is better between the state-space model and the GLM theoretically.

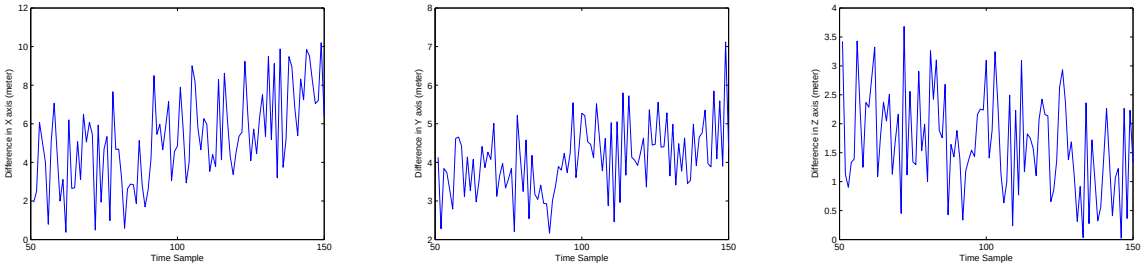

Figure 7.5: Position estimation errors in ECEF coordinates by the AR process.

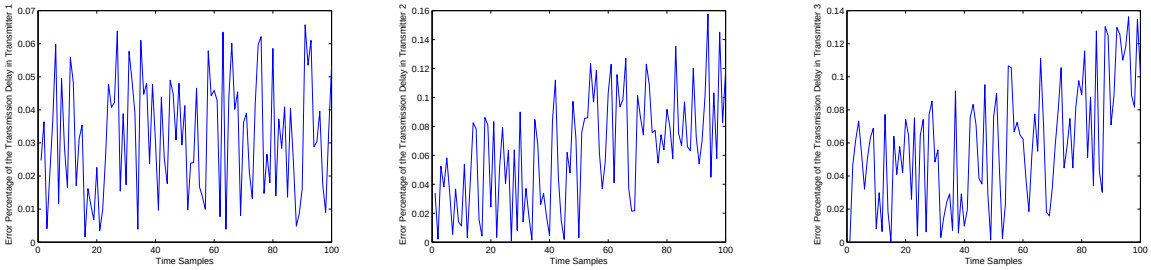

Figure 7.6: Error percentage of $\{\eta_i^k, i = 1, 2, 3\}$ predicted by the AR process.

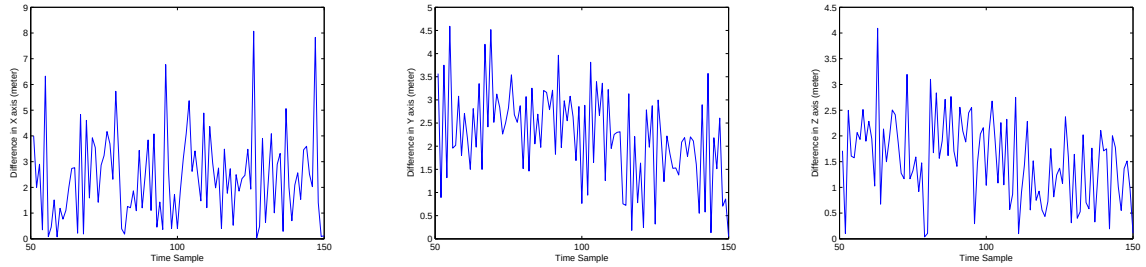

Figure 7.7: Position estimation errors in ECEF coordinates by the GLM.

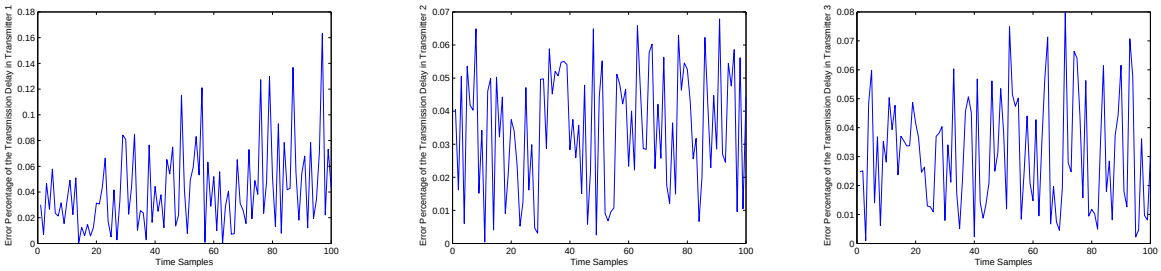

Figure 7.8: Error percentage of $\{\eta_i^k, i = 1, 2, 3\}$ predicted by the GLM.

Chapter 8

Conclusions and Future Work

This study discussed control and estimation algorithms design over two communication systems: Bernoulli packet loss links and CR systems. The problems considered can be formulated in general as follows:

$$\begin{aligned} x_{k+1} &= Ax_k + \nu_k Bu_k + \upsilon_k \\ y_k &= \gamma_k Cx_k + \omega_k \end{aligned} \tag{8.1}$$

where γ_k and ν_k represent packet loss indicators.

We first designed the optimal filter over the Bernoulli packet loss link. The arrival information was assumed to be unknown. The nonlinear filter, which is the optimal one, was derived here using the exact hybrid filter by taking Bernoulli i.i.d process as a Markov process and compared with the linear optimal estimator, which assumed the state estimate was a linear combination of the measurement. The simulation results compared the nonlinear optimal filter and the linear optimal estimator and also showed that the performance of the optimal filter was better than the linear optimal estimator, especially for unstable systems.

