On Fast Optimal STDMA Scheduling over Fading Wireless Channels

Jialiang Zhang, Soung Chang Liew, Liqun Fu
Department of Information Engineering
The Chinese University of Hong Kong
Hong Kong SAR, P. R. China
Email: {jlzhang, soung, lqfu6}@ie.cuhk.edu.hk

***Abstract*—Most prior studies on wireless spatial-reuse TDMA (STDMA) link scheduling for throughput optimization deal with the situation where instantaneous channel state information (CSI) is available. Under fast fading, however, the channel may change too quickly for the scheduler to track the instantaneous CSI. In this paper, instead of the instantaneous CSI, the scheduler performs its task according to the stochastic behavior of the channel state. A basic schedule consists of a set of simultaneously transmitting links. The essence of the scheduling problem is to determine a mixed schedule consisting of a weighted sum (TDMA mixture) of a number of basic schedules to optimize a certain utility objective. A key to reducing scheduling complexity is to identify the Pareto-efficient basic schedules, referred to as the extreme-point schedules, so that the construction of the optimal mixed schedule can be based on the extreme-point schedules rather than all the basic schedules. The precise identification of the extreme-point schedules, however, is intractable computationally. We show in this paper that identifying a slightly larger superset of the extreme-point schedules can be highly efficient using a Perron-Frobenius condition: simulation experiments indicate that oftentimes there are only very few extraneous non-extreme-point schedules in the superset. Building on the effective identification method, we propose a fast scheduling algorithm. This algorithm beats the algorithm without using the identification method by a complexity-reduction factor of 166 in a 15-link network. In addition, numerical results suggest that our algorithm is robust to variations of system parameters.**

***Index Terms*—STDMA scheduling; Perron-Frobenius Theorem**

I. Introduction

Spatial-reuse TDMA (STDMA) link scheduling [1] for throughput optimization in wireless networks is an important issue that has attracted much research attention. In STDMA scheduling, each scheduling period consists of a number of time slots, and in each time slot a subset of links are scheduled to transmit together. Essentially, scheduling is an exercise of a tradeoff between spatial reuse (allowing as many links to transmit simultaneously as possible) and mutual interferences between links (which lower the throughput of each link when it transmits).

The majority of past work deals with non-fading or slow-fading channels, where the scheduler is assumed to be capable of keeping track of the instantaneous channel state information (CSI) of links so that it can schedule transmissions accordingly [2-7]. The existing techniques no longer apply, however, when the channels are fast varying in a way that within a scheduling period (i.e., the interval between two successive schedule updates) the channel states change substantially. In this case, we will need to take the statistical variations of channel states within a scheduling period into account.

This paper assumes that the scheduling system cannot track the fast changing **small-time-scale fading**, but can track the **large-time-scale fading (log-normal shadowing)**. Typically, the time scales of small-scale fading and log-normal shadowing are of order of milliseconds and seconds, respectively. For example, for a mobile station traveling at 20m/s operating at carrier frequency 2.4GHz, typical coherence time due to Doppler spread is around **1.1ms** (according to Eq. 4.40b in [8]), whereas typical coherence time for shadowing is **5s** [9]. As a reference point, the scheduling period in our system could be assumed to be around, say, **2s**. This scheduling period is long enough for the channel to visit a good portion of its possible small-scale fading states so that the **ensemble-average throughputs** of links would be a good approximation of the **actual throughputs**. This allows us to design schedules based on the ensemble-average throughputs computed from the estimated channel parameters. Henceforth in this paper, **by throughput we mean ensemble-average throughput**.

We define a **basic schedule** to be a set of simultaneously transmitting links. A **mixed schedule** is a time-division schedule, i.e., a weighted sum of a number of basic schedules in which the weights are fractions of airtimes allocated to basic schedules in a scheduling period. **Our general goal is to efficiently determine a mixed schedule with the link throughputs optimized according to some utility objective**.

The primary challenge we face is the **computational bottleneck** arising from the excessive number of possible basic schedules for the construction of the mixed schedule. Consider an n-link network. A mixed schedule corresponds to 2^n variables (weights), each associated with a basic schedule. In our simulation experiments, it takes 15.5 seconds on average to schedule a 15-link network, which exceeds the typical shadowing coherence time (5s).

