WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2015; **15**:787–799
Published online 8 July 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/wcm.2385

RESEARCH ARTICLE

Capacity region and dynamic control of wireless networks under per-link queueing

Zongrui Ding, Yang Song, Dapeng Wu* and Yuguang Fang

Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, U.S.A.

ABSTRACT

The capacity region of wireless networks with per-destination (PD) queueing model has been studied extensively in the literature. However, the PD queueing structure is not scalable because the number of queues in a node can be as large as the number of all possible source–destination pairs. In this work, we study the capacity region of wireless networks with per-link (PL) queueing model. The advantage of the PL queueing structure is that the number of queues in a node can be reduced significantly to the number of its neighboring nodes. In this paper, the capacity region of a wireless network with PL queueing structure is characterized, and a dynamic routing and power control policy, namely, DRPC-PL, is proposed to stabilize the network whenever the input rate is within the capacity region. Copyright © 2013 John Wiley & Sons, Ltd.

***Correspondence**
Dapeng Wu, Department of Electrical & Computer Engineering, University of Florida, Gainesville, FL, U.S.A.
E-mail: wu@ece.ufl.edu

1. INTRODUCTION

Wireless networks are acting as ubiquitous infrastructure for many applications with increasing capacity demand. As introduced in [1,2], the capacity region of a wireless network theoretically characterizes the set of the input rates that the network can support under stability. Therefore, it is of great importance to derive the capacity region of a wireless network because (i) fair and effective networking techniques can be designed to take full advantage of the network resource; (ii) the capacity region provides important information for admission control; and (iii) the capacity region can serve as a theoretical guidance in deriving effective dynamic control policy to improve the network performance, for example, throughput and network utilities.

In wireless networks, there are two commonly used queueing structures: the per-destination (PD) queueing model as in [1,3–5] and the per-link (PL) queueing model as in [3,6,7]. In the PD queueing networks, every node needs to maintain a queue for each flow, that is, every possible source–destination pair, which could be prohibitive in a large-scale network. In contrast, it is shown in [7] that the PL, queueing structure requires less complexity and preserves good design features such as network decomposition [8]. By "per-link queue," we mean that packets destined to the same next hop node are put into the same queue. Thus, the number of queues for a node to maintain is the number of its neighbors within one hop.

Most existing works on capacity region analysis are based on the PD queueing structure, for example, [1,2,9–15]. The seminal work [2] first defines the stability region of a wireless network and proposes a throughput optimal algorithm MaxWeight, also known as back-pressure algorithm, which has been studied extensively in the literature, for example, [13–15]. In [1], Neely *et al*. further explore the capacity region of a stochastic network with time-varying channels and derive the necessary and sufficient conditions of the capacity region on the basis of the Lyapunov drift method [11]. The proposed dynamic routing and power control (DRPC) algorithm [1] can stabilize the network in the capacity region. Song *et al*. propose a minimum energy scheduling in [9], which achieves both minimum energy consumption and throughput optimality by considering link retransmissions. In [6], Bui *et al*. propose a shadow algorithm under PL queueing model and use counters to mimic the PD queueing dynamics with static routing.

In this work, we study the capacity region of wireless networks with PL queueing structure and propose a dynamic routing and power control policy (DRPC-PL) to stabilize the network. Our system model shares some similarities with the route-dependent case in [3,6]. Whereas existing works assume static routing under the PL queueing model, our work relaxes this assumption by considering all possible power allocation and routing schemes that can stabilize the network. Although the Lyapunov drift method provides a powerful tool to study the network stability issue, as in [11], the input and output of the PL queues are difficult to characterize, which makes the problem challenging. Our key result is to derive the sufficient and necessary conditions of the capacity region with PL queueing and propose a stabilizing routing and power control algorithm to stabilize the network.

The rest of the paper is organized as follows. In Section 2, we describe the system model for the PL queueing networks. In Section 3, we define the capacity region and derive the necessary and sufficient conditions, followed by a stabilizing control policy DRPC-PL. Section 4 theoretically analyzes the performance of the proposed DRPC-PL, and Section 5 concludes the paper.

2. THE NETWORK MODEL

In this paper, we consider a network modeled by a graph, $\mathcal{G} = (\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{L}$ is the set of links. A link (i, j) denoted by $l_{i,j}$ exists if it is in $\mathcal{L}$. The set of the neighboring nodes of node i is denoted by $\mathcal{N}_i$. We assume a slotted system, and the time slot is denoted by t. In every time slot, let $s_{i,j}(t)$ represent the instantaneous channel state of $l_{i,j} \in \mathcal{L}$, where i and j are the transmitter and receiver of the link. The channel state matrix of the network can be denoted by $S(t) = [s_{i,j}(t)]$.

Let $\mathcal{F}$ be the set of flows consisted in the network where each of them is indexed by $f = 1, 2, ..., |\mathcal{F}|$ and $|\cdot|$ is the cardinality of a set. The destination node of flow f is denoted by $d(f)$. In the network, let μ_l represent the transmission rate on link l, and then a schedule $\pi = \left(\mu_1^\pi, \mu_2^\pi, \ldots, \mu_{|\mathcal{L}|}^\pi\right)$ is the link rates that can be supported simultaneously by the network. We use the strong stability in [16] to define the network stability.

2.1. The per-link queueing model

The PL queueing model is illustrated in Figures 1 and 2. Solid lines are links between two nodes, and dashed lines indicate the route between two queues to exchange data. In these two figures, $q_{i,j}$, defined on link $l_{i,j}$, holds the data in node i with node j as the next hop node. Therefore, the number of the queues in node i is at most the number of its neighbors. If $l_{i,j}$ is active, packets in $q_{i,j}$ are transmitted.

Let $\underline{\mu}_{i,j}(t)$ denote the allocated input data rate of $q_{i,j}$ and $\underline{\mu}_{m,i,j}(t)$ denote the allocated data rate from $q_{m,i}$ to $q_{i,j}$ at time t. For a packet in $q_{i,j}$, when it is transmitted to node j, the destination of it will be checked against j. If the packet is destined to node j, it goes out of the queueing system. Otherwise, it shall enter another queue, say $q_{j,k}$ on the basis of the routing. Then the packets not destined to node j yield a data rate from $q_{i,j}$ to $q_{j,k}$. Let

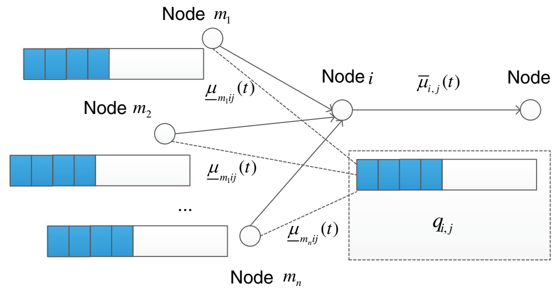


Figure 1. Input model for PL queue.

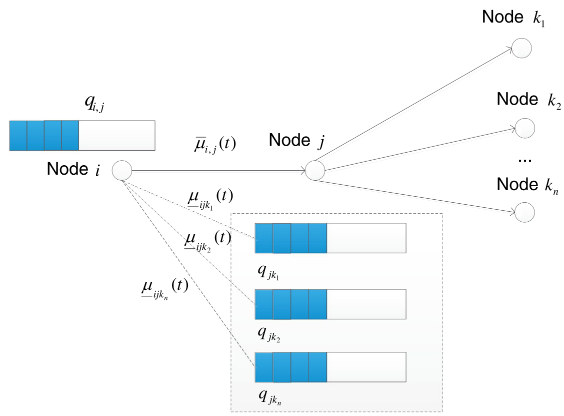


Figure 2. Output model for PL queue.

$\overline{\mu}_{i,j}(t)$ represent the allocated output data rate of $q_{i,j}$ and $\mu_{i,j,k}(t)$ represent the allocated data rate from $q_{i,j}$ to $q_{j,k}$ at time t. We have

$$\underline{\mu}_{i,j}(t) = \sum_m \underline{\mu}_{m,i,j}(t)$$

$$\overline{\mu}_{i,j}(t) = \sum_k \mu_{i,j,k}(t) + \alpha_{i,j}(t)$$

where $\alpha_{i,j}(t)$ is the data rate delivered from node i to node j at slot t.

