AN ABSTRACT OF THE THESIS OF

Duong Nguyen-Huu for the degree of Master of Science in

Electrical and Computer Engineering presented on May 20, 2013.

Title: Fast Queuing Policies via Convex Relaxation

Abstract approved: ______________________________

Thinh Nguyen

Traditionally, networking protocol designs have placed much emphasis on point-to-point reliability and efficiency. With the recent rise of mobile and multimedia applications, other considerations such as power consumption and/or Quality of Service (QoS) are becoming increasingly important factors in designing network protocols. As such, we present a new flexible framework for designing robust network protocols under varying network conditions that attempts to integrates various given objectives while satisfying some pre-specified levels of Quality of Service. The proposed framework abstracts a network protocol as a queuing policy, and relies on the optimization methods of convex relaxation and the theory of mixing time for finding the fast queuing policies that drive the distribution of packets in a queue to a given target stationary distribution. It is argued that a target stationary distribution can be used to characterize various performance metrics of network flow. Thus, finding a fast queuing policy that produces a given target stationary distribution is vital in achieving some given objectives. In addition, we show how to augment the basic proposed framework in order to obtain a queuing policy that produces ϵ-approximation to the target distribution with even faster convergence time. This fast adaptation is especially useful for networking applications in fast-changing network conditions. Both theory and simulation results are presented to verify our framework.

Fast Queuing Policies via Convex Relaxation

by

Duong Nguyen-Huu

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented May 20, 2013
Commencement June 2013

Master of Science thesis of Duong Nguyen-Huu presented on May 20, 2013.

APPROVED:

Major Professor, representing Electrical and Computer Engineering

---

Director of the School of Electrical Engineering and Computer Science

---

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Duong Nguyen-Huu, Author

ACKNOWLEDGEMENTS

I would like to thank my adviser, all my friends and colleagues in Oregon State University for making the stay here enjoyable and memorable.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF APPENDICES

Fast Queuing Policies via Convex Relaxation

Chapter 1 – Introduction

Traditionally, networking protocol designs have placed much emphasis on point-to-point reliability and efficiency. With the recent rise of mobile and multimedia applications made possible by various wireless network architectures, other considerations such as power consumption and/or Quality of Service (QoS), e.g., the requirements on minimum bandwidth, maximum jitter, delay, or loss, are becoming increasingly important factors in designing network protocols. Indeed, many current proprietary network protocols are often optimized for certain objectives, based on various requirements of particular devices and/or applications. For example, a real-time video conference application might employ a real-time video streaming protocol which is designed with emphasis on packet delay. On the other hand, a network protocol with small power consumption is preferable for smart phones. That said, the situation is made more complex by the fact that a protocol optimized for i-Phones might result in much power consumption on a Samsung Galaxy due to the fundamental differences in operating systems as well as hardware architecture. In addition, to be efficient, today network protocols must cope with the fast-changing and non-stationary characteristics of wireless channels as well as fluctuating traffic amount induced by the diversity of modern applications. Therefore, in this paper, we present a framework for customized designs of robust network protocols that achieve various objectives and requirements imposed by the heterogeneity of applications and hardware architectures under fast-changing, non-stationary environments.

The proposed framework relies on three components: (1) the abstraction of a network protocol as a queuing policy in order to allow for generalization of protocol designs as well as tractable analysis using queuing theory; (2) the optimization formulation, specifically the convex relaxation, that provides the flexibility in specifying various objectives and constraints induced by different applications and hardware architectures; (3) the theory of mixing time that helps design the protocols to promptly achieve the given objective to minimize the effect of the abrupt changes in the environments. Regarding the first component, the abstraction of a network protocol as a queuing policy is commonly used in many network simulators [1] and has been the corner stone for the highly successful queuing theory. In fact, since the development of packet-switched networks in early 1960s, queuing theory [2] has been a critical part in the performance analysis for many transmission protocols. The performance of current wireless transmission protocols such as the IEEE 802.11 protocols can be formally analyzed using queuing theory. As for the second component, recent advances in convex optimization methods and computing power have made solving complex problems in real time using numerical methods and algorithms an attractive approach [3]. Optimization is a powerful framework for finding optimal solutions to a variety of complex problems with multiple constraints using algorithms which often cannot be obtained via analytical methods. Also, it is often relatively easy for a network protocol designer to cast his/her problem in the canonical form of the convex optimization problem. The optimization package can then uses variety of algorithms to find the solutions numerically. Finally, regarding the third component, traditional adaptive protocols attempt to change their behaviors based on the updated information from the surrounding. In contrast, the proposed framework assumes that in some situations, it might not be possible to gather the updated information accurately and timely so that a protocol can use it to respond appropriately. Instead, the proposed approach uses the theory of mixing time to provide the insight about how to design robust protocols by quickly achieving the given objective based on the current information before the environments change.

Network Protocol as Queuing Policy. A primary objective of a network protocol is to regulate the transmission rate of data between two endpoints of a communication link. For example, TCP adapts its sending rates to the current traffic conditions by linearly increasing the rate when no packet loss is observed and multiplicatively decreasing the rate when a loss occurs. Packets in transit between two endpoints on the Internet can be thought to be in a queue. Thus, a network protocol can be viewed as a queuing policy that controls the rates of incoming packets at the sender and outgoing packets at the receiver. In many other settings, queues are implemented explicitly in communicating devices, and the network protocols make use of them directly. Notably, in a number of randomized medium access control (MAC) protocols used in wireless environments, the queue fullness, i.e., the number of packets in the queue is used as an indication of the current traffic conditions. This information is then used to determine the probability of accessing the wireless channel which is proportional to the sending rate. Thus, understanding the dynamics of packets in queues over time as a result of employing certain queuing policy/network protocol, enables the system engineers to characterize and to predict various properties of the data flow such as bandwidth, packet loss and delay.

Stationary Distribution. Central to our approach is the notion of the *stationary distribution* of packets in the queue associated with each queuing policy. The stationary distribution is important in characterizing various properties of a protocol. As will be discussed shortly, stationary distribution can be used to characterize the traditional QoS metrics such as loss and delay. It is an important parameters to be optimized for many objectives including the average consumption power of the protocol. Therefore, finding a queuing policy that produces a desired stationary distribution in the fastest time is one of the main goals of the paper.

Contribution: We consider a general class of queuing polices/network protocols with the ability to adjust the sending and receiving rates *probabilistically*. The probabilistic framework arises naturally from the unavoidable uncertainties in when and how fast packets arrive due to the fluctuations in network traffic. Furthermore, in some scenarios the ability to send packets out (de-queue) successfully at any time is probabilistic. For example, in a Wi-Fi network, a wireless node might not be able to successfully send out a packet (de-queue) at a certain time slot due to possible collision with other node's transmission. Also, its random back-off mechanism after a collision can in fact be viewed as a dequeuing operation with a certain probability. Our contributions include an convex optimization framework for providing Quality of Service (QoS) using a *fast* queuing policy that achieves a given stationary distribution. Indeed, a given stationary distribution allows for a more general and precise control of various QoS requirements. In addition, we show how an even faster queuing policy can be achieved when the queuing policy only needs to produce a stationary distribution that is ϵ-close to the given target stationary distribution. The fast adaptive queuing policies are especially useful for applications in fast-changing network conditions. Our framework is developed based on the theory of fast mixing Markov chain and convex optimization. Finally, we show how the proposed framework can be applied to optimize for a wide range of objectives beyond the standard QoS requirement, such as power consumption.

Outline. The thesis is organized as follows. In Section 2.1, we discuss the approach to QoS via stationary distribution. In Section 2.2, we provide some background on the Markov Chain and queuing theories as they are necessary for the development of our proposed framework. In Section 3.1, we present a novel convex optimization framework with multiple formulations for finding fastest mixing time queuing policies. In Section 3.3, we describe the application of the above framework for the tridiagonal queuing cases. In Section 3.4 and Chapter 4, we show an application of our framework to finding a queuing policy that optimizes for a given objective for a flow while ensuring the mean and variance of queuing delay are within given bounds. Finally, we provide a few concluding remarks in Chapter 6.

Chapter 2 – Preliminaries

In this chapter we present the main definitions and some results which will be useful in our analysis later.

2.1 Queuing Policy, Stationary Distribution, and QoS

A network protocol is abstracted as a queuing policy which is governed by a tridiagonal transition probability matrix as shown in Fig. 2.1. The dimension N of the matrix represents the maximum length of the queue. The diagonal, left-of-diagonal, right-of-diagonal entries in the tridiagonal transition probability matrix represents the probabilities of the number of packets in the queue stays the same, decreases by one, or increases by one.

We first use a simple example of a discrete-time version of the classical $M/M/1/k$ queuing model to illustrate the relationship between the stationary distribution induced by a queuing policy and QoS. Assume that at the beginning of each time step, exactly one packet arrives at the queue with probability $p = 0.4$. Otherwise, with probability $1 - p = 0.6$, no packet arrives during that entire time step. Furthermore, a queuing policy is used such that at the beginning of each time step, exactly one packet is dequeued with probability $q = 0.6$. Otherwise, with probability $1 - q = 0.4$, no packet is dequeued during that entire time step. Furthermore, for simplicity let $k = 2$ be the maximum queue size, and a newly arrived packet is

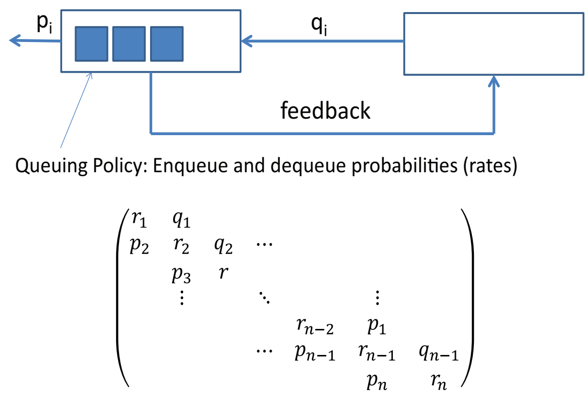


Figure 2.1: Queuing policy can be viewed as a tridiagonal transition probability matrix

dropped if the queue is full. The dynamic of the number of packets in the queue over time can be shown to be governed by the following transition probability matrix:

$$P = \begin{pmatrix} 0.84 & 0.16 & 0 \\ 0.36 & 0.48 & 0.16 \\ 0 & 0.36 & 0.64 \end{pmatrix},$$

where P_{ij} denotes the probability that the queue will have j packets in the next time step, given that it currently has i packets with $i, j \in \{0, 1, 2\}$. For each aperiodic and irreducible P, there exists a unique corresponding stationary distribution π such that $\pi^T P = \pi^T$. In this particular case,

$$\pi = \begin{pmatrix} 0.61 \\ 0.27 \\ 0.12 \end{pmatrix}.$$

The stationary distribution π characterizes the long term or stationary probability of the queue occupancy. In this case, out of all the observed time slots, 61% of time the queue is empty, 27% of the time the queue has exactly one packet, and 12% of the time the queue has two packets. Knowing exactly this distribution, the average queuing delay can be precisely calculated. One can also immediately bound the probability of dropped packets to no more than 0.12. In fact, any statistical measure, e.g., moments of any order can be theoretically calculated for the given stationary distribution.

