Throughput-optimal scheduling for cooperative relaying in wireless access networks

Ngoc-Thai Pham, Thong Huynh and Won-Joo Hwang[*]

Abstract

This article considers the cooperation between base station and relay stations to increase system throughput in time-slotted relaying wireless networks, such as dynamic time division multiple access systems. We focus on optimal throughput scheduling policies for the cooperative relaying at the network layer level. It is shown that the resulting policy for this cooperative protocol obtains the optimal throughput region. Random packet queueing at the relay stations may cause a packet-reordering effect, which may be an obstacle for real-time applications. We alter the design for throughput-optimal scheduling to remove this effect and guarantee a near optimal throughput region.

Keywords: scheduling, optimal control, cooperative communication, relaying, access networks

1 Introduction

Cooperative communications that take advantage of the broadcasting nature of wireless environments have shown excellent performance in both theoretical aspects and implementations, e.g., per-node throughput of the cooperative network is a constant factor while per-node throughput of the conventional wireless network decreases when increasing node density [1]. Cooperative communication at the physical layer requires extra computation and synchronization capabilities. This is usually applicable for base station (BS) and relay station (RS) but not for mobile devices due to their limited computational capacity and power resources [2]. However, the cooperation at the network level can provide performance gain, as shown in [3], even without the aforementioned capabilities.

We consider cooperation at network layer between BS and RS in slotted-time wireless access networks, e.g., dynamic time division multiple access or orthogonal frequency-division multiple access. An example of this network is multi-hop relaying IEEE 802.16 WiMax. In this scenario, the RS overhears all packets transmitted from the BS to the subordinate mobile station (MS) and retransmits only the packets that could not be decoded at the MS. This cooperative protocol does not require

any extra capability at the physical layer and is applicable for both downlink and uplink transmission. We refer to it as the cooperative packet relaying (CPR) protocol. We also refer to the direct transmission from the BS to the MS as single-hop protocol and the conventional relaying [2] through the RS as multi-hop protocol. This slotted-time system requires a scheduling policy that dynamically determines a set of active users at a given time regarding available resources and interference constraints between links.

Throughput-optimal scheduling policy that provides the largest region of feasible arrival rates is desirable in wireless networks. It has been shown that the Backpressure policy [4], which makes scheduling decision basing on difference of queue length between source and destination on each link, is throughput-optimal in multi-hop wireless networks. However, in the scenario of the CPR protocol, since the BS sends packets to the RS and MS at the same time, on contract to a single destination in conventional relaying, queue length updating at each time slot are thus different: the queues are updated according to the transmission state of the BS. Furthermore, since the transmission states are known only by acknowledgments from receivers, the scheduling decision must consider random packet loss and packet re-transmission. This necessitates a new design for throughput-optimal scheduling in the CPR protocol.

* Correspondence: ichwang@inje.ac.kr
Department of Information and Communications Engineering, HSV-TRC, Inje University, Gimhae, Gyeongnam, Korea

The packet queueing at the RS is important to achieve the throughput-optimal scheduling. This is because the queueing capability allows a higher degree of freedom in choosing the active links at every time slot. For example, if the channel quality is not good enough to directly forward the received packet, the RS temporarily buffers the packet and uses the available time slot for the waiting users that have better channel quality [5]. However, the packets queuing at the RS introduce a packet re-ordering problem at the MS due to the fact that packets are randomly queued at the RS. A scheduling policy that allows only one packet of a flow to be queued at the RS is necessary to remove this effect.

We focus on designing a throughput-optimal scheduling policy for the CPR protocol and addressing the above challenges. This scheduling policy minimizes sum square of the queue lengths at every time slot so as to equalize and keep the queues stable, thus maximizing the achievable throughput region. The throughput region under the proposed policy for CPR is significantly larger than the throughput region of the conventional relaying protocol and depends less on the link qualities. This is because the CPR protocol takes advantage of every single transmission from the BS. Even a failed transmission from the BS can contribute to the delivery of packets by pushing the packets to the RS. From a practical perspective, this approach is ideally suited for the scheduling in multi-hop relaying system, e.g., IEEE 802.16j WiMax. The primary contributions of this study are as follows.

- We characterize the throughput region of the wireless network under the CPR protocol with packet queueing at the RS. We show that this region is larger than that obtained by the conventional multi-hop relaying protocol and single-hop protocol in wireless networks in Section 3.
- We consider the queueing network as a dynamic system and formulate an optimal control problem for the throughput-optimal scheduling policy. The scheduling policy determines a set of active links at each time slot based only on the current queue length information and channel quality. A theoretical analysis using Lyapunov functions shows that this online policy obtains the maximum throughput region in Section 4.
- We design a scheduling policy that allows only one packet for each flow queue at the RS. This scheme eliminates the packet-reordering effect caused by opportunistic packet queueing. The achievable throughput region of this policy is analytically proved to be near optimal while the simulation results show it outperforms the conventional multi-hop relaying protocol and single-hop protocol in Section 5.

2 Related works

In the cooperative transmission at physical layer, source nodes and relay nodes cooperatively deliver signal to the destination node. It requires additional capabilities to synchronize between source nodes and relay nodes and to combine signals at the destination [6]. The relay nodes may have different strategies when forwarding the signal from the source node: the relay nodes amplify then forward (amplify-and-forward) [7]; the relay nodes decode then forward (decode-and-forward) [7]; or the relay nodes retransmit the signal that they receive from the sources to the destination only when the destination fails to decode the information packet (incremental relaying) [6]. In particular, the research in [6] showed that the incremental relaying protocol provides the best outage behavior among these cooperation protocols as there are occasions that the information packets need to send only once. The behavior of the CPR protocol is similar to that of the incremental relaying protocol except that CPR protocol is performed at network layer. However, the CPR protocol is more flexible than the incremental relaying protocol since it does not require extra capabilities at physical layer.

A number of studies have also exploited broadcast nature of wireless channels at the network layer level. In [8,9], authors investigated the opportunistic routing problems in ad hoc wireless networks. Each transmission from source nodes is broadcast and the routing decision to the destination, or decision of next hop, is decided opportunistically basing on the result of the local broadcast. While the algorithm in [8] bases on queue lengths between nodes to make routing decision, the algorithm in [9] is derived from a finite-state Markovian decision problem. Analysis of throughput region and delay in cooperation at network level is studied in [3,10] under TDMA-based scheduling. To fully exploit the available resource, the unutilized time slots in TDMA scheduling are used by the relay nodes to retransmit the packets to the destination [3]. Although these works use a similar protocol with CPR for cooperation at network layer, the objectives of these studies are different from our research. The studies in [8,9] target the opportunistic routing. The studies in [3,10] focus on analysis of stability region and delay for network-level cooperation protocol under TDMA-based scheduling. The objective of our research is to find a throughput-optimal scheduling on relaying networks.

Research on throughput-optimal scheduling for single-hop and multi-hop wireless networks is addressed in [4,11]. This is to find scheduling policies that support the largest throughput region on a specific network topology. The seminal studies in [4] have shown that Backpressure scheduling policy, which bases on the

difference of queue length between source and destination nodes of links, can achieve the maximum throughput region. The studies in [11] extend the scheduling policy to a general framework for scheduling and flow control in multi-hop wireless networks. Recently, these results have been extended to cooperative networks. In [12], two-hop relaying networks with cooperation at physical layer are studied with an optimal scheduling policy and its stability region. The authors focus on throughput-optimal scheduling in relaying network under decode-and-forward protocol at physical layer. On the contrary, this article focuses on finding an optimal scheduling policy for cooperation at network layer.

