Distributed CSMA Algorithms for Link Scheduling in Multi-hop MIMO Networks under SINR Model

Dajun Qian[*], Dong Zheng[†], Junshan Zhang[*], Ness Shroff[‡] and Changhee Joo[§]

[*] School of ECEE, Arizona State University, Tempe, AZ, USA

[†]Broadcom Corp., Sunnyvale, CA, USA

[‡]Department of ECE, Ohio State University, Columbus, OH, USA

[§] School of ECE, UNIST, Ulsan, Korea

Abstract—In this paper, we study distributed scheduling in multi-hop MIMO networks. We first develop a "MIMO-pipe" model that provides the upper layers a set of rates and SINR requirements that capture the rate-reliability tradeoff in MIMO communications. The main thrust of this study is then dedicated to developing distributed CSMA algorithms for MIMO-pipe scheduling under the SINR interference model. We choose the SINR model over the extensively studied protocol-based interference models because it more naturally captures the impact of interference in wireless networks. The coupling among the links caused by the interference under the SINR model makes the problem of devising distributed scheduling algorithms very challenging. To that end, we explore the CSMA algorithms for MIMO-pipe scheduling from two perspectives. We start with an idealized continuous-time CSMA network, where control messages can be exchanged in a collision-free manner; and devise a CSMA-based link scheduling algorithm that can achieve throughput-optimality under the SINR model. Next, we consider a discrete-time CSMA network, where the message exchanges suffer from collisions. For this more challenging case, we develop a "conservative" scheduling algorithm by imposing a more stringent SINR constraint on the MIMO-pipe model. We show that the proposed conservative scheduling achieves an efficiency ratio bounded from below.

Index Terms—MIMO, scheduling, SINR interference model, CSMA, multi-hop networks.

I. INTRODUCTION

We study distributed scheduling in multi-hop networks with MIMO links, where each node is equipped with an antenna array. There has been a tremendous body of work on the multiple-input multiple-output (MIMO) technology from a PHY-layer communication perspective. For single-user wireless channels, it has been shown that using the MIMO technique can lead to dramatic improvements on capacity and link reliability [2], [3]. Recent studies have explored the fundamental tradeoffs and relations between the different gains in single-user MIMO systems [4]. In contrast to the extensive studies on the single-user settings, however, there has been little work on exploring multi-hop MIMO networks. Obtaining a rigorous understanding of the tradeoffs between the possible MIMO gains therein has remained a largely open problem.

Leveraging MIMO gains in a multi-hop network is intimately related to link scheduling, because the intrinsic rate-reliability tradeoff hinges heavily on the SINR values of the coupled MIMO links due to mutual interference (see, e.g., [5], [6]). In this study, we will take two steps to explore the scheduling in multi-hop MIMO networks:

- Step 1: Develop a link abstraction that can capture the rate-reliability tradeoff in MIMO communications;
- Step 2: Pursue a deep understanding of throughput-optimal scheduling under the SINR model[1], and use this as a basis for studying distributed MIMO link scheduling.

More specifically, to facilitate the development of low-complexity scheduling, we propose an appropriate "MIMO-pipe" model that provides an abstraction of the rate-reliability tradeoff in MIMO communications. Clearly, choosing the highest rate for a given MIMO link may not be optimal for the network, since it may prevent other links from being simultaneously active and degrade the overall network throughput. Instead, we model a MIMO-link using a set of achievable "configurations," under which a link can transmit multiple data streams at the same time; and different configurations have different SINR requirements for reliable communication. Each MIMO link can select one among a set of configurations according to its SINR requirement. Observe that the MIMO communications expands the space of possible network states, and if not designed intelligently it would further complicate scheduling schemes that are already very complex [7].

Recently, low-complexity scheduling schemes based on carrier sense multiple access (CSMA) have been proposed (see [8], [9], [10], [11], [12] and the references therein). In these CSMA algorithms, nodes first sense the channel activity, and only when the channel is sensed to be idle can the nodes continue with data transmissions. When the channel is detected busy, the nodes need to backoff for a random amount of time before reattempting the transmission. Due to its simplicity, CSMA and its variants have been widely opted in practical MAC protocols (e.g., IEEE 802.11). It has been shown in [8], [11] that under an idealized CSMA model, where the backoff time is continuous and collisions never happen, the network state dynamics can be captured by a continuous-time

---

[1]A scheduling algorithm is said to be throughput-optimal if it can achieve every point in the capacity region [7].

Markov Chain (CTMC)[2]. The throughput-optimal scheduling algorithm is developed based on the Markov chain modeling of the CSMA network. However, in practical scenarios, collisions could not be avoided completely. Recent work [9] has proposed a discrete-time CSMA scheduling algorithm where the evolution of network states follows a discrete-time Markov Chain (DTMC). A common theme in these works is to capture the network dynamics by a time-reversible Markov chain, and to drive, via adaptive scheduling, the corresponding stationary distribution to achieve the throughput-optimality. Note that all the algorithms noted above have been developed under protocol-based interference models where two links cannot transmit simultaneously if one link is within a certain range (or hops) of the other link.

In this paper, we study CSMA-based scheduling in a multi-hop MIMO network, under the SINR interference model. Different from protocol-based models, the rate-reliability tradeoff of a MIMO link hinges heavily on its SINR value. More specifically, under the SINR model, a link transmission is said to be successful if its SINR value is greater than a pre-determined threshold for a given rate. A critical observation is that *a successful link transmission under the SINR model depends on its aggregated interference level, and not on the activity of a particular link.* As we will elaborate in Section II, the SINR model induces intrinsic global coupling, making it challenging to develop distributed scheduling schemes. In general, it has been largely open on how to design distributed scheduling algorithms under the SINR model (even for the SISO case), and a primary goal of this study is to take some steps in this direction.

We will explore the CSMA algorithms for MIMO-pipe scheduling, for both continuous-time and discrete-time networks. We summarize below the main contributions in this study.

1) We take a bottom-up approach to develop the MIMO-pipe model, which consists of multiple stream configurations, each with a feasible rate and the corresponding S-INR requirement. Using this model, the tradeoff between diversity and multiplexing of MIMO communications can be captured by the selection of MIMO configurations. In a nutshell, we treat each configuration as a virtual link with a fixed rate and the corresponding SINR requirement, and each MIMO link is mapped to multiple virtual links with different rates and SINR requirements.

2) We consider the CSMA algorithms for MIMO-pipe scheduling in a continuous-time network. To tackle the intrinsic challenge in the "aggregate interference effect" under the SINR model, we propose to separate the control channel for signal exchanges from that for data transmissions. Assuming that there is no collision of control signals, we show that the network dynamics can be captured by a continuous-time Markov chain. Further, we characterize the optimal backoff parameters of different stream configurations, for throughput-optimal scheduling.

3) We then focus on the CSMA algorithms for MIMO-pipe scheduling in a discrete-time network, where control signals may "collide." To tackle the collisions and the link coupling problem under the SINR model, we devise a distributed scheduling algorithm using a "conservative" strategy. Specifically, we impose a more stringent SINR constraint to ensure that the transitions of the network states only happen in the feasible state region, at the cost of reduced network throughput. We then systematically quantify the performance gap between the optimal scheduling and the conservative scheduling approach. We show that this conservative distributed scheduling can achieve an efficiency ratio bounded below.

II. SYSTEM SETUP AND RELATED WORK

Consider a multi-hop MIMO network consisting of K links, where each link employs N_t transmit antennas and N_r receive antennas. The received signal at the i-th receiver can be given by

$$\mathbf{y}_i = \sqrt{\frac{P}{N_t d_{ii}^{\alpha}}} \mathbf{H}_{ii}\mathbf{s}_i + \sum_{j \neq i} \sqrt{\frac{P}{N_t d_{ji}^{\alpha}}} \mathbf{H}_{ji}\mathbf{s}_j + \mathbf{n}_i, \qquad (1)$$

where P is the total transmission power at each transmitter; $\mathbf{s}_i$ is the $N_t \times 1$ transmitted signal from the i-th transmitter, with normalized power at each antenna array to be 1, in each symbol period; α is the path loss exponent; d_{ji} is the distance from the j-th transmitter to the i-th receiver. We consider a frequency flat fading MIMO channel [3] such that $\mathbf{H}_{ji}$ is the $N_r \times N_t$ channel matrix between the j-th transmitter to the i-th receiver, where the entries of each matrix are i.i.d. complex circular symmetric Gaussian with unit variance. Furthermore, the entries of $\mathbf{H}_{ji}$ are independent from those of $\mathbf{H}_{ji'}$ if $i \neq i'$; $\mathbf{n}_i$ is the additive White Gaussian noise with $\sigma^2 = E[||\mathbf{n}_i^2||]/N_r$.

The first term in (1) is the desired data signal for link i, while the last two terms are co-channel interference and noise, respectively. As is standard, we assume that the channel matrix $\mathbf{H}_{ii}$ is known at the receiver but unknown at the transmitter of link i (CSI at the receiver) [13]. Moreover, in practical systems, it is difficult, if not impossible, to obtain the MIMO channel matrices $\{\mathbf{H}_{ji}, j \neq i\}$ from the interferers, simply because the signals are not intended for the desired link and it is infeasible to estimate and track these complex matrices. Based on the above signal model, it is clear that unlike single-user MIMO systems, multi-hop networks are interference-limited, and MIMO communications are intimately tied to the SINR values that are coupled across the links.

As in [13], let I_i denote average power level of interference-plus-noise at the receiver of link i, i.e.,

$$I_i = \sum_{j \neq i} \frac{P}{N_t d_{ji}^{\alpha}} \frac{E[\text{Tr}\{\mathbf{H}_{ji}\mathbf{H}_{ji}^H\}]}{N_r} + \sigma^2, \qquad (2)$$

and let SINR_i denote the SINR at the receiver of link i, i.e.,

$$\text{SINR}_i = \frac{P d_{ii}^{-\alpha}}{\sum_{j \neq i} P d_{ji}^{-\alpha} \frac{E[\text{Tr}\{\mathbf{H}_{ji}\mathbf{H}_{ji}^H\}]}{N_t N_r} + \sigma^2}. \qquad (3)$$

---

[2]Strictly speaking, the algorithms in [8], [9] are CSMA/CA. We use the term CSMA to refer to a class of algorithms based on the CSMA mechanism.

