Optimal Dynamic Packet Scheduling in Wireless Relay Networks

Hussein Al-Zubaidy
e-mail: hussein@sce.carleton.ca

September 3, 2010

Abstract

In this paper, we present a model and a methodology to investigate the optimal packet scheduling policy in wireless relay networks. We define the optimal policy as the one that maximizes the network throughput. We model this network by a system of queues, representing the subscriber stations and the relay stations, with random channel connectivity. Time is slotted. The scheduler selects one of the system queues to be served in every time slot. In this system, we use finite state Markov channel (FSMC) to model the wireless channel. This problem is formulated as a Markov decision process (MDP). We solve this problem using value iteration for the optimal scheduling policy. We present results for several special cases. The numerical results obtained showed that the optimal policy is threshold type. Furthermore, in a system with homogeneous connectivity and arrivals, a policy that tries to balance the lengths of all the system queues is optimal.

1 Introduction

The wireless networks evolved towards IP-based, high-speed data networks. The IP-based infrastructure made possible the deployment of mature existing technologies, that were initially developed for wired IP networks. On the other hand, the higher data rate was always subject to wireless channel limitations. The wireless channel is inherently random and unpredictable. The bandwidth resources are scarce and have to be shared by all the users in the network. Many technologies, e.g., adaptive modulation and coding (AMC), hybrid ARQ, etc., were developed to mitigate the effect of channel randomness and increase the bandwidth efficiency.

In modern wireless networks, the data rate is adapted to the channel's signal-to-noise ratio (SNR) through the use of adaptive modulation and coding. This makes the achievable data rate highly dependent on the channel state. In order to mitigate the unpredictability of the user's channel, different types of diversity were proposed, e.g., time diversity, frequency diversity, spatial diversity and user diversity. The basic idea of the first three techniques is to combine (or choose among) multiple copies of the transmitted signal to maximize the received SNR and hence the data rate. User diversity, on the other hand, chooses a signal among the transmitted signals of different users to find the user with better channel state.

Traditionally, spatial diversity is achieved using multiple antennas at the transmission and reception sides. In this case, spatial diversity is accomplished using a technique called "multiple input multiple output" (MIMO). However, some low-cost user equipments that are available nowadays do not have multiple antennas and therefore are incapable of handling such technique. This means lower channel quality and lower data rates for those users. An alternative technique for this type of users is to use "cooperative diversity". This technique is a form of spatial diversity technique. Users use cooperation among themselves to achieve higher transmission rates. Single antenna users in a multi-user environment share their antennas in a virtual multi-antenna environment to enhance their data rates (Figure 1). In such environment, users act as relays for other users. Although diversity gain is achievable,

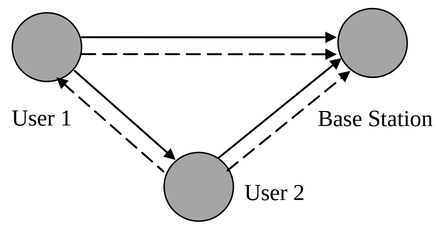


Figure 1: User cooperation in wireless networks.

the cost of this gain will manifest itself as elevated power and network resources consumption. The gain of user cooperation can be further exploited by optimally allocating the energy and bandwidth resources among users based on the channel state information (CSI) at every node [1].

In wireless networks with well-defined infrastructure, such as fourth generation (4G) wireless systems, another scheme to achieve cooperative diversity is proposed. In such scheme, dedicated wireless relay nodes (WRN) is spread over the coverage range of the cell. These relay nodes usually possess limited functionality and have low power consumption. They are significantly cheaper than a full scale base station. They have the effect of increasing coverage within a cell and facilitating the targeted 4G data rate of 100 Mbps for mobile users. A hybrid of the two techniques (WRN and user cooperation) is also possible.

Early studies of cooperative communication were initiated by [2], [3] and [4]. Since then, the subject has attracted the attention of many researchers as well as the industry, c.f. [5] [6]. IEEE 802.16j task group recommended the use of relay nodes in cellular networks design [7].

In 4G wireless networks, time is slotted into fixed length transmission frames. The channel resources may be scheduled in one of two modes. In the first mode, the transmission frame is divided into two periods. The first is dedicated for subscriber station (SS) nodes transmission, while the second is dedicated for Relay station (RS) nodes transmission. Then relay nodes do not have to compete with SS nodes for channel resources. In the second mode, all nodes (SS or RS nodes) are competing for channel resources.

Cellular networks (e.g. 4G wireless networks) are usually infrastructure-based networks. Therefore, a centralized controller can be easily deployed for the implementation of packet-scheduling function in such networks. In this case, the channel state information (CSI) will be available to the controller. This will enable the redistribution of the available resources to improve network performance. Optimal packet scheduling policies can be determined under various operating constraints to optimize various performance criteria and to achieve performance gain.

Most of the existing work in this area aimed at exploiting the diversity and multiplexing gain to improve some performance criteria, e.g., capacity and bandwidth utilization, outage probability, error rate, etc. These are often achieved through the use of adaptive modulation techniques, distributed space-time coding, or error-correction coding. However, optimization in these systems involves several layers and therefore requires a cross-layer design perspective.

In this work, we study the problem of cooperative communication in wireless cellular networks from new perspective, the dynamic packet scheduling perspective. We are interested in the scheduling of packets on the uplink. Each of the relaying nodes has a limited buffer capacity and its channel is time varying and can be modelled as a random process. The reward that we consider in this work is a function of the system throughput as well as queue occupancy.

The rest of this paper is organized as follows; In section 2, we present a queuing model for the relay network under investigation. In section 3, related literature was discussed. In section 4, the formulation of the optimal control problem as a MDP is presented. In section 5, we present value iteration, a technique used to determine the optimal policy. In section 6, the optimal policies for special cases are computed. Conclusions are given in section 7.

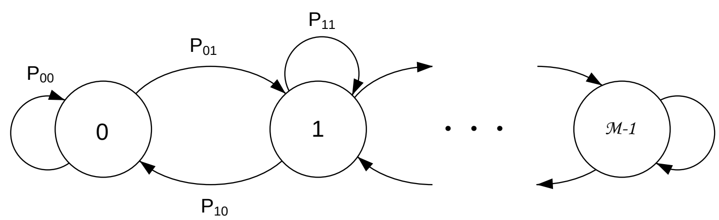


Figure 2: FSMC model for wireless channel.

2 Problem Definition and Model Description

2.1 Problem Definition

We consider a two-hop relay network that is composed of a base station (BS), L subscriber stations and K relay stations. L can be larger, smaller or equal to K. Each of the wireless channels connecting a SS to a RS and each RS to the base station is modelled by M-state finite state Markov channel (FSMC) following [8]. Assuming that the fading is slow enough that the channel states for consecutive time epochs are neighbouring states, then the model will be reduced into a discrete time birth and death process, as shown in Figure 2.

Modern cellular networks use pilot channel to estimate, among other things, the channel quality by measuring the received signal power at the receiving end [9]. This information can be made available to the controller with minimal efforts. In this work, we assume that the controller have complete knowledge of the channel state when the decision time arrives.

The transmission rate for a station is directly proportional to its channel state (which is a realistic assumption for rate adaptation in wireless networks) [10]. Furthermore, we assume that i packets can be transmitted from a given node at any given time frame when the node's respective channel is in state $i > 0$ and no packets can be transmitted when the channel is in state 0, i.e., disconnected. All channels are assumed to be stochastically independent from each other and from the arrivals to the subscriber stations. However, only one station (SS or RS) is allowed to transmit at a time. A special case of this problem is to have the SS nodes transmit independently from the RS nodes, i.e., using half a time frame for each transmission. The problem in this case will be reduced to finding the optimal scheduling policy for each of the two sets of nodes separately. We plan to investigate this case in our future research.

A scheduled packet completes transmission with probability 1 at a given time frame and therefore no retransmission is required. This is a simplifying assumption, however, the results obtained for this model can be easily adapted for a system with retransmission. We assume that the scheduler has complete knowledge of the system state at all times. This is a realistic assumption for cellular networks since they use a centralized control provided by the base station. They deploy a dedicated control channel that can be used to communicate such information.

2.2 Model Description

We model this network using a discrete-time queuing system[1] as shown in Figure 3. In this model, we assume a finite buffer size for the subscriber stations (denoted by B_s) as well as the relay stations (which we denote by B_r). The objective is to find the optimal dynamic packet scheduling policy (or equivalently, the channel allocation decision at every time slot) for this network. The optimal policy is the one that maximizes a reward function of the system throughput and queue lengths.

In this model, time is slotted into constant intervals each of which is equal to one transmission frame. Let $\mathcal{T}$ denote the set of decision epochs and $\mathcal{T} = \{1, 2, \ldots\}$. We assume that these decision epochs happens at the beginning of every time slot. A packet that arrives during the current time slot can only be considered for transmission in the subsequent time slots. At time $t \in \mathcal{T}$, we denote by $\mathbf{S}(t)$ and $\mathbf{D}(\mathbf{S})$, the system state and the scheduler decision at that state respectively.