Transition Probability is induced by Queuing Policy. Suppose the QoS requirements are given in terms of maximum average packet latency and minimum packet drop rate, then one can find a stationary distribution π that satisfies such requirements (in Section 3.4.2, we show a general procedure to obtain the stationary distribution π subject to any constraints). However, there are many transition probability matrices P that have the same stationary distribution π. It is important to note that each transition probability P is a result of applying a certain queuing policy. For the example above, the associated queuing policy is to send packets with probability of 0.6. One can easily implement another policy that sends packets with a different probability which results in a different transition probability. Moreover, we need not restrict ourselves to the class of policies that sends and receives packets with some fixed probabilities. Rather, one can design a policy that sends and receives packets with different probabilities based on the number of packets presently in the queue.

Constraints on Queuing Policy. Intuitively, for a high priority flow $\pi = [1, 0, 0]^T$ seems to be the best stationary distribution since the queue is always empty. However, this implies that a packet is always dequeued at every time slot. This policy might not be possible or optimal due to several reasons. For example, let us consider a wireless network consisting of multiple nodes. First, if an application does not require much throughput, then sending packets all the time consumes more power than necessary. Second, if every node in the wireless network implements the same greedy queuing policy, then collisions will happen all the time, resulting in low overall throughput. Thus, the transition probability matrix (hence the queuing policy) must be selected from a pre-specified class of transition probability matrices that gives rise to reasonable queuing policies for the given settings. This constraint is an input to our convex optimization framework to be described shortly.

Fastest Queuing Policy. We noted above that there are many transition probability matrices P (equivalently many queuing policies) that have the same given stationary distribution π, and all satisfy the pre-specified QoS requirements. So which transition probability matrix should one choose? The theory of Markov chain shows that if we apply the same queuing policy over many time steps, the distribution of packets in the queue will converge to a unique stationary distribution corresponding to a stochastic, aperiodic and irreducible matrix P, regardless of the initial distribution of packets in the queue. Mathematically, let ν be any initial distribution of packets in the queue, then

$$\lim_{n\to\infty} \nu^T P^n = \pi^T, \tag{2.1}$$

where n is the number of time steps.

If the network traffic is stationary, then π can be obtained approximately using the same queuing policy after some sufficiently large number of time steps. Ideally, we want the queuing policy that drives the distribution of packets in the queue to the desired stationary distribution in the fastest time,i.e., smallest n for any initial distribution. This is especially useful when the network conditions change and thus fast adapting queuing policy is preferable. Another important point about this fast adapting principle is that if for some reasons, the network traffic becomes bursty for a short while that temporarily fill up the queue, a fast queuing policy will quickly drive the queue to the desirable average number of packets, i.e., the target stationary distribution.

2.2 Mixing Time and Spectral Gap

In order to quantify "fast" queuing policy, i.e., how fast a queuing policy drive an initial distribution to a given target stationary distribution, it is necessary to define a similarity measure between two distributions. One common similarity measure is the total variance distance defined below:

Definition 1 (Total variation distance) *For any two probability distributions ν and π on a finite state space Ω, we define the total variation distance as:*

$$\|\nu - \pi\|_{TV} = \frac{1}{2} \sum_{i \in \Omega} |\nu(i) - \pi(i)| \,.$$

□

We now use the similarity measure to define an important notion called mixing time below:

Definition 2 (Mixing time) *For a discrete, aperiodic and irreducible Markov chain with transition probability P and stationary distribution π, given an $\epsilon > 0$, the mixing time $t_{mix}(\epsilon)$ is defined as*

$$t_{mix}(\epsilon) = \inf \left\{ n : \|\nu^T P^n - \pi^T\|_{TV} \leq \epsilon \textit{ for all probability distributions } \nu \right\}.$$

□

Essentially, the mixing time of a discrete time Markov chain is the minimum number of time step n until the total variance distance between the n-step distribution ad the stationary distribution is less than ϵ. We will use the mixing time to characterize the convergence rate of a queuing policy. One of the successful techniques for bounding the mixing time of a stochastic matrix is via its spectral characterization, i.e., its eigenvalues.

Eigenvalues and Eigenvectors. A non-zero vector v_i is called a right (left) eigenvector of a square matrix P if there is a scalar λ_i such that: $Pv_i = \lambda_i v_i$ or $(v_i^T P = \lambda v_i^T)$. The scalar λ_i is said to be an eigenvalue of P. If P is a stochastic matrix, then $|\lambda_i| \leq 1, \forall i$. Denote the set of eigenvalues in non-increasing order:

$$1 = \lambda_1(P) \geq \lambda_2(P) \geq \cdots \geq \lambda_{|\Omega|}(P) \geq -1$$

Definition 3 (Second largest eigenvalue modulus) *The second largest eigenvalue modulus (SLEM) of a matrix P is defined as:*

$$\mu(P) = \max_{i=2,\ldots,|\Omega|} |\lambda_i(P)| = \max\{\lambda_2(P), -\lambda_{|\Omega|}(P)\} \tag{2.2}$$

□

In this paper, we also make use the reversibility property of Markov chain defined as follows:

Definition 4 (Reversible Markov Chain) *[4] A discrete Markov chain with a transition probability P is said to be reversible if there exist a probability π satisfies the following detailed balance equations:*

$$P_{ij}\pi(i) = P_{ji}\pi(j) \tag{2.3}$$

□

Proposition 1 *For any discrete-time Markov chain with tridiagonal stochastic transition matrix, the chain is reversible.* □

Proof: See Appendix. ■

We now show an important bound that relates the mixing time of the Markov chain to the SLEM of a reversible matrix P.

Theorem 1 (Bound on mixing time) *[4]. Let P be the transition matrix of a reversible, irreducible and aperiodic Markov chain with state space Ω, and let $\pi_{min} := \min_{x\in\Omega} \pi(x)$. Then*

$$t_{mix}(\epsilon) \leq \frac{1}{1-\mu(P)} \log\big(\frac{1}{\epsilon\pi_{min}}\big). \tag{2.4}$$

□

Proof: See Appendix. ■

Suppose at time step t we have $d(t) = \|\nu^T P^t - \pi^T\|_{TV} \leq \epsilon$ then from the definition of mixing time: $t_{mix}(\epsilon) \geq t$. From the theorem (1), we have:

$$t \leq \frac{1}{1-\mu(P)} \log\big(\frac{1}{\epsilon\pi_{min}}\big)$$

Hence

$$d(t) \leq \epsilon \leq \pi_{min}^{-1} e^{-(1-\mu(P))t}$$

Therefore, $d(t)$ converges to 0 asymptotically as $e^{-(1-\mu(P))t}$. Here, we consider the quantity $1 - \mu(P)$ as **mixing rate**.

Since the mixing time is a function of ϵ and the mixing times of two matrices might be smaller or larger than the other at different ϵ, we are more interested in the mixing rate. Thus, finding the matrix P with minimum $\mu(P)$ would result in the fastest convergence time which will be the topic in the next section.

Chapter 3 – Robust Queuing Policies/Network Protocols via Convex Relaxation Optimization

In this chapter, we present a number of convex relaxation optimization formulations for finding tridiagonal transition probability matrix with fast mixing rate and achieve a given target stationary distribution. Based on this, we present an augmented framework for finding fast queuing policies that are optimized any objectives that is convex in stationary distribution.

3.1 Fast Mixing Tridiagonal Matrix for a Given Stationary Distribution

We assume that a stationary distribution is given. The goal is to find a tridiagonal transition probability matrix with fastest mixing rate. It was shown in [5] that

$$\mu(P) = ||D_\pi^{1/2} P D_\pi^{-1/2} - \sqrt{\pi}(\sqrt{\pi})^T||_2, \tag{3.1}$$

where π denotes the stationary distribution of P, D_π denotes the square diagonal matrix whose diagonal entries are taken from each elements of π, and $||.||$ denote l_2-induced matrix norm. In (3.1), P must be reversible. Furthermore, $\mu(P)$ is a convex function in P.

Our first convex optimization is: given the some requirements, i.e., a desired stationary distribution of Markov Chain, design the fastest chain with transition matrix (P) that drives the chain from any state to the desired stationary distribution. It was first formulated broadly in [5] as:

Problem 1 - FMMC.

$$
\begin{aligned}
&\text{Minimize } ||D_{\pi^*}^{1/2} P D_{\pi^*}^{-1/2} - \sqrt{\pi^*}(\sqrt{\pi^*})^T||_2 \\
&\text{Subject to : } \begin{cases} P\mathbf{1} = \mathbf{1} \\ D_{\pi^*} P = P^T D_{\pi^*} \\ \text{other convex constraints on } P. \end{cases}
\end{aligned} \tag{3.2}
$$

The objective function is SLEM. The first constraint ensures P is a stochastic matrix. The second constraint is for reversibility. The third constraint is imposed by limitations of certain settings of the chain. The solution of the problem (if exists) is a transition matrix P_{opt} which has the smallest SLEM, resulting fastest convergence time to the given target distribution π^*. However, these constraints, especially the third constraint, can be restricted that given a stationary distribution π^*, there might not be a P that simultaneously satisfies all the constraints and produces the desired stationary distribution. For example, consider a queuing policy, if one restricts the queuing policy to always send packets at some constant rate (q) regardless of how many packets in the queue, then there is less flexibility in producing the desired π^*. In addition, in many settings, finding a queuing policy that produces a stationary distribution that is within some small ϵ of the target stationary distribution, but has faster convergence rate might be preferable. This is especially useful when network conditions change quickly. On the other hand, a slow adapting chain is optimal for the past rather than the present network conditions. Based on this, we propose the following optimization problem (P2):

Problem 2.

$$\text{Minimize } ||D_\pi^{1/2} P D_\pi^{-1/2} - \sqrt{\pi}(\sqrt{\pi})^T||_2$$
$$\text{Subject to : } \begin{cases} P\mathbf{1} = \mathbf{1} \\ D_\pi P = P^T D_\pi \\ \text{Other constraints on } P. \\ ||\pi^* - \pi||_2 \leq \epsilon \end{cases} \tag{3.3}$$

The optimization variables in (P2) are both P and π. Unfortunately, (P2) is non-convex. Therefore, we propose the following convex problem (P3) to find the approximate solution for (P2).