A remedy is to reduce the number of basic schedules to

This work was supported by the Competitive Earmarked Research Grant (Project Number 414507) established under the University Grant Committee of the Hong Kong Special Administrative Region, China.

978-1-4244-3513-5/09/$25.00 ©2009 IEEE

be considered, v. In particular, if the link throughputs of a basic schedule can be achieved by a convex combination of other basic schedules, then removing the basic schedule from consideration will not sacrifice optimality, but will reduce v by 1. The more such "redundant" basic schedules can be found, the more reduction on v. We refer to a basic schedule which can not be replaced by a convex combination of other basic schedules as an **extreme-point schedule**. The complete set of extreme-point schedules form the **bases** of the feasible space of schedules - that is, any feasible schedule can be expressed as a convex combination of the bases. If we could identify the bases and use only them in scheduling, then our program can be speeded up significantly. The **identification of the bases**, however, is non-trivial, and is the **central focus** of this paper.

A. Related Work

To the best of our knowledge, very few papers have been devoted to studies assuming the above fast fading scenario. An exception is [10], which addresses the power control problem to balance the outage probabilities across links. Another is [11], which considers a framework that incorporates both fast and slow fading (relative to application-time window) in proportional fairness resource allocation. Ref. [10] assumes all links are active simultaneously and therefore does not make use of scheduling to boost system throughput. Ref. [11] assumes a cellular network structure with base stations operating at orthogonal frequency channels, where only one client station is allowed to transmit in each orthogonal channel at any one time. In particular, simultaneous transmissions on the same channel that may mutually interfere with each other are not considered. In contrast to [10] and [11], our paper here focuses on scheduling of simultaneously transmitting links to optimally trade off spatial reuse against mutual interference to boost system throughput.

There have also been extensive studies on **instantaneous-CSI-aware** STDMA scheduling since [1]. Our work is related to prior works in [2-7] in that we also assume a "cumulative interference" model. While earlier works assume a graph-based interference model that considers only pairwise-interference relationships between links, these recent studies were devoted to the more realistic SINR-based model that considers cumulative interferences among links (i.e., the physical model in [12]). They formulated the scheduling problem as optimization problems with airtimes allocated to the links as optimization variables. Ref. [3] conjectured that the complexity to get the optimal solution in general requires exponential time in n due to the 2^n possible basic schedules. Ref. [4] suggested that the complexity can be reduced by carefully identifying a minimal set of rate vectors that contribute to the capacity region. However, details are missing on how to do so. Ref. [5-7] developed suboptimal algorithms to trade off optimality for computational efficiency. Ref. [5] proposed a polynomial-time algorithm with a theoretically-proved approximation factor, assuming nodes are placed uniformly randomly in a unit square. Ref. [6] considered a column-generation method to iteratively determine a mixed schedule based on a small-size subset of basic schedules to reduce the primal-dual gap until it is less than a given tolerance. Ref. [7] developed a heuristic for a better initialization of the column-generation method and incorporated a Perron-Frobenius condition to further improve the computational efficiency. These prior papers assume pretty much a static scenario. By contrast, we consider the fast fading case. In addition, we focus on fast scheduling algorithms without sacrificing optimality (i.e., this paper is not about suboptimal heuristic algorithms, although the insights obtained here suggest ways to design good heuristics).

The major contribution of this paper is that we develop an efficient algorithm which reduces the runtime dramatically. It achieves its efficiency by removing a majority of the non-extreme-point schedules from consideration. The elimination process requires only simple channel statistics such as the stochastic means of the channel gains, and is robust to the variation of system parameters. Using the algorithm developed in this paper, it takes only **93.6ms** on average to compute the solution for the 15-link network mentioned above, beating the straightforward scheduling algorithm by a factor of **166**. More importantly, this computational time is much smaller than the targeted scheduling period (2s).

The remainder of this paper is organized as follows. Section II presents the system model and problem formulation. In section III, we derive simple criteria for eliminating non-extreme-point schedules and develop an efficient scheduling algorithm based on this elimination. Section IV concludes this paper and remarks on the possible future research directions.