We define the following notation that we use to describe the model under investigation. Throughout this paper, we will use UPPER CASE, **bold face** and lower case letters to

[1] In this model, the direct link between a subscriber station and the destination (the base station) can be considered as a degenerate case.

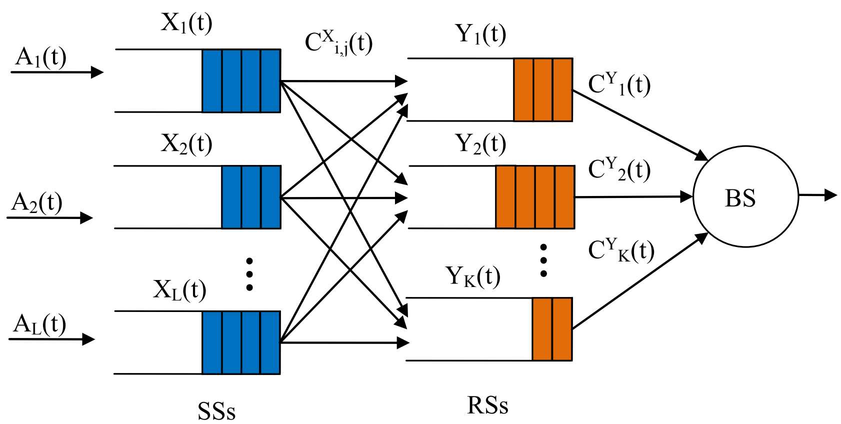


Figure 3: A model for packet scheduling in a 2-hop relay networks.

represent random variables, vector/matrix quantities and sample values respectively.

- $\mathbf{X}(t) = (X_1(t), X_2(t), \ldots, X_L(t))$ is the queue lengths vector for SS nodes (measured in number of packets) at time slot t. $X_i(t) \in \mathcal{X} = \{0, 1, 2, \ldots, B_s\}$.

- $\mathbf{Y}(t) = (Y_1(t), Y_2(t), \ldots, Y_L(t))$ is the queue lengths vector for RS nodes at time slot t. $Y_i(t) \in \mathcal{Y} = \{0, 1, 2, \ldots, B_r\}$.

- $\mathbf{A}(t) = (A_1(t), A_2(t), \ldots, A_L(t))$, where $A_i(t)$ is the number of exogenous arrivals to queue i during time slot t.

- $\mathbf{C^X}(t)$ is an $L \times K$ matrix, where $C^X_{i,j}(t) \in \mathcal{M} \equiv \{0, 1, \ldots, M-1\}$, is the channel connectivity random variable between SS node i and RS node j during time slot t, M is the number of states in the FSMC model.

- $\mathbf{C^Y}(t) = (C^Y_1(t), C^Y_2(t), \ldots, C^Y_K(t))$, where $C^Y_i(t) \in \mathcal{M}$ is the channel connectivity random variable between RS node i and the base station during time slot t.

- $\mathbf{U^X}(t)$ is an $L \times K$ matrix, where $U^X_{i,j}(t) = 1$ if SS node i is scheduled for transmission to RS node j during time slot t and $U^X_{i,j}(t) = 0$ otherwise.

- $\mathbf{U}^{\mathbf{Y}}(t) = (U_1^Y(t), U_2^Y(t), \ldots, U_K^Y(t))$, where $U_j^Y(t) = 1$ if RS node j is scheduled for transmission to the base station during time slot t and $U_i^Y(t) = 0$ otherwise.

- $\mathbf{S}(t) = (\mathbf{X}(t), \mathbf{Y}(t), \mathbf{C}^{\mathbf{X}}(t), \mathbf{C}^{\mathbf{Y}}(t))$ denotes the "state" of the system[2] at the beginning of time slot t.

The "decision" taken by the scheduler when the system is in state $\mathbf{s}(t)$ during time slot t is given by:

$$\mathbf{D}(\mathbf{s}) = (\mathbf{U}^{\mathbf{X}}(t), \mathbf{U}^{\mathbf{Y}}(t)), \tag{1}$$

subject to

$$\sum_{i=1}^{L} \sum_{j=1}^{K} U_{i,j}^X(t) + \sum_{j=1}^{K} U_j^Y(t) = 1, \tag{2}$$

This constraint insures that one node (SS or RS) is allowed to transmit at any given time slot.

We define the *withdrawal control vector* at time $t \in \mathcal{T}$ by:

$$\mathbf{Z}(t) = (\mathbf{Z}^{\mathbf{X}}(t), \mathbf{Z}^{\mathbf{Y}}(t)), \quad s.t. \tag{3}$$

- $\mathbf{Z}^{\mathbf{X}}(t) = (Z_1^X(t), Z_2^X(t), \ldots, Z_L^X(t))$ is the SS nodes withdrawal control vector, where

$$Z_i^X(t) = \sum_{j=1}^{K} U_{i,j}^X(t) \cdot C_{i,j}^X(t), \tag{4}$$

  for any i and $Z_i^X(t) \in \{0, 1, \ldots, M-1\}$ denotes the number of packets withdrawn from SS queue i during time slot t.

- $\mathbf{Z}^{\mathbf{Y}}(t) = (Z_1^Y(t), Z_2^Y(t), \ldots, Z_k^Y(t))$ is the RS nodes withdrawal control vector, where

$$Z_i^Y(t) = U_i^Y(t) \cdot C_i^Y(t), \tag{5}$$

  for any i and $Z_i^Y(t) \in \{0, 1, \ldots, M-1\}$ denotes the number of packets withdrawn from RS queue i during time slot t.

[2]The state information can be communicated through one of the available control channels in the network. Therefore, this information can be made available to the controller at the decision epoch.

The above definition will be useful in the computation of system state transition probability as we will show later.

The control decision $\mathbf{D}(\mathbf{s})$ that was defined in Equation (1) identify the node that is scheduled to transmit when in state $\mathbf{s}(t)$ during time slot t. On the other hand, $\mathbf{Z}(t)$ provides extra information regarding the number of packets withdrawn from that node's buffer. The two controls are related via Equations (4) and (5). The queue length for any SS node evolves according to the following relation:

$$X_i(t+1) = \max\left(\left[X_i(t) - Z_i^X(t)\right]^+ + A_i(t), B_s\right) \tag{6}$$

Similarly, the queue length evolution for any RS node is given by the following relation:

$$Y_j(t+1) = \max\left(\left[Y_j(t) - Z_j^Y(t)\right]^+ + \sum_{i=1}^{L} U_{i,j}^X(t) \cdot C_{i,j}^X(t), B_r\right) \tag{7}$$

The exogenous arrivals to any SS nodes are assumed to be Bernoulli with parameter q_i for node i and is given by:

$$A_i(t) = \begin{cases} v & \text{with probability } q_i, \quad v \in \mathcal{Z}_+ \\ 0 & \text{with probability } 1 - q_i \end{cases}$$

However, the arrivals to any RS node at time t is equal to the number of packets transmitted from any SS node to that RS node during that time slot. In either case, arrivals during time slot t will be added only after the removal of the scheduled packets at that time slot.

3 Previous Work and Our Contributions

This work lies in the area of optimal control in queuing systems. A related problem was first studied by Roseburg et al in [11]. They investigated an optimal control policy of service rate in a system of two queues in tandem. They proved, using dynamic programming argument, that a threshold policy is optimal. An extension to their work is to include random

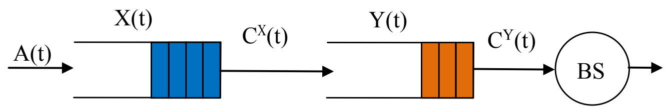


Figure 4: Two tandem queues with random connectivity

connectivity between the two queues to represent wireless channel, see Figure 4. This is a special case of the model we presented in Figure 3, where $L = K = 1$.

Another related problem is the one that was studied by Al-Zubaidy et al [12]. In this work, the authors investigated an optimal packet scheduling policy in a homogeneous system of parallel queues and multiple servers with independent server-queue connectivity. They proved, via stochastic dominance technique and coupling arguments, that a class of policies, namely the most balancing (MB) policies, is optimal in that it minimizes a class of cost functions of the system state. Our model, similar to this one, is characterised by the mesh connectivity pattern between the first set of queues (SS) and the second set of queues (RS).

In [13], the authors proposed a cooperative multiplexing and scheduling algorithm for a wireless relay network with a single relay node. They showed that this algorithm outperforms the traditional opportunistic techniques in terms of spectral efficiency. The authors in [14] studied link scheduling in WRN with bandwidth and delay guarantees. They modelled the system using simple directed graph. They proposed an efficient algorithm to provide delay guarantee over WRN. They proved that the worst-case performance of the proposed algorithm compared to an optimal one is bounded.

In [15], the authors presented opportunistic relaying with decode-and-forward (DaF) and amplify-and-forward (AaF) strategies under an aggregate power constraint. They showed that opportunistic DaF relaying is outage-optimal. Furthermore, they showed that opportunistic AaF relaying is outage-optimal among single-relay selection methods. The authors in [16] proposed a utility maximization framework for wireless cooperative cellular data network with a base station and many cooperative subscribers. They claim that their strategy optimizes relay selection as well as power allocation.

