Northwestern University

Utility-Based Downlink Scheduling of Wireless Networks with Hybrid ARQ

A Project

Submitted to the Graduate School
in partial fulfillment of the requirements

for the degree

Master of Science

Field of Electrical Engineering

By

Jianwei Huang

Evanston, Illinois

May 9, 2003

ABSTRACT

We consider scheduling problems for downlink data transmission in wireless networks. There are multiple users in the system and each user is assigned a utility function that depends on queue length at the base station. Due to the burst error nature of the wireless channel, retransmissions sometimes are necessary for providing reliable communications. Here we study type II hybrid ARQ scheme in which the receiver jointly decodes several packets received from different retransmissions. Thus the decoding probability changes according to the number of transmissions. We seek an optimal scheduling policy, which maximizes the total utility or long-run average total utility at the base station, depending on the nature of the arrivals, while take into account retransmissions.

We consider two special cases cases in this report. First, we consider the case with linear utility functions and Poisson arrival processes. We formulate the problem in the framework of the classic scheduling model due to Klimov and find that the optimal scheduling rule is fixed priority rule. We then give a simple algorithm to calculate the priority orders of different packets. Next we consider the case with decreasing concave utility functions and batch arrivals. We show this can also be viewed as a variation of a Klimov problem. In that case, we show that a packet is transmitted until it is successfully decoded. We then formulate the problem as Markov Decision Process. This allows us to show that the optimal scheduling rule exhibits a switching curve, which depends on the queue lengths of the different users. Simulation results are presented for both of the preceding cases, and a sensitivity analysis of the optimal scheduling policy with respect to channel and utility variations is presented. Finally the scheduling problem is generalized to account for arbitrary arrival processes. This leads to a formulation as a restless bandit problem. We conclude by outlining future work.

ACKNOWLEDGMENTS

I would like to thank my advisors, Professor Michael Honig and Professor Randall Berry, for their guidance and inspiration for this work. This project can not be done without their advice, suggestions and patience, and I feel very lucky to be student of both of them.

I am very thankful to my colleagues in communications lab. They are Yakun Sun, Wiroonsak Santipach, Naveen Arulselvan, Peijuan Liu, Yishen Sun and Xiangping Qin. When I have questions, they are always there ready to help.

Finally, I would like to thank my parents, my brother and my fiancee Biying Shou, for their unconditional love and support. I can not image achieving anything significant in my life without them.

Contents

List of Figures

Chapter 1

Introduction

Stochastic scheduling is a very important topic for a fast growing number of communication applications. Many local area networks, such as the IEEE 802.4 token bus and the IEEE 802.5 token ring (Chapters 4.5.3-4.5.4 in [1]), can be modelled as a polling system, which is a scheduling model where a single server is responsible for scheduling multiple queues. Another important application is transmission of packets at the output of a packet switch in a wide area network. In this report we will focus on scheduling data packets for the downlink channel (from the base station to mobiles) in a wireless network.

In the system we consider, there are N mobile users and the packets for different users arrive and accumulate in different queues at the base station. Packet wait to be transmitted one at a time. With each user is associated a utility function that depends on that user's queue length. Due to the burst error nature of the wireless channel, retransmissions sometimes are necessary for providing reliable communications. We consider a system using using type II hybrid Automatic-repeat-request(ARQ) scheme in which the receiver jointly decodes several packets received from different retransmissions. Thus the decoding probability changes according to number of transmissions. We seek an optimal scheduling policy, which maximizes the total utility or long-run average total utility at the base station, depending on the nature of the arrivals, while takeing into account retransmissions.

This chapter is organized as follows: we will first introduce some background on stochastic scheduling related to our problem. Subsequently, we will discuss the characteristics of wireless channels and our approach to address the associated challenges. Then we will give an overview of this report.

1.1 Stochastic Scheduling

Stochastic scheduling problems arise when we consider a system with a set of jobs having some random features (such as arrival patterns, service requirements, etc) waiting to be processed. The system does not have enough service facilities to process these jobs simultaneously. Our goal is to find an optimal policy (or rule of service) that (1) schedules the jobs waiting in the system, and (2) maximizes the system performance (for example, the total reward from processing the jobs), minimizes the system cost (such as the waiting time cost and processing cost).

Depending on the number of the servers in the system, stochastic scheduling models can be classified as single-server and multi-server systems. If we differentiate systems by the availability of the jobs to be processed, then there are two main categories: (1) models with jobs available at the beginning of the processing with no new arrivals, and (2) models with jobs arriving at random time during the process. Here we consider only a single-server system. For a more comprehensive review on stochastic scheduling, see [4].

The following definitions pertaining to scheduling policies will be useful: *Nonidling* means that the server will not idle as long as the system is not empty. *Nonpreemptive* means the processing of a job will not be interrupted before it is completed. *Preemptive* means that the processing of a job can be interrupted before it is completed and the process can be resumed later. *Nonanticipative* means scheduling decisions are not based on the future information (in other word decisions are causal).

The major policies that have been well studied are *fixed-priority* and *alternating-priority*. Under a *fixed-priority* policy, at each decision epoch, the server will find the job in the system with the highest priority to serve, and switch to another job (again using the same policy) at service completion (under nonpreemptive policy) or when a new job with higher priority enters the system (under preemptive policy). The assignment of priority is usually done through an index calculation: an index is computed for each job type (possible depending on its current state, but not on that of others), and at each decision epoch jobs of higher index are assigned higher service priority. Thus *fixed-priority* policy is sometimes also called *priority-index* rule. Under *alternating-priority* policy, each queue in its turn has the highest priority. A typical rule of this class is called *cyclic polling* rule, that is, at beginning of a cycle, the server decides the order of serving all the queues of the systems, visits them in this order, then decides a new order of visiting for the next cycle and follows it. For the case that the order is fixed for every cycle, it is called a Round-Robin policy. Two common rules, which determine how long to stay in each queue are *exhaustive* and *gated* policies. Under an *exhaustive* policy, the server will not move to a next queue until all the packets in the current queues have been served. Under a *gated* policy, the sever only serves those packets in a queue that are already there when the server begins to serve the queue. In other words, the packets arriving during the service of the queue are not considered.

1.1.1 Scheduling for Systems with No Random Arrivals

The simplest case in this class of models is *scheduling for a batch of stochastic jobs.* In this problem there are m jobs with random service times to be scheduled on one server. An important application for this model is the decision process of a hospital for admitting disaster victims into its surgery room.

The pioneering work on this model was done by Smith [6], in which the Shortest Weighted Processing Time rule was shown to be the optimal policy in the deterministic setting for minimization of the expected weighted flowtime (waiting time in the queue plus service time) with a single server. Rothkopt [8] gives the optimal scheduling rule for single server scheduling of tasks with continuous discounted linear waiting costs, and also proposes a dynamic programming algorithm for some generalized cases. Stochastic problems with nonpreemptive and preemptive policies have been studied, respectively, by Rothkopt [7] and Sevcik [5], and optimal policies are again characterized as priority-index rules.

If we allow jobs to stay in the system after they are processed, and have the freedom to change their attributes after each service epoch, then this corresponds to the *restless bandit problem*, which is formulated as follows [10]: consider N random processes (or called bandit process), where each process is a semi-Markov process with a finite state space. At each discrete time epoch $t = 0, 1, ...$, a subset of M bandit processes are selected to be "active" and all the others are "passive". During each time epoch, an active (respectively passive) reward is collected from each active (respectively passive) bandit process depending on its current state. The total reward is discounted in time by a factor β between 0 and 1. Meanwhile the state of each active (respectively passive) bandit process changes its state in a Markovian fashion. At the next discrete-time epoch we are free to choose a subset of M bandit processes to be "active" and all others to be "passive". The goal is to find a scheduling policy to maximize the total expected discounted reward over an infinite horizon, and compute its optimal value.

A special case of the restless bandit problem is called the *multi-armed bandit model*, in which we let $M = 1$, and prevent the passive processes from changing states. The optimal scheduling policy is given by the well known Gittins priority-index rule [11]: an index γ_j is computed (in a finite number of steps) for each bandit process state j, and the rule selects at each time a process with the highest current index. There have been numerous papers showing the optimality of Gittins rules using different techniques (see [9] and the citations therein).

For the general restless bandit problem, finding optimal rule is an open problem. Linear programming relaxation/partial conservation laws have been used to characterize some heuristic indexing rules and their performance bounds. See [10] for related work on this.

1.1.2 Scheduling for Systems with Random Arrivals - Queueing Control Models

For systems with random arrivals, the arrival process will affect the optimal policy. In this report, we focus our attention on a special queueing control model called a *polling system*, in which one server is responsible for several parallel queues, and there are possible switching or setting time costs when the server moves from one queue to the other.

One of the simplest models in this category is a polling system with N parallel infinite-buffer queues. The goal is to minimize the average weighted sum of queue lengths. The weight factor of each queue is some positive constant c_i , $i = 1, ..., N$. The service time of each queue is exponentially distributed with parameter μ_i. Cox and Smith [12] showed that the optimal scheduling rule is the $c\mu$ rule, that is, the priorities of the queues are ordered by the values $c_i\mu_i$, $i = 1, 2, ..., N$. Many extensions of this model have been studied, and $c\mu$ or similar rules have been proven to be optimal. See Nain [14] for a continuous-time, discounted cost and partial feedback, and De Serres [15] for situations when scheduling and flow control are optimized simultaneously.

An important extension of the $c\mu$ rule was introduced by Klimov [16] for multiclass $M/G/1$ systems with feedback, and serves as the basis for a large part of this research. We will give a detailed description of this problem in Chapter 2. Tcha and Pliska [17] studied a similar problem as Klimov with extensions to the discounted case with preemptive/nonpreemtive service disciplines.

Most of the scheduling literature developed in the context of manufacturing has assumed that every job is successfully processed once it is scheduled. For the communications applications, this is more appropriate for wireline communications with reliable channel conditions. In the context of wireless communications, this assumption is not generally valid due to the special features of the wireless channels. In the next section we will explain the difficulties of scheduling in the wireless setting and review the related literature. We will also introduce the contribution of our work.

1.2 Scheduling in Wireless Networks

The impairments of the wireless channel introduces many new challenges to the scheduling problem. First, the channel may cause burst errors during which a packet cannot be transmitted successfully, so that retransmission is necessary. Second, the channels are location-dependent, thus the same shared channel may be asymmetric, that is, has different qualities for different mobile users. This raises an issue of fairness,if the system only transmits to users in good channels, thus lend to unfair treatment of users with poor channels.

1.2.1 Channel Errors and Retransmissions

The effect of channel errors on scheduling has been addressed by several authors in different ways. Tassiulas and Ephremides in [23] modelled the channel as "connected" or "disconnected" to each user according to a binomial random variable. They then studied stability and delay properties in a polling system with N queues. Altman and Kushner in [24] studied a similar system in the context of queueing with vacations, and characterized the system performance in the heavy traffic regime where the server has little idle time under average system load. Buche and Kushner in [27] consider a power scheduling problem with various assumptions for downlink wireless transmission and again worked in the heavy traffic regime. The heavy traffic approach leads to state space collapse, and therefore is very useful in studying complex queueing control systems. Other examples of work in this area can be found in [25] [28] [29] and [24].

When considering retransmissions, almost all authors assume that if a retransmission is needed, the entire original packet is transmitted and is independently decoded from the previously received version [23] [24] [25]. This is true if the system uses standard ARQ as the retransmission scheme, in which the transmitter resends the original packet when an error occurs. It is also true with a type I hybrid ARQ scheme, in which the transmitter resends the original packet if the receiver can not correct the error within its designed error-correcting capability. But this independently decoding assumption is not valid in a type II hybrid ARQ scheme, in which the receiver has the ability to jointly decode a packet with previously received versions of the same packet to improve the likelihood of decoding success [44]. For example, in the EDGE system, the transmitter encodes the message (a finite number of information bits) into a codeword of infinite length and transmits a portion (burst) of the codeword to the receiver. The receiver attempts to decode the message after receiving the first transmission. If the decoding fails, the receiver requests another portion of the codeword from the transmitter, and jointly decodes the message by combining the portions received from different transmissions. This process continues until decoding success.

Type-II hybrid ARQ schemes have generated considerable interest in recent years [31] [32] [33] [34] and [35]. Reliability-based type II ARQ scheme have been introduced in [31] and [35]. In their work, the reliability of the received message is computed as the average magnitude of the log-likelihood ratios of the information bits over the packet, and is used to determine the probability of decoding success. This measure of reliability has been used in [36], and can be computed by a MAP decoder as described in [37]. Tripathi et al. in [31] used simulation to determine the relationship between the probability of decoding failure and the total number of transmitted bits under different channel conditions. In contrast to [31] and [35], here we assume that each retransmission has the same size and contains the a fixed amount of redundancy (e.g. number of coded bits) per unit of information. We model the decoding failure probability of each retransmission as a function of the number of transmissions (corresponding to the amount of accumulated redundancy received [38]) and the channel. We still use the term "packet" to denote the amount of bits transmitted during one transmission; this is equivalent to the concept of a "burst" in the EDGE system. This model for scheduling with retransmissions and packet combining has not yet been considered.

1.2.2 Fairness and Multi-user Diversity - a Utility-based Approach

Let us go back to look at the asymmetric properties of the channel conditions to different users. Because different users have different channels, this provides multiuser diversity for the scheduler, e.g., the scheduler has a good chance finding a user with a good channel to transmit [23] [25]. This raises the issue of fairness because of the tendency of the schedular to select only users with good channels for maximizing throughput. This problem is particularly important in a slow fading environment where the channel states change slowly and the users may be stuck in a bad channel state for a long time. Following the utility-based approach in [2], we assume that the QoS requirement for each user is embedded in a utility function, which is generally a decreasing and concave function of delay (or queue length). Thus maximizing the overall utility rate automatically trades off fairness for throughput, and provides a general framework for scheduling in an integrated-service networks. (Retransmission are not considered in [2]).

Most of the literature on scheduling considers linear costs in the form of expected average weighted queue lengths or waiting times. This is generally appropriate for manufacturing application where the goal is to minimize the (weighted) number of jobs in the system. Delay costs in communications systems are typically nonlinear, due to the different service requirements for different applications. For example, packets associated with email or file transfer typically can wait for quite some time without annoying effects, whereas packets belonging to a video system require immediate service with little tolerance for excess delay. Thus it is useful to consider cases with nonlinear delay costs.

In the literature, nonlinear delay cost typically applies to the delay of a specific packet, and the delay cost function is quadratic (e.g., see [18] [19] [20]). Arbitrary increasing delay cost functions have been studied in [19] [21] [22]. Most of the authors focus on developing bounds or heuristic policies, and very few give explicit optimal scheduling rules. An example is [20] which assumes heavy traffic.

In addition to the linear holding cost, we also consider a nonlinear delay cost, which depends on the queue length instead of the delay of the individual packet. This corresponds to controlling the average performance of different queues instead of individual packets. We give simple scheduling rules under this assumption, without considering the heavy traffic regime.

1.3 Outline

In what follows, we give a brief chapter-by-chapter outline of the report.

- Chapter 2 describes the system model. A utility-based stochastic scheduling problem for downlink data transmission in a wireless network using hybrid type-II ARQ is formulated. The queueing control model with feedback from Klimov [16] is also presented, which serves as the basis for the analysis in the next two chapters.

- Chapter 3 discusses the scheduling problem with linear utility functions. The optimal scheduling policy is presented given poisson arrivals.

- Chapter 4 discusses the scheduling problem allowing nonlinear utility functions. Optimal scheduling policy is presented for batch arrivals with long enough interarrival time.

- Chapter 5 summarizes the contributions made in this report. An overview of future research directions to be taken are also presented.

Chapter 2

Problem Formulation

In this chapter, we first present the scheduling model for downlink wireless data transmission with type-II hybrid ARQ. We then present the Klimov queueing control model [16], which will serve as the basis for our analysis in Chapters 3 and 4.

2.1 System Model

We consider the downlink channel of a single-cell wireless communication system using Time Division Multiplexing (TDM). There are a fixed number N of remote mobile users in the system and a single base station transmitter. The data to be transmitted to the N remote mobile users arrive according to N independent random processes and are queued in N infinite-buffer queues at the base station. At the start of each scheduling interval, the scheduler decides which of the N Head of Line (HOL) packets to serve. We assume that the base station transmits to one user at a time with the full available power.

The scheduling intervals (equivalently, transmission time for a packet) are assumed to have fixed length T_s for all users. For user i, where $i \in \{1, 2, ..., N\}$, the length of a packet transmitted during a scheduling interval is therefore $L_i = R_i T_s$, where R_i is the rate at which data can be transmitted to user i. The rate R_i depends on the channel gain h_i and the specific coding and modulation scheme chosen for user i. We assume that R_i is an increasing function of h_i and P respectively, where P is the available power at the base station. Thus

$$R_i = \mathcal{C}(h_i P) \tag{2.1}$$

We consider the situation that the channel gain h_i remains constant for all time. This models the slow fading situation where T_s is short compared to the channel coherence time.

The variation of the channel is due to the thermal noise. Decoding failure happens when signal to noise ratio is not high enough during a transmission, and the packet will stay at the HOL[1] waiting to be retransmitted until it is decoded successfully. We assume that the system uses type-II hybrid ARQ, that is, the receiver jointly decodes the most recent transmission with previously received unsuccessful transmissions. Each packet for user i has the same length L_i, and contains the same amount of accumulated redundancy. Let d_i denote the redundancy per information bit contained in one packet for user i. The quantity d_i could equal, per information bit, the number of coded bits, the energy, or the reliability contained in one packet [38]. Let A_{i,r_i} denote the total accumulated redundancy up to and including the $(r_i+1)\,st$ transmission to user i.

$$y_{i,r_i} = (r_i+1)\,d_i, \qquad r_i \geq 0 \tag{2.2}$$

[1] In EDGE, a new burst will be transmitted instead of the burst that has been transmitted, and the receiver will keep track of all the bursts received for a same message. For the purpose of our scheduling problem, we can mark the packet at the HOL with transmission attempts so that the receiver does not have to keep track of the transmission attempts.

Then the conditional probability that decoding fails with accumulated redundancy y_{r_i}, given that decoding failed with accumulated redundancy A_{r_i}, is

$$g_i(r_i) = \Pr(\text{decoding failure with } y_{i,r_i} \mid \text{decoding failure with } y_{i,r_i-1}), \quad r_i > 1 \tag{2.3}$$

and we define

$$g_i(0) = \Pr(\text{decoding failure with } y_{i,1}) \tag{2.4}$$

$g_i(r_i)$ can also be interpreted as the probability of decoding failure after transmission of packet for user i given the packet has already been transmitted r_i attempts, and has not been decoded successfully. [2]We assume that the probability is nonincreasing with transmission attempts:

$$g_i(r_i) \geq g_i(r_i') \quad \text{if} \quad r_i \leq r_i' \tag{2.5}$$

and there is a maximum transmission attempts $r_i^{\max}$ for user i so that

$$g_i(r_i) = 0 \quad \text{if} \quad r_i \geq r_i^{\max} \tag{2.6}$$

That is, a packet is always successfully decoded after $r_i^{\max} + 1$ transmissions.

We also assume that $g_i(\cdot)$ is time-invariant and only depends on the channel for user i. Also notice that we ignore the transmission and feedback delay, that is, the packet becomes immediately available for retransmission if it is not decoded successfully.

At the decision epoch t (the beginning of a scheduling interval t, $t = 1, 2, ...m$), a utility of $Z_i(x_i(t))$ is collected from the queue for user i at the scheduler, where $x_i(t)$ is the corresponding number of jobs in the queue for user i at time t. If we do not differentiate the packets with different transmission attempts, then $x_i(t)$ is a scalar, otherwise $x_i(t)$ could be a vector. In this

[2]What we really need is that the probability of decoding failure is a function of transmission attempts. The theory of accumulated redundancy is just one of the possible reasons that make this statement true.

chapter we assume that $x_i(t)$ is the total number of jobs for user i (thus $x_i(t)$ is a scalar). The utility function $Z_i(\cdot)$ is assumed to be decreasing, concave with $x_i(t)$, and $Z_i(0) = 0$. (There is no utility to be collected when the queue is empty.) From the definition we can see that $Z_i(\cdot)$ is a nonpositive function with $x_i(t) \geq 0$. Thus we can a cost function for user i: $U_i(\cdot) = -Z_i(\cdot)$, which is always nonnegative. And utility maximization is equivalent to cost minimization. In the rest of the report we will work with cost functions instead of utility functions.

At the decision epoch t (the beginning of scheduling interval t), $t = 1, 2, ...,$ the system can be fully characterized by the set of state vectors:

$$S(t) = \{(r_i(t), x_i(t))\}, \; i \in \{1, 2, ..., N\} \tag{2.7}$$

where $r_i(t) \in \{0, 1, ...\}$ is the transmission attempts experienced by the packet at the HOL of queue i at time t, and $x_i(t) \in \{0, 1, ...\}$ is the length of queue i at time t.

The arrival process of packets for user i is denoted as $A_i(t)$. The set of arrival processes $\{A_i(t)\}$ with $i \in \{1, 2, ...N\}$ are assumed to be stationary and independent. Define Π as the set of nonidling and nonpreemptive policies. Our goal is to find an optimal scheduling policy $\pi \in \Pi$ to minimize the long time average expected cost of the system

$$\lim_{\tau \to \infty} \frac{1}{\tau} \left(\sum_{t=1}^{\tau} E_{\pi} \left[\sum_{i=1}^{N} U_i(x_i(t)) \right] \right) \tag{2.8}$$

Suppose that the set of arrival processes $\{A_i(t)\}$ with $i \in \{1, 2, ..., N\}$ are batch arrivals with inter-arrival time long enough to let the system finishing processing all the packets arriving in a batch before a new batch arrival. In that case our goal is to find an optimal scheduling policy $\pi' \in \Pi$, which minimize the total expected cost for each batch arrival:

$$\sum_{t=1}^{\tau'} E_{\pi'} \left[\sum_{i=1}^{N} U_i(x_i(t)) \right] \tag{2.9}$$

where τ' is the waiting time for the last packet in the queues. In Chapters 3 and 4 we will discuss these two cases with further restrictions on cost functions $U_i(\cdot)$.

In the next section, we will describe the Klimov model and optimal scheduling for a multiclass M/G/1 queueing system with Markovian feedback [16].

2.2 Klimov Model

The Klimov model is defined as followings [16] : There are K infinite-capacity queues. Define $\Omega = \{1, 2, ..., K\}$ to be the set of all queues. The service times of each queue are mutually independent, and the service time for queue $k \in \Omega$ has the distribution function $B_k(x)$. Jobs arrive according to a Poisson process independent of service times with rate λ and are assigned to queue k with probability p_k $\left(\sum_{k=1}^{K} p_k = 1\right)$. A nonidling single server is allocated to one queue at a time in a nonpreemptive way. That is, a new allocation of the server to a queue may only occur at a service completion time of one job, or when a job enters an empty system. A job that has been served in queue k is sent to queue j with probability p_{kj}, and leaves the system with the probability $p_{k0} = 1 - \sum_{j=1}^{K} p_{kj}$, independent of the state of the system.

There are three system assumptions:

1. The transition matrix

$$P = \{p_{kj}\}_{k,j \in \Omega} \tag{2.10}$$

is such that every job eventually leaves, i.e.

$$P^n \to 0 \quad \text{as} \quad n \to \infty \tag{2.11}$$

This implies that $I - P$ is invertible, where I is the identity matrix;

2. The first and second moments of the duration of the service in any queue are finite

$$b_k := \int_0^\infty x \; dB_k(x) < \infty; \; b_k^{(2)} := \int_0^\infty x^2 \; dB_k(x) < \infty \tag{2.12}$$

for $k \in \Omega$.

3. The stability condition (under any nonidling policy) is satisfied:

$$\lambda p(I-P)^{-1}b < 1 \tag{2.13}$$

where $p = \{p_k, k \in \Omega\}$ and $b = \{b_k, k \in \Omega\}$. This means that the arrival rate does not exceed the processing capacity of the system.

Let $c_k \geq 0$ ($k \in \Omega$) be the holding cost rate, that is, holding cost per unit of time per job in queue k. Let $y_k(t)$ be the numbers of jobs at time $t \geq 0$ in queue k ($k \in \Omega$). Define Π as the set of all nonidling and nonpreemptive policies. Our objective is to find a scheduling policy $\pi \in \Pi$ that minimizes the long-run average expected cost incurred over an infinite horizon:

$$\lim_{T\to\infty} \frac{1}{T} E\left[\int_0^T \sum_{k=1}^K c_k y_k(t) dt\right] \tag{2.14}$$

. Klimov showed that the optimal scheduling policy is a fixed-priority rule, and gave an algorithm to calculate the priority index.

For any set $M \subset \Omega$, define $T_k^{(M)}$ to be the mean total time for servicing a job (not including its waiting time) beginning with queue $k \in M$ up to its first exit from the queue set M. Clearly,

$$T_k^{(\Omega)} = \sum_{j\in\Omega} p_{kj} T_j^{(\Omega)} + b_k \tag{2.15}$$

, This can be written as

$$T^{(\Omega)} = (I-P)^{-1}b \tag{2.16}$$

, where $T^{(\Omega)} = \left\{T_k^{(\Omega)}, k \in \Omega\right\}$ and $b = \{b_k, k \in \Omega\}$. For any $M \subset \Omega$, the delays $T_k^{(M)}$, $k \in M$, are determined in a similar way,

$$T^{(M)} = (I^{(M)} - P^{(M)})^{-1} b^{(M)} \tag{2.17}$$

, where $T^{(M)} = \left\{T_k^{(M)}, k \in M\right\}$. Here $I^{(M)}$ is $|M| \times |M|$ identity matrix, and $P^{(M)}$ is obtained by deleting the rows and columns in P corresponding to the queues in $\Omega - M$, and $b^{(M)} = \left\{b_k^{(M)}, k \in M\right\}$ is obtained by deleting the corresponding components of the vector b.

We can now illustrate the algorithm for computing the priority indexes of the queues. Initialize by defining $M_K = \Omega$ and the holding cost rate $C_k^{(M_K)} = c_k$, $k \in M_K$. During each backward iteration n (n starts from K and decreases to 1), find a queue $\alpha^{(M_n)}$ with the lowest ratio of $C_k^{(M_n)}/T_k^{(M_n)}$, $k \in M_n$, where $C_k^{(M_n)}$ and $T_k^{(M_n)}$ the are holding cost rate and delays of queue k defined on set M_n. Define a new set $M_{n-1} = M_n - \{\alpha^{(M_n)}\}$, update the holding cost rate of each queue k ($k \in M_{n-1}$) and denote it as $C_k^{(Mn)}$ $(k \in \Omega)$. Continue the iterations until $n = 1$. In this way we order the queues as $(\alpha^{(M_1)}, \alpha^{(M_2)}, ..., \alpha^{(M_K)})$ from the highest priority to the lowest.

The algorithm is summarized as the follows:

Initialization:

$$M_K = \Omega,\ C_k^{(M_K)} = c_k,\ k \in M_K \tag{2.18}$$

Iterations:

$$\text{Step 1: } \alpha^{(M_n)} = \underset{k \in Mn}{\arg\min} \frac{C_k^{(M_n)}}{T_k^{(M_n)}};\ \theta^{(M_n)} = \frac{C_{\alpha^{(M_n)}}^{(M_n)}}{T_{\alpha^{(M_n)}}^{(M_n)}} \tag{2.19}$$

$$\text{Step 2: } M_{n-1} = \Omega - \left\{\alpha^{(M_K)}, \cdots, \alpha^{(M_n)}\right\} = M_n - \left\{\alpha^{(M_n)}\right\} \tag{2.20}$$

$$\text{Step 3: } C_k^{(M_{n-1})} = T_k^{(M_n)} \left[\frac{C_k^{(M_n)}}{T_k^{(M_n)}} - \theta^{(M_n)}\right],\ k \in M_{n-1} \tag{2.21}$$

Then the fixed-priority policy π that always assigns the server to the nonempty queue $\alpha^{(M_n)}$ with the smallest index n is optimal among all nonidling and nonpreemptive policies.

Chapter 3

Scheduling with Linear Cost Functions

In this chapter, we will limit our attention to scheduling problems with linear utility functions (equivalently, linear cost functions) and stationary arrival processes. We first consider the case when the arrival process is Poisson, and find the optimal scheduling policy based on the Klimov model described in section 2.2. Then we briefly discuss the case with arbitrary stationary arrival processes.