II. System Model and Problem Formulation

A. Channel Fading Model

Consider a system consisting of n unidirectional links (transmitter-receiver pairs) labeled $1, ..., n$, sharing the radio spectrum. By transmitter i and receiver i, we mean the transmitter and receiver of link i, respectively. Transmitter i transmits at power P_i. Receiver j detects power of $P_i G_{ji}$ from transmitter i, where G_{ji} is the cross gain, which depends on the aggregate effects of path loss, log-normal shadowing and fast channel fading. Cross gain G_{ji} can be specified in the following form $G_{ji} = \beta_{ji}^S \beta_{ji}^F d_{ji}^{-\alpha}$ [8], where d_{ji} is the Euclidean distance between the transmitter i and receiver j; α is the path loss exponent; and β_{ji}^S and β_{ji}^F are the gains that correspond to channel fluctuations due to large-time-scale log-normal Shadowing and small-time-scale channel Fading, respectively. Within a scheduling period, we assume the relative distance between nodes and the log-normal shadowing gains remain unchanged. Thus, we have $G_{ji} = \overline{G_{ji}} \beta_{ji}^F$. The small-scale fading gain β_{ji}^F is normalized to a random number with unit mean, whereas the stochastic mean of G_{ji}, $\overline{G_{ji}} \triangleq E_{\beta_{ji}^F}[G_{ji}] = \beta_{ji}^S d_{ji}^{-\alpha}$, is regarded as a constant over the scheduling period and can be tracked by the scheduler. We also assume β_{ji}^F's are identically and independently distributed (i.i.d.) for different (i, j) pairs. Throughout this paper, our analysis will focus on the **Rayleigh fading model**. In this case, β_{ji}^F's are **exponentially distributed**.

B. Throughput Model

Note that there are altogether 2^n **basic schedules**: $\{1\}$, $\{2\}$, …, $\{1, 2\}$, …, $\{1, 2, \ldots, n\}$ and ϕ (empty set). They constitute the ***power set*** of $\{1, 2, \ldots, n\}$, denoted by $2^{\{1,\ldots,n\}}$. Associated with each basic schedule S_b is an $n \times 1$ **basic throughput vector** $\mathbf{T}^{(S_b)} = [T_1^{(S_b)}, T_2^{(S_b)}, \ldots, T_n^{(S_b)}]^T$, where $T_j^{(S_b)}$ is the throughput of link j achieved under S_b. Consider a transmitting link $j \in S_b$. Due to the cross gains between links and the ambient noise, receiver j receives its target signal (with power $G_{jj}P_j$) mixed with the cumulative interferences from other links ($\sum_{i\neq j, i\in S_b} G_{ji}P_i$) plus additive white Gaussian noise (with power N_j). We adopt the following conventional SINR-based cumulative-interference model and throughput model (called the "physical model" in [12]):

Each link $j \in S_b$ can successfully transmit at data rate R_j if its SINR is above its SINR threshold γ_j, i.e.,

$$SINR_j^{(S_b)} = G_{jj}P_j/(\sum\nolimits_{i\neq j, i\in S_b} G_{ji}P_i + N_j) > \gamma_j, \quad (1)$$

Otherwise, an outage occurs, resulting in packet corruption. The corrupted packets will be retransmitted in future frames. In this paper, we assume non-adaptive modulation between two successive schedule updates. During a scheduling period, the parameters ($\{\overline{G_{ji}}\}_{1\le i,j\le n}$, $\{P_j\}_{1\le j\le n}$, $\{N_j\}_{1\le j\le n}$, $\{R_j\}_{1\le j\le n}$ and $\{\gamma_j\}_{1\le j\le n}$) are assumed to be fixed finite positive numbers. The ensemble-average throughput of link j is then given by

$$T_j^{(S_b)} = R_j \Pr(\overline{G_{jj}}\beta_{jj}^F P_j > \gamma_j \sum_{i\in S_b, i\neq j} \overline{G_{ji}}\beta_{ji}^F P_i + \gamma_j N_j). \quad (2)$$

Assuming Rayleigh fading, (2) can be further expressed in a *product form* [13] according to the following lemma in [10]. As will be shown later, this nice structure can be exploited to develop fast scheduling algorithm.