[3]As in [13], shadow fading is not considered in this channel model.

Since the entries of $\mathbf{H}_{ji}$ are identically distributed with unit power, we have $E[\text{Tr}\{\mathbf{H}_{ji}\mathbf{H}_{ji}^H\}] = N_t N_r$. Then, the SINR value at i-th link receiver can be given by

$$\text{SINR}_i = \frac{P d_{ii}^{-\alpha}}{\sum_{j\neq i} P d_{ji}^{-\alpha} + \sigma^2}. \qquad (4)$$

The SINR value plays a critical role in link scheduling.

A. Feasible States and Capacity Region in a MIMO Network

Throughout the paper, we say that two active links can coexist if they can make successful transmissions at the same time. An interference model specifies the link coexistence constraint. We say that the network is in a *feasible state* if the set of active links satisfy the *coexistence constraint* of the interference model. In a network with K links, we use a binary vector $\mathbf{x}^i = \{0,1\}^K$ to describe a feasible state. We define that $x_l^i = 1$, if link l is active in state i; $x_l^i = 0$ otherwise. With some abuse of notation, we also treat $\mathbf{x}^i$ as the set of active links in state i, i.e., $l \in \mathbf{x}^i$ if $x_l^i = 1$. In SISO networks, it suffices to use a binary vector $\mathbf{x}$ to represent the data rate of each link, if each link transmits at unit rate [8], [9]. In contrast, each MIMO link has multiple stream configurations with different transmission rates. Hence, to describe a feasible state in a MIMO network, we also need to specify the configuration and the corresponding transmission rate of each active link. Without loss of generality, we consider a MIMO network with K links, where each link has J configurations. We use $\mathbf{z}^i = (z_1^i, z_2^i, ..., z_K^i)$ to denote the configuration of each link at feasible state i, where $z_l^i \in [1...J]$ indicates the configuration of link l. We also use $\mathbf{c}^i = (c_1^i, c_2^i, ..., c_K^i)$ to denote the data rates, where c_l^i is the data rate at link l at state i. Furthermore, we define $\Theta(\cdot)$ as the mapping from the configuration index to the corresponding normalized transmission rate, i.e., $c_l^i = \Theta(z_l^i)$. Finally, we set $c_l^i = 0$ and $z_l^i = 0$ if link l is not active at state i.

Let $\mathcal{S}$ be the set of rate vectors corresponding to the feasible states of a MIMO network. By definition [7], the capacity region Λ is the convex hull of the vectors in $\mathcal{S}$. Assume that the traffic load at link l is represented by the normalized arrival rate $\lambda_l \geq 0$. The scheduling algorithm is said to be throughput-optimal if it can keep the network stable at any arrival rate vector $\lambda = (\lambda_l, \lambda_2, ..., \lambda_K)$ within the capacity region Λ [7].

B. SINR Model versus Protocol Model

Clearly, different interference models yield different link coexistence constraints and hence different sets of feasible states. Roughly speaking, existing interference models can be classified into two categories: the protocol model and the SINR model [14]. Under the protocol model, the transmission of link l is deemed successful if no other links within a certain transmission range are active. Therefore, the coexistence relationship between two links is mainly determined by the geometry, and hence is "static" and "binary." Due to its simplicity, the protocol model has been widely used.

In contrast, under the SINR model, the coexistence relationship is neither static nor binary, and the success of a transmission depends on its own channel condition and the

level of the aggregated interference. Specifically, a transmission of a link is said to be successful if its SINR value (4) is greater than a pre-determined threshold for a given rate. The SINR model, built upon recent advances in PHY-layer communication theory, opens a new avenue for more efficient resource allocation in wireless networks.

As noted before, one significant challenge under the SINR model is that multiple links can transmit successfully through a common channel, even if they observe some interference signal from each other, which is drastically different from that under the protocol model. Furthermore, link relationship is a function of distance to the neighboring links and their status that may change over time. Therefore, the link coexistence relationship under the SINR model is "multi-lateral" and "dynamic." As a result, link scheduling under the SINR model is much more complicated.

In principle, every link in the network can contribute interference to an active receiver under the SINR model. However, when the links are sparsely located and the interference power level decreases over distance due to the free space path loss as in [15] and [16], it is reasonable to assume the aggregated interference from the transmitters beyond certain distance can be upper bounded by a threshold [17]. Specifically, we define a "close-in" radius for each link l such that the aggregated interference power to l from the transmitters beyond the close-in range is no more than a given parameter σ_{int}^2. Denote $N(l)$ as the set of links whose transmitters are in the close-in range of link l, called *interfering links* of link l and $N(l)^c$ as the set of links whose transmitters are outside the close-in range of link l. It follows that $\sum_{k \in N(l)^c} P d_{kl}^{-\alpha} < \sigma_{\text{int}}^2$. Based on σ_{int}, the close-in range of each link can be obtained in an initialization stage before link scheduling, where each link informs its incurring interference power level to neighbors by broadcasting a dummy packet sequentially. Next, each link l ranks its neighboring links in an ascending order based on their interference. A neighboring link k (staring from the link incurring the lowest interference to the highest) is deemed to be outside the close-in range of l as long as the aggregated inference from the links beyond the close-in radius and link k is lower than σ_{int}^2.

For ease of exposition, we approximately treat the aggregated interference from active links in $N(l)^c$ as white noise with power σ_{int}^2. By doing so, we define the following "nominal" SINR constraint, where link l can successfully transmit if the following condition holds:

$$\text{SINR}_l = \frac{P d_{ll}^{-\alpha}}{I_l^{in} + \sigma^2 + \sigma_{\text{int}}^2} \geq \beta_l \qquad (5)$$

where I_l^{in} is the aggregated interference from the active links in $N(l)$; σ^2 is the power of Gaussian noise; β_l is the threshold of successful transmission. In the following study, unless otherwise specified, the SINR model is defined based on the nominal SINR constraint in (5) [4].

---

[4]In Section V, we also defined a conservative SINR constraint that is more stringent than the nominal SINR constraint.

TABLE I
PARAMETERS IN CSMA-BASED ALGORITHM (AT MIMO LINK l)

Continuous time case	
R_{l_v}	backoff rate of configuration v at link l
r_{l_v}	$r_{l_v} = \log(R_{l_v})$
Discrete time case	
p_{l_v}	link activation probability of configuration v at link l
$\bar{p}_{l_v}$	$\bar{p}_{l_v} = 1 - p_{l_v}$

C. Review: CSMA Scheduling under Protocol Model

We provide below a brief review of [8], [9], which are perhaps the most related works to our study here.

Under the protocol model, an "idealized" CSMA scheduling algorithm is proposed in [8] for a continuous-time network. It is assumed that random backoff time and data transmission time follow continuous distributions. It also takes the assumption that the range of carrier-sensing is large enough and signal propagation delay is zero, which remove potential hidden terminal problem (see [18] for further discussions on hidden terminal problems). Therefore, the probability for two conflicting links to start transmission at the same time is 0 and the collisions can be ignored. Under these assumptions, the state transitions of the CSMA network can be modeled as a continuous-time Markov chain, where transitions only occur between the feasible states that differ from each other by only one link status. It follows that the stationary distribution of feasible states $\mathbf{x}^i$ can be characterized by

$$p(\mathbf{x}^i) = \frac{1}{C} \prod_{l \in \mathbf{x}^i} R_l, \tag{6}$$

where R_l is defined as backoff rate and C is the normalization term satisfying $\sum_i p(\mathbf{x}^i) = 1$. In [9], the idea has been extended to a time-slotted system, where simultaneous transmissions in a time slot may collide. It has been shown that the network states can be modeled as a discrete-time Markov chain, and the corresponding stationary distribution can also be written in a product-form:

$$p(\mathbf{x}^i) = \frac{1}{C} \prod_{l \in \mathbf{x}^i} \frac{p_l}{\bar{p}_l}, \tag{7}$$

where p_l is defined as link activation probability in [9] and $\bar{p}_l = 1 - p_l$. Furthermore, it has been shown that adaptive CSMA scheduling algorithms that adjust link parameter based on local queue information can achieve throughput-optimality. We extend the results to more general MIMO scenarios. To this end, we define similar parameters for each MIMO configuration v of link l as shown in Table I.

III. MIMO-PIPE MODELING: RATES, SINR, AND INTERFERENCE TOLERANCE LEVELS

A first key step in our study on MIMO scheduling is to develop a PHY-based tractable model that captures the rate-reliability tradeoff for a single MIMO link, which we call the "MIMO-pipe" model.

In MIMO networks, every MIMO link can offer stream multiplexing by opening up multiple spatial data streams in the same frequency channel, and achieve spatial multiplexing gain. The number of data streams depends on the stream configuration of the link. Given the number of antennas and the total transmission power at each node,[5] we assume that the transmission power is equally split among the transmit antennas. Clearly, the greater the number of data streams there are at each MIMO link, the lower the reliability and the interference tolerance capability per stream. Accordingly, the required average SINR per receive antenna [13], called SINR requirement, is more stringent. In the following, we will elaborate the tradeoff between stream multiplexing gain and interference tolerance capability (determined by the corresponding SINR requirements).

A. MIMO Configurations and SINR Requirements

Without loss of generality, suppose that each link has J configurations, and for configuration v, $v \in [1...J]$, there are $\Theta(v)$ date streams. For simplicity, we set the transmission rate of each stream to be the same, denoted as $\mathcal{R}_s$, and hence the link rate is $\mathcal{R}_s\Theta(v)$ at configuration v. Without loss of generality, we assume the stream rate $\mathcal{R}_s$ is fixed at 1 in this study. The SINR requirement of stream r at configuration v, can be in general given as

$$\beta_{vr} = f(v, r, H, P_e), \tag{8}$$

which depends on the channel matrix H and the average BER requirement P_e for reliable communication. The function f depends on the physical-layer techniques, such as coding and modulation.