3.1 Linear Cost Functions and Poisson Arrivals

We first formulate a problem based on the system model described in section 2.1 with further assumptions on the cost functions and arrival processes. We then cast the problem into a standard Klimov problem and find the optimal scheduling policy. After that we give some numerical examples.

3.1.1 Problem Formulation: LP Scheduling Problem

We will refer to the scheduling problem with linear cost functions and Poisson arrivals as the **LP** scheduling problem.

Assume there are N users in the system. Denote $\Xi = \{1, 2, ..., N\}$. When we say queue $i \in \Xi$, we mean the queue belongs to user i at the scheduler. For user $i \in \Xi$, denote $\Gamma_i = \{0, 1, ..., r_i^{\max}\}$, where r_i is the transmission attempts of the packet and $r_i^{\max}$ is the maximum transmission attempts defined in (2.6). The cost functions are linear. The holding cost per packet per unit time (holding cost rate) not only depends on i (the user index), but also on r_i (the transmission attempts experienced by the packet). For user i, we denote the set of holding cost rates as

$$c_i^{LP} = \left\{ c_{i,r_i}^{LP}, r_i \in \Gamma_i \right\} \tag{3.1}$$

with c_{i,r_i}^{LP} representing the holding cost rate for a packet of user i with r_i transmissions. Assume that

$$0 \leq c_{i,r_i}^{LP} \leq c_{i,r_i'}^{LP}, \quad \text{for } r_i \leq r_i' \tag{3.2}$$

which means the more transmissions experienced by the packet, the more expensive it is to keep the packet in the system. For the HOL packet of queue $i \in \Xi$, denote its transmission time as r_i^{HOL} and its holding cost rate as $c_{i,r_i^{HOL}}^{LP}$. All other packets in queue i are new packets (have not been transmitted even once) with holding cost rate $c_{i,0}^{LP}$.

Assume that the the arrival process of user i is a Poisson process with rate λ_i^{LP}, and is independent of the other arrival processes.

Denote the number of packets in queue i at decision epoch t ($t = 1, 2, ...$) as $x_i^{LP}(t)$. Denote the set of nonempty queues at decision epoch t as $\Xi^{NE}(t)$. The holding cost of the nonempty queue i at time t is

$$U_i(x_i^{LP}(t)) = c_{i,0}^{LP}\left(x_i^{LP}(t) - 1\right) + c_{i,r_i^{HOL}}^{LP}, \quad i \in \Xi^{NE}(t) \tag{3.3}$$

Define Π as the set of all nonidling and nonpreemptive policies. Our goal is to find an optimal policy $\pi^{LP} \in \Pi$ that minimizes the long-run average expected cost

$$\lim_{\tau \to \infty} \frac{1}{\tau} \left(\sum_{t=1}^{\tau} E_{\pi^{LP}} \left[\sum_{i \in \Xi^{NE}(t)} U_i(x_i^{LP}(t)) \right] \right) \tag{3.4}$$

In the next section we will show how to reformulate the **LP** as a standard Klimov problem. We will refer to this scheduling problem as **LPK**: linear cost functions and Poisson arrival processes based on the Klimov model.

3.1.2 LPK Scheduling Problem

LPK is a relaxation of **LP** with respect to service discipline. In **LPK**, we allow the server to begin serving a new packet for user $i \in \Xi$ before the decoding success of a previous packet from the same user. The reasons for doing this is two-fold: Firstly, with this relaxation **LPK** fits nicely into the framework of the Klimov model; Secondly, even when we make this relaxation assumption, the optimal scheduling rule for the **LPK** also turns out to be optimal for **LP**. In this section and the next, we use the notation for the Klomov model in section 2.2.

According to the relaxed service assumptions of **LPK**, for each user $i \in \Xi$, we can have at most $r_i^{\max} + 1$ type packets denoted by

$$\phi_i = \{\phi_{i,r_i}, r_i \in \Gamma_i\} \tag{3.5}$$

corresponding to the packets that have been transmitted for $0, 1, ..., r_i^{\max}$ times respectively. Equivalently, we say that for user $i \in \Xi$, there are $r_i^{\max} + 1$ queues denoted by

$$Q_i = \{Q_{i,r_i}, r_i \in \Gamma_i\} \tag{3.6}$$

at the scheduler with packets ϕ_{i,r_i} staying in queue Q_{i,r_i}. Thus there will be a total of $K = \sum_{i\in\Xi}(r_i^{\max}+1)$ in the system and define

$$\Omega = \{Q_i, i \in \Xi\} \tag{3.7}$$

to be the set of total K queues.

After a packet from queue Q_{i,r_i} $(0 \le r_i < r_i^{\max}, i \in \Xi)$ has been served, it will enter queue Q_{i,r_i+1} with probability $p_{Q_{i,r_i},Q_{i,r_i+1}} = g_i(r_i)$, or leave the system with probability $p_{Q_{i,r_i},0} = 1 - g_i(r_i)$. If a packet from queue $Q_{i,r_i^{\max}}$ has been served, it will not enter any other queue and will leave the system with probability $p_{Q_{i,r_i^{\max}},0} = 1$. Define the transition matrix

$$P = \left\{p_{Q_{i,r_i},Q_{j,r_j}}, i,j \in \Xi, r_i \in \Gamma_i, r_j \in \Gamma_j\right\} \tag{3.8}$$

Now the service discipline of **LPK** can be restated as the following: at each decision epoch t $(t = 1, 2, ...)$, the server decides which of the K HOL packets to serve.

Imagine that there is an Poisson arrival process to the whole system with rate $\lambda = \sum_{i\in\Xi}\lambda_i^{LP}$, and the incoming packet is assigned to queue $Q_{i,0}$ with probability $p_i = \lambda_i^{LP}/\lambda$. Thus the arrival process for each queue $Q_{i,0}$ is a Poisson process with rate λ_i^{LP}. Define

$$p = \{p_i, i \in \Xi\} \tag{3.9}$$

Define $b_{Q_{i,r_i}}$ be the mean value of service times of queue Q_{i,r_i}, and we assume $b_{Q_{i,r_i}} = 1$ (here we scale T_s to 1 without affecting the optimal policy). Define

$$b_i = \left\{b_{Q_{i,r_i}}, r_i \in \Gamma_i\right\} \tag{3.10}$$

and

$$b = \{b_i, i \in \Xi\} \tag{3.11}$$

Let I be the identity matrix, and assume that the arrival process satisfies the stability assumption of the Klimov model:

$$\lambda p(I-P)^{-1}b < 1 \tag{3.12}$$

For user $i \in \Xi$, denote the number of packets in Q_i at decision epoch t $(t = 1, 2, ...)$ as

$$x_i(t) = \left\{x_{Q_{i,r_i}}(t), r_i \in \Gamma_i\right\} \tag{3.13}$$

And denote the holding cost rate for packets in Q_i as

$$c_i = \left\{c_{Q_{i,r_i}} = c^{LP}_{i,r_i}, r_i \in \Gamma_i\right\} \tag{3.14}$$

Thus

$$c_{Q_{i,r_i}} \leq c_{Q_{i,r_i'}} \quad \text{for} \quad r_i \leq r_i' \tag{3.15}$$

The holding cost of user i charged at t is:

$$U_i(x_i(t)) = \sum_{r_i \in \Gamma_i} c_{Q_{i,r_i}} x_{Q_{i,r_i}}(t) \tag{3.16}$$

Our goal is to find an optimal scheduling policy $\pi^{LPK} \in \Pi$ to minimize the long-run average expected cost of the system

$$\lim_{\tau\to\infty} \frac{1}{\tau}\left(\sum_{t=1}^{\tau} E_{\pi^{LPK}}\left[\sum_{i\in\Xi} U_i(x_i(t))\right]\right) \tag{3.17}$$

For any set $M \subset \Omega$, define $T^{(M)}_{Q_{i,r_i}}$ as the mean total time of servicing a job (not including its waiting time) beginning with queue $Q_{i,r_i} \in M$ up to its first exit from the set M. Clearly:

$$T^{(\Omega)}_{Q_{i,r_i}} = \sum_{Q_{j,r_j}\in\Omega} p_{Q_{i,r_i},Q_{j,r_j}} T^{(\Omega)}_{Q_{j,r_j}} + b_{Q_{i,r_i}}, \quad \forall Q_{i,r_i} \in \Omega \tag{3.18}$$

Equivalently:

$$T^{(\Omega)} = (I-P)^{-1}b \tag{3.19}$$

Where $T^{(\Omega)} = \left\{T^{(\Omega)}_{Q_{i,r_i}}, Q_{i,r_i} \in \Omega\right\}$. For any $M \subset \Omega$, $T^{(M)} = \left\{T^{(M)}_{Q_{i,r_i}}, Q_{i,r_i} \in M\right\}$ are determined in a similar way.

$$T^{(M)} = (I^{(M)} - P^{(M)})^{-1} b^{(M)} \tag{3.20}$$

$P^{(M)}$ can be obtained by deleting the rows and columns in P corresponding to the queues in $\Omega - M$. $b^{(M)} = \left\{b^{(M)}_{Q_{i,r_i}}, Q_{i,r_i} \in M\right\}$ can be obtained by deleting the components of the vectors b corresponding to the same queues. $I^{(M)}$ is identity matrix with its size decided by the size of set M. Notice that all the entries of $b^{(M)}$ is 1 in our problem.

LPK problem is a standard Klimov model and we rewrite the Klimov algorithm for calculating the priorities here for **LPK** problem.

Initialization:

$$M_K = \Omega,\ C^{(M_K)}_{Q_{i,r_i}} = c_{Q_{i,r_i}},\ Q_{i,r_i} \in M_K \tag{3.21}$$

Iterations:

$$\text{Step 1: } \alpha^{(M_n)} = \underset{Q_{i,r_i} \in Mn}{\arg\min} \frac{C^{(M_n)}_{Q_{i,r_i}}}{T^{(M_n)}_{Q_{i,r_i}}};\ \theta^{(M_n)} = \frac{C^{(M_n)}_{\alpha^{(M_n)}}}{T^{(M_n)}_{\alpha^{(M_n)}}} \tag{3.22}$$

$$\text{Step 2: } M_{n-1} = \Omega - \left\{\alpha^{(M_K)}, \cdots, \alpha^{(M_n)}\right\} = M_n - \left\{\alpha^{(M_n)}\right\} \tag{3.23}$$

$$\text{Step 3: } C^{(M_{n-1})}_{Q_{i,r_i}} = T^{(M_n)}_{Q_{i,r_i}} \left[\frac{C^{(M_n)}_{Q_{i,r_i}}}{T^{(M_n)}_{Q_{i,r_i}}} - \theta^{(M_n)}\right],\ Q_{i,r_i} \in M_{n-1} \tag{3.24}$$

3.1.3 Optimal Policies of LPK and LP Scheduling Problems

It is useful to define $\Delta_{\left\{\gamma_i^*, i \in \Xi\right\}} \subset \Omega$ as the set with the following properties $\forall i \in \Xi$:

1. $-1 \le \gamma_i^* \le r_i^{\max}$.

2. If $\gamma_i^* = -1$, then

$$\forall\ r_i \text{ such that } 0 \le r_i \le r_i^{\max},\ Q_{i,r_i} \notin \Delta_{\left\{\gamma_i^*, i\in\Xi\right\}} \tag{3.25}$$

which means that there is no queue belonging to user i in the set.

3. If $0 \le r_i^* \le r_i^{\max}$, then

$$\forall r_i \text{ such that } \gamma_i^* \le r_i \le r_i^{\max},\ Q_{i,r_i} \in \Delta_{\left\{\gamma_i^*, i\in\Xi\right\}} \tag{3.26}$$

$$\forall r_i \text{ such that } \ 0 \le r_i < \gamma_i^*,\ Q_{i,r_i} \notin \Delta_{\left\{\gamma_i^*, i\in\Xi\right\}} \tag{3.27}$$

Follow the notation in the last section, we have the following results for **LPK** scheduling problem:

Lemma 3.1 (LPK-T) *Given any set $\Delta_{\left\{\gamma_i^*, i\in\Xi\right\}} \subset \Omega$, $\forall Q_{i,r_i} \in \Delta_{\left\{\gamma_i^*, i\in\Xi\right\}}$, we have*

1. $T_{Q_{i,r_i}}^{\left(\Delta_{\left\{\gamma_i^*, i\in\Xi\right\}}\right)} = T_{Q_{i,r_i}}^{(\Omega)}$

2. $T_{Q_{i,r_i}}^{\left(\Delta_{\left\{\gamma_i^*, i\in\Xi\right\}}\right)} \ge T_{Q_{i,r_i'}}^{\left(\Delta_{\left\{\gamma_i^*, i\in\Xi\right\}}\right)} > 0$ *for* $r_i \le r_i' \le r_i^{\max}$.

where for any set $M \subset \Omega$, $T_{Q_{i,r_i}}^{(M)}$ is defined in (3.20).

Proof. Obviously $\Delta_{\{0,i\in\Xi\}} = \Omega$. Thus $\forall\ Q_{i,r_i} \in \Delta_{\{0,i\in\Xi\}}$, we have $T_{Q_{i,r_i}}^{\left(\Delta_{\{0,i\in\Xi\}}\right)} = T_{Q_{i,r_i}}^{(\Omega)}$. According to (3.18) and (3.8):

$$\begin{cases} T_{Q_{i,r_i}}^{(\Omega)} = 1 + g_i(r_i) T_{Q_{i,r_i+1}}^{(\Omega)}, & 0 \le r_i < r_i^{\max}, i \in \Xi \\ T_{Q_{i,r_i^{\max}}}^{(\Omega)} = 1 & i \in \Xi \end{cases} \tag{3.28}$$

Solve the above equations:

$$T_{Q_{i,r_i}}^{(\Omega)} = 1 + \sum_{j=r_i}^{r_i^{\max}-1} \prod_{l=r_i}^{j} g_i(l) > 0,\ \forall\ Q_{i,r_i} \in \Omega \tag{3.29}$$

We can see that $T^{(\Omega)}_{Q_{i,r_i}}$ only depend on the value of $g_i(r_i)$, $g_i\left(r_i+1\right), ..., g_i\left(r_i^{\max}\right)$, thus depend on the presence of queues $Q_{i,r_i'}$ with $r_i \leq r_i' \leq r_i^{\max}$. Together with (2.5), we have

$$T^{(\Omega)}_{Q_{i,r_i}} \geq T^{(\Omega)}_{Q_{i,r_i'}} > 0 \ \text{ with } r_i \leq r_i' \leq r_i^{\max}. \tag{3.30}$$

which means that the more transmission experienced by the packet, the less expected service time up to its exit of the system.

Given set $\Delta_{\{\gamma_i^*, i\in\Xi\}}$, $\forall\ Q_{i,r_i} \in \Delta_{\{\gamma_i^*, i\in\Xi\}}$, we can see from the definition of $\Delta_{\{\gamma_i^*, i\in\Xi\}}$ that $Q_{i,r_i'}$ with $r_i \leq r_i' \leq r_i^{\max}$ are in the set. Thus $T^{\left(\Delta_{\{\gamma_i^*, i\in\Xi\}}\right)}_{Q_{i,r_i}} = T^{(\Omega)}_{Q_{i,r_i}}$ and

$$T^{\left(\Delta_{\{\gamma_i^*, i\in\Xi\}}\right)}_{Q_{i,r_i}} \geq T^{\left(\Delta_{\{\gamma_i^*, i\in\Xi\}}\right)}_{Q_{i,r_i'}} > 0 \quad \text{for} \quad r_i \leq r_i' \leq r_i^{\max}. \tag{3.31}$$

■

Lemma 3.2 (LPK-C-α) *For the **LPK** scheduling problem, each set $M_n\,(n=1,2,...,K)$ can be denoted in the form of $\Delta_{\left\{\gamma_i^{(M_n)}, i\in\Xi\right\}}$. $\forall Q_{i,r_i} \in M_n$, we have*

1. $C^{(M_n)}_{Q_{i,r_i}} = c_{Q_{i,r_i}} - T^{(\Omega)}_{Q_{i,r_i}} \sum\limits_{l=n+1}^{K} \theta^{(M_l)} \geq 0$

2. $\alpha^{(M_n)} = \underset{Q_{i,\gamma_i^{(M_n)}} \in M_n}{\arg\min} \left\{ \dfrac{c_{Q_{i,\gamma_i^{(M_n)}}}}{T^{(\Omega)}_{Q_{i,\gamma_i^{(M_n)}}}} \right\}.$

Where M_n, $C^{(M_n)}_{Q_{i,r_i}}$, $c_{Q_{i,r_i}}$, $\theta^{(M_n)}$, $\alpha^{(M_n)}$ are defined in (3.23) (3.24) (3.14) (3.22) *and* (3.22) *respectively.*

Proof. Consider $n = K$. From definition of M^K, we have $M^K = \Omega = \Delta_{\{0, i\in\Xi\}} = \Delta_{\left\{\gamma_i^{(M_K)}, i\in\Xi\right\}}$, thus $\gamma_i^{(M_K)} = 0$ for $i \in \Xi$.

$\forall Q_{i,r_i} \in M_K$, from definition $T^{(M_K)}_{Q_{i,r_i}} = T^{(\Omega)}_{Q_{i,r_i}}$, and from Lemma 3.1, we have

$$T^{(M_K)}_{Q_{i,r_i}} \geq T^{(M_K)}_{Q_{i,r'_i}} > 0 \quad \text{for} \quad r_i \leq r'_i \leq r_i^{\max} \tag{3.32}$$

From (3.21) and (3.14), $C^{(M_K)}_{Q_{i,r_i}} = c_{Q_{i,r_i}}$. Together with (3.15), we have

$$0 \leq C^{(M_K)}_{Q_{i,r_i}} \leq C^{(M_K)}_{Q_{i,r'_i}} \quad \text{for} \quad r_i \leq r'_i \leq r_i^{\max} \tag{3.33}$$

Thus

$$0 \leq \frac{C^{(M_K)}_{Q_{i,r_i}}}{T^{(M_K)}_{Q_{i,r_i}}} \leq \frac{C^{(M_K)}_{Q_{i,r'_i}}}{T^{(M_K)}_{Q_{i,r'_i}}} \quad \text{for} \quad r_i \leq r'_i \leq r_i^{\max} \tag{3.34}$$

Then

$$\alpha^{(M_K)} = \operatorname*{arg\,min}_{Q_{i,r_i} \in M_K} \frac{C^{(M_K)}_{Q_{i,r_i}}}{T^{(M_K)}_{Q_{i,r_i}}} = \operatorname*{arg\,min}_{Q_{i,0} \in M_K} \left\{ \frac{C^{(M_K)}_{Q_{i,0}}}{T^{(M_K)}_{Q_{i,0}}} \right\} = \operatorname*{arg\,min}_{Q_{i,\gamma_i^{(M_K)}} \in M_K} \left\{ \frac{c_{Q_{i,\gamma_i^{(M_K)}}}}{T^{(\Omega)}_{Q_{i,\gamma_i^{(M_K)}}}} \right\} \tag{3.35}$$

This shows that the lemma holds for M_K.

Assume the lemma holds for M_{n+1} $(1 < n \leq K)$, together with Lemma 3.1, we have

$$M_{n+1} = \Delta_{\left\{\gamma_i^{(M_{n+1})}, i \in \Xi\right\}} \tag{3.36}$$

and $\forall Q_{i,r_i} \in M_{n+1}$,

$$T^{(M_{n+1})}_{Q_{i,r_i}} = T^{(\Omega)}_{Q_{i,r_i}} \tag{3.37}$$

$$T^{(M_{n+1})}_{Q_{i,r_i}} \geq T^{(M_{n+1})}_{Q_{i,r'_i}} > 0 \text{ for } \ r_i \leq r'_i \leq r_i^{\max} \tag{3.38}$$

$$C^{(M_{n+1})}_{Q_{i,r_i}} = c_{Q_{i,r_i}} - T^{(\Omega)}_{Q_{i,r_i}} \sum_{l=n+2}^{K} \theta^{(M_l)} \geq 0 \tag{3.39}$$

$$\alpha^{(M_{n+1})} = \operatorname*{arg\,min}_{Q_{i,\gamma_i^{(M_n)}} \in M_{n+1}} \left\{ \frac{c_{Q_{i,\gamma_i^{(M_{n+1})}}}}{T^{(\Omega)}_{Q_{i,\gamma_i^{(M_{n+1})}}}} \right\} \tag{3.40}$$

Now consider M_n . Assume $\alpha^{(M_{n+1})}$ belongs to the queues of user i^*, thus from induction assumption: $\alpha^{(M_{n+1})} = Q_{i^*,\gamma_{i^*}^{(M_{n+1})}}$. Let $\gamma_{i^*}^{(M_n)} = \gamma_{i^*}^{(M_{n+1})} + 1$ if $\gamma_{i^*}^{(M_{n+1})} < \gamma_{i^*}^{\max}$, otherwise let $\gamma_{i^*}^{(M_n)} = -1$. Let $\gamma_i^{(M_n)} = \gamma_i^{(M_{n+1})}$ for all $i \in \Xi$ and $i \neq i^*$. Thus $M_n = \Delta_{\left\{\gamma_i^{(M_n)}, i\in\Xi\right\}}$.

$\forall Q_{i,r_i} \in M_n$, according to Lemma 3.1, we have $T_{Q_{i,r_i}}^{(Mn)} = T_{Q_{i,r_i}}^{(\Omega)}$ and

$$T_{Q_{i,r_i}}^{(M_n)} \geq T_{Q_{i,r_i'}}^{(M_n)} > 0 \quad \text{for } r_i \leq r_i' \leq r_i^{\max} \tag{3.41}$$

From definition (3.24) and (3.22)

$$\begin{aligned} C_{Q_{i,r_i}}^{(M_n)} &= C_{Q_{i,r_i}}^{(M_{n+1})} - T_{Q_{i,r_i}}^{(M_{n+1})} \theta^{(M_{n+1})} &(3.42)\\ &= C_{Q_{i,r_i}}^{(M_{n+1})} - T_{Q_{i,r_i}}^{(M_{n+1})} \min_{\widetilde{Q}_{i,r_i}} \frac{C_{\widetilde{Q}_{i,r_i}}^{(M_{n+1})}}{T_{\widetilde{Q}_{i,r_i}}^{(M_n)}} &(3.43)\\ &\geq C_{Q_{i,r_i}}^{(M_{n+1})} - T_{Q_{i,r_i}}^{(M_{n+1})} \frac{C_{Q_{i,r_i}}^{(M_{n+1})}}{T_{Q_{i,r_i}}^{(M_n)}} &(3.44)\\ &= 0 &(3.45) \end{aligned}$$

From definition (3.24) and induction assumption (3.39)

$$\begin{aligned} C_{Q_{i,r_i}}^{(M_n)} &= C_{Q_{i,r_i}}^{(M_{n+1})} - T_{Q_{i,r_i}}^{(M_{n+1})} \theta^{(M_{n+1})} &(3.46)\\ &= c_{Q_{i,r_i}} - T_{Q_{i,r_i}}^{(\Omega)} \sum_{l=n+1}^{K} \theta^{(M_l)} &(3.47) \end{aligned}$$

Together with (3.15) (3.41) and (3.45) , we have

$$0 \leq C_{Q_{i,r_i}}^{(Mn)} \leq C_{Q_{i,r_i'}}^{(M_n)} \quad \text{for} \quad r_i \leq r_i' \leq r_i^{\max} \tag{3.48}$$

Thus

$$0 \leq \frac{C_{Q_{i,r_i}}^{(M_n)}}{T_{Q_{i,r_i}}^{(M_n)}} \leq \frac{C_{Q_{i,r_i'}}^{(M_n)}}{T_{Q_{i,r_i'}}^{(M_n)}} \quad \text{for} \quad r_i \leq r_i' \leq r_i^{\max} \tag{3.49}$$

Then

$$\alpha^{(M_n)} = \arg\min_{Q_{i,r_i} \in M_n} \frac{C^{(M_n)}_{Q_{i,r_i}}}{T^{(M_n)}_{Q_{i,r_i}}} = \arg\min_{Q_{i,\gamma_i^{(M_n)}} \in M_n} \left\{ \frac{C^{(M_n)}_{Q_{i,\gamma_i^{(M_n)}}}}{T^{(M_n)}_{Q_{i,\gamma_i^{(M_n)}}}} \right\} = \arg\min_{Q_{i,\gamma_i^{(M_n)}} \in M_n} \left\{ \frac{c_{Q_{i,\gamma_i^{(M_n)}}}}{T^{(\Omega)}_{Q_{i,\gamma\sigma_i^{(M_n)}}}} \right\} \tag{3.50}$$

Thus the lemma holds for M_n $(1 \leq n \leq K)$. ■

Theorem 3.1 (LPK-Priority Rule) *For the **LPK** scheduling problem, the optimal scheduling rule is a fixed priority rule in which the priorities of queues in the order of (from the highest to the lowest)*

$$\alpha^{(M_1)}, \alpha^{(M_2)}, \cdots, \alpha^{(M_K)} \tag{3.51}$$

where $\alpha^{(M_n)}$ $(1 \leq n \leq K)$ *was calculated by the Klimov algorithm* (3.22) *and satisfy*

$$\frac{c_{\alpha^{(M_1)}}}{T^{(\Omega)}_{\alpha^{(M_1)}}} \geq \frac{c_{\alpha^{(M_2)}}}{T^{(\Omega)}_{\alpha^{(M_2)}}} \geq \cdots \geq \frac{c_{\alpha^{(M_K)}}}{T^{(\Omega)}_{\alpha^{(M_K)}}} \tag{3.52}$$

Proof. Given set M_n $(1 \leq n \leq K)$ defined in (3.22), let $\sigma_s^{(M_n)}$ be the queue with the *sth* smallest ratio of $c_{Q_{i,r_i^{(M_n)}}}/T^{(\Omega)}_{Q_{i,r_i^{(M_n)}}}$ within all queues $Q_{i,r_i} \in M_n$. According to Lemma 3.2,

$$\alpha^{(M_n)} = \sigma_1^{(M_n)} = \sigma_2^{(M_{n+1})} = \cdots = \sigma_{K-n+1}^{(M_K)} \tag{3.53}$$

Thus we can rewrite the sequences of queues $\alpha^{(M_1)}, \alpha^{(M_2)}, \cdots, \alpha^{(M_K)}$ as $\sigma_K^{(M_K)}, \sigma_{K-1}^{(M_2)}, \cdots, \sigma_1^{(M_K)}$. From the definition of $\gamma_s^{(M_n)}$, we have

$$\frac{c_{\alpha^{(M_1)}}}{T^{(\Omega)}_{\alpha^{(M_1)}}} \geq \frac{c_{\alpha^{(M_2)}}}{T^{(\Omega)}_{\alpha^{(M_2)}}} \geq \cdots \geq \frac{c_{\alpha^{(M_K)}}}{T^{(\Omega)}_{\alpha^{(M_K)}}} \tag{3.54}$$

■

Now we are ready to state the optimal scheduling rule for the **LP** scheduling problem, which is the one we are originally interested:

Theorem 3.2 (LP-Priority Rule) *For the* ***LP*** *scheduling problem, the optimal scheduling rule is the following: at the beginning of each scheduling interval* t $(t = 1, 2, ...)$*, the base station choose to transmit the HOL packet with the highest ratio of* $c^{LP}_{i,r_i^{HOL}}/T^{(\Omega)}_{i,r_i^{HOL}}$ *among all the nonempty queues. Here* $c^{LP}_{i,r_i^{HOL}}$ *is the holding cost rate for the packet at the HOL at the queue of user* i*, and* $T^{(\Omega)}_{i,r_i^{HOL}}$ *is the mean total time of servicing the HOL packet (not including its waiting time) at the queue of user* i *up to its exit from the system.*

Proof. In the **LPK** scheduling problem, $\forall Q_{i,r_i}, Q_{i,r'_i} \in \Omega$ with $0 \leq r_i \leq r'_i \leq r_i^{\max}$, we have

$$\frac{c_{Q_{i,r_i}}}{T^{(\Omega)}_{Q_{i,r_i}}} \leq \frac{c_{Q_{i,r'_i}}}{T^{(\Omega)}_{Q_{i,r'_i}}} \tag{3.55}$$

According to Theorem 3.1,

$$\text{Priority}\,(Q_{i,r_i}) \leq \text{Priority}\left(Q_{i,r'_i}\right) \tag{3.56}$$

This shows that the priority of a packet is nondecreasing with transmission attempts. Thus as soon as the system starts transmitting a new packet from user $i \in \Xi$, it will not start to transmit another new packet from the same user until the previous packet from this user has been successfully decoded and leave the system. This means that for each user $i \in \Xi$, there will be at most one packet with $r_i > 0$ and this packet has priority over all the other packets (which are all new packets) belonging to the same user. In fact, this packet corresponds to the HOL of packet at the queue of user i in **LP** scheduling problem.