For a wireless relay network, the choices of relay node, relay strategy, and the allocation of power and bandwidth for each user are important design parameters which were investigated thoroughly in recent literature. Relay selection and cooperation strategies for relay networks have been investigated by [17] and [18] among others. Power control has been investigated by [19] and [20] and many others. However, the modern wireless networks are mostly IP-based, and therefore the optimization problem may be reduced to finding the optimal dynamic packet scheduling policies in these networks.

In this work, we developed a queuing model to study the function of packet scheduling in wireless relay network. Our main contributions can be summarized by the following:

- A queuing model is developed to study packet scheduling in wireless relay networks.
- We present a methodology for the optimization of WRN from dynamic packet scheduling perspective.
- The optimal scheduling problem is formulated using dynamic programming and a reward function is proposed.
- A program is developed to compute the optimal policy numerically using value iteration.
- The optimal packet scheduling policy is computed and presented graphically for several special cases.

Dynamic programming is used widely in the investigation of optimal control problems. Our literature survey revealed several proposed algorithms for packet scheduling in WRN. Many of the scheduling policies resulted from these algorithms may outperform other existing policies under some operating conditions but they are not optimal. The theoretical as well as practical value of the optimal scheduling policy is what motivated this work.

Our main goal is to study the structural characteristics of the optimal policy. The next step in our investigation of this problem is to provide theoretical proofs for the optimality of these policies using dynamic programming and/or stochastic coupling arguments. This is part of our proposed future work.

In the following section, we present a dynamic programming formulation [21], [22] for the model in Figure 3 and we solve it numerically for the optimal channel allocation policy using value iteration.

4 MDP Formulation

In this section we will present an approach, based on Markov Decision Process (MDP), to find the optimal packet scheduling policy for the model presented in Figure 3. In order to do that, we need to define the quintuple $(\mathcal{T}, \mathcal{S}, \mathcal{D}, P_{\mathbf{ss'}}(\mathbf{d}), R_T(\mathbf{s}, \mathbf{d}))$, where $\mathcal{S}$ and $\mathcal{D}$ are the state and decision spaces, $P_{\mathbf{ss'}}(\mathbf{d}) \triangleq Pr(\mathbf{S}(t+1) = \mathbf{s}'|\mathbf{S}(t) = \mathbf{s}, \mathbf{D}(\mathbf{s}) = \mathbf{d})$ is the state transition probability, and $R_T(\mathbf{s}, \mathbf{d})$ is the instantaneous reward function (to be defined shortly) when at state $\mathbf{s}$ and taking the decision $\mathbf{d}$. The decision space is the set of all possible controls $\mathbf{D}(\mathbf{s})$.

Note that for a given $L < \infty$ and $K < \infty$, the state space of the system, $\mathcal{S} = \{\mathcal{X}^L \times \mathcal{Y}^K \times \mathcal{M}^{K(L+1)}\}$, has finite size since the buffers have finite sizes and the channel state space $(\mathcal{M})$ is also finite.

4.1 State Transition Probability

The state transition probability $P_{\mathbf{ss'}}(\mathbf{d})$ is the probability that choosing the decision $\mathbf{d}$ at time slot t when the system state is $\mathbf{s}$ will result in the state $\mathbf{s}'$ at time slot $t+1$. Using the definitions of the system state $\mathbf{S}(t)$ and the decision $\mathbf{D}(t)$, $P_{\mathbf{ss'}}(\mathbf{d})$ can be stated as follows

$$\begin{aligned} P_{\mathbf{ss'}}(\mathbf{d}) &\triangleq Pr(\mathbf{S}(t+1) = \mathbf{s}'|\mathbf{S}(t) = \mathbf{s}, \mathbf{D}(t) = \mathbf{d}) = Pr(x_1', \ldots, x_L', y_1', \ldots, y_K', c_{1,1}^{x'}, \ldots, c_{L,K}^{x'}, \\ &c_1^{y'}, \ldots, c_K^{y'}|x_1, \ldots, x_L, y_1, \ldots, y_K, c_{1,1}^x, \ldots, c_{L,K}^x, c_1^y, \ldots, c_K^y, u_{1,1}^x, \ldots, u_{L,K}^x, u_1^y, \ldots, u_K^y) \end{aligned} \quad (8)$$

where $\mathbf{s}' = (\mathbf{x}', \mathbf{y}', \mathbf{c}^{\mathbf{x}'}, \mathbf{c}^{\mathbf{y}'})$ and $\mathbf{d} = (\mathbf{u}^{\mathbf{x}}, \mathbf{u}^{\mathbf{y}})$. where the primed variables represent the values of those variables during time slot $t+1$.

The joint conditional probability in Equation (8) can be decomposed, using conditioning, as follows:

$$
\begin{aligned}
P_{\mathbf{ss}'}(\mathbf{d}) = \ & Pr(x'_1, \ldots, x'_L, y'_1, \ldots, y'_K | x_1, \ldots, x_L, y_1, \ldots, y_K, c^x_{1,1}, \ldots, c^x_{L,K}, c^y_1, \ldots, c^y_K, \\
& u^x_{1,1}, \ldots, u^x_{L,K}, u^y_1, \ldots, u^y_K) \cdot Pr(c^{x'}_{1,1}, \ldots, c^{x'}_{L,K}, c^{y'}_1, \ldots, c^{y'}_K | x'_1, \ldots, x'_L, y'_1, \ldots, y'_K, x_1, \\
& \ldots, x_L, y_1, \ldots, y_K, c^x_{1,1}, \ldots, c^x_{L,K}, c^y_1, \ldots, c^y_K, u^x_{1,1}, \ldots, u^x_{L,K}, u^y_1, \ldots, u^y_K)
\end{aligned}
\tag{9}
$$

Applying conditioning again to the second part of Equation (9) we have:

$$
\begin{aligned}
P_{\mathbf{ss}'}(\mathbf{d}) = \ & Pr(x'_1, \ldots, x'_L, y'_1, \ldots, y'_K | x_1, \ldots, x_L, y_1, \ldots, y_K, c^x_{1,1}, \ldots, c^x_{L,K}, c^y_1, \ldots, c^y_K, \mathbf{d}) \\
& \cdot Pr(c^{x'}_{1,1} | x'_1, \ldots, x'_L, y'_1, \ldots, y'_K, x_1, \ldots, x_L, y_1, \ldots, y_K, c^x_{1,1}, \ldots, c^x_{L,K}, c^y_1, \ldots, c^y_K, \mathbf{d}) \cdot \ldots \\
& \cdot Pr(c^{x'}_{L,K} | c^{x'}_{1,1}, \ldots, c^{x'}_{L,K-1}, x'_1, \ldots, x'_L, y'_1, \ldots, y'_K, x_1, \ldots, x_L, y_1, \ldots, y_K, c^x_{1,1}, \ldots, c^x_{L,K}, \\
& c^y_1, \ldots, c^y_K, \mathbf{d}) \cdot Pr(c^{y'}_1 | c^{x'}_{1,1}, \ldots, c^{x'}_{L,K}, x'_1, \ldots, x'_L, y'_1, \ldots, y'_K, x_1, \ldots, x_L, y_1, \ldots, y_K, \\
& c^x_{1,1}, \ldots, c^x_{L,K}, c^y_1, \ldots, c^y_K, \mathbf{d}) \cdot \ldots \cdot Pr(c^{y'}_K | c^{x'}_{1,1}, \ldots, c^{x'}_{L,K}, c^{y'}_1, \ldots, c^{y'}_{K-1}, x'_1, \ldots, x'_L, \\
& y'_1, \ldots, y'_K, x_1, \ldots, x_L, y_1, \ldots, y_K, c^x_{1,1}, \ldots, c^x_{L,K}, c^y_1, \ldots, c^y_K, \mathbf{d})
\end{aligned}
\tag{10}
$$

Recall that the wireless channel is modelled by a Markov process and it is assumed that channels for different nodes are independent. Therefore, the next channel state depends only on the current channel state. Therefore, the channel state transition probability can be expressed as follows:

$$Pr(c^{x'}_{i,j} | \mathbf{s}) = Pr(c^{x'}_{i,j} | c^x_{i,j}) \triangleq P_{c^x_{i,j} c^{x'}_{i,j}} \tag{11}$$

and,

$$Pr(c^{y'}_k | \mathbf{s}) = Pr(c^{y'}_k | c^y_k) \triangleq P_{c^y_k c^{y'}_k} \tag{12}$$

Hence, Equation (10) can be rewritten as follows:

$$
\begin{aligned}
P_{\mathbf{ss}'}(\mathbf{d}) \ = \ & Pr(x'_1, \ldots, x'_L, y'_1, \ldots, y'_K | x_1, \ldots, x_L, y_1, \ldots, y_K, c^x_{1,1}, \ldots, c^x_{L,K}, c^y_1, \ldots, c^y_K, \mathbf{d}) \\
\cdot \ & \prod_{i=1}^{L} \prod_{j=1}^{K} P_{c^x_{i,j} c^{x'}_{i,j}} \cdot \prod_{k=1}^{K} P_{c^y_k c^{y'}_k}
\end{aligned}
\tag{13}
$$