Second, estimation and control via CR systems modeled by the two-switch model were considered. This new communication link introduces packet losses during the transmission due to activities of primary users. Estimator and controller design of

a closed-loop system over CR links were addressed. The separation principle was shown not to hold and the controller was a nonlinear function of the state estimate. Several stability conditions were discussed and numerical examples were presented to illustrate the method developed.

Third, we studied the state estimation over the CR system governed by semi-Markov processes. Two cases were considered in this work. The first assumed packet acknowledgement at the estimator, while the other did not. Sufficient stability conditions were derived for the peak covariance process of the optimal filter of the first case, and a suboptimal estimator was proposed for the second case. Illustrative examples were provided to demonstrate the method's viability. The optimal controller was also derived for the case with only one semi-Markov channel in the CR system, but it was untractable due to coupled Riccati equations. We then designed a suboptimal but tractable controller based on LMIs.

Future work will focus on the suboptimal solution other than the optimal filter for the Bernoulli packet loss problem, e.g. using an exponentially weighted loss function. Moreover, the distributed estimator and controller design over CR systems and the application of them in specified systems will be examined. Also, other alternative suboptimal control algorithms may be developed from the optimal solution, and the performance of the suboptimal solutions will be compared. Additionally, since semi-Markov processes do not satisfy the semi-group property, it is a good point to discuss the uniqueness of the solution.

Another topic addressed is the link quality prediction of WSNs. This study presented the design and evaluation of an integrated system architecture for providing the link-layer quality estimation as a service to upper-layer applications. The contribution of this work is three-fold. First, we present a novel, state-space driven prediction method for the link quality. Our evaluation results in Matlab showed that this method can predict the future link quality with high accuracy. Second, we presented a queue management model that integrates the dynamic memory with a doubly-linked list for implementing packet queues. Our evaluation results on the

LiteOS platform showed that this was feasible even in extremely resource constrained environments. Finally, we presented an application case study where we modified the well-known surge example with consideration for changes in link conditions. The processes we used and presented here can be beneficial for future designs of similar sensor network applications. Future work will implement the proposed model in the real platform and compare the results of the state-space model with previous AR models.

The last topic in this dissertation is navigation with TPS, which is largely intended to be used as a backup in GPS-denied environments. We considered the user moving along the earth's surface and employed three models–a state-space model, an AR process, and a GLM–to predict the error generated by environmental delays in its transmission, thus improving the estimation accuracy of TPS fixes. We have also proposed a stochastic approximation algorithm to solve the pseudorange equations. A navigation scheme is then provided and illustrated by an example. Future work will focus on more complex nonlinear models to further improve the accuracy.

Bibliography

[1] J. Nilsson (1998), *Real-time control systems with delays*. Ph.D. dissertation, Department of Automatic Control, Lund Institute of Technology. 8, 9

[2] K.J. Rajasekaran, N. Satyanarayana and M.D. Srinath (1971), "Optimum linear estimation of stochastic signals in the presence of multiplicative noise," *IEEE Trans. Aerospace and Electronic Systems*, vol. AES-7, pp. 462 – 468. 8

[3] M. Hadidi, and S.Schwartz (1979)," Linear recursive state estimators under uncertain observations," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 944 – 948. 8, 53

[4] B. Azimi-Sadjadi (2003), "Stability of networked control systems in the presence of packet losses," *42nd IEEE CDC*, 676 – 681. 8, 38

[5] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poola, M. Jordan and S. Satry (2004), "Kalman filtering with intermittent observations," *IEEE Trans. Automat. Contr.*, vol. 49(9), pp. 458 – 463. 8, 58

[6] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poola and S. Satry (2007), "Foundations of control and estimation over lossy networks," *Proc. of the IEEE*, vol. 95, pp. 163 – 187. 8, 47

[7] M. Huang and S. Dey (2007), "Stability of Kalman filtering with Markovian packet losses," *Automatica*, vol. 43, pp. 598 – 607. 8, 48, 59, 61, 62, 68, 146, 147

[8] X. Li, and L. Xie (2007), "Peak covariance stability of a random riccati equation arising from Kalman filtering with observation losses," *Journal System Science & Complexity*, vol. 20, pp. 262 − 272. 66, 67

[9] V. Gupta, B. Hassbi and R. M. Murray (2007), "Optimal LQG control across packet-dropping links," *Systems and Control Letters*, vol. 56, pp. 439 − 446. 8, 53

[10] L. Xiao, A. Hassbi and J. P. How (2000), "Control with random Communication delays via a discrete-time jump system approach," *ACC*, Chicago, Illinois, pp. 2199 − 2204. 8

[11] L. Xiao, M. Jonathan, H. Hindi, S. Boyd, and A. Goldsmith (2003), "Joint optimization of communication rates and linear systems," *IEEE Trans. Automat. Contr.*, vol. 48(1), pp. 148 − 153. 8, 9

[12] N. Nahi(1969), "Optimal recursive estimation with uncertain observation," *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 457 − 462. 8, 14, 16

[13] A. Jaffer, and S. Gupta (1971), "Recursive Bayesian estimation with uncertain observation," *IEEE Trans. Inform. Theory*, vol. 17, pp. 614 − 616. 9, 23