Due to self-interference cross data streams on the same MIMO link, the SINR values of different streams can be different. To guarantee the decodability of all data streams, the SINR requirement of configuration v should be set as $\beta_v = \max\{\beta_{v1}, \beta_{v2}, ..., \beta_{v\Theta(v)}\}$, i.e., the highest SINR requirement corresponding to the bottleneck stream. Such bottleneck stream usually has the least number of transmit antennas. Therefore, it is reasonable to consider a subset of configurations in which transmit antennas are equally divided for each stream. Clearly, the collection of configurations for a MIMO link with N_t transmit antennas corresponds to an integer set $\{n_v \,|n_v$ is a divisor of N_t, $v = 1, 2, 3..., J\}$ and the number of configurations equals the number of divisors of N_t. Specifically, the configuration v has n_v data streams and each stream has $\frac{N_t}{n_v}$ transmit antennas. For example, for the 4×4 MIMO link, we consider three configurations: 1-transmit antenna per stream, 2-transmit antennas per stream, and 4-transmit antennas per stream, with data rates $4\mathcal{R}_s, 2\mathcal{R}_s, \mathcal{R}_s$, and SINR requirements $\beta_1 > \beta_2 > \beta_3$, respectively.

B. Interference Tolerance

Under the SINR model, the successful transmission depends on the current SINR value at the MIMO receiver. By definition of the nominal SINR constraint in (5), we assume that the MIMO link l can successfully transmit with v-th configuration at time t if the following condition holds:

$$\mathrm{SINR}_l(t) = \frac{Pd_{ll}^{-\alpha}}{I_l^{in}(t) + \sigma^2 + \sigma_{\mathrm{int}}^2} \geq \beta_{lv}, \tag{9}$$

[5]In this study, the transmission power is assumed to be fixed. Dynamic power control is beyond the scope of this paper.

where $I_l^{in}(t)$ is the aggregated interference from the active links in $N(l)$; β_{lv} is the SINR requirement of v-th configuration at link l; other items follow the same definitions as in (5). Given a link activation setting, we define the *interference tolerance level* as the interference power that the receiver can further tolerate without violating the SINR requirement. By (9), for the v-th configuration of link l, its interference tolerance at time t can be given by:

$$T_{l_v}(t) = \frac{P d_{ll}^{-\alpha}}{\beta_{lv}} - I_l^{in}(t) - \sigma^2 - \sigma_{int}^2. \tag{10}$$

Clearly, the interference tolerance can be calculated by the receiver based on the interference power level $I_l^{in}(t)$ that the receiver currently experiences. Note that the interference tolerance level depends on the aggregated interference from the neighbors, and will change dynamically over time according to the on/off status of nearby links.

Fig. 1 illustrates the relationship between interference tolerance (reliability) and rate of a single 4×4 MIMO link. *We emphasize that the stream configurations here correspond to a few points on the rate-reliability tradeoff curve, and that the rates are set to multiplications of the basic rate $\mathcal{R}_s$ to reflect the multiplexing gain. In general, one can find multiple pairs of (rate, interference tolerance level) of a MIMO link.*
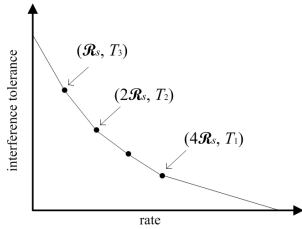

Fig. 1. Rate-reliability tradeoff for a MIMO link with 4×4 antennas.

Scheduling problem under the MIMO-pipe model is to decide which link to transmit and which configuration to use in data transmission. Clearly, the configuration with more data streams (higher multiplexing degree) can achieve a higher data rate, but in the meanwhile, fewer transmit antennas are assigned to each stream which results in a lower interference tolerance level. Once a link chooses a higher rate configuration, it would not be able to co-exist with many nearby links. Hence, there exists an intrinsic tradeoff between the throughput for a single link and overall network.

IV. CSMA Algorithm for MIMO-pipe Scheduling: A Continuous-time Model

In this section, we study the CSMA algorithm for a continuous-time network, under the SINR model. For ease of exposition, we first focus on the distributed scheduling for SISO case and further generalize our study to the MIMO-pipe model.

A. SINR-aware Channel Probing: A Dual Band Approach

We aim to develop the scheduling algorithm under the SINR model by utilizing the Markov chain structure of a CSMA network, where the network states evolve as a continuous-time Markov chain and each state in the Markov chain corresponds to a feasible link activation. According to [8], a CSMA network can be described by a continuous-time Markov chain when it satisfies the following requirements:

(R1) Network state transitions only occur between the feasible states that differ from each other by only one link status.
(R2) For each link, the backoff time and the data transmission time are both exponentially distributed.

To meet the first requirement, a key challenge is to ensure that the CSMA network always stays in a feasible state under the SINR model. In other words, the scheduling algorithm can guarantee the coexistence of active links under the SINR model. Specifically, when a link is activated, it should tolerate the aggregated interference from other active links, and meanwhile, its incurring interference would not violate the SINR requirements of other on-going transmissions.

To tackle this issue, we propose the following "SINR-aware" channel probing approach. This mechanism enables each link to assess its coexistence relationship with other active links under the SINR model by utilizing carrier-sensing and control messages exchange. The key idea is that each receiver keeps sensing the channel and broadcasts its interference tolerance level to the neighbors. With that information, when an inactive link, say k, is about to be active, the transmitter of link k can decide whether its potential transmission will violate the SINR requirements of any ongoing transmission. Simply put, for each active link l, the receiver calculates its interference tolerance $T_l(t)$ according to (10). Then, it broadcasts $T_l(t)$ in the control message to its nearby links, i.e., to any link k with $k \in N(l)$. Based on the interference power information acquired during the initialization stage (see Section II-B), the transmitter of link k can estimate how much interference it would incur to other receivers. By doing so, link k can judge its coexistence feasibility with the existing active links and avoid possible violations to the nominal SINR requirements.

To ensure that the data transmission would not collide with the control signal, we consider a dual-band approach where we separate the frequency band into data channel and control channel for each signal. By doing so, a receiver can broadcast control message and receive data packets at the same time. From the idealized CSMA assumption as in [8], the transmissions of control signal can be completed instantaneously (i.e., zero propagation delay) and do not collide in the control channel. The details of the channel probing mechanism are summarized in Algorithm 1. Note that the channel probing is a sub-step of CSMA-based scheduling that will be explained in Algorithm 2.

Note that the continuous backoff time ensures that no more than one link decides to transmit at the same instance. Therefore, only one link can change its state during each transition. By using the proposed SINR-aware channel probing approach, the state transitions of the CSMA network only take place among the feasible states under the SINR model. Furthermore, both the backoff time and data transmission time can be designed to follow exponential distributions, which will be shown in the following section. Building on these, the CSMA network can satisfy the requirements **R1** and **R2**, and

its dynamics can be captured by a continuous-time Markov chain.

Algorithm 1 SINR-aware channel probing (at link l)

At the receiver
- Idle period
  - The receiver keeps sensing the data channel and updating its current $T_l(t)$ by (10).
- Data transmission period
  - When link l starts transmission, its receiver broadcasts $T_l(t)$ through the control channel.
  - When receiver senses "new" interference during data receiving, $T_l(t)$ will be updated and broadcasted again through the control channel.
  - When link l finishes transmission, its receiver broadcasts $T_l(t) = \infty$.

At the transmitter
- Keeps overhearing the control messages from the control channel.
- Once receiving a control message from the receiver of link k, the transmitter can estimate its possible interference incurring to k based on the interference information acquired at initialization stage.

Check the link coexistence requirements

At time t, link l can coexist with nearby active links without violations to the SINR requirements (assuming other existing active links can also coexist) under the following two necessary conditions:
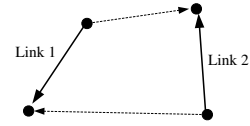1) $T_l(t) > 0$.
2) For any active link $k \in N(l)$, the interference from link l to k is no great than $T_k(t)$.

(a) An example network with two 4×4 MIMO links

(b) State transition graph for the continuous-time Markov chain associated with the network in (a)

Fig. 2. MIMO network with virtual links and the corresponding Markov chain model.

TABLE II
FEASIBLE STATE

feasible state	A	B	C	D	E	F	G	H	I
z	0,0	0,1	1,0	0,2	2,0	3,0	0,3	1,1	1,2
c	0,0	0,1	1,0	0,2	2,0	4,0	0,4	1,1	1,2

Note: The link configurations and link rates for each feasible state are represented by $\mathbf{z} = (z_1, z_2)$ and $\mathbf{c} = (c_1, c_2)$ as defined in Section II-A. The feasible states in the table are given for illustration purpose only.

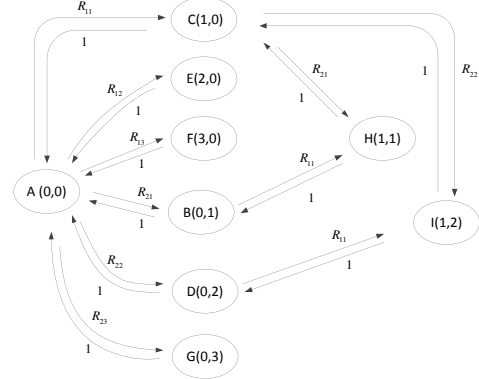## B. CSMA Algorithm for MIMO-pipe Scheduling

We next devise the CSMA scheduling algorithm for MIMO links. Recall that under the MIMO-pipe model, each link has multiple stream configurations, and can choose a feasible configuration as long as it satisfies the SINR requirement. Therefore, the MIMO network will have a much larger set of feasible states compared to the SISO case. We develop CSMA scheduling for MIMO-pipe links such that the network state transitions still can be captured by a continuous-time Markov chain, using our SINR-aware channel probing.

We model each MIMO configuration as a "virtual link," with separate mean backoff time and interference tolerance. Specifically, letting l_v denote a virtual link with configuration v at link l, the backoff time of l_v is exponentially distributed with mean $1/R_{l_v}$, where R_{l_v} is called "backoff rate." With some abuse of notation, we treat $\mathbf{z}^i$ as the set of active virtual links at state i. At state i, if link l transmits at stream configuration v, then $l_v \in \mathbf{z}^i$ and $z_l^i = v$.