In addition, because each queue contains the traffic for different flows, we can write $\mu_{i,j,k}(t)$ and $\underline{\mu}_{m,i,j}(t)$ in terms of the flow rates. Then we have

$$\mu_{i,j,k}(t) = \sum_{f:d(f)\neq j} \mu_{i,j,k}^f(t)$$

$$\underline{\mu}_{m,i,j}(t) = \sum_{f:d(f)\neq i} \mu_{m,i,j}^f(t)$$

where $\mu_{i,j,k}^f(t)$ is the allocated rate for flow f on $l_{i,j}$ from $q_{i,j}$ to $q_{j,k}$ at time t.

Wirel. Commun. Mob. Comput. 2015; **15**:787–799 © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/wcm

2.2. The network model

We consider a wireless network in consecutive K time slots. The PL queueing model is characterized by the following properties.

(1) Convergent wireless channels

Let $T_S(t, K)$ be the set of time slots at which the channel state matrix $S(t) = S$ during the interval $0 \le t \le K-1$. The wireless channel process $S(t)$ is assumed to be convergent with a finite number of channel states $\{S\}$ and state probabilities π_S. The convergence interval [1] K is the number of time slots, such that for a given value $\delta_1 > 0$, we have

$$\sum_{\{S\}} \left| \frac{\mathbb{E}\{|T_S(t,K)|\}}{K} - \pi_S \right| \le \delta_1 \qquad (1)$$

where $\mathbb{E}\{\cdot\}$ is the expectation.

(2) Bounded and convergent arrival rates

For a given $\delta_2 > 0$, an arrival process $a_{i,j}^f(t)$ convergent with the exogenous arrival rate $\lambda_{i,j}^f$ within interval K satisfies

$$\left| \lambda_{i,j}^f - \frac{1}{K} \sum_{t=0}^{K-1} \mathbb{E}\left\{ a_{i,j}^f(t) \right\} \right| \le \delta_2 \qquad (2)$$

Besides, the second moment of exogenous arrivals at each node is bounded every time slot by some finite maximum value $a_{\max}^f$ regardless of past history, so that for any $i \in \mathcal{N}$, $j \in \mathcal{N}_i$,

$$\mathbb{E}\left\{ \left[a_{i,j}^f(t) \right]^2 \right\} \le \left(a_{\max}^f \right)^2$$

$$\mathbb{E}\left\{ \left[\sum_f a_{i,j}^f(t) \right]^2 \right\} \le (a_{\max})^2$$

where $a_{\max}^f$ and $a_{\max}$ are constants.

(3) Upper semicontinuous power-rate function.

Let $\overline{\mu}_{i,j}(P(t), S(t))$ denote the power-rate function under some power allocation matrix $P(t) = [p_{i,j}(t)]$, $P(t) \in \mathcal{P}$ and channel state matrix $S(t)$, where $\mathcal{P}$ is the set of feasible power allocation. Each element $p_{i,j}(t)$ is the allocated power on $l_{i,j}$ at time t. The transmission rates are bounded for every time slot t by $\mu_{\max}$, so that

$$\mu_{\max} = \max_{j \in \mathcal{N}_i, S, P \in \mathcal{P}} \overline{\mu}_{i,j}(P, S)$$

which can also be deduced by the fact that the power-rate function is bounded with finite transmission power.

(4) Queue update dynamics

Let $U_{i,j}(t)$ denote the backlog of $q_{i,j}$ at time t. Then the queueing dynamics in the network satisfy

$$U_{i,j}(t+1) \le \left[U_{i,j}(t) - \overline{\mu}_{i,j}(t) \right]^+ + \underline{\mu}_{i,j}(t) + \sum_f a_{i,j}^f(t) \qquad (3)$$

where $[x]^+ = \max(x, 0)$ and it is an inequality instead of an equality because the arrivals may be less than the allocated output data if the neighboring nodes have little or no data to transmit [1].

3. STABILIZING CONTROL POLICY

In this section, we design a dynamic control policy, namely, DRPC-PL, to stabilize the PL queueing networks. To provide some intuitions on how to design the stabilizing algorithms, we first give a brief introduction to the Lyapunov drift method.

3.1. Lyapunov drift method

A sufficient condition for network stability using the Lyapunov theory is given in Lemma 2 in [17]. Assume the set of queues in a system is denoted by $\mathcal{X}$ and $\mathbf{U}(t) = [U_x(t)]$ is the queue backlog vector for multiple queues and $L(\mathbf{U}(t)) = \sum_x U_x^2(t)$ is the Lyapunov function for the queue states. Note that this model is applicable to any system that contains multiple queues. If there exists a positive integer K such that for every time slot, the Lyapunov function-evaluated K steps into the future satisfies

$$\mathbb{E}\{L(\mathbf{U}(K+t)) - L(\mathbf{U}(t)) | \mathbf{U}(t)\} \le B_1 - \sum_x \theta_x U_x(t) \quad (4)$$

for some positive constants B_1, $[\theta_x]$, and if $\mathbb{E}\{\mathbf{U}(t)\} \le \infty$ for $t \in \{0, 1, \cdots, K-1\}$, then the network is stable, and

$$\limsup_{K \to \infty} \frac{1}{K} \sum_{t=0}^{K-1} \left[\sum_x \theta_x \mathbb{E}\{U_x(t)\} \right] \le B_1$$

Similar to (3), the queue backlog for K slots into the future can be bounded in terms of the current unfinished work

$$U_x(t+K) \le \left[U_x(t) - \sum_{t=0}^{K-1} \mu_x(t) \right]^+ + \sum_{t=0}^{K-1} a_x(t)$$

where $\mu_x(t)$ is the output rate of queue x and $a_x(t)$ is the queue input rate including both the exogenous arrivals and the traffic from other queues. Squaring both sides of the inequality above and taking conditional expectations with respect to $\mathbf{U}(t)$, we have

$$\mathbb{E}\left\{U_x^2(t+K) - U_x^2(t)|\mathbf{U}(t)\right\} \leq K^2\left(\mu_{\max}^2 + a_{\max}^2\right) - 2K\sum_x(\bar{\mu}_x - \lambda_x)U_x(t) \quad (5)$$

where $\mu_{\max} = \max_x \mu_x(t)$ and $a_{\max} = \max_x a_x(t)$, and the service rate $\mu(t)$ and the arrival rate $a(t)$ are convergent to the rate $\bar{\mu} = [\bar{\mu}_x]$ and $\lambda = [\lambda_x]$ similar to (2).

Comparing (4) and (5), we can see that if there exists $\epsilon > 0$, such that $\overline{\mu}_x - \lambda_x \geq \epsilon$, then the constant θ_x can be found to stabilize all the queues. Therefore, a stabilizing algorithm should be designed to keep the average service rate larger than the arrival rate for each queue in the network during K slots.

3.2. Per-Destination Per-Link (PDPL) network

We first define a PDPL queueing network where each node keeps a queue for each source–destination-link combination. Thus, a node in a PDPL network with $|\mathcal{F}|$ flows, and $|\mathcal{L}|$ links will have $|\mathcal{F}|\times|\mathcal{L}|$ PDPL queues. This PDPL network contains more queues than that with the PD queues and will be used as a reference network in our following analysis.