Thus the optimal scheduling rule for **LP** scheduling problem is a fixed priority rule with focus only on the HOL packets at the nonempty queues. According to Theorem 3.1, the packet with the highest ratio of $c_{Q_{i,r_i}}/T^{(\Omega)}_{Q_{i,r_i}}$ has the highest priority. From definitions, we have

$$c^{LP}_{i,r_i^{HOL}} = c_{Q_{i,r_i}},\ \forall\ i \in \Xi, r_i \in \Gamma_i \tag{3.57}$$

$$T^{(\Omega)}_{i,r_i^{HOL}} = T^{(\Omega)}_{Q_{i,r_i}},\ \forall\ i \in \Xi, r_i \in \Gamma_i \tag{3.58}$$

Thus the base station will choose to transmit the HOL packet with the highest ratio of $c^{LP}_{i,r_i^{HOL}}/T_{i,r_i^{HOL}}$ among all the nonempty queues. ■

In the next section we will give some numerical examples to illustrate the optimal policies of LP scheduling problem.

3.1.4 Numerical Study

In this section, we study the properties of the optimal scheduling policy of LP problem through some numerical examples

Same Holding Cost Rates within Each User

We first consider the cases that the holding cost rate is the same for all the packets of a same user $i \in \Xi$, that is,

$$c^{LP}_{i,r_i} = c^{LP}_{i,0}\,,\ \forall r_i \in \Gamma_i \tag{3.59}$$

and we omit the subscript r_i and write the holding cost of user $i \in \Xi$ as c^{LP}_i directly.

Sensitivity Study: Channel Variation with equal Holding Cost Rates Figure 3.1 shows a typical example of the optimal policy of scheduling problem containing 2 users with same holding cost rates but different channel conditions (reflected in different decoding failure probabilities). The parameters are given beside the figure. In particular, we assume that the decoding failure probability decreases exponentially with transmission attempts and the decreasing rate is determined by the channel condition. We use η_i ($i \in \{1, 2\}$) to denote the parameter for the exponential function for user i thus the decoding failure probability can be written as

$$\begin{cases} g_i(r_i) = \eta_i^{r_i+1}, & 0 \leq r_i < r_i^{\max} \\ g_i\left(r_i^{\max}\right) = 0 \end{cases} \tag{3.60}$$

in this example $\eta_1 = 0.1$ and $\eta_2 = 0.2$. Obviously user 1 has a better channel than user 2.

The horizontal (vertical) axis represents the transmission attempts experienced by the packets of user 1 (user 2). The symbol (dot, circle or triangle) at the point (r_1, r_2) represents the action in the situation of choosing a packet from two HOL packets ϕ_{1,r_1} and ϕ_{1,r_2} (ϕ_{i,r_i} is defined in (3.5)) to schedule. Dot represents that it is optimal to schedule (or transmit) user 1 (or packet ϕ_{1,r_1}), circle represents that it is optimal to schedule (or transmit) user 2 (or packet ϕ_{2,r_2}), while triangle means it is optimal to schedule either user. From the figure we can see that the optimal scheduling policy can be characterized as a monotonic switching curve, which increases with transmission attempts for both users. A unique point in the graph is $(r_1^{\max}, r_2^{\max}) = (10, 10)$, where both users have the same priority. This is due to the assumption that both users have the same holding cost rates, thus $c_1/T^{(\Omega)}_{1,r_1^{\max}} = c_1 = c_2 = c_2/T^{(\Omega)}_{2,r_2^{\max}}$.

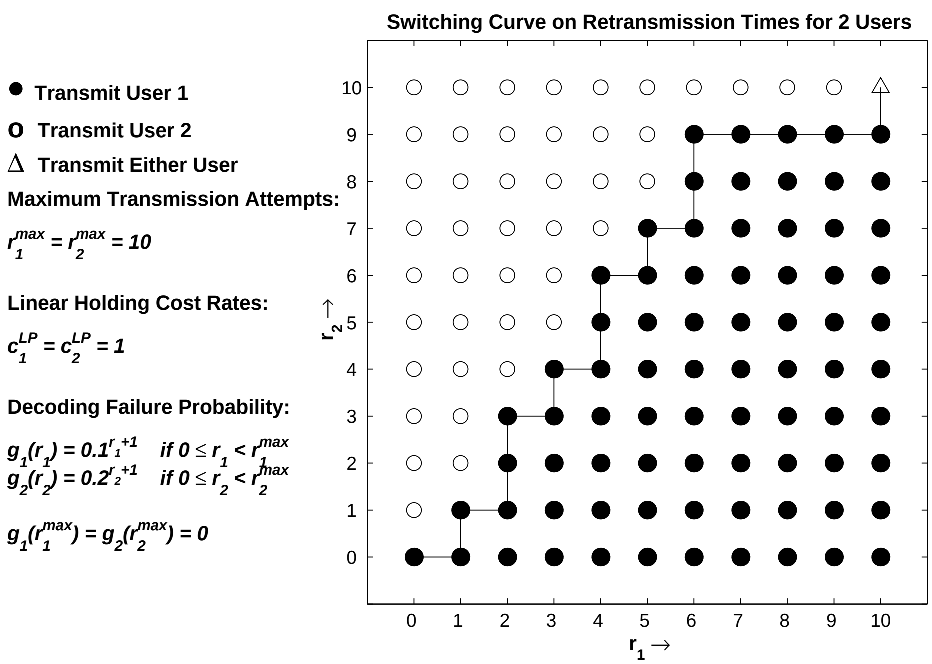

Figure 3.1: Scheduling with same linear holding cost rates and different channel conditions

If we fix the channel of user 1 and change the channel of user 2, we can see the change of packet priority due to channel changes. Figure 3.2 shows the results for this type of change. Here we fix the channel of user 1 ($\eta_1 = 0.1$) and change the channel of user 2 (η_2 changes from 0.009 to 0.4 in the direction of good channel to bad channel). The horizontal axis represents the channel condition of user 2 from the best to the worst, and the vertical axis represents the packet priority from the highest to the lowest. We use ζ_{i,r_i} to represent the priority of packet ϕ_{i,r_i}. Since $r_i^{\max} = 2$, there are all together $K = \sum_{i=1}^{2}(r_i^{\max}+1) = 6$ types of packets in the system, represented by priority 1 to 6 (from highest to lowest). When two types of packets have the same priority $j \in \{1, 2, ..., K-1\}$, we label both of them having priority j and let the priority $j+1$ empty.

From the graph we have the following observations: (1) The packets $\phi_{i,r_i^{\max}}$ always share the highest priority. (2) The priorities of the packets of user 2 decrease as the channel of user 2 get worse. (3) For user i ($i \in \{1, 2\}$), packet $\phi_{i,r_i}\phi_{i,r_i'}$ always has lower priority than over $\phi_{i,r_i'}$ if $r_i < r_i'$. All the observations agree with our optimal policy.

We can also see that the optimal scheduling rule is not very sensitive to variations of channel conditions. We say that *user i has total priority over user j* if the lowest packet priority of user i is higher than the highest packet priority of user j. Thus in this example, after excluding the packets $\phi_{i,r_i^{\max}}$, which always share the highest priorities, user 2 does not have total priority over user 1 until the channel of user 2 is significantly better than that of user 1 ($\eta_2/\eta_1 \cong 0.009/0.1 \cong 0.1$) , and user 1 do not have total priority over user 2 until the channel of user 2 is significantly worse than that of user 1 ($\eta_2/\eta_1 \cong 0.35/0.1 = 3.5$).

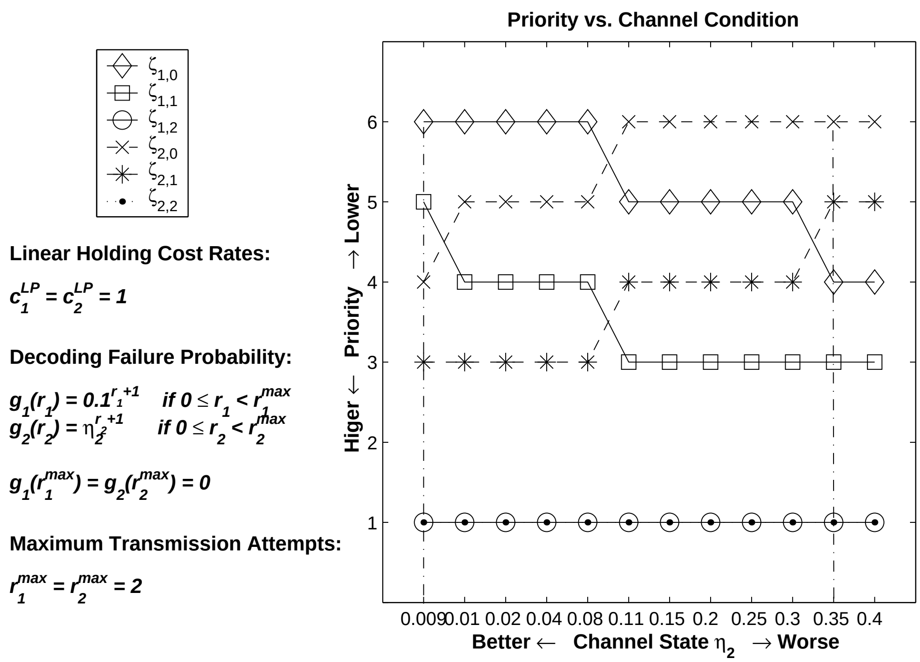

Figure 3.2: Change of scheduling priorities with channel conditions under same linear holding cost rates ($c_1^{LP} = c_2^{LP} = 1$)

According to Theorem 3.2, we determine the priority order between two packets ϕ_{i,r_i} and ϕ_{j,r_j}, by the ratio:

$$\frac{c_{i,r_i}^{LP}/T_{i,r_i}^{(\Omega)}}{c_{j,r_j}^{LP}/T_{j,r_j}^{(\Omega)}} = \frac{c_{i,r_i}^{LP}}{c_{j,r_j}^{LP}}\frac{T_{j,r_j}^{(\Omega)}}{T_{i,r_i}^{(\Omega)}} \overset{i}{\underset{j}{\gtrless}} 1 \tag{3.61}$$

When $c_{i,r_i}^{LP} = c_i^{LP} = c_j^{LP} = c_{j,r_j}^{LP}$, the above ratio is just $T_{j,r_j}^{(\Omega)}/T_{i,r_i}^{(\Omega)}$ and does not depend on the absolute value of c_i^{LP} and c_j^{LP}, thus the above sensitivity analysis of channel conditions does not depend on the values of the holding cost rates if they are the same. In figure 3.3, we let $c_1^{LP} = c_2^{LP} = 100$, and we see the same priority orders as in figure 3.2 where we have $c_1^{LP} = c_2^{LP} = 1$.

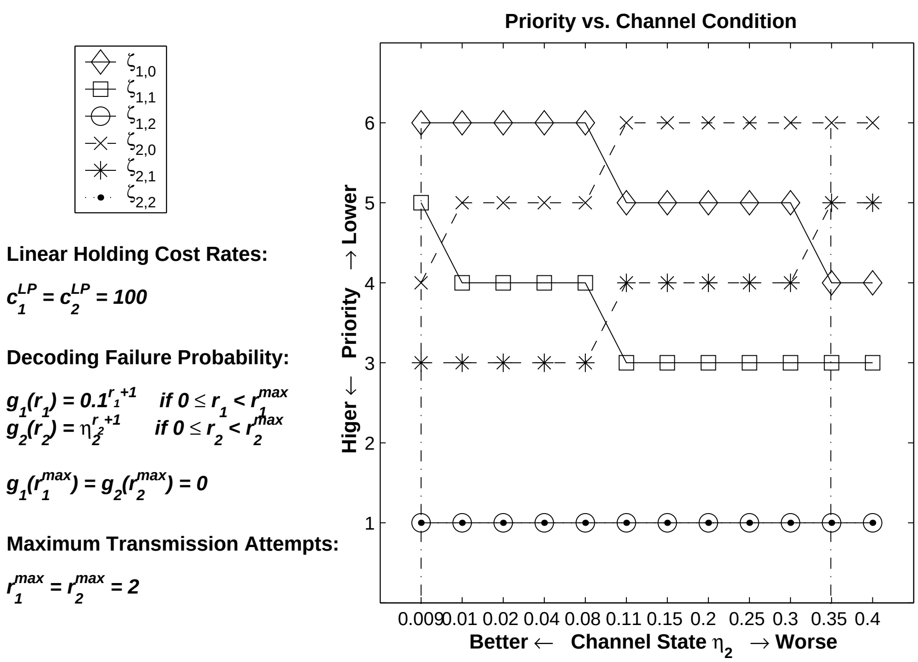

Figure 3.3: Change of scheduling priorities with channel conditions under same linear holding cost rates ($c_1^{LP} = c_2^{LP} = 100$)

Sensitivity Study: Holding Cost Rates Variation with the Same Channel Conditions

Figure 3.4 shows a typical example of the optimal scheduling policy with 2 users and different holding cost rates, but same channel conditions (i.e., the same decoding failure probabilities). Again the optimal switching policy switches on transmission attempts. As we can see the channel condition in this example is not very good (with $\eta_1 = \eta_2 = 0.1$, representing on average 10 percent of packets needed to be retransmitted after the first transmission), the delay curve is already very sensitive to holding cost rate. There are only tiny difference

$$\left(c_1^{LP} - c_2^{LP}\right)/c_1^{LP} = (1 - 0.9999)/1 = 0.01\% \tag{3.62}$$

between the holding cost rates, the optimal policy already has obvious bias over the user with higher holding cost (user 1 in this example). We can see this trend more obviously in the figure 3.5.

If we fix the holding cost rate of user 1 and change the holding cost rate of user 2, we can see the change of packet priority due to change of holding cost rate. Figure 3.5 shows the results for this type of change.

We can see that the optimal scheduling rule is very sensitive to variation in holding cost rate. User 2 has total priority over user 1 when c_2^{LP} is only slightly higher than c_1^{LP}

$$\left(c_2^{LP} - c_1^{LP}\right)/c_1^{LP} \cong (1.11 - 1)/1 = 11\% \tag{3.63}$$

and user 1 has total priority over user 2 when c_2^{LP} is only slightly lower than c_1^{LP}

$$\left(c_1^{LP} - c_2^{LP}\right)/c_1^{LP} \cong (1 - 0.89)/1 = 11\% \tag{3.64}$$

Rewrite the decision rule (3.61) as

$$\frac{c_{i,r_i}^{LP}/T_{i,r_i}^{(\Omega)}}{c_{j,r_j}^{LP}/T_{j,r_j}^{(\Omega)}} = \frac{c_{i,r_i}^{LP}}{c_{j,r_j}^{LP}}\frac{T_{j,r_j}^{(\Omega)}}{T_{i,r_i}^{(\Omega)}} \underset{j}{\overset{i}{\gtrless}} 1 \tag{3.65}$$

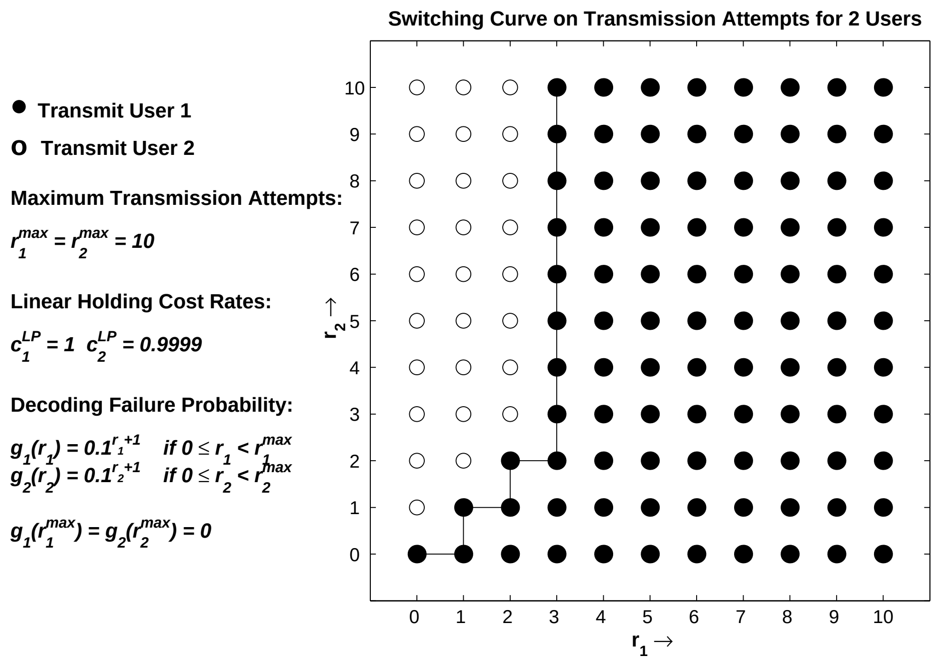

Figure 3.4: Scheduling with different linear holding cost rates and same channel conditions

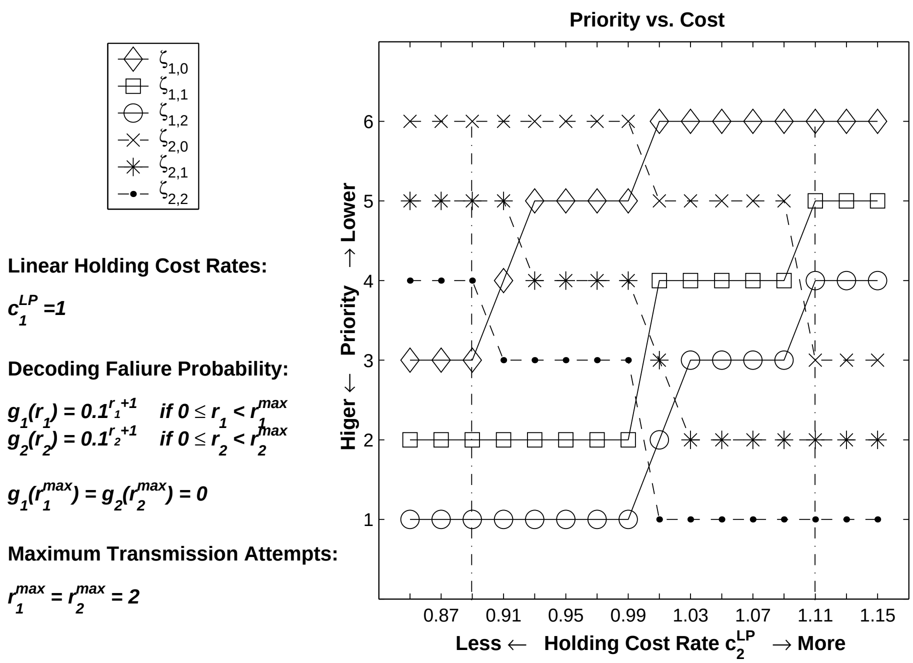

Figure 3.5: Change of scheduling priorities with holding cost rates under same channel conditions ($\eta_1 = \eta_2 = 0.1$)

If the channel condition parameter $\eta_1 = \eta_2$ approaches 0, the ratio

$$\frac{T_{j,r_j}^{(\Omega)}}{T_{i,r_i}^{(\Omega)}} = \frac{1 + \sum_{v=r_j}^{r_j^{\max}-1} \prod_{l=r_j}^{v} g_j(l)}{1 + \sum_{v=r_i}^{r_i^{\max}-1} \prod_{l=r_i}^{v} g_i(l)} = \frac{1 + \sum_{v=r_j}^{r_j^{\max}-1} \prod_{l=r_j}^{v} \eta_1^{l+1}}{1 + \sum_{v=r_i}^{r_i^{\max}-1} \prod_{l=r_i}^{v} \eta_2^{l+1}} \tag{3.66}$$

will approach 1 and the ratio in (3.65) will approach $c_{i,r_i}^{LP}/c_{j,r_j}^{LP}$, thus the priority order between two packets from two different users will only depend on relationship of the holding cost rates of each user and is not related to their transmission attempts. See figure 3.6 for the cases when $\eta_1 = \eta_2 = 0.005$.

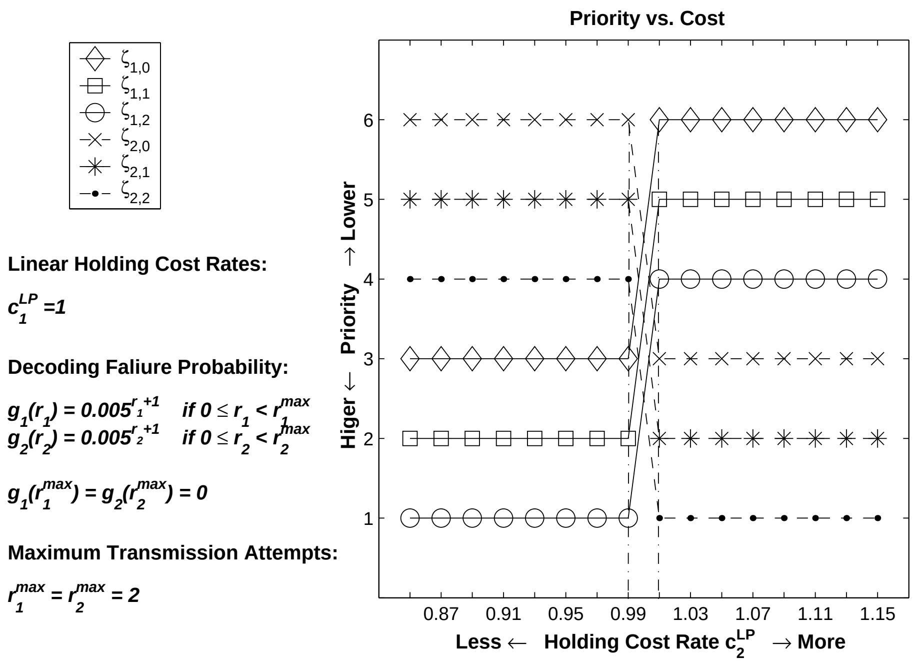

Figure 3.6: Change of scheduling priorities with holding cost rates under same channel conditions ($\eta_1 = \eta_2 = 0.01$)

In a practical communication environment, the decoding failure probability of first transmission is typically on the order of 10^{-2} or less, thus the ratio $\left(c_{i,r_i}^{LP}/T_{i,r_i}^{(\Omega)}\right) / \left(c_{j,r_j}^{LP}/T_{j,r_j}^{(\Omega)}\right)$ will be very sensitive to the variations of the holding cost rates if not totally depending on the ratio of $c^{LP}_{i,r_i}/c^{LP}_{j,r_j}$. See figure 3.7 for the cases when $\eta_1 = \eta_2 = 0.03$.

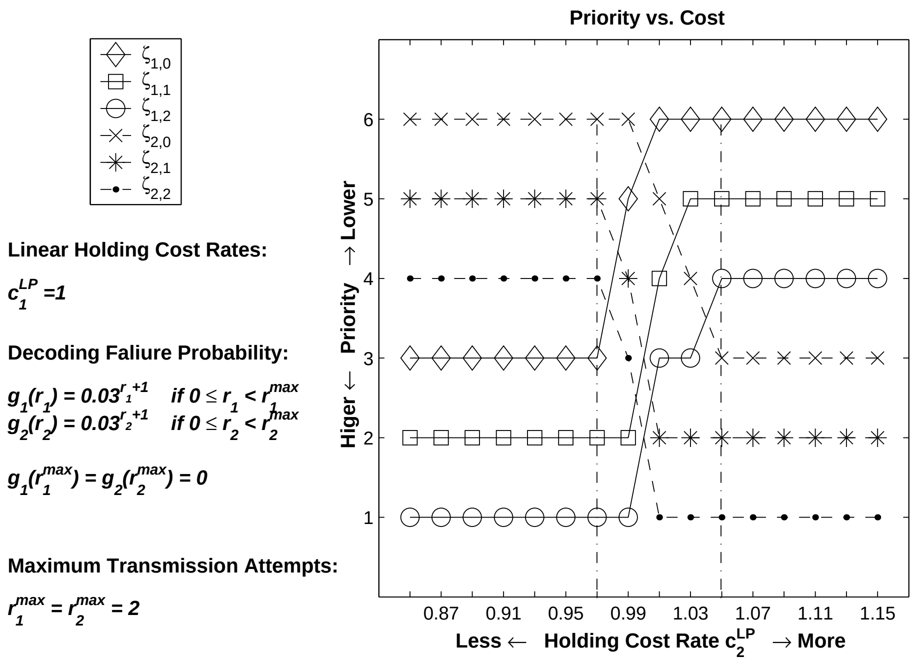

Figure 3.7: Change of scheduling priorities with holding cost rates under same channel conditions ($\eta_1 = \eta_2 = 0.03$)

General Case: Different Holding Cost Rates and Different Channel Conditions Figure 3.8 illustrates the optimal scheduling rule for two users with different holding cost rates and different channel conditions. It shows that the policy really depends on both factors and we have to take both into consideration while calculating the priorities.

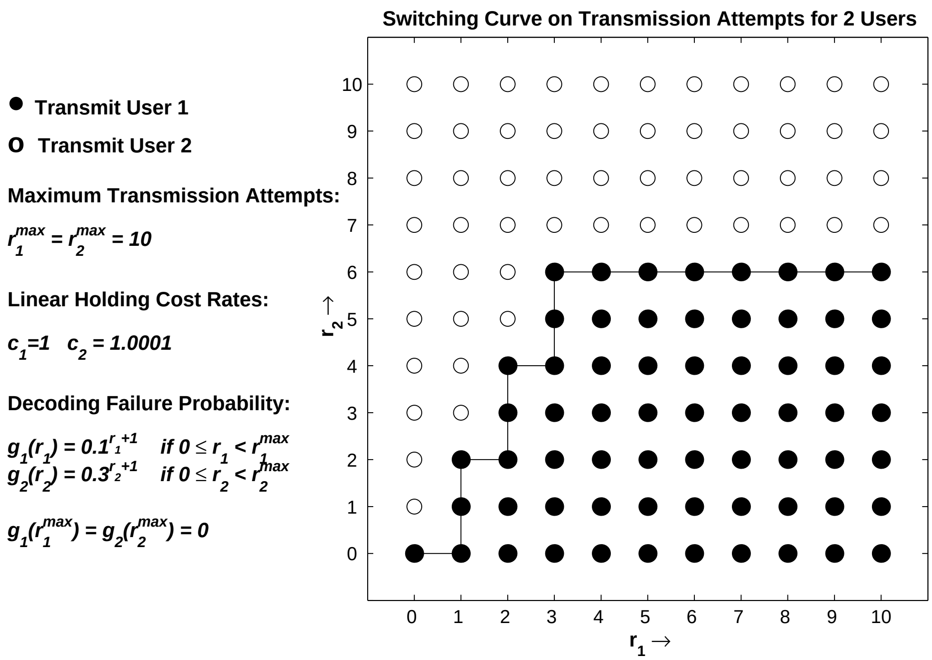

Figure 3.8: Scheduling with different linear holding cost rates and different channel conditions

Heterogeneous Holding Cost Rate within Each User

Finally we consider the case that the holding cost rate is heterogeneous within user $i \in \Xi$, that is, it depends on the transmission attempts the packet already has. This is the general assumption in this chapter:

$$c_{i,r_i}^{LP} \leq c_{i,r_i'}^{LP}, \quad \text{for} \quad r_i \leq r_i', i \in \Xi \tag{3.67}$$

which means the more transmission the packet has, the more expensive to keep it in the system. This can be true in a system where different users have different holding cost rates but each user definitely does not want to be interrupted during a retransmission. Thus we can assign higher holding cost rates for packets in retransmission so that they will have higher priority over any new packet from any user. Thus in this way, as long as the scheduler has begun transmitting a new packet, it will continue to transmit it until the packet is successfully decoded. See figure 3.9 for an example of how to guarantee the retransmission of a packet of user 2 who is in a relatively bad channel and with lower holding cost rate for new packets. For comparison, see figure 3.10 when we do not have this degree of freedom of changing the holding cost rate within user 2. Thus after the scheduler has begun transmitting a packet of user 2, before the transmission attempts of the packet reaches 5, its (re)transmission maybe stopped but any decision epoch when a new packet of user 1 arrivals at the system. At this moment, the scheduler will turn to transmit the new packet from user 1, continue to transmit until it is decoded successfully, then continue transmit the packets of user 1 until the queue for user 1 is empty, then turn back to the packet of user 2 which has been suspended before and continue to transmit it.