[14] R. J. Elliott, F. Dufour, and D. D. Sworder (1996), "Exact hybrid filters in discrete time," *IEEE Transactions on Automatic Control*, vol. 41, no. 12, pp. 1807 − 1810. 1, 9, 16, 18, 19, 20

[15] N. J. Gordon, and V. Krishnamurthy (2001), "Particle filters for state estimation of jump Markov linear systems," *IEEE Trans. Signal Processing*, vol. 49(3), pp. 613 − 624. 2, 9

[16] A. Doucet, and C. Andrieu (2001), "Iterative algorithms for state estimation of jump Markov linear systems," *IEEE Trans. Signal Processing*, vol. 49(6), pp. 1216 − 1227. 9

[17] O.L.V. Costa (1994), "Linear minimum mean square error estimation for discrete-time Markovian jump linear systems ," *IEEE Trans. Auto. Control*, vol. 39(8), pp. 1685 − 1689. 9

[18] O.L.V. Costa, MD. Fragoso, and R.P. Marques (2005), *Discrete-time Markov jump linear systems*, Springer.

[19] A. Germani, C. Manes, and P. Palumbo (2009), "State estimation of stochastic systems with switching measurements: A polynomial approach," *International Journal of Robust and Nonlinear Control*, vol. 19(14), pp. 1632 − 1655. 9

[20] S. Nakamori, R. Caballero-Aguila, A.H. Carazo, and J.L. Perez (2003), "Second-order polynomial estimators from uncertain observations using covariance information," *Apllied Mathematics and Computation*, vol. 143(2-3), pp. 319 − 338.

[21] M. Askar, and H. Derin (1984), "Recursive algorithms for Bayes smoothing with uncertain observations," *IEEE Trans. Auto. Control*, vol. 29(5), pp. 459 − 461. 2

[22] L. Aggoun, and R.J. Elliott (2004), "Measure theory and filtering: Introduction with applications," *Cambridge University Press*, pp. 169 − 177. 19, 22

[23] R. J. Elliott, F. Dufour, W. P. Malcolm (2005), "State and mode estimation for discrete time jump Markov systems," *SIAM J. Control and Optimization*, vol. 44, No. 3, pp. 1081 − 1104. 22

[24] N.J. Gordon, D.J. Salmond, and A.F.M. Smith (1993), "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proc.-F*, vol. 140, no. 2, pp. 107 − 113. 9

[25] S.A. Jafar and S. Srinivasa (2007), "Capacity limits of cognitive radio with distributed and dynamic spectral activity," *IEEE JSAC*, vol. 25, pp. 529 − 537.

[26] W.L. De Koning (1984), "Optimal estimation of linear discrete-time systems with stochastic parameters," *Automatica*, vol. 20(1), pp. 113 − 115. 24, 34, 45, 47

[27] D. Simon (2006), *Optimal state estimation: Kalman, H-infinity, and nonlinear approaches*, John Wiley Sons, Inc., Hoboken, New Jersey. 34

[28] D.P. Bertsekas (1995), *Dynamic programming and optimal control*, volume 1. Athena Scientific, Belmont, MA, USA.

[29] B.A. Sadjadi (2003), "Stability of network control systems in the presence of packet losses," *Proceedings of IEEE Conference on Decision and Control*, pp. 676 − 681. 39

[30] O.C. Imer, S. Yuksel, and T. Basar (2004), "Optimal control of dynamical systems over unreliable communication links," in *NOLCOS*, Stutgart, Germany, 1429 − 1439. 47

[31] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, S. Sastry (2008), "Optimal linear LQG control over lossy networks without packet acknowledgment," *Asian Journal of Control*, vol. 10, pp. 3 − 13.

[32] S. Srinivasa, and S. A. Jafar (2007), "The throughput potential of cognitive radio: A theoretical perspective," *IEEE Communication Magazine*, pp. 73 − 79. xiv, 2, 3, 9, 27, 28, 29, 30

[33] NTIA (2003), "FCC frequency allocation chart," http://www.ntia.doc.gov/osmhome/allochrt.pdf. 2, 27

[34] FCC Spectrum Policy Task Force (2002), "Report of the spectrum efficiency working group," *Tech. rep.02-135*, http://www.fcc.gov/sptf/files/SEWGFinalReport 1.pdf 2, 27

[35] J. Mitola (2000), "Cognitive radio: An integrated agent architecture for software defined radio," *Doctor of Technology, Royal Inst. Technol. (KTH)*, Stockholm, Sweden. 2, 27

[36] I.F. Akyildiz, W.Y. Lee, M.C. Vuran and S. Mohanty (2006), "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," Computer Networks 50, pp. 2127 − 2159. 2

[37] X. Ma, S.M. Djouadi, T. Kuruganti, J.J. Nutaro, and H. Li (2009), "Optimal estimation over unreliable communication links with application to cognitive radio," *48th IEEE CDC and 28th CCC*, pp. 4062 − 4067. 9, 28, 54

[38] X. Ma, S.M. Djouadi, T. Kuruganti, J.J. Nutaro, and H. Li (2010), "Control and estimation through cognitive radio with distributed and dynamic spectral activity," *ACC*, pp. 289 − 294. 9, 28, 54

[39] S. Geirhofer, L. Tong abd B.M. Sadler (2007), "Dynamic spectrum access in the time domain: Modeling and exploiting whitespace," *IEEE Communications Magazine*, pp. 66 − 72. 3, 13, 53, 54

[40] Y. Shi, Y.T. Hou, and H. Zhou (2009), "Per-node based optimal power control for multi-hop cognitive radio networks," *IEEE Trans. Wireless Communication*, vol. 8, no. 10, pp. 5290 − 5299.