Along the same line as in conventional CSMA, each virtual link contends for transmission using the backoff timer. However, the timer freezes when the virtual link cannot make transmission because it would violate any existing transmission of nearby links. This feasibility test can be done with the information obtained from the SINR-aware channel probing. When the virtual link starts data transmission, it should broadcast its interference tolerance level though the control channel. The details of the CSMA algorithm for MIMO link scheduling are summarized in Algorithm 2.

With the help of the SINR-aware channel probing, the MIMO network remains in feasible states and can be modeled as a Markov chain as in the SISO case. To get a more concrete sense, we consider an example network with two 4×4 MIMO links in Fig. 2(a). The feasible states in Table II are given for illustration purpose only. The network states transition can be captured by a continuous-time Markov chain whose state transition graph is depicted in Fig. 2(b), where each cycle corresponds to a feasible state (z_1, z_2) and z_1 and z_2 represent the configuration of link 1 and link 2, respectively. In the state transition graph in Fig. 2(b), we denote the transition between two states by a directional line with the transition rate. For any two connecting states, the left state transits to the right state with a rate of R_{l_v}, and the right state transits to the left state with a rate of 1. The stationary distribution of the feasible

state $\mathbf{z}^i$ can be obtained as

$$p(\mathbf{z}^i) = \frac{1}{C} \prod_{l_v \in \mathbf{z}^i} R_{l_v}, \qquad (11)$$

where C is the normalization term. For each link l, let R_l^i denote the backoff rate of the active virtual link at state i, i.e.,

$$R_l^i = \begin{cases} R_{l_v} & \text{if } z_l^i = v \\ 1 & \text{if } z_l^i = 0 \text{ (i.e., link } l \text{ is inactive).} \end{cases} \qquad (12)$$

Then, we can rewrite (11) as:

$$p(\mathbf{z}^i) = \frac{\exp(\sum_{l=1}^K r_l^i)}{\sum_j \exp(\sum_{l=1}^K r_l^j)}, \qquad (13)$$

where $r_l^i = \log(R_l^i)$ for each virtual link. The normalized throughput of link l is given by

$$\theta_l = \sum_i \Theta(z_l^i) \cdot p(\mathbf{z}^i). \qquad (14)$$

Algorithm 2 Continuous-time CSMA scheduling under the MIMO-pipe model (at link l)

Transmission initiation
- For each virtual link l_v, $v \in [1...J]$, the transmitter checks its coexistence with the active nearby links using **Algorithm 1**.
- When a virtual link l_v satisfies the link coexistence constraint, it waits for a period of time (backoff) that is exponentially distributed with mean $1/R_{l_v}$.

Random backoff
When a nearby link begins transmission, l_v updates its interference tolerance level and checks the link coexistence constraint using **Algorithm 1**. If l_v can no longer coexist with the current active links, l_v would suspend its backoff and resume it after the coexistence constraint is satisfied, i.e., after some nearby active link finishes its transmission.

Data transmission
- Once the back-off time of virtual link l_v expires, link l would launch the data transmission at the stream configuration v. The transmission time is exponentially distributed with mean 1.
- Other virtual links of link l suspend the backoff and would resume it until link l finishes data transmission.

The next key step is to optimize the backoff time of each virtual link, so that the corresponding adaptive CSMA algorithm can converge to the throughput-optimal one. A central problem is how to use local information to adapt the backoff time so as to meet the throughput requirement of each link, i.e., $\theta_l \geq \lambda_l$. Along the lines in [8], we have the following result.

Lemma 4.1: Under the time-scale-separation assumption [8] [6], the CSMA algorithm for MIMO scheduling can achieve any throughput λ in the capacity region, by adjusting the backoff rate of each virtual link as follows:

---

[6] As shown in [19], it is possible to achieve the throughput-optimality under certain conditions without the time-scale-separation assumption.

For link l,

$$y_l(t+1) = [y_l(t) + \xi(\lambda_l - \theta_l(t))]^+,$$

where y_l is shown to be proportional to the queue length at link l [8], and $\xi > 0$ denotes a small constant (step size). Each virtual link adapts its backoff time according to

$$R_{l_v} = \exp(y_l\Theta(v)), \qquad v \in [1...J],$$

where $\Theta(v)$ is the data rate of configuration v.

The proof of Lemma 4.1 is relegated to Appendix A.

In the idealized CSMA network, it is assumed that control messages have zero propagation delay, and would never collide. The proposed channel probing approach is based on such "collision-free" assumption. However, it would not work very well in a more realistic discrete-time network where collisions can happen.

V. CSMA ALGORITHM FOR MIMO-PIPE SCHEDULING: A DISCRETE-TIME MODEL

In the following, we extend our distributed MIMO-pipe scheduling approach to a synchronized time-slotted network.

A. CSMA Algorithm for Conservative MIMO-pipe Scheduling

We study the CSMA algorithms for link scheduling under the SINR model in a discrete-time network, where the time is slotted. At each time slot t, the scheduling algorithm decides a transmission schedule $\mathbf{z}(t)$, i.e., the set of links that transmit simultaneously at t.

In [9], the authors develop a CSMA scheduling scheme for the protocol model, which operates as follows: let $\mathbf{z}(t-1)$ denote the transmission schedule in time slot $t-1$. At the beginning of time slot t, a feasible schedule denoted by *decision schedule* $\mathcal{M}(t)$ is calculated. A subset of links in $\mathcal{M}(t)$ is discarded if they interfere with any link in $\mathbf{z}(t-1)$. Each link in the remaining $\mathcal{M}(t)$ independently determines whether it will be active in time slot t or not using its own link information, and all the other links remain in the same state as in time slot $t-1$. Finally, links in $\mathbf{z}(t)$ transmit data packets in time slot t. It is required all the links in $\mathcal{M}(t) \oplus \mathbf{z}(t-1)$ can coexist satisfying the underlying interference constraints. Such requirement is not difficult to be satisfied under the protocol model, due to the static link coexistence relationship [9]. However, under the SINR model, the coexistence relationship between two links becomes dynamic and depends on the states of the neighboring links within their close-in radius. Therefore, a key challenge here is to ensure the coexistence of the links in $\mathcal{M}(t) \oplus \mathbf{z}(t-1)$ under the SINR model.

To tackle the above challenge, we impose a more stringent requirement for link coexistence beyond the previously discussed "nominal" SINR constraint so that the link coexistence relationship becomes static again. Under this "conservative" SINR constraint, we further develop the "conservative" CSMA link scheduling algorithm. For ease of exposition, we first consider a SISO network. Specifically, for each link l, we rank its interfering links $N(l)$ (the links within its close-in radius), in an ascending order based on the interference they incur to link l. We partition the interfering links in $N(l)$ into two
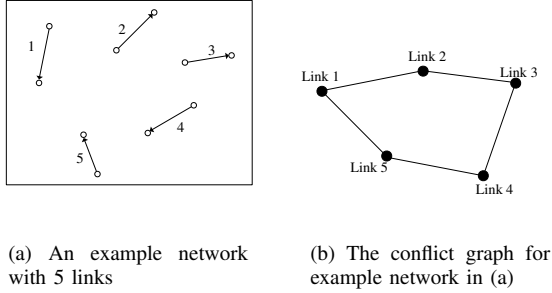
(a) An example network with 5 links

(b) The conflict graph for example network in (a)

Fig. 3. A example network and the associated conflict graph.

TABLE III
TOLERABLE SET AND INTOLERABLE SET

link	tolerable sets	intolerable sets
1	3, 4	2, 5
2	4, 5	1, 3
3	1, 5	2, 4
4	1, 2	3, 5
5	2, 3	1, 4

disjoint sets $N_a(l)$ and $N_b(l)$, i.e., $N_b(l) = N(l) \backslash N_a(l)$. Let $N_a(l)$ contain all the neighboring links (starting from the link incurring the lowest interference to the highest) such that their potential aggregated interference to link l is no greater than T_l^o, where T_l^o is defined as the initial interference tolerance level when no other neighboring links of l are active, i.e., $T_l^o = P d_{ll}^{-\alpha}/\beta_l - \sigma^2 - \sigma_{\text{int}}^2$ and $\sum_{k \in N_a(l)} P d_{kl}^{-\alpha} < T_l^o$. For convenience, we call $N_a(l)$ the "tolerable set" and $N_b(l)$ the "intolerable set." The partition of these two sets depends on the estimation of interference power levels, which requires the information of channel gains between link l and the neighboring links. As in the continuous-time case, such information can be acquired in the initialization stage. Clearly, for each link l, the sets $N_a(l)$ and $N_b(l)$ are independent with the states of nearby links. Given a fixed network topology, the $N_a(l)$ and $N_b(l)$ will not change over time.

Using the above definitions, we impose the following more stringent coexistence constraint:

Conservative coexistence constraint for SISO links: $\forall\ k \in N(l)$ and $\forall\ l \in N(k)$, links l and k can coexist if and only if $k \in N_a(l)$ and $l \in N_a(k)$.

Thanks to this new coexistence condition, the link coexistence relationship between two links becomes static again, so that the complexity of scheduling can be greatly reduced. In the meanwhile, the conservative model still takes into account the "aggregate interference effect," and provides a more realistic characterization of co-channel interference compared to the protocol model. As elaborated in Section V-B, despite the throughput loss due to the conservative coexistence constraint, the conservative scheduling can at least achieve a guaranteed fraction of the optimal throughput region.

Due to the static coexistence relationship, we can now depict a conflict graph G for the network, where each vertex corresponds to a link, and there is an edge between two vertexes if they conflict with each other. For convenience, we say that link l and link k are "severely conflicting" if they cannot satisfy the conservative coexistence constraint. Since only the links in $N_a(l)$ are allowed to transmit simultaneously with l, the aggregated interference from $N_a(l)$ is guaranteed to be lower than T_l^o, so that the nominal SINR requirement is certainly satisfied.