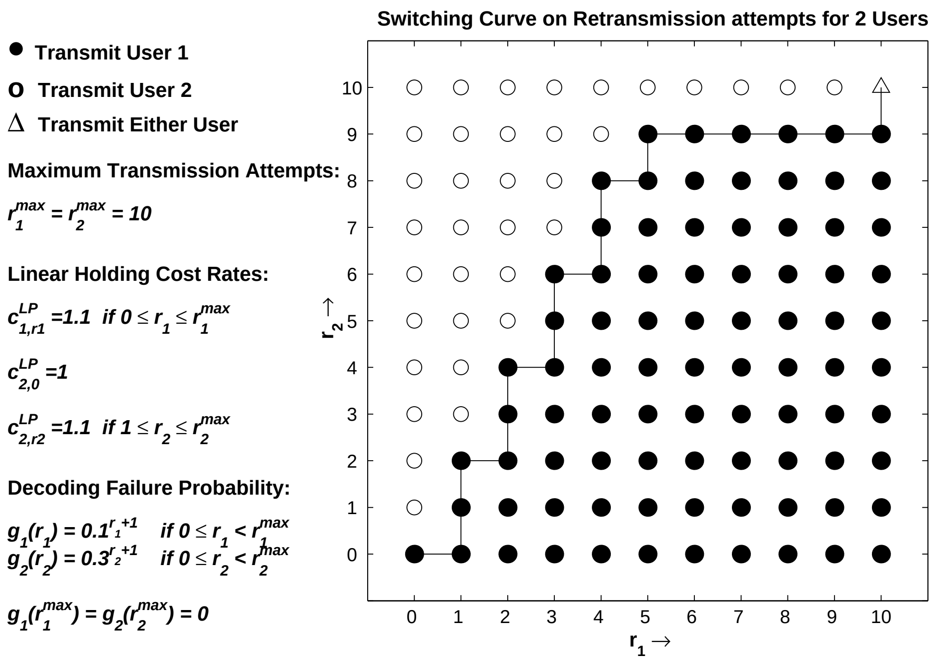

Figure 3.9: Scheduling with heterogeneous linear holding cost rates within each user

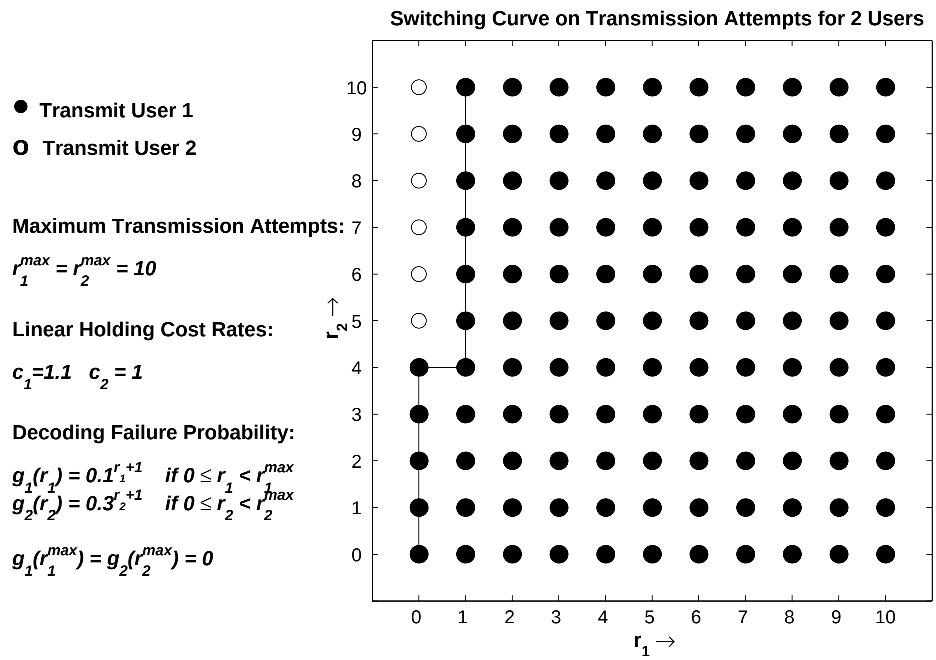

Figure 3.10: Scheduling with homogeneous linear holding cost rates within each user

3.2 Scheduling with Linear Cost Functions and Stationary Arrival Processes

For the scheduling problems with linear cost functions and stationary arrival processes, we can view it as a special case of branching bandits problem, which is defined by Weiss in [40]: there is a single server who serve several classes of jobs with a similar performance criterion as the Klimov problem. At each service completion, however, the served customer is replaced by a random number of jobs of every other class. The distribution of new jobs are determined by the the class of job that has just been served. Thus we can see that this is a more general model than the Klimov model in the sense that the arrival does not have to be Poisson.

The branching bandits problem is typically solved as the extensions of the classical multi-armed bandit problem, but also can be solved by the achievable approach, which is way of solving optimal control problem of stochastic systems by characterizing the space of all possible performances (the achievable region) of the system and optimize the overall system-wide performance objective over this space. For a review on the achievable region approach, see the paper by Dacre, Glazebrook and Niño-Mora [41].

The optimal scheduling policies for scheduling problems with linear cost functions and stationary arrival processes are again characterized by priority rules (possibly randomized, as opposed to fixed). For a detailed treatment of using the achievable region approach and its application to the branching bandits problem, see [42].

Chapter 4

Scheduling with Increasing Convex Cost Functions

In this chapter, we study scheduling problems with general nonlinear increasing convex cost functions. We first consider the case of batch arrival processes (which will be defined precisely in the next section). Surprisingly, we can again find the optimal scheduling policy for this particular model using results from the Klimov model. We then present some numerical studies illustrating the optimal policy. Finally, we briefly discuss the case with arbitrary stationary arrival processes.

4.1 Increasing Convex Cost Functions and Batch Arrival Processes

Let us first define the batch arrival process precisely: Assume there are N users in the system. For user $i \in \{1, 2, ...N\}$, the number of packets contained in one arrival is a random variable a_i^B with distribution $p_i(j) = \Pr\left(a_i^B = j\right)$ for $j = 0, ..., A_i$, where A_i is an upper bound on a_i^B.

We assume the batch arrivals for every user arrivals simultaneously, with the interarrival time between any two batch arrivals large enough such that all transmission from the prior batch are completed. In other words, the interarrival time should be greater than the longest possible total transmission time for the largest possible batch arrival, i.e. $A = \sum_{i=1}^{N} A_i$ packets, which corresponds to the event that all N users have arrivals at the same time and each of them receives the largest possible number of packets in one batch.

We first formulate the problem based on the system model described in Section 2.1. Then we show the problem can be again reformulated into a variation of the Klimov problem. This is used to find the structure of the optimal scheduling policy. Next we further simplify the problem and find the optimal scheduling policy. Finally we study the optimal scheduling policy numerically.

4.1.1 Problem Formulation: CB Scheduling Problem

We will denote the scheduling problem with increasing convex cost functions and batch arrivals as the **CB** scheduling problem.

Assume there are N users in the system, and let $\Xi = \{1, 2, ..., N\}$ be the set of users. By the queue $i \in \Xi$, we mean the queue belonging to user i at the scheduler. At each decision epoch t ($t = 0, 1, 2, ...$), denote the nonempty queues as $\Xi^{NE}(t)$. For user $i \in \Xi$, let $\Gamma_i = \{0, 1, ..., r_i^{\max}\}$ and $\chi_i = \{1, ..., A_i\}$ denote the set of transmission attempts possible and batch size, respectively.

For the HOL packet of queue $i \in \Xi$ at time t, denote its transmission attempts as $r_i^{HOL}(t)$. We will write it as r_i^{HOL} when it is clear that we are interested only in time t.

The number of packets in queue i at decision epoch t ($t = 1, 2, ...$) is $x_i(t)$. The holding cost function of user i is $U_i(\cdot)$, which is a convex increasing function of the queue length $x_i(t)$. We assume that $U_i(0) = 0$.

Assume that a batch of packets arrive at the system at time $t = t_0$ with queue lengths:

$$x(t_0) = \left\{x_i(t_0), i \in \Xi^{NE}(t_0)\right\}. \tag{4.1}$$

According to the assumptions, there is no arrival to the system during the transmissions. Our goal is to find an optimal policy $\pi^{CB} \in \Pi$ that minimizes the total expected holding cost for each batch arrival, i.e.

$$\sum_{t=t_0}^{m'} E_{\pi^{CB}}\left[\sum_{i\in\Xi^{NE}(t)} U_i(x_i(t))\right]. \tag{4.2}$$

Here Π is the set of all nonidling and nonpreemptive policies.

In the next section we will show how to reformulate the **CB** problem into a variation of a Klimov problem. We will call the formulation based on the Klimov model the CBK problem.

4.1.2 CBK Scheduling Problem

We will first define a variation of Klimov model called draining Klimov problem then we will show that the scheduling problem with increasing convex cost functions and batch arrivals can be reformulated as a draining Klimov model, and we can find the structure of the optimal scheduling policy of the problem.

Draining Klimov model

We define the draining Klimov model as a problem that has the same assumptions and structures as the Klimov model [16] except the following:

1. There are no arrivals to the system. (In the Klimov model, arrivals are Poisson.)

2. The goal is to find an optimal policy to minimize the total expected holding cost for an batch of packets initially in the system. (In the Klimov model, the goal is to find an optimal policy to minimize the long-run average expected cost incurred over an infinite horizon.)

We have the following result for the optimal scheduling policy of the draining Klimov model.

Lemma 4.1 *The optimal scheduling policy is a priority rule. The priorities of the queues can be calculated by the same algorithm used in Klimov model (*(2.18) *to* (2.21)*)*

Proof. (Outline) The basic idea behind the proof is the following: the priority rule specified by the Klimov algorithm ((2.18) − (2.21)) is optimal for the Klimov model in the sense that it minimizes the total holding cost for each busy period (the period when the scheduler is busy serving jobs between two empty states) of the system, starting from any initial system state (number and types of jobs in the system). In the draining problem the goal is to find an optimal policy to minimize the total holding cost of a special busy period (a period with no arrivals to the system) with an arbitrary initial state. Thus the priority rule specified by the Klimov algorithm must also be optimal for the draining Klimov model.

The detail proof can be easily adapted from the proof given by Nain et al. [45] for the Klimov model. ■

Next we reformulate the **CB** problem into a draining Klimov model.

CBK Scheduling Problem

In the **CB** problem, there are N queues in the system corresponding to N users. In the **CBK** problem, we view each user $i \in \Xi$ as having $K_i = (r_i^{\max} + 1)\, A_i$ queues denoted by

$$Q_i = \{Q_{i,r_i,x_i}, r_i \in \Gamma_i, x_i \in \chi_i\} \tag{4.3}$$

where i is the user index, r_i is the transmission attempts and x_i is the queue length in the **CB** problem. Thus there will be a total of

$$K = \sum_{i \in \Xi} K_i = \sum_{i \in \Xi} (r_i^{\max} + 1)\, A_i \tag{4.4}$$

queues in the **CBK** problem. Define the set of K queues as

$$\Omega = \{Q_i, i \in \Xi\} \tag{4.5}$$

We then make correspondences between the states and events of the **CB** and the **CBK** problems, as seen in Table 4.1

CB problem	**CBK** problem
x_i ($x_i \in \chi_i$) packets in the queue $i \in \Xi$ with the HOL packet having been transmitted for r_i times	a packet in queue Q_{i,r_i,x_i}
transmission of the HOL packet in queue i described above	a packet in queue Q_{i,r_i,x_i} is transmitted
The HOL packet in queue i described above has been successfully decoded and leaves the system	a packet enters queue Q_{i,r_i,x_i-1} if $x_i > 1$; nothing happens if $x_i = 1$
The HOL packet in queue i described above stays in the system because of decoding failure	a packet enters queue Q_{i,r_i+1,x_i}

Table 4.1: Comparison between the **CB** problem and **CBK** problem

Notice that for the **CBK** problem, at each decision epoch t $(t = 0, 1, 2, ...)$, for user $i \in \Xi^{NE}(t)$, there is packet in queue $Q_{i,r_i^{HOL},x_i(t)}$. Moreover, it is the only packet in the set of queues Q_i. Denote $N^{NE}(t)$ as the size of the set $\Xi^{NE}(t)$, therefore there are $N^{NE}(t)$ "packets" in the **CBK** system, and the scheduler should decide which one to serve.

Define $b_{Q_{i,r_i,x_i}}$ as the mean value of service time for queue Q_{i,r_i,x_i} and let $b_{Q_{i,r_i,x_i}} = 1$. Define

$$b_i = \left\{ b_{Q_{i,r_i,x_i}}, r_i \in \Gamma_i, x_i \in \chi_i \right\}, \tag{4.6}$$

and

$$b = \{b_i, i \in \Xi\}. \tag{4.7}$$

After a packet from queue Q_{i,r_i,x_i} $(i \in \Xi, r_i \in \Gamma_i, x_i \in \chi_i)$ has been served, the probability of entering another queue in the system or leave the system can be specified in Table 4.2:

Table 4.2:

Serve a packet from the following queue	Possible events and their probabilities
Q_{i,r_i,x_i} $(0 \le r_i < r_i^{\max}, 1 < x_i \le A_i)$	Enter Q_{i,r_i+1,x_i} w/ prob $p_{Q_{i,r_i,x_i},Q_{i,r_i+1,x_i}} = g_i(r_i)$
	Enter $Q_{i,0,x_i-1}$ w/ prob $p_{Q_{i,r_i,x_i},Q_{i,0,x_i-1}} = 1 - g_i(r_i)$
$Q_{i,r_i,1}$ $(0 \le r_i < r_i^{\max})$	Enter $Q_{i,r_i+1,1}$ w/ prob $p_{Q_{i,r_i,1},Q_{i,r_i+1,1}} = g_i(r_i)$
	Leave the system w/ prob $p_{Q_{i,r_i,1},0} = 1 - g_i(r_i)$
$Q_{i,r_i^{\max},x_i}$ $(1 < x_i \le A_i)$	Enter $Q_{i,0,x_i-1}$ w/ prob $p_{Q_{i,r_i^{\max},x_i},Q_{i,0,x_i-1}} = 1$
$Q_{i,r_i^{\max},1}$	Leave the system w/ prob $p_{Q_{i,r_i^{\max},1},0} = 1$

Table 4.3: Feedback of the **CBK** problem

The transition matrix of the system can be defined as:

$$P = \left\{ p_{Q_{i,r_i,x_i},Q_{j,r_j,x_j}}, i,j \in \Xi, r_i \in \Gamma_i, r_j \in \Gamma_j, x_i \in \chi_i, x_j \in \chi_j \right\}. \tag{4.8}$$

Denote the holding cost rate for a packet in queue Q_{i,r_i,x_i} as

$$c_{Q_{i,r_i,x_i}} = U_i\left(x_i\right) \tag{4.9}$$

Since the holding cost defined for **CB** is increasing and convex, we have

$$c_{Q_{i,r_i,x_i}} < c_{Q_{i,r_i,x_i'}} \quad \text{for } \ 1 \leq x_i < x_i' \leq A_i, i \in \Xi, r_i \in \Gamma_i, \tag{4.10}$$

and

$$c_{Q_{i,r_i,x_i+1}} - c_{Q_{i,r_i,x_i}} \leq c_{Q_{i,r_i,x_i'+1}} - c_{Q_{i,r_i,x_i'}} \ \text{ for } 1 \leq x_i < x_i' < A_i, i \in \Xi, r_i \in \Gamma_i. \tag{4.11}$$

Let $\mathbf{1}\left(\cdot\right)$ denote the indicator function so that

$$\mathbf{1}\left(Q_{i,r_i,x_i}\right) = \begin{cases} 1\,, & \text{if there is a packet in queue } Q_{i,r_i,x_i} \\ 0, & \text{if there is no packet in queue } Q_{i,r_i,x_i} \end{cases}. \tag{4.12}$$

The holding cost of user i at time t is then

$$\sum_{r_i \in \Gamma_i} \sum_{x_i \in \chi_i} \mathbf{1}\left(Q_{i,r_i,x_i}\right) c_{Q_{i,r_i,x_i}}. \tag{4.13}$$

When a batch of packets arrive at the **CB** system at time $t = t_0$ with queue lengths

$$x\left(t_0\right) = \left\{x_i\left(t_0\right), i \in \Xi^{NE}\left(t_0\right)\right\}, \tag{4.14}$$

these can be identified with a batch of arrivals to the **CBK** system with one packet in each of the following queues:

$$\left\{Q_{i,0,x_i(t_0)}, i \in \Xi^{NE}\left(t_0\right)\right\}. \tag{4.15}$$

Our goal is to find a optimal policy $\pi^{CBK} \in \Pi$ that minimizes the total expected holding cost for each batch arrival, i.e.

$$\sum_{t=t_0}^{m'} E_{\pi^{CBK}}\left[\sum_{i \in \Xi^{NE}(t)} \sum_{r_i \in \Gamma_i} \sum_{x_i \in \chi_i} \mathbf{1}\left(Q_{i,r_i,x_i}\right) c_{Q_{i,r_i,x_i}}\right]. \tag{4.16}$$

Here Π is the set of all nonidling and nonpreemptive policies.

For any set $M \subset \Omega$, let $T^{(M)}_{Q_{i,r_i,x_i}}$ be the mean total time of servicing a job (not including its waiting time) beginning with queue $Q_{i,r_i,x_i} \in M$ up to its first exit from the set M. As in Chapter 3:

$$T^{(\Omega)}_{Q_{i,r_i,x_i}} = \sum_{Q_{j,r_j,x_j} \in \Omega} p_{Q_{i,r_i,x_i},Q_{j,r_j,x_j}} T^{(\Omega)}_{Q_{j,r_j,x_j}} + b_{Q_{i,r_i,x_i}}, \quad \forall Q_{i,r_i,x_i} \in \Omega, \tag{4.17}$$

or equivalently:

$$T^{(\Omega)} = (I - P)^{-1} b. \tag{4.18}$$

Where $T^{(\Omega)} = \left\{ T^{(\Omega)}_{Q_{i,r_i,x_i}}, Q_{i,r_i,x_i} \in \Omega \right\}$. For any $M \subset \Omega$, $T^{(M)} = \left\{ T^{(M)}_{Q_{i,r_i,x_i}}, Q_{i,r_i,x_i} \in M \right\}$ are determined in a similar way, i.e.,

$$T^{(M)} = (I^{(M)} - P^{(M)})^{-1} b^{(M)}, \tag{4.19}$$

where $P^{(M)}$ is obtained by deleting the rows and columns in P corresponding to the queues in $\Omega - M$, and $b^{(M)} = \left\{ b^{(M)}_{Q_{i,r_i,x_i}}, Q_{i,r_i,x_i} \in M \right\}$ is obtained by deleting the components of the vectors b corresponding to the same queues. $I^{(M)}$ is identity matrix with its size decided by the size of set M. Notice that all the entries of $b^{(M)}$ is 1 are equal to 1 here.

CBK problem is now a draining Klimov model. We can apply the Klimov algorithm for calculating the priorities for the **CBK** problem.

Initialization:

$$M_K = \Omega,\ C^{(M_K)}_{Q_{i,r_i,x_i}} = c_{Q_{i,r_i,x_i}},\ Q_{i,r_i,x_i} \in M_K \tag{4.20}$$

Iterations:

$$\text{Step 1: } \alpha^{(M_n)} = \underset{Q_{i,r_i,x_i} \in Mn}{\arg\min} \frac{C^{(M_n)}_{Q_{i,r_i,x_i}}}{T^{(M_n)}_{Q_{i,r_i,x_i}}};\ \theta^{(M_n)} = \frac{C^{(M_n)}_{\alpha^{(M_n)}}}{T^{(M_n)}_{\alpha^{(M_n)}}} \tag{4.21}$$

$$\text{Step 2: } M_{n-1} = \Omega - \left\{\alpha^{(M_K)}, \cdots, \alpha^{(M_n)}\right\} = M_n - \left\{\alpha^{(M_n)}\right\} \tag{4.22}$$

$$\text{Step 3: } C^{(M_{n-1})}_{Q_{i,r_i,x_i}} = T^{(M_n)}_{Q_{i,r_i,x_i}} \left[\frac{C^{(M_n)}_{Q_{i,r_i,x_i}}}{T^{(M_n)}_{Q_{i,r_i,x_i}}} - \theta^{(M_n)} \right],\ Q_{i,r_i,x_i} \in M_{n-1} \tag{4.23}$$

Next we will find out the structure of the optimal scheduling policy for **CBK** problem.

Structure of the Optimal Scheduling Policy

It is useful to define $\Delta_{\left\{\gamma^*_{i,x_i}, i\in\Xi, x_i\in\chi_i\right\}} \subset \Omega$ as the set with the following properties $i \in \Xi$ and $x_i \in \chi_i$:

1. $-1 \le \gamma^*_{i,x_i} \le r_i^{\max}$.

2. If $\gamma^*_{i,x_i} = -1$, then

$$\forall\ r_i \text{ such that } 0 \le r_i \le r_i^{\max},\ Q_{i,r_i,x_i} \notin \Delta_{\left\{\gamma^*_{i,x_i}, i\in\Xi, x_i\in\chi_i\right\}} \tag{4.24}$$

   which means that there is no queue belonging to user i in the set.

3. If $0 \le \gamma^*_{i,x_i} \le r_i^{\max}$, then

$$\forall r_i \text{ such that } \gamma^*_{i,x_i} \le r_i \le r_i^{\max},\ Q_{i,r_i,x_i} \in \Delta_{\left\{\gamma^*_{i,x_i}, i\in\Xi, x_i\in\chi_i\right\}} \tag{4.25}$$

$$\forall r_i \text{ such that } \quad 0 \le r_i < \gamma^*_{i,x_i},\ Q_{i,r_i,x_i} \notin \Delta_{\left\{\gamma^*_{i,x_i}, i\in\Xi, x_i\in\chi_i\right\}} \tag{4.26}$$

Using the same notation as in the last section, we have the following results for the **CBK** scheduling problem:

Lemma 4.2 *For* ***CBK*** *scheduling problem, each set* M_n $(n = 1, 2, ..., K)$ *can be specified as* $\Delta_{\left\{\gamma_{i,x_i}^{(M_n)}, i\in\Xi, x_i\in\chi_i\right\}}$. *For all* $Q_{i,r_i,x_i} \in M_n$, *we have:*

1. $T_{Q_{i,r_i,x_i}}^{(M_n)} \geq T_{Q_{i,r'_i,x_i}}^{(M_n)} > 0$ *for* $r_i \leq r'_i \leq r_i^{\max}$.

2. $0 \leq C_{Q_{i,r_i,x_i}}^{(M_n)} \leq C_{Q_{i,r'_i,x_i}}^{(M_n)}$ *for* $r_i \leq r'_i \leq r_i^{\max}$.

3. $\alpha^{(M_n)} = \underset{Q_{i,\gamma_{i,x_i}^{(M_n)},x_i} \in M_n}{\arg\min} \left\{ \dfrac{C_{Q_{i,\gamma_{i,x_i}^{(M_n)},x_i}}^{(M_n)}}{T_{Q_{i,\gamma_{i,x_i}^{(M_n)},x_i}}^{(M_n)}} \right\}$.

Where M_n, $T_{Q_{i,r_i,x_i}}^{(M_n)}$, $C_{Q_{i,r_i,x_i}}^{(M_n)}$, $\alpha^{(M_n)}$ *are defined in* $(4.20) - (4.23)$,

Proof. Consider $n = K$. From the definition of M^K, we have $M^K = \Omega = \Delta_{\{0, i\in\Xi, x_i\in\chi_i\}} = \Delta_{\left\{\gamma_{i,x_i}^{(M_K)}, i\in\Xi, x_i\in\chi_i\right\}}$, thus $\gamma_{i,x_i}^{(M_K)} = 0$ for $i \in \Xi, x_i \in \chi_i$.

$\forall Q_{i,r_i,x_i} \in M^K$, $T_{Q_{i,r_i,x_i}}^{(M_K)} = T_{Q_{i,r_i,x_i}}^{(\Omega)}$. According to Table (4.3) and formula (4.17):

$$\begin{array}{lll} T_{Q_{i,r_i,x_i}}^{(M_K)} = 1 + g_i(r_i)\, T_{Q_{i,r_i+1,x_i}}^{(M_K)} + (1 - g_i(r_i))\, T_{Q_{i,0,x_i-1}}^{(M_K)}, & \text{for } 0 \leq r_i < r_i^{\max}, 1 < x_i \leq A_i \\ T_{Q_{i,r_i,1}}^{(M_K)} = 1 + g_i(r_i)\, T_{Q_{i,r_i+1,1}}^{(M_K)}, & \text{for } 0 \leq r_i < r_i^{\max} \\ T_{Q_{i,r_i^{\max},x_i}}^{(M_K)} = 1 + (1 - g_i(r_i))\, T_{Q_{i,0,x_i-1}}^{(M_K)}, & \text{for } 1 < x_i \leq A_i \\ T_{Q_{i,r_i^{\max},0}}^{(M_K)} = 1 & \end{array} \tag{4.27}$$

Rewrite the above equations yield:

$$T_{Q_{i,r_i,x_i}}^{(M_K)} = 1 + \sum_{j=r_i}^{r_i^{\max}-1} \prod_{l=r_i}^{j} g_i(l) + T_{Q_{i,0,x_i-1}}^{(M_K)} > 0, \quad i \in \Xi, r_i \in \Gamma_i,\ 1 < x_i \leq A_i \tag{4.28}$$

$$T_{Q_{i,r_i,1}}^{(M_K)} = 1 + \sum_{j=r_i}^{r_i^{\max}-1} \prod_{l=r_i}^{j} g_i(l), \qquad i \in \Xi, r_i \in \Gamma_i \tag{4.29}$$

We can see that $T^{(M_K)}_{Q_{i,r_i,x_i}}$ only depends on the value of $g_i(r_i)$, $g_i\left(r_i+1\right),...,g_i\left(r_i^{\max}\right)$ and $T^{(M_K)}_{Q_{i,0,x_i-1}}$(for $x_i>1$), and therefor depends on the presence of queues Q_{i,r'_i,x_i} with $r_i \le r'_i \le r_i^{\max}$ and $Q_{i,0,x_i-1}$ (for $x_i>1$). Together with (2.5), for all$Q_{i,r_i,x_i} \in M_K$,

$$T^{(M_K)}_{Q_{i,r_i,x_i}} \ge T^{(M_K)}_{Q_{i,r'_i,x_i}} > 0 \qquad \text{for} \quad r_i \le r'_i \le r_i^{\max} \tag{4.30}$$

For all $Q_{i,r_i,x_i} \in M_K$, by definition $C^{(M_K)}_{Q_{i,r_i,x_i}} = U_i\left(x_i\right)$, so that

$$0 \le C^{(M_K)}_{Q_{i,r_i,x_i}} \le C^{(M_K)}_{Q_{i,r'_i,x_i}} \quad \text{for} \quad r_i \le r'_i \le r_i^{\max} \tag{4.31}$$

Thus

$$0 \le \frac{C^{(M_K)}_{Q_{i,r_i,x_i}}}{T^{(M_K)}_{Q_{i,r_i,x_i}}} \le \frac{C^{(M_K)}_{Q_{i,r'_i,x_i}}}{T^{(M_K)}_{Q_{i,r'_i,x_i}}} \quad \text{for} \quad r_i \le r'_i \le r_i^{\max}, \tag{4.32}$$

and

$$\alpha^{(M_K)} = \underset{Q_{i,r_i,x_i} \in M_K}{\arg\min} \frac{C^{(M_K)}_{Q_{i,r_i,x_i}}}{T^{(M_K)}_{Q_{i,r_i,x_i}}} = \underset{Q_{i,0,x_i} \in M_K}{\arg\min} \frac{C^{(M_K)}_{Q_{i,0,x_i}}}{T^{(M_K)}_{Q_{i,0,x_i}}} = \underset{Q_{i,\gamma^{(M_K)}_{i,x_i},x_i} \in M_n}{\arg\min} \left\{ \frac{C^{(M_K)}_{Q_{i,\gamma^{(M_K)}_{i,x_i},x_i}}}{T^{(M_K)}_{Q_{i,\gamma^{(M_K)}_{i,x_i},x_i}}} \right\}. \tag{4.33}$$

This shows that the lemma holds for M_K.