3 CPR in wireless networks
3.1 CPR protocol
We consider a single macro cell that is equipped with a single BS, N_R fixed RS, and N_M MS. The RS are installed to increase throughput and coverage. They receive and forward packets in both downlinks and uplinks. Due to the capacity of RS, error probability on links between the BS and the RS are low. For simplicity, we assume that there is no error on this link. In practice, this assumption may not be true; however, it can be remedied using acknowledgements without any difficulty. We also assume that the nearby stations cannot interfere with each other. This is because the RS use low power and their interference are considered in deployment and resource allocation, e.g., using orthogonal sub-carrier assignment in OFDMA [13].

We focus on a typical scenario where all MS associate with BS and another RS so that they can receive packets from both stations. This scenario is practical since current standards usually consider MS associating with only one RS to reduce the control information for cooperation between the RS [2]. The case where one MS associates with more than one RS is outside the scope of this article. Let $r \in N_R$ and $c \in N_M$ represent the index of the RS and MS, respectively. We also use c to refer to the flow associated with the MS c. Let N_r be the set of MS that associate with the RS r.

We assume a slotted time frame structure that is widely used in implementation of practical systems, e.g., WiMAX, and theoretic analyses, e.g., in [11,12]. The packet has a fixed length and the transmission time of one packet is one time slot. The time is divided into equal units $t = \{0, 1, 2, \ldots\}$ called time slots. The time slot in the CPR protocol is divided into three subslots: control subslot, data subslot, and ACK subslot (Figure 1a).

In the downlink mode, at the control subslot, the stations send the pilot signal to measure channels. This channel state information is used in the scheduling algorithm to compute the link activation vectors $\mu(t)$. This computation of scheduling follows the policies explained in the subsequent sections. At the data subslot, the selected links are activated to transmit data packets. In the case where BS is active, the packet is transmitted to the corresponding MS. The associated RS of the receiving MS overhears the transmitting packet. Note that we assume there is no error at the RS. At the ACK subslot, the receiving MS sends an acknowledgment to the BS and RS to inform the state of the packet. If it is successful, the RS and BS remove the packet from the queue. If it fails, the BS removes the packet from the queue; however, the RS keeps the received packet in the queue and waits to transmit it in the next time slots. In the case where the RS are activated, they transmit the queueing packet to the corresponding MS. Note that this protocol does not drop packets when transmission fails.

At every time slot, scheduling policy decides links to be activated. Depending on the network state, e.g., link qualities, activated links can be the links between BS and RS or the links between RS and MS. In the case where the links between a BS and an MS are activated and there are packet of activated MS queued at the RS, packets transmitted from the BS may successfully arrive the MS before the queueing packets at the RS. Packet re-ordering happens in this situation. To remedy this problem, the scheduling algorithm must allow to transmit packets of the MS, which are queueing at the RS, before initiating a new packet transmission to that MS.

In the uplink mode, the scenario is similar with the one of downlink mode. In the data subslot, when MS is active to send data packets to BS, associated RS overhears the packet and transmits the packet to the BS when the BS fails to receive the packet directly from the MS. The error probabilities of re-transmissions are low because links between the RS and the BS are more reliable than links between the MS and the BS. Due to the symmetry of the downlink and uplink mode, we consider only the downlink model in our analysis. The uplink model can be extended without major changes.

3.2 Network model
The proposed system shown in Figure lb is modeled by two layers: a network layer and a link layer. A summary of frequently used notations is provided in Table 1.

At the link layer, let $\mu_{b,\,c}(t)$ and $\mu_{r,\,c}(t)$ indicate the activations of the commodity flow c on link (b, c) and (r, c) at time slot t. If the flow c on the link (b, c) is scheduled, then $\mu_{b,\,c}(t) = 1$; otherwise $\mu_{b,\,c}(t) = 0$. Note that due to the broadcast nature, the link (b, r) is identical to the link (b, c) and hence their activations and data rates are identical. The success probabilities depend on the channel conditions, modulation, coding type, and power level, as shown in [14]. Using the pilot signal, these probabilities can be estimated at the beginning of each time slot. They are denoted by $p_{b,\,c}(t)$ and $p_{r,\,c}(t)$.

Figure 1 Network system model. **(a)** Network node model. **(b)** Time slot structure.
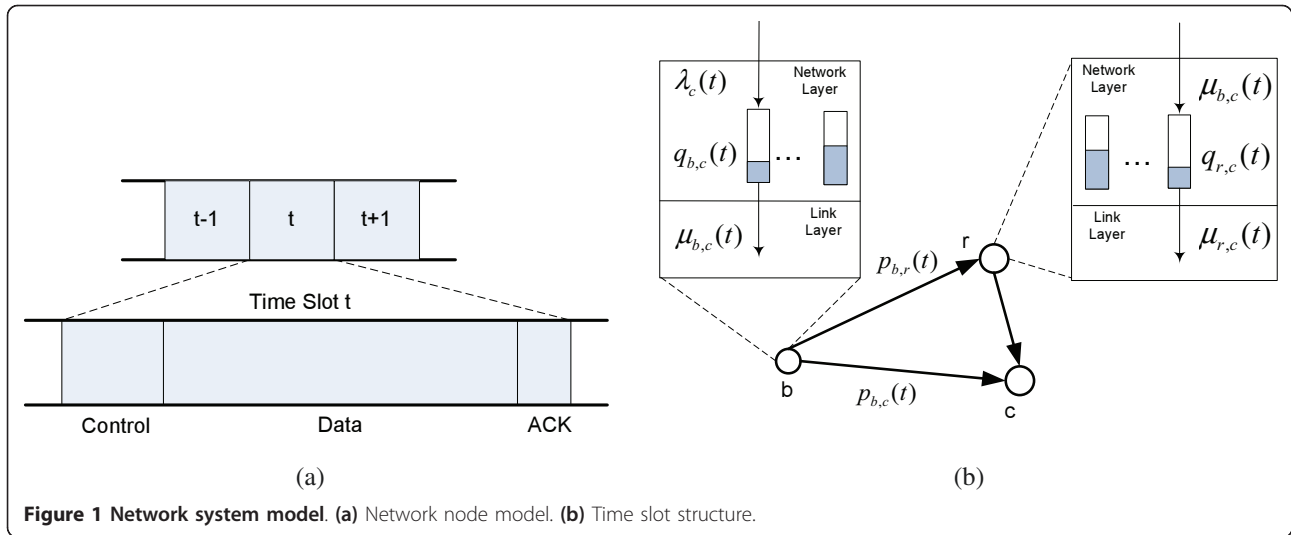
We assume that these probabilities evolve over slots according to a finite-state Markov chain and thus the steady state is measurable and is denoted by $p_{b,\,c}$ and $p_{r,\,c}$. The interference between links imposes a feasible region to activation vector $\mu(t) = \{\mu_{b,c}(t),\ \mu_{r,c}(t)\}_{c\in N_M}$. This feasible region specifies a set of links that can be activated at the same time slot. For example, since the BS covers all the RS and MS, when the BS is active, other stations must be inactive. An element of the feasible set thus should include only one active link from BS and all other inactive links in the system. Let the feasible region of $\mu(t)$ be Ω; then $\mu(t) \in \Omega$.

At the network layer, let $a(t) = \{a_c(t)\}_{c\in N_M}$ represents the exogenous traffic arriving at the time slot t with the mean $\lambda_c = \mathrm{E}\{a_c(t)\}$ and $a_c(t) \le a_{\max}$. The arrival processes are i.i.d over slots and the second moment

E$\{a_c^2(t)\}$ of the arrival process is finite. The network layer uses a queue separation scheme where every flow has a separate queue at each node. Let $q_{b,\,c}(t)$ and $q_{r,\,c}(t)$ be the queue of flow c at the BS and RS. Let $s_{b,\,c}(t)$ and $s_{r,\,c}(t)$ indicate the transmission state of a packet on the corresponding links at the end of the time slot t. For example, $s_{b,\,c}(t) = 1$ if the transmission is successful; otherwise $s_{b,\,c}(t) = 0$. Thus we have the success probabilities $Pr\ [s_{b,\,c}(t) = 1] = p_{b,\,c}(t)$ and $Pr\ [s_{r,\,c}(t) = 1] = p_{r,\,c}(t)$.