Following the same approach, the queue size joint probability, i.e., the first term in Equation (13), can be decomposed as follows:

$$\begin{aligned}
&Pr(x'_1,\ldots,x'_L,y'_1,\ldots,y'_K|x_1,\ldots,x_L,y_1,\ldots,y_K,c^x_{1,1},\ldots,c^x_{L,K},c^y_1,\ldots,c^y_K,\mathbf{d})\\
&= Pr(x'_1|x_1,\ldots,x_L,y_1,\ldots,y_K,c^x_{1,1},\ldots,c^x_{L,K},c^y_1,\ldots,c^y_K,\mathbf{d})\cdot\ldots\\
&\quad \cdot Pr(x'_L|x'_1,\ldots,x'_{L-1},x_1,\ldots,x_L,y_1,\ldots,y_K,c^x_{1,1},\ldots,c^x_{L,K},c^y_1,\ldots,c^y_K,\mathbf{d})\\
&\quad \cdot Pr(y'_1|x'_1,\ldots,x'_L,x_1,\ldots,x_L,y_1,\ldots,y_K,c^x_{1,1},\ldots,c^x_{L,K},c^y_1,\ldots,c^y_K,\mathbf{d})\cdot\ldots\\
&\quad \cdot Pr(y'_K|x'_1,\ldots,x'_L,y'_1,\ldots,y'_{K-1},x_1,\ldots,x_L,y_1,\ldots,y_K,c^x_{1,1},\ldots,c^x_{L,K},c^y_1,\ldots,c^y_K,\mathbf{d}) \qquad (14)
\end{aligned}$$

The queue length for each SS node (respectively RS node) evolves according to Equation (6) (respectively Equation (7)). From Equation (6) and the independence assumption of the different queues random variables (arrivals and channel connectivities), we can see that for SS queue i, x'_i (the queue size at time $t+1$) depends on x_i (its current queue size), $c^x_{i,1},\ldots,c^x_{i,K}$ (its connectivity state), $u^x_{i,1},\ldots,u^x_{i,K}$ (the decision) and a_i (the arrivals to queue i during time slot t) and independent of everything else. Therefore,

$$Pr(x'_i|\mathbf{s}) = Pr(x'_i|x_i,c^x_{i,1},\ldots,c^x_{i,K},u^x_{i,1},\ldots,u^x_{i,K}) \triangleq P_{x_ix'_i}(\mathbf{c^x},\mathbf{d}) \qquad (15)$$

Similarly, Equation (7) show that for RS queue j, y'_j (the queue size at time $t+1$) depends on y_j (the current queue size), its connectivity state c^y_j, $c^x_{1,j},\ldots,c^x_{L,j}$ and the decision $u^x_{1,j},\ldots,u^x_{L,j},u^y_j$ and independent of the other variables in the system. Therefore,

$$Pr(y'_j|\mathbf{s}) = Pr(y'_j|y_j,c^x_{1,j},\ldots,c^x_{L,j},c^y_j,u^x_{1,j},\ldots,u^x_{L,j},u^y_j) \triangleq P_{y_jy'_j}(\mathbf{c^x},\mathbf{c^y},\mathbf{d}) \qquad (16)$$

Note that the arrivals to RS queue (i.e., relay node) j at any given time slot is equal to $\sum_{i=1}^{L} u^x_{i,j}\cdot c^x_{i,j}$ packets.

In light of the above, Equation (13) is reduced to the following:

$$P_{\mathbf{ss'}}(\mathbf{d}) = \prod_{i=1}^{L} P_{x_ix'_i}(\mathbf{c^x},\mathbf{d}) \prod_{k=1}^{K} P_{y_ky'_k}(\mathbf{c^x},\mathbf{c^y},\mathbf{d}) \cdot \prod_{i=1}^{L}\prod_{j=1}^{K} P_{c^x_{i,j}c^{x'}_{i,j}} \cdot \prod_{k=1}^{K} P_{c^y_kc^{y'}_k} \qquad (17)$$

Equation (17) states that the state transition probability is the product of the individual queue state transition probabilities of the $L \times K$ stations in the system and their FSMC state transition probabilities. According to [8], the FSMC channel state transition probabilities can be estimated from practical measurements and made available to the scheduler. Next we will derive the queue state transition probabilities for the SS and RS nodes.

4.1.1 Determination of SS queue state transition probability

The queue state transition probability for SS i is given by

$$P_{x_i x_i'}(\mathbf{c^x}, \mathbf{d}) = Pr(X_i(t+1) = x_i' | X_i(t) = x_i, C_{i,1}^X(t) = c_{i,1}^x, \ldots, C_{i,K}^X(t) = c_{i,K}^x, \\ U_{i,1}^X(t) = u_{i,1}^x, \ldots, U_{i,K}^X(t) = u_{i,K}^x) \tag{18}$$

The queue length $X_i(t+1)$ is related to $X_i(t)$ via Equation (6). Let

$$w_i^s = [x_i - z_i^x]^+$$

From Equation (6), we can partition the probability space into the following two cases:

Case (1): $\left[X_i(t) - Z_i^X(t)\right]^+ + A_i(t) < B_s$, where $Z_i^X(t)$ is defined by Equation (4). In this case Equation (6) reduces to:

$$X_i(t+1) = \left[X_i(t) - Z_i^X(t)\right]^+ + A_i(t) \tag{19}$$

and Equation (18) can be rewritten as:

$$\begin{aligned} P_{x_i x_i'}(\mathbf{c^x}, \mathbf{d}) &= Pr(\left[X_i(t) - Z_i^X(t)\right]^+ + A_i(t) = x_i' | X_i(t) = x_i, C_{i,1}^X(t) = c_{i,1}^x, \\ &\quad \ldots, C_{i,K}^X(t) = c_{i,K}^x, U_{i,1}^X(t) = u_{i,1}^x, \ldots, U_{i,K}^X(t) = u_{i,K}^x) \\ &= Pr([x_i - z_i^x]^+ + A_i(t) = x_i') \\ &= Pr(A_i(t) = x_i' - [x_i - z_i^x]^+) \end{aligned} \tag{20}$$

where,

$$z_i^x = \sum_{j=1}^{K} u_{i,j}^x \cdot c_{i,j}^x \tag{21}$$

Recall that the arrivals to SS queue i has Bernoulli distribution with parameter q_i. Therefore, the queue state transition probability in this case is given by:

$$P_{x_i x_i'}(\mathbf{c^x}, \mathbf{d}) = \begin{cases} q_i & \text{if } x_i' = w_i^s + v \\ 1 - q_i & \text{if } x_i' = w_i^s \\ 0 & \text{Otherwise} \end{cases} \tag{22}$$

Case (2): $\left[X_i(t) - Z_i^X(t)\right]^+ + A_i(t) \geq B_s$ or equivalently $X_i(t+1) = B_s$. In this case, Equation (18) can be rewritten as:

$$\begin{aligned} P_{x_i B_s}(\mathbf{c^x}, \mathbf{d}) &= Pr(\left[X_i(t) - Z_i^X(t)\right]^+ + A_i(t) \geq B_s | X_i(t) = x_i, \ldots, C_{i,1}^X(t) = c_{i,1}^x, \\ &\quad C_{i,K}^X(t) = c_{i,K}^x, U_{i,1}^X(t) = u_{i,1}^x, \ldots, U_{i,K}^X(t) = u_{i,K}^x) \\ &= Pr([x_i - z_i^x]^+ + A_i(t) \geq B_s) \\ &= Pr(A_i(t) \geq B_s - [x_i - z_i^x]^+) \end{aligned} \tag{23}$$

Similar to case (1), we can state the queue state transition probability as follows:

$$P_{x_i B_s}(\mathbf{c^x}, \mathbf{d}) = \begin{cases} 1 - q_i & \text{if } w_i^s = B_s \\ q_i & \text{if } w_i^s < B_s, w_i^s + v \geq B_s \\ 0 & \text{Otherwise} \end{cases} \tag{24}$$

Equation (24) defines the transition probability of queue i from state x_i to the boundary state B_s (i.e., full buffer state).