Let $q_{i,j}^f$ denote the queue for flow f on $l_{i,j}$, and let $\overline{\mu}_{i,j}^f(t)$ and $\underline{\mu}_{i,j}^f(t)$ denote the instantaneous input and output rates of $q_{i,j}^f$ at time t to obtain

$$\underline{\mu}_{i,j}^f(t) = \sum_m \mu_{m,i,j}^f(t) \quad (6a)$$

$$\overline{\mu}_{i,j}^f(t) = \sum_k \mu_{i,j,k}^f(t) \quad (6b)$$

Define Λ_{pdpl} as the set of all input rate matrices $\left[\lambda_{i,j}^f\right]$ of the PDPL network such that there exist flow rate variables $\left[d_{i,j,k}^f\right]$ satisfying

$$d_{i,j,k}^f \geq 0, \forall i,j,k \in \mathcal{N} \quad (7a)$$

$$d_{i=d(f),j,k}^f = 0 \quad (7b)$$

$$\sum_k d_{i,j,k}^f - \sum_m d_{m,i,j}^f \geq \lambda_{i,j}^f, \forall d(f) \neq j \quad (7c)$$

$$[G_{i,j}] \geq \left[\sum_k d_{i,j,k}^f\right] \quad (7d)$$

where the inequality of (7d) is considered entrywise and $\left[d_{i,j,k}^f\right]$ is derived in (27) later. Note that the input rate matrix $\left[\lambda_{i,j}^f\right]$ is different from the link rate matrix $G = [G_{i,j}]$ and is in a time-average sense.

The following theorem gives the necessary and sufficient condition of the capacity region of a network under PDPL queueing structure.

Theorem 1. *Capacity region for the PDPL queueing network*

(a) *A necessary condition for network stability is* $\left[\lambda_{i,j}^f\right] \in \Lambda_{\text{pdpl}}$.
(b) *A sufficient condition for network stability is that* $\left[\lambda_{i,j}^f\right]$ *is strictly interior to* Λ_{pdpl}.

The proof is in Appendix A.

Note that the capacity region of a wireless network does not depend on a certain routing or scheduling algorithm.

3.3. The dynamic routing and power control policy algorithm

Following the clues provided by the Lyapunov drift method, we develop a control policy that stabilizes the PL queueing network whenever the input rate matrix $\left[\lambda_{i,j}^f\right]$ is inside the network capacity region Λ_{pl}.

First, we define the PDPL shadow queues. The PDPL shadow queues are counters that keep track of the PD traffic for each flow in the PL queues. Let $V_{i,j}^f(t)$ denote the queue backlog for flow f in $q_{i,j}$ at time t. Then the PDPL shadow queue length can be updated according to

$$V_{i,j}^f(t+1) = \left[V_{i,j}^f(t) - \overline{\mu}_{i,j}^f(t)\right]^+ + a_{i,j}^f(t) + \underline{\mu}_{i,j}^f(t) \quad (8)$$

where $\overline{\mu}_{i,j}^f(t)$, $\underline{\mu}_{i,j}^f(t)$ and $a_{i,j}^f(t)$ are defined in (6).

3.3.1. Dynamic routing and power control policy algorithm.

For every time slot t, the DRPC-PL schedules the network as follows:

(1) For each link $l_{i,j}$, the weight $W_{i,j}(t)$ is calculated by

$$W_{i,j}(t) = \overline{\mu}_{i,j}(t)\max_f\left(V_{i,j}^f(t) - \min_k V_{j,k}^f(t)\right)$$

where $\overline{\mu}_{i,j}(t)$ is the transmission rate on $l_{i,j}$ at time t under power control.

(2) Power allocation to $l_{i,j}$ follows:

$$P(t) = \arg\max_{P\in\mathcal{P}}\sum_{i,j} W_{i,j}(P(t), S(t)) \quad (9)$$

(3) Transmit the packets in $q_{i,j}$ according to the allocated rate yielded by the power allocation in (9) and route packets going through $l_{i,j}$ for flow f to q_{j,k^*}, where

Wirel. Commun. Mob. Comput. 2015; **15**:787–799 © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/wcm

$$k^* = \arg\min_k \left(V^f_{j,k}(t)\right) \tag{10}$$

Then calculate $\left[V^f_{i,j}(t+1)\right]$ according to (8), where $\overline{\mu}^f_{i,j}(t)$ and $\underline{\mu}^f_{i,j}(t)$ can be calculated according to (6) by monitoring the flow-based data transmitted on each link $l_{i,j}$.

Different from the DRPC algorithm proposed in [1], DRPC-PL transmits the packets for different flows from the PL queues if the link is active. DRPC-PL essentially chooses a set of links without interference, which gives the maximum weight of the network on the basis of the PDPL shadow queue difference. The traffic for a flow is dynamically routed to choose the next hop. Therefore, the DRPC algorithm schedules $l_{i,j}$ and performs routing according to the maximum PD queue difference between two nodes, whereas the DRPC-PL algorithm schedules $l_{i,j}$ and performs routing according to the maximum PDPL shadow queue difference between two links. From (9), we can see that the power allocation can also be regarded as a MaxWeight problem similar in DRPC. Much progress has been made in easing the computational complexity and deriving decentralized solutions for the centralized MaxWeight algorithm [14,15,18–26]. The DRPC-PL algorithm is also different from the PD shadow queue scheme introduced in [6]. The PDPL shadow queues keep real-time backlog for flows in the PL queues, whereas the PD shadow queues in [6] do not reflect the real-time backlog of different flows.

The performance of the DRPC-PL algorithm is stated as follows.

Theorem 2. *The proposed DRPC-PL algorithm is throughput optimal, that is, for an arbitrary network admission rate vector* $\left[\lambda^f_{i,j}\right]$ *inside of the network capacity region denoted by* Λ_{pl}, *the DRPC-PL algorithm stabilizes the network.*

The proof is presented in the next section.

4. PERFORMANCE ANALYSIS

In this section, we evaluate the performance of DRPC-PL by comparing it with the dynamic control policy for PDPL queueing networks, that is, DRPC–PDPL algorithm. In Section 4.1, we present the DRPC–PDPL algorithm that can stabilize the PDPL queueing network within the capacity region Λ_{pdpl}. Then, we prove that the DRPC-PL algorithm stabilizes the PL queues whenever the input rate is within the capacity region of the PDPL network Λ_{pdpl} in 4.2. In Section 4.3, we prove that the capacity region of a network under PDPL queueing and PL queueing is the same to complete the proof of Theorem 2.

4.1. Performance of the DRPC–PDPL algorithm

Under the PDPL network model, the DRPC–PDPL algorithm schedules the network as follows:

(1) For each link $l_{i,j}$, choose flow f^* to calculate the link weight $W_{i,j}(t)$, where

$$f^* = \arg\max_f \left(V^f_{i,j}(t) - \min_k V^f_{j,k}(t)\right)$$

and

$$W_{i,j}(t) = \overline{\mu}_{i,j}(t)\left(V^{f^*}_{i,j}(t) - \min_k V^{f^*}_{j,k}(t)\right)$$

(2) The power allocation to $l_{i,j}$ follows

$$P(t) = \arg\max_{P\in\mathcal{P}} \sum_{i,j} W_{i,j}(P(t), S(t))$$

(3) Route the packets of flow f^* in $q^f_{i,j}$ to $q^{f^*}_{j,k^*}$ if $d(f^*) \neq j$, where

$$k^* = \arg\min_k V^{f^*}_{j,k}(t)$$

Here, we reuse $V^f_{i,j}(t)$ to denote the queue length of the physical PDPL queue $q^f_{i,j}$.

Note that DRPC–PDPL is different from the DRPC algorithm [1] because of the queueing model. In DRPC, the packets for a flow are transmitted and routed between two nodes, whereas the packets of a flow are transmitted and routed between two links in DRPC–PDPL. The performance of the DRPC–PDPL is stated in the following theorem.

Theorem 3. *The proposed DRPC–PDPL algorithm is throughput optimal, that is, for an arbitrary network admission rate vector* $\left[\lambda^f_{i,j}\right]$, *which is inside of the network capacity region* Λ_{pdpl}, *DRPC–PDPL stabilizes the network.*

The proof is in Appendix B.

4.2. Performance of the dynamic routing and power control policy

To prove Theorem 2, we first prove that for $\forall\ \left[\lambda^f_{i,j}\right] \in \Lambda_{\text{pdpl}}$, the DRPC-PL algorithm can stabilize the network. Then, we prove that the capacity region of a network under PDPL queueing is the same as that under PL queueing in Section 4.3.