Queue length $q_{b,\,c}(t + 1)$ depends on the queue length $q_{b,\,c}(t)$, input packet $a_c(t)$, and output packet $\mu_{b,\,c}(t)$ at time t as shown in (1). In (2), a packet is added to the queue at RS only when $(1 - s_{b,\,c}(t))\,\mu_{b,\,c}(t) = 1$ that means the packet is transmitted, or $\mu_{b,\,c}(t) = 1$, and the MS fails to receive it, or $1 - s_{b,\,c}(t) = 1$. Similarly, packet is removed from the queue only when it is successfully delivered to the MS, that means $s_{r,\,c}(t)\,\mu_{r,\,c}(t) = 1$.

$$q_{b,c}(t + 1) = q_{b,c}(t) - \mu_{b,c}(t) + a_c(t) \tag{1}$$

$$q_{r,c}(t + 1) = q_{r,c}(t) - s_{r,c}(t)\mu_{r,c}(t) \\ + (1 - s_{b,c}(t))\mu_{b,c}(t). \tag{2}$$

3.3 Throughput region of the CPR protocol
We first recall definitions of stability and the throughput region of wireless networks.

Definition 1 (Network stability): A queue q is stable if $\limsup\limits_{T->\infty} \frac{1}{T} \sum_{t=0}^{T-1} E\{q(t)\} < \infty$. A network is stable if all queues are stable.

Definition 2 (Stability region): The stability region of a scheduling policy is the set of arrival rates $\{\lambda_m\}_{m\in\{1,\,\ldots,\,M\}}$ that stabilizes the system under the policy. The

Table 1 Frequently used notations

Notations	Meaning
N_R	Set of RS
N_M	Set of MS
r	Relay station
c	MS
N_r	MS associated to relay r
$r_{\max}$	Relay station with the largest number of MS
$\mu_{b,\,c}(t),\ \mu_{r,\,c}(t)$	Variables that indicate the activation of link (b, c) and (r, c)
$p_{b,\,c}(t),\ p_{r,\,c}(t)$	Success probability on link (b, c) and (r, c)
$p_{b,\,c},\ p_{r,\,c}$	Averate success probability on link (b, c) and (r, c)
Ω	Feasible region of active links
$a_c(t)$	Arrival rate of c at time t
λ_c	Arrival rate of c
$q_{b,\,c}(t),\ q_{r,\,c}(t)$	Queue size of MS c at BS and RS
$s_{b,\,c}(t),\ s_{r,\,c}(t)$	Transmission state of packet of the MS c at the end of time slot t

union of stability regions of all scheduling policies is the network stability region.

Definition 3 (Throughput region): The throughput region, denoted by Λ, is a set of all arrival rate vectors that can be stably supported by the network under the assumption that network queues are always saturated.

It is clear that on the boundary of the network stability region, the queues are always saturated, and thus the network stability region and throughput region are identical. Definitions of stability region and throughput region are important to show the throughput-optimal property of a policy. A policy is throughput-optimal if it can be shown that it stabilizes all arrival rates that belong to the throughput region.

The establishment of the throughput region in the conventional wireless network has been comprehensively studied in a number of researches through the flow conservation constraints and the link capacity constraints, e.g., [11,15]. We establish the throughput region basing on flow conservation and link activation time constraints. Let $l_{b,\,c}(t)$ and $l_{r,\,c}(t)$ are the average data rate over links (b, c) and (r, c), respectively. The total portion of time for each link must be smaller than one. The rate vector $\lambda \in \Lambda$ if there exists a vector $\beta = \{\beta_1, \ldots, \beta_{N_R}, \beta_R\}$ such that:

$$\beta_c l_{b,c} = \lambda_c \ \forall c \ \in \ N_M \tag{3}$$

$$\gamma_{r,c} p_{r,c} l_{r,c} = (1 - p_{b,c}) \lambda_c \tag{4}$$

$$\beta_r = \sum_{c \in M_r} \gamma_{r,c} \ \forall r \ \in \ N_R \tag{5}$$

$$\beta_R = \max_{r = \in N_R} \{\beta_r\} \tag{6}$$

$$1 \geq \sum_{c \in N_M} \beta_c + \beta_R \tag{7}$$

The flow conservation constraints specify that input flow and output flow at each node, BS and RS, must be equal. Equations (3) and (4) follow from the link conservation constraints at the BS and RS. The vector $\gamma_r = \{\gamma_1, \ldots, \gamma_{M_r}\}$ represents the portion of activation time of users associated to r and β_r represents the total portion of activation time on each RS. Since RS do not interfere with each other, links in different RS can activate at the same time. Thus, the total portion of activation time for RS is the most congested station. This is represented in equations (5) and (6). Finally, total portion of activation time is not greater than 1. This constraint is formulated in equation (7).

The CPR protocol provides better throughput region than that of the conventional protocols while it does not require extra complexity at the physical layer as compared to the conventional cooperative protocol. This can be explained intuitively as follows. In CPR protocol, packets are re-transmitted from the RS meanwhile packets are re-transmitted from the BS in single-hop protocol. Due to the higher quality of the link between RS and MS, CPR protocol is thus more efficient than the single-hop protocol. In addition, in CPR protocol, once packets are failed to be delivered using direct transmission from BS, which takes only one time slot, they are forwarded throughput the RS. This forwarding actually is the multi-hop relay protocol which costs two time slots to forward one packet. In other words, multi-hop relay protocol is less efficient than CPR protocol. The advantage of the CPR protocol is formally shown in Theorem 1.

Theorem 1. *The throughput region of the CPR protocol is larger than the throughput region of the multi-hop relay protocol and single-hop protocol.*

Proof. See Appendix A.

4 Throughput-optimal scheduling for the CPR protocol (TOSC)

4.1 TOSC modeling

In order to find a policy that stabilizes the queueing system whenever the arrival rates are within the throughput region, we find a policy that minimizes the total queue length average. To do this, we construct the scheduling problem as an optimal control problem. This framework can be applied to construct the scheduling policies in general cases where there are no constraints on the policy, e.g., queue length constraints.

We consider a discrete-time linear time-invariant system with dynamics in equations (1) and (2), where $\Phi(t) = \{q_{b,c}(t), q_{r,c}(t)\}_{c \in N_M} \geq 0$ are the states and $\mu(t) \in \Omega$ is the control input. Note that only $\mu(t)$ is variable here, since the scheduling decision is considered in this design. The design goal is to serve as much as possible the arrival traffic to MS. Since these traffic loads directly affect physical queue lengths, minimizing queue lengths helps to support as much as traffic load. We define the cost function as sum square of queue lengths. The cost function in form of sum square of queue lengths pushes the scheduling decision to equalize the queue length and to prevent queue lengths from infinity. Thus it stabilizes the network. The optimal control problem minimize the cost function defined as follows:

$$J = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T} E \left\{ \sum_{c \in N_M} \left(q_{b,c}(t)^2 + q_{r,c}(t)^2 \right) \right\}$$

4.2 TOSC policy

In general, the linear stochastic optimal control problem can be solved effectively using the optimal policy for only few special cases [16,17]. However, there exist different methods to find *suboptimal control policy* [16]. Regarding the computational complexity and its compliance to the objective function, we consider Control-Lyapunov feedback method. The suboptimal control policy of Control-Lyapunov feedback method is given by:

$$\mu(t) = \arg\min_{\mu \in \Omega} \left(\sum_{m=1}^{M} (q_{b,c}(t+1)^2 + q_{r,c}(t+1)^2) \right) \quad (8)$$