Fig 3(a) depicts an example network with 5 links under the conservative coexistence constraint. We assume that the tolerable sets and the intolerable sets of each link are predetermined as shown in Table III. According to the conservative coexistence constraint, only the following link pairs can coexist: (1,

3), (1, 4), (2, 4), (2, 5), (3, 5). The corresponding conflict graph of this network is shown in Fig. 3(b).

Next, we generalize the above constraint to the MIMO-pipe case by using the concept of "virtual link" introduced in the previous section. Let $\mathcal{V}(l)$ be the set of virtual links corresponding to link l, and $l_v \in \mathcal{V}(l)$ be the virtual link corresponding to the v-th configuration of link l. As before, we use $\mathbf{z}(t)$ to denote the active virtual links at time slot t, where $l_v \in \mathbf{z}(t)$ and $z_l(t) = v$, if link l chooses configuration v in the slot t.

For virtual link $l_v \in \mathcal{V}(l)$, it has a unique SINR requirement, and thus has a unique initial interference tolerance level $T_{l_v}^o$. We also define its tolerable set of virtual links as $\hat{N}_a(l_v)$ and intolerable set of virtual links as $\hat{N}_b(l_v)$ in the similar way. We impose the conservative SINR constraint under the MIMO-pipe model as follows:

Conservative coexistence constraint for the MIMO-pipe model:

- At each slot, only one virtual link in $\mathcal{V}(l)$ can transmit data.
- For two links l and k, their virtual links l_v and k_j can coexist if and only if $l_v \in \hat{N}_a(k_j)$ and $k_j \in \hat{N}_a(l_v)$.

We next devise CSMA algorithm for MIMO link scheduling by using the above conservative coexistence constraint. We combine channels for control message and data transmission, by dividing a time slot into a control slot and a data slot, each with multiple mini-slots as in [9]. During the control slot, each link contends to be included in the decision schedule $\mathcal{M}$ by broadcasting a control message. To ensure that the links in $\mathcal{M}$ can *conform the conservative constraints*, each virtual link includes the information of its intolerable set in the control message. Once a virtual link l_v sends the control message and successfully joins $\mathcal{M}$, the interfering links in $N(l)$ can check its coexistence relationship with l_v based on the information of $\hat{N}_b(l_v)$, and will give up contenting if the coexistence constraint fails to hold. Staring from an empty set, and adding links to $\mathcal{M}$ one-by-one, we can obtain the decision schedule $\mathcal{M}$ such that all the links included in $\mathcal{M}$ can conform the conservative coexistence constraint.

A complication may occur when there is a "collision" during the control slot, i.e., more than one link sends control packet to contend for channel at the same mini-slot, and they conflict under the conservative constraint. For example, suppose link l_v and link k_j that conflict under the conservative constraints contend for channel at the same mini-slot. It is possible that each link can decode its own control packet but fails to decode the packet from the other link. As a result, both links would include themselves in the decision schedule $\mathcal{M}$ independently even they conflict under the conservative constraints. To avoid this situation, we assume that once there is a collision in the

control channel (the receiver can detect the collision from the SINR level), each link will give up joining decision schedule and no one can be included in $\mathcal{M}$ in that slot. Once we obtain a decision schedule $\mathcal{M}$, we remove some links in $\mathcal{M}$ that conflict any link in $\mathbf{z}(t-1)$ and change the status of the rest links in $\mathcal{M}$ with certain probability. The proposed scheduling algorithm is summarized in Algorithm 3.

Algorithm 3 Discrete-time CSMA scheduling under the MIMO-pipe model (at link l)

Initialization: Find $\hat{N}_a(l_v)$ and $\hat{N}_b(l_v)$ for every virtual link l_v.

Selection of decision schedule $\mathcal{M}$
1) Virtual link l_v selects a random backoff time uniformly in $[1, W_l]$ mini-slots, and begins backoff.
2) Virtual link l_v stops the backoff timer and will not be included in the decision schedule, if one of the following two conditions is valid: (1) l_v hears an INTENT message[7] from virtual link k_j, and link l_v and k_j are severely conflicting links, or (2) other virtual links in $\mathcal{V}(l)$ send INTENT messages.
3) After the backoff timer expires, virtual link l_v sends INTENT message to announce its intention to be included in the decision schedule.
4) After l_v sends INTENT message, it keeps sensing the channel. If its INTENT message collides with other control messages, l_v will not be included in $\mathcal{M}(t)$ in this control slot. Otherwise, l_v will join in the decision schedule.

Setup of the transmission state
- If virtual link l_v satisfies both the following conditions: 1) $l_v \in \mathcal{M}$; 2) $l_v \notin \hat{N}_b(k_j)$ and $k_j \notin \hat{N}_b(l_v)$ for all $k_j \in \mathbf{z}(t-1)$, it will change its state: active ($z_l(t) = v$) with activation probability p_{l_v}, and inactive ($z_l(t) = 0$) with probability $\bar{p}_{l_v} = 1 - p_{l_v}$. Otherwise, l_v remains in the same state as in previous time slot, i.e., $z_l(t) = z_l(t-1)$.

Data transmission
- If $z_l(t) = v$, l will transmit using configuration v in the data slot.
- If $z_l(t) = 0$, l will not transmit in the data slot.

Observe that in Algorithm 3, each virtual link can make decisions on its transmission state independently. It is clear that the network state $\mathbf{z}(t)$ can be modeled as a discrete-time Markov chain, since the state transition probability depends on the selection probability of decision schedule $\mathcal{M}$ and the activation probability of each virtual link. As in [9], the transition probability from $\mathbf{z}$ to $\mathbf{z}'$ is given as:

$$p(\mathbf{z}, \mathbf{z}') = \sum_{\mathcal{M} \in A(\mathbf{z}, \mathbf{z}')} \epsilon(\mathcal{M}) \prod_{l_\alpha \in a} \bar{p}_{l_\alpha} \cdot \prod_{k_\beta \in b} p_{k_\beta} \cdot \prod_{i_\gamma \in c} p_{i_\gamma} \cdot \prod_{j_\theta \in d} \bar{p}_{j_\theta},$$
(15)

[7]INTENT message has the similar definitions as in [9]. The index of links in $\hat{N}_b(l_v)$ is included in the INTENT message, so any link k_j receiving this INTENT message can examine if l_v and k_j can coexist.

where $A(\mathbf{z}, \mathbf{z}')$ denotes the set of possible decision schedules $\mathcal{M}$ that include all links differ in $\mathbf{z}$ and $\mathbf{z}'$. Furthermore, $\epsilon(\mathcal{M}) > 0$ is the probability that the decision schedule $\mathcal{M}$ will be chosen in the control slot. For all virtual links included in $\mathcal{M}$ with no severely conflicting links active in the previous slot, they can be classified into four sets: set a denotes the virtual links active in $\mathbf{z}$ and inactive in $\mathbf{z}'$; set b denotes the virtual links inactive in $\mathbf{z}$ and active in $\mathbf{z}'$; set c denotes the virtual links which keep active in two states; and set d denotes the virtual links which keep inactive in two states. Also, p and $\bar{p}$ are the corresponding activation probabilities specified in Algorithm 2. It can be verified that the stationary distribution of feasible state $\mathbf{z}^i$ is given by:

$$p(\mathbf{z}^i) = \frac{1}{C} \prod_{l_v \in \mathbf{z}^i} \frac{p_{l_v}}{\bar{p}_{l_v}},$$
(16)

where C is the normalization term satisfying $\sum_i p(\mathbf{z}^i) = 1$.

As in the continuous-time case, each p_{l_v} can be adapted using local queue information.

Lemma 5.2: Under the time-scale-separation assumption [8], the CSMA algorithm for MIMO scheduling can achieve any network throughput λ in the capacity region corresponding to the conservative coexistence constraint, by adjusting the activation probability of virtual links as follows:

For link l,

$$y_l(t+1) = [y_l(t) + \xi(\lambda_l - \theta_l(t))]^+,$$

where y_l is shown to be proportional to the queue length at link l [8], and $\xi > 0$ is a small constant (step size). Each virtual link can update its activation probability according to

$$p_{l_v} = \frac{e^{y_l \Theta(v)}}{1 + e^{y_l \Theta(v)}}$$

where $\Theta(v)$ is the data rate of configuration v.

We provide the proof of Lemma 5.2 in Appendix B.

Note that each link may not fully utilize its initial interference tolerance due to the conservative coexistence constraint. Since the feasible states under the conservative SINR constraint will be a subset of those under the nominal SINR constraint, it is clear that the capacity region corresponding to the conservative coexistence constraint is only a fraction of that under the nominal SINR constraint. Hence, the "conservative scheduling" achieves a suboptimal performance. In the following, we will show that the conservative scheduling at least achieves a guaranteed fraction of the optimal throughput region.

B. Efficiency Ratio of Conservative MIMO-pipe Scheduling

In this section, we characterize the throughput performance achieved by the conservative SINR-based scheduling. Specifically, we provide a lower bound of $\gamma \in [0, 1]$ such that for any traffic arrival rate λ in the capacity region under the nominal SINR constraint, $\gamma\lambda$ is supported by the conservative scheduling. The fraction γ is called as the efficiency ratio.

Recall that the throughput region of our suboptimal scheduling algorithm is the convex hull of the set of feasible states under the conservative SINR constraint. To compare

the throughput region of CSMA algorithm under different interference constraints, it suffices to compare the convex hulls formed by their feasible states.

For convenience, let $\mathcal{S}$ and $\mathcal{C}$ be the sets of the rate vectors obtained from the feasible states under the nominal SINR model and the conservative SINR model, respectively. For a MIMO-pipe model with K links, we use a K-dimension vector to denote the feasible rates, where each element is the link transmission rate at the corresponding state. For each feasible rate $s \in \mathcal{S}$, there exists a subset $C \subset \mathcal{C}$ such that the set of the active virtual links in s, can be "covered" by the union of the sets of the active virtual links for the feasible rate in C, i.e.,

$$\{l \in 1, 2, \cdots, K : s_l = r \text{ in State } s\}$$
$$\subset \bigcup_{c \in C} \{l \in 1, 2, \cdots, K : c_l = r \text{ in State } c\}. \quad (17)$$

Note that there may exist multiple different subsets $C \subset \mathcal{C}$ that "cover" the set of the active links of s. Nevertheless, we will show that only the subsets with the least cardinality are closely related to the efficiency ratio.