[41] Y. Chen, G. Yu, Z. Zhang, H. Chen and P. Qiu (2008), "On cognitive radio networks with opportunistic power control strategies in fading channels," *IEEE Trans. Wireless Communication*, vol. 7, no. 7, pp. 2752 − 2761.

[42] X. Zhou, J. Ma, G.Y. Li, Y.H. Kwon, and A.C.K. Soong (2009), "Probability-based optimization of inter-sensing duration and power control in cognitive radio," *IEEE Trans. Wireless Communication*, vol. 8, no. 10, pp. 4922 − 4927.

[43] A.T. Hoang, and Y.C. Liang (2008), "Downlink channel assignment and power control for cognitive radio networks," *IEEE Trans. Wireless Communication*, vol. 7, no. 8, pp. 3106 − 3117.

[44] M.M. Olama, S.M. Djouadi, and C.D. Charalambous (2009), "Stochastic differential equations for modeling, estimation and identification of mobile-to-mobile communication channels," *IEEE Transactions on Wireless Communications*, vol. 8, no. 4, pp. 1754 − 1763.

[45] S.M. Djouadi, M.M. Olama and Y. Li (2008), "Optimal approximation of the impulse response of wireless channels by stochastic differential equations," *IEEE Signal Processing Letters*, vol. 15, pp. 896 − 899. 93

[46] C.D. Charalambous, R.J.C Bultitude, J. Zhang, and Xin Li (2008), "Modeling wireless fading channels via stochastic differential equations: identification and estimation based on measurements," *IEEE Transactions on Wireless Communications*, Vol. 7, No.2, pp.434 − 439. 93

[47] C.D. Charalambous and S. Djouadi and D. Stojan (2005), "Stochastic power control for wireless networks via SDEs: probabilistic QoS measures" *IEEE Transactions on Information Theory*, vol.51, no.12, pp.4396 − 4401. 79, 93

[48] M.M. Olama, S.M. Djouadi, and C.D. Charalambous (2006), "Stochastic power control for time-varying long-term fading wireless networks," *EURASIP Journal on Applied Signal Processing*, Volume 2006, Article ID 89864, 13 pages. 79, 93

[49] M. Olama, S. Djouadi, I. Papageorgious, and C. Charalambous (2008), "Position and velocity tracking in mobile networks using particle and Kalman filtering with comparison," *IEEE Trans. Vehicular Technology*, vol. 57, no. 2, pp. 1001 − 1010. 77, 78

[50] S. Haykin, K. Huber, and Z. Chen (2004), "Bayesian sequential state estimation for MIMO wireless communications," *Proc. IEEE (Special Issue on Sequential State Estimation)*, vol. 92, no. 3, pp. 439 − 454. 78

[51] N. Elia (2004), "When bode meets shannon: Control-oriented feedback communication schemes," *IEEE Trans. Automat. Contr.*, vol. 49, no. 9, pp. 1477 − 1488. 79

[52] S.Z. Denic, and C.D. Charalambous (2007), "Control of jump linear systems over jump communication channels - source-channel matching approach," *International Symposium on Information Theory*, pp. 2491 − 2495. 79

[53] J. Janssen, A. Blasi, R. Blasi and R. Manca (2002), "Discrete time homogeneous and reliability models," *Universit'a "LaSapienza" Roma.* 56

[54] R. Pyke (1961), "Markov renewal rrocesses with finitely many states," *Am. Math. Sta. 32*, pp. 1245 − 1259. 56

[55] V. J. Mathews and J. K. Tugnait (1983), "Detection and estimation with fixed lag for abruptly changing systems," *IEEE Trans. Aerospace and Electronic Systems*, vol. AES−19, No. 5, pp. 730 − 739. 57

[56] L. Campo, P. Mookerjee, and Y. Bar-Shalom (1991), "State estimation for systems with sojourn-time-dependent Markov model switching," *IEEE Trans. Automatic Control*, vol. 36, pp. 238 − 243. 57, 71

[57] H.A.P. Blom and Y. Bar-Shalom (1988), "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 780 − 783. 72

[58] X. Ma, S.M. Djouadi, and H. Li (2011), "Stability conditions for optimal filtering over cognitive radio system," *American Control Conference*, pp. 3447 − 3452. 54

[59] X. Ma, H. Li, and S.M. Djouadi (2011), "Networked system state estimation in smart grid over cognitive radio infrastructures," *the 45th Conference on Information Science System*, pp. 1 − 5. 54

[60] H.J. Chizeck and Y. Ji (1988), "Optimal quadratic control of jump linear systems with Gaussian noise in discrete-time," *Proc. 27th Conf. Decision and Control*, pp. 1989 − 1993. 9

[61] H.J. Chizeck, A.S. Willsky, and D. Castanon (1986), "Discrete-time Markovian-jump linear quadratic optimal control," *Int. J. Control*, vol. 43, no. 1, pp. 213 − 231.

[62] O.L.V. Costa and M.D. Fragoso (1993), "Stability results for discrete-time linear systems with Markovian jumping parameters," *J. Math. Anal. Appl.*, vol. 179, pp. 154 − 178. 87

[63] O.L.V. Costa and R.P. Marques (1998), "Mixed H_2/H_∞ control of discrete-time Markovian jump linear systems," *IEEE Trans. Autom. Control*, vol.43, no. 1, pp. 95 − 100.