Proposition 1: *The following inequality holds at every time slot t*

$$\sum_{c \in N_M} (q_{b,c}(t+1)^2 + q_{r,c}(t+1)^2) \leq$$
$$\sum_{c \in N_M} (q_{b,c}(t)^2 + q_{r,c}(t)^2) + B + 2\sum_{c \in N_M} q_{b,c}(t)a_c(t)$$
$$- 2\underbrace{\sum_{c \in N_M} (\mu_{b,c}(t)(q_{b,c}(t) - (1 - s_{b,c}(t))q_{r,c}(t)) - \mu_{r,c}(t)s_{r,c}(t)q_{r,c}(t))}_{Scheduling} \quad (9)$$

where

$$B = \sum_{c \in N_M} \left(s_{r,c}(t)^2 \mu_{b,c}(t)^2 + (1 - s_{b,c}(t))^2 \mu_{r,c}(t)^2 \right)$$
$$+ \mu_{b,r}(t)^2 + a_c(t)^2.$$

Proof. Squaring both sides of equations (1) and (2), summing over all *c* entries, and re-arranging yields (9). □

A suboptimal policy can be derived from equation (8) by minimizing the expectation of the right side of (9). Note that $E\{s_{b,c}(t)\} = p_{b,c}(t)$ and $E\{s_{r,c}(t)\} = p_{r,c}(t)$. Thus, the scheduling policy, namely TOSC, is defined as follows.

The TOSC policy: *At each time slot, TOSC selects an activation vector μ(t) such that*:

$$\mu(t) = \arg\max_{\mu \in \Omega} \sum_{c \in N_M} (\mu_{b,c}(t)w_{b,c}(t) + \mu_{r,c}(t)w_{r,c}(t)) \quad (10)$$

where

$$w_{b,c}(t) = q_{b,c}(t) - (1 - p_{b,c}(t))q_{r,c}(t)$$
$$w_{r,c}(t) = p_{r,c}(t)q_{r,c}(t).$$

The TOSC policy has a similar form with the well-known Backpressure algorithm [11] however, the weight of links, $w_{b,c}(t)$ and $w_{r,c}(t)$, are calculated differently. In the TOSC, the weights take into account the success probability on each link and the cooperation on re-transmission of the failed packet $(1 - p_{b,c}(t))$. The decision of the TOSC

depends only on the current queue length state $q(t)$ and success probability of links. Theorem 2 shows that the TOSC is throughput-optimal since it achieves the throughput region Λ defined in Section 3. For any arrival rate $\lambda \in \Lambda$, the TOSC policy performs better than the stationary randomized policy, as shown in [11], in stabilizing the queueing system $q(t)$.

Theorem 2. *If $\overrightarrow{\lambda} \in \Lambda$ then the queueing system is stable under TOSC.*

Proof. See Appendix B.

4.3 Complexity analysis

According to special structure of interference between links, TOSC policy must choose activation vectors from only the following two possibilities: BS is activated and other RS are muted; or BS is muted and RS are activated. Furthermore, in a RS, only one mobile state is activated at a time slot. TOSC policy thus selects a MS with maximum weight $w_{r,c}(t)$ in each RS, and compare the sum of the weights from the RS with weights $w_{b,c}(t)$ at the BS to activate BS or RS. Scheduling decision (11) can be written as follows:

$$\mu(t) = \arg\max_{\mu \in \Omega} \left\{ \{\mu_{b,c}(t)w_{b,c}(t)\}_{c \in N_M}, \sum_{r \in N_r} \max_{c \in N_r} \{\mu_{r,c}(t)w_{r,c}(t)\} \right\} \quad (11)$$

Implementation of TOSC policy requires two steps: gathering network state information, which includes queue length and channel condition; and computing the optimal activation vector. Messages incurred when gathering network state information are called messages passing overhead. The complexity to calculate the activation vector at the BS and the RS is complexity of TOSC.

Messages passing overhead

Link states $p_{b,c}(t)$ and $p_{r,c}(t)$ can be estimated from the previous transmissions. However, since the channel state varies rapidly over time slots, this method cannot provide a good estimation. In order to obtain a better measurement, the BS and the RS send a short pilot signal and the other stations send back acknowledgments that contain the link state. The total messages required to finish this measurement are: N_M messages from the MS to BS, N_M messages to the RS, and N_R messages from RS to BS. Note that for simplicity, we use name of set to refer to the size of the set, e.g., N_M is also used as the number of the MS. Thus the total number of message is $2N_M + N_R$ and the message passing overhead is $O(N_M)$.

The overhead of the scheduling policy for the single-hop protocol, e.g., In the modified largest queue length first [18], to accomplish the acknowledgement of all MS, requires $O(N_M)$ messages. This overhead has the same order with the overhead in TOSC. In addition, it is

important to note that since these messages are sent to the BS through control channel, they does not directly affect the throughput performance of the networks.

Computational complexity

According to (11), under the assumption of using a binary search, $O(\log(N_M + 1))$ is required to compute the maximum link weights $w_{b,c}(t)$ at the BS and $O(N_R \log(N_{rmax}))$ to find the largest link weight $w_{r,c}(t)$ at N_R RS. Since $N_{rmax} \le N_M$ and N_R is a small constant, computational complexity of the TOSC is thus $O(\log(N_M))$. In single-hop protocol, to select one MS out of N_M MS, it requires $O(\log(N_M))$ using binary search. Thus complexity of TOSC is the same with complexity of single-hop protocol.

5 Throughput-optimal scheduling for CPR without packet re-ordering (TOSC-wo)

With the TOSC, in order to obtain the optimal throughput region, the scheduling decision maximizes the weighted sum of queue lengths and continues to transmit new packets regardless of whether there are old packets queueing at the RS. Multiple packets queueing at the RS causes a packet re-ordering problem and may introduce a significant instant delay at the MS. This effect may be harmful for the real-time applications, e.g., video streaming. We design a scheduling policy that allows only one packet of each flow to be buffered at the RS. This policy guarantees that there is no effect of packet re-ordering. The complexity of the policy is identical to that of TOSC due to the similarity in computation of the link weights. We refer to this policy as TOSC-wo.

Since the scheduling decision is based on the weights corresponding to the links (b, c) and (r, c), we construct the weights such that the scheduling policy declines to activate the link on which the destination queue is already occupied. Specifically, the weights $w_{b,c}(t) \le 0$ if the queue $q_{r,c}(t) \ge 1$. This property of the policy is proved in Lemma 1.

The TOSC-wo Policy: *At each time slot, the policy selects an activation vector $\mu(t)$ such that*:

$$\mu(t) = \arg\max_{\mu \in \Omega} \sum_{c \in N_M} (\mu_{b,c}(t)w_{b,c}(t) + \mu_{r,c}(t)w_{r,c}(t))$$

where $p_{max} = max\{p_{b,c}(t)\}_{t=1, 2, \ldots}$ and

$$w_{b,c}(t) = q_{b,c}(t)\left((1-p_{max}) - (1-p_{b,c}(t))\,q_{r,c}(t)\right)$$
$$w_{r,c}(t) = q_{b,c}(t)p_{r,c}(t)q_{r,c}(t).$$

Lemma 1. *The TOSC-wo allows no more than one packet of each flow to be buffered at the RS.*

Proof. Consider the case where $q_{r,c}(t) = 1$, and we have $w_{b,c}(t) = q_{b,c}(t)(p_{b,c}(t) - p_{max}) \le 0$. In this case,

the link (b, c) is eliminated from the activation and links at the RS are chosen to activate since $w_{r,c}(t) \ge 0$ for every time slot. Thus, there are no more input packets to the queue $q_{r,c}$ if $q_{r,c}(t) = 1$. Or, in other words, $q_{r,c}(t) \le 1$ for all t. □

By limiting the queue length $q_{r,c}(t) \le 1$, there is no effect of packet-reordering and the incurred instant delay can be reduced. However, since the queue length conditions do not allow the scheduling decision to take fully the advantage of the link condition, the feasible region of the activation vector $\mu(t)$ is reduced. Consequently, the throughput region is also reduced. We show in Theorem 3 that the TOSC wo guarantee a specific region on the throughput region Λ.