Lemma 1: Suppose $z_1, \ldots, z_n$ are independent **exponentially** distributed random variables with mean $Ez_i = 1/\lambda_i$, and κ is a constant. Then

$$\Pr(z_1 > \sum\nolimits_{i=2}^{n} z_i + \kappa) = \exp(-\lambda_1\kappa) \prod\nolimits_{i=2}^{n} (1 + \lambda_1/\lambda_i)^{-1}.$$

With Lemma 1, we can prove that for each S_b,

$$T_j^{(S_b)} = \begin{cases} c_j \prod\limits_{\forall i\in S_b, i\neq j} a_{ji}, & \text{if } j \in S_b \\ 0, & \text{if } j \notin S_b \end{cases}, \quad (3)$$

where $c_j \triangleq R_j \exp(-\gamma_j N_j/P_j\overline{G_{jj}})$, $a_{ji} \triangleq (1 + \gamma_j B_{ji} P_i/P_j)^{-1}$ and

$$B_{ji} \triangleq \begin{cases} 0, & \text{if } j = i \\ \overline{G_{ji}}/\overline{G_{jj}}, & \text{if } j \neq i \end{cases}$$

is the normalized cross gain. The assumption of finite positive parameters implies

$$0 < B_{ji}, c_j < +\infty,\ 0 < a_{ji} < 1,\ \forall i, j, i \neq j, 1 \le i, j \le n. \quad (4)$$

A TDMA mixture of basic schedules can be used to generate a **mixed schedule**:

$$S_{mix} = \sum\nolimits_{S_b \in 2^{\{1,\ldots,n\}}} w^{(S_b)} S_b$$
$$(w^{(S_b)} \ge 0, \forall S_b \in 2^{\{1,\ldots,n\}}, \sum\nolimits_{S_b\in 2^{\{1,\ldots,n\}}} w^{(S_b)} \le 1),$$

in which the weight $w^{(S_b)}$ is the fraction of airtime allocated to S_b. For each S_{mix}, the associated throughput vector is

$$\mathbf{T}^{(S_{mix})} = \sum\nolimits_{S_b\in 2^{\{1,\ldots,n\}}} w^{(S_b)} \mathbf{T}^{(S_b)}. \quad (5)$$

The full set of $\mathbf{T}^{(S_{mix})}$ is the convex hull of $\{\mathbf{T}^{(S_b)}\}_{S_b\in 2^{\{1,\ldots,n\}}}$, denoted by $Co(\{\mathbf{T}^{(S_b)}\}_{S_b\in 2^{\{1,\ldots,n\}}})$, and it constitutes the **feasible rate region**.

C. Problem Formulation

Suppose that we need to satisfy a target throughput vector $\mathbf{T}^* = [T_1^*, T_2^*, \ldots, T_n^*]^T$, where T_j^* is the throughput demand of link j. Within a scheduling period, the scheduler needs to
1) determine whether $\mathbf{T}^*$ is feasible;
2) if feasible, find a **mixed schedule** that achieves $\mathbf{T}^*$ with minimal airtime consumption.

The scheduling problem can be cast as a standard-form linear program (LP) as follows:

$$\begin{aligned} \text{minimize:} \quad & w_{sum} = \sum\nolimits_{S_b\in 2^{\{1,\ldots n\}}} w^{(S_b)} \\ \text{subject to:} \quad & \sum\nolimits_{S_b\in 2^{\{1,\ldots n\}}} w^{(S_b)} \mathbf{T}^{(S_b)} = \mathbf{T}^* \quad (6) \\ & w^{(S_b)} \ge 0, \forall S_b \in 2^{\{1,\ldots n\}}. \end{aligned}$$

If the LP returns an optimal solution $w_{sum}^{opt} \le 1$, then $\mathbf{T}^*$ is feasible and can be achieved by a mixed schedule with minimal total airtime w_{sum}^{opt}; otherwise, $\mathbf{T}^*$ is infeasible.

In practice, it takes $O(n^2 v)$ operations to solve a standard-form LP with n constraints and v optimization variables[†]. Since $v = 2^n$, we need $O(n^2 2^n)$ operations to solve (6) when $\{\mathbf{T}^{(S_b)}\}_{S_b\in 2^{\{1,\ldots,n\}}}$is given, and another $O(n^2 2^n)$ operations to compute $\{\mathbf{T}^{(S_b)}\}_{S_b\in 2^{\{1,\ldots,n\}}}$ from (3). Thus, the overall complexity is $O(n^2 2^n)$.