Let $V_k^* \subset \mathcal{C}$ be the *minimal covering set* for state s_k in the sense that 1) V_k^* satisfies (17), and 2) for any other subset $V \subset \mathcal{C}$ that satisfies (17), we have that the cardinality of V_k^* is no larger than that of V, i.e., $|V_k^*| \leq |V|$.

Define the *effective interference number* as the maximum of the cardinalities among the minimal covering set for all the feasible rates in $\mathcal{S}$, i.e.,

$$N(\mathcal{S}, \mathcal{C}) \triangleq \max_{\{k : s_k \in \mathcal{S}\}} |V_k^*|.$$

Under the conservative SINR model, any s_k in $\mathcal{S}$ can be decomposed into no more than $N(\mathcal{S}, \mathcal{C})$ states in $\mathcal{C}$, where $N(\mathcal{S}, \mathcal{C})$ depends on the coexistence relationship of links.

Theorem 5.1: The conservative MIMO-pipe scheduling results in an efficiency ratio $\gamma \geq 1/N(\mathcal{S}, \mathcal{C})$.

The proof is given in Appendix C.

The above result reveals that the efficiency ratio is bounded from below by the reciprocal of the effective interference number. Note that determining the effective interference number requires globe information of all the feasible states in general. In the following, we develop a local search algorithm to find an upper bound on the effective interference number.

Observe that for any virtual link l_v, there may exist a set of virtual links $L = \{l_v\} \cup \{\mathcal{N} | \mathcal{N} \subset \hat{N}(l_v)\}$, such that the virtual links in L can coexist under the nominal SINR constraint, where $\hat{N}(l_v) = N_a(l_v) \cup N_b(l_v)$. We call L a "local feasible state," and clearly virtual link l_v can have multiple local feasible states. We use $L(l_v, j)$ to denote the j-th local feasible state of l_v, and $n_v(l_v, j)$ to denote the number of links in $L(l_v, j)$ severely conflicting with l_v under the *conservative SINR constraint*, i.e., $n_v(l_v, j) = |L(l_v, j) \cap N_b(l_v)|$. We further define

$$n_e \triangleq \max_{l_v} \max_{L(l_v, j)} n_v(l_v, j).$$

It follows that for any virtual link, $n_v(l_v, j)$ would be no greater than n_e. Detailed algorithm to find n_e is provided in Algorithm 4. We next have the following result.

Theorem 5.2: The effective interference number is upper bounded by $n_e + 1$, i.e., $N(\mathcal{S}, \mathcal{C}) \leq n_e + 1$.

The proof is given in Appendix D.

Algorithm 4 Local search algorithm

let $n_e = 0$;
for $l = 1$ to K **do**
　For link l, let $n_l = 0$
　for $v = 1$ to J **do**
　　For virtual link l_v, let $n_v(l_v) = 0$
　　repeat
　　　For local feasible state $L(l_v, j)$
　　　if $n_v(l_v, j) \geq n_v(l_v)$, **then**
　　　　$n_v(l_v) = n_v(l_v, j)$
　　　end if
　　until all local feasible states of l_v has been enumerated
　　if $n_v(l_v) \geq n_l$, **then**
　　　$n_l = n_v(l_v)$
　　end if
　end for
　if $n_l \geq n_e$, **then**
　　$n_e = n_l$
　end if
end for

Combining Theorems 5.1 and 5.2, we conclude that

$$\gamma \geq \frac{1}{N(\mathcal{S}, \mathcal{C})} \geq \frac{1}{n_e + 1}. \quad (18)$$

VI. NUMERICAL EXAMPLES

In this section, we illustrate, via numerical examples, the performance of the proposed CSMA algorithms in a multi-hop MIMO-pipe network. We explore the cases for both continuous-time model and discrete-time model.

A. Simulation Settings

Specifically, we study a network with six 4×4 MIMO links. Assume that each link has three possible configurations, with data rate 1 (data unit/ms), 2 (data units/ms) and 4 (data units/ms), respectively. We construct the network topology as follows. Consider an area of 20×20 square unit, we randomly deploy six transmitter-receiver pairs, such that each receiver is within distance 3 from the corresponding transmitter. According to (1), the signal power from the transmitter attenuates as it propagates through space. In the simulations, the path loss exponent α is fixed at 2 and the transmission power P is set to 1 unit. For the white noise, we set $SNR_{dB} = 10 \log P/\sigma^2 = 20$dB. We also choose $\sigma_{int}^2 = 0$, and hence the close-in range of each link includes other 5 links. The SINR requirements corresponding to three configurations are 8dB, 16dB and 24dB, respectively.

We illustrate the queue length behaviors of MIMO-pipe scheduling under different traffic loads. To illustrate the throughput optimality, we first find an arrival rate vector at the boundary of capacity region, denoted as $\bar{\lambda}$. Then, we consider a "load factor" ρ, $\rho > 0$, and set the traffic load at $\lambda = \rho \bar{\lambda}$ as in [20]. Clearly, the traffic load is in the capacity

region if $\rho < 1$ and outside the capacity region if $\rho > 1$. We build up $\bar{\lambda}$ by using a set of feasible states under the nominal SINR constraints. Specifically, for feasible state i, let $\mathbf{c}^i$ denote the rate vector of active links, and let s^i denote the summation of the active link rates, i.e., $s^i = \left\| \mathbf{c}^i \right\|_1$. Among all the feasible rate vectors, let $\mathbf{M}$ be the set of vectors with maximal value of s^i, i.e., $\mathbf{M} = \{ c^i : s^i = \max_j s^j \}$. Clearly, a convex combination of a set of rate vectors in $\mathbf{M}$ corresponds to a point on the boundary of the capacity region. In the simulations, we simply choose $\bar{\lambda} = \frac{1}{|\mathbf{M}|} \sum \mathbf{c}^i$, $\mathbf{c}^i \in \mathbf{M}$.

B. Continuous-time Network Model

To illustrate the throughput-optimality, we compare the queue behaviors of continuous-time CSMA algorithm under different traffic loads. Specifically, the queue length usually keeps increasing if the network throughput cannot meet the traffic demands. Note that a scheduling algorithm is said to be throughput-optimal if it can yield stable queue length behaviors at any traffic load in the capacity region, corresponding to $\rho < 1$ [21]. We first consider $\rho = 0.98$ such that the traffic arrival rate vector $\lambda = \rho\bar{\lambda}$ is in the interior of capacity region. As shown in Fig. 4, the scheduling algorithm yields stable queue length behavior at each link, indicating it can achieve network throughput λ. Fig. 5 exemplifies the throughput-optimality by comparing the total queue length under various ρ. As expected, the total queue length tends to be stable under traffic load in the capacity region ($\rho < 1$). However, while $\rho > 1$, the queue length grows rapidly, and the system will become unstable, which means the scheduling algorithm fails to support the traffic loads beyond the capacity region.
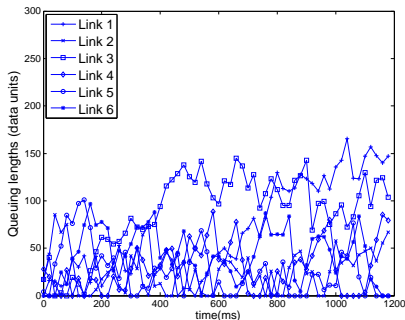

Fig. 4. Continuous-time model: queueing length behavior at each MIMO link with $\rho = 0.98$.
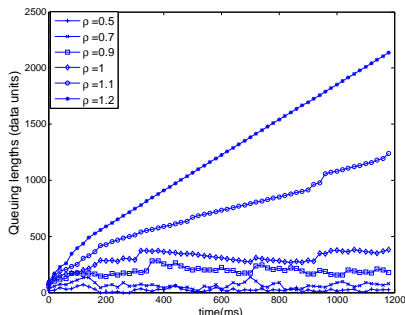

Fig. 5. Continuous-time model: total queue lengths of 6 MIMO links with different ρ values.

C. Discrete-time Network Model

We evaluate the conservative CSMA scheduling scheme under the discrete-time model. Due to its throughput suboptimality, the conservative scheduling scheme can only achieve a fraction of the capacity region and cannot support all the traffic loads with $\rho < 1$. We illustrate its throughput performance by comparing the total queue lengths under various ρ in Fig. 6. We observe that when $\rho \geq 0.6$ the queue length keeps increasing, indicating that the scheme can no longer support the traffic loads with $\rho \geq 0.6$. We also compare the queue behaviors for the continuous-time case and the discrete-time case in Fig. 7. In this figure, we depict the total queue lengths averaged over the period from 1600ms to 2000ms. We observe that the queue length corresponding to the discrete-time case grows rapidly at a smaller ρ than that of the continuous-time case, indicating its inferior performance to the continuous-time scheduling scheme.