Theorem 3. *The TOSC-wo guarantees the throughput region for arrival rate λ such that $\lambda + \varepsilon \in \Lambda$, where*
$$\varepsilon > \frac{1}{(1-p_{max})} \sum_{c \in N_M} (1 - p_{b,c})\,(\lambda_c - \lambda_c^2).$$
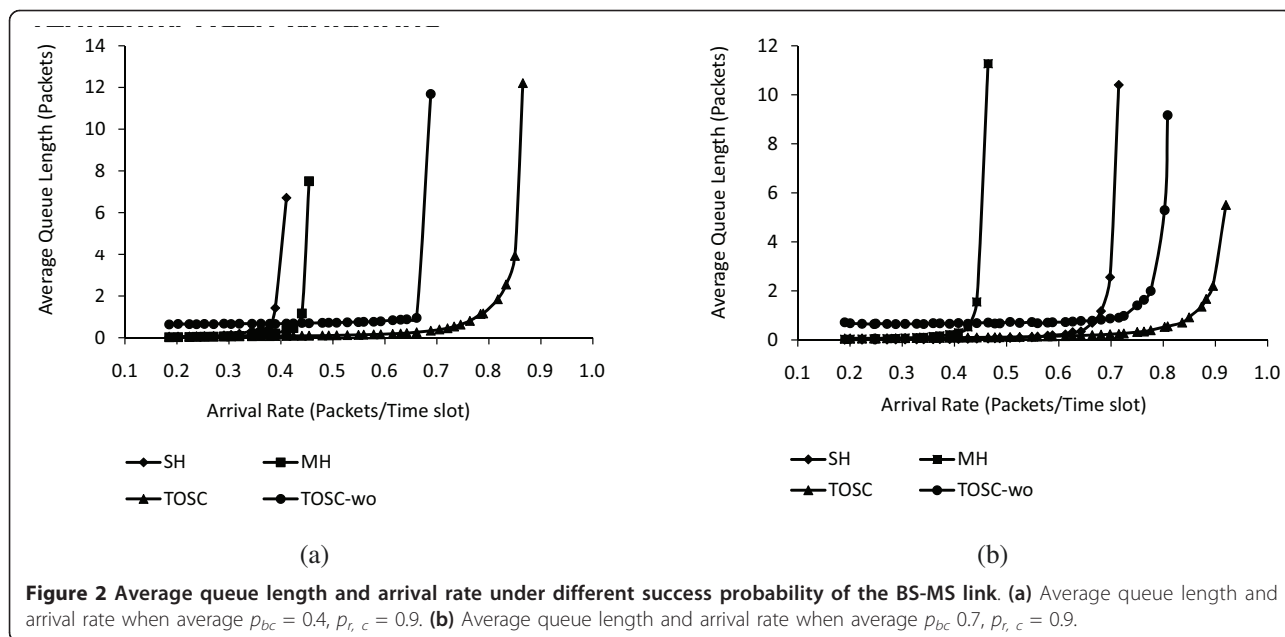
Proof. See Appendix C.

6 Performance evaluation

In this section, we present our simulations to illustrate the theoretical results and compare the proposed policies with the aforementioned well-known protocols.

We set up a wireless access point that includes a BS, RS located on the border of the cell, and randomly placed MS. We consider only the MS that can connect to the RS and omit the others. The link qualities and the arrival rates are stationary processes. At each time slot, the simulation runs three steps that correspond to the three sub-slots in a time slot. At the first step, the link success probabilities are generated using a stationary random process with a given mean. The extemal packet arrival rate at each node is i.i.d. with mean λ_c. At the second step, we find the activation vector according to the scheduling policy based on the interference model, the physical queue $q(t)$, and the instant success probability $p_{b,c}(t)$ and $p_{r,c}(t)$. At the last step, the queueing evolution is updated according to the generated transmission states $s_{b,c}(t)$ and $s_{r,c}(t)$ using the probabilities $p_{b,c}(t)$ and $p_{r,c}(t)$, respectively. We use the modified largest queue length first [18] as the scheduling policy for the multi-hop relaying protocol (MH) and single-hop protocol (SH). No packet is discarded during the physical queueing update.

In the first scenario, we show the average queue lengths and throughput regions in random topologies. In order to make the average queue lengths comparable between protocols, we adjust the queue length in the TOSC and TOSC-wo using the principles of Little's theorem. Since only a portion of $(1 - p_{b,c})$ of flow c traverses through the RS, the total queue length is $q = \sum_{c \in N_M} q_{b,c} + \frac{q_{r,c}}{1-p_{b,c}}$. Figure 2a shows the throughput region under the scenario where the link quality

Figure 2 Average queue length and arrival rate under different success probability of the BS-MS link. **(a)** Average queue length and arrival rate when average $p_{bc} = 0.4$, $p_{r, c} = 0.9$. **(b)** Average queue length and arrival rate when average p_{bc} 0.7, $p_{r, c} = 0.9$.

between BS and MS have very poor quality, $p_{b, c} = 0.4$, and the links between the RS and MS have very good quality, $p_{b, c} = 0.9$. Although the multi-hop relaying protocol provides better performance than the single-hop protocol in this case, since it does not use the Bs-to-MS link, its performance is far below that of the TOSC and TOSC-wo. Similar results are shown in Figure 2b in the case where link qualities between BS and MS are higher at $p_{b, c} = 0.7$: TOSC and TOSC-wo obtain the throughput region more than 0.8 (packets/time slot) while the multi-hop relay protocol remains below 0.5 (packets/time slot) and the single-hop protocol is 0.7 (packets/time slot). In particular, the queue lengths are very similar between the protocols, and thus this gain of the throughput region does not sacrifice any delay performance. The reason behind this result is that the TOSC and TOSC-wo have wider degrees of freedom in choosing active links at every time slot.