From previous discussion, we know that this complexity is unacceptable even for a moderate network size $n = 15$, and a possible angle of attack is to reduce v by removing the non-extreme-point schedules from $2^{\{1,\ldots,n\}}$. Recall that an **extreme-point schedule** is a basic schedule which can not be replaced by a convex combination of other basic schedules. Its associated throughput vector is an **extreme point** of the feasible rate region. Imagine that we have a fictitious "**extreme-point oracle**" who can immediately tell the set of extreme-point schedules $\{S_{ep}\}$. Then v becomes $|\{S_{ep}\}|$, and both numbers of operations required for computing $\{\mathbf{T}^{(S_b)}\}_{S_b\in 2^{\{1,\ldots,n\}}}$ and solving LP reduce to $O(n^2|\{S_{ep}\}|)$, where $|\{S_{ep}\}|$ is the number of extreme-point schedules. Thus the scheduling efficiency is speeded up by a factor of the order

[†] Although the best known theoretical worst-case complexity bound is $O(nv\max^3(n, v))$ [14], the worse cases are rare in practice. The complexity could be "almost always" captured by $O(n^2 v)$ [15]. We consider this practical complexity bound throughout this paper.

of $2^n/|\{S_{ep}\}|$. If $|\{S_{ep}\}| << 2^n$, the improvement can be very significant. For example, with the help of the oracle, the expected computational time for the aforementioned 15-link network reduces to **93.2 ms** in our simulation experiment, with a speedup factor of around 166!

In reality, the "extreme-point oracle" does not exist and the scheduler has to identify the set of extreme points itself. However, identifying $\{S_{ep}\}$ (or $\{\mathbf{T}^{(S_{ep})}\}$) could be more time-consuming than solving (6) directly! Note that a basic throughput vector is an extreme point if and only if it can not be represented by a convex combination of other basic throughput vectors. Trivially, $\mathbf{T}^{(\phi)} = \mathbf{0}$ is an extreme point. To get the remaining $\{\mathbf{T}^{(S_{ep})}\}$, one need to solve the following LP, for each basic throughput vector $\mathbf{T}^{(S)}(S \in 2^{\{1,...,n\}}\backslash\phi)$:

$$\begin{aligned} \text{minimize:} \quad & w_{sum} = \sum\nolimits_{S_b \in 2^{\{1,...n\}}\backslash S} w^{(S_b)} \\ \text{subject to:} \quad & \mathbf{T}^{(S)} = \sum\nolimits_{S_b \in 2^{\{1,...n\}}\backslash S} w^{(S_b)}\mathbf{T}^{(S_b)} \qquad (7) \\ & w^{(S_b)} \geq 0, \forall S_b \in 2^{\{1,...n\}}\backslash S. \end{aligned}$$

$\mathbf{T}^{(S)}$ is **not** an extreme point if and only if (7) returns $w_{sum}^{opt} \leq 1^{\ddagger}$. LP (7) is quite similar to (6) except that $v = 2^n - 1$. The overall complexity for examining all 2^n basic throughput vector becomes $O(n^2 4^n)$, which is even worse than $O(n^2 2^n)$.

A question thus arises: is there any intelligent way to effectively reduce the complexity? The answer is positive, thanks to the nice mathematical structure of the product form given in (3). Details will be presented in next section.

III. Optimal Scheduling Based on Redundant-Schedule Removal: Criteria and Algorithm

This section is dedicated to the central concern of the fast optimal scheduling: finding criteria in terms of the parameters ($\{\overline{G_{ji}}\}_{1 \leq i,j \leq n}$, $\{P_j\}_{1 \leq j \leq n}$, $\{N_j\}_{1 \leq j \leq n}$, $\{R_j\}_{1 \leq j \leq n}$ and $\{\gamma_j\}_{1 \leq j \leq n}$) to efficiently eliminate redundant schedules from $2^{\{1,...,n\}}$.

A. A Simple Case

To gain some intuition, it would be helpful to consider the simple case of n = 2 first. There are four basic schedules: ϕ, $\{1\}$, $\{2\}$ and $\{1, 2\}$. From (3), the corresponding throughput vectors are $\mathbf{T}^{(\phi)} = [0,0]^T, \mathbf{T}^{(\{1\})} = [c_1, 0]^T, \mathbf{T}^{(\{2\})} = [0, c_2]^T$, $\mathbf{T}^{(\{1,2\})} = [c_1 a_{12}, c_2 a_{21}]^T$. For any mixed schedule $S_{mix} = w_0\phi + w_1\{1\} + w_2\{2\} + w_3\{1,2\}$, $w_0 + w_1 + w_2 + w_3 = 1$, the associated throughput vector is $\mathbf{T}^{(S_{mix})} = [w_1 c_1 + w_3 c_1 a_{12}, w_2 c_2 + w_3 c_2 a_{21}]^T$.