By comparing the PDPL shadow queue lengths and the PL queue lengths in the DRPC-PL, we can easily get

$$U_{i,j}(t) = \sum_{f} V_{i,j}^{f}(t)$$

Thus, we want to evaluate the stability of the PL queues by evaluating the stability of the PDPL shadow queues.

The one-step Lyaponuv drift of the PDPL shadow queues can be written as

$$\left(V_{i,j}^{f}(t+1)\right)^2 - \left(V_{i,j}^{f}(t)\right)^2 \leq \left((\mu_{\max})^2 + (\mu_{\max} + a_{\max})^2\right) - 2V_{i,j}^{f}(t)\left(\overline{\mu}_{i,j}^{f}(t) - \underline{\mu}_{i,j}^{f}(t) - a_{i,j}^{f}(t)\right) \tag{11}$$

Next, we sum (11) over the whole network on all shadow queues and obtain

$$\sum_{f;i,j}\left(V_{i,j}^{f}(t+1)\right)^2 - \sum_{f;i,j}\left(V_{i,j}^{f}(t)\right)^2 \leq B - 2\sum_{f;i,j} V_{i,j}^{f}(t)\left(\overline{\mu}_{i,j}^{f}(t) - \underline{\mu}_{j,k}^{f}(t) - a_{i,j}^{f}(t)\right) \tag{12}$$

where

$$B = |\mathcal{L}||\mathcal{F}|\left((\mu_{\max})^2 + (\mu_{\max} + a_{\max})^2\right) \tag{13}$$

is a constant.

Taking the conditional expectation of (12) with respect to $\mathbf{V}(t) = \left[V_{i,j}^{f}(t)\right]$, we have

$$z`\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^{f}(t+1)\right)^2 \middle| \mathbf{V}(t)\right) - \mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^{f}(t)\right)^2 \middle| \mathbf{V}(t)\right) \leq B - 2\sum_{f;i,j} V_{i,j}^{f}(t)\mathbb{E}\left(\overline{\mu}_{i,j}^{f}(t) - \underline{\mu}_{i,j}^{f}(t) - a_{i,j}^{f}(t)|\mathbf{V}(t)\right) \tag{14}$$

Define the routing scheme at time t associated with each flow on $l_{i,j}$ by $\left[\beta_{i,j,k}^{f}(t)\right]$ $(d(f) \neq j)$, where $\beta_{i,j,k}^{f}(t) \in [0,1]$ and

$$\sum_{k} \beta_{i,j,k}^{f}(t) = 1, (d(f) \neq j)$$

Then, we can rewrite the output rate of each flow in terms of the routing parameters as

$$\underline{\mu}_{m,i,j}^{f}(t) = \beta_{m,i,j}^{f}(t)\overline{\mu}_{m,i}^{f}(t) \tag{15}$$

Substituting (15) into (14), we have

$$\begin{aligned}
&\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^{f}(t+1)\right)^2 \middle| \mathbf{V}(t)\right) - \mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^{f}(t)\right)^2 \middle| \mathbf{V}(t)\right) \\
&\leq B - 2\sum_{f;i,j} V_{i,j}^{f}(t)\mathbb{E}\left(\overline{\mu}_{i,j}^{f}(t) - \sum_{m}\beta_{m,i,j}(t)\overline{\mu}_{m,i}^{f}(t)|\mathbf{V}(t)\right) \\
&- \mathbb{E}\left(a_{i,j}^{f}(t)|\mathbf{V}(t)\right) = B - \mathbb{E}\left(a_{i,j}^{f}(t)|\mathbf{V}(t)\right) \\
&- 2\sum_{f;i,j}\mathbb{E}\left(\overline{\mu}_{i,j}^{f}(t)\left(V_{i,j}^{f}(t) - \sum_{k}\beta_{i,j,k}^{f}(t)V_{j,k}^{f}(t)\right)|\mathbf{V}(t)\right)
\end{aligned} \tag{16}$$

Now, define the maximum queue difference for each flow f and $l_{i,j}$ as

$$\begin{aligned}
\Delta V_{i,j,k}^{f}(t) &= V_{i,j}^{f}(t) - V_{j,k}^{f}(t) \\
\Delta V_{i,j}^{\max}(t) &= \max_{f;k} \Delta V_{i,j,k}^{f}(t) \\
&= \max_{f}\left(V_{i,j}^{f}(t) - \min_{k} V_{j,k}^{f}(t)\right)
\end{aligned}$$

Define

$$J_V^{PL}(t) = 2\mathbb{E}\left(\sum_{f;i,j}\overline{\mu}_{i,j}^{f}(t)\left(\Delta V_{i,j}^{\max}(t) + \Delta V_{i,j,k}^{f}(t)\right)|\mathbf{V}(t)\right)$$

and

$$\overline{J^{PL}}(t) = \mathbb{E}\left(J_V^{PL}(t)\right)$$

Next, we add both sides by $J_V^{PL}(t)$ to (14) and have

$$\begin{aligned}
&\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^{f}(t+1)\right)^2 \middle| \mathbf{V}(t)\right) - \mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^{f}(t)\right)^2 \middle| \mathbf{V}(t)\right) \\
&+ J_V^{PL}(t) \leq B + J_V^{PL}(t) \\
&- 2\sum_{f;i,j} V_{i,j}^{f}(t)\mathbb{E}\left(\overline{\mu}_{i,j}^{f}(t) - \underline{\mu}_{i,j}^{f}(t) - a_{i,j}^{f}(t)|\mathbf{V}(t)\right)
\end{aligned} \tag{17}$$

Wirel. Commun. Mob. Comput. 2015; **15**:787–799 © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/wcm

Denote the right-hand side of (17) as Θ, which can be rewritten as

$$B - \mathbb{E}\left(a_{i,j}^f(t)|\mathbf{V}(t)\right) - 2\sum_{f;i,j}\mathbb{E}\left(\overline{\mu}_{i,j}^f(t)\left(V_{i,j}^f(t) - \sum_k \beta_{i,j,k}^f(t)V_{j,k}^f(t)\right)|\mathbf{V}(t)\right) + 2\sum_{f;i,j}\mathbb{E}\left(\overline{\mu}_{i,j}^f(t)\left(\Delta V_{i,j}^{\max}(t) + \Delta V_{i,j,k}^f(t)\right)|\mathbf{V}(t)\right) \tag{18}$$

The last two terms of (18) can be simplified as

$$2\sum_{i,j}\mathbb{E}\left(\left(\sum_f \overline{\mu}_{i,j}^f(t)\right)\Delta V_{i,j}^{\max}(t)\right) - 2\sum_{f;i,j}\mathbb{E}\left(\overline{\mu}_{i,j}^f(t)\left(V_{i,j}^f(t) - \sum_k \beta_{i,j,k}^f(t)V_{j,k}^f(t) - \Delta V_{i,j,k}^f(t)\right)|\mathbf{V}(t)\right)$$

It is of great importance to observe that the DRPC-PL algorithm essentially minimizes the right-hand side of (17) over all possible scheduling algorithms.