Assume the lemma holds for M_{n+1} $(1 < n \le K)$; we then have

$$M_{n+1} = \Delta_{\left\{\gamma^{(M_{n+1})}_{i,x_i}, i \in \Xi, x_i \in \chi_i\right\}}, \tag{4.34}$$

and $\forall Q_{i,r_i,x_i} \in M_{n+1}$,

$$T^{(M_{n+1})}_{Q_{i,r_i,x_i}} \ge T^{(M_{n+1})}_{Q_{i,r'_i,x_i}} > 0 \quad \text{for} \quad r_i \le r'_i \le r_i^{\max}, \tag{4.35}$$

$$0 \le C^{(M_{n+1})}_{Q_{i,r_i,x_i}} \le C^{(M_{n+1})}_{Q_{i,r'_i,x_i}} \quad \text{for} \quad r_i \le r'_i \le r_i^{\max}, \tag{4.36}$$

$$\alpha^{(M_{n+1})} = \underset{Q_{i,\gamma^{(M_{n+1})}_{i,x_i},x_i} \in M_{n+1}}{\arg\min} \left\{ \frac{C^{(M_{n+1})}_{Q_{i,\gamma^{(M_{n+1})}_{i,x_i},x_i}}}{T^{(M_{n+1})}_{Q_{i,\gamma^{(M_{n+1})}_{i,x_i},x_i}}} \right\}. \tag{4.37}$$

Now consider M_n. Assume $\alpha^{(M_{n+1})} = Q_{\widehat{i},\gamma^{(M_{n+1})}_{\widehat{i},\widehat{x_i}},\widehat{x_i}}$. Let $\gamma^{(M_n)}_{\widehat{i},\widehat{x_i}} = \gamma^{(M_{n+1})}_{\widehat{i},\widehat{x_i}} + 1$ if $\gamma^{(M_{n+1})}_{\widehat{i},\widehat{x_i}} < r^{\max}_{\widehat{i}}$, otherwise let $\gamma^{(M_n)}_{\widehat{i},\widehat{x_i}} = -1$. Let $\gamma^{(M_n)}_{i,x_i} = \gamma^{(M_{n+1})}_{i,x_i}$ for all (i, x_i) if $i \in \Xi/\left\{\widehat{i}\right\}$ or $x_i \in \Gamma_i/\left\{\widehat{x_i}\right\}$. Thus $M_n = \Delta_{\left\{\gamma^{(M_n)}_{i,x_i}, i\in\Xi, x_i\in\chi_i\right\}}$.

For all $Q_{i,r_i,x_i} \in M_n$, from the definition of $\Delta_{\left\{\gamma^{(M_n)}_{i,x_i}, i\in\Xi, x_i\in\chi_i\right\}}$,

$$\forall r'_i \text{ such that } \gamma^{(M_n)}_{i,x_i} \le r_i \le r'_i \le r^{\max}_i,\ Q_{i,r'_i,x_i} \in \Delta_{\left\{\gamma^{(M_n)}_{i,x_i}, i\in\Xi, x_i\in\chi_i\right\}} \tag{4.38}$$

Then by the results obtained for M_K, we have

$$T^{(M_n)}_{Q_{i,r_i,x_i}} = 1 + \sum_{j=r_i}^{r^{\max}_i - 1} \prod_{l=r_i}^{j} g_i(l) + T^{(M_n)}_{Q_{i,0,x_i-1}} > 0, \quad \text{if } Q_{i,0,x_i-1} \in M_n \tag{4.39}$$

$$T^{(Mn)}_{Q_{i,r_i,1}} = 1 + \sum_{j=r_i}^{r^{\max}_i - 1} \prod_{l=r_i}^{j} g_i(l), \qquad \text{if } Q_{i,0,x_i-1} \notin M_n \tag{4.40}$$

Thus

$$T^{(M_n)}_{Q_{i,r_i,x_i}} \ge T^{(M_n)}_{Q_{i,r'_i,x_i}} > 0 \quad \text{for } r_i \le r'_i \le r^{\max}_i \tag{4.41}$$

and for all $Q_{i,r_i,x_i} \in M_n$, for $r_i \le r'_i \le r^{\max}_i$, from (4.23) (4.35) and (4.36),

$$C^{(M_n)}_{Q_{i,r'_i,x_i}} = C^{(M_{n+1})}_{Q_{i,r'_i,x_i}} - T^{(M_{n+1})}_{Q_{i,r'_i,x_i}} \theta^{(M_{n+1})} \tag{4.42}$$

$$\ge C^{(M_{n+1})}_{Q_{i,r_i,x_i}} - T^{(M_{n+1})}_{Q_{i,r'_i,x_i}} \theta^{(M_{n+1})} \tag{4.43}$$

$$\ge C^{(M_{n+1})}_{Q_{i,r_i,x_i}} - T^{(M_{n+1})}_{Q_{i,r_i,x_i}} \theta^{(M_{n+1})} \tag{4.44}$$

$$= C^{(M_n)}_{Q_{i,r_i,x_i}} \tag{4.45}$$

$$= C^{(M_{n+1})}_{Q_{i,r_i,x_i}} - T^{(M_{n+1})}_{Q_{i,r_i,x_i}} \min_{Q_{\bar{i},\overline{r_i},\overline{x_i}} \in Mn+1} \frac{C^{(M_{n+1})}_{Q_{\bar{i},\overline{r_i},\overline{x_i}}}}{T^{(M_{n+1})}_{Q_{\bar{i},\overline{r_i},\overline{x_i}}}} \tag{4.46}$$

$$\ge C^{(M_{n+1})}_{Q_{i,r_i,x_i}} - T^{(M_{n+1})}_{Q_{i,r_i,x_i}} \frac{C^{(M_{n+1})}_{Q_{i,r_i,x_i}}}{T^{(M_{n+1})}_{Q_{i,r_i,x_i}}} \tag{4.47}$$

$$= 0 \tag{4.48}$$

Thus

$$0 \le \frac{C^{(M_n)}_{Q_{i,r_i,x_i}}}{T^{(M_n)}_{Q_{i,r_i,x_i}}} \le \frac{C^{(M_n)}_{Q_{i,r'_i,x_i}}}{T^{(M_n)}_{Q_{i,r'_i,x_i}}} \quad \text{for} \quad r_i \le r'_i \le r_i^{\max} \tag{4.49}$$

and

$$\alpha^{(M_n)} = \operatorname*{arg\,min}_{Q_{i,r_i,x_i} \in M_n} \frac{C^{(M_n)}_{Q_{i,r_i,x_i}}}{T^{(M_n)}_{Q_{i,r_i,x_i}}} = \operatorname*{arg\,min}_{Q_{i,\gamma^{(M_n)}_{i,x_i},x_i} \in M_n} \left\{ \frac{C^{(M_n)}_{Q_{i,\gamma^{(M_n)}_{i,x_i},x_i}}}{T^{(M_K)}_{Q_{i,\gamma^{(M_n)}_{i,x_i},x_i}}} \right\}. \tag{4.50}$$

Therefore the lemma holds for M_n $(1 \le n \le K)$. ■

Now we can state the following result on the structure of the optimal scheduling policy of **CBK** problem:

Theorem 4.1 *For the **CBK** scheduling problem, $\forall\ i \in \Xi$ and $x_i \in \chi_i$, the priority of queue Q_{i,r_i,x_i} is lower than the priority of queue Q_{i,r'_i,x_i} if and only if $0 \le r_i < r'_i \le r_i^{\max}$.*

Proof. Assume the theorem is not true. Then $\exists\ i^* \in \Xi$, $x^*_{i^*} \in \chi_{i^*}$, $0 \le r^*_{i^*} < r^{*\prime}_{i^*}$ such that priority of queue $Q_{i^*,r^*_{I^*},x^*_{i^*}}$ is higher than the priority of $Q_{i^*,r^{*\prime}_{I^*},x^*_{i^*}}$. Also assume $Q_{i^*,r^{*\prime}_{I^*},x^*_{i^*}} = \alpha^{(M_{n^*})}$ $(1 < n \le K)$, then $Q_{i^*,r^{*\prime}_{I^*},x^*_{i^*}} \notin M_{n-1}$ but $Q_{i^*,r^*_{I^*},x^*_{i^*}} \in M_{n-1}$, thus M_{n-1} can not be characterized as a set of $\Delta_{\left\{\gamma^{(M_{n-1})}_{i,x_i},i\in\Xi,x_i\in\chi_i\right\}}$, which contradicts with Lemma 4.2. Thus the assumption is not true. ■

Theorem 4.2 *For the **CB** scheduling problem, once the base station begin to transmit a new packet for user $i \in \Xi$, it continues to transmit the packet until it is decoded successfully.*

Proof. Assume in the **CB** scheduling problem that at time t the scheduler decides to transmit a new packet for user i, this corresponds to serving a packet from queue $Q_{i,0,x_i(t)}$ in the **CBK** problem. If this packet cannot be successfully decoded after this transmission, then the this packet will leave the **CBK** system and a new packet will enter the queue $Q_{i,1,x_i(t)}$. Since all other packets in the **CBK** system stays the same except this new packet, and queue

$Q_{i,0,x_i(t)}$ has the highest priority among all the nonempty queues in the system at the previous decision epoch, then according to the theorem 4.1, queue $Q_{i,1,x_i(t)}$ has the highest priority among all the nonempty queues in the system at this decision epoch, and the scheduler will serve the packet in queue $Q_{i,1,x_i(t)}$. This will continue until after a packet leaves the **CBK** system without having feedback, which corresponds to the packet being successfully decoded in the **CB** system. The scheduler will then pick a new packet to serve according to the priorities of the remaining nonempty queues in the system. ■

This means that we do not have to consider transmission attempts while scheduling, and now scheduling decisions are only made when the previous packet has been decoded successfully.

Next we prove another structural characteristic of the optimal scheduling policy, which is described as a "monotonic switching curve on the queue length". We will formulate the scheduling problem in a Markov Decision Process (MDP) framework and prove this structure.

4.1.3 Problem Reformulated as Markov Decision Process

Here we formulate the **CB** problem with 2 users in the system as a Markov decision process in which:

System State $S = \{(r_1, r_2, x_1, x_2)\,|0 \leq r_i \leq r_i^{\max}, 0 \leq x_i \leq A_i, i \in \{1,2\}\}$ where r_1 and r_2 are the transmission attempts experienced by the HOL packets of the user 1 and user 2 respectively, and x_1 and x_2 are the queue lengths of the queues of user 1 and user 2 respectively.

Decision Epochs Since we normalize the transmission time of one packet to be 1, decision epochs are at discrete moments t ($t = t_0, t_0 + 1, t_0 + 2, ...$), where t_0 is the arrival time of a batch of packets.

Action Space $V = \{v_0, v_1, v_2\}$ where v_0 represents idling (if there is no packet in the system); v_1 represents transmitting the HOL packet from user 1 (if the queue of user 1 is not empty); v_2 represents transmitting the HOL packet from user 2 (if the queue of user 2 is not empty).

Since we want to minimize the total holding cost of serving a batch of arrivals while no arrival occurring during the service, the problem can be formulated as a stochastic shortest path problem over infinite time horizon [43]. We denote $J(r_1, r_2, x_1, x_2)$ as the optimal cost-to-go start from state (r_1, r_2, x_1, x_2), and we have the following Bellman's equations:

- Region 0 : $((x_1 = 0)\&(r_1 > 0))|((x_2 = 0)\&(r_2 > 0))$ (infeasible region)
- Region 1 : $x_1 = 0, x_2 = 0$ (empty queues)

$$J(0, 0, 0, 0) = 0 \tag{4.51}$$

- Region 2 : $x_1 > 0, x_2 = 0$ (empty queue 2)

$$J(r_1, 0, x_1, 0) = U_1(x_1) + g_1(r_1)J(r_1 + 1, 0, x_1, 0) + [1 - g_1(r_1)]J(0, 0, x_1 - 1, 0) \tag{4.52}$$

- Region 3 : $x_1 = 0, x_2 > 0$ (empty queue 1)

$$J(0, r_2, 0, x_2) = U_2(x_2) + g_2(r_2)J(0, r_2 + 1, 0, x_2) + [1 - g_2(r_2)]J(0, 0, 0, x_2 - 1) \tag{4.53}$$

- Region 4 : $x_1 > 0, x_2 > 0$

$$\begin{aligned} J(r_1, r_2, x_1, x_2) = {} & U_1(x_1) + U_2(x_2) \\ & + \min_{v \in \{v_1, v_2\}} \{g_1(r_1)J(r_1 + 1, r_2, x_1, x_2) + [1 - g_1(r_1)]J(0, r_2, x_1 - 1, x_2), \\ & g_2(r_2)J(r_1, r_2 + 1, x_1, x_2) + [1 - g_2(r_2)]J(r_1, 0, x_1, x_2 - 1)\} \end{aligned} \tag{4.54}$$

we have the following result on the structure of the optimal scheduling policy:

Theorem 4.3 *The optimal scheduling policy for **CB** is a monotonic switching curve on the queue lengths x_1 and x_2 which can be described as the following: if it is optimal to transmit user 1 (respectively user 2) at state (r_1, r_2, x_1, x_2) , then it is still optimal to transmit user 1 at state (r_1, r_2, x_1', x_2) (respectively transmit user 2 at state (r_1, r_2, x_1, x_2')) with $x_1' > x_1$ (respectively $x_2' > x_2$).*

Proof. See Appendix A. ■

In the next section we are ready to study some numerical examples, which illustrate the optimal scheduling policy.

4.1.4 Numerical Study

In this section, we study the properties of the optimal scheduling policy for the **CB** problem through some numerical examples. As we see from Theorem 4.1, we do not have to consider transmission attempts in scheduling for batch arrivals and we only have to consider the queue lengths for different users. Let us denote the priority of a queue of user i with length x_i as ζ_{i,x_i}. We again assume that the decoding failure probability of user i decreases exponentially with transmission attempts and the decrease rate is determined by the channel condition and represented by parameter η_i. That is

$$\begin{cases} g_i(r_i) = \eta_i^{r_i+1}, & 0 \le r_i < r_i^{\max} \\ g_i\left(r_i^{\max}\right) = 0 & \end{cases} \tag{4.55}$$

Different Channel Conditions with Same Cost Functions

We first consider the case that the cost functions of both users are the same, that is

$$U_i(x_i) = x_i^{\kappa_i}, \ i \in \{1, 2\} \tag{4.56}$$

with $\kappa_1 = \kappa_2$. We will see the behavior of optimal policies when users are in different channel conditions reflected in the parameters η_i ($\eta_1 \neq \eta_2$) and see how this behavior depends on the function parameters κ_i ($\kappa_1 = \kappa_2$).

Figure 4.1 shows a example with the same cost function parameters $\kappa_1 = \kappa_2 = 1.1$ but different channel condition parameters $\eta_1 = 0.01$ and $\eta_2 = 0.05$.

The horizontal (vertical respectively) axis represents the queue lengths of user 1 (user 2 respectively). The symbol (dot, circle) at the point (x_1, x_2) represents the action, a choice of HOL packet to transmit when the queue length of user 1(user 2, respectively) is x_1 (x_2, respectively). A dot means that it is optimal to schedule (or transmit) user 1, circle means that it is optimal to schedule (or transmit) user 2.

From the figure we can see that the optimal scheduling policy is a monotonic switching curve as concluded in Theorem 4.3. Each user's priority increases with its queue lengths, and the policy favors the user with the better channel condition (user 1 in this example).

If we fix the channel of user 1 and change the channel of user 2, we can see the change of priority of queues with different lengths of different users. We can plot this change vs. the channel condition for user 2 , which is reflected in parameter η_2. Based on these results, we make the following conjecture

Conjecture 4.4 *For user $i \in \{1, 2\}$, the priority of a queue of length x_i is always lower than the queue of length x_i' if $x_i < x_i'$.*

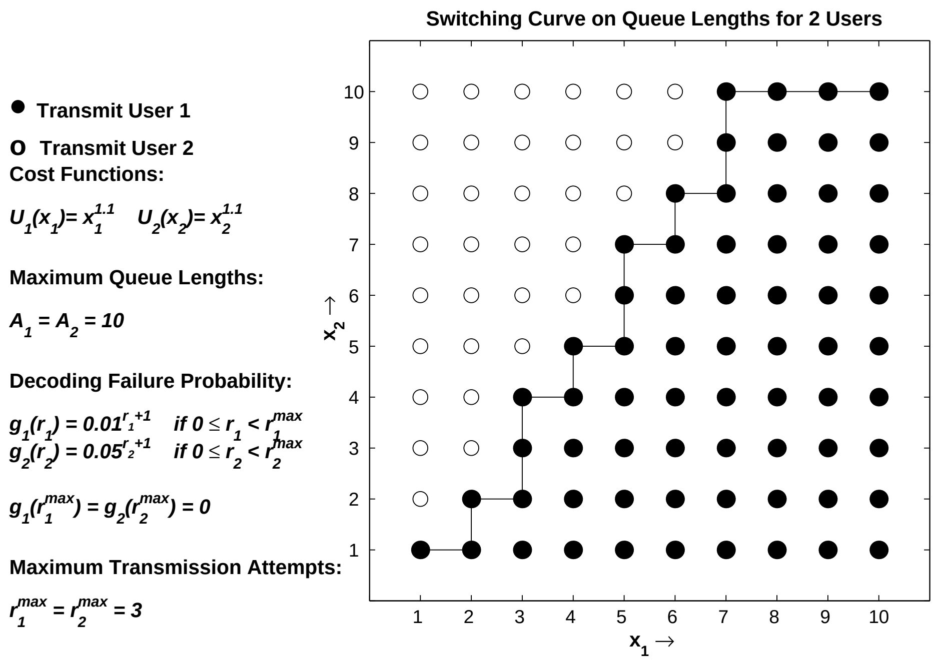

Figure 4.1: Scheduling with the same cost functions and different channel conditions

The motivation for this conjecture is twofold: (1) Since the cost is increasing convex, the changing rate of the cost function will increase with the queue length, which intuitively means that it is more expensive to keep a job in the system when the queue is long compared to when the queue is short. Thus in order to keep the total holding cost down, it make sense to give a queue with longer length higher priority within the same user. (2) Since in practice what we care about is queue priorities of different users at any decision epochs, the assumptions on the queue priorities of the same user will not affect our scheduling decision as long as we do not change the queue priorities among different users.

Figure 4.2 shows the change of queue priorities if we fix cost functions with parameters $\kappa_1 = \kappa_2 = 1.05$, maximum queue length at $A_1 = A_2 = 3$, channel condition of user 1 at $\eta_1 = 0.2$, maximum transmission attempts $r_1^{\max} = r_2^{\max} = 3$, and change the channel condition of user 2 by varying η_2. The horizontal axis represents the channel condition of user 2 from the best to the worst, and the vertical axis represents the queue priority from the highest to the lowest. We user ζ_{i,x_i} to represent the queue priority with length x_i of user i. Since $A_1 = A_2 = 3$ and we do not consider empty queues, there are all together $\sum_{i=1}^{2} A_i = 6$ types of queues in the system, represented by priority 1 to 6 (from highest to lowest). We can see that the queue priorities of user 2 decrease as the channel condition of user 2 gets worse. During a wide range of channel conditions (η_2 from 0.001 to 0.25), the queues from two users have mixed priorities which must be taken into account for optimal scheduling.

The range of queue priorities from different users depends on the cost functions each user has. Figure 4.3 shows the special case when two users have the same linear holding cost ($\kappa_1 = \kappa_2 = 1$) and all the other parameters are the same as figure (4.2). We can see that the queue priorities depend only on the user and do not mix at all with different queue lengths. (Recall that what we obtain from the simulation is the relative priority orders between different users, while the

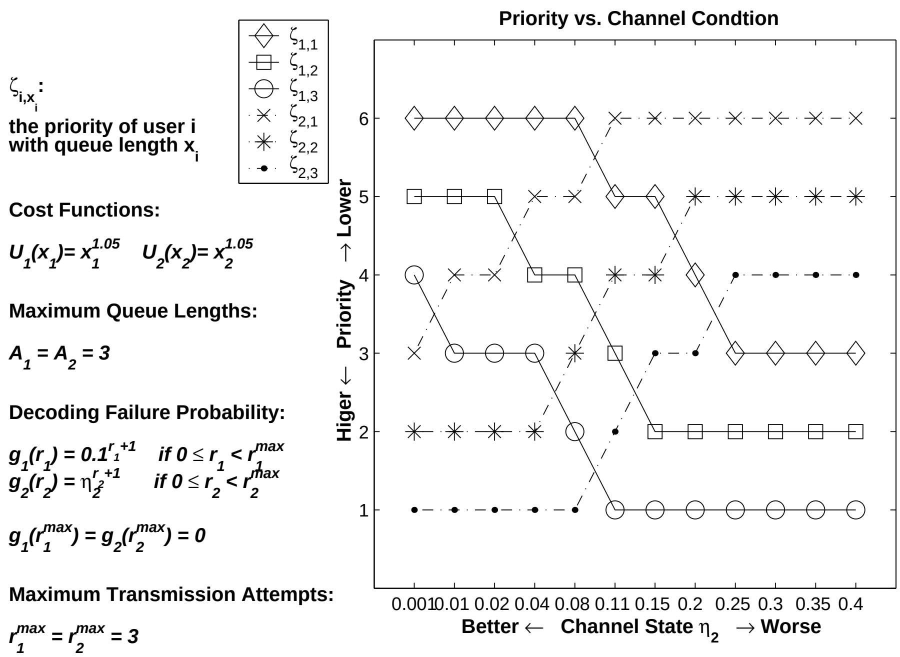

Figure 4.2: Scheduling prioirities vs. channel condition with the same cost functions ($\kappa_1 = \kappa_2 = 1.05$)

priority order for the different queue lengths associated with each user are obtained from the conjecture 4.4.) This result coincides with the result presented in chapter 3 for linear cost functions, which state that the priority of packets does not depend on the current queue length.

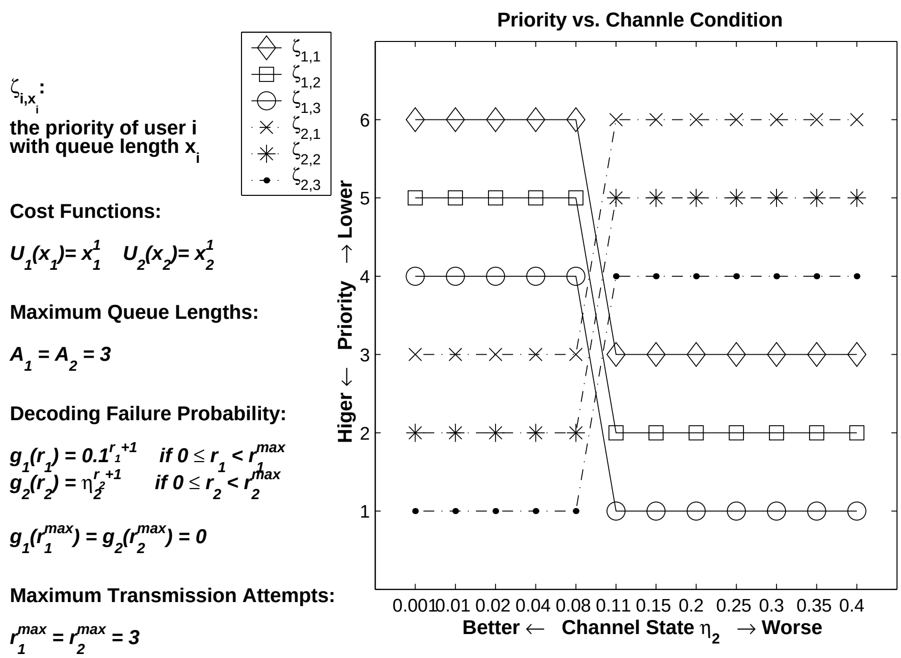

Figure 4.3: Scheduling prioirities vs. channel condition with the same cost functions ($\kappa_1 = \kappa_2 = 1$)

When the changing rate of the cost function grows fast enough with the queue length, the queue priorities mainly depends on the queue length and is relatively insensitive to different channel conditions. This is due to the very high cost of keeping a long queue in the system, and the optimal thing to do is to avoid long queues for any user. See figure 4.4, which assumes the same cost functions with parameters $\kappa_1 = \kappa_2 = 2$.

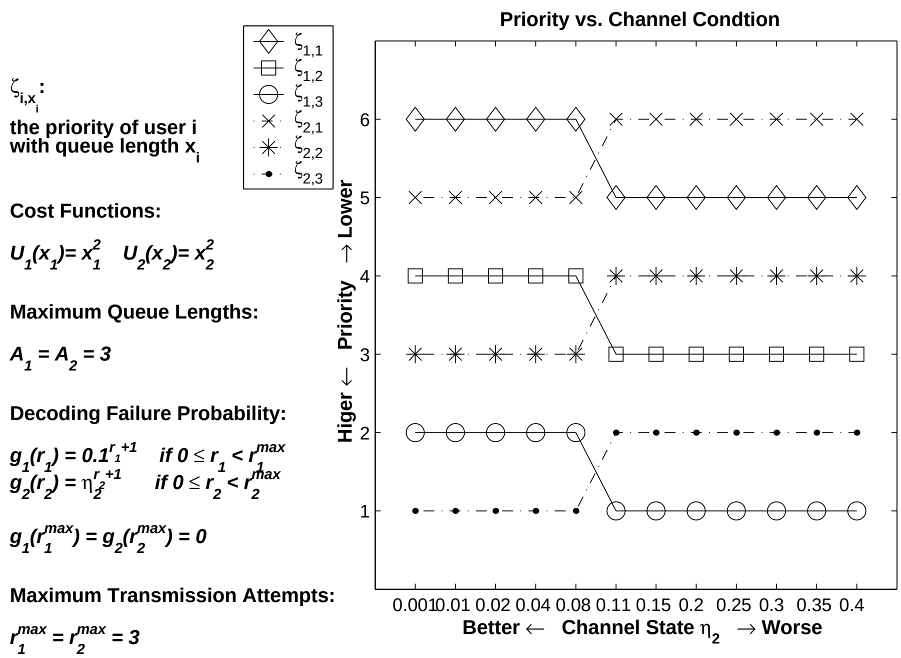

Figure 4.4: Scheduling prioirities vs. channel condition with the same cost functions ($\kappa_1 = \kappa_2 = 2$)

Different Cost Functions with Same Channel Conditions

Now we consider the cases when both users have the same channel conditions with parameters $\eta_1 = \eta_2$ but have different increasing convex cost functions on the queue lengths with parameter κ_i ($\kappa_1 \neq \kappa_2$).

Figure 4.5 shows an example with channel condition parameters $\eta_1 = \eta_2 = 0.01$ and different cost functions with parameter $\kappa_1 = 2$ and $\kappa_2 = 1.8$. The optimal scheduling policy is a monotonic switching curve increasing with queue lengths of both users and favors the user with the higher cost function (user 1 here).

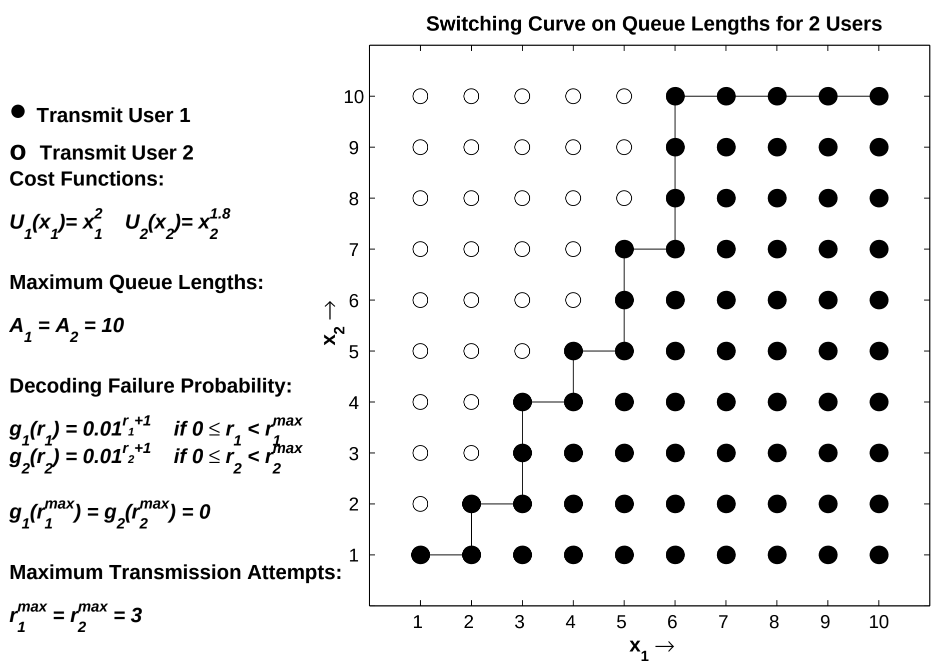

Figure 4.5: Scheduling with same channel conditions and different cost functions

If we fix the cost of user 1 and change the cost of user 2, we will see the change in queue priorities with different queue lengths of different users. Again we use conjecture 4.4 to plot the results.