[64] Y. Ji and H.J. Chizeck (1988), "Controllability, observability, and discrete-time markovian jump linear quadratic control," *Int. J. Control*, vol. 48, no. 2, pp. 481 − 498.

[65] Y. Ji and H.J. Chizeck (1990), "Jump linear quadratic gaussian control: Steady-state solution and testable conditions, *Control Theory Adv. Technol.*, vol. 6, no. 3, pp. 289 − 318.

[66] Y. Ji, H.J. Chizeck, X. Feng, and K.A. Loparo (1991), "Stability and control of discrete-time jump linear systems," *Control Theory Adv. Technol.*, vol. 7, no. 2, pp. 247 − 270. 9

[67] E.K. Boukas and P. Shi (1997), "H control for discrete-time linear systems with markovian jumping parameters," *Proc. 36th Conf. Decision and Control*, pp. 4134 − 4139.

[68] M. Mariton (1990), *Jump Linear Systems in Automatic Control*, Marcel Dekker.

[69] J.B.R. doVal, J.C. Geromel, and O.L.V. Costa (1999), "Solutions for the linear quadratic control problem of markov jump linear systems," *Joumal of Optiniirarion Theory and Applications*, 103(2):283 − 311. 10

[70] J. D. Birdwell, D.A. Castanon, and M. Athans (1986), "On reliable control system designs," IEEE Trans. Systems, Man, and Cybernetics, vol. SMC-16, no. 5, pp. 703 − 711. 9

[71] P. Seiler and R. Sengupta (2005), "An H_∞ approach to networked control," *IEEE Trans. Automat. Contr.*, vol. 50, no. 3, pp. 356 − 364. 10

[72] D.D. Sworder (1980), "Control of a linear system with nonmarkovian modal changes," *Journal of Economic Dynamics and Control*, vol. 2, pp.233 − 240. 9, 10, 83

[73] C. Schwartz (2003), *Conrrol of semi-Markov jump linear systems with application la the bunch-train cavihzteraction.* PhD thesis, Northwestern University. 9, 10, 83

[74] J. Huang, and Y. Shi (2012), "Stochastic stability and robust stabilization of semi-Markov jump linear systems," Int. J. of Robust and Nonlinear Control, pp. 1 − 16. 9, 10

[75] K. Hochberg, and E. Shmerling (2005), "Stability and opimal control for semi-Markov jump parameter linear systems," *Recent Advances in Applied Probability*, pp. 205 − 221. 10

[76] J. Hespanha, P. Naghshtabrizi, and Y. Xu (2007), "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138 − 162. 10

[77] O.L.V. Costa, M.D. Fragoso, and R.P. Marques (2005), *Discrete-time Markov jump linear systems*, Springer. 87

[78] S. Aberkane (2011), "Stochastic stabilization of a class of nonhomogeneous Markovian jump linear systems," Systems & Control Letters 60, pp. 156 − 160. 87, 91

[79] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan (1994), *Linear matrix inequlities in system and control theory*, SIAM. 89

[80] CC2420 Product Data Sheet (2010), http://www.ti.com. Texas Instruments. 94

[81] Y. Chen, and A. Terzis (2010), "On the mechanisms and effects of calibrating RSSI measurements for 802.15.4 radios," *Proceedings of the European Conference on Wireless Sensor Networks (EWSN)*, pp. 256 − 271. 94

[82] K. Farkas, T. Hossmann, F. Legendre, B. Plattner, and S.K. Das (2008), "Link quality prediction in mesh networks," *Computer Communications*, 31:1497 − 1512. 10

[83] J. Shu, L. Liu, Y. Fan and K. Yu (2010), "A link quality prediction mechanism for wsns based on time series model," *Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing*, pp. 175 − 179. 10

[84] J. Shu, Z. Wu L. Liu, J. Li and Y. Chen (2010), "Cci-based link quality estimation mechanism for wireless sensor networks under perceive packet loss," *Journal of Software*, 5(4), pp.1 − 5. 10

[85] T. Liu, and A.E. Cerpa (2011), "Foresee (4c): Wireless link prediction using link features," *ISPN11*, pp. 294 − 305. 10

[86] W.C. Wong Z. Wu and J. Wu (2009), "Adaptive link quality estimation for wireless ad hoc networks," *ICICS 2009*, pp. 1 − 5. 10

[87] Datasheet (2006), Crossbow technology inc. San Jose, California. 109

[88] CC2420. 2.4 ghz ieee 802.15, 4/zigbee-ready rf transceiver specification. 109

[89] S.F. Smith, M. Bobrek, C. Jin, and M. R. Moore (2005), "Design of an agile radionavigation system using SDR techniques," *Proceedings of IEEE Milcom*, pp. 1127 − 1132. 6, 7, 11, 111

[90] B.W. Parkinson, and J.J. Spilker (1996), *Global positioning system: Theory and applications*, *AIAA series*, vol. 16. 11, 111, 112, 116

[91] M. Mehartash (2008), *GPS navigation toolbox*, pp. 5 − 10. 11, 111, 112, 113, 116

[92] S. Cooper, and H. Durrant-Whyte (1994), "A Kalman filter model for GPS navigation of land vehicles," Intelligent Robots and Systems 94, vol.1, pp. 157 − 163. 11, 112, 116