Because $\left[\lambda_{i,j}^f\right]$ lies in the interior of the PDPL network capacity region Λ_{pdpl}, it immediately follows that there exists a small positive constant $\epsilon > 0$ such that

$$\left[\lambda_{i,j}^f\right] + \epsilon \in \Lambda_{\text{pdpl}}$$

We can get similar results to the Corollary 3.9 in [11] under PDPL queueing that there exists a randomized scheduling policy, denoted by RA, that stabilizes the PDPL network while providing a data rate of

$$\overline{\mu}_{i,j}^f(t) - \underline{\mu}_{j,k}^f(t) = \lambda_{i,j}^f + \epsilon$$

and $\overline{\mu}_{i,j}^f(t)$ and $\underline{\mu}_{j,k}^f(t)$ are the link data rates induced by the RA algorithm. Thus, we have

$$\theta^{RA} = B - 2\epsilon\sum_{f;i,j}V_{i,j}^f(t) + \sum_{f;i,j}\mathbb{E}\left(\overline{\mu}_{i,j}^f(t)\left(\underline{\Delta V}_{i,j}^{\max}(t) + \underline{\Delta V}_{i,j,k}^f(t)|\mathbf{V}(t)\right)\right) \tag{19}$$

where the last term in (19) is the queue difference under the RA algorithm during slot t. Therefore, the PDPL queues are bounded, so do the PDPL queue differences $\underline{\Delta V}_{i,j}^{\max}(t)$ and $\underline{\Delta V}_{i,j,k}^f(t)$. Thus, the last term of (19) is also bounded, and we denote the bound by $J_{\max}$. Then, we have

$$\theta^{RA} = B - 2\epsilon\sum_{f;i,j}V_{i,j}^f(t) + J_{\max}$$

In light of (17), it is obtained that

$$\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(t+1)\right)^2\middle|\mathbf{V}(t)\right) - \mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(t)\right)^2\middle|\mathbf{V}(t)\right) + J_V^{PL}(t) \le \Theta^{DRPC-PL} \le \Theta^{RA} \le B - 2\epsilon\sum_{f;i,j}V_{i,j}^f(t) + J_{\max} \tag{20}$$

Taking expectation with respect to $\mathbf{V}(t)$ to (20), we have

$$\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(t+1)\right)^2\right) - \mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(t)\right)^2\right) + \overline{J^{PL}}(t) \le B - 2\epsilon\sum_{f;i,j}\mathbb{E}\left(V_{i,j}^f(t)\right) + J_{\max} \tag{21}$$

Summing (21) over time slots 0 to $K-1$ yields

$$\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(t+1)\right)^2\right) - \mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(t)\right)^2\right) + \sum_{t=0}^{K-1}\overline{J^{PL}}(t) \le KB - 2\epsilon\sum_{t=0}^{K-1}\sum_{f;i,j}\mathbb{E}\left(V_{i,j}^f(t)\right) + KJ_{\max} \tag{22}$$

Then, we divide (22) by K and manipulate the result and have

$$2\epsilon\frac{1}{K}\sum_{t=0}^{K-1}\sum_{f;i,j}\mathbb{E}\left(V_{i,j}^f(t)\right) \le B + \frac{1}{K}\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(0)\right)^2\right) + J_{\max} - \frac{1}{K}\mathbb{E}\left(\sum_{f;i,j}\left(V_{i,j}^f(K)\right)^2\right) - \frac{1}{K}\sum_{t=0}^{K-1}\overline{J^{PL}}(t) \tag{23}$$

Note that the last two terms of (23) are both nonpositive. By taking $\limsup_{K\to\infty}$ to both sides of (23), we obtain

$$\limsup_{K\to\infty}\frac{1}{K}\sum_{t=0}^{K-1}\sum_{f;i,j}\mathbb{E}\left(V_{i,j}^f(t)\right) \le \frac{B + J^{\max}}{2\epsilon} < \infty \tag{24}$$

Remember that the PL queue lengths are the sum of a finite number of the PDPL shadow queue lengths. Then, we have

$$\limsup_{K\to\infty}\frac{1}{K}\sum_{t=0}^{K-1}\sum_{i,j}\mathbb{E}\left(U_{i,j}(t)\right) \le \infty$$

which implies the stability of the PL queues.

Therefore, we prove that the DRPC-PL algorithm stabilizes both the PDPL shadow queues and the PL queues within the PDPL capacity region Λ_{pdpl}.

4.3. The capacity region of per-link queueing network

Next, we study the relationship between the capacity region of a network under PDPL queueing and that under PL queueing as follows.

Theorem 4. *Let Λ_{pdpl} denote the capacity region of a wireless network under the PDPL queueing and Λ_{pl} denote the capacity region of the network under PL queueing, then $\Lambda_{\text{pdpl}} = \Lambda_{\text{pl}}$.*

Proof. First, we prove that $\Lambda_{\text{pl}} \in \Lambda_{dpdl}$. For any $[\lambda_{i,j}^f] \in \Lambda_{\text{pl}}$, there exists a scheduling algorithm R that can stabilize the network. At slot t, R yields a power allocation matrix $P0(t)$ for the links and a transmission rate matrix $\left[\mu 0_{i,j,k}^f(t)\right]$ for the flows under PL queueing. Suppose initial queue state of the network under PDPL is the same as the PDPL shadow queue state of the network under PDPL queueing. Then, by allocating the power according to $P0(t)$ to the links under PDPL queueing and scheduling the PDPL queues the same as the PDPL shadow queues under PL queueing, the transmission rate $\left[\mu 0_{i,j,k}^f(t)\right]$ can be achieved under PDPL queueing. This implies that $\lambda_{i,j}^f \in \Lambda_{\text{pdpl}}$. Second, any $\left[\lambda_{i,j}^f\right] \in \Lambda_{\text{pdpl}}$ can be supported by DRPC-PL algorithm in Section 3.3 under PL queueing. Therefore, $\Lambda_{\text{pdpl}} \in \Lambda_{\text{pl}}$. In summary, we have $\Lambda_{\text{pdpl}} = \Lambda_{\text{pl}}$. □

5. CONCLUSIONS AND FUTURE WORK

In this paper, we study the capacity region of wireless networks with PL queueing structure, which significantly reduces the queueing complexity of each node compared with the well-studied PD queueing structure. We first characterize the capacity region of the PL network and propose a dynamic control policy denoted by DRPC-PL to schedule the links and route the packets to stabilize the network. We show that DRPC-PL can stabilize the network whenever the input rate matrix is inside the capacity region of the PL queueing networks. In our future work, we will compare the capacity regions of wireless networks under both PL and PD queueing structure.

APPENDIX A: PROOF OF THEOREM 1

In this section, we provide the sketch of the proof of Theorem 1. First, we prove that $\left[\lambda_{i,j}^f\right] \in \Lambda_{dpdl}$ is the necessary condition for network stability, and then we prove that whenever the input rate is inside the capacity region, there exists a power control and routing algorithm that can stabilize the network.

Necessity

Consider a PDPL queueing network with convergent input rates $[\lambda_{i,j}^f]$ defined in (2), and let $A_{i,j}^f(t)$ represent the amount of packets that enter the network exogenously at node i with node j as the next hop node during the interval $[0, t]$. Suppose the system is stabilizable by some routing and power control policy. Let $Q_{i,j}^f(t)$ represent the resulting unfinished work for $q_{i,j}^f$. Let $D_{m,i,j}(t)$ represent the total number of packets that enter $q_{i,j}^f$ from node m during the interval $[0, t]$ and $D_{i,j,k}^f(t)$ represent the total number of packets that depart from $q_{i,j}^f$ and go into $q_{j,k}^f$. We can obtain

$$D_{i,j,k}(t) \geq 0 \tag{25a}$$

$$\sum_k D_{i,j,k}^f(t) - \sum_m D_{m,i,j}^f(t) = A_{i,j}^f(t) - Q_{i,j}^f(t) \tag{25b}$$

$$\int_0^t \mu_{i,j}^f(P(t), S(t))dt \geq \sum_k D_{i,j,k}^f(t) \tag{25c}$$

where (25b) follows that the unfinished work in any node is equal to the difference between the total number of packets that have arrived and departed. Inequality (25c) holds because the number of packets transferred over any link is less than or equal to the offered transmission rate integrated over the time interval $[0, t]$ and (25b) holds for the scenario that $D_{i,j,k}^f(t)$ contains padded null packets if there is not enough traffic for transmission. Note that although some control policy can stabilize the network, the power allocation process $P(t)$ is not necessarily ergodic nor are the internal bit streams produced by routing decisions.