Problem 3 - EFMMC.

$$\text{Minimize } ||D_{\pi^*}^{1/2} P D_{\pi^*}^{-1/2} - \sqrt{\pi^*}(\sqrt{\pi^*})^T||_2$$
$$\text{Subject to : } \begin{cases} P\mathbf{1} = \mathbf{1} \\ ||\pi^{*T} P - \pi^{*T}||_2 \leq \delta \\ P \text{ is reversible.} \\ \text{Other convex constraints on } P. \end{cases} \tag{3.4}$$

Unlike (P2), P is the only optimization variable in (P3). It is not difficult to see that (P3) is convex. One issue to consider is how to pick δ in the constraint $||\pi^{*T} P - \pi^{*T}||_2 \leq \delta$, so that the solution to (P3) indeed satisfies all the constraints in (P2). Specifically, we want to determine the bound on the value of δ to guarantee that the constraint $||\pi^* - \pi||_2 \leq \epsilon$ in problem (P2) is satisfied. We have the following proposition.

Proposition 2 *For any irreducible aperiodic reversible P, we have:*

$$||\pi^* - \pi||_2 \leq \frac{\pi_{max}^{1/2}}{\pi_{min}^{1/2}} \frac{||\pi^{*T} P - \pi^{*T}||_2}{1 - \lambda_2}. \tag{3.5}$$

□

Proof: See Appendix. ■

From Proposition 2, it is straightforward to see that if we pick $\delta \geq \epsilon \sqrt{\frac{\pi_{min}}{\pi_{max}}}(1 - \lambda_2)$, then $||\pi^* - \pi||_2 \leq \epsilon$. On the other hand, we cannot possibly know π_{min}, π_{max}, and λ_2 without knowing P first. However, one often can find some upper and lower bounds on these quantities by looking the structure of the class of the transition matrix. For example, one can bound λ_2 via the conductance obtained by examining the corresponding graph $G(V, E)$ [4]. Up to this point, the framework is applicable for a general class of reversible matrix. As mention before that tridiagonal Markov chain is reversible, we now lift the reversibility constraint by limiting the transition probabilities P to the class of tridiagonal matrix. Specifically, we have the following results on the upper and lower bounds of the quantities above for a class of tridiagonal transition probability matrices.

Proposition 3 *Let P be a tridiagonal matrix with $\alpha \leq P_{ij} \leq \beta; (0 < \alpha < \beta)$ for* *all* (i, j) *in the off-diagonal line, we have*

$$\begin{cases} \pi_{min} \geq \alpha^{|\Omega|-1} \\ \pi_{max} \leq \beta \\ \lambda_2 \leq 1 - 2\alpha^{|\Omega|} \end{cases}$$

□

Proof: See Appendix. ■

Using Proposition 3, the following corollary is obtained for selecting the right δ based on ϵ.

Corollary 2 *For the class of tridiagonal matrices defined in Proposition 3, pick* $\delta = \epsilon \frac{2\alpha^{(5|\Omega|-1)/2}}{\beta^{1/2}}$ *will guarantee that*

$$||\pi^* - \pi||_2 \leq \epsilon \tag{3.6}$$

We are ready to show the main result on bounding the optimal objective value of problem (P2) with that of problem (P3). We have the following proposition:

Proposition 4 *Let the* μ_2 *and* μ_3 *be the optimal objective values of problems (P2) and (P3), respectively. Let* $\Delta = \frac{\epsilon}{\sqrt{\pi^*_{min}}}$. π^*_{min} *and* π^*_{max} *denote the maximum and minimum entries in* π^**, respectively. Then,*

$$|\mu_2 - \mu_3| \leq C, \tag{3.7}$$

where

$$C = \frac{\Delta(2\sqrt{\pi^*_{min}} - \Delta)}{(\sqrt{\pi^*_{min}} - \Delta)^2} + (\sqrt{\pi^*_{max}} + 2\Delta)\frac{\Delta^2}{{\pi^*_{min}}^{3/2}} + |\Omega|\Delta(2\sqrt{\pi^*_{max}} + 3\Delta) \quad (3.8)$$

□

Proof: See Appendix. ■

Proposition 4 provides a bound on using solution to (P3) as an approximate solution for (P2). Therefore, we can use (P3) framework to obtain a solution matrix P whose stationary distribution is ϵ-close to stationary π, and has faster mixing time than that of (P1) framework.

Note that our framework is applicable to a variety of Markov environments whose dynamics can be modeled as a tridiagonal matrices. For a queuing system, the corresponding transition probability matrix is tridiagonal since the number of packets in the queue can only increase, decrease, or remain the same in the next time step. Similarly, the Birth-and-Death process has a tridiagonal transition matrix since the state can only decrease and increase by at most one in each step. In the next section, we will apply these two frameworks for a general tridiagonal Markov Chain case (Birth-and-Death process) and further extend these frameworks for queuing policies problem.

3.2 Algorithmic Solution to Proposed Framework

The proposed FMMC and EFMMC formulations are convex optimization problems in which there are various well-known methods to find the solutions. Since the differentiation of the objective function (the SLEM) is difficult to compute, here we introduce an algorithmic approach using projected subgradient method for maximum eigenvalue of a symmetric matrix.

Note that in the frameworks we want to optimize for the SLEM of a reversible matrix P, not the maximum eigenvalue of a symmetric matrix. However, we can easily convert the matrix P to matrix $A = D_\pi^{1/2} P D_\pi^{-1/2} - \sqrt{\pi}(\sqrt{\pi})^T$ where A is symmetric and $\lambda_{max}(A) = \mu(P)$ [4, Section 12.1].

Definition 5 (Subgradient of the SLEM) *A subgradient of λ_{max} at a symmetric matrix P is a symmetric matrix G that satisfies the inequality*

$$\lambda_{max}(\tilde{P}) \geq \lambda_{max}(P) + \mathbf{Tr}\ G(\tilde{P} - P) = \mu(P) + \sum_{i,j} G_{ij}(\tilde{P}_{ij} - P_{ij}) \tag{3.9}$$

for any symmetric matrix $\tilde{P}$. □

Proposition 5 (Subgradient via eigenvector) *Suppose P is a symmetric matrix and y is a unit eigenvector associated with $\lambda_{max}(P)$. Then the matrix $G = yy^T$ is a subgradient of $\lambda_{max}(P)$.*

□

Proof: See Appendix. ■

Now we give the following algorithm to optimize the SLEM of symmetric matrix P subject to constraints with subgradient method and projection at each step.

Projected subgradient method:

1: Given a feasible matrix P and begin the first step $k := 1$

2: Phase 1: Find eigenvector v of matrix P and compute subgradient G^k. Then let

$$\tilde{P} := P - \alpha_k G^k / ||G^k||$$

where the stepsize α_k satisfies the diminishing stepsize rule:

$$\alpha_k \geq 0,\ \alpha_k \to 0,\ \sum_k \alpha_k = \infty$$

3: Phase 2: Project $\tilde{P}$ into the feasible set by solving the following problem:

$$\text{minimize } ||P - \tilde{P}|| \text{ subject to constraints on } P \tag{3.10}$$

4: Phase 1 and phase 2 is repeated at each step k until the optimal solution is found (or we reach the stopping condition).

In (3.10), the constraints for matrix P is different for each problem as follows. For FMMC framework in (3.2):

$$\text{Constraints on P: } \begin{cases} P\mathbf{1} = \mathbf{1} \\ D_{\pi^*} P = P^T D_{\pi^*} \\ \text{other convex constraints on } P. \end{cases}$$

For EFMMC framework in (3.4):

$$\text{Constraints on P:} \begin{cases} P\mathbf{1} = \mathbf{1} \\ ||\pi^{*T}P - \pi^{*T}||_2 \leq \delta \\ P \text{ is reversible.} \\ \text{Other convex constraints on } P. \end{cases}$$

In fact, the subgradient method is not different from the original gradient method for differentiable function, meaning we start from a feasible point, move based on the subgradient that is computed at this point and repeat the process until we get to the optimal solution. However, it is should noted that unlike the gradient method, the subgradient does not guarantee the descent of the function during the whole process. Since then the convergence of the algorithm to the optimal point depends heavily on the choice of stepsize [6].

3.3 Application to Birth-and-Death Process

Birth-and-Death process [7] is a well-known Markov Chain using to model some population in which the size in one step can only increase and decrease by at most 1. Other speaking, there is at most one birth or one death in a time slot.

The transition probabilities can be specified by $\{p_i, r_i, q_i\}_{i=0}^{n}$ where:

- $p_i + r_i + q_i = 1$ for each i
- p_i is the probability of moving from state i to state $i+1$
- r_i is the probability of remain in the state i

- q_i is the probability of moving from state i to state $i-1$
- $q_0 = p_n = 0$ since there is no birth when the population reach its maximum value and no death when the population is zero

Suppose the maximum number of population is Ω, the transition probability matrix is formulated as follows.

$$P = \begin{pmatrix} r_0 & p_0 & & & \\ q_1 & r_1 & p_1 & & \\ & \ddots & \ddots & \ddots & \\ & & q_{|\Omega|-1} & r_{|\Omega|-1} & p_{|\Omega|-1} \\ & & & q_{|\Omega|} & r_{|\Omega|} \end{pmatrix} \tag{3.11}$$

To present the factors which can affect the birth rate and the death rate, we further require that: $r_i, p_i, q_i \in (\alpha, \beta), \forall i$. $(\alpha, \beta) \subset (0, 1)$ are pre-specified that models certain limitations.