Figure 4.6 shows the change of queue priorities if we fix channel conditions with parameters $\eta_1 = \eta_2 = 0.1$, maximum transmission attempts $r_1^{\max} = r_2^{\max} = 3$, cost function of user 1 with parameter $\kappa_1 = 1.5$ and change the cost function user 2 by varying parameter κ_2. We can see the queue priorities for user 2 increase as the cost function of user 2 increases.

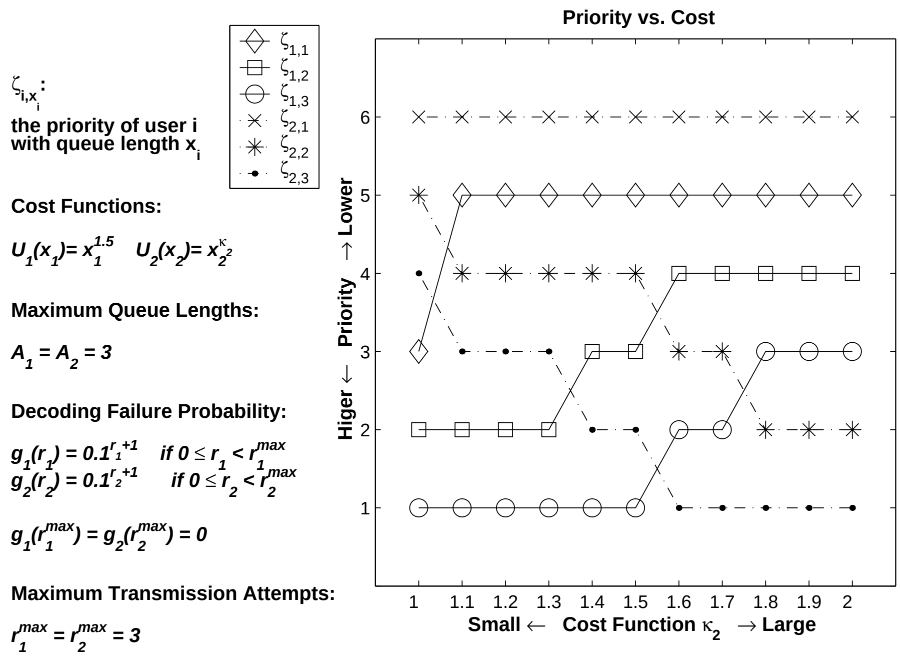

Figure 4.6: Scheduling prioirities vs. cost functions with the same channel conditions ($\eta_1 = \eta_2 = 0.1$)

The change of queue priorities here is relatively insensitive to the (same) channel conditions we assume for both users. Figure 4.7 shows an example where the channel conditions are with parameter $\eta_1 = \eta_2 = 0.01$ and all the other parameters are the same with figure 4.6. As we can see the queue priorities are the same as figure 4.6.

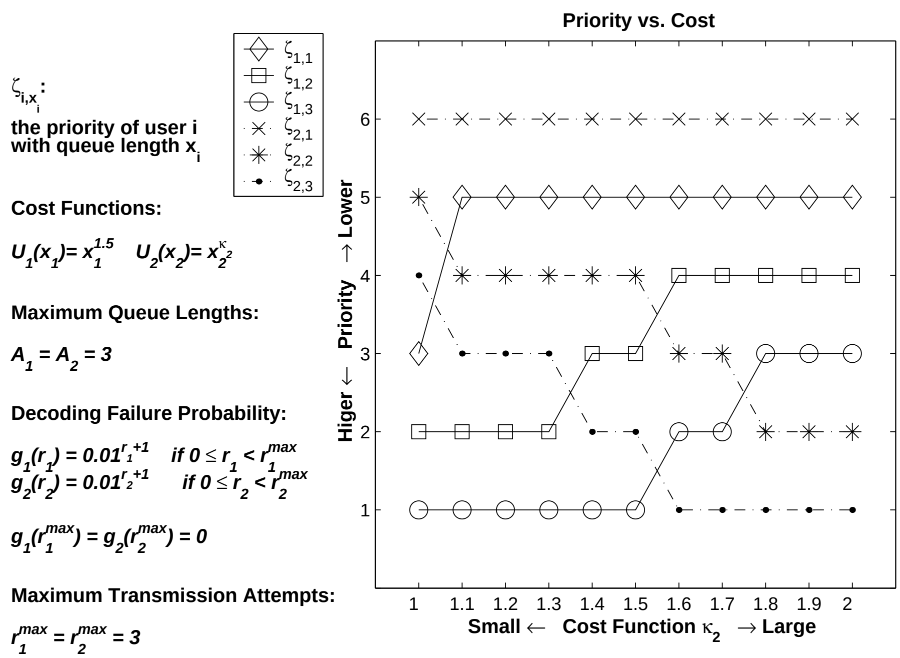

Figure 4.7: Scheduling prioirities vs. cost functions with the same channel conditions ($\eta_1 = \eta_2 = 0.01$)

General Case: Different Cost Functions and Different Channel Conditions

Figure 4.8 illustrates the optimal scheduling rule for two users with different cost functions with parameters $\kappa_1 = 2$ and $\kappa_2 = 1.8$, and different channel condition with parameters $\eta_1 = 0.1$ and $\eta_2 = 0.01$. It shows that the optimal policy is still a monotonic switching curve depending on both factors.

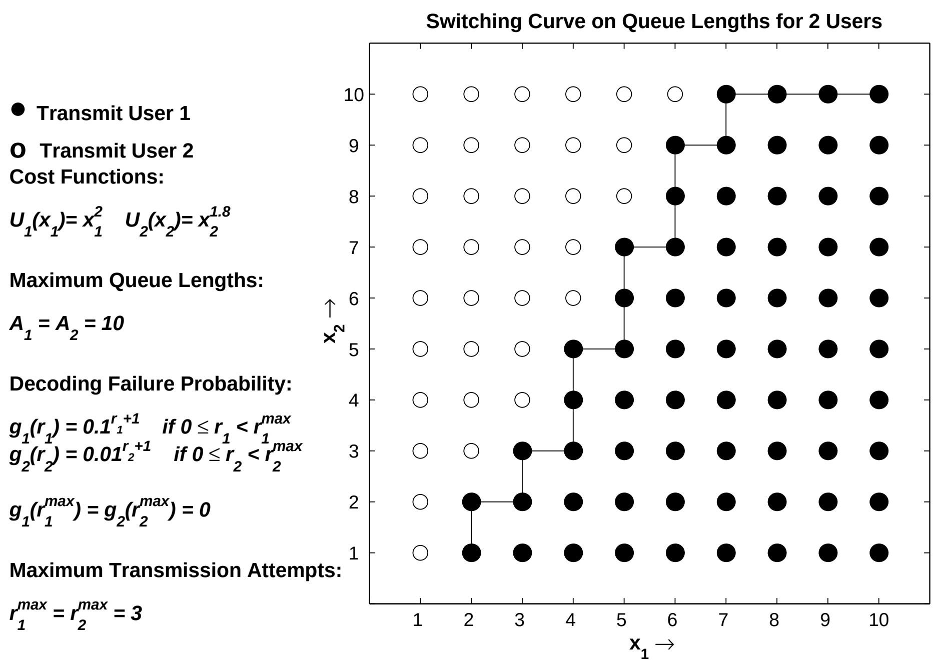

Figure 4.8: Scheduling with different cost functions and different channel conditions

4.2 Scheduling with Increasing Convex Cost Functions and Stationary Arrival Processes

For the scheduling problems with increasing convex cost functions on the queue lengths and stationary arrival processes (CCSA scheduling problem) can be viewed as a restless bandit problems as descried in chapter 1. We can reformulate the problem as a restless bandit problem (CCSARB), in analogy with the formulation of a **CBK** problem in section 4.1.2. The only difference is that in the CCSARB problem, although we can only serve one packet (user) at a time, all the other packets (users) can depart and arrive during the same time interval during to the arrival of packets in the CCSA problem. This corresponds to the restless bandit problem in that we set one bandit process to be "active" at a time and it changes its state according to "active" transition probabilities, which depend on the service and the arrival processes. All the other "passive" bandit processes can change states according to the "passive" transition probabilities, which depend only on the arrival process.

As we mentioned in chapter 1, finding an optimal rule for the general restless bandit model is an open problem. Some efforts on characterizing the performance of some heuristic scheduling rules by using linear programming relaxation/partial conservation laws are reported in [10].

Chapter 5

Conclusions and Future Work

This report has considered scheduling problems for downlink data transmission in wireless networks. There are N mobile users in the system, and the packets for different users arrive and accumulate in different queues at the base station. Packet wait to be transmitted one at a time. Each user has a unique utility function which is a decreasing concave function of the corresponding queue length. The utility function is chosen to reflect the user's QoS requirements. The objective of a scheduling policy is to choose one of the Head of Line (HOL) packets at each scheduling epoch maximize the total expected utility or the long-run average total expected utility, depending on the arrival processes at the base station.

Due to the burst error nature of wireless channels, retransmissions are inevitable. Type-II hybrid ARQ was considered so that the receiver can take advantage of the accumulated redundancy received in all packets, and thus increase the probability of decoding success. Since the packet that has not been successfully decoded by the receiver stays at the head of the queue at the base station and waits to be retransmitted with decoding success probability which depend on the number of transmissions. We have to take retransmissions into account in searching for the optimal scheduling policy. Fixed-length retransmission schemes were considered and feedback delay from the receiver to the base station was ignored.

We have studied two special cases in detail.

We first studied the case with linear utility functions and Poisson arrival processes. By treating packets with different transmission attempts from the same user as a different class of packets, we augmented the queues in the system, and formulated the problem in the framework of Klimov model [16]. The later considers a queueing network with linear holding cost rates, Poisson arrival processes and fixed transition probabilities, which determines the queue joined by the packet at the end of each scheduling epoch. Under a stability assumption given by Klimov, we showed that the optimal scheduling policy is a fixed priority rule, where the priority of the packets depends on the ratio of user's utility to the packet's transmission time. The optimal policy does not depend on the arrival process as long as the system is stable. Based on the Klimov algorithm, and by exploiting the specific structure of our problem, simple rule for computing the priority index was presented. For each class of packet corresponding to a different user or different transmission time, the priority is computed and then ordered in order to update the priorities at the decision epoch.

We then studied the case with general decreasing concave utility functions and batch arrivals with long interval arrival times. By treating queues for different user, with different lengths and different transmission attempts of the HOL packet as different classes of packets, we again augmented the queues in the system and formulated the problem in the framework of a variation of the Klimov model (which we called draining Klimov model). The optimal scheduling policy again was shown to be a strict priority rule with the priorities depending on the user's utility, the queue length and the transmission attempts of the packet. It was shown that the priority is nondecreasing with the number of transmission for a fixed user and queue length. Thus in the setting of batch arrivals with long interarrival time, we conclude that scheduling decisions only have to be made when a packet is successfully decoded at the receiver. In other words, once the base station starts transmitting a packet, it will not switch to another user until this packet has been successfully decoded. Thus we only have to consider the user's utility function and queue lengths when making scheduling decisions. We then formulated the same problem in the framework of a Markov Decision Process and found that the optimal scheduling policy has a monotonic switching curve which depend on the queue length. That is, if it is optimal to transmit to user i with queue length x_i, and if the queue lengths of all the other users are fixed, then it is also optimal to transmit to user i at queue length x_i' if $x_i' > x_i$.

Simulation results were presented for both cases discussed above. It has been observed that the priorities for of the optimal scheduling policy depend greatly on the choice of the utility functions, and are relatively insensitively to the channel variations across users. This suggests that when the user's utility functions are very different, it may be a good idea to schedule users with higher utilities and not worry about the priority changes caused by channel variations or retransmissions. When the utility functions across users are quite similar or even the same, it is very important to take retransmissions into account and differentiate users by their channel conditions.

For the general scheduling problem with decreasing concave utility functions and stationary arrival processes, we have shown how to formulate it as a restless bandit problem, which is still open in that no explicit optimal scheduling rule has been reported. In particular, the case with linear utility function and stationary arrivals can be formulated as a branching bandits problem. In this case, a fixed priority rule is optimal, and priority can be determined by a linear program. Similar extensions might be considered in future research.

There are other related areas which might be interesting for the future research. One possibility is to include feedback delay into the scheduling model. In that case, a nonidling policy may not be optimal since each transmission of a packet could prevent the base station from transmitting the same packet immediately afterwards if it is not successfully decoded at the receiver. Another issue is to include time-varying channels into the scheduling model. In that case, the scheduling policy must account for channel changes and a stationary policy may not be optimal. Also as in the prior work of [38], we can allow variable-length retransmissions, so that the scheduler must decide both to whom to transmit, and how much to transmit.

Appendix A

Proof of Theorem 4.3:

Proof. From the Bellman's equation we can see that it is sufficient to prove the following statements:

PX.1: $g_2(r_2)J(r_1, r_2+1, x_1, x_2) + [1-g_2(r_2)]J(r_1, 0, x_1, x_2-1) - \{g_1(r_1)J(r_1+1, r_2, x_1, x_2) + [1-g_1(r_1)]J(0, r_2, x_1-1, x_2)\}$ is nondecreasing in x_1 :

$$\begin{aligned}
& g_2(r_2)[J(r_1, r_2+1, x_1+1, x_2) - J(r_1, r_2+1, x_1, x_2)] \\
& + [1-g_2(r_2)][J(r_1, 0, x_1+1, x_2-1) - J(r_1, 0, x_1, x_2-1)] \\
& + g_1(r_1)[J(r_1+1, r_2, x_1, x_2) - J(r_1+1, r_2, x_1+1, x_2)] \\
& + [1-g_1(r_1)][J(0, r_2, x_1-1, x_2) - J(0, r_2, x_1, x_2)] \\
& \geq 0
\end{aligned} \tag{A.1}$$

PX.2: $g_2(r_2)J(r_1, r_2+1, x_1, x_2) + [1-g_2(r_2)]J(r_1, 0, x_1, x_2-1) - \{g_1(r_1)J(r_1+1, r_2, x_1, x_2) + [1-g_1(r_1)]J(0, r_2, x_1-1, x_2)\}$ is nonincreasing in x_2 :

$$
\begin{aligned}
& g_1(r_1)[J(r_1+1,r_2,x_1,x_2+1) - J(r_1+1,r_2,x_1,x_2)] \\
& + [1-g_1(r_1)][J(0,r_2,x_1-1,x_2+1) - J(0,r_2,x_1-1,x_2)] \\
& + g_2(r_2)[J(r_1,r_2+1,x_1,x_2) - J(r_1,r_2+1,x_1,x_2+1)] \\
& + [1-g_2(r_2)][J(r_1,0,x_1,x_2-1) - J(r_1,0,x_1,x_2)] \\
& \geq 0 \qquad \text{(A.2)}
\end{aligned}
$$

These statements are trivially true for value iteration at $k = 0$ where we assume that all cost-to-go functions are 0. Assume they hold at iteration $k-1$, look at iteration k:

Proof for **PX.1**:

$$
\begin{aligned}
& g_2(r_2)J(r_1,r_2+1,x_1,x_2) + [1-g_2(r_2)]J(r_1,0,x_1,x_2-1) \\
& - \{g_1(r_1)J(r_1+1,r_2,x_1,x_2) + [1-g_1(r_1)]J(0,r_2,x_1-1,x_2)\} \qquad \text{(A.3)}
\end{aligned}
$$

is nondecreasing in x_1 for $x_1 \geq 1, x_2 \geq 1$.

$$
\begin{aligned}
& g_2(r_2)[J_k(r_1,r_2+1,x_1+1,x_2) - J_k(r_1,r_2+1,x_1,x_2)] \\
& + [1-g_2(r_2)][J_k(r_1,0,x_1+1,x_2-1) - J_k(r_1,0,x_1,x_2-1)] \\
& + g_1(r_1)[J_k(r_1+1,r_2,x_1,x_2) - J_k(r_1+1,r_2,x_1+1,x_2)] \\
& + [1-g_1(r_1)][J_k(0,r_2,x_1-1,x_2) - J_k(0,r_2,x_1,x_2)] \\
& \geq 0 \qquad \text{(A.4)}
\end{aligned}
$$

we first consider the cases when $x_1 > 1$ and $x_2 > 1$, then we consider when $x_1 = 1$ and/or $x_2 = 1$.

PX.1-Case I: First we consider the cases when $x_1 > 1$ and $x_2 > 1$.

For convenience, we define:

$$A = A.1 - A.2 = J_k(r_1, r_2+1, x_1+1, x_2) - J_k(r_1, r_2+1, x_1, x_2) \tag{A.5}$$

$$B = B.1 - B.2 = J_k(r_1, 0, x_1+1, x_2-1) - J_k(r_1, 0, x_1, x_2-1) \tag{A.6}$$

$$C = C.1 - C.2 = J_k(r_1+1, r_2, x_1, x_2) - J_k(r_1+1, r_2, x_1+1, x_2) \tag{A.7}$$

$$D = D.1 - D.2 = J_k(0, r_2, x_1-1, x_2) - J_k(0, r_2, x_1, x_2) \tag{A.8}$$

We need to show that:

$$g_2(r_2)A + [1 - g_2(r_2)]B + g_1(r_1)C + [1 - g_1(r_1)]D \geq 0 \tag{A.9}$$

Using the optimality equation, we have:

$$\begin{aligned}
A = {} & U_1(x_1+1) - U_1(x_1) \\
& + \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) + [1 - g_1(r_1)]J_{k-1}(0, r_2+1, x_1, x_2) \\ g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1+1, x_2) + [1 - g_2(r_2+1)]J_{k-1}(r_1, 0, x_1+1, x_2-1) \end{cases} \\
& - \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + [1 - g_1(r_1)]J_{k-1}(0, r_2+1, x_1-1, x_2) \\ g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) + [1 - g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1) \end{cases}
\end{aligned} \tag{A.10}$$

$$\begin{aligned}
B = {} & U_1(x_1+1) - U_1(x_1) \\
& + \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1 - g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1) \\ g_2(0)J_{k-1}(r_1, 1, x_1+1, x_2-1) + [1 - g_2(0)]J_{k-1}(r_1, 0, x_1+1, x_2-2) \end{cases} \\
& - \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) + [1 - g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1) \\ g_2(0)J_{k-1}(r_1, 1, x_1, x_2-1) + [1 - g_2(0)]J_{k-1}(r_1, 0, x_1, x_2-2) \end{cases}
\end{aligned} \tag{A.11}$$

$$C = U_1(x_1) - U_1(x_1+1)$$
$$+ \min_{v\in\{v_1,v_2\}} \begin{cases} g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1-1, x_2) \\ g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + [1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1, x_2-1) \end{cases}$$
$$- \min_{v\in\{v_1,v_2\}} \begin{cases} g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1+1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1, x_2) \\ g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) + [1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1+1, x_2-1) \end{cases} \tag{A.12}$$

$$D = U_1(x_1-1) - U_1(x_1)$$
$$+ \min_{v\in\{v_1,v_2\}} \begin{cases} g_1(0)J_{k-1}(1, r_2, x_1-1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-2, x_2) \\ g_2(r_2)J_{k-1}(0, r_2+1, x_1-1, x_2) + [1-g_2(r_2)]J_{k-1}(0, 0, x_1-1, x_2-1) \end{cases}$$
$$- \min_{v\in\{v_1,v_2\}} \begin{cases} g_1(0)J_{k-1}(1, r_2, x_1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-1, x_2) \\ g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) + [1-g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1) \end{cases} \tag{A.13}$$

Furthermore, we let:

$$A' = A'.1 - A'.2 \tag{A.14}$$
$$= \min_{v\in\{v_1,v_2\}} \begin{cases} A'.1.v1 \\ A'.1.v2 \end{cases} - \min_{v\in\{v_1,v_2\}} \begin{cases} A'.2.v1 \\ A'.2.v2 \end{cases}$$
$$= \min_{v\in\{v_1,v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) + [1-g_1(r_1)]J_{k-1}(0, r_2+1, x_1, x_2) \\ g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1+1, x_2) + [1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1+1, x_2-1) \end{cases}$$
$$- \min_{v\in\{v_1,v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + [1-g_1(r_1)]J_{k-1}(0, r_2+1, x_1-1, x_2) \\ g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) + [1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1) \end{cases} \tag{A.15}$$

$$
\begin{aligned}
B' &= B'.1 - B'.2 \\
&= \min_{v \in \{v_1, v_2\}} \begin{cases} B'.1.v1 \\ B'.1.v2 \end{cases} - \min_{v \in \{v_1, v_2\}} \begin{cases} B'.2.v1 \\ B'.2.v2 \end{cases} \\
&= \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1) \\ g_2(0)J_{k-1}(r_1, 1, x_1+1, x_2-1) + [1-g_2(0)]J_{k-1}(r_1, 0, x_1+1, x_2-2) \end{cases} \\
&- \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1) \\ g_2(0)J_{k-1}(r_1, 1, x_1, x_2-1) + [1-g_2(0)]J_{k-1}(r_1, 0, x_1, x_2-2) \end{cases}
\end{aligned} \tag{A.16}
$$

$$
\begin{aligned}
C' &= C'.1 - C'.2 \\
&= \min_{v \in \{v_1, v_2\}} \begin{cases} C'.1.v1 \\ C'.1.v2 \end{cases} - \min_{v \in \{v_1, v_2\}} \begin{cases} C'.2.v1 \\ C'.2.v2 \end{cases} \\
&= \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1-1, x_2) \\ g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + [1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1, x_2-1) \end{cases} \\
&- \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1+1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1, x_2) \\ g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) + [1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1+1, x_2-1) \end{cases}
\end{aligned} \tag{A.17}
$$

$$
\begin{aligned}
D' &= D'.1 - D'.2 \\
&= \min_{v \in \{v_1, v_2\}} \begin{cases} D'.1.v1 \\ D'.1.v2 \end{cases} - \min_{v \in \{v_1, v_2\}} \begin{cases} D'.2.v1 \\ D'.2.v2 \end{cases} \\
&+ \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(0)J_{k-1}(1, r_2, x_1-1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-2, x_2) \\ g_2(r_2)J_{k-1}(0, r_2+1, x_1-1, x_2) + [1-g_2(r_2)]J_{k-1}(0, 0, x_1-1, x_2-1) \end{cases} \\
&- \min_{v \in \{v_1, v_2\}} \begin{cases} g_1(0)J_{k-1}(1, r_2, x_1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-1, x_2) \\ g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) + [1-g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1) \end{cases}
\end{aligned} \tag{A.18}
$$

Then:

$$
\begin{aligned}
& g_2(r_2)A + [1 - g_2(r_2)]B + g_1(r_1)C + [1 - g_1(r_1)]D \\
& = g_2(r_2)[U_1(x_1+1) - U_1(x_1)] + [1 - g_2(r_2)][U_1(x_1+1) - U_1(x_1)] \\
& + g_1(r_1)[U_1(x_1) - U_1(x_1+1)] + [1 - g_1(r_1)][U_1(x_1-1) - U_1(x_1)] \\
& + g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
& = [1 - g_1(r_1)]\{[U_1(x_1+1) - U_1(x_1)] - [U_1(x_1) - U_1(x_1-1)]\} \\
& + g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \qquad \text{(A.19)}
\end{aligned}
$$

According to assumption $A.2$ (the convexity of $U_i(x_i)$), the first term is nonnegative. Therefore, we only need to show that

$$g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \geq 0 \qquad \text{(A.20)}$$

Expand the equation:

$$
\begin{aligned}
& g_2(r_2)[\min_{v\in\{v_1,v_2\}}(A'.1.v1, A'.1.v2) - \min_{v\in\{v_1,v_2\}}(A'.2.v1, A'.2.v2)] \\
& + [1 - g_2(r_2)][\min_{v\in\{v_1,v_2\}}(B'.1.v1, B'.1.v2) - \min_{v\in\{v_1,v_2\}}(B'.2.v1, B'.2.v2)] \\
& + g_1(r_1)[\min_{v\in\{v_1,v_2\}}(C'.1.v1, C'.2.v2) - \min_{v\in\{v_1,v_2\}}(C'.2.v1, C'.2.v2)] \\
& + [1 - g_1(r_1)][\min_{v\in\{v_1,v_2\}}(D'.1.v1, D'.1.v2) - \min_{v\in\{v_1,v_2\}}(D'.2.v1, D'.2.v2)] \\
& \geq 0 \qquad \text{(A.21)}
\end{aligned}
$$

In order to get rid of all the min's, we have assume relations between the 8 pairs, which have all together $2^8 = 256$ possibilities. Based on the induction assumptions and theorem 4.1 , we can write down the relationship among the 8 states so that we can eliminate some impossible combinations. In the following table,1* represent that we fix the action of the the corresponding state to be 1, and all the other 1 represent the corresponding action triggered by the 1^* in the same column:

$A'.1$	(r_1, r_2+1, x_1+1, x_2)	1^*	1						
$A'.2$	(r_1, r_2+1, x_1, x_2)		1^*						
$B'.1$	$(r_1, 0, x_1+1, x_2-1)$	1	1	1^*	1			1	1
$B'.2$	$(r_1, 0, x_1, x_2-1)$		1		1^*			1	1
$C'.1$	(r_1+1, r_2, x_1, x_2)		1			1^*		1	1
$C'.2$	(r_1+1, r_2, x_1+1, x_2)	1	1			1	1^*	1	1
$D'.1$	$(0, r_2, x_1-1, x_2)$							1^*	
$D'.2$	$(0, r_2, x_1, x_2)$							1	1^*

Then we can limit our attention to the following 20 cases:

Case Number		1	2	3	4	5	6	7	8	9	10	11
$A'.1$	(r_1, r_2+1, x_1+1, x_2)	1	1	1	1	1	1	1	1	1	2	2
$B'.1$	$(r_1, 0, x_1+1, x_2-1)$	1	1	1	1	1	1	1	1	1	1	1
$C'.1$	(r_1+1, r_2, x_1, x_2)	1	1	1	1	1	1	1	2	2	1	1
$D'.1$	$(0, r_2, x_1-1, x_2)$	1	1	2	2	2	2	2	2	2	1	2
$A'.2$	(r_1, r_2+1, x_1, x_2)	1	2	1	2	1	2	2	2	2	2	2
$B'.2$	$(r_1, 0, x_1, x_2-1)$	1	1	1	1	1	1	2	1	2	1	1
$C'.2$	(r_1+1, r_2, x_1+1, x_2)	1	1	1	1	1	1	1	1	1	1	1
$D'.2$	$(0, r_2, x_1, x_2)$	1	1	1	1	2	2	2	2	2	1	2

Case Number		12	13	14	15	16	17	18	19	20
$A'.1$	(r_1, r_2+1, x_1+1, x_2)	2	2	2	2	2	2	2	2	2
$B'.1$	$(r_1, 0, x_1+1, x_2-1)$	1	1	1	1	1	1	2	2	2
$C'.1$	(r_1+1, r_2, x_1, x_2)	1	1	2	2	2	2	1	2	2
$D'.1$	$(0, r_2, x_1-1, x_2)$	2	2	2	2	2	2	2	2	2
$A'.2$	(r_1, r_2+1, x_1, x_2)	2	2	2	2	2	2	2	2	2
$B'.2$	$(r_1, 0, x_1, x_2-1)$	1	2	2	2	1	1	2	2	2
$C'.2$	(r_1+1, r_2, x_1+1, x_2)	1	1	2	1	2	1	1	2	1
$D'.2$	$(0, r_2, x_1, x_2)$	1	2	2	2	2	2	2	2	2

PX.1-Case I-1:

$$
\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v1 - A'.2.v1) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1-g_1(r_1)](D'.1.v1 - D'.2.v1) \\
=&g_2(r_2)[g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) + [1-g_1(r_1)]J_{k-1}(0, r_2+1, x_1, x_2)] \\
&- g_2(r_2)[g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + [1-g_1(r_1)]J_{k-1}(0, r_2+1, x_1-1, x_2)] \\
&+ [1-g_2(r_2)][g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1)] \\
&- [1-g_2(r_2)][g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1)] \\
&+ g_1(r_1)[g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1-1, x_2)] \\
&- g_1(r_1)[g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1+1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1, x_2)] \\
&+ [1-g_1(r_1)][g_1(0)J_{k-1}(1, r_2, x_1-1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-2, x_2)] \\
&- [1-g_1(r_1)][g_1(0)J_{k-1}(1, r_2, x_1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-1, x_2)]
\end{aligned} \tag{A.22}
$$