$\mathbf{T}^{(\{1\})}$ and $\mathbf{T}^{(\{2\})}$ are extreme points. This is because if $\mathbf{T}^{(\{1\})}$ were not an extreme point, one could find a mixed schedule S_{mix} with $w_1 = 0$ such that $\mathbf{T}^{(\{1\})} = \mathbf{T}^{(S_{mix})} = [w_3 c_1 a_{12}, w_2 c_2 + w_3 c_2 a_{21}]^T$; however, from (4), this solution does not exist. Similarly, $\mathbf{T}^{(\{2\})}$ is an extreme point. In fact, similar argument shows that all single-link schedules for a general network with $n \geq 2$ are extreme-point schedules.

The location of $\mathbf{T}^{(\{1,2\})}$ is crucial as to whether it is an extreme point. Specifically, $\mathbf{T}^{(\{1,2\})} = [T_1^{(\{1,2\})}, T_2^{(\{1,2\})}]^T$ is an extreme point if and only if it lies above the line segment joining $\mathbf{T}^{(\{1\})}$ and $\mathbf{T}^{(\{2\})}$, or equivalently, $T_1^{(\{1,2\})}/c_1 + T_2^{(\{1,2\})}/c_2 > 1$. After further simplification, the condition becomes $\gamma_1\gamma_2 B_{12} B_{21} < 1$. Thus, the quantity $\gamma_1\gamma_2 B_{12} B_{21}$ can be considered as a *discriminant* for extreme points for the case of $n = 2$.

B. Non-Extreme-Point-Schedule Removal

In the above case, $\{1, 2\}$ is not an extreme-point schedule when $\gamma_1\gamma_2 B_{12} B_{21} \geq 1$. This is due to the severe mutual interferences between active links as embodied in B_{12} and B_{21} relative to the SINR requirements as embodied in γ_1 and γ_2. For the general case, in place of $\gamma_1\gamma_2 B_{12} B_{21} < 1$, we have $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$ as a *necessary* condition for extreme points, as established in the key theorem below. This condition can be used to eliminate redundant non-extreme-point schedules.

Notations

Define the SINR diagonal matrix $\mathbf{\Gamma} \overset{\Delta}{=} diag(\gamma_1, \gamma_2, ..., \gamma_n)$ and normalized cross gain matrix $\mathbf{B} \overset{\Delta}{=} (B_{ji})_{n \times n}$. In the following, for an $n \times n$ nonnegative matrix $\mathbf{M} = (m_{jk})$ and a non-empty basic schedule $S = \{i_1, i_2, ..., i_L\} \in 2^{\{1,...,n\}}$, $\mathbf{M}_{(S)}$ denotes the submatrix of $\mathbf{M}$ resulting from deleting all its rows and columns except those labeled by $i_1, i_2, ..., i_L$ (e.g., $\mathbf{M}_{(\{2,5\})} = \begin{pmatrix} m_{22} & m_{25} \\ m_{52} & m_{55} \end{pmatrix}$); $\rho(\mathbf{M}_{(S)})$ denotes the spectral radius of $\mathbf{M}_{(S)}$. Similarly, for an n-dimensional vector $\mathbf{T}$, $\mathbf{T}_{(S)}$ denotes the subvector of $\mathbf{T}$ resulting from deleting all its elements except those labeled by $i_1, i_2, ..., i_L$. For component-wise matrix or vector comparison,

1) $\mathbf{A} \geq \mathbf{B}$ ($\mathbf{A}$ weakly dominates $\mathbf{B}$) implies $\mathbf{A} - \mathbf{B}$ is nonnegative;
2) $\mathbf{A} > \mathbf{B}$ ($\mathbf{A}$ strictly dominates $\mathbf{B}$) implies $\mathbf{A} \geq \mathbf{B}$ and at least one element of $\mathbf{A} - \mathbf{B}$ is positive;
3) $\mathbf{A} \succ \mathbf{B}$ ($\mathbf{A}$ component-wisely strictly dominates $\mathbf{B}$) implies each element of $\mathbf{A} - \mathbf{B}$ is positive.