The two cases above are the only possible cases. The union of these two probability space partitions represents the SS queue state transition probability. This probability can be summarized as follows:

$$P_{x_i x_i'}(\mathbf{c^x}, \mathbf{d}) = \begin{cases} 1 - q_i & \text{if } x_i' = w_i^s = B_s \\ q_i & \text{if } x_i' = B_s, w_i^s < B_s, w_i^s + v \geq B_s \\ q_i & \text{if } x_i' = w_i^s + v < B_s \\ 1 - q_i & \text{if } x_i' = w_i^s < B_s \\ 0 & \text{Otherwise} \end{cases} \tag{25}$$

4.1.2 Determination of RS queue state transition probability

The queue state transition probability for RS node j is given by

$$P_{y_j y_j'}(\mathbf{c^x}, \mathbf{c^y}, \mathbf{d}) = Pr(Y_j(t+1) = y_j' | Y_j(t) = y_j, C_{1,j}^X(t) = c_{1,j}^x, \ldots, C_{L,j}^X(t) = c_{L,j}^x,$$
$$C_j^Y(t) = c_j^y, UX_{1,j}(t) = u_{1,j}^x, \ldots, UX_{L,j}(t) = u_{L,j}^x, U_j^Y(t) = u_j^y) \tag{26}$$

The queue length $Y_j(t+1)$ is related to $Y_j(t)$ via Equation (7). From Equation (6), we can see that given the state $\mathbf{s}$ and the decision $\mathbf{d}$, the queue state y_j' can be deterministically calculated through Equation (7). Let

$$w_j^r = \left[y_j - z_j^y\right]^+ + \sum_{i=1}^{L} u_{i,j}^x \cdot c_{i,j}^x,$$

then the RS queue state transition probability can be stated as follows:

Case (1): $\left[Y_j(t) - Z_j^Y(t)\right]^+ + \sum_{i=1}^{L} U_{i,j}^X(t) \cdot C_{i,j}^X(t) < B_r$, where $Z_j^Y(t)$ is defined by Equation (5). Then

$$P_{y_j y_j'}(\mathbf{c^x}, \mathbf{c^y}, \mathbf{d}) = \begin{cases} 1 & \text{if } w_j^r = y_j' \\ 0 & \text{if } w_j^r \neq y_j' \end{cases} \tag{27}$$

Case (2): $\left[Y_j(t) - Z_j^Y(t)\right]^+ + \sum_{i=1}^{L} U_{i,j}^X(t) \cdot C_{i,j}^X(t) \geq B_r$ or equivalently $Y_j(t+1) = B_r$. In this case

$$P_{y_j B_r}(\mathbf{c^x}, \mathbf{c^y}, \mathbf{d}) = \begin{cases} 1 & \text{if } w_j^r \geq B_r \\ 0 & \text{if } w_j^r < B_r \end{cases} \tag{28}$$

The two cases above can be summarized as follows:

$$P_{y_j y'_j}(\mathbf{c^x}, \mathbf{c^y}, \mathbf{d}) = \begin{cases} 1 & \text{if } y'_j = B_r, w^r_j \geq B_r \\ 1 & \text{if } 0 \leq y'_j < B_r, y'_j = w^r_j \\ 0 & \text{Otherwise} \end{cases} \tag{29}$$

This conclude the derivation of the state transition probability. In the next section, we compute the optimal policy for some case studies.

4.2 Reward Function

At every time slot t, we define the instantaneous reward $R_T(\mathbf{s}, \mathbf{d})$ that is a function of the system state and the decision taken during time slot t. The investigated optimal policy is the one that maximizes this reward function at every time slot.

Let $w^s_i(t) = [x_i(t) - z^x_i(t)]^+$, and let $w^r_j(t) = \left[y_j(t) - z^y_j(t)\right]^+ + \sum_{i=1}^{L} u^x_{i,j}(t) \cdot c^x_{i,j}(t)$

For the model under consideration, we define the reward function $R_T(\mathbf{s}, \mathbf{d})$ as:

$$R_T(\mathbf{s}, \mathbf{d}) = \sum_{i=1}^{K} \min\Big(u^y_i(t) \cdot c^y_i(t), y_i(t)\Big) - c\Big(\sum_{j=1}^{L} \mathbb{1}_{\{w^s_j(t)+v>B_s\}} + \sum_{k=1}^{K} \mathbb{1}_{\{w^r_k(t)>B_r\}}\Big) \tag{30}$$

where $c \geq 0$.

The first part of the suggested reward function (the first term in Equation (30)) represents the number of packets received by the base station. This is indicative of the system throughput. The second part (i.e., the second term in Equation (30)) is a cost for actions resulting in packet drops, e.g., an action that involves routing a packet from SS station to a full RS buffer.

5 Optimal Policy Determination

In this section, we will use value iteration to determine the optimal scheduling policy for the model we presented earlier. We investigate an infinite-horizon MDP. To find the optimal scheduling policy, we use the expected discounted cost optimality criterion with a discount factor λ, where $0 < \lambda < 1$. A policy π defines a sequence of measurable functions $(\mathbf{D}(\mathbf{s}))$ of the system state $\mathbf{s}(t)$ at every time slot t. The optimal policy is the policy π, among all policies, that maximizes the *value function* $V^{\pi}(s)$.

Let $V^*(\mathbf{s})$ be the maximal discounted value function (i.e., $V^*(\mathbf{s}) = \sup_{\pi} V^{\pi}(\mathbf{s})$), attained when applying the optimal policy π^*. Then we use the following optimality equation (Bellman equation) to characterize the optimal policy [21]

$$V^*(\mathbf{s}) = \max_{\mathbf{d}\in\mathcal{D}} [R_T(\mathbf{s}, \mathbf{d}) + \lambda \sum_{\mathbf{s}'\in\mathcal{S}} P_{\mathbf{s}\mathbf{s}'}(\mathbf{d}) V^*(\mathbf{s}')] \tag{31}$$

We use *Value iteration* to solve this model numerically for the optimal policy. We define $V_0(\mathbf{s})$ to be an arbitrary bounded function. Then $V_n(\mathbf{s})$, $n > 0$ can be determined using the following recursion

$$V_n(\mathbf{s}) = \max_{\mathbf{d}\in\mathcal{D}} [R_T(\mathbf{s}, \mathbf{d}) + \lambda \sum_{\mathbf{s}'\in\mathcal{S}} P_{\mathbf{s}\mathbf{s}'}(\mathbf{d}) V_{n-1}(\mathbf{s}')] \tag{32}$$

The function V_n converges to V^* as $n \to \infty$ [22]. The algorithm will be stopped after n iterations, provided that

$$\|V_{n+1} - V_n\| < \epsilon(1-\lambda)/2\lambda \tag{33}$$

for a given $\epsilon > 0$, where $\|v\| = \sup_{\mathbf{s}\in\mathcal{S}} |v(\mathbf{s})|$. If (33) holds, then $\|V_{n+1} - V^*\| < \epsilon/2$, according to [21].

6 Results and Analysis

In this section, the optimal policy is computed and presented for several special cases of the general model presented in section 2 using value iteration, as discussed in section 5 above. The cases we studied here corresponds to the following[3]: (i) $L = K = 1$, (ii) $L = K = 2$, and

[3]We studied other cases for this model and the results match those presented for the cases we present here. We did not include them here for brevity.

(ii) $L = 1$, $K = 2$. The results obtained from studying these cases, regarding the optimal policy structural characteristics, can be extended to the general case.

In all the cases we studied, the following parameters are used: $\lambda = 0.95$, $\epsilon = 0.1$, $c = 1$, $v = 1$ and $M = 2$. Let $\mathbf{p}$ be the two-state channel transition probability matrix, where

$$\mathbf{p} = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix} \tag{34}$$

Therefore, the channel is "connected" with probability $\pi_1 = \frac{\alpha}{\alpha+\beta}$ and "not-connected" otherwise (i.e., $\pi_0 = \frac{\beta}{\alpha+\beta}$), where π is the stationary distribution of the Markov chain for the channel model in this system.

6.1 Case Study 1: $L = K = 1$

We computed the optimal policy for the case $L = K = 1$, i.e., the case of two tandem queues with random connectivity. This case is shown in Figure 4.

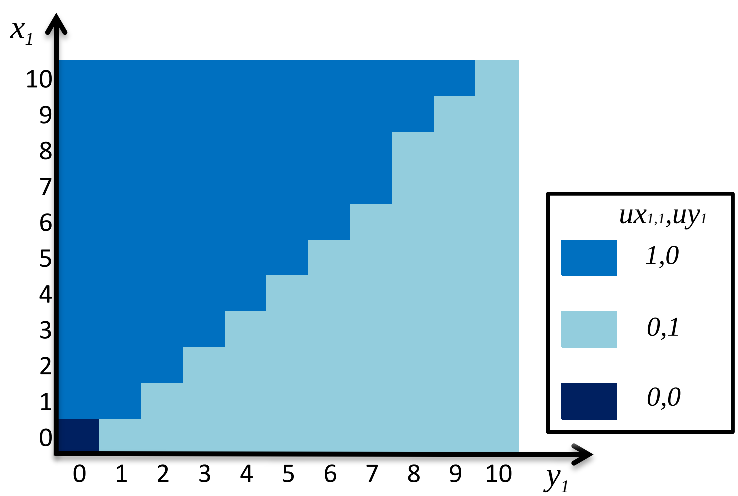
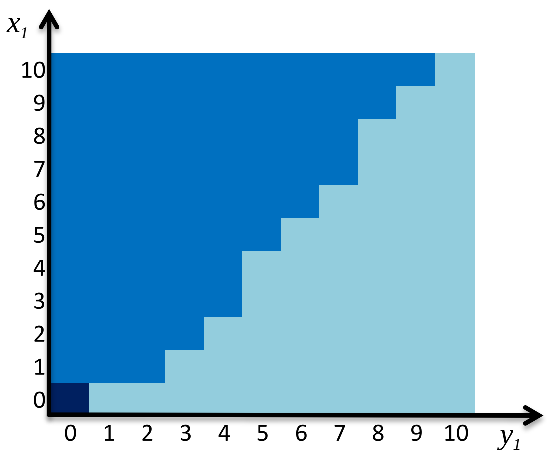
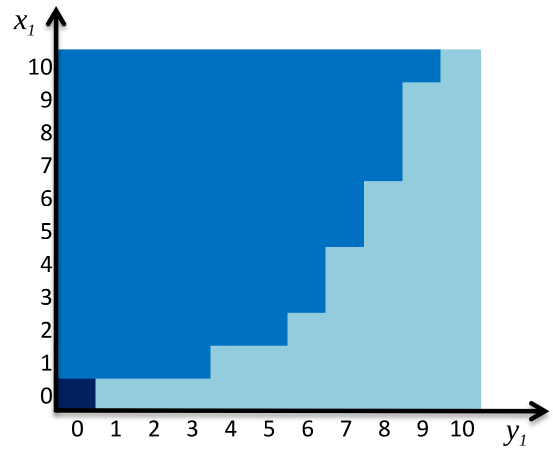