Simulation 1 *Specifically, we set* $(\alpha, \beta) = (0.05, 0.95)$*, the maximum size* $|\Omega| = 20$*, and* $\delta = 0.1$*. The purpose is to design a birth-and-death chain which converges to a target stationary distribution* π^* *(shown in Fig. 3.1) in a fastest mixing time.*

□

First, in case (a), we consider a limited class of birth-and-death process where there is no remaining probability. These processes can be modeled as a tridiagonal matrix with the following requirement: $r_i = 0$ for $i = 2, \ldots, |\Omega| - 1$.

Given π^*, we solve problem (P1) to find the fastest policy that converges to π^*. Now in case (b), we enlarge the class of birth-and-death process by lifting the restriction on $r_i = 0$. Then, we solve problem (P1). Intuitively, the chain found in case (b) should likely to have faster convergence time than that of case (a) since it is found from a larger class of chains. Indeed, this is the case. Fig. 3.2 shows the total variation distance between the target stationary distribution and the current distribution as a function of time steps. As seen, the curve for case (a) decreases slower than that of case (b). At the time step $n = 300$, the total variation distance for case (b) is almost zero while that of (a) is still around 0.08.

We now consider case (c). In this case, the class of birth-and-death process is the same at that of case (b). However, we solve problem (P3) in which, we intentionally find a chain that might not produce exactly the target stationary distribution π^*, but close enough, i.e., $||\pi - \pi^*||_2 \leq \epsilon$. Intuitively, this chain should produce even faster adaptation than those of cases (a) and (b). In fact, this is the case. Fig. 3.2 shows the curve for case (c) which drops down quickly compared with the other two. At time $n = 50$, the total variation distance is 0.1284 for case (c) while they are more than 0.7 for the other two cases. The curve for case (c) however does not converge, i.e., decreases to zero, but stays around 0.12. This is intuitive since the solution to problem (P3) is not designed to obtain a chain that converges to the target stationary distribution. Fig. 3.1 shows that there are not much difference in the distribution π obtained by solving problem (P3) and the target distribution π^*. Thus, the problem requirements would not be violated by using the chain obtained from problem (P3).

We now study the trade-off between the accuracy of obtaining the target distribution and the convergence time. Fig. 3.3 shows the mixing times from problems (P3) and (P1) which decrease significantly when the allowable deviation (ϵ) from

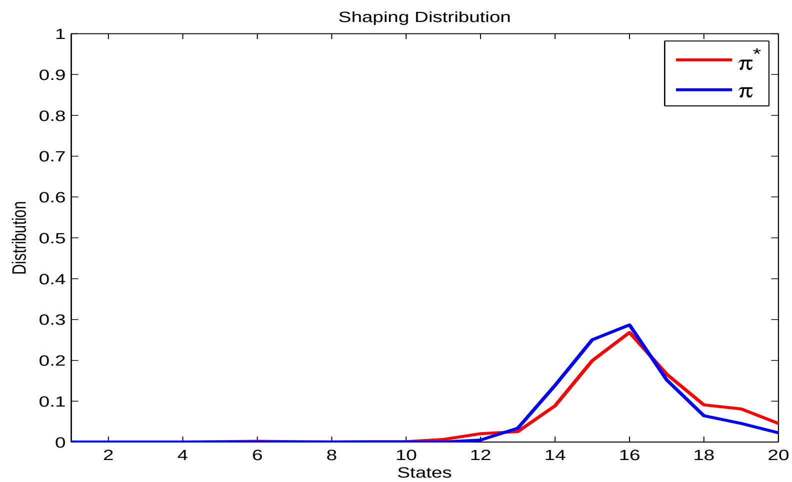


Figure 3.1: Target and resulted distributions in case (c)

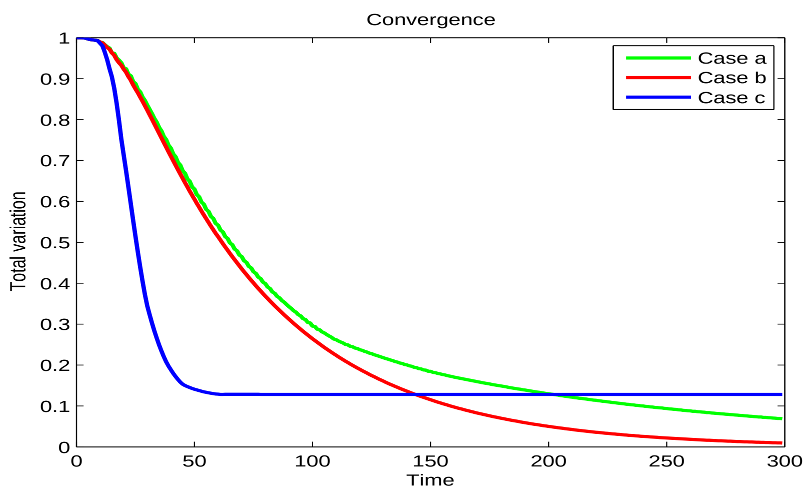


Figure 3.2: Comparison of the convergence times in 3 cases

the stationary distribution increases. For the class of chains in the simulation, setting $\epsilon = 0.02687$ seems to be the best as it reduces the mixing time significantly while keeping π close to π^*.

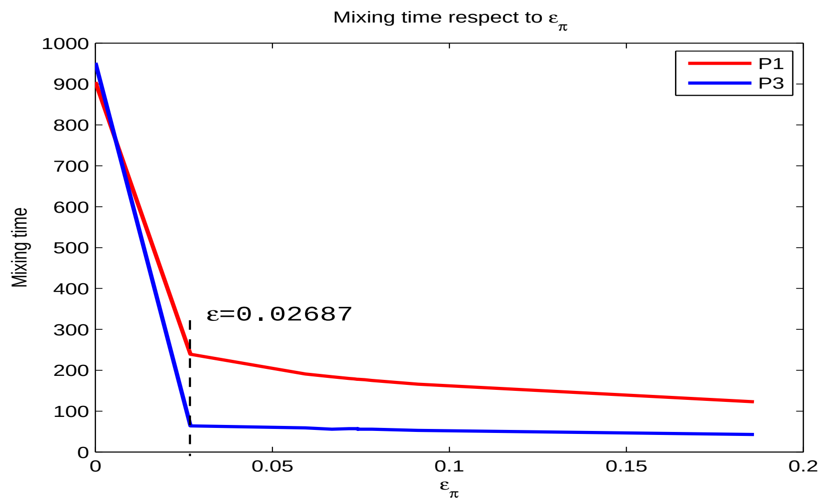


Figure 3.3: Mixing time in case (b) and case (c) with respect to ϵ_π

3.4 Finding Fast Queuing Policies Optimized For A Given Objective Function

3.4.1 Finding feasible queuing policy

Depending on specific settings, the tridiagonal transition probability matrix will not produce a valid queuing policy, more precisely, produce a feasible way for controlling the enqueue and dequeue rates. Let us consider the following scenario in which the arrival and departure rates at the queue can be controlled to some extent by a queuing policy. Let us assume that as a result of a queuing policy, the probabilities of a packet arriving at the queue and departing from the queue when the queue length is i, are a_i and s_i, respectively. We assume that packets can only arrive and depart at the beginning of each discrete time slot. We note that the ability to control the arrival rate seems impossible for physical queues in the Internet routers, however, it is frequently implemented in high level network protocols

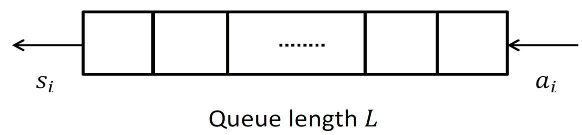


Figure 3.4: Discrete queue model

such as TCP in which virtual queues are typically used to provide feedback to the sender for the purpose of rate control. Using this queuing model as shown in Fig. 3.4, let us denote:

- $|\Omega|$: Maximum queue length
- $s = (s_0, \ldots, s_{|\Omega|})$ where $s_0 = 0$: Departing probability vector
- $a = (a_0, \ldots, a_{|\Omega|})$ where $a_{|\Omega|} = 0$: Arrival probability vector.

Then it is not difficult to see that the dynamics of the number of packets in a queue over time is governed by a discrete Markov chain with the transition probability matrix below:

$$Q = \begin{pmatrix} 1 - a_0 & a_0 & & \\ s_1(1-a_1) & 1 - s_1 - a_1 + 2s_1a_1 & (1-s_1)a_1 & \\ & \ddots & \ddots & \ddots \\ & & s_{|\Omega|} & 1 - s_{|\Omega|} \end{pmatrix} \tag{3.12}$$

Note that for each non-zero entry of each row, the left, middle, and right entries denote the probabilities that the number of packets in the queue decreases by 1, stays the same, or increases by 1, respectively.

Now, let us compare the above matrix Q to the matrix P which is the solution obtained from the problem (P1) or (P3) above. In general, P is a tridiagonal matrix with the entries: r_i, q_i, p_i.

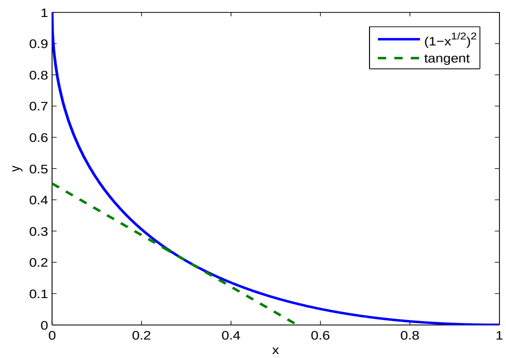


Figure 3.5: Tangent at $x_0 = 0.3$ of the function $f(x) = (1 - \sqrt{x})^2$

$$P = \begin{pmatrix} r_0 & p_0 & & & \\ q_1 & r_1 & p_1 & & \\ & \ddots & \ddots & \ddots & \\ & & q_{|\Omega|} & r_{|\Omega|} & \end{pmatrix} \tag{3.13}$$

The main challenge is how to find the corresponding s_i and a_i, i.e., enqueue and dequeue rates for given r_i, q_i, p_i. It turns out that s_i and a_i might be negative or complex numbers which cannot be used in a feasible queuing policy. However, we can determine the conditions on q_i and p_i for which there exist real and non-negative solutions for s_i and a_i, leading to a feasible queuing policy. We proceed to derive the conditions as follows.