Rearrange terms,

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=& g_1(r_1)\{g_2(r_2)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)] \\
& + [1-g_2(r_2)][J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)] \\
& - g_1(r_1+1)[J_{k-1}(r_1+2, r_2, x_1+1, x_2 - J_{k-1}(r_1+2, r_2, x_1, x_2))] \\
& - [1-g_1(r_1+1)][J_{k-1}(0, r_2, x_1, x_2) - J_{k-1}(0, r_2, x_1-1, x_2)]\} \\
& + [1-g_1(r_1)]\{g_2(r_2)[J_{k-1}(0, r_2+1, x_1, x_2) - J_{k-1}(0, r_2+1, x_1-1, x_2)] \\
& + [1-g_2(r_2)][J_{k-1}(0, 0, x_1, x_2-1) - J_{k-1}(0, 0, x_1-1, x_2-1)] \\
& - g_1(0)[J_{k-1}(1, r_2, x_1, x_2) - J_{k-1}(1, r_2, x_1-1, x_2)] \\
& - [1-g_1(r_1)][J_{k-1}(0, r_2, x_1-1, x_2) - J_{k-1}(0, r_2, x_1-2, x_2)]
\end{aligned}
\tag{A.23}
$$

Let

$$
\begin{aligned}
& PX.1\_I.1.1 \\
=& g_2(r_2)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)] \\
& + [1-g_2(r_2)][J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)] \\
& - g_1(r_1+1)[J_{k-1}(r_1+2, r_2, x_1+1, x_2 - J_{k-1}(r_1+2, r_2, x_1, x_2))] \\
& - [1-g_1(r_1+1)][J_{k-1}(0, r_2, x_1, x_2) - J_{k-1}(0, r_2, x_1-1, x_2)]
\end{aligned}
\tag{A.24}
$$

and

$$
\begin{aligned}
& PX.1\_I.1.2 \\
=& g_2(r_2)[J_{k-1}(0, r_2+1, x_1, x_2) - J_{k-1}(0, r_2+1, x_1-1, x_2)] \\
& + [1-g_2(r_2)][J_{k-1}(0, 0, x_1, x_2-1) - J_{k-1}(0, 0, x_1-1, x_2-1)] \\
& - g_1(0)[J_{k-1}(1, r_2, x_1, x_2) - J_{k-1}(1, r_2, x_1-1, x_2)] \\
& - [1-g_1(r_1)][J_{k-1}(0, r_2, x_1-1, x_2) - J_{k-1}(0, r_2, x_1-2, x_2)] \qquad \text{(A.25)}
\end{aligned}
$$

According to the induction assumption of **PX.1,** we have:

$$PX.1\_I\_1.1 \geq 0 \quad \text{and} \quad PX.1\_I\_1.2 \geq 0 \qquad \text{(A.26)}$$

Thus

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
& = g_1(r_1)PX.1\_I\_1.1 + [1-g_1(r_1)]PX.1\_I\_1.2 \geq 0 \qquad \text{(A.27)}
\end{aligned}
$$

This completes the proof for **PX.1-Case I-1.**

<u>PX.1-Case I-2:</u> since we take action 2 for state $A'.2$, we have

$$A'.2.v1 \geq A'.2.v2 \qquad \text{(A.28)}$$

Combined with the conclusion of **PX.1-Case I-1,** we have

$$\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v1 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1-g_1(r_1)](D'.1.v1 - D'.2.v1) \\
\geq&g_2(r_2)(A'.1.v1 - A'.2.v1) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1-g_1(r_1)](D'.1.v1 - D'.2.v1) \\
\geq&0
\end{aligned} \tag{A.29}$$

This completes the proof for **PX.1-Case I-2.**

PX.1-Case I-3:

Rearrange terms:

$$\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=&g_1(r_1)\{g_2(r_2)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)] \\
&+ [1-g_2(r_2)][J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)] \\
&- g_1(r_1+1)[J_{k-1}(r_1+2, r_2, x_1+1, x_2 - J_{k-1}(r_1+2, r_2, x_1, x_2))] \\
&- [1-g_1(r_1+1)][J_{k-1}(0, r_2, x_1, x_2) - J_{k-1}(0, r_2, x_1-1, x_2)]\} \\
&+ [1-g_1(r_1)]\{g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) + [1-g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1) \\
&- g_1(0)J_{k-1}(1, r_2, x_1, x_2) - [1-g_1(0)]J_{k-1}(0, r_2, x_1-1, x_2)
\end{aligned} \tag{A.30}$$

Let:

$$\begin{aligned}
&PX.1_I_3.1\\
=&g_2(r_2)[J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)-J_{k-1}(r_1+1,r_2+1,x_1,x_2)]\\
&+[1-g_2(r_2)][J_{k-1}(r_1+1,0,x_1+1,x_2-1)-J_{k-1}(r_1+1,0,x_1,x_2-1)]\\
&-g_1(r_1+1)[J_{k-1}(r_1+2,r_2,x_1+1,x_2-J_{k-1}(r_1+2,r_2,x_1,x_2))]\\
&-[1-g_1(r_1+1)][J_{k-1}(0,r_2,x_1,x_2)-J_{k-1}(0,r_2,x_1-1,x_2)]
\end{aligned}\tag{A.31}$$

and

$$\begin{aligned}
&PX.1_I_3.2\\
=&g_2(r_2)J_{k-1}(0,r_2+1,x_1,x_2)+[1-g_2(r_2)]J_{k-1}(0,0,x_1,x_2-1)\\
&-g_1(0)J_{k-1}(1,r_2,x_1,x_2)-[1-g_1(0)]J_{k-1}(0,r_2,x_1-1,x_2)
\end{aligned}\tag{A.32}$$

According to induction assumption of **PX.1**,

$$PX.1\_I\_3.1\geq 0\tag{A.33}$$

and since we take action 1 at state $D'.2$, thus

$$PX.1\_I\_3.2\geq 0\tag{A.34}$$

Therefore

$$\begin{aligned}
&g_2(r_2)A'+[1-g_2(r_2)]B'+g_1(r_1)C'+[1-g_1(r_1)]D'\\
&=g_1(r_1)PX.1_I_3.1+[1-g_1(r_1)]PX.1_I_3.2\\
&\geq 0
\end{aligned}\tag{A.35}$$

Which completes the proof of **PX.1-Case I-3**.

PX.1-Case I-4: since we take action 2 for state $A'.2$, we have

$$A'.2.v1 \geq A'.2.v2 \tag{A.36}$$

Together with the conclusion of **PX.1-Case I-3**, we have

$$\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v1 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v1) \\
\geq&g_2(r_2)(A'.1.v1 - A'.2.v1) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v1) \\
\geq&0
\end{aligned} \tag{A.37}$$

This completes the proof for **PX.1-Case I-4.**

PX.1-Case I-5:

$$g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D'$$

$$= g_2(r_2)(A'.1.v1 - A'.2.v1) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1)$$

$$+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2)$$

$$= g_2(r_2)g_1(r_1)J_{k-1}(r_1 + 1, r_2 + 1, x_1 + 1, x_2) + g_2(r_2)[1 - g_1(r_1)]J_{k-1}(0, r_2 + 1, x_1, x_2)$$

$$- g_2(r_2)g_1(r_1)J_{k-1}(r_1 + 1, r_2 + 1, x_1, x_2) - g_2(r_2)[1 - g_1(r_1)]J_{k-1}(0, r_2 + 1, x_1 - 1, x_2)$$

$$+ [1 - g_2(r_2)]g_1(r_1)J_{k-1}(r_1 + 1, 0, x_1 + 1, x_2 - 1) + [1 - g_2(r_2)][1 - g_1(r_1)]J_{k-1}(0, 0, x_1, x_2 - 1)$$

$$- [1 - g_2(r_2)]g_1(r_1)J_{k-1}(r_1 + 1, 0, x_1, x_2 - 1) - [1 - g_2(r_2)][1 - g_1(r_1)]J_{k-1}(0, 0, x_1 - 1, x_2 - 1)$$

$$+ g_1(r_1)g_1(r_1 + 1)J_{k-1}(r_1 + 2, r_2, x_1, x_2) + g_1(r_1)[1 - g_1(r_1 + 1)]J_{k-1}(0, r_2, x_1 - 1, x_2)$$

$$- g_1(r_1)g_1(r_1 + 1)J_{k-1}(r_1 + 2, r_2, x_1 + 1, x_2) - g_1(r_1)[1 - g_1(r_1 + 1)]J_{k-1}(0, r_2, x_1, x_2)$$

$$+ [1 - g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2 + 1, x_1 - 1, x_2) + [1 - g_1(r_1)][1 - g_2(r_2)]J_{k-1}(0, 0, x_1 - 1, x_2 - 1)$$

$$- [1 - g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2 + 1, x_1, x_2) - [1 - g_1(r_1)][1 - g_2(r_2)]J_{k-1}(0, 0, x_1, x_2 - 1) \quad \text{(A.38)}$$

Rearrange terms:

$$g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D'$$

$$= g_1(r_1)\{g_2(r_2)[J_{k-1}(r_1 + 1, r_2 + 1, x_1 + 1, x_2) - J_{k-1}(r_1 + 1, r_2 + 1, x_1, x_2)]$$

$$+ [1 - g_2(r_2)][J_{k-1}(r_1 + 1, 0, x_1 + 1, x_2 - 1) - J_{k-1}(r_1 + 1, 0, x_1, x_2 - 1)]$$

$$+ g_1(r_1 + 1)[J_{k-1}(r_1 + 2, r_2, x_1, x_2) - J_{k-1}(r_1 + 2, r_2, x_1 + 1, x_2)]$$

$$+ [1 - g_1(r_1 + 1)][J_{k-1}(0, r_2, x_1 - 1, x_2) - J_{k-1}(0, r_2, x_1, x_2)]\} \quad \text{(A.39)}$$

which is nonnegative according to the induction on **PX.1**. This completes the proof of **PX.1-Case I-5**.

PX.1-Case I-6: since we take action 2 for state $A'.2$, we have

$$A'.2.v1 \geq A'.2.v2 \tag{A.40}$$

Together with the conclusion of **PX.1-Case I-5**, we have

$$\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v1 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&g_2(r_2)(A'.1.v1 - A'.2.v1) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&0
\end{aligned} \tag{A.41}$$

This completes the proof for **PX.1-Case I-6.**

PX.1-Case I-7: since we take action 2 for state $B'.2$, we have

$$B'.2.v1 \geq B'.2.v2 \tag{A.42}$$

Together with the conclusion of **PX.1-Case I-6**, we have

$$\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v1 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v2) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&g_2(r_2)(A'.1.v1 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&0
\end{aligned} \tag{A.43}$$

This completes the proof for **PX.1-Case I-7.**

PX.1-Case I-8:

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=& g_2(r_2)(A'.1.v1 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1) \\
& + g_1(r_1)(C'.1.v2 - C'.2.v1) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2) \\
=& g_2(r_2)g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) + g_2(r_2)[1-g_1(r_1)]J_{k-1}(0, r_2+1, x_1, x_2) \\
& - g_2(r_2)g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) - g_2(r_2)[1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1) \\
& + [1-g_2(r_2)]g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1-g_2(r_2)][1-g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1) \\
& - [1-g_2(r_2)]g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) - [1-g_2(r_2)][1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1) \\
& + g_1(r_1)g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + g_1(r_1)[1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1, x_2-1) \\
& - g_1(r_1)g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1+1, x_2) - g_1(r_1)[1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1, x_2) \\
& + [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1-1, x_2) + [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1-1, x_2-1) \\
& - [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) - [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1)
\end{aligned}
\tag{A.44}
$$

Rearrange terms:

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=& g_2(r_2)\{g_1(r_1)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + [1-g_1(r_1)]J_{k-1}(0, r_2+1, x_1-1, x_2) \\
& - g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) - [1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1)\} \\
& + g_1(r_1)\{g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) + [1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1+1, x_2-1) \\
& - g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1+1, x_2) - [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1, x_2)\}
\end{aligned}
\tag{A.45}
$$

Let

$$PX.1\_I\_8.1 \tag{A.46}$$
$$=g_1(r_1)J_{k-1}(r_1+1,r_2+1,x_1,x_2)+[1-g_1(r_1)]J_{k-1}(0,r_2+1,x_1-1,x_2)$$
$$-g_2(r_2+1)J_{k-1}(r_1,r_2+2,x_1,x_2)-[1-g_2(r_2+1)]J_{k-1}(r_1,0,x_1,x_2-1)$$

and

$$PX.1\_I\_8.2$$
$$=g_2(r_2)J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)+[1-g_2(r_2)]J_{k-1}(r_1+1,0,x_1+1,x_2-1)$$
$$-g_1(r_1+1)J_{k-1}(r_1+2,r_2,x_1+1,x_2)-[1-g_1(r_1+1)]J_{k-1}(0,r_2,x_1,x_2) \tag{A.47}$$

Since we take action 2 at state $A'.2$ and take action 1 at state $C'.2$, thus:

$$PX.1\_I\_8.1 \geq 0 \text{ and } PX.1\_I\_8.2 \geq 0 \tag{A.48}$$

and

$$g_2(r_2)A'+[1-g_2(r_2)]B'+g_1(r_1)C'+[1-g_1(r_1)]D'$$
$$=g_2(r_2)PX.1\_I\_8.1+g_1(r_1)PX.1\_I\_8.2$$
$$\geq 0 \tag{A.49}$$

This completes the proof of **PX.1-Case I-8**.

PX.1-Case I-9:since we take action 2 for state $B'.2$, we have

$$B'.2.v1 \geq B'.2.v2 \tag{A.50}$$

Together with the conclusion of **PX.1-Case I-8**, we have

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=& g_2(r_2)(A'.1.v1 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v2) \\
& + g_1(r_1)(C'.1.v2 - C'.2.v1) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq& g_2(r_2)(A'.1.v1 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1) \\
& + g_1(r_1)(C'.1.v2 - C'.2.v1) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq& 0
\end{aligned}
\tag{A.51}
$$

This completes the proof for **PX.1-Case I-9.**

<u>PX.1-Case I-10:</u>

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=& g_2(r_2)(A'.1.v2 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1) \\
& + g_1(r_1)(C'.1.v1 - C'.2.v1) + [1-g_1(r_1)](D'.1.v1 - D'.2.v1) \\
=& g_2(r_2)[g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1+1, x_2) + [1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1+1, x_2-1)] \\
& - g_2(r_2)[g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) + [1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1)] \\
& + [1-g_2(r_2)][g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1)] \\
& - [1-g_2(r_2)][g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1)] \\
& + g_1(r_1)[g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1-1, x_2)] \\
& - g_1(r_1)[g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1+1, x_2) + [1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1, x_2)] \\
& + [1-g_1(r_1)][g_1(0)J_{k-1}(1, r_2, x_1-1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-2, x_2)] \\
& - [1-g_1(r_1)][g_1(0)J_{k-1}(1, r_2, x_1, x_2) + [1-g_1(0)]J_{k-1}(0, r_2, x_1-1, x_2)]
\end{aligned}
\tag{A.52}
$$

Rearrange the terms we have:

$$
\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)] \\
&+ [1 - g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)]\} \\
&+ g_1(r_1)\{[1 - g_2(r_2)][J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)] \\
&+ g_1(r_1+1)[J_{k-1}(r_1+2, r_2, x_1, x_2) - J_{k-1}(r_1+2, r_2, x_1+1, x_2)] \\
&+ [1 - g_1(r_1+1)][J_{k-1}(0, r_2, x_1-1, x_2) - J_{k-1}(0, r_2, x_1, x_2)]\} \\
&+ [1 - g_1(r_1)]\{[1 - g_2(r_2)][J_{k-1}(0, 0, x_1, x_2-1) - J_{k-1}(0, 0, x_1-1, x_2-1)] \\
&+ g_1(0)[J_{k-1}(1, r_2, x_1-1, x_2) - J_{k-1}(1, r_2, x_1, x_2)] \\
&+ [1 - g_1(0)][J_{k-1}(0, r_2, x_1-2, x_2) - J_{k-1}(0, r_2, x_1-1, x_2)]\} \qquad \text{(A.53)}
\end{aligned}
$$

we add

$$
g_1(r_1)g_2(r_2)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)] \qquad \text{(A.54)}
$$

to the second term of (A.53), and add

$$
[1 - g_1(r_1)]g_2(r_2)[J_{k-1}(0, r_2+1, x_1, x_2) - J_{k-1}(0, r_2+1, x_1-1, x_2)] \qquad \text{(A.55)}
$$

to the third term of (A.53), and subtract the above two terms from the first term of (A.53), we have

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=& g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1,r_2+2,x_1+1,x_2) - J_{k-1}(r_1,r_2+2,x_1,x_2)] \\
& + [1-g_2(r_2+1)][J_{k-1}(r_1,0,x_1+1,x_2-1) - J_{k-1}(r_1,0,x_1,x_2-1)] \\
& + g_1(r_1)[J_{k-1}(r_1+1,r_2+1,x_1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)] \\
& + [1-g_1(r_1)][J_{k-1}(0,r_2+1,x_1-1,x_2) - J_{k-1}(0,r_2+1,x_1,x_2)]\} \\
& + g_1(r_1)\{g_2(r_2)[J_{k-1}(r_1+1,r_2+1,x_1+1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1,x_2)] \\
& + [1-g_2(r_2)][J_{k-1}(r_1+1,0,x_1+1,x_2-1) - J_{k-1}(r_1+1,0,x_1,x_2-1)] \\
& + g_1(r_1+1)[J_{k-1}(r_1+2,r_2,x_1,x_2) - J_{k-1}(r_1+2,r_2,x_1+1,x_2)] \\
& + [1-g_1(r_1+1)][J_{k-1}(0,r_2,x_1-1,x_2) - J_{k-1}(0,r_2,x_1,x_2)]\} \\
& + [1-g_1(r_1)]\{g_2(r_2)[J_{k-1}(0,r_2+1,x_1,x_2) - J_{k-1}(0,r_2+1,x_1-1,x_2)] \\
& + [1-g_2(r_2)][J_{k-1}(0,0,x_1,x_2-1) - J_{k-1}(0,0,x_1-1,x_2-1)] \\
& + g_1(0)[J_{k-1}(1,r_2,x_1-1,x_2) - J_{k-1}(1,r_2,x_1,x_2)] \\
& + [1-g_1(0)][J_{k-1}(0,r_2,x_1-2,x_2) - J_{k-1}(0,r_2,x_1-1,x_2)]\}
\end{aligned} \tag{A.56}
$$

Let

$$
\begin{aligned}
& PX.1\_I\_10.1 \\
=& g_2(r_2+1)[J_{k-1}(r_1,r_2+2,x_1+1,x_2) - J_{k-1}(r_1,r_2+2,x_1,x_2)] \\
& + [1-g_2(r_2+1)][J_{k-1}(r_1,0,x_1+1,x_2-1) - J_{k-1}(r_1,0,x_1,x_2-1)] \\
& + g_1(r_1)[J_{k-1}(r_1+1,r_2+1,x_1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)] \\
& + [1-g_1(r_1)][J_{k-1}(0,r_2+1,x_1-1,x_2) - J_{k-1}(0,r_2+1,x_1,x_2)]
\end{aligned} \tag{A.57}
$$

and

$$
\begin{aligned}
&PX.1\_I\_10.2 \\
&= g_2(r_2)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)] \\
&+ [1-g_2(r_2)][J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)] \\
&+ g_1(r_1+1)[J_{k-1}(r_1+2, r_2, x_1, x_2) - J_{k-1}(r_1+2, r_2, x_1+1, x_2)] \\
&+ [1-g_1(r_1+1)][J_{k-1}(0, r_2, x_1-1, x_2) - J_{k-1}(0, r_2, x_1, x_2)] \qquad \text{(A.58)}
\end{aligned}
$$

furthermore:

$$
\begin{aligned}
&PX.1\_I\_10.3 \\
&= g_2(r_2)[J_{k-1}(0, r_2+1, x_1, x_2) - J_{k-1}(0, r_2+1, x_1-1, x_2)] \\
&+ [1-g_2(r_2)][J_{k-1}(0, 0, x_1, x_2-1) - J_{k-1}(0, 0, x_1-1, x_2-1)] \\
&+ g_1(0)[J_{k-1}(1, r_2, x_1-1, x_2) - J_{k-1}(1, r_2, x_1, x_2)] \\
&+ [1-g_1(0)][J_{k-1}(0, r_2, x_1-2, x_2) - J_{k-1}(0, r_2, x_1-1, x_2)] \qquad \text{(A.59)}
\end{aligned}
$$

According to the induction assumption of **PX.1,** we have:

$$
PX.1\_I\_11.1 \geq 0 \quad \text{and} \quad PX.1\_I\_10.2 \geq 0 \;\; \text{and} \;\; PX.1\_I\_10.3 \geq 0 \qquad \text{(A.60)}
$$

Thus

$$
\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=&g_2(r_2)PX.1\_I\_10.1 + g_1(r_1)PX.1\_I\_10.2 + [1-g_1(r_1)]PX.1\_I\_10.3 \geq 0 \qquad \text{(A.61)}
\end{aligned}
$$

This completes the proof for **PX.1-Case I-10.**

<u>PX.1-Case I-11:</u>

$$\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v2 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2) \\
=&g_2(r_2)[g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1+1, x_2) + [1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1+1, x_2-1)] \\
&- g_2(r_2)[g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) + [1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1)] \\
&+ [1-g_2(r_2)]g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1-g_2(r_2)][1-g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1) \\
&- [1-g_2(r_2)]g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) - [1-g_2(r_2)][1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1) \\
&+ g_1(r_1)g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1, x_2) + g_1(r_1)[1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1-1, x_2) \\
&- g_1(r_1)g_1(r_1+1)J_{k-1}(r_1+2, r_2, x_1+1, x_2) - g_1(r_1)[1-g_1(r_1+1)]J_{k-1}(0, r_2, x_1, x_2) \\
&+ [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1-1, x_2) + [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1-1, x_2-1) \\
&- [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) - [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1)
\end{aligned} \tag{A.62}$$

Cancel same terms out, we have

$$\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=&g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)] \\
&+ [1-g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)] \\
&+ [1-g_1(r_1)][J_{k-1}(0, r_2+1, x_1-1, x_2) - J_{k-1}(0, r_2+1, x_1, x_2)]\} \\
&+ g_1(r_1)\{[1-g_2(r_2)][J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)] \\
&+ g_1(r_1+1)[J_{k-1}(r_1+2, r_2, x_1, x_2) - J_{k-1}(r_1+2, r_2, x_1+1, x_2)] \\
&+ [1-g_1(r_1+1)][J_{k-1}(0, r_2, x_1-1, x_2) - J_{k-1}(0, r_2, x_1, x_2)]\}
\end{aligned} \tag{A.63}$$

We add term

$$g_2(r_2)g_1(r_1)[J_{k-1}(r_1+1,r_2+1,x_1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)] \tag{A.64}$$

to the first term in (A.63) and subtract the same term from the second term in (A.63), we have

$$\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=&g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1,r_2+2,x_1+1,x_2) - J_{k-1}(r_1,r_2+2,x_1,x_2)] \\
&+ [1-g_2(r_2+1)][J_{k-1}(r_1,0,x_1+1,x_2-1) - J_{k-1}(r_1,0,x_1,x_2-1)] \\
&+ g_1(r_1)[J_{k-1}(r_1+1,r_2+1,x_1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)] \\
&+ [1-g_1(r_1)][J_{k-1}(0,r_2+1,x_1-1,x_2) - J_{k-1}(0,r_2+1,x_1,x_2)]\} \\
&+ g_1(r_1)\{g_2(r_2)[J_{k-1}(r_1+1,r_2+1,x_1+1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1,x_2)] \\
&+ [1-g_2(r_2)][J_{k-1}(r_1+1,0,x_1+1,x_2-1) - J_{k-1}(r_1+1,0,x_1,x_2-1)] \\
&+ g_1(r_1+1)[J_{k-1}(r_1+2,r_2,x_1,x_2) - J_{k-1}(r_1+2,r_2,x_1+1,x_2)] \\
&+ [1-g_1(r_1+1)][J_{k-1}(0,r_2,x_1-1,x_2) - J_{k-1}(0,r_2,x_1,x_2)]\}
\end{aligned} \tag{A.65}$$

Let

$$\begin{aligned}
&PX.1_I_11.1 \\
=&g_2(r_2+1)[J_{k-1}(r_1,r_2+2,x_1+1,x_2) - J_{k-1}(r_1,r_2+2,x_1,x_2)] \\
&+ [1-g_2(r_2+1)][J_{k-1}(r_1,0,x_1+1,x_2-1) - J_{k-1}(r_1,0,x_1,x_2-1)] \\
&+ g_1(r_1)[J_{k-1}(r_1+1,r_2+1,x_1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)] \\
&+ [1-g_1(r_1)][J_{k-1}(0,r_2+1,x_1-1,x_2) - J_{k-1}(0,r_2+1,x_1,x_2)]
\end{aligned} \tag{A.66}$$

and

$$\begin{aligned}
&PX.1_I_11.2\\
=&g_2(r_2)[J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)-J_{k-1}(r_1+1,r_2+1,x_1,x_2)]\\
&+[1-g_2(r_2)][J_{k-1}(r_1+1,0,x_1+1,x_2-1)-J_{k-1}(r_1+1,0,x_1,x_2-1)]\\
&+g_1(r_1+1)[J_{k-1}(r_1+2,r_2,x_1,x_2)-J_{k-1}(r_1+2,r_2,x_1+1,x_2)]\\
&+[1-g_1(r_1+1)][J_{k-1}(0,r_2,x_1-1,x_2)-J_{k-1}(0,r_2,x_1,x_2)] \qquad \text{(A.67)}
\end{aligned}$$

According to the induction assumption of **PX.1,** we have:

$$PX.1\_I\_11.1 \geq 0 \quad \text{and} \quad PX.1\_I\_11.2 \geq 0 \qquad \text{(A.68)}$$

Thus

$$\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D'\\
=&g_2(r_2)PX.1_I_11.1 + g_1(r_1)PX.1_I_11.2 \geq 0 \qquad \text{(A.69)}
\end{aligned}$$

This completes the proof for **PX.1-Case I-11.**

PX.1-Case I-12:since we take action 1 for state $D'.2$, we have

$$D'.2.v1 \leq D'.2.v2 \qquad \text{(A.70)}$$

Together with the conclusion of **PX.1-Case I-11**, we have

$$\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v1) \\
\geq&g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&0
\end{aligned} \tag{A.71}$$

This completes the proof for **PX.1-Case I-12.**

PX.1-Case I-13:since we take action 2 for state $B'.2$, we have

$$B'.2.v1 \geq B'.2.v2 \tag{A.72}$$

Together with the conclusion of **PX.1-Case I-11**, we have

$$\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v2) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&0
\end{aligned} \tag{A.73}$$

This completes the proof for **PX.1-Case I-13.**

PX.1-Case I-14:

$$g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D'$$

$$= g_2(r_2)(A'.1.v2 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v2)$$

$$+ g_1(r_1)(C'.1.v2 - C'.2.v2) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2)$$

$$= g_2(r_2)g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1+1, x_2) + g_2(r_2)[1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1+1, x_2-1)$$

$$- g_2(r_2)g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) - g_2(r_2)[1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1)$$

$$+ [1-g_2(r_2)]g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1-g_2(r_2)][1-g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1)$$

$$- [1-g_2(r_2)]g_2(0)J_{k-1}(r_1, 1, x_1, x_2-1) - [1-g_2(r_2)][1-g_2(0)]J_{k-1}(r_1, 0, x_1, x_2-2)$$

$$+ g_1(r_1)g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + g_1(r_1)[1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1, x_2-1)$$

$$- g_1(r_1)g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - g_1(r_1)[1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1+1, x_2-1)$$

$$+ [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1-1, x_2) + [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1-1, x_2-1)$$

$$- [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) - [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1) \quad (A.74)$$

Cancel out same terms and arrange terms, we have:

$$g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D'$$

$$= g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)]$$

$$+ [1-g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)]$$

$$+ g_1(r_1)[J_{k-1}(r_1+1, r_2+1, x_1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1+1, x_2)]$$

$$+ [1-g_1(r_1)][J_{k-1}(0, r_2+1, x_1-1, x_2) - J_{k-1}(0, r_2+1, x_1, x_2)]\}$$

$$+ [1-g_2(r_2)]\{g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1)$$

$$- g_2(0)J_{k-1}(r_1, 1, x_1, x_2-1) - [1-g_2(0)]J_{k-1}(r_1, 0, x_1, x_2-2)\} \quad (A.75)$$

Let

$$
\begin{aligned}
& PX.1\_I\_14.1 \\
& = g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)] \\
& + [1-g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)] \\
& + g_1(r_1)[J_{k-1}(r_1+1, r_2+1, x_1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1+1, x_2)] \\
& + [1-g_1(r_1)][J_{k-1}(0, r_2+1, x_1-1, x_2) - J_{k-1}(0, r_2+1, x_1, x_2)]
\end{aligned} \tag{A.76}
$$

and

$$
\begin{aligned}
& PX.1\_I\_14.2 \\
& = g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) + [1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1) \\
& - g_2(0)J_{k-1}(r_1, 1, x_1, x_2-1) - [1-g_2(0)]J_{k-1}(r_1, 0, x_1, x_2-2)
\end{aligned} \tag{A.77}
$$

According to the induction assumption of **PX.1**,

$$PX.1\_I\_14.1 \geq 0 \tag{A.78}$$

and according to the assumption that we take action 2 at state $B'.2$,

$$PX.1\_I\_14.2 \geq 0 \tag{A.79}$$

Thus

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
= & g_2(r_2)PX.1\_I\_14.1 + [1-g_2(r_2)]PX.1\_I\_14.2 \geq 0
\end{aligned} \tag{A.80}
$$

This completes the proof for **PX.1-Case I-14.**

PX.1-Case I-15: since we take action 1 for state $C'.2$, we have

$$C'.2.v1 \leq C'.2.v2 \tag{A.81}$$

Together with the conclusion of **PX.1-Case I-14**, we have

$$\begin{aligned}
& g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=& g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v2) \\
& + g_1(r_1)(C'.1.v2 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq & g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v2) \\
& + g_1(r_1)(C'.1.v2 - C'.2.v2) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq & 0
\end{aligned} \tag{A.82}$$

This completes the proof for **PX.1-Case I-15.**

PX.1-Case I-16:

$$g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D'$$
$$= g_2(r_2)(A'.1.v2 - A'.2.v2) + [1-g_2(r_2)](B'.1.v1 - B'.2.v1)$$
$$+ g_1(r_1)(C'.1.v2 - C'.2.v2) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2)$$
$$= g_2(r_2)g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1+1, x_2) + g_2(r_2)[1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1+1, x_2-1)$$
$$- g_2(r_2)g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) - g_2(r_2)[1-g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1)$$
$$+ [1-g_2(r_2)]g_1(r_1)J_{k-1}(r_1+1, 0, x_1+1, x_2-1) + [1-g_2(r_2)][1-g_1(r_1)]J_{k-1}(0, 0, x_1, x_2-1)$$
$$- [1-g_2(r_2)]g_1(r_1)J_{k-1}(r_1+1, 0, x_1, x_2-1) - [1-g_2(r_2)][1-g_1(r_1)]J_{k-1}(0, 0, x_1-1, x_2-1)$$
$$+ g_1(r_1)g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + g_1(r_1)[1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1, x_2-1)$$
$$- g_1(r_1)g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - g_1(r_1)[1-g_2(r_2)]J_{k-1}(r_1+1, 0, x_1+1, x_2-1)$$
$$+ [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1-1, x_2) + [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1-1, x_2-1)$$
$$- [1-g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) - [1-g_1(r_1)][1-g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1) \quad \text{(A.83)}$$

Cancel out terms and rearrange terms:

$$g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D'$$
$$= g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)]$$
$$+ [1-g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)]$$
$$+ g_1(r_1)[J_{k-1}(r_1+1, r_2+1, x_1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1+1, x_2)]$$
$$+ [1-g_1(r_1)][J_{k-1}(0, r_2+1, x_1-1, x_2) - J_{k-1}(0, r_2+1, x_1, x_2)]\} \quad \text{(A.84)}$$

Which is nonnegative according to the induction assumption on **PX.1**. This completes the proof for **PX.1-Case I-16**.