For this scenario, we find that the effective interference number $N(\mathcal{S}, \mathcal{C})$ is no more than 2 by using Algorithm 4 and hence the efficiency ratio γ is no less than 0.5 by Theorem 5.1. It follows that the conservative scheduling can at least achieve a $\frac{1}{2}$ fraction of the capacity region, which is confirmed by Fig. 6. Indeed, the network remains stable under traffic load with $\rho = 0.55$.
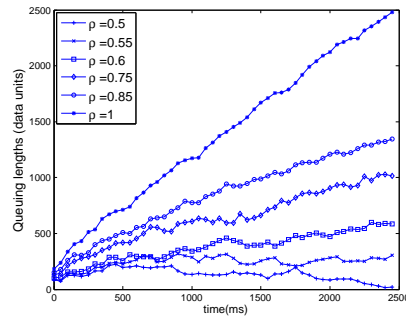

Fig. 6. Discrete-time model: total queue lengths of 6 MIMO links with different ρ values.
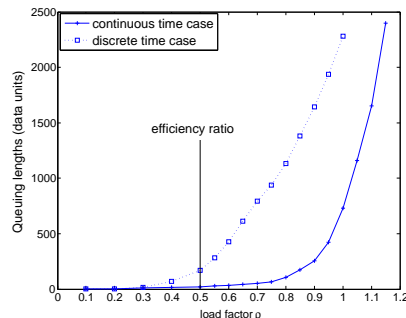

Fig. 7. Comparisons of total queue lengths for continuous-time model and discrete-time model

VII. CONCLUSION AND FUTURE WORK

We investigate CSMA algorithms in multi-hop MIMO networks under the SINR interference model. To this end, we first developed a MIMO-pipe model that provides the upper

layers a set of rates and SINR requirements, which capture the rate-reliability tradeoffs in MIMO communications. We then focused on developing distributed scheduling for MIMO-pipe networks under the SINR model. Specifically, we explored the CSMA algorithms for MIMO-pipe scheduling in both a continuous-time system and a discrete-time system. Particulary, in the idealized continuous-time CSMA network, we proposed a dual-band approach to facilitate the message passing on interference tolerance levels, and showed that the CSMA scheduling algorithm can achieve throughput optimality under the SINR model. For the more difficult discrete-time case, we developed a "conservative" scheduling algorithm in which a more stringent SINR constraint is imposed. We showed that an efficiency ratio bounded below can be achieved by our distributed scheduling algorithm.

We believe that the studies here on SINR-based distributed scheduling scratch only the tip of the iceberg. Clearly, there are still many open issues in the MIMO network scheduling. One interesting issue is how to generalize the MIMO-pipe model into different types of channel fading scenarios. In addition to the SINR level, it is also intriguing to consider other parameters in a realistic MIMO scenario to evaluate the QoS of MIMO communication. It is worth studying the joint design of link scheduling and dynamic power control to better leverage the interference among MIMO links. For the more practical discrete-time case, it remains open to develop a CSMA scheduling algorithm with throughput-optimality under the SINR interference model. We are currently investigating these issues along this avenue.

REFERENCES

[1] D. Qian, D. Zheng, J. Zhang, and N. Shroff, "CSMA-based distributed scheduling in multi-hop MIMO networks under SINR model," in *IEEE INFOCOM*, 2010.
[2] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, 1998.
[3] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecom.*, vol. 10, pp. 585–595, Nov. 1999.
[4] L. Zheng and D. Tse, "Diversity and multiplexing: A fundamental trade-off in multiple-antenna channels," *IEEE Transactions on Information Theory*, vol. 49, no. 5, pp. 1073–1096, 2003.
[5] J. Liu, Y. T. Hou, Y. Shi, H. D. Sherali, and S. Kompella, "On the capacity of multiuser MIMO networks with interference," *IEEE Transactions on Wireless Communications*, vol. 7, no. 2, pp. 488–494, 2008.
[6] B. Hamdaoui and K. G. Shin, "Characterization and analysis of multi-hop wireless MIMO network throughput," in *ACM MobiHoc*, 2007.
[7] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, 1992.
[8] L. Jiang and J. Walrand, "A distributed csma algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 960–972, 2010.
[9] J. Ni and R. Srikant, "Distributed CSMA/CA Algorithms for Achieving Maximum Throughput in Wireless Networks," *ITA*, 2009.
[10] P. Marbach, A. Eryilmaz, and A. Ozdaglar, "Achievable rate region of CSMA schedulers in wireless networks with primary interference constraints," in *IEEE CDC*, 2007.
[11] S. Rajagopalan and D. Shah, "Distributed algorithm and reversible network," in *CISS*, 2008.
[12] J. Liu, Y. Yi, A. Proutiere, M. Chiang, and H. Poor, "Maximizing utility via random access without message passing," Microsoft Research, Tech. Rep., 2008.
[13] A. Lozano, A. M. Tulino, and S. Verdú, "Multiple-antenna capacity in the low-power regime," *IEEE Transaction on Information Theory*, vol. 49, no. 10, pp. 2527–2544, 2003.
[14] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.
[15] G. Brar, D. Blough, and P. Santi, "Computationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks," in *ACM MobiCom*, 2006.
[16] P. Pinto, "Communication in a Poisson field of interferers," Ph.D. dissertation, Massachusetts Institute of Technology, 2007.
[17] O. Goussevskaia, Y. A. Oswald, and R. Wattenhofer, "Complexity in geometric SINR," in *ACM MobiHoc*, 2007.
[18] L. Jiang and J. Walrand, "Approaching throughput-optimality in distributed csma scheduling algorithms with collisions," *IEEE/ACM Transactions on Networking*, vol. 19, no. 3, pp. 816–829, 2011.
[19] S. Rajagopalan, D. Shah, and J. Shin, "Network adiabatic theorem: an efficient randomized protocol for contention resolution," in *Proceedings of the eleventh international joint conference on Measurement and modeling of computer systems*, 2009.
[20] L. Jiang and J. Walrand, "Convergence and stability of a distributed CSMA algorithm for maximal network throughput," in *IEEE CDC*, 2009.
[21] C. Joo and N. Shroff, "Local greedy approximation for scheduling in multi-hop wireless networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 99, pp. 414 – 426, 2011.
[22] R. Diestel, "Graph theory," *Springer, Heidelberg*, vol. 91, 2005.

APPENDIX

A. Proof of Lemma 4.1

Following the same lines as in [8], we study the backoff time adaption algorithm based on the following entropy maximization problem:

$$\begin{aligned} \max \quad & -\sum_i u_i \log u_i \\ s.t. \quad & \sum_i u_i \cdot c_l^i \geq \lambda_l, \\ & u_i \geq 0, \quad \sum_i u_i = 1. \end{aligned} \quad (19)$$

Assume that each i relates to a feasible state in the MIMO network. In contrast to the binary data rate in the SISO link case [8], the MIMO link rate c_l^i can take multiple values depending on the link configuration. If this problem is feasible, the optimal point $\mathbf{u}^*$ would satisfy the constraint $\sum_i u_i^* \cdot c_l^i \geq \lambda_l$. That is to say, as long as the optimal value u_i^* equals the stationary distribution of feasible states (13), then each MIMO link will meet the throughput requirement $\theta_l \geq \lambda_l$ according to (14). With this insight, a key challenge is to find a sufficient condition for the equivalence of these two distributions, i.e., $p(\mathbf{z}^i) = u_i^*$. The Lagrangian of (19) can be written as

$$\begin{aligned} L_1 = & -\sum_i u_i \log u_i + \sum_l y_l (\sum_i u_i \cdot c_l^i - \lambda_l) \\ & + \mu(\sum_i u_i - 1) + \sum_i w_i u_i, \end{aligned} \quad (20)$$

where y, μ and w are dual variables. Based on the KKT condition, we obtain that

$$u_i^* = \frac{\exp(\sum_{l=1}^K y_l c_l^i)}{\sum_j \exp(\sum_{l=1}^K y_l c_l^j)}. \quad (21)$$

With (13), it can ensure $p(\mathbf{z}^i) = u_i^*$ if the following condition holds:

$$\exp\left(\sum_{l=1}^K y_l c_l^i\right) = \exp\left(\sum_{l=1}^K r_l^i\right), \quad \forall i. \quad (22)$$

From $c_l^i = \Theta(z_l^i)$ and $r_l^i = r_{l_v}$ when l_v is the active link for state i, a sufficient condition for (22) is

$$r_{l_v} = y_l \Theta(v), \quad \forall v \in [1...J].$$

This condition can also be rewritten as:

$$R_{l_v} = \exp(y_l\Theta(v)), \quad \forall\, v \in [1...J]. \tag{23}$$

As in [8], the optimal dual variable y_l^* is essentially proportional to queue length at link l, and can be achieved by using the following gradient method:

$$y_l(t+1) = [y_l(t) + \xi(\lambda_l - \theta_l(t))]^+.$$

Meanwhile, each virtual link can adjust its backoff time according to (23). Note that the above adaptive algorithm depends on accurate estimation of link throughput $\theta_l(t)$. As in [8], we take the same time-scale-separation assumption, i.e., the variable y_l changes slowly enough so that the CSMA Markov chain can converge to its stationary distribution within each duration t and $t+1$. By doing so, we can always obtain a good estimation of the link throughput.

B. Proof of Lemma 5.2

Based on the Markov chain modeling, the activation probability of each virtual link can be obtained by the same gradient method as in Section IV-B. The only additional requirement is that the stationary distribution of the feasible states in the discrete-time network (16) equals the distribution (21). A sufficient condition for this requirement turns out to be:

$$\frac{p_{l_v}}{\bar{p}_{l_v}} = \exp(y_l\Theta(v)), \quad \forall\, l, v,$$

and equivalently

$$p_{l_v} = \frac{e^{y_l\Theta(v)}}{1 + e^{y_l\Theta(v)}}, \tag{24}$$

where $\Theta(v)$ is the data rate of configuration v. Clearly, y_l can be achieved along the same line as in the continuous-time network, and each virtual link can update its activation probability according to (24). It follows that the adaptive algorithm also requires the time-scale-separation assumption in [8].

C. Proof of Theorem 5.1

For any feasible traffic arrival rate $\lambda = \{\lambda_1, \lambda_2, \cdots, \lambda_K\}^T$ under the SINR model, there exists a state probability vector $\mathbf{P} = \{P_1, P_2, \cdots, P_{|\mathcal{S}|}\}^T$ such that $\sum_{i=1}^{|\mathcal{S}|} P_i = 1$, and

$$\mathbf{P}^T\mathbf{A}^S \geq \lambda, \tag{25}$$

where $\mathbf{A}^S$ is a $|\mathcal{S}| \times K$ matrix, with

$$A_{k,l}^S \triangleq (\text{Transmission rate of link } l \text{ in state } s_k), \forall\, k, l. \tag{26}$$

To show that $\gamma \geq \frac{1}{N(\mathcal{S},\mathcal{C})}$, it suffices to show that there exists a state probability vector $\mathbf{Q} = \{Q_1, Q_2, \cdots, Q_{|\mathcal{C}|}\}^T$ such that

$$\mathbf{Q}^T\mathbf{A}^C \geq \gamma\lambda, \tag{27}$$

where $\mathbf{A}^C$ is defined in the same way as $\mathbf{A}^S$ in (26). We use induction on $|\mathcal{S}|$ to show that (27) is valid for some $\mathbf{Q}$, for any given $\mathbf{P}$ satisfying (25). It is easy to verify when $|\mathcal{S}| = 1$. Assume that the conclusion holds when $|\mathcal{S}| = n$. Now we consider the case $|\mathcal{S}| = n+1$, pick the state s_k in $\mathcal{S}$ such that $|V_k^*| = N(\mathcal{S}, \mathcal{C})$. Without the loss of generality, suppose $k = n+1$.