Because the channel process is convergent to a finite state space as defined in (1), when measured over any sufficiently large interval $[0, t]$, the time fraction of a channel state S satisfies $||T_S(t)||/t \to \pi_S$ with probability 1. Besides, the input process is rate convergent to $[\lambda_{i,j}^f]$ as defined in (2). From Lemma 1 in [17], we known that if the network is stable, for any $\delta \geq 0$, there exists a finite value M, for which arbitrarily large times $\tilde{t}$ can be found so that $Pr\left[\sum_{i,j} U_{i,j}^f(\tilde{t})\right] \geq 1 - \delta$. Therefore, there must exist some finite value M such that at arbitrarily large times $\tilde{t}$, the unfinished work in all queues is simultaneously less than M with probability at least $\frac{1}{2}$. Hence, there exists a time $\tilde{t}$ such that with probability at least $\frac{1}{2}$, all of the following inequalities are satisfied:

$$Q^f_{i,j}(\tilde{t}) \le M \tag{26a}$$

$$\frac{M}{\tilde{t}} \le \epsilon \tag{26b}$$

$$\frac{A^f_{i,j}}{\tilde{t}} \ge \Lambda^f_{i,j} - \epsilon \tag{26c}$$

$$\frac{||T_S(\tilde{t})||}{\tilde{t}} \le \pi_S + \epsilon, \forall S \tag{26d}$$

Now, define:

$$d^f_{i,j,k} \triangleq \frac{D^f_{i,j,k}(\tilde{t})}{\tilde{t}} \tag{27}$$

By substituting (26) to (25b), it follows that for all $l_{i,j}$,

$$(\lambda^f_{i,j} + \epsilon) \le \sum_m d^f_{m,i,j} - \sum_k d^f_{i,j,k}$$

Thus, the necessity is proved.

Sufficiency

The proof in Section B directly proves that DRPC–PDPL can stabilize the network when the input rate is within the capacity region of the PDPL network.

APPENDIX B: PROOF OF THEOREM 3

In this section, we prove that the DRPC–PDPL algorithm can stabilize the network whenever the input rate is within the capacity region defined in Section 3.2.

Performance of the Static PDPL (STAT-PDPL) algorithm

The STAT-PDPL is a stationary randomized algorithm assuming that the values of $[G_{i,j}]$, $\left[d^f_{i,j,k}\right]$ are known to the scheduler. Suppose the input rate matrix satisfying $\lambda^f_{i,j} + \epsilon \in \Lambda_{\text{pdpl}}$ and the channel probability π_S are known in advance, then a set of flow variables $\left[d^f_{i,j,k}\right]$ and a link rate matrix $[G_{i,j}]$ must exist following Theorem 1.

Because the physical channel and interference model are the same as those in [1], Lemma 8 (graph family achievability) also holds for the PL queueing networks. Therefore, a stationary randomized power allocation policy P^1 can be implemented, yielding a transmission rate matrix $\mu 1(t)$, which is entrywise convergent with rate matrix $[G_{i,j}]$, and we can have

$$\left| \frac{1}{K} \sum_{t=0}^{K-1} \mathbb{E}\{\mu 1_{i,j}(t)\} - G_{i,j} \right| \le \delta \tag{28}$$

The STAT-PDPL algorithm is described as follows.

(1) Power allocation—For every time slot, observe the channel state $S(t)$, and allocate the power according to P^1 [17];
(2) Scheduling and routing—For every $l_{i,j}$, transmit a single flow f randomly chosen with probability $\frac{\sum_k d^f_{i,j,k}}{\sum_f \sum_k d^f_{i,j,k}}$, and route the packet for flow f randomly to $q^f_{j,k}$ with probability $\frac{d^f_{i,j,k}}{\sum_k d^f_{i,j,k}}$, then we have

$$\mu 1^f_{i,j}(t) = \begin{cases} \frac{\mu 1_{i,j}(t) \sum_k d^f_{i,j,k}}{G_{i,j}}, & \text{if flow } f \text{ is chosen} \\ 0, & \text{otherwise} \end{cases} \tag{29}$$

If a node does not have enough (or any) packets of a certain flow to send over its output links, null packets are delivered, so that links have idle times that are not used by other flows. In the light of (29), we have

$$\mathbb{E}\{\mu 1^f_{i,j}(t) | \mu 1_{i,j}(t)\} = \frac{\mu 1_{i,j}(t) \sum_k d^f_{i,j,k}}{G_{i,j}} \tag{30}$$

and

$$\mathbb{E}\{\mu 1^f_{i,j,k}(t) | \mu 1_{i,j}(t)\} = \frac{\mu 1_{i,j}(t) d^f_{i,j,k}}{G_{i,j}} \tag{31}$$

The performance of the STAT-PDPL algorithm is given as follows.

Lemma 1. *Consider a wireless network with capacity region* Λ_{pdpl} *and input rates* $\left[\lambda^f_{i,j}\right]$ *such that* $\left[\lambda^f_{i,j} + \epsilon\right] \in \Lambda_{\text{pdpl}}$ *for some* $\epsilon \ge 0$. *Then the algorithm STAT-PDPL stabilizes the network.*

Proof. Here, we prove that if the input rate $\left[\lambda^f_{i,j}\right] \in \Lambda$, the joint power control and routing algorithm STAT-PDPL can stabilize the network.

Take conditional expectations with respect to $\mu 1_{i,j}(t)$ on both sides of (30) and (31), and substitute them into (28). We have

$$\left| \frac{1}{K} \sum_{t=0}^{K-1} \left[\mathbb{E}\left\{ \mu 1^f_{i,j,k}(t) \right\} - d^f_{i,j,k} \right] \right| \le \frac{\epsilon_1}{6} \tag{32}$$

$$\left| \frac{1}{K} \sum_{t=0}^{K-1} \left[\mathbb{E}\left\{ \mu 1^f_{m,i,j}(t) \right\} - d^f_{m,i,j} \right] \right| \le \frac{\epsilon_2}{6} \tag{33}$$

where $\epsilon_1 > 0$ and $\epsilon_2 > 0$ are carefully selected constants. Summing (32) over k produces

$$\begin{aligned}&\left|\frac{1}{K}\sum_{t=0}^{K-1}\left[\sum_k \mathbb{E}\left\{\mu 1^f_{i,j,k}(t)\right\}-\sum_k d^f_{i,j,k}\right]\right|\\&\quad\le \sum_k\left|\frac{1}{K}\sum_{t=0}^{K-1}\left[\mathbb{E}\left\{\mu 1^f_{i,j,k}(t)\right\}-d^f_{i,j,k}\right]\right|\\&\quad\le \frac{||N_j||\epsilon_1}{6}\\&\quad\le \frac{\epsilon}{6}\end{aligned}\tag{34}$$

Summing (33) over m and following the similar manipulations, we have

$$\left|\frac{1}{K}\sum_{t=0}^{K-1}\left[\sum_m \mathbb{E}\left\{\mu 1^f_{m,i,j}(t)\right\}-\sum_m d^f_{m,i,j}\right]\right|\le\frac{\epsilon}{6}\tag{35}$$

Note that the input process is entrywise rate convergent to the input rate $\left[\Lambda^f_{i,j}\right]$, so we can have

$$\left|\lambda^f_{i,j}-\frac{1}{K}\sum_{t=0}^{K-1}E\left\{a^f_{i,j}(t)\right\}\right|\le\frac{\epsilon}{6}\tag{36}$$

Because the input rate is within the capacity region, we have for any $l_{i,j}$ that

$$\left(\lambda^f_{i,j}+\epsilon\right)\le\sum_k d^f_{i,j,k}-\sum_m d^f_{m,i,j}\tag{37}$$

Substituting (34), (35), and (36) into (37), we have

$$\begin{aligned}&\frac{1}{K}\sum_{t=0}^{K-1}\mathbb{E}\left\{\sum_k\mu 1^f_{i,j,k}(t)-\sum_m\mu 1^f_{m,i,j}(t)-a^f_{i,j}(t)\right\}\\&\quad\ge\frac{\epsilon}{2}\end{aligned}\tag{38}$$

Now, define the Lyapunov function as $L(\mathbf{Q}(t)) = \sum_{i,j;f:d(f)\neq i}\left(Q^f_{i,j}(t)\right)^2$. The K-step dynamics of unfinished work satisfies the queue update law, and we rewrite it with respect to $\mu 1_{i,j,k}(t)$ and $\mu 1_{m,i,j}(t)$.