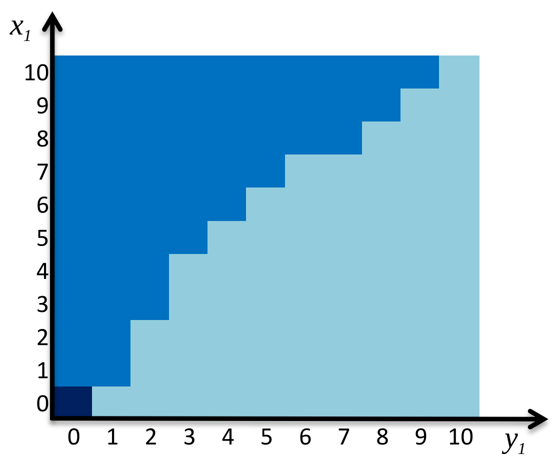
Figure 5: The optimal policy for case 1, $c^x_{1,1} = c^y_1 = 1$, $\alpha_1 = \alpha_2 = 0.6, \beta_1 = \beta_2 = 0.4$ and $q_1 = 0.5$.



(a) $\alpha_1 = \alpha_2 = 0.6, \beta_1 = \beta_2 = 0.4$ and $q_1 = 0.6$.



(b) $\alpha_1 = 0.5, \beta_1 = 0.5, \alpha_2 = 0.6, \beta_2 = 0.4$ and $q_1 = 0.5$.



(c) $\alpha_1 = 0.7, \beta_1 = 0.3, \alpha_2 = 0.6, \beta_2 = 0.4$ and $q_1 = 0.5$.

Figure 6: The optimal policy for case 1 with different parameters, $c^x_{1,1} = c^y_1 = 1$. The legend in this figure is the same as that in Figure 5.

The results show that the optimal policy allocates transmission slots only to connected queues. When only one queue is connected, then the optimal policy allocates the transmission slot to that queue. When both queues are not connected then the optimal policy will idle, i.e., no packets are transmitted. Figure 5 depicts the optimal policy for this system with homogeneous connectivity (i.e., $\alpha_1 = \alpha_2 = 0.6, \beta_1 = \beta_2 = 0.4$), when both queues are connected. In this case, both queues (corresponding to SS and RS nodes) are competing for the channel resources. Figure 6 shows the optimal policy for three other scenarios (of case 1) with different connectivity and arrival parameters.

From Figures 5 and 6, it is clear that the optimal policy is threshold policy as expected. This agrees with the conclusions in [11] for tandem queues. Furthermore, we noticed that the threshold depends on the connectivity and arrival parameters. Figure 5 also shows that in a homogeneous connectivity system a balancing policy, a policy that tries to balance the length of the two queues, is optimal.

Figure 6(a) shows the effect of arrival probability on the optimal policy structure. We noticed that when both queues are connected, higher arrival probability (i.e., higher arrival rate to SS node) shifts the threshold of the optimal policy in favour of the SS queue. This agrees with intuition, since the more packets arrive to SS node the faster the SS queue grows. Therefore, more resources (transmission slots) need to be allocated to this queue to mitigate the effect of higher arrival rate.

Figures 6(b) and 6(c) show that when both queues are connected the optimal policy favours the queue with lower channel connectivity. This makes intuitive sense, since the queue with higher channel connectivity has more opportunity to transmit than the other queue.

The case we studied in this subsection represents the core for many scheduling and routing problems in multi-hop wireless networks including wireless relay networks. The results we present in this section can be extended to many problems in multi-hop wireless networks and can shed light on the structure and behaviour of the optimal policy in such networks.

6.2 Case Study 2: $L = K = 2$

In this subsection, we determine the optimal policy for the model in Figure 3 with $L = K = 2$. We consider a system with homogeneous connectivity where $\alpha = \beta = 0.5$ for every channel in this system. We used value iteration to find the optimal policy for this case. The results show that the optimal policy schedules packets for transmission on connected channels only. Therefore, we elected to show the case where all the channels are connected, i.e., $c^x_{i,j} = c^y_j = 1, \forall i, j = 1, 2$. In this case, there are seven feasible actions for the policy to choose from. The first action is to idle the server, i.e., no transmission is scheduled during that time slot, $u^x_{i,j} = u^y_j = 0, \forall i, j = 1, 2$. The optimal policy chooses this action only when $x_1 = x_2 = y_1 = y_2 = 0$. The other six actions corresponds to either one of the following, $u^x_{1,1}, u^x_{1,2}, u^x_{2,1}, u^x_{2,2}, u^y_1$ or u^y_2 is equal to 1 while the rest equal to 0. The graphical depiction of this policy requires 5-dimensional representation of the four queue lengths (x_1, x_2, y_1, y_2) and the action chosen by the optimal policy. Since, to our best knowledge, a tool to represent a 5-dimensional graph on a 2-dimensional surface does not exist, and due to the importance of the pictorial depiction of scheduling policies over multi-dimensional space to our research, we developed a new method for representing these policies, please refer to Figure 7.

In Figure 7, we depict the four dimensions representing the queue lengths for the four queues in the system (i.e., x_1, x_2, y_1, y_2) as four axes in a 2-dimensional space. Each axis starts from the point of origin $(0, 0, 0, 0)$, that is represented by a small circle in the middle, and forms a right angle with two of the other three axes. We use colour to represent the fifth dimension, the one corresponding to the selected action by the policy under consideration. In other words, we use different colours to represent each of the seven possible actions. Each intersection point of the four axes in the original 5-dimensional space (i, j, k, l, a) is represented by a closed curve that intersects all the four axes in the proposed two-dimensional graph at points $x_1 = i$, $x_2 = j$, $y_1 = k$ and $y_2 = l$, i.e., a point in the original 5-dimensional space is mapped to a closed curve in the 2-dimensional depiction of the graph. The fifth point (a) is represented by one of seven different colours. To represent a point that has one (or more) axis with 0 value, e.g., $x_1 = 0$, then the curve is drawn such that it passes through