[93] E.J. Krakiwsky, C.B. Harris, and R.V.C. Wong (1988), "A Kalman filter for integrating dead reckoning, map matching and GPS positioning," *Position Location and Navigation Symposium*, pp. 39 − 46. 11, 112, 116

[94] G. Beyerle, and K. Hocke (2001), "Observation and simulation of direct and reflected GPS signals in Radio Occultation Experiments," *GEOPHYSICAL RESEARCH LETTERS*, vol. 28, no. 9, pp. 1895 − 1898. 11, 112, 116

[95] A. Giremus, A. Doucet, V. Calmettes, and J.Y. Tourneret (200), "A Rao-Blackwellized particle filter for INS/GPS integration," *Acoustics, Speech, and Signal Processing*, iii-964-7 vol. 3. 11, 112, 116

[96] L.L. Scharf, and D.L. Alspach (1972), "On stochastic approxiamtion and an adaptive Kalman filter," *Decision and Control, 1972 and 11th Symposium on Adaptive Processes*, pp. 253 − 257. 116, 117

[97] M.S. Grewal, L.R. Weill, and A.P. Andrews (2001), *Global positioning systems, inertial navigation, and integration*, John Wiley & Sons, Inc. 116

[98] M. Matosevic, Z. Salcic, and S. Berber (2006), "A comparison of accuracy using a GPS and a low-cost DGPS," *IEEE Trans. Instrument and Measurement*, vol. 55, no. 5, pp. 1677 − 1683.

[99] I. Skog, and P. Handol (2011), "Time synchronization errors in loosely coupled GPS-aided inertial navigation systems," *IEEE Trans. Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1014 − 1023. 6

[100] S. Hong, M.H. Lee, S.H. Kong, and H.H. Chun (2004), "A car test for the estimation of GPS/INS alignment errors," *IEEE Trans. Intelligent Transportation Systems*, vol. 5, no. 3, pp. 208 − 218. 6

[101] M.M. Olama, S.M. Djouadi, and C.D. Charalambous (2009), "Stochastic differential equations for modeling, estimation and identification of mobile-to -mobile communication channels," *IEEE Trans. Wirelesse Communication*, vol. 8, no. 4, pp. 1754 − 1763. 78, 79, 118

[102] D. Pfeffermann, and R. Tiller (2003), "State-space modeling with correlated measurements with application to small area estimation under benchmark constraints," Southampton, UK, Southampton Statistical Sciences Research Institute, 21pp, *S3RI Methodology Working Papers*. 118

[103] X. Ma, H. Li, and S.M. Djouadi (2011), "Stochastic modeling of short-term power consumption for smart grid: A state space approach and real measurement demonstration," *IEEE 45th Conference on Information Sciences and Systems*, Baltimore, MD, pp. 1 − 5. 118

[104] R.P. Roesser (1975), "A discrete state-space model for linear image processing," *IEEE Transaction on Automatic Control*, AC-20, no.1, pp. 1 − 10. 118

[105] R.L. Smith, and J.E. Miller (1986), "A non-Gaussian state space model and application to prediction of records," *J.R. Statistic. Soc. B*, vol.48, no.1, pp. 77 – 89. 118

[106] R.J. Elliott and V. Krishnamurthy (1999), "New finite-dimensional filters for parameter estimation of discrete-time linear guassian models," *IEEE Trans. on Automatic Control*, vol. 44, pp. 938 – 951. 97

[107] L. Ljung (1999), *System identification: Theory for the sser*, Upper Saddle River, NJ, Prentice-Hal PTR. 119

[108] T.C. Mills (1990), *Time series techniques for economists*. Cambridge University Press. 120

[109] R.J. Bhansali (1978), "Linear prediction by autoregressive model fitting in the time domain," *The Annals of Statistics*, vol. 6, no. 1, pp 224 – 231.

[110] R. Lewis, and G. C. Reinsel (1985), "Prediction of multivariate time series by autoregressive model fitting," *Journal of Multivariate Analysis*, vol. 16, pp. 393 – 411. 120

[111] S.R. Dubois, and F.H. Glanz (1986), "An autoregressive model approach to two-dimensional shape classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 55 – 66. 120

[112] X. Zhang, and X. Wu (2008), "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," *IEEE Transactions on Image Processing*, vol. 17, pp. 887 – 896. 120

[113] M. McAleera, and M.C. Medeiros (2008), "A multiple regime smooth transition Heterogeneous Autoregressive model for long memory and asymmetries," *Journal of Econometrics*, vol. 147, pp. 104 – 119. 120

[114] L. Ljung, and T. Soderstrom (1983), *Theory and practice of recursive identification*, The MIT Press. 97, 121

[115] M. Verhaegen and V. Verdult (2007), *Filtering and system identification: A least squares approach*, CAMBRIDGE. 100, 101, 103

[116] Q. Cao, T. Abdelzaher, J. Stankovic, and T. He (2008), "The LiteOS operating system: Towards unix-like abstractions for wireless sensor networks," *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 233 − 244. 103

[117] T.D. Wickens (2004), "The general linear model," *Graduate Summer School Program Institute for Pure and Applied Mathematics University of California*, Los Angeles.