From (3.12) and (3.13), we need to solve these following equations:

$$\begin{cases} s_i(1-a_i) = q_i \rightarrow a_i = 1 - q_i/s_i \\ (1-s_i)a_i = p_i \rightarrow a_i = p_i/(1-s_i) \end{cases}$$

$$\iff 1 - q_i/s_i = p_i/1 - s_i \text{ for } i = 1, \ldots, |\Omega| - 1$$

$$\iff (1 - s_i)s_i = (1 - s_i)q_i + s_i p_i \text{ for } i = 1, \dots, |\Omega| - 1$$

$$\iff s_i^2 - s_i(1 + q_i - p_i) + q_i = 0 \text{ for } i = 1, \dots, |\Omega| - 1 \tag{3.14}$$

Let us denote s_i' and s_i'' as two roots of (3.14), we have:

$$\begin{cases} s_i' + s_i'' = q_i \\ s_i' s_i'' = 1 + q_i - p_i \end{cases}$$

Since $q_i, p_i \in (0, 1)$ for $i = 1, \dots, \Omega - 1$, if (s_i', s_i'') are real, least one of s_i' or s_i'' will be in the range of $(0, 1)$ which satisfies the requirements for departing probability vector s and arrival probability vector a.

Hence, in order to guarantee the existence of feasible solution of (3.14), we need:

$$\Delta = (1 + q_i - p_i)^2 - 4q_i \geq 0 \text{ for } i = 1, \dots, |\Omega| - 1 \tag{3.15}$$

It appears that we can add these constraints directly to the two convex formulations above. However, these constraints are not convex, thus making it hard to solve in general. Therefore, our approach is to relax (3.15) by making it a convex constraint as follows.

$$(1 + q_i - p_i)^2 - 4q_i \geq 0 \iff 1 + q_i - p_i > 2\sqrt{q_i} \text{ since } q_i > 0$$

$$\iff (1 - \sqrt{q_i})^2 > p_i \tag{3.16}$$

Consider function $f = (1 - \sqrt{x})^2$ for $x \in (0, 1)$, we can find an approximate lower bound function $f(.)$ in the form of tangent $y = ax + b$ where $a = f'(x_0)$ and

$f'(x) = (\sqrt{x} - 1)/\sqrt{x}$ (See (Fig. 3.5)).

Hence, (3.16) is equivalent to the following convex constraints:

$$a(x_0)q_i + b(x_0) > p_i \text{ for } i = 1, \ldots, |\Omega| - 1 \tag{3.17}$$

Now, we can incorporate these constraints in (3.17) to the (P1) and (P3) problems and let us denote as FMMC framework and EFMMC framework respectively, then we still guarantee convex formulations to find feasible queuing policies.

3.4.2 Procedure for Optimizing a Given Objective via Queuing Policy

In this section, we provide an example of applying our proposed framework to find fast queuing policy that optimizes a given objective while still satisfying other standard QoS requirements. Our approach consists of two steps. In the first step, we find a stationary distribution π^* that optimizes a given objective subject to all the given constraints. Efficiently, the objective and the constraints can be formulated from the real-world conditions. In the second step, we substitute π^* into either the FMMC or EFMMC framework with the convex constraints in (3.17) to find the fastest queuing policy. We give a specific example below.

Step 1. Let X be discrete random variable representing the number of packets in the queue ($X \in [0, \ldots, L]$).

Suppose a video application requires that the queuing delay average and second moment must be bounded within a range. For example,

$$\begin{cases} E[X] < Y1 \\ E[X^2] < Y2 \end{cases}$$

Then $E[X]$ and $E[X^2]$ can be computed from the stationary distribution π:

$$\begin{cases} E[X] = \sum_{x=0}^{L} \pi(x)x \\ E[X^2] = \sum_{x=0}^{L} \pi(x)x^2 \end{cases}$$

Furthermore, suppose that there is a cost function $c(x)$ where x denotes the number of packets in the queue. $c(x)$ could be any function that might represent energy, resources that depends on the queue occupancy. Now, suppose we want to minimize the total expected cost:

$$T = \sum_{x=0}^{x=L} c(x)\pi(x).$$

Then the optimization problem can be formulated as follows.

$$\text{Minimize} \sum_{x=0}^{x=L} c(x)\pi(x)$$
$$\text{Subject to :} \begin{cases} \sum_{x=0}^{L} \pi(x)x < Y1 \\ \sum_{x=0}^{L} \pi(x)x^2 < Y2 \\ \sum_{x=0}^{L} \pi(x) = 1 \\ \pi_{min} < \pi(x) \ \forall x = 0, 1, \ldots, L \end{cases} \tag{3.18}$$

Step 2. The solution of (3.18), i.e., P gives us the target stationary distribution π^* satisfying the QoS requirements and the given objective. Now, we apply the FMMC and EFMMC formulations to find tridiagonal matrices with fast mixing rates. Next, using P and the method shown in Section 3.4.1, we can find the matrix Q, i.e., the dequeuing and dequeuing rates as a function of the number of packets in the queue. This will result in a queuing policy that achieves the target distribution quickly, yet satisfy the QoS requirements.

Chapter 4 – Performance Evaluation of Queuing Policies

In this section, we present the performance evaluations of our approach using the example above with specific parameters. We assume the maximum physical queue length $L = 9$ or total number of states is $|\Omega| = 10$. To demonstrate the flexibility of our approach, the cost function $c(x)$ is shown in Fig. 4.1 where Case 1: $\{Y1 = 15; Y2 = 50;\ \pi_{min} = 0.01\}$ and Case 2: $\{Y1 = 5; Y2 = 19;\ \pi_{min} = 0.01\}$;

Note that the cost function can be chosen arbitrarily depend on the factors we consider. Here, we only consider two very simple cases. In case 1, the cost can be viewed as the utility of the queue, so the cost is high when the queue is not in full and achieves the lowest value when the queue has no available slot left. Differently, the cost in the case 2 is most-affordable only when the queue is about half-occupied and slightly higher for the other states, which implies that we have to pay off to avoid the heavy traffic in the queue and also try to avoid the idle mode in the queue.

Using the approximation method for obtaining a feasible queuing policy in Section 3.4, we choose the tangent at $x_0 = 0.2$; we set $\delta = 0.001$ in the EFMMC framework. To show the robustness of the framework, we also consider one typical queuing policy in the feasible solution set of the FMMC framework and denote in the figures as 'Feasible' .

Fig. 4.2 shows the shape of the target stationary distribution π^* and π as the results of steps 1 and 2 in Section 3.4.2, respectively. As seen, π^* and π are very

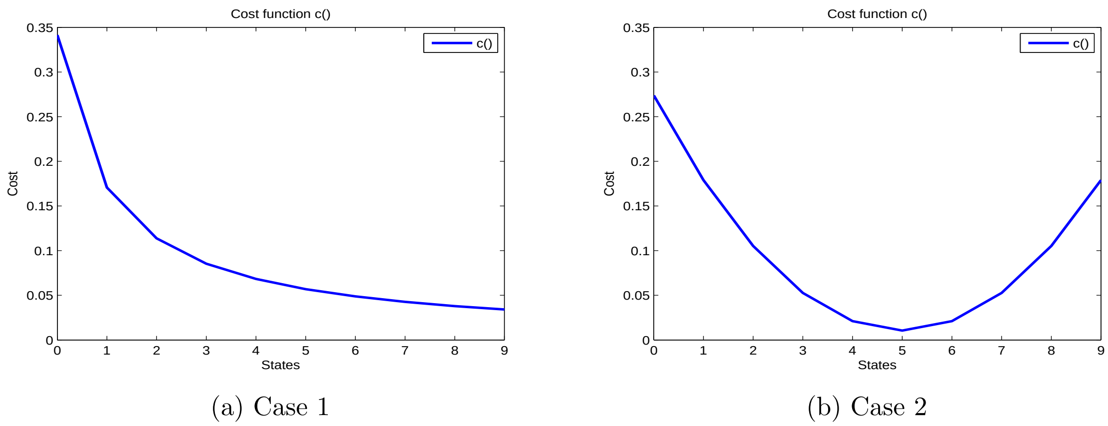


Figure 4.1: Cost function $c(x)$

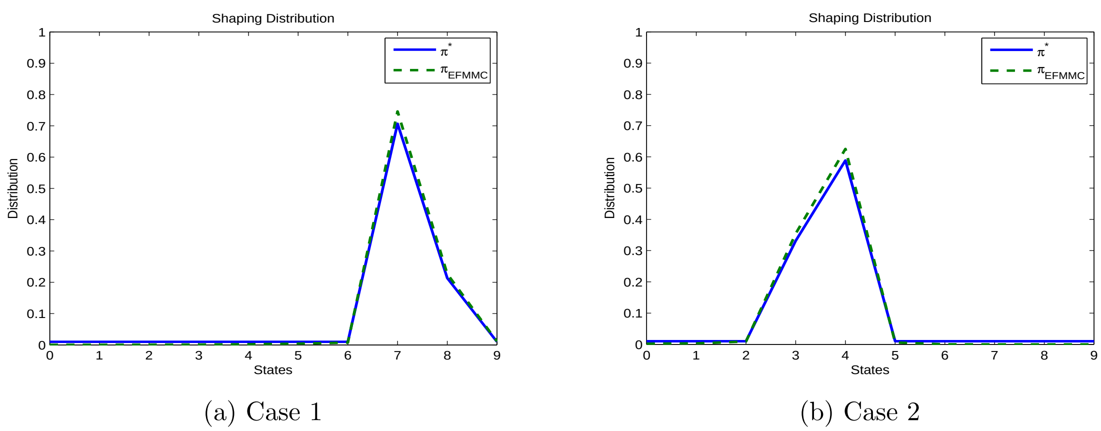
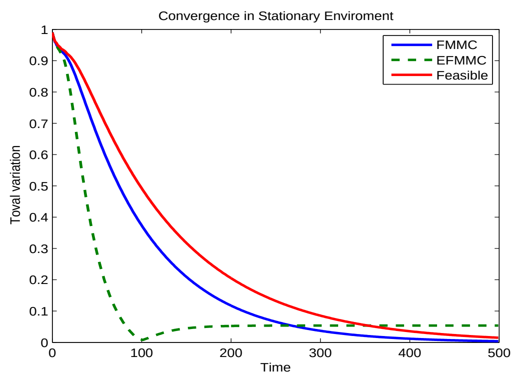


Figure 4.2: Target and resulted distribution

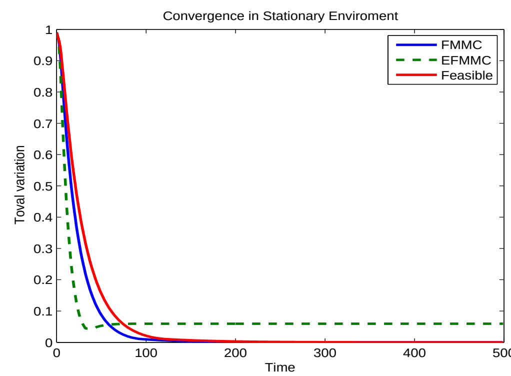
close indicating a very good approximation of our approach.