PX.1-Case I-17: since we take action 1 for state $C'.2$, we have

$$C'.2.v1 \leq C'.2.v2 \tag{A.85}$$

Together with the conclusion of **PX.1-Case I-16**, we have

$$\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v2 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v1 - B'.2.v1) \\
&+ g_1(r_1)(C'.1.v2 - C'.2.v2) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq&0
\end{aligned} \tag{A.86}$$

This completes the proof for **PX.1-Case I-17.**

PX.1-Case I-18:

$$g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D'$$

$$= g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v2 - B'.2.v2)$$

$$+ g_1(r_1)(C'.1.v1 - C'.2.v1) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2)$$

$$= g_2(r_2)g_2(r_2 + 1)J_{k-1}(r_1, r_2 + 2, x_1 + 1, x_2) + g_2(r_2)[1 - g_2(r_2 + 1)]J_{k-1}(r_1, 0, x_1 + 1, x_2 - 1)$$

$$- g_2(r_2)g_2(r_2 + 1)J_{k-1}(r_1, r_2 + 2, x_1, x_2) - g_2(r_2)[1 - g_2(r_2 + 1)]J_{k-1}(r_1, 0, x_1, x_2 - 1)$$

$$+ [1 - g_2(r_2)]g_2(0)J_{k-1}(r_1, 1, x_1 + 1, x_2 - 1) + [1 - g_2(r_2)][1 - g_2(0)]J_{k-1}(r_1, 0, x_1 + 1, x_2 - 2)$$

$$- [1 - g_2(r_2)]g_2(0)J_{k-1}(r_1, 1, x_1, x_2 - 1) - [1 - g_2(r_2)][1 - g_2(0)]J_{k-1}(r_1, 0, x_1, x_2 - 2)$$

$$+ g_1(r_1)g_1(r_1 + 1)J_{k-1}(r_1 + 2, r_2, x_1, x_2) + g_1(r_1)[1 - g_1(r_1 + 1)]J_{k-1}(0, r_2, x_1 - 1, x_2)$$

$$- g_1(r_1)g_1(r_1 + 1)J_{k-1}(r_1 + 2, r_2, x_1 + 1, x_2) - g_1(r_1)[1 - g_1(r_1 + 1)]J_{k-1}(0, r_2, x_1, x_2)$$

$$+ [1 - g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2 + 1, x_1 - 1, x_2) + [1 - g_1(r_1)][1 - g_2(r_2)]J_{k-1}(0, 0, x_1 - 1, x_2 - 1)$$

$$- [1 - g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2 + 1, x_1, x_2) - [1 - g_1(r_1)][1 - g_2(r_2)]J_{k-1}(0, 0, x_1, x_2 - 1) \quad \text{(A.87)}$$

Rearrange terms, we have:

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
& = g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1,r_2+2,x_1+1,x_2) - J_{k-1}(r_1,r_2+2,x_1,x_2)] \\
& + [1-g_2(r_2+1)][J_{k-1}(r_1,0,x_1+1,x_2-1) - J_{k-1}(r_1,0,x_1,x_2-1)] \\
& + [1-g_1(r_1)][J_{k-1}(0,r_2+1,x_1-1,x_2) - J_{k-1}(0,r_2+1,x_1,x_2)]\} \\
& + [1-g_2(r_2)]\{g_2(0)[J_{k-1}(r_1,1,x_1+1,x_2-1) - J_{k-1}(r_1,1,x_1,x_2-1)] \\
& + [1-g_2(0)][J_{k-1}(r_1,0,x_1+1,x_2-2) - J_{k-1}(r_1,0,x_1,x_2-2)] \\
& + [1-g_1(r_1)][J_{k-1}(0,0,x_1-1,x_2-1) - J_{k-1}(0,0,x_1,x_2-1)]\} \\
& + g_1(r_1)\{g_1(r_1+1)[J_{k-1}(r_1+2,r_2,x_1,x_2) - J_{k-1}(r_1+2,r_2,x_1+1,x_2)] \\
& + [1-g_1(r_1+1)][J_{k-1}(0,r_2,x_1-1,x_2) - J_{k-1}(0,r_2,x_1,x_2)]\}
\end{aligned} \tag{A.88}
$$

we add

$$g_1(r_1)g_2(r_2)[J_{k-1}(r_1+1,r_2+1,x_1,x_2) - J_{k-1}(r_1+1,r_2+1,x_1+1,x_2)] \tag{A.89}$$

to the first term of (A.88), and add

$$[1-g_2(r_2)]g_1(r_1)[J_{k-1}(r_1+1,0,x_1,x_2-1) - J_{k-1}(r_1+1,0,x_1+1,x_2-1)] \tag{A.90}$$

to the second term of (A.88), and subtract the above two terms from the third term of (A.88), we have

$$
\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
&= g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)] \\
&+ [1-g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)] \\
&+ g_1(r_1)[J_{k-1}(r_1+1, r_2+1, x_1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1+1, x_2)] \\
&+ [1-g_1(r_1)][J_{k-1}(0, r_2+1, x_1-1, x_2) - J_{k-1}(0, r_2+1, x_1, x_2)]\} \\
&+ [1-g_2(r_2)]\{g_2(0)[J_{k-1}(r_1, 1, x_1+1, x_2-1) - J_{k-1}(r_1, 1, x_1, x_2-1)] \\
&+ [1-g_2(0)][J_{k-1}(r_1, 0, x_1+1, x_2-2) - J_{k-1}(r_1, 0, x_1, x_2-2)] \\
&+ g_1(r_1)[J_{k-1}(r_1+1, 0, x_1, x_2-1) - J_{k-1}(r_1+1, 0, x_1+1, x_2-1)] \\
&+ [1-g_1(r_1)][J_{k-1}(0, 0, x_1-1, x_2-1) - J_{k-1}(0, 0, x_1, x_2-1)]\} \\
&+ g_1(r_1)\{g_1(r_1+1)[J_{k-1}(r_1+2, r_2, x_1, x_2) - J_{k-1}(r_1+2, r_2, x_1+1, x_2)] \\
&+ [1-g_1(r_1+1)][J_{k-1}(0, r_2, x_1-1, x_2) - J_{k-1}(0, r_2, x_1, x_2)] \\
&+ g_2(r_2)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)] \\
&+ [1-g_2(r_2)][J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)]\} \qquad \text{(A.91)}
\end{aligned}
$$

Let

$$
\begin{aligned}
&PX.1\_I\_18.1 \\
&= g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)] \\
&+ [1-g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)] \\
&+ g_1(r_1)[J_{k-1}(r_1+1, r_2+1, x_1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1+1, x_2)] \\
&+ [1-g_1(r_1)][J_{k-1}(0, r_2+1, x_1-1, x_2) - J_{k-1}(0, r_2+1, x_1, x_2)] \qquad \text{(A.92)}
\end{aligned}
$$

and

$$
\begin{aligned}
&PX.1\_I\_18.2 \\
&= g_2(0)[J_{k-1}(r_1, 1, x_1 + 1, x_2 - 1) - J_{k-1}(r_1, 1, x_1, x_2 - 1)] \\
&+ [1 - g_2(0)][J_{k-1}(r_1, 0, x_1 + 1, x_2 - 2) - J_{k-1}(r_1, 0, x_1, x_2 - 2)] \\
&+ g_1(r_1)[J_{k-1}(r_1 + 1, 0, x_1, x_2 - 1) - J_{k-1}(r_1 + 1, 0, x_1 + 1, x_2 - 1)] \\
&+ [1 - g_1(r_1)][J_{k-1}(0, 0, x_1 - 1, x_2 - 1) - J_{k-1}(0, 0, x_1, x_2 - 1)]
\end{aligned}
\tag{A.93}
$$

furthermore:

$$
\begin{aligned}
&PX.1\_I\_18.3 \\
&= g_1(r_1 + 1)[J_{k-1}(r_1 + 2, r_2, x_1, x_2) - J_{k-1}(r_1 + 2, r_2, x_1 + 1, x_2)] \\
&+ [1 - g_1(r_1 + 1)][J_{k-1}(0, r_2, x_1 - 1, x_2) - J_{k-1}(0, r_2, x_1, x_2)] \\
&+ g_2(r_2)[J_{k-1}(r_1 + 1, r_2 + 1, x_1 + 1, x_2) - J_{k-1}(r_1 + 1, r_2 + 1, x_1, x_2)] \\
&+ [1 - g_2(r_2)][J_{k-1}(r_1 + 1, 0, x_1 + 1, x_2 - 1) - J_{k-1}(r_1 + 1, 0, x_1, x_2 - 1)]
\end{aligned}
\tag{A.94}
$$

According to the induction assumption of **PX.1,** we have:

$$
PX.1\_I\_18.1 \geq 0 \quad \text{and} \quad PX.1\_I\_18.2 \geq 0 \;\; \text{and} \;\; PX.1\_I\_18.3 \geq 0 \tag{A.95}
$$

Thus

$$
\begin{aligned}
&g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D' \\
=&g_2(r_2)PX.1\_I\_18.1 + [1 - g_2(r_2)]PX.1\_I\_18.2 + g_1(r_1)PX.1\_I\_18.3 \geq 0
\end{aligned}
\tag{A.96}
$$

This completes the proof for **PX.1-Case I-18.**

PX.1-Case I-19:

$$g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D'$$

$$= g_2(r_2)(A'.1.v2 - A'.2.v2) + [1 - g_2(r_2)](B'.1.v2 - B'.2.v2)$$

$$+ g_1(r_1)(C'.1.v2 - C'.2.v2) + [1 - g_1(r_1)](D'.1.v2 - D'.2.v2)$$

$$= g_2(r_2)g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1+1, x_2) + g_2(r_2)[1 - g_2(r_2+1)]J_{k-1}(r_1, 0, x_1+1, x_2-1)$$

$$- g_2(r_2)g_2(r_2+1)J_{k-1}(r_1, r_2+2, x_1, x_2) - g_2(r_2)[1 - g_2(r_2+1)]J_{k-1}(r_1, 0, x_1, x_2-1)$$

$$+ [1 - g_2(r_2)]g_2(0)J_{k-1}(r_1, 1, x_1+1, x_2-1) + [1 - g_2(r_2)][1 - g_2(0)]J_{k-1}(r_1, 0, x_1+1, x_2-2)$$

$$- [1 - g_2(r_2)]g_2(0)J_{k-1}(r_1, 1, x_1, x_2-1) - [1 - g_2(r_2)][1 - g_2(0)]J_{k-1}(r_1, 0, x_1, x_2-2)$$

$$+ g_1(r_1)g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1, x_2) + g_1(r_1)[1 - g_2(r_2)]J_{k-1}(r_1+1, 0, x_1, x_2-1)$$

$$- g_1(r_1)g_2(r_2)J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - g_1(r_1)[1 - g_2(r_2)]J_{k-1}(r_1+1, 0, x_1+1, x_2-1)$$

$$+ [1 - g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1-1, x_2) + [1 - g_1(r_1)][1 - g_2(r_2)]J_{k-1}(0, 0, x_1-1, x_2-1)$$

$$- [1 - g_1(r_1)]g_2(r_2)J_{k-1}(0, r_2+1, x_1, x_2) - [1 - g_1(r_1)][1 - g_2(r_2)]J_{k-1}(0, 0, x_1, x_2-1) \quad \text{(A.97)}$$

Rearrange terms, we have

$$g_2(r_2)A' + [1 - g_2(r_2)]B' + g_1(r_1)C' + [1 - g_1(r_1)]D'$$

$$= g_2(r_2)\{g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)]$$

$$+ [1 - g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)]$$

$$- g_1(r_1)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)]$$

$$- [1 - g_1(r_1)][J_{k-1}(0, r_2+1, x_1, x_2) - J_{k-1}(0, r_2+1, x_1-1, x_2)]\}$$

$$+ [1 - g_2(r_2)]\{g_2(0)[J_{k-1}(r_1, 1, x_1+1, x_2-1) - J_{k-1}(r_1, 1, x_1, x_2-1)]$$

$$+ [1 - g_2(0)][J_{k-1}(r_1, 0, x_1+1, x_2-2) - J_{k-1}(r_1, 0, x_1, x_2-2)]$$

$$- g_1(r_1)[J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)]$$

$$- [1 - g_1(r_1)][J_{k-1}(0, 0, x_1, x_2-1) - J_{k-1}(0, 0, x_1-1, x_2-1)]\} \tag{A.98}$$

Let

$$PX.1\_I.19.1$$

$$= g_2(r_2+1)[J_{k-1}(r_1, r_2+2, x_1+1, x_2) - J_{k-1}(r_1, r_2+2, x_1, x_2)]$$

$$+ [1 - g_2(r_2+1)][J_{k-1}(r_1, 0, x_1+1, x_2-1) - J_{k-1}(r_1, 0, x_1, x_2-1)]$$

$$- g_1(r_1)[J_{k-1}(r_1+1, r_2+1, x_1+1, x_2) - J_{k-1}(r_1+1, r_2+1, x_1, x_2)]$$

$$- [1 - g_1(r_1)][J_{k-1}(0, r_2+1, x_1, x_2) - J_{k-1}(0, r_2+1, x_1-1, x_2)] \tag{A.99}$$

and

$$
\begin{aligned}
&PX.1\_I.19.2 \\
&= g_2(0)[J_{k-1}(r_1, 1, x_1+1, x_2-1) - J_{k-1}(r_1, 1, x_1, x_2-1)] \\
&+ [1-g_2(0)][J_{k-1}(r_1, 0, x_1+1, x_2-2) - J_{k-1}(r_1, 0, x_1, x_2-2)] \\
&- g_1(r_1)[J_{k-1}(r_1+1, 0, x_1+1, x_2-1) - J_{k-1}(r_1+1, 0, x_1, x_2-1)] \\
&- [1-g_1(r_1)][J_{k-1}(0, 0, x_1, x_2-1) - J_{k-1}(0, 0, x_1-1, x_2-1)] \qquad \text{(A.100)}
\end{aligned}
$$

According to the induction assumption of **PX.1,** we have:

$$PX.1\_I\_19.1 \geq 0 \quad \text{and} \quad PX.1\_I\_19.2 \geq 0 \qquad \text{(A.101)}$$

Thus

$$
\begin{aligned}
&g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
&= g_2(r_2)PX.1\_I\_19.1 + [1-g_2(r_2)]PX.1\_I\_19.2 \geq 0 \qquad \text{(A.102)}
\end{aligned}
$$

This completes the proof for **PX.1-Case I-19.**

PX.1-Case I-20:since we take action 1 for state $C'.2$, we have

$$C'.2.v1 \leq C'.2.v2 \qquad \text{(A.103)}$$

Together with the conclusion of **PX.1-Case I-19**, we have

$$
\begin{aligned}
& g_2(r_2)A' + [1-g_2(r_2)]B' + g_1(r_1)C' + [1-g_1(r_1)]D' \\
=& g_2(r_2)(A'.1.v2 - A'.2.v2) + [1-g_2(r_2)](B'.1.v2 - B'.2.v2) \\
& + g_1(r_1)(C'.1.v2 - C'.2.v1) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq & g_2(r_2)(A'.1.v2 - A'.2.v2) + [1-g_2(r_2)](B'.1.v2 - B'.2.v2) \\
& + g_1(r_1)(C'.1.v2 - C'.2.v2) + [1-g_1(r_1)](D'.1.v2 - D'.2.v2) \\
\geq & 0
\end{aligned} \tag{A.104}
$$

This completes the proof for **PX.1-Case I-20.**

This completes the proof for **PX.1-Case I**.

PX.1-Case II: Now we consider the cases when $x_1 = 1$ and $x_2 > 1$. Use the same notation as in **PX.1-Case I**, except that we can only take action 2 at state $D'.1 = (0, r_2, x_1 - 1, x_2)$. Thus we need not consider the case $1, 2$ and 10. The proof for all the other cases are exact the same as **PX.1-Case I**. This completes the proof of **PX.1-Case II**.

PX.1-Case III: Now we consider the cases when $x_1 \geq 1$ and $x_2 = 1$. Use the same notation as in **PX.1-Case I**, except that we can only take action 1 at state $B'.1 = (r_1, 0, x_1 + 1, x_2 - 1)$ and state $B'.2 = (r_1, 0, x_1, x_2 - 1)$. Thus we need not consider the case $7, 9, 13, 14, 15, 18, 19$ and 20. The proof for all the other cases are exact the same as **PX.1-Case I**. This completes the proof of **PX.1-Case III**.

PX.1-Case IV: Now we consider the cases when $x_1 = 1$ and $x_2 = 1$. Use the same notation as in **PX.1-Case I**, except that we can only take action 1 at state $D'.1 = (0, r_2, x_1 - 1, x_2)$, state $B'.1 = (r_1, 0, x_1 + 1, x_2 - 1)$ and state $B'.2 = (r_1, 0, x_1, x_2 - 1)$. Thus we need not consider the case $1, 2, 7, 9, 10, 13, 14, 15, 18, 19$ and 20. The proof for all the other cases are exact the same as **PX.1-Case I**. This completes the proof of **PX.1-Case IV**.

This completes the proof of **PX.1**.

Proof for **PX.2**: since PX.2 is completely symmetric of **PX.1**, we can use exact the same argument for its proof. This completes the proof of **PX.2**.

This completes the proof of the theorem. ■

Bibliography

[1] D.P. Bertsekas and R.G. Gallager, Data networks. Prentice-Hall, 1987

[2] P. Liu, R. Berry and M. Honig, Delay-sensitive packet scheduling in wireless networks, IEEE WCNC 2003, New Orleans, LA, March 16-20, 2003

[3] E. Altman and H. Kushner, Control of polling in presence of vacations in heavy traffic with applications to satellite and mobile radio systems, Procedings of the 37rd Allerton Conference on Communication, Control, and Computing, Illinois, USA, Sept. 1999

[4] J. Niño-Mora, Stochastic scheduling, Updated version of article in Encyclopedia of Optimization, C.A. Floudas and P.M. Pardalos, eds., vol. V, 367-372,2001. Kluwer, Dordrecht

[5] K.C. Sevcik, Scheduling for minimum total loss using service time distributions, Journal of the Association for Computing Machinery, Vol. 21, No. 1, January 1974, pp. 66-75

[6] W.E. Smith, Various optimizers for singel-stage production, Naval Research Logistics Quarterly, 3:59-66,1956

[7] M. H. Rothkopf, Scheduling with random service times, Management Science, Vol. 12, No. 9, May, 1966

[8] M. Rothkopf, Scheduling independent tasks on parallel processors, Management Science, Vol. 12, 437-447, 1996

[9] J.N. Tsitsiklis, A short proof for the Gittins index theorem, the Annals of Applied Probability, Vol. 4, 1:194-199, 1994

[10] D. Bertsimas and J. Niño-Mora, Restless bandits, linear programming relaxations and a primal-dual index heuristic, Vol. 48, 1:80-90, 2000

[11] J.C. Gittins and D. M. Jones, A dynamic allocation index for the sequential design of experiments, in Progress in Statistics (European Meeting of Statisticians, Budapest, 1972), 241-266, Colloq. Math. Soc. János Bolyai, Bol. 9. North-Holland, Amsterdam, 1974

[12] D.R. Cox and W.L. Smith, Queues, Methuen (London) and Wiley (New York), 1961

[13] C. Buyukkoc, P. Varaiya, and J. Walrand. The $c\mu$ rule revisited. Adv. Appl. Prob., 17:237-238, 1985

[14] P. Nain, Interchange arguments for classical scheduling problems in queues. Systems & Control Letters, 12:177-184, 1989

[15] Y.D. Serres, Simultaneous optimization of flow control and scheduling in a single server queue with two job classes, Operations Research Letters, 10:103-112, 1991

[16] G.P. Klimov, Time-sharing service systems I, Theory of Probability and its Applications, 19(3): 558-576, 1974

[17] D.W. Tcha and S.T. Pliska, Optimal Controlof single-server queing networks and multi-class M/G/1 queues with feedback, Operation Research, Vol. 25, No. 2, March-April 1977

[18] L. Kleinrock and R. P. Finkelstein, Time dependent priority queues, Operation Research 15:104-116, 1967.

[19] A. Netterman and I. Adiri, A dynamic priority queue with general concave priority functions, Operation Research 27:1088-1100, 1979.

[20] R. Nelson, Heavy traffic response times for a priority queue with linear priorities, Operation Research, 38:560-563, 1990

[21] U. Bagchi and R. Sullivan, Dynamic, Non-preemptive priority queues with general, linearly increasing priority function, Operation Research, 33:1278-1298, 1985

[22] P.A. Franaszek and R.D. Nelson, Properties of delay-cost scheduling in time-sharing systems, IBM Journal of Research and Development, Vol. 39, 3:295-314, 1995

[23] L. Tassiulas and A. Ephremides, Dynamic server allocation to parallel queue with randomly varying connnectivity, IEEE Transaction on Information Theory, Vol. 39, pp. 466-478, March 1993

[24] E. Altman and H. Kushner, Control of polling in presence of vacations in heavy traffic with applications to satellite and mobile radio systems, Procedings of the 37rd Allerton Conference on Communication, Control, and Computing, Illinois, USA, Sept. 1999

[25] P. Bhagwat and P. Bhattacharya, A. Krishna, S. K. Tripathi, Enhancing throughput over wireless LANs using channel state dependent packet scheduling, INFOCOM, April, 1996

[26] S. Shakkottai and R. Srikant, The exponential rule or wireless scheduling: pathwise optimality and state space collapse, INFORM Telecommunication Conference, 2002

[27] R. Buche and H.J. Kushner, Control of mobile communications with time varying channels in heavy traffic, Lefschetz Center for Dynamic System Technical Report, Brown Univeristy, 2001

[28] V. Bharghavan, S. Lu and T. Nandagopal, Fair queueing in wireless networks: issues and approaches

[29] S. Shakkottai and A. L. Stolyar, Scheduling for multiple flows sharing a time-varying channel: the exponential rule, Bell Laboratories Technical Report, Decemenber 2000, submitted

[30] M. Andrews, K. Kumaran, K. Ramanan, A. L. Stolyar, R. Vijayakumar and P. Whiting, CDMA data QoS scheudling on the forward link with avriable channel conditions, Bell Laboratories Technical Report, April, 2000

[31] V. Tripathi, E. Visotsky, R. Peterson, and M. Honig, Reliability-based type ii hybrid arq schemes, Proc. Int. Conf. on Comm., Anchorage, Alaska, May 2003.

[32] Y.W. Blankenship and B.K. Classon, Bounding techniques applied to modulation and coding design, Proc. Third Generation Wireless and Beyond 2002, May, 2002

[33] A. Das, F. Khan, A. Nanda, A^2IR: an asynchronous and adaptive hybrid ARQ scheme for 3G evolution, Proc. Vehicular Technology Conference (VTC), Spring 2001

[34] G. Caire, D. Tuninetti, The throughput of hybrid-ARQ protocols for the gaussian collision channel, IEEE Trans. Inform. Thoery, Vol. 47, No. 5, pp20-25, Jul-Aug 2001

[35] J.M. Shea, Reliability-based hybrid ARQ, Electronics Letters, Vol. 38, 13:644-645, June 2002

[36] I. Land and P.A. Hoeher, Using the mean reliability as a design and stopping criterion for turbo codes, Proc. information Theory Workshop 2001, Cairns, Australia

[37] L. Bahl, J. Cocke, F. Jelinek and J. Raviv, Optimal decoding of linear codes for minimizing symbol error rate, IEEE Trans. Inform. Theory, pp. 284-287, March. 1974

[38] E. Visotsky, V. Tripathi and M. Honig, Optimum ARQ design: a dynamic programming approach, IEEE International Symposium on Information Theory, 2003, Yakohama, Japan

[39] Z. Liu, P. Nain and D. Towsley, Sample path methods in the control of queues, QUESTA, 21:293-335, 1995

[40] G. Weiss, Branching bandit processes, Prob. Eng. INform. Sci., Vol. 2, 269-278, 1988

[41] M. Dacre, K. Glazebrook and J. Niño-Mora, The achievalbe region approach to the optimal control of stochastic systems, J. R. Statist. So. B 61, Part 4, 747-791, 1999

[42] D. Bertsimas, J. C. Paschalidis and J. N. Tsitsiklis, Branching bandits and Klimov's problem, achievable region and side constriaints, IEEE Transactions on Automatic Control, Vol. 40, No. 12, December 1995

[43] D. P. Bertsekas, Dynamic programming and optimal control, second edition, Athena Scientific, Belmont, Massachusetts, 2001

[44] S. Lin and D. J. Costello, Error control coding - fundamentals and applications, Prentice-Hall, 1983

[45] P. Nain, P. Tsoucas and J. Walrand, Interchange arguments in stochastic scheduling, J. Appl. Prob. 27, 815-826, 1989