Since the performance of protocols largely depends on the link quality, in the second scenario, we investigate the throughput region of the protocols under variance of the link qualities. We vary the average success probabilities and evaluate the throughput region of each protocol. We measure the throughput region by measuring the maximum average arrival rate of the protocol such that the average queue length is smaller than 100. Figure 3a shows the throughput regions under variance of the average $p_{b, c}$ and $p_{r, c} = 0.9$. Figure 3b shows the throughput regions under the variance of average $p_{r, c}$ and $p_{b, c} = 0.5$. These results confirm the typical scenarios in wireless networks: the performance of the multi-hop and single-hop protocol depends on the link qualities. When the BS-MS link
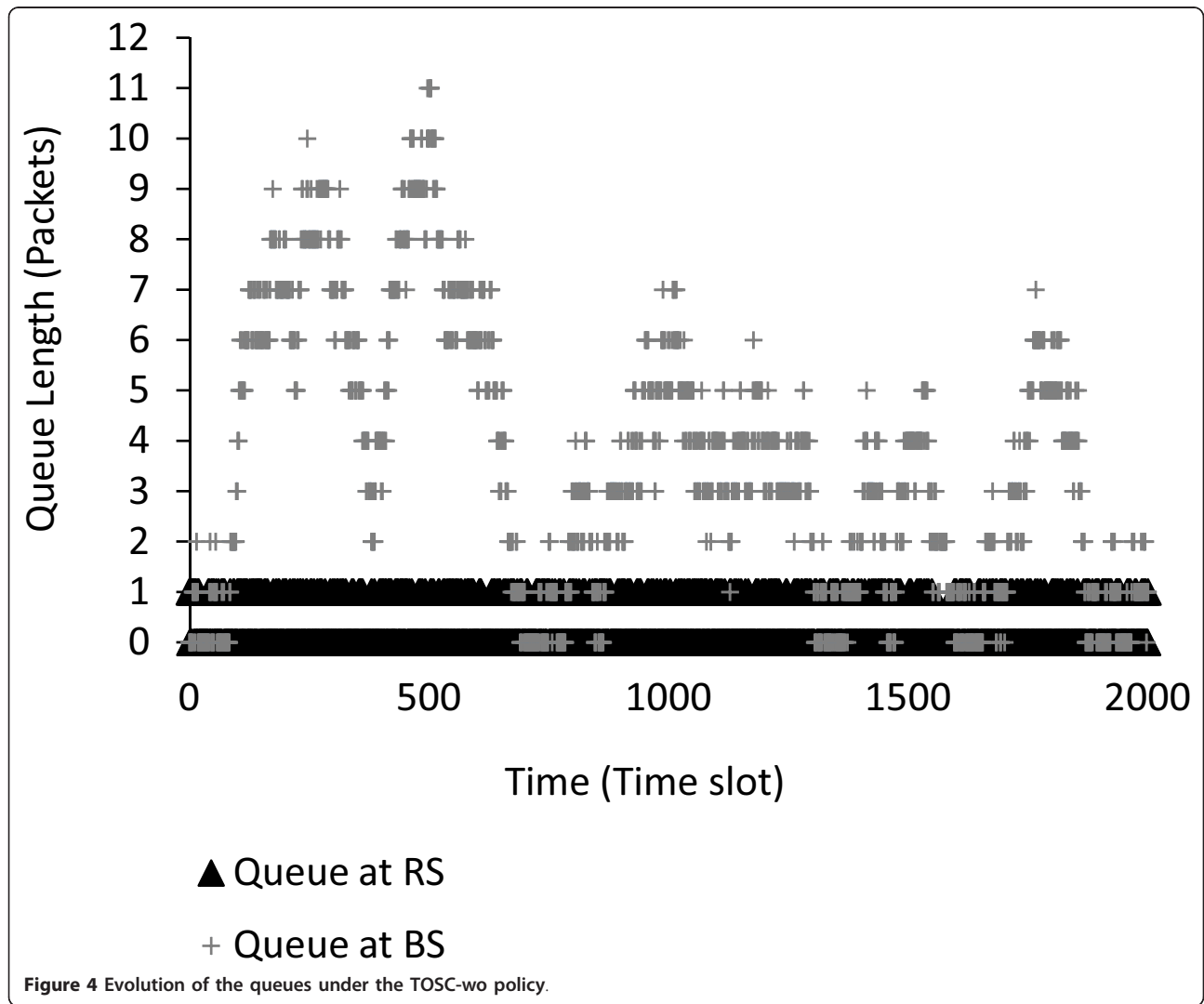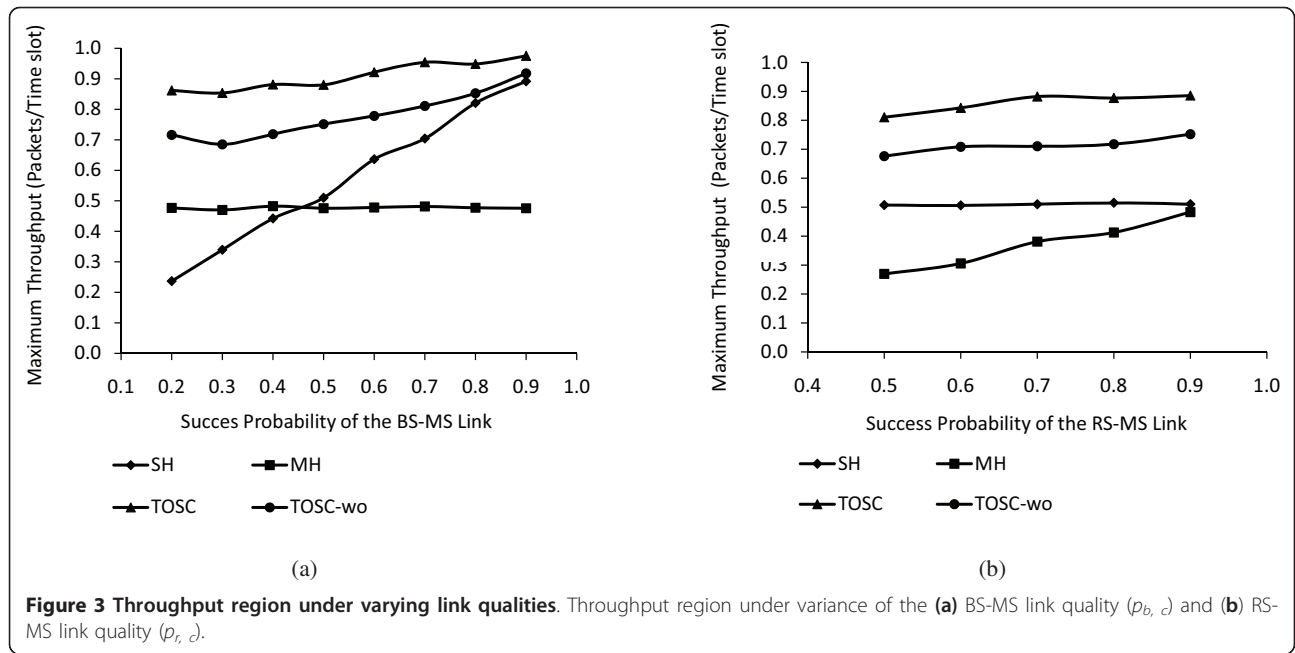
quality $p_{b, c}$ is low, the multi-hop relaying protocol performs better than the single-hop protocol. When the BS-MS link quality is high, the single-hop protocol performs better. However, in all cases, we observe again that the TOSC and TOSC-wo support a significantly larger throughput region in comparison with the two conventional protocols.

It is important to note that the performance of the TOSC and TOSC-wo slightly depends on the link qualities $p_{b, c}$ and $p_{r, c}$. This reflects the nature of the CPR protocol: the failed transmission can contribute to the delivery of packets. These results also confirm the computation of the activation time of the CPR protocol (see Appendix A, (12)): the success probability impacts only part of the throughput region.

Finally, we evaluate the operation of the TOSC-wo by running a simulation on a network of two RS and two MS and investigate the queue length. This investigation shows the queue length property of TOSC-wo policy, which is specified in Lemma 1, and verifies that packet-reordering problem is removed in TOSC-wo. The evolution of the queue lengths at one RS is shown in Figure 4. Queue length of the RS is deterministically less than or equal to 1, as shown in Lemma 1. This property confirms that no packet is transmitted from the BS when there is packet queueing at the RS. The packet-reordering problem thus does not appear in TOSC-wo.

7 Conclusion

In this article, we investigate the CPR protocol on relaying networks. The analytical results and performance evaluation show that this protocol outperforms the

Figure 3 Throughput region under varying link qualities. Throughput region under variance of the **(a)** BS-MS link quality ($p_{b,c}$) and **(b)** RS-MS link quality ($p_{r,c}$).

Figure 4 Evolution of the queues under the TOSC-wo policy.

conventional protocol without requiring extra capability in contrast with the cooperative relaying at physical layer. The results of this research can be easily extended to uplink models, network utility maximization, and power control without difficulties.

A Proof of Theorem 1

To compare the throughput region among the CPR protocol, multi-hop relaying protocol, and single-hop protocol, we first make the following assumptions, which is derived from the network model in Section 3.1, for links between stations: $l_{b,r} = l_0 \; \forall r \in N_R$; $p_{b,r} = 1$, links between MR and MS: $l_{r,c} = l_0 \; \forall r \in N_R, c \in N_M$; $p_{r,c} = p_0$, the link between BS and MS: $l_{b,c} = l_0 \forall c \in N_M$; $p_{b,c} = p_1$.