In addition, Fig. 4.3 shows that the EFMMC framework has a faster convergence rate than that of FMMC as expected while the Feasible queuing policy has the slowest convergence of all. *Importantly, a faster convergence rate is especially useful in non-stationary settings.*

To illustrate this point, Fig. 4.4 shows the total variance distance between the current distributions produced by the FMMC and EFMMC frameworks and the



(a) Case 1

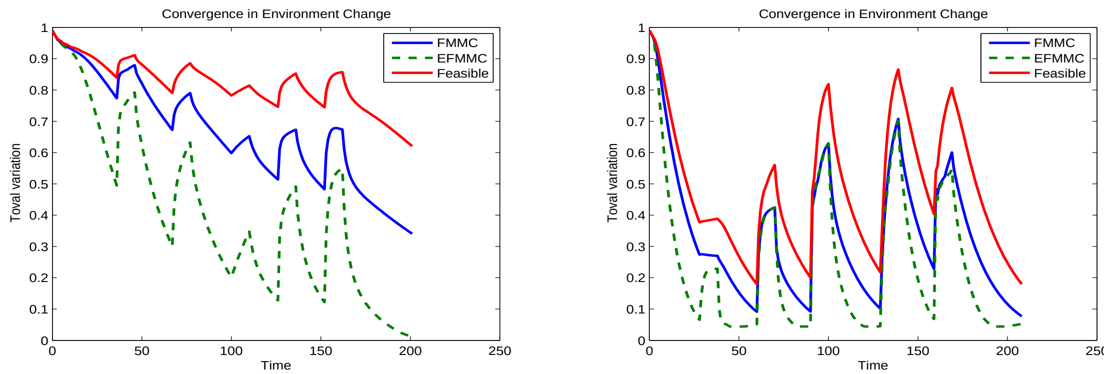


(b) Case 2

Figure 4.3: Comparison of the convergence times in two cases

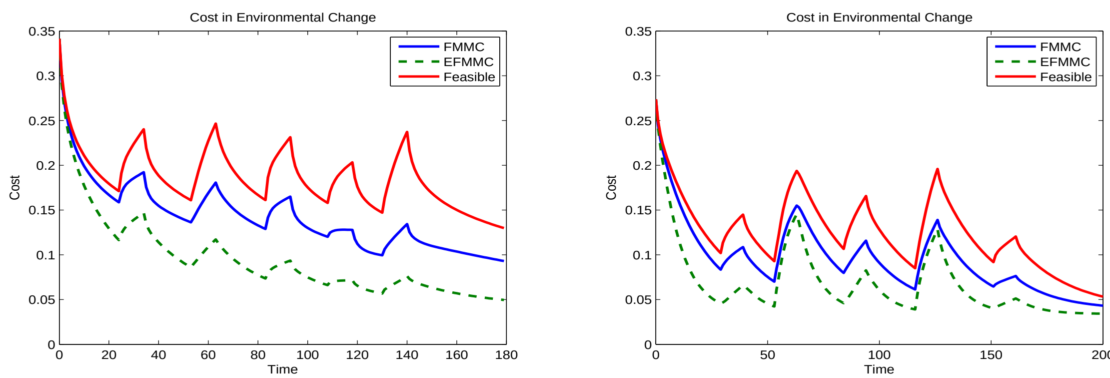
Feasible queuing policy, and the target stationary distribution in a non-stationary environment. The non-stationary environment is simulated based on the bursty traffic Poisson patterns with $\lambda = 30$. Specifically, in addition to the regular traffic, there are bursts of 5 packets arriving at the queue. On average, the time duration between these bursts are 30 time slots. As shown in Fig. 4.4, all three curves have spikes when the bursts of packets arrive. This prevents the current distributions in both frameworks from approaching the target stationary distribution (i.e, the curves approaching zero). On the other hand, the queuing policy based on EFMMC framework is better than that of FMMC since it produces as close as possible to the target distribution quickly.

Similarly, Fig. 4.5 shows the current cost of the systems by applying the Feasible queuing policies and also that of FMMC and EFMMC frameworks under the same non-stationary environment in the two cases. It can be seen the cost induced by EFMMC policy are the lowest of all and also it approaches the optimum cost in the fastest time. Hence, EFMMC policy is the most efficient policies in both



(a) Case 1 (b) Case 2

Figure 4.4: Convergence of the system during environmental change



(a) Case 1 (b) Case 2

Figure 4.5: Cost of the system during environmental change

cases.

Chapter 5 – Related Work

In this section, we discuss a number of related work on network protocols and queuing policies with an emphasis on protocols that aim to achieve QoS for multimedia flows.

Network protocols. There exists a vast literature on network protocol designs. Typically, network protocols are designed based on a few principles, and are optimized for specific situations. Wireless network protocols such as Wi-Fi protocols are completely different from the network protocols running on a wired network such as TCP in terms of operations as well as objectives. There are also protocols tailored for multimedia transmission applications [8] where emphasis is on achieving QoS and protocols for sensing applications with a focus on minimizing power consumption [9]. All of these protocols are typically designed to achieve some certain objectives. For example, TCP attempts to improve the bandwidth efficiency through congestion control and avoidance mechanisms. While many protocols are designed to respond quickly to changes in network conditions, they are often designed in heuristic ways. This thesis provides a flexible framework for designing network protocols that achieves a wide class of objectives, and formalizes the notion of a fast response protocol via the notion of mixing time.

Network protocols for multimedia traffic. Another aspect of network protocols/queuing policies design aims at satisfying a given of QoS as specified by certain multimedia applications such as audio/video interactive and streaming applications. The underlying principle for providing QoS under a limited resource setting is to treat packets differently based on their priorities. For example, packets of flows of different priorities are classified and marked at the ingress routers in the proposed DiffServ architecture [10]. The markings are then used by the intermediate routers to determine their forwarding/queuing policies. For example, packets with Expedited Forwarding (EF) marking are intended for flows/applications with low-loss, low-latency such as video conference traffic. The intermediate routers then implement certain queuing policies that ensure the EF packets have higher forwarding priority than other best effort packets. In a way, this is an attempt to provide scalable end-to-end QoS by enforcing differentiated service of flows on a per-hop behavior basis.

The same principle is also applied in local wireless area networks (WLAN). Specifically, using the MAC protocol 802.11e in the Enhanced Distributed Channel Access (EDCA) mode [11], packets are classified into different types: Background (AC_BK), Best Effort (AC_BE), Video (AC_VI), Voice (AC_VO). The minimum and maximum contention window (CWmin, CWmax) and Arbitration Inter-Frame Space (AIFS) are primary parameters to control the priorities for different packet types. A flow using small contention windows and AIFS will have higher chance to access the wireless medium. For example, CWmax for best-effort packets is set to 1023 while it is set to 32 for video packets.

Another approach to provisioning flows of different priorities is to employ multiple physical or virtual queues at a router. Each queue consists of packets of the same type. A queuing policy is used at each transmission opportunity, to decide which of the queues whose a packet should be transmitted. A simple fair queuing policy will transmit packets from each non-empty queue in a round robin fashion [12]. On the other hand, a priority or weighted queuing policy give preference for transmitting packets from higher priority queues [13].

Queuing policies. There is also a number of queuing policies related to our work, but are designed for different objectives. For example, queue can be implemented to give priority to small service requests in order to reduce the mean queue length [14]. In these types of policies, the optimal one is known as Shortest-Remaining-Processing-Time [15, 16], which shows a dramatic improvement in term of the mean response time [17], [18]. Besides, in cases where the coming traffic is unpredicted, they proposed the Foreground-Background technique [19], [20]. However, unlike our work, all the above techniques do not not analyze the convergence rate of the queuing policies which as discussed, can play a critical role in non-stationary environments.

Chapter 6 – Conclusion

In this paper, we introduce a novel approach of applying a convex optimization framework into networking applications for finding fast mixing policy that drives the system from any initial distribution to a target distribution subject to any constraints. We then show how to extend the proposed technique to find a policy that produces ϵ-approximation to the given distribution with even faster convergence time. Importantly, these two frameworks is used to design fast feasible queuing policies that provides statistical guarantees on QoS requirements as well as optimize arbitrary given objective. The former is useful in settings whose network conditions change slowly, while the later is appropriate for fast-changing network conditions. The analysis and simulation results verify the benefits of the proposed approach.

6.1 Future Work

The proposed framework can be applied into multiple networking and communication applications. One potential future work is to customize the framework for applying to the communication channel systems that can be modeled as a Markov Chain that is feasible for both FMMC and EFMMC frameworks. In a way, for each signal transmission, due to the feedback of bit rate error at receiver, the power can be adjusted at the sender in order to achieve better performance on the next transmission. The power will induce a corresponding cost function that also depends on the environment conditions. Hence, the framework can be used to optimize the cost function to find the target stationary distribution that implies the optimal state of the systems and also the fast policy that drives the system to this target state.

APPENDICES

Appendix A – Proofs of Propositions and Theorems

A.1 Proof of Proposition 1

Proposition 1 The proof can be found in [4, Proposition 2.8].

Proof: Obviously, any tridiagonal stochastic transition matrix corresponds to a Birth-and-Death process so we need to prove that a Birth-and-Death process is reversible.