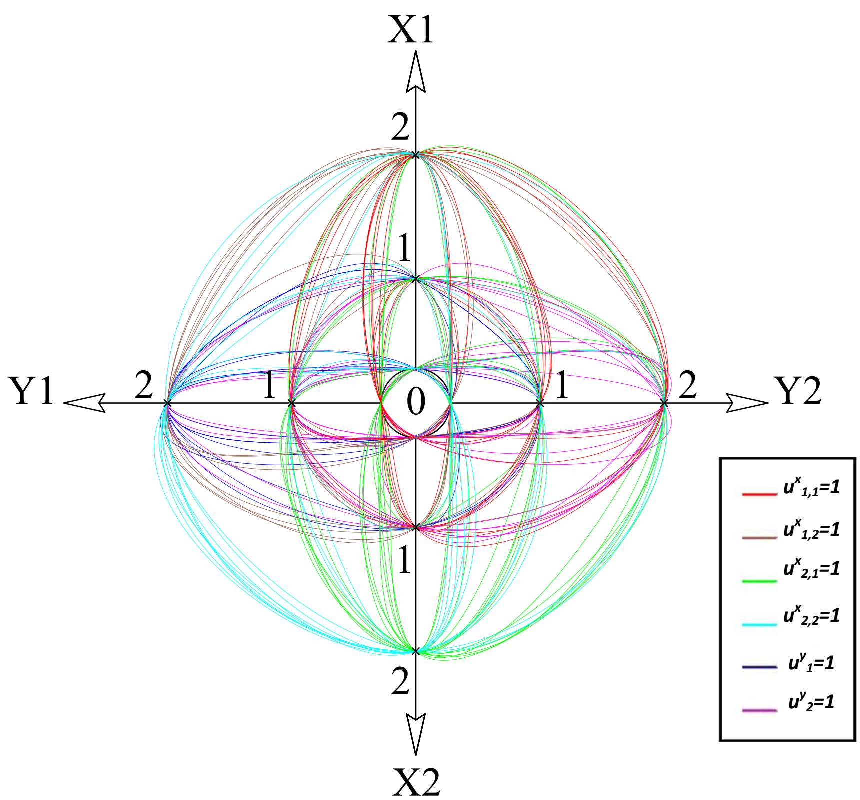
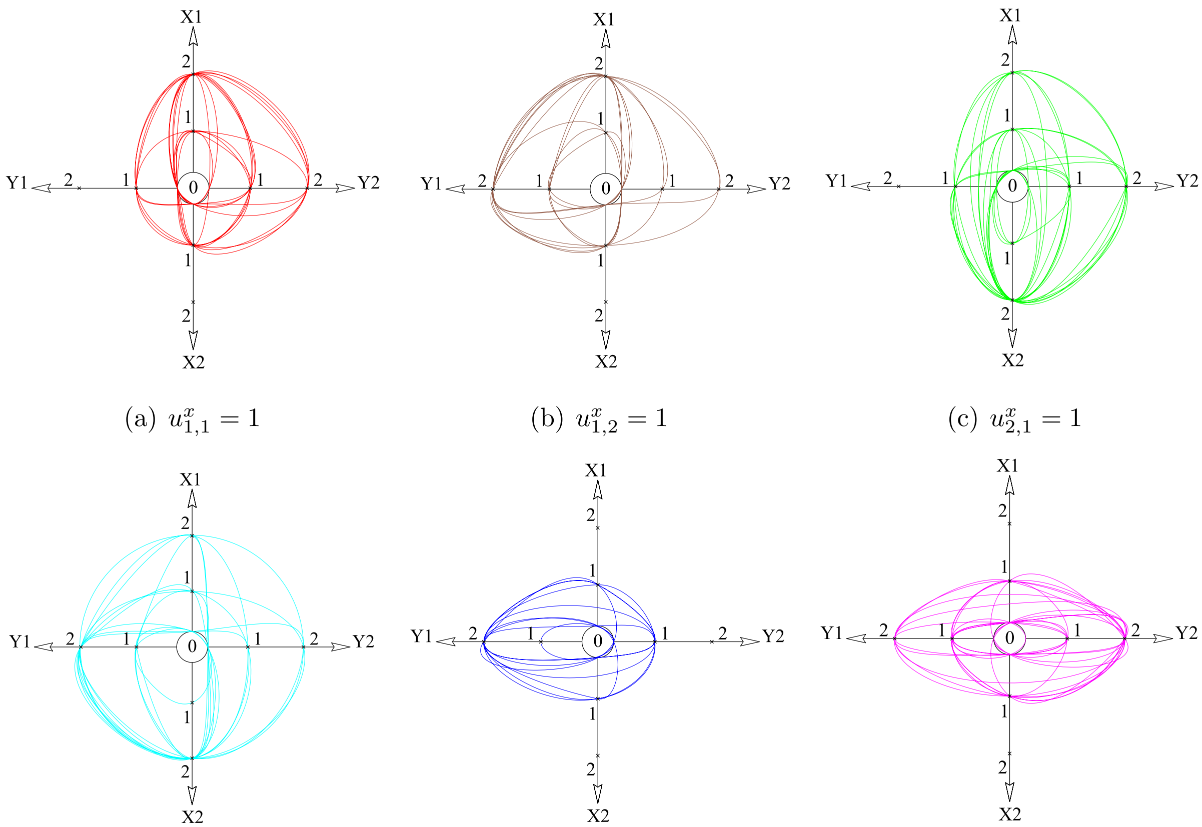


Figure 7: The optimal policy for case 2, $c^x_{i,j} = c^y_j = 1, \forall i, j$ and $q_1 = q_2 = 0.25$.



(a) $u^x_{1,1} = 1$ (b) $u^x_{1,2} = 1$ (c) $u^x_{2,1} = 1$

(d) $u^x_{2,2} = 1$ (e) $u^y_1 = 1$ (f) $u^y_2 = 1$

Figure 8: Optimal actions for case 2, $c^x_{i,j} = c^y_j = 1, \forall i, j$ and $q_1 = q_2 = 0.25$.

the point where that axis meets the center circle as shown in Figure 7.

The number of curves required to represent all combinations of the queue lengths increases exponentially with the number of queues in the system as well as the buffer size. In Figure 7 for example, there are $3^4 = 81$ curves correspond to all 81 combination of the four axis. Although we ran the program for larger buffer size, the figure shows a partial depiction of the computed policy. Such graph is easier to visualize and to comprehend than a more inclusive but cluttered one. The conclusions derived from this graph hold for the general case. Furthermore, from the experimental data we collected, we concluded that increasing the complexity of this graph will not add more information regarding the structure of the optimal policy.

To study the structure of the optimal policy for this case, we decompose the graph in Figure 7 into six graphs, one graph for every action as shown in Figure 8. From these figures we conclude that the optimal policy is a most balancing (MB) policy. Such policy tries to balance the queues in the system by allocating more resources (i.e., transmission slots) to the longest connected queue (LCQ). If this queue is a SS one then the served packet will be routed to the shortest connected RS queue, i.e., join the shortest queue (JSQ). To illustrate, Figure 8(a) shows the queue length combinations (represented as closed curves) where the optimal policy selects the action $u^x_{1,1} = 1$, i.e., a packet from queue x_1 is served and it is routed to queue y_1. A careful examination of this figure reveals that the optimal policy chooses this action only when $x_1 \geq x_2, y_1, y_2$ and $y_1 \leq y_2$. The equality reflects the cases where some queues have the same lengths as x_1 and hence choosing either one will result in the same reward. The density of the curves passing through a point on one axis compared to other axis is a visual indication of the above argument.

Figure 8(b) on the other hand, corresponding to the optimal action $u^x_{1,2} = 1$, shows that $x_1 \geq x_2, y_1, y_2$ while $y_1 \geq y_2$. Therefore, the shortest RS queue in this case is y_2 and hence the action $u^x_{1,2} = 1$ is selected.

Similar conclusions can be derived from the other four graphs in Figure 8. Notice that these actions will result in a balancing effect. This result matches our previous conclusions for case 1.

6.3 Case Study 3: $L = 1, K = 2$

In this subsection, we determine the optimal policy for the model in Figure 3 with $L = 1$ and $K = 2$. The arrival probability is $q_1 = 0.5$. In this case we select a two-state connectivity with $\alpha = \beta = 0.6$ for every channel in this system. We calculated the optimal policy for this case using value iteration. Similar to the previous cases, the results show that the optimal policy schedules transmissions on connected channels only. Therefore, we present the case where all the channels are connected, i.e., $c^x_{i,j} = c^y_j = 1, \forall i, j$. There are five feasible actions for the policy to choose from including the idling option. The optimal policy chooses idling option only when $x_1 = y_1 = y_2 = 0$. Each of the remaining actions corresponds to one of the following controls: $u^x_{1,1}, u^x_{1,2}, u^y_1$ and u^y_2, equals to 1 while the rest equal to 0. We use the same graphical depiction technique we introduced earlier to map the resulted (4-dimensional) policy on a 2-dimensional surface. The optimal policy for case 3 is shown in Figure 6.3. The plots for each of the four non-idling actions are shown in Figure 6.3.

From these figures we noticed that the optimal policy in case 3 have similar properties to those in cases 1 and 2. The optimal policy selects to serve the longest connected queue. From Figures 10(a) and 10(b), we can see that the scheduled packet is routed to the shortest queue. The optimal policy in this case is a balancing policy, i.e., a policy that tries to balance the three queues in the system.

7 Conclusion

In this work, we studied the wireless relay networks optimization problem from dynamic packet scheduling perspective. We provided a queuing model for these networks, then we formulated this problem as a Markov decision process. We solved the problem for several special cases, using value iteration, to obtain the optimal policy in each case. From these