$$\begin{aligned}Q^f_{i,j}(t+K)\le&\left[Q^f_{i,j}(t)-\sum_{t=0}^{K-1}\sum_k\mu 1^f_{i,j,k}(t)\right]^+\\&+\sum_{t=0}^{K-1}\sum_m\mu 1^f_{m,i,j}(t)+\sum_{t=0}^{K-1}a^f_{i,j}(t)\end{aligned}\tag{39}$$

Following the same manipulations as in Section 3.1, we have the K-slot Lyapunov drift

$$\begin{aligned}&\left(Q^f_{i,j}(t+K)\right)^2-\left(Q^f_{i,j}(t)\right)^2\\&\quad\le K^2\left[\left(\tilde{a}^f_{i,j}+\sum_m\tilde{\mu}1^f_{m,i,j}\right)^2+\left(\sum_k\tilde{\mu}1^f_{i,j,k}\right)^2\right]\\&\qquad-2KQ^f_{i,j}(t)\left(\sum_k\tilde{\mu}1^f_{i,j,k}-\sum_m\tilde{\mu}1^f_{m,i,j}-\tilde{a}^f_{i,j}\right)\\&\quad\le K^2B-2K\epsilon Q^f_{i,j}(t)\end{aligned}\tag{40}$$

where B is defined in (13) and

$$\begin{aligned}\tilde{\mu}1^f_{i,j,k}&\triangleq\frac{1}{K}\sum_{t=0}^{K-1}\mu 1^f_{i,j,k}(t)\\\tilde{\mu}1^f_{m,i,j}&\triangleq\frac{1}{K}\sum_{t=0}^{K-1}\mu 1^f_{m,i,j}(t)\\\tilde{a}^f_{i,j}&\triangleq\frac{1}{K}\sum_{t=0}^{K-1}a^f_{i,j}(t)\end{aligned}$$

By summing (40) over all $l_{i,j}$'s and taking conditional expectations yields

$$\begin{aligned}&\mathbb{E}\left\{L(\mathbf{Q}(t+K))-L(\mathbf{Q}(t))|\mathbf{Q}(t)\right\}\\&\quad\le K^2B-2K\epsilon\sum_{i,j}Q^f_{i,j}(t)\end{aligned}\tag{41}$$

then the network is stable according to Lemma 2 in [11]. □

Note that the STAT-PDPL algorithm is developed with the network statistics known to serve as a benchmark for the performance of the DRPC-PL in Section 3.

We next evaluate the performance of DRPC–PDPL through two lemmas. The first lemma compares the Lyapunov drift of the STAT-PDPL algorithm to the drift of a modified frame-based DRPC–PDPL algorithm, that is, FRAME-PDPL. The second lemma compares the drift of FRAME-PDPL to that of DRPC–PDPL.

Performance of the FRAME-PDPL algorithm

We rewrite the K-step drift bound (41) to have

$$\begin{aligned}&\mathbb{E}\{L(Q(t+K))-L(Q(t))|\mathbf{Q}(t)\}\\&\quad\le K^2B-2K[\Phi(\mathbf{Q}(t))-\beta(\mathbf{Q}(t))]\end{aligned}\tag{42}$$

Wirel. Commun. Mob. Comput. 2015; **15**:787–799 © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/wcm

where

$$\Phi(\mathbf{Q}(t)) \triangleq \frac{1}{K}\sum_{t=0}^{K-1}\sum_{i,j;f} Q_{i,j}^{f}(t)\mathbb{E}\left\{\left[\sum_{k}\mu_{i,j,k}^{f}(t) - \sum_{m}\mu_{m,i,j}^{f}(t)\right]\middle|\mathbf{Q}(t)\right\}$$

$$\beta(\mathbf{Q}(t)) \triangleq \frac{1}{K}\sum_{t=0}^{K-1}\sum_{i,j;f} Q_{i,j}^{f}(t)\mathbb{E}\left\{a_{i,j}^{f}(t)|\mathbf{Q}(t)\right\}$$

We then consider a K-slot-based modification of the DRPC–PDPL policy FRAME-PDPL, which maximizes the $\Phi(\mathbf{Q}(t))$ function over all possible control policies. The FRAME-PDPL algorithm is defined as follows: scheduling, power allocation, and routing are performed every time slot exactly as in the DRPC–PDPL algorithm, with the exception that backlog $\mathbf{Q}(t)$ updates every K slots. Specifically, for any time slot t within a K-slot frame $\{0, 1, \cdots, K-1\}$, the power is allocated to maximize $\sum_{i,j} W_{i,j}(t)$ defined in (9) evaluated at $\mathbf{Q}(t)$ subject to $P \in \mathcal{P}$. Thus, current channel state information is used every slot, and $\mu_{i,j,k}^{f}(t)$ is calculated according to the real queue status.

Lemma 2. *The control algorithm FRAME-PDPL maximizes $\Phi(\boldsymbol{Q}(t))$ over all possible power allocation, routing, and scheduling strategies. That is,*

$$\Phi^{F-PDPL}(\boldsymbol{Q}(t)) \geq \Phi^{X}(\boldsymbol{Q}(t)) \tag{43}$$

for any other strategy X, including the STAT-PDPL algorithm.

Proof. The FRAME-PDPL algorithm acts the same as the DRPC–PDPL algorithm with the exception that queue backlog is updated only on frame boundaries $t = \{0, K, 2K, \cdots\}$. Thus, for every $t \in \{0, 1, \cdots, K-1\}$, the algorithm FRAME-PDPL allocates a power matrix $P(t)$ to maximize

$$\begin{aligned}&\sum_{i,j;f} Q_{i,j}^{f}(t)\left(\sum_{k}\mu_{i,j,k}^{f}(t) - \sum_{m}\mu_{m,i,j}^{f}(t)\right)\\ &= \sum_{i,j;f}\mu_{i,j}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right)\end{aligned} \tag{44}$$

Taking conditional expectations with respect to $\mathbf{Q}(t)$ shown earlier and summing over t yield an alternative way to express $\Phi(\mathbf{Q}(t))$:

$$\Phi(\mathbf{Q}(t)) = \frac{1}{K}\sum_{t=0}^{K-1}\mathbb{E}\left\{\sum_{i,j;f}\mu_{i,j}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right)\right\} \tag{45}$$

The value of $\Phi^{X}(\mathbf{Q}(t))$ is obtained from (45) by using the $\mu_{i,j}(t)$ corresponding to some policy X. Thus, the FRAME-PDPL essentially maximizes (45) to yield $\Phi^{F-PDPL}(\mathbf{Q}(t))$ by choosing the weight for a link as (9). Taking conditional expectations and comparing $\Phi^{F-PDPL}(\mathbf{Q}(t))$ with $\Phi^{X}(\mathbf{Q}(t))$, we have

$$\Phi^{F-PDPL}(\mathbf{Q}(t)) \geq \Phi^{X}(\mathbf{Q}(t))$$

□

Next, we compare FRAME-PDPL with DRPC-PDPL.