[118] P. McCullagh, and J. Nelder (1989), *Generalized linear models*, Second Edition. Boca Raton: Chapman and Hall/CRC. 121, 122

[119] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis (2009), "Collection tree protocol," *In Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pp. 1 − 14. 4

[120] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J.A Stankovic, T.F. Abdelzaher, J. Hui, and B. Krogh (2006), "VigilNet: An integrated sensor network system for energy-efficient surveillance," *ACM Transactions on Sensor Networks*, 2(1):1 − 38. 5

[121] L. Luo, Q. Cao, C. Huang, T. Abdelzaher, J.A. Stankovic, and M. Ward (2007), "EnviroMic : Towards cooperative storage and retrieval in audio sensor networks," *In Proceedings of the International Conference on Distributed Computing Systems (ICDCS)*, pp. 1 − 22. 4

[122] G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh (2006), "Fidelity and yield in a volcano monitoring sensor network," *In Proceedings of the*

USENIX Symposium on Operating Systems Design and Implementation (OSDI), vol. 7, pp. 381 − 396. 4

[123] Q. Cao, T. He, L. Fang, T. Abdelzaher, J. Stankovic, and S. Son (2011), "Efficiency centric communication model for wireless sensor networks," *INFOCOM*, pp. 1 − 12.

[124] X. Ma, S.M. Djouadi, P. Crilly, S. Sahyoun, and S.F. Smith (2011), "Improved localization in GPS-denied environments using an autoregressive model and a generalized linear model," *Military Communications Conference*, pp. 760 − 765. 112

[125] X. Ma, S.M. Djouadi, P. Crilly, S. Sahyoun, and S.F. Smith (2011), "Navigation in GPS-denied environments," *14th International Conference on Information Fusion*, pp. 1686 − 1692. 112

[126] X. Ma, S. Djouadi, and Q. Cao (2011), "LIPS: Link prediction as a service for adaptive data aggregation in wireless sensor networks," *IEEE International Conference on Distributed Computing in Sensor Systems*, 1 − 5. 94

[127] X. Ma, S.M. Djouadi, and H. Li (2012), "State estimation over a semi-Markov model based cognitive radio system," *IEEE Trans. Wireless Communication*, pp. 2391 − 2401 . 54

[128] X. Ma, S. Djouadi, and C. Charalambous (2011), "Optimal filtering over uncertain wireless communication channels," *IEEE Signal Processing Letters*, vol. 18, pp. 359 − 362. 14

Appendix

Appendix A

Summary of Equations

A.1 Proofs

A.1.1 Proof of Lemma 3.3.1

Proof. Since the separation principle holds in this case, estimation and control can be performed separately. Consider the error covariance

$$P_{k+1} = \mathbb{E}\{e_{k+1}e_{k+1}^T|I_k\} = AP_kA^T + Bp_jVB^T - s_r^kAP_kC^T(CP_kC^T + W)^{-1}CP_kA^T \tag{A.1}$$

where $P_{k+1} = P_{k+1|k}$ and $p_j = \mathbb{E}\{s_i^k|s_r^k\}$.

Define $I = \max\{I_0, I_1\}$ and $F(P) = APA^T + p_jBVB^T - APC^T(CPC^T + W)^{-1}CPA^T$. When $1 \leq i \leq \max(I_0, 1)$, there always exist $c_i^{(1)} \geq 0$ and $c_i^{(0)}$ that satisfy the following inequality [7]:

$$\parallel F^i(P) \parallel \leq \ c_i^{(1)} \parallel P \parallel + c_i^{(0)}$$

where $\parallel X \parallel$ refers to the matrix induced norm $\parallel X \parallel = \max_{|X|=1} |MX|$ where $|X|$ and $|MX|$ denote the usual Euclidean norm for vectors.

By [7], where only estimation is considered, the peak covariance process $\{P_n^p\}_{n\geq 1}$ of $\{P_k\}_{k\geq 1}$ is stable if condition (i) above holds and

$$(1-q)qc_1^{(1)}[1+\sum_{i=1}^{I_0-1}c_i^{(1)}q^i]\sum_{j=1}^{\infty}\parallel A^j\parallel^2(1-q)^{j-1}<1 \tag{A.2}$$

is satisfied.

Consider the control part, from the close-loop equation above, we have

$$\hat{x}_{k+1}=s_r^kAK_kCe_k+(A-s_r^kBF)\hat{x}_k+s_r^kAK_kw_k \tag{A.3}$$

Let $M_{k+1}=\mathbb{E}\{\hat{x}_{k+1}\hat{x}_{k+1}^T|I_k\}$, then we have

$$M_{k+1}=AM_kA^T+T_k-s_r^k(BFM_kA^T+AM_kF^TB^T-BFM_kF^TB^T) \tag{A.4}$$

where $T_k=s_r^kAP_kC^T(CP_kC^T+W)^{-1}CP_kA^T$. $\parallel T_k\parallel$ is bounded if condition (i) and (A.2) hold. To see this, note $\{P_n^p\}_{n\geq 1}$ is stable, then from each β_n to α_{n+1}, $s_r^k=1$ for a successive period, it follows that P_k is bounded in this period based on Kalman filtering theory. This leads that $\parallel T_k\parallel$ is also bounded in that period. While, once s_r^k becomes 0, $T_k=0$ in that period.