A birth-and-death chain has state space $\Omega = \{0, 1, 2, \ldots, n\}$. In one step the state can increase or decrease by at most 1. The current state can be thought of as the size of some population; in a single step of the chain there can be at most one birth or death. The transition probabilities can be specified by $\{p_k, r_k, q_k\}_{i=0}^{n}$, where $p_k + r_k + q_k = 1$ for each k and

- p_k is the probability of moving from k to state $k+1$ when $0 \leq k < n$,
- r_k is the probability of moving from k to $k-1$ when $0 < k \leq n$,
- q_i is the probability of moving from state i to state $i-1$,
- $q_0 = p_n = 0$.

A function w on Ω satisfies the detailed balance equations (2.3) if and only if

$$p_{k-1}w_{k-1} = q_k w_k$$

for $1 \leq k \leq n$. For our birth-and-death chain, a solution is given by $w_0 = 1$ and

$$w_k = \prod_{i=1}^{k} \frac{p_{i-1}}{q_i}$$

for $1 \leq k \leq n$. Normalizing so that the sum is unity yields

$$\pi_k = \frac{w_k}{\sum_{j=0}^{n} w_j}$$

for $0 \leq k \leq n$. One can check that π is a probability on Ω and satisfies the detailed balance equations (2.3) so π is also a stationary distribution and the chain is reversible. ■

A.2 Proof of Theorem 1

Theorem 1 The proof can be found in [4, Theorem 12.3].

Proof: Let P be reversible with respect to π. By [4, Lemma 12.2], we have:

- The inner product space $(\mathbb{R}^{\Omega}, \langle \cdot, \cdot \rangle_{\pi})$ has an orthonormal basis of real-valued eigenfunctions $\{f_j\}_{j=1}^{|\Omega|}$ corresponding to real eigenvalues $\{\lambda_j\}$. and also
- The eigenfunction f_1 corresponding to the eigenvalue 1 can be taken to be the constant vector $\mathbf{1}$, in which case

$$\frac{P^t(x, y)}{\pi(y)} = 1 + \sum_{j=2}^{|\Omega|} f_j(x) f_j(y) \lambda_j^t \tag{A.1}$$

Using (A.1) and applying the Cauchy-Schwarz inequality yields

$$|\frac{P^t(x,y)}{\pi(y)} - 1| \leq \sum_{j=2}^{|\Omega|} |f_j(x)f_j(y)|\lambda_*^t \leq \lambda_*^t[\sum_{j=2}^{|\Omega|} f_j^2(x) \sum_{j=2}^{|\Omega|} f_j^2(y)]^{1/2} \tag{A.2}$$

where $\lambda_* = \mu(P)$.

Let δ_y be the function

$$\delta_y = \begin{cases} 1 & \text{if } y = x \\ 0 & \text{if } y \neq x \end{cases}$$

Hence, the function δ_y can be written via basis decomposition as

$$\delta_y = \sum_{j=1}^{|\Omega|} \langle \delta_y, f_j \rangle_\pi f_j = \sum_{j=1}^{|\Omega|} f_j(y)\pi(y)f_j \tag{A.3}$$

Using (A.3) and the orthonormality of $\{f_j\}$ shows that

$$\pi(x) = \langle \delta_x, \delta_x \rangle_\pi = \langle \sum_{j=1}^{|\Omega|} f_j(x)\pi(x)f_j, \sum_{j=1}^{|\Omega|} f_j(x)\pi(x)f_j \rangle_\pi = \pi(x)^2 \sum_{j=1}^{|\Omega|} f_j(x)^2$$

Consequently, $\sum_{j=2}^{|\Omega|} f_j(x)^2 \leq \pi(x)^{-1}$. This bound and (A.2) imply that

$$|\frac{P^t(x,y)}{\pi(y)} - 1| \leq \frac{\lambda_*^t}{\sqrt{\pi(x)\pi(y)}} \leq \frac{\lambda_*^t}{\pi_{min}} = \frac{(1-\gamma_*)^t}{\pi_{min}} \leq \frac{e^{-\gamma_* t}}{\pi_{min}}$$

where $\gamma_* = 1 - \lambda_*$ is called the absolute spectral gap.

Applying [4, Lemma 6.13] shows that $d(t) \leq \pi_{min}^{-1} \exp(-\gamma_* t)$ where $d(t) = \sup_{\nu \in \Omega} \|\nu^T P^t - \pi^T\|_{TV}$. The conclusion now follows from the definition of $t_{mix}(\epsilon)$. ■

A.3 Proof of Proposition 2

Proposition 2

Proof: We assume P has n eigenvalues $\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ and n left eigenvectors $\{v_1, v_2, \ldots, v_n\}$ such that: $1 = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq -1$.

Let $\langle f, g \rangle_{\frac{1}{\pi}} := \sum_{i \in \Omega} \frac{f(i)g(i)}{\pi(i)}$ denote the inner product with respect to $\pi(i)$. Due to the reversibility of P, it can be shown that the set of eigenvectors $\{v_i\}$ forms an orthonormal basis with $\langle ., . \rangle_{\frac{1}{\pi}}$. The eigenvector corresponds to the largest eigenvalue $\lambda_1 = 1$ is equal to the stationary distribution: $v_1 = \pi$. We have:

$$\pi^{*T} - \pi^T = \sum_{i=1}^{n} \langle \pi^* - \pi, v_i \rangle_{\frac{1}{\pi}} v_i^T$$

Since $v_i^T P = \lambda_i v_i^T$,

$$(\pi^{*T} - \pi^T)(P - I) = \sum_{i=1}^{n} (\lambda_i - 1) \langle \pi^* - \pi, v_i \rangle_{\frac{1}{\pi}} v_i^T$$

Also,

$$\begin{aligned}
\langle \pi^* - \pi, v_1 \rangle_{\frac{1}{\pi}} &= \langle \pi^* - \pi, \pi \rangle_{\frac{1}{\pi}} \\
&= \sum_{i=1}^{n} (\pi^*(i) - \pi(i)) \\
&= 0
\end{aligned}$$

Then

$$||\pi^{*T} - \pi^T||_{\frac{1}{\pi}} = ||\pi^* - \pi||_{\frac{1}{\pi}} = \sqrt{\sum_{i=2}^{n} \langle \pi^* - \pi, v_i \rangle_{\frac{1}{\pi}}^2}$$

and

$$||(\pi^{*T} - \pi^T)(P - I)||_{\frac{1}{\pi}} = \sqrt{\sum_{i=2}^{n} (\lambda_i - 1)^2 \langle \pi^* - \pi, v_i \rangle^2_{\frac{1}{\pi}}}$$

Therefore:

$$||(\pi^{*T} - \pi^T)(P - I)||_{\frac{1}{\pi}} \geq \min_{i=2,\ldots,n} |1 - \lambda_i| ||(\pi^* - \pi)||_{\frac{1}{\pi}}$$

$$\rightarrow ||(\pi^{*T} - \pi^T)(P - I)||_{\frac{1}{\pi}} \geq (1 - \lambda_2) ||(\pi^* - \pi)||_{\frac{1}{\pi}}$$

$$\rightarrow ||(\pi^{*T} P - \pi^{*T})||_{\frac{1}{\pi}} \geq (1 - \lambda_2) ||(\pi^* - \pi)||_{\frac{1}{\pi}}$$

Since for any vector x:

$$\frac{||x||_2}{\sqrt{\pi_{min}}} \geq ||x||_{\frac{1}{\pi}} \geq \frac{||x||_2}{\sqrt{\pi_{max}}}$$

Then we conclude:

$$||\pi^* - \pi||_2 \leq \frac{\pi_{max}^{1/2}}{\pi_{min}^{1/2}} \frac{||\pi^{*T} P - \pi^{*T}||_2}{1 - \lambda_2}$$

■

A.4 Proof of Proposition 3

Proposition 3

Proof: Since $\pi^T P = \pi^T$, for any $1 \leq k \leq n$ we have:

$$\pi_k = \sum_i \pi_i P_{i,k}$$

Since $P_{i,k} = 0$ for $|k - i| > 1$,

$$\pi_k = \pi_{k-1} P_{k-1,k} + \pi_k P_{k,k} + \pi_{k+1} P_{k,k+1}$$

where $k = 1$ or $k = n$ we consider $P_{1,0} = 0$ and $P_{n,n+1} = 0$.

Hence for any k,

$$\pi_k < \pi_{k-1} \max P_{k-1,k} + \pi_k \max P_{k,k} + \pi_k + 1 \max P_{k,k+1}$$

$$\rightarrow \pi_k < (\pi_{k-1} + \pi_k + \pi_{k+1})\beta < \beta \text{ since } \pi_{k-1} + \pi_k + \pi_{k+1} < 1 \tag{A.4}$$

$$\rightarrow \pi_{max} < \beta \tag{A.5}$$

We see that P has the form:

$$P = \begin{pmatrix} P_{1,1} & P_{1,2} & & & \\ P_{2,1} & P_{2,2} & P_{2,3} & & \\ & \ddots & \ddots & \ddots & \\ & & P_{n-1,n-2} & P_{n-1,n-1} & P_{n-1,n} \\ & & & P_{n,n-1} & P_{n,n} \end{pmatrix} \tag{A.6}$$

and P^2 has the form:

$$P^2 = \begin{pmatrix} P^2_{1,1} & P^2_{1,2} & P^2_{1,3} & & & \\ P^2_{2,1} & P^2_{2,2} & P^2_{2,3} & P^2_{2,4} & & \\ & \ddots & \ddots & \ddots & \ddots & \\ & & P^2_{n-1,n-3} & P^2_{n-1,n-2} & P^2_{n-1,n-1} & P^2_{n-1,n} \\ & & & P^2_{n,n-2} & P^2_{n,n-1} & P^2_{n,n} \end{pmatrix} \tag{A.7}$$

where $P^2_{i,j}$ are entries of P^2. We see that the non-zero entries of P^2 has enlarged to one in each row compare to P and these entries has minimum value equal α^2. By induction, P^{n-1} would have no zero entries and the minimum entry value of