Lemma 3. *The control policy DRPC–PDPL produces a $\Phi^{D-PDPL}(\boldsymbol{Q}(t))$ satisfying*

$$\Phi^{D-PDPL}(\boldsymbol{Q}(t)) \geq \Phi^{F-PDPL}(\boldsymbol{Q}(t)) - \frac{(K-1)|\mathcal{F}||\mathcal{L}|\tilde{B}}{2} \tag{46}$$

where

$$\tilde{B} = 2\mu_{\max}(a_{\max} + 2\mu_{\max})$$

Proof. For every time slot, DRPC–PDPL maximizes (44) over all possible control decisions. Hence,

$$\begin{aligned}&\sum_{i,j;f}\mu_{i,j}^{D-DPDL}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right)\\ &\geq \sum_{i,j;f}\mu_{i,j}^{F-DPDL}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right)\end{aligned} \tag{47}$$

Rewriting the left-hand side of (47), we have

$$\begin{aligned}&\sum_{i,j;f}\mu_{i,j}^{D-DPDL}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right) + \sum_{i,j;f}\left|\Delta_{i,j}^{f}(t)\right|2\mu_{\max}\\ &\geq \sum_{i,j;f}\mu_{i,j}^{F-DPDL}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right) \qquad (48)\\ &\geq \sum_{i,j;f}\mu_{i,j}^{F-DPDL}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right) - \sum_{i,j;f}\left|\Delta_{i,j}^{f}(t)\right|2\mu_{\max} \qquad (49)\end{aligned}$$

where $\Delta_{i,j}^{f}(t) \triangleq Q_{i,j}^{f}(t) - Q_{i,j}^{f}(t)$ and (48) follows the fact that

$$\sum_{i,j;f}\Delta_{i,j}^{f}(t)\left(Q_{i,j}^{f}(t) - Q_{j,k}^{f}(t)\right) \leq \sum_{i,j;f}\left|\Delta_{i,j}^{f}(t)\right|2\mu_{\max}$$

Summing (49) over $t \in \{0, 1, \cdots, K-1\}$ and taking conditional expectations with respect to $\mathbf{Q}(t)$ yield

$$\Phi^{D-PDPL}(Q(t)) \geq \Phi^{F-PDPL}(\mathbf{Q}(t)) - \frac{2}{K}\sum_{t=0}^{K-1}\mathbb{E}\left\{\sum_{i,j;f}\left|\Delta_{i,j}^{f}(t)\right|\right\}\mu_{\max} \qquad (50)$$

Substituting (50) into (42) yields (46). □

ACKNOWLEDGEMENTS

This work was supported in part by the US National Science Foundation under Grant CNS-0643731 and the US Office of Naval Research under Grant N000140810873.

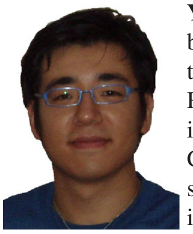
REFERENCES

1. Neely M, Modiano E, Rohrs C. Dynamic power allocation and routing for time varying wireless networks, In *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*, IEEE INFOCOM, vol.1, 2003.
2. Tassiulas L, Ephremides A. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control* 1992; **37**(12): 1936–1948.
3. Lin X, Shroff N. Joint rate control and scheduling in multihop wireless networks, In *43rd IEEE Conference on Decision and Control, 2004. CDC*, vol.2, 2004.
4. Chiang M. Balancing transport and physical layers in wireless multihop networks: jointly optimal congestion control and power control. *IEEE Journal on Selected Areas in Communications* 2005; **23**(1): 104–116.
5. Shakkottai S, Srikant R. *Network optimization and control*. Now Publishers: Hanover, USA, 2008.
6. Bui L, Srikant R, Stolyar A. Novel architectures and algorithms for delay reduction in back-pressure scheduling and routing, In *INFOCOM 2009, IEEE*, IEEE, 2009; 2936–2940.
7. Ding Z, Wu D. Sliding mode based joint congestion control and scheduling in multi-hop ad hoc networks with multi-class services, In *IEEE Globecom*, 2010.
8. Chiang M, Low S, Calderbank A, Doyle J. Layering as optimization decomposition: a mathematical theory of network architectures. *PROCEEDINGS-IEEE* 2007; **95**(1): 255–311.
9. Song Y, Zhang C, Fang Y, Niu Z. Energy-conserving scheduling in multi-hop wireless networks with time-varying channels, In *Proceedings of the 29th Conference on Information Communications*, IEEE Press, 2010; 2624–2632.
10. Eryilmaz A, Srikant R. Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control. *IEEE/ACM Transactions on Networking (TON)* 2007; **15**(6): 1333–1344.
11. Georgiadis L, Neely M, Neely M, Tassiulas L. *Resource allocation and cross layer control in wireless networks*. Now Pub, 2006.
12. Neely M. Universal scheduling for networks with arbitrary traffic, channels, and mobility, 2010. Arxiv preprint arXiv:1001.0960.
13. Stolyar A. Maximizing queueing network utility subject to stability: greedy primal-dual algorithm. *Queueing Systems* 2005; **50**(4): 401–457.
14. Gupta A, Lin X, Srikant R. Low-complexity distributed scheduling algorithms for wireless networks. *IEEE/ACM Transactions on Networking (TON)* 2009; **17**(6): 1846–1859.
15. Akyol U, Andrews M, Gupta P, Hobby J, Saniee I, Stolyar A. Joint scheduling and congestion control in mobile ad-hoc networks, In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, IEEE, 2008; 619–627.
16. Neely M. Stability and capacity regions for discrete time queueing networks, 2010. Arxiv preprint arXiv:1003.3396.
17. Neely M. Dynamic power allocation and routing for satellite and wireless networks with time varying channels, Ph.D. dissertation, Massachusetts Institute of Technology, 2003.
18. Lin X, Shroff N. The impact of imperfect scheduling on cross-layer rate control in wireless networks, In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, IEEE, vol.3, 2005; 1804–1814.
19. Warrier A, Janakiraman S, Ha S, Rhee I. Diffq: practical differential backlog congestion control for wireless networks, In *INFOCOM 2009, IEEE*, IEEE, 2009; 262–270.
20. Yi Y, Proutière A, Chiang M. Complexity in wireless scheduling: impact and tradeoffs, In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ACM, 2008; 33–42.
21. Modiano E, Shah D, Zussman G. Maximizing throughput in wireless networks via gossiping, In *ACM SIGMETRICS Performance Evaluation Review*, ACM, vol.34, no.1, 2006; 27–38.

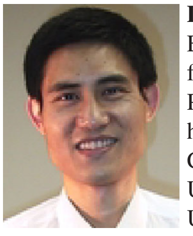
22. Jiang L, Walrand J. A distributed csma algorithm for throughput and utility maximization in wireless networks. *IEEE/ACM Transactions on Networking (TON)* 2010; **18**(3): 960–972.
23. Joo C. A local greedy scheduling scheme with provable performance guarantee, In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ACM, 2008; 111–120.
24. Brzezinski A, Zussman G, Modiano E. Distributed throughput maximization in wireless mesh networks via pre-partitioning. *Networking, IEEE/ACM Transactions on* 2008; **16**(6): 1406–1419.
25. Joo C, Lin X, Shroff N. Understanding the capacity region of the greedy maximal scheduling algorithm in multihop wireless networks. *IEEE/ACM Transactions on Networking (TON)* 2009; **17**(4): 1132–1145.
26. Stolyar A. Dynamic distributed scheduling in random access networks. *Journal of Applied Probability* 2008; **45**(2): 297–313.

AUTHORS' BIOGRAPHIES

Zongrui Ding received her PhD degree in Electrical and Computer Engineering from the University of Florida, Gainesville in 2012. Her research interests include the stochastic optimization in networking and the Smart Grid. She joined Interdigital Inc., San Deigo, CA, USA in 2012, where she is currently a senior engineer working on Machine to Machine communications. She is a recipient of the Best Paper Award from IEEE GLOBECOM 2011, Houston, TX, USA.



Yang Song is a Research Staff Member at the IBM Almaden Research Center, San Jose, CA, USA. Before IBM Research, he received his PhD degree in the Department of Electrical and Computer Engineering at the University of Florida in 2010. His research interests include data center management and optimization, scalable machine learning for big data analytics, and communication networks.



Dapeng Wu received his PhD in Electrical and Computer Engineering from Carnegie Mellon University, Pittsburgh, PA in 2003. Currently, he is a professor of Electrical and Computer Engineering Department at University of Florida, Gainesville, FL, USA. His research interests include wireless communications, video coding, information theory, and machine learning. He is an IEEE fellow since 2013.

Yuguang Fang received a PhD degree in Systems Engineering from Case Western Reserve University in January 1994 and a Ph.D. degree in Electrical Engineering from Boston University in May 1997. He is currently a professor in the Department of Electrical and Computer Engineering at University of Florida, Gainesville, FL, USA. His current research interests include cybersecurity, cognitive radio networks, wireless networks and mobile computing. He is an IEEE fellow since 2008.