Define $G(M)=AMA^T+T_s-(BFMA^T+AMF^TB^T-BFMF^TB^T)$, where $T_s:=\{T_k:\parallel T_k\parallel=sup_{g\geq 1}\parallel T_g\parallel\}$. Similarly, when $1\leq i\leq \max(I_1,1)$, there always exist $e_i^{(1)}\geq 0$ and $e_i^{(0)}$ that satisfy the following inequality [7]:

$$\parallel G^i(M)\parallel\leq\ e_i^{(1)}\parallel M\parallel+e_i^{(0)}$$

Following the same arguments in [7], besides condition (i) and (A.2),

$$(1-q)qe_1^{(1)}[1+\sum_{i=1}^{I_1-1}e_i^{(1)}q^i]\sum_{j=1}^{\infty}\parallel A^j\parallel^2(1-q)^{j-1}<1 \tag{A.5}$$

is also satisfied (note in (A.5) I_1 is used instead of I_0 in (A.2)), then the peak covariance process $\{M_n^p\}_{n\geq 1}$ of $\{M_k\}_{k\geq 1}$ is stable. We have now

$$L_{k+1} = \mathbb{E}\left\{ \begin{bmatrix} e_{k+1} \\ \hat{x}_{k+1} \end{bmatrix} \begin{bmatrix} e_{k+1} \\ \hat{x}_{k+1} \end{bmatrix}^T | I_k \right\} = \begin{bmatrix} P_{k+1} & 0 \\ 0 & M_{k+1} \end{bmatrix}$$

Thus, $L_n^p = \begin{bmatrix} P_n^p & 0 \\ 0 & M_n^p \end{bmatrix}$ is stable. Define for $1 \leq i \leq \min\{(I_0-1),(I_1-1)\}$, $d_i^{(1)} = max\{c_i^{(1)}, e_i^{(1)}\}$; For $i > \min\{(I_0-1),(I_1-1)\}$, if $I_0 \geq I_1$, $d_i^{(1)} = c_i^{(1)}$, otherwise $d_i^{(1)} = e_i^{(1)}$. Then, we can combine (A.2) and (A.5) together and get condition (ii) in the lemma. □

A.2 Equations

A.2.1 Expressions of Theorem 3.3

Let $\hat{K}_k = \mathbb{E}\{K_k C A K_k\}$, then

$\Phi_k^1 = \begin{bmatrix} \Phi_k^{11} & \Phi_k^{12} \\ \Phi_k^{13} & \Phi_k^{14} \end{bmatrix}$, $\Phi_k^2 = \begin{bmatrix} \Phi_k^{21} & \Phi_k^{22} \\ \Phi_k^{23} & \Phi_k^{24} \end{bmatrix}$, $\Phi_k^3 = \begin{bmatrix} \Phi_k^{31} & \Phi_k^{32} \\ \Phi_k^{33} & \Phi_k^{34} \end{bmatrix}$, $\Phi_k^4 = \begin{bmatrix} \Phi_k^{41} & \Phi_k^{42} \\ \Phi_k^{43} & \Phi_k^{44} \end{bmatrix}$,

where

$$\Phi_k^{11} = A^2 - pqA^2\tilde{K}_k CA + pqA\hat{K}_k C$$
$$\Phi_k^{12} = (1-p)pq(A\hat{K}_k C + A\tilde{K}_k CBF)$$
$$\Phi_k^{13} = pqA^2\tilde{K}_k C - pqA\hat{K}_k C$$
$$\Phi_k^{14} = A^2 - pqABF - (1-p)pqA\hat{K}_k C - pqA\tilde{K}_k CA + p^2 qA\tilde{K}_k CBF$$
$$\Phi_k^{21} = (1-p)pq(A\hat{K}_k C + BFA\tilde{K}_k C)$$
$$\Phi_k^{22} = -(1-p)PQ(A\hat{K}_k C + BFA\tilde{K}_k C) - (1-p)pq(A\tilde{K}_k CBF + BFBF)$$
$$\Phi_k^{23} = -(1-p)pq(A\hat{K}_k C + BFA\tilde{K}_k C)$$
$$\Phi_k^{24} = (1-p)pq(A\tilde{K}_k C + BF)BF - (1-p)pq(A\hat{K}_k C + BFA\tilde{K}_k C)$$

$$\Phi_k^{31} = pqA\tilde{K}_kCA - pqA\hat{K}_kC$$

$$\Phi_k^{32} = -(1-p)PQ(A\hat{K}_kC + A\tilde{K}_kCBF)$$

$$\Phi_k^{33} = pqA\hat{K}_kC$$

$$\Phi_k^{34} = pqA\hat{K}_kCA - p^2qA\hat{K}_kCBF + (1-p)pqA\hat{K}_kC$$

$$\Phi_k^{41} = A^2 - pqBFA - pqA^2\tilde{K}_kC + p^2qBFA\tilde{K}_kC - (1-p)pqA\hat{K}_kC$$

$$\Phi_k^{42} = (1-p)pqBF(A\tilde{K}_kC + BF) - (1-p)pq(A\hat{K}_kC + A\tilde{K}_kCBF)$$

$$\Phi_k^{43} = pqA^2\tilde{K}_kC - p^2qBFA\tilde{K}_kC + (1-p)pqA\hat{K}_kC$$

$$\Phi_k^{44} = A^2 - pqABF - pqBFA + p^2qBFBF + (1-p)pqA\hat{K}_kC$$

Vita

Xiao Ma received the B.S. and M.S. degrees from Harbin Institute of Technology, Harbin, China, in 2005 and 2007, respectively. He is currently pursuing the Ph.D. degree at the University of Tennessee, Knoxville. His research interests include control over communication channels and sensor networks, modeling and identification in wireless channels, localization and navigation in navigation systems.