α^{n-1}. Since $\pi^T P^{n-1} = \pi^T$, for any $1 \leq k \leq n$ we have:

$$\pi_k = \sum_i \pi_i P_{i,k}^{n-1}$$

where $P_{i,j}^{n-1}$ denote the entry of row i and column j of matrix P^{n-1} Hence,

$$\pi_k \geq \sum_i \pi_i \min P_{i,k}^{n-1} = \sum_i \pi_i \alpha^{n-1} = \alpha^{n-1}$$

$$\rightarrow \pi_{min} \geq \alpha^{n-1} \tag{A.8}$$

Let $Q(S, S^C) = \sum_{i \in S; j \in S^C} \pi(i) P_{ij}$ for any subset S in state space and S_c is compliment set of S. By definition of Conductance [4], we have:

$$\Phi_* = min\{\Phi_S : S \in \Omega; \pi(S) \leq 1/2\}$$

where $\Phi_S = \frac{Q(S,S_c)}{\pi(S)}$ for any subset S of state space and $\pi(S) = \sum_{i \in S} \pi(i)$.
Now, let have a lower bound on $Q(S, S^C)$ for any subset S:

$$Q(S, S^C) = \sum_{i \in S; j \in S^C} \pi(i) P_{ij} \geq \pi_{min} \min P_{ij} \geq \alpha^{n-1} \alpha = \alpha^n$$

Hence, Conductance $\Phi_* \geq \frac{\pi_{min}.\alpha}{1/2} = 2\alpha^n$
Also, for a reversible Markov chain, let $\gamma = 1 - \lambda_2$ then $\frac{\Phi_*^2}{2} \leq \gamma \leq 2\Phi_*$ where Φ_* is Conductance (Bottleneck ratio revisited) of the chain [4].
Therefore,

$$\frac{\Phi_*^2}{2} \leq \gamma \rightarrow \gamma \geq 2\alpha^{2n}$$

$$\rightarrow \lambda_2 = 1 - \gamma \leq 1 - 2\alpha^{2n} \tag{A.9}$$

∎

A.5 Proof of Proposition 4

Proposition 4

Proof:

Denote a vector $s = \sqrt{\pi^*} - \sqrt{\pi}$ then $|s_i| \leq \Delta\ \forall i \in \Omega$ where $\Delta = \frac{\epsilon_\pi}{\sqrt{\pi^*_{min}}} > 0$

Using Taylor series for function $f(x) = \frac{1}{c+x}$ at point $x = 0$ in the interval $x \in (-\Delta, \Delta)$, we have:

$$\frac{1}{\sqrt{\pi_i^*}} = \frac{1}{\sqrt{\pi_i} + s_i} = \frac{1}{\sqrt{\pi_i}} - \frac{1}{\pi_i} s_i + R_i$$

where R_i is the Taylor Remainder then $|R_i| \leq \frac{1}{{\pi^*_{min}}^{3/2}} \Delta^2$.
Denote R is a vector whose entries are R_i then

$$\begin{cases} D_{\pi^*}^{1/2} = D_\pi^{1/2} + D_s \\ D_{\pi^*}^{-1/2} = D_\pi^{-1/2} - D_{s/\pi} + D_R \end{cases}$$

We also denote:

$$\begin{cases} A = D_{\pi^*}^{1/2} P D_{\pi^*}^{-1/2} - \sqrt{\pi^*}(\sqrt{\pi^*})^T \rightarrow \mu_3 = ||A||_2 \\ B = D_\pi^{1/2} P D_\pi^{-1/2} - \sqrt{\pi}(\sqrt{\pi})^T \rightarrow \mu_2 = ||B||_2 \end{cases} \quad \text{(A.10)}$$

Then we have:

$$\begin{aligned} A &= (D_\pi^{1/2} + D_s)P(D_\pi^{-1/2} - D_{s/\pi} + D_R) \\ &\quad - (\sqrt{\pi} + s)(\sqrt{\pi} + s)^T \\ &= B + D_s P D_\pi^{-1/2} - D_\pi^{1/2} P D_{s/\pi} \\ &\quad - D_s P D_{s/\pi} + D_\pi^{1/2} P D_R + D_s P D_R \\ &\quad - s(\sqrt{\pi})^T - \sqrt{\pi} s^T - s s^T \end{aligned} \tag{A.11}$$

Since $||P|| = 1$, using sub-multiplicative property of matrix norm each element in the right side of (A.11) (except B) can be bound as following:

$$\begin{cases} ||D_s P D_\pi^{-1/2}|| \le max_i|\frac{s_i}{\sqrt{\pi_i}}| = \frac{\Delta}{\sqrt{\pi^*_{min}} - \Delta} \\ ||D_\pi^{1/2} P D_{s/\pi}|| \le max_i|\frac{s_i}{\sqrt{\pi_i}}| = \frac{\Delta}{\sqrt{\pi^*_{min}} - \Delta} \\ ||D_s P D_{s/\pi}|| \le max_i|\frac{s_i^2}{\pi_i}| = \frac{\Delta^2}{(\sqrt{\pi^*_{min}} - \Delta)^2} \\ ||D_\pi^{1/2} P D_R|| \le max_i|\sqrt{\pi_i}||R_i| = (\sqrt{\pi^*_{max}} + \Delta)\frac{\Delta^2}{{\pi^*_{min}}^{3/2}} \\ ||D_s P D_R|| \le \max_i |s_i||R_i| = \frac{\Delta^3}{{\pi^*_{min}}^{3/2}} \\ ||s(\sqrt{\pi})^T|| \le |\Omega| \max_i |s_i(\sqrt{\pi^*_i} - s_i)| = |\Omega|\delta(\sqrt{\pi^*_{max}} + \Delta) \\ ||(\sqrt{\pi})s^T|| \le |\Omega| \max_i |s_i(\sqrt{\pi^*_i} - s_i)| = |\Omega|\delta(\sqrt{\pi^*_{max}} + \Delta) \\ ||ss^T|| \le |\Omega| \max_i |s_i^2| = |\Omega|\Delta^2 \end{cases}$$

Sum up all these elements, we now have:

$$\begin{aligned} ||A - B|| \le C &= \frac{\Delta(2\sqrt{\pi^*_{min}} - \Delta)}{(\sqrt{\pi^*_{min}} - \Delta)^2} + (\sqrt{\pi^*_{max}} + 2\Delta)\frac{\Delta^2}{{\pi^*_{min}}^{3/2}} \\ &\quad + |\Omega|\Delta(2\sqrt{\pi^*_{max}} + 3\Delta) \end{aligned} \tag{A.12}$$

Also,

$$|(\min ||A|| - \min ||B||)| \leq \max ||A - B||$$

From (A.10), we have:

$$|\mu_3 - \mu_2| \leq C \tag{A.13}$$

■

A.6 Proof of Proposition 5

Proposition 5 The proof can be found in [5, Section 5.1]

Proof: Since P is symmetric and y is a unit eigenvector associated with $\lambda_{max}(P)$, we have

$$\begin{aligned} \lambda_{max}(P) &= y^T P y \\ \lambda_{max}(\tilde{P}) &\geq y^T \tilde{P} y \end{aligned}$$

From two above equations, we have the desired inequality

$$\lambda_{max}(\tilde{P}) \geq \lambda_{max}(\tilde{P}) + y^T(\tilde{P} - P)y = \lambda_{max}(P) + \sum_{i,j} y_i y_j (\tilde{P}_{ij} - P_{ij})$$

Hence $G = yy^T$ is a subgradient of P. ■

Bibliography

[1] M. Guizani, A. Rayes, B. Khan, and A. Al-Fuqaha, *Network Modeling and Simulation: A Practical Perspective.* John Wiley and Sons, Feb 2010.

[2] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris, *Fundamentals of Queueing Theory.* John Wiley and Sons, 1961.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2009.

[4] D. A.Levin, Y. Peres, and E. L.Wilmer, *Markov Chains and Mixing Times.* American Mathematical Society, 2008.

[5] S. Boyd, P. Diaconis, and L. Xiao, "Fastest mixing markov chain on a graph," *SIAM Review*, vol. 46(4), pp. 667–689, December 2004.

[6] S. Boyd, L. Xiao, and A. Mutapcic, "Subgradient methods," 2003, notes for EE392o, Stanford University.

[7] T. L. Saaty, *Elements of queueing theory: with applications.* McGraw-Hill, 1961.

[8] S. Floyd, J. Padhye, and J. Widmer, "TCP friendly rate control (TFRC): protocol specification," http://tools.ietf.org/html/rfc5348. [Online]. Available: http://tools.ietf.org/html/rfc5348

[9] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *IEEE INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, vol. 3, 2002, pp. 1567–1576 vol.3.

[10] S. Blake, D. Black, M. Carson, E. Davis, Z. Wang, and W. Weiss, "An architecture for differentiated services," in *RFC2475*, December 1998.

[11] I. S. 802.11e/D6.0, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements*, 2003.

[12] J. Nagle, "On packet switches with infinite storage," *Communications, IEEE Transactions on*, vol. 35, no. 4, pp. 435 – 438, apr 1987.

[13] D. Stiliadis and A. Varma, "Latency-rate servers: a general model for analysis of traffic scheduling algorithms," *Networking, IEEE/ACM Transactions on*, vol. 6, no. 5, pp. 611 –624, oct 1998.

[14] M. Harchol-Balter, B. Schroeder, N. Bansal, and M. Agrawal, "Size-based scheduling to improve web performance," *ACM Transactions on Computer Systems (TOCS)*, pp. 207–233, 2003.

[15] L. Schrage, "The queue m/g/1 with feedback to lower priority queues," *Management Science*, pp. 466–474, 1967.

[16] ——, "A proof of the optimality of the shortest remaining service time discipline," *Queueing Systems*, pp. 670–690, 1968.

[17] M. Crovella and A. Bestavros, "Self-similarity in world wide web trafic: evidence and possible causes," *Proceedings ACM Sigmetrics, Philadelphia, USA*, pp. 160–169, 1996.

[18] M. Taqqu, W. Willinger, and A. Erramilli, *A bibliographical guide to self-similar trafic and performance modeling for modern high-speed networks.* Oxford University Press, 1996.

[19] I. Rai, E. Biersack, and G. Urvoy-Keller, "Las scheduling approach to avoid bandwidth hogging in heterogeneous tcp networks," in *7th IEEE International Conference on High Speed Networks and Multimedia Communications, Toulouse, France*, 2004, pp. 179–190.

[20] ——, "Size-based scheduling to improve the performance of short tcp flows," *IEEE Network Magazine*, pp. 12–17, 2005.