The total portion of activation time required for the arrival rate λ, in the single-hop protocol, multi-hop relaying protocol, and the CPR protocol, which are denoted as τ_{sh}, τ_{mh}, and τ_{CPR}, respectively, is given by:

$$\tau_{sh} = \sum_{c \in N_M} \frac{\lambda_c}{p_1 l_0}$$

$$\tau_{mh} = \sum_{c \in N_M} \left(\frac{\lambda_c}{l_0} + \frac{\lambda_c}{p_0 l_0} \right) \quad (12)$$

$$\tau_{CPR} = \sum_{c \in N_M} \frac{\lambda_c}{l_0} + \max_{r \in N_r} \left\{ \sum_{c \in N_r} \frac{(1-p_1)\lambda_c}{p_0 l_0} \right\}$$

Considering the most congested RS r_{max} and its associated MSs, we have

$$\max_{r=1,...,N_r} \left\{ \sum_{c \in N_r} \frac{(1-p_1)\lambda_c}{p_0 l_0} \right\} = \sum_{c \in N_{r\,max}} \frac{(1-p_1)\lambda_c}{p_0 l_0} \quad (13)$$

Using (13) and noting that $p_1 \leq 1$ and $p_1 \leq p_0$ since the MS is closer the MR than the BS, we have:

$$\tau_{CPR} - \tau_{sh}$$
$$= \sum_{c \in N_M; c \notin N_{r\,max}} \frac{\lambda_c}{l_0} \frac{(p_1-1)}{p_1} + \sum_{c \in N_{r\,max}} \frac{\lambda_c}{l_0} \left(\frac{p_1-1}{p_1} + \frac{1-p_1}{p_0} \right)$$
$$= \sum_{c \in N_M; c \notin N_{r\,max}} \frac{\lambda_c}{l_0} \frac{(p_1-1)}{p_1} + \sum_{c \in N_{r\,max}} \frac{\lambda_c (p_1-1)}{l_0} \left(\frac{1}{p_1} - \frac{1}{p_0} \right)$$
$$\leq 0.$$

and

$$\tau_{CPR} - \tau_{msh} = \sum_{c \in N_{r\,max}} \frac{(1-p_1)\lambda_c}{p_0 l_0} - \sum_{c \in N_M} \frac{\lambda_c}{p_0 l_0}$$
$$\leq \sum_{c \in N_{r\,max}} \frac{\lambda_c}{p_0 l_0} \sum_{c \in N_M} \frac{\lambda_c}{p_0 l_0}$$
$$\leq 0$$

Thus, we have $\tau_{CPR} \leq \tau_{sh}$ and $\tau_{CPR} \leq \tau_{mh}$. This means that the total portion of activation time required to support λ is smallest among the three protocols. In other words, CPR has the largest throughput region among the three protocols. The theorem is proven.

B Proof of Theorem 2

In order to show that TOSC stabilizes all the queues when $\lambda \in \Lambda$, we develop the proof in following steps. First, we define a Lyapunov function, which is sum quadratic of the queue lengths. Second, we show that the drift of the Lyaponov function at every time slot is bounded and the TOSC policy does better than a randomized algorithm in minimizing this drift in every time slot. Consequently, this bound thus holds for the TOSC policy. Finally, we prove that TOSC policy guarantees sum of queue length bounded. Details of the proof are as follows:

Consider a Lyapunov function of the system with dynamic evolution (1) and (2):

$$L(q(t)) = \sum_{c \in N_M} (q_{b,c}(t)^2 + q_{r,c}(t)^2).$$

Value of this Lyaponov function generally can be understood as an index to represent the queue lengths. A drift is defined as a change of the Lyapunov function over each time slots. The drift of the Lyapunov function is calculated as $\Delta L(q(t)) = \sum_{c \in N_M} ((q_{b,c}(t+1)^2 + q_{r,c}(t+1)^2) - (q_{b,c}(t)^2 + q_{r,c}(t)^2))$. From (9) we have:

$$\Delta L(q(t)) \leq B + 2 \sum_{c \in N_M} q_{b,c}(t) a_c(t)$$
$$- 2 \sum_{c \in N_M} \mu_{b,c}(t) \left(q_{b,c}(t) - (1-s_{b,c}(t)) q_{r,c}(t) \right) \quad (14)$$
$$- 2 \sum_{c \in N_M} \mu_{r,c}(t) s_{r,c}(t) q_{r,c}(t).$$

Rearranging and taking the expectation yields

$$E\{\Delta L(q(t))|q(t)\} \leq \overline{B}$$
$$- 2 \sum_{c \in N_M} q_{b,c}(t) \left(E\{\mu_{b,c}(t)|q(t)\} - \lambda_c(t) \right) \quad (15)$$
$$- 2 \sum_{c \in N_M} q_{r,c}(t) \left(E\{s_{r,c}(t)\mu_{r,c}(t) - (1-s_{b,c}(t))\mu_{b,c}(t)|q(t)\} \right)$$

where $\overline{B} = E\{B\}$. Since terms in B are bounded, $\overline{B}$ is bounded.

It is known in [11] that there exists a stationary randomized algorithm that makes decision on $\mu(t)$ at every time slot, so that for $\varepsilon > 0$ and $\lambda + \varepsilon \in \Lambda$, we have $E\{s_{r,c}(t) \mu_{r,c}(t) - (1 - s_{b,c}(t)) \mu_{b,c}(t) \mid q(t)\} = \varepsilon$ and $E\{\mu_{b,c}(t) - \lambda_c \mid q(t)\} = \varepsilon$ for all $c \in N_M$. Using the above results in (15) yields

$$E\{\Delta L(q(t))|q(t)\} \leq \overline{B} - 2\varepsilon \sum_{c \in N_M} (q_{b,c}(t) + q_{r,c}(t)) \quad (16)$$

Note that TOSC minimizes the left side of (9) over all possible choices of $\mu(t) \in \Omega$ at every time slot that includes the scheduling of the randomized scheduling algorithm. Thus, (16) also holds for the TOSC policy.

Consider the system over a given period, let it be M time slots. Summing (9) over M time slots with $t = 0, 1, 2, ..., M - 1$ and note that $E\{L(q(0))\} = 0$ and $E\{L(q(M))\} \geq 0$ we obtain

$$\frac{1}{M} \sum_{t=1,...,M-1} \sum_{c \in N_M} (q_{b,c}(t) + q_{r,c}(t)) \leq \frac{\overline{B}}{\varepsilon}. \quad (17)$$

Taking the limit $M \to \infty$ and due to the fact that $\overline{B}$ is bounded, the queueing system is bounded under TOSC. Theorem 2 holds.