It follows from (25) that for $l = 1, 2, \cdots, K$,

$$\sum_{i=1}^{n} P_i A_{i,l}^S + P_{n+1} A_{n+1,l}^S \geq \lambda_l, \tag{28}$$

which indicates that for $l = 1, 2, \cdots, K$,

$$\sum_{i=1}^{n} P_i' A_{i,l}^S \geq \lambda_l', \tag{29}$$

where

$$P_i' \triangleq \frac{P_i}{1 - P_{n+1}}, \quad \lambda_l' \triangleq \frac{\lambda_l - P_{n+1} A_{n+1,l}^S}{1 - P_{n+1}}. \tag{30}$$

By induction, based on (29), there exists $\mathbf{Q}'$ such that $\sum_{j=1}^{|\mathcal{C}|} Q_j' = 1$, and for $l = 1, 2, \cdots, K$,

$$\sum_{j=1}^{|\mathcal{C}|} Q_j' A_{j,l}^C \geq \gamma'\lambda_l', \tag{31}$$

where $\gamma' \triangleq \frac{1}{N(\mathcal{S}',\mathcal{C})}$ and $\mathcal{S}' = s_k, k = 1, 2, \cdots, n$. It is clear that, $\gamma' \geq \gamma$, and it follows that

$$\sum_{j=1}^{|\mathcal{C}|} Q_j' A_{j,l}^C \geq \gamma\lambda_l', \forall\, l = 1, 2, \cdots, K. \tag{32}$$

Similar, we can find $\mathbf{Q}''$ such that $\sum_{j=1}^{|\mathcal{C}|} Q_j'' = 1$, and

$$\sum_{j=1}^{|\mathcal{C}|} Q_j'' A_{j,l}^C \geq \gamma A_{n+1,l}^S, \forall\, l = 1, 2, \cdots, K. \tag{33}$$

Define

$$Q_j \triangleq Q_j'(1 - P_{n+1}) + Q_j'' P_{n+1}, \forall\, j = 1, 2, \cdots, |\mathcal{C}|. \tag{34}$$

Observe that $\mathcal{Q} = \{Q_1, Q_2, \cdots, Q_{|\mathcal{C}|}\}^T$ defined above is a state probability vector, i.e.,

$$\sum_j Q_j = (\sum_j Q_j')(1 - P_{n+1}) + (\sum_j Q_j'')P_{n+1} = 1. \tag{35}$$

Furthermore, multiplying (32) with $(1 - P_{n+1})$ on both sides yields that

$$\sum_{j=1}^{|\mathcal{C}|} Q_j'(1 - P_{n+1})A_{j,l}^C \geq \gamma(\lambda_l - P_{n+1}A_{n+1,l}^S), \forall\, l = 1, 2, \cdots, K. \tag{36}$$

Further, multiplying (33) with P_{n+1} yields that

$$\sum_{j=1}^{|\mathcal{C}|} Q_j'' P_{n+1} A_{j,l}^C \geq \gamma P_{n+1} A_{n+1,l}^S, \forall\, l = 1, 2, \cdots, K. \tag{37}$$

Adding the above two equations together, we see that $\mathcal{Q}$ defined in (34) satisfies (27), and the proof is concluded.

D. Proof of Theorem 5.2

Under the conservative SINR model, we can build a conflict graph G associated with the MIMO-pipe network, where each vertex corresponds to a virtual link. The feasible state, under the SINR model, $s_k \in \mathcal{S}$ corresponds to a subgraph of G, and the feasible state under the conservative SINR model corresponds to an independent set of G. Let $G(s_k)$ be the subgraph of G, which only contains the vertexes corresponding to the active virtual links in s_k and their associated edges.

The value $|V_k^*|$ relating to state s_k can be interpreted as the minimum number of independent sets to construct the subgraph $G(s_k)$. The problem of finding these independent sets boils down to a graph coloring problem [22]. According to graph theory, we can decompose any subgraph $G(s_k)$ into no more than $\Delta(G(s_k))+1$ independent sets, where $\Delta(G(s_k))$ is the maximum degree of $G(s_k)$.

Next, we establish the relationship between $\Delta(G(s_k))$ and n_e from local search algorithm in Section V. In the conflict graph, let $v(l_v)$ denote the vertex corresponding to virtual link l_v. Define $\deg(l_v, G(s_k))$ as the degree of vertex $v(l_v)$ in subgraph $G(s_k)$. Then we have the following result:

$$\max_{v(l_v) \in G(s_k)} \deg(l_v, G(s_k)) = \Delta(G(s_k)). \qquad (38)$$

Recall that n_e is the maximum number of links severely conflicting with l_v in any local feasible state under the conservative SINR constraint, where there is no interference from links other than $l_v \cup N(l_v)$. If any link other than $l_v \cup N(l_v)$ is active, some links in $L(l_v, j)$ may no longer satisfy the nominal SINR constraint. Hence, the number of conflicting links which can be active simultaneously with any virtual link l_v, under the nominal SINR constraint, must be no greater than n_e. Therefore, we conclude that

$$n_e \geq \deg(l_v, G(s_k)), \quad \forall\, l_v \in s_k, \forall\, s_k \in \mathcal{S}. \qquad (39)$$

It follows that

$$
\begin{aligned}
n_e &\geq \max_{l_v \in s_k} \deg(l, G(s_k)), \forall\, s_k \in S, \\
&= \Delta(G(s_k)), \forall\, s_k \in S. \qquad (40)
\end{aligned}
$$

In conclusion, $n_e + 1$ is an upper bound for $|V_k^*|$ for $\forall s_k \in \mathcal{S}$, and hence an upper bound for $N(\mathcal{S}, \mathcal{C})$ as well.

Junshan Zhang received his Ph.D. degree from the School of ECE at Purdue University in 2000. He joined the EE Department at Arizona State University in August 2000, where he has been Professor since 2010. His research interests include communications networks, cyber-physical systems with applications to smart grid, stochastic modeling and analysis, and wireless communications. His current research focuses on fundamental problems in information networks and network science, including network optimization/control, smart grid, cognitive radio, and network information theory.

Prof. Zhang is a fellow of the IEEE, and a recipient of the ONR Young Investigator Award in 2005 and the NSF CAREER award in 2003. He received the Outstanding Research Award from the IEEE Phoenix Section in 2003. He served as TPC co-chair for WICON 2008 and IPCCC'06, TPC vice chair for ICCCN'06, and a member of the technical program committees of INFOCOM, SECON, GLOBECOM, ICC, MOBIHOC, BROADNETS, and SPIE ITCOM. He was the general chair for IEEE Communication Theory Workshop 2007. He was an Associate Editor for IEEE Transactions on Wireless Communications. He is currently an editor for the Computer Network journal and IEEE Wireless Communication Magazine. He co-authored a paper that won IEEE ICC 2008 best paper award, and one of his papers was selected as the INFOCOM 2009 Best Paper Award Runner-up. He is TPC co-chair for INFOCOM 2012.

Ness B. Shroff (S'91-M'93-SM'01-F'07) received his Ph.D. degree from Columbia University, NY in 1994 and joined Purdue university immediately thereafter as an Assistant Professor. At Purdue, he became Professor of the school of Electrical and Computer Engineering in 2003 and director of CWSA in 2004, a university-wide center on wireless systems and applications. In July 2007, he joined the ECE and CSE departments at The Ohio State University, where he holds the Ohio Eminent Scholar Endowed Chair position in Networking and Communications. He is interested in fundamental problems in the design, performance, control, and security of communication, social, and cyberphysical systems. Dr. Shroff is a past editor for IEEE/ACM Trans. on Networking and the IEEE Communications Letters. He currently serves on the editorial board of the IEEE Network Magazine, the Computer Networks Journal, and the Networks Science journal. He has served on the technical and executive committees of several major conferences and workshops. He has received numerous awards for his work, including two best paper awards at IEEE INFOCOM (in 2006 and 2008), the IEEE IWQoS 2006 best student paper award, the 2005 best paper of the year award for the Journal of Communications and Networking, the 2003 best paper of the year award for Computer Networks, and the NSF CAREER award in 1996 (his IEEE INFOCOM 2005 paper was selected as one of two runner-up papers).

Dajun Qian received his B.S. and M.S. degrees of Electrical Engineering from Southeast University, Nanjing, China, in 2006 and 2008, respectively. Currently, he is a Ph.D. student in the Department of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ. His research interests include wireless communications, social networks and cyber-physical systems.
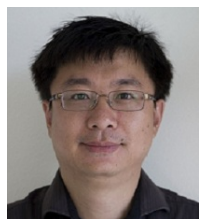
Dong Zheng received the Ph.D. degree in Electrical Engineering from Arizona State University, Arizona, USA, in 2007. Before that, He received the B.S. and M.S. degree from Shanghai Jiaotong University and Mississippi State University in 2000 and 2002, respectively. He is currently a scientist in Broadcom. Dong Zheng's research interests are in the area of cross-layer design for wireless networks, stochastic control, network performance evaluation and optimization.

Changhee Joo (S'98-M'05) received his Ph.D degree from the school of Electrical and Computer Engineering, Seoul National University, Korea, 2005. He was with the Center of Wireless Systems and Applications, Purdue University, and also work at the Ohio State University as a research scientist. In 2010, he joined Korea University of Technology and Education as a faculty member and he now works at the Ulsan National Institute of Science and Technology (UNIST). His research interests span the area of communication network systems, including cross-layer network optimization, network performance and controls, and wireless sensor networks. He is a member of IEEE, and a recipient of the IEEE INFOCOM 2008 best paper award.