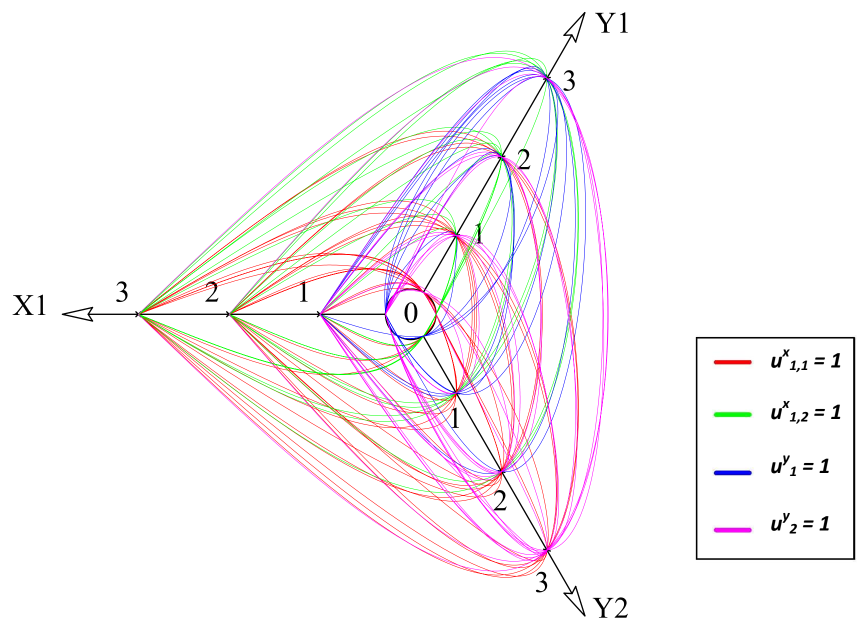
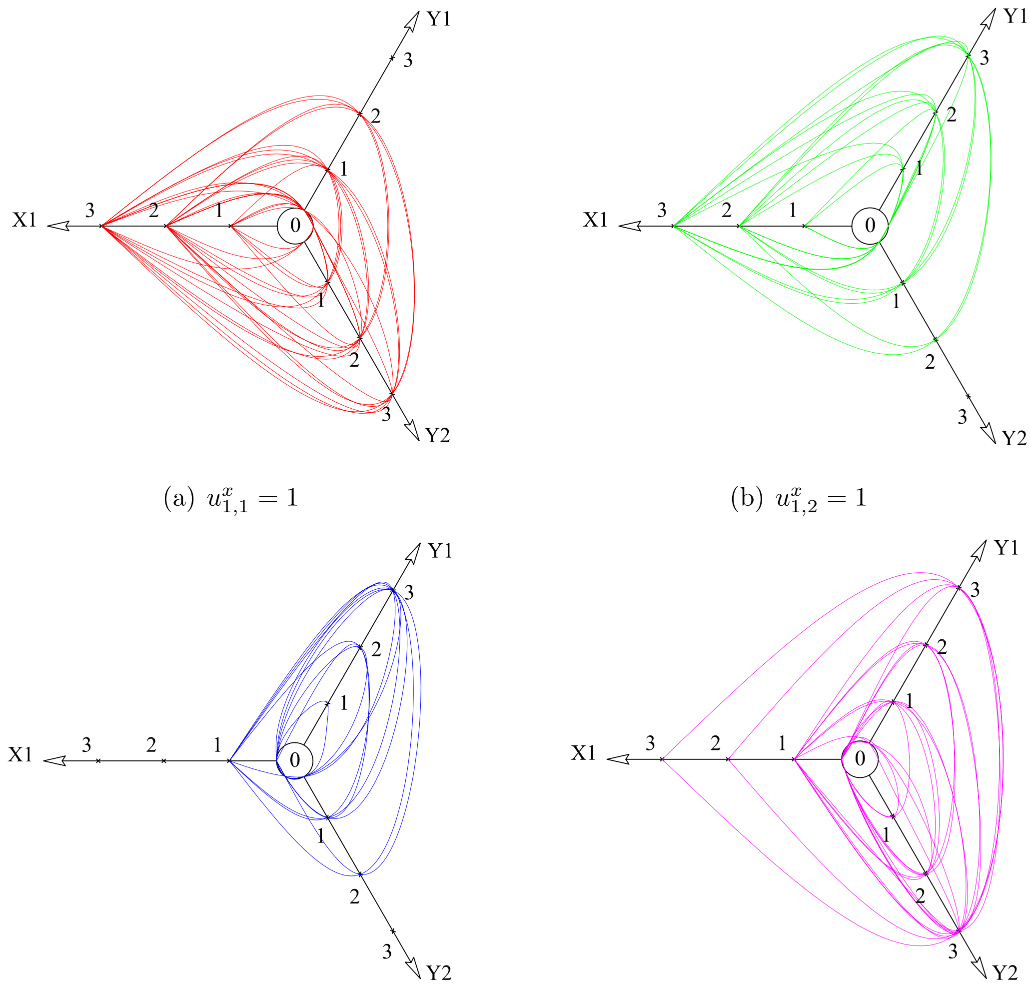


Figure 9: The optimal policy for case 3, $c^x_{1,j} = c^y_j = 1, \forall j$ and $q_1 = 0.5$.

results, we concluded that the optimal policy is a threshold type policy. Furthermore, for homogeneous system the policy shown to be a most balancing one. We developed a method to graphically map the optimal policy from the $L+K+1$-dimensional space to a 2-dimensional space, in order to facilitate the visualization of the optimal policy structure and to deduce conclusions.

The numerical determination of the optimal policy is a high computational complexity task. For systems with large L and K, this complexity can be a prohibiting factor. However, the information regarding the structural characteristics of the optimal policy, that was obtained for the special cases we studied here, can be generalized to larger systems. For practical deployment, the optimal policy can be computed in non-real time and then stored in a lookup table that can be accessed later.

For future work, we propose to design a heuristic algorithm, based on the results we obtained here, that can be used to find the optimal policy for any wireless relay network.



(a) $u^x_{1,1} = 1$

(b) $u^x_{1,2} = 1$

(c) $u^y_1 = 1$

(d) $u^y_2 = 1$

Figure 10: Optimal actions for case 3, $c^x_{1,j} = c^y_j = 1, \forall j$ and $q_1 = 0.5$.

We are also working on the theoretical proof of the results we obtained in this paper.

References

[1] Y. Hong, W. Huang, F. Chiu, and C. C. J. Kuo, Cooperative communications in resource-constrained wireless networks, IEEE Signal Processing Magazine, vol. 24, pp. 4757, May 2007.

[2] A. Sendonaris, E. Erkip and B. Aazhang, "User Cooperation Diversity–Part I: System Description," IEEE Transactions on Communications, vol. 51, no. 11, pp. 1927-1938, 2003.

[3] N. Laneman, D. N. C. Tse and G. W. Wornell, "Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behaviour," IEEE Trans. Inform. Theory, vol. 50, pp: 3062–3080, 2004.

[4] A. Nosratinia, T.E. Hunter, A. Hedayat, " Cooperative Communication in Wireless Networks," IEEE communications Magazine, 2004.

[5] A. Sendonaris, E. Erkip, and B. Aazhang, Increasing uplink capacity via user cooperation diversity, in Proc. of IEEE International Symposium on Information Theory, (Cambridge, USA), Aug. 1998.

[6] V. Stankovic, A. Host-Madsen, and Z. Xiong, Cooperative diversity for wireless ad hoc networks, IEEE Signal Processing Magazine, vol. 23, pp. 3749, Sep. 2006.

[7] "MMR Harmonized Contribution on 802.16j (Mobile Multihop Relay) Usage Models," Document No. IEEE 802.16j-06/015, Sep. 2006.

[8] H. S. Wang and N. Moayeri, "Finite-State Markov Channel–A Useful Model for Radio Communication Channels," IEEE Transactions on Vehicular Technology, vol. 44, pp. 163171, Feb. 1995.

[9] J. P. Castro, *All IP in 3G CDMA Networks*. USA: John wiley & Sons Inc., 2004.

[10] Q. Zhang and S. Kassam, " Finite-state Markov model for Rayleigh fading channels," IEEE Transactions on Communications, vol. 47, no. 11, Nov. 1999.

[11] Z. Roseburg, P. Varaiya, J. Walrand, "Optimal Control of Service in Tandem Queues," IEEE Trans. Auto. Control. AC27:600–610 (1982).

[12] H. Al-Zubaidy, I. Lambadaris and I. Viniotis, "Optimal Resource Scheduling in Wireless Multi-service Systems With Random Channel Connectivity, IEEE Globecom09, Honolulu, HI, USA, Dec. 2009.

[13] Y. Shi, W. Zhang, K. B. Latief, "Cooperative Multiplexing and Scheduling in Wireless Relay Networks," in Proc. IEEE International Conference on Communications (ICC 08), 2008.

[14] C.Y. Hong, A.C. Pang, "Link Scheduling with QoS Guarantee for Wireless Relay Networks," IEEE Conference on Computer Communications (INFOCOM), Rio De Janeiro, Brazil, 2009.

[15] A. Bletsas, H. Shin, and M. Z.Win, Cooperative communications with outage optimal opportunistic relaying, IEEE Trans. on Wireless Commun., vol. 6, pp. 34503460, Sep. 2007.

[16] T. C.-Y. Ng and W. Yu, Joint optimization of relay strategies and resource allocations in cooperative cellular networks, IEEE Journal on Selected Areas in Commun., vol. 25, pp. 328339, Feb. 2007.

[17] V. Sreng, H. Yanikomeroglu, and D. Falconer, Relay selection strategies in cellular networks with peer-to-peer relaying, in IEEE Vehicular Technology Conf. (VTC03-Fall), Oct. 2003, pp. 19491953.

[18] M. Yu and J. Li, Is amplify-and-forward practically better than decode-and- forward or vice versa? in IEEE Inter. Conf. Acoustics, Speech, and Signal Processing, (ICASSP 05), vol. 3, Mar. 2005, pp. 365368.

[19] A. Host-Madsen and J. Zhang, Capacity bounds and power allocation for wireless relay channels, IEEE Trans. Information Theory, vol. 51, no. 6, pp. 20202040, June 2005.

[20] M. Chen, S. Serbetli, and A. Yener, Distributed power allocation for parallel relay networks, IEEE Global Telecommunication Conf. (GLOBECOM 05), vol. 3, Nov. 2005, pp. 11771181.

[21] M. Puterman, *Markov Decision Process: Discrete Stochastic Dynamic Programming.* USA: John wiley & Sons Inc., 1994.

[22] S. M. Ross, *Introduction to Stochastic Dynamic Programming.* USA: Academic Press, 1983.