C Proof of Theorem 3

Consider the following Lyaponov function

$$L(q(t+1)) = \sum_{c \in N_M} (1 - p_{max}) q_{b,c}(t+1)^2$$
$$+ \sum_{c \in N_M} q_{b,c}(t+1) q_{r,c}(t+1)^2. \quad (18)$$

Squaring both side of (1) and (2) yields

$$q_{b,c}(t+1)^2 \leq q_{b,c}(t)^2 + \mu_{b,c}(t)^2 + a_c(t)^2$$
$$- 2q_{b,c}(t)(\mu_{b,c}(t) - a_c(t))$$

$$q_{r,c}(t+1)^2 \leq q_{r,c}(t)^2$$
$$+ s_{r,c}(t)^2 \mu_{r,c}(t)^2 + (1 - s_{b,c}(t))^2 \mu_{b,c}(t)^2$$
$$- 2s_{r,c}(t)\mu_{r,c}(t)(1 - s_{b,c}(t))\mu_{b,c}(t)$$
$$- 2q_{r,c}(t)(s_{r,c}(t)\mu_{r,c}(t) - (1 - s_{b,c}(t))\mu_{b,c}(t))$$

and

$$q_{b,c}(t+1) \leq q_{b,c}(t) + a_{max}.$$

Thus, summing over all the flows c and using the fact that $q_{r,c}(t+1)^2 \leq 1$ yields

$$L(q(t+1)) \leq L(q(t)) + B_1 + B_2 \sum_{c \in N_M} q_{b,c}(t)$$
$$- 2 \sum_{c \in N_M} (1 - p_{max}) q_{b,c}(t)(\mu_{b,c}(t) - a_c(t)) \quad (19)$$
$$- 2 \sum_{c \in N_M} q_{b,c}(t) q_{r,c}(t)(s_{r,c}(t)\mu_{r,c}(t) - (1 - s_{b,c}(t))\mu_{b,c}(t))$$

where

$$B_1 = \sum_{c \in N_M} (1 - p_{max})(\mu_{b,c}(t)^2 + a_c(t)^2) + N_M a_{max}$$

$$B_2 = \sum_{c \in N_M} s_{r,c}(t)^2 \mu_{r,c}(t)^2 + (1 - s_{b,c}(t))^2 \mu_{b,c}(t)^2$$
$$- 2 \sum_{c \in N_M} s_{r,c}(t)\mu_{r,c}(t)(1 - s_{b,c}(t))\mu_{b,c}(t).$$

This is equivalent to

$$L(q(t+1)) \leq L(q(t))$$
$$+ B_1 + B_2 \sum_{c \in N_M} q_{b,c}(t) + 2(1 - p_{max}) \sum_{c \in N_M} q_{b,c}(t)a_c(t)$$
$$- 2 \sum_{c \in N_M} \mu_{b,c}(t)q_{b,c}(t)((1 - p_{max}) - (1 - s_{b,c}(t))q_{r,c}(t)) \quad (20)$$
$$- 2 \sum_{c \in N_M} \mu_{r,c}(t)q_{b,c}(t)q_{r,c}(t)$$

Assuming that $\lambda \in \Lambda$, thus there is $\varepsilon > 0$ such that $\lambda + \varepsilon \in \Lambda$. Using the randomized stationary algorithm, as in the proof of Theorem 2, and taking the expectation of (20) yields

$$E(\Delta L(q(t+1))) \leq \overline{B}_1 - \beta q_{b,c}(t) \quad (21)$$

where $\beta = \overline{B}_2 - 2\varepsilon(1 - p_{max}) - q_{r,c}(t)$, $E\{B_1\} = \overline{B}_1$ and $E\{B_2\} = \overline{B}_2$. Note that the TOSC-wo minimizes the left side of (20) over all possible $\mu(t)$ including the solution of the randomized algorithm. Thus, (21) holds for TOSC-wo.

Besides, since $s_{r,c}(t) = \{0, 1\}$ and $\mu_{r,c}(t) = \{0, 1\}$, we have $E\{s_{r,c}(t)^2 \mu_{r,c}(t)^2\} = E\{s_{r,c}(t) \mu_{r,c}(t)\}$. Furthermore, due to $q_{r,c}(t) \leq 1$, the queue $q_{r,c}(t)$ is always stable, we have $E\{s_{r,c}(t) \mu_{r,c}(t)\} = E\{(1 - p_{b,c}(t)) \mu_{b,c}(t)\} = (1 - p_{b,c}) \lambda_c$. Plugging these equations into $\overline{B}_2$ yields

$$\overline{B}_2 = 2 \sum_{c \in N_M} (1 - p_{b,c})(\lambda_c - \lambda_c^2). \quad (22)$$

As shown in (16), the sufficient condition to stabilize the queuing system is $\beta > 0$ for all $q_{r,c}(t) \geq 0$. Thus, the sufficient condition for ε is $\overline{B}_2 - 2\varepsilon(1 - p_{max}) > 0$. Using (22), the sufficient condition for the stability of the queueing system under TOSC-wo is $\varepsilon > \frac{1}{(1-p_{max})} \sum_{c \in N_M} (1 - p_{b,c})(\lambda_c - \lambda_c^2)$. This proves Theorem 3.

Competing interests

The authors declare that they have no competing interests

References

1. A Ozgur, O Leveque, DL Tse, Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks. IEEE Trans Inf Theory **53**(10), 3549–3572 (2007)

2. SW Peters, RW Heath, The future of WiMAX: multihop relaying with IEEE 802.16j. Commun Mag. **47**, 104–111 (2009)

3. A Sadek, K Liu, A Ephremides, Cognitive multiple access via cooperation: protocol design and performance analysis. IEEE Trans Inf Theory **53**(10), 3677–3696 (2007)

4. L Tassiulas, A Ephremides, Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. IEEE Trans Automat Control **37**(12), 1936–1948 (1992). doi:10.1109/9.182479

5. A Muller, HC Yang, Dual-hop adaptive packet transmission systems with regenerative relaying. IEEE Trans Wirel Commun. **9**, 234–244 (2010)

6. J Laneman, D Tse, G Wornell, Cooperative diversity in wireless networks: efficient protocols and outage behavior. IEEE Trans Inf Theory **50**(12), 3062–3080 (2004). doi:10.1109/TIT.2004.838089

7. G Kramer, M Gastpar, P Gupta, Cooperative strategies and capacity theorems for relay networks. IEEE Trans Inf Theory **51**(9), 3037–3063 (2005). doi:10.1109/TIT.2005.853304

8. MJ Neely, R Urgaonkar, Optimal backpressure routing for wireless networks with multi-receiver diversity. Ad Hoc Netw. **7**, 862–881 (2009). doi:10.1016/j.adhoc.2008.07.009

9. C Lott, D Teneketzis, Stochastic routing in ad-hoc networks. IEEE Trans Automat Control. **51**, 52–70 (2006). doi:10.1109/TAC.2005.860280

10. B Rong, A Ephremides, Protocol-level cooperation in wireless networks: stable throughput and delay analysis, in *Proceedings of the 7th international conference on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT'09)*, Seoul, Korea, 151–160 (2009)

11. L Georgiadis, MJ Neely, L Tassiulas, Resource allocation and cross-layer control in wireless networks. Found. Trends Netw. **1**, 1–144 (2006)

12. E Yeh, R Berry, Throughput optimal control of cooperative relay networks. IEEE Trans Inf Theory **53**(10), 3827–3833 (2007)

13. Y Shi, S Sharma, YT Hou, S Kompella, Optimal relay assignment for cooperative communications, in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc '08)*, Hong Kong, China, 3–12 (2008)

14. RC Daniels, RW Heath, An online learning framework for link adaptation in wireless networks, in *Proceedings of the Information Theory and Applications Workshop*, San Diego, CA, USA, 138–140 (2009)

15. M Kodialam, T Nandagopal, Characterizing the capacity region in multi-radio multi-channel wireless mesh networks, in *Proceedings of the 11th annual international conference on Mobile computing and networking (MobiCom '05)*, Cologne, Germany, 73–87 (2005)

16. Y Wang, S Boyd, Performance bounds for linear stochastic control. Syst Control Lett. **58**(3), 178–182 (2009). doi:10.1016/j.sysconle.2008.10.004

17. RW Sargent, Optimal control. Comput Appl Math. **124**, 361–371 (2000). doi:10.1016/S0377-0427(00)00418-0

18. M Andrews, K Kumaran, K Ramanan, A Stolyar, P Whiting, R Vijayakumar, Providing quality of service over a shared wireless link. IEEE Commun Mag. **39**(2), 150–154 (2001). doi:10.1109/35.900644