Key Theorem 1: Consider a basic schedule S with L links, where $2 \leq L \leq n$. A necessary condition for it to be an extreme-point schedule is $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$.

The proof of the Key Theorem requires Lemmas 2 and 3 below. The proofs can be found in Appendices A and B.

Lemma 2: Any non-origin extreme point $\mathbf{T}^{(S_{ep})}$ must be **Pareto-efficient**, i.e., we **cannot** find a mixed schedule S_{mix} such that $\mathbf{T}^{(S_{mix})} > \mathbf{T}^{(S_{ep})}$.

Lemma 3: For an irreducible nonnegative square matrix $\mathbf{M}$, there exists a vector $\mathbf{x} > \mathbf{0}$ with components summing to 1 such that $\mathbf{Mx} \geq \mathbf{x}$ if and only if $\rho(\mathbf{M}) \geq 1$.

Comment: Lemma 3 is a corollary of the well-known **Perron-Frobenius theorem** (see Theorem 1.6 in [16]): An irreducible nonnegative square matrix $\mathbf{M}$ has a real positive eigenvalue $\rho(\mathbf{M})$ which is the spectral radius of $\mathbf{M}$. Furthermore, there is an eigenvector $\mathbf{x} > \mathbf{0}$ corresponding to

[‡] The "only if" part is obvious. For the "if" one, $\mathbf{T}^{(S)}$ must be a convex combination of other basic throughput vectors when $w_{sum}^{opt} \leq 1$. This is because one can always top up the airtime with the null basic schedule ϕ. Adding $(1 - w_{sum}^{opt})\mathbf{T}^{(\phi)}(= \mathbf{0})$ to the weighted sum in (7) will not affect the equality. This gives the "if" part.

$\rho(\mathbf{M})$. Lemma 3 broadens the application of Perron-Frobenius theorem, especially in proving Perron-Frobenius-type matrix inequalities.

Proof of the Key Theorem: The proof for $L = 2$ has been given in Section III.A. That is, $\gamma_1\gamma_2 B_{12}B_{21}$ is the spectral radius of concern. Without loss of generality, we consider the case $S = \{1, 2, 3\}$ ($L = 3$). The proofs for $L > 3$ are similar. From Lemma 2 and the definition of extreme-point schedules, if an L-link basic schedule S can be weakly outperformed by a mixed schedule which is a convex combination of its $(L-1)$-link subset schedules, then it could not be an extreme-point schedule. Following from this fact, if there exists nonnegative weights w_1, w_2 and w_3 summing to 1 such that $[\mathbf{T}^{(\{1,2,3\})}] \leq w_1[\mathbf{T}^{(\{2,3\})}] + w_2[\mathbf{T}^{(\{1,3\})}] + w_3[\mathbf{T}^{(\{1,2\})}]$, then $[\mathbf{T}^{(\{1,2,3\})}]$ could not be an extreme point.

Since (3) gives $[\mathbf{T}^{(\{1,2\})}]_{(\{1,2,3\})} = [c_1a_{12}, c_2a_{21}, 0]^T$, $[\mathbf{T}^{(\{1,3\})}]_{(\{1,2,3\})} = [c_1a_{13}, 0, c_3a_{31}]^T$, $[\mathbf{T}^{(\{2,3\})}]_{(\{1,2,3\})} = [0, c_2a_{23}, c_3a_{32}]^T$, and $[\mathbf{T}^{(\{1,2,3\})}]_{(\{1,2,3\})} = [c_1a_{12}a_{13}, c_2a_{21}a_{23}, c_3a_{31}a_{32}]^T$, the inequality above becomes

$$\begin{pmatrix} c_1a_{12}a_{13} \\ c_2a_{21}a_{23} \\ c_3a_{31}a_{32} \end{pmatrix} \leq \begin{pmatrix} 0 & c_1a_{13} & c_1a_{12} \\ c_2a_{23} & 0 & c_2a_{21} \\ c_3a_{32} & c_3a_{31} & 0 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$$

which gives

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} \leq \begin{pmatrix} 0 & a_{12}^{-1}-1 & a_{13}^{-1}-1 \\ a_{21}^{-1}-1 & 0 & a_{23}^{-1}-1 \\ a_{31}^{-1}-1 & a_{32}^{-1}-1 & 0 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}.$$

Define the power diagonal matrix $\mathbf{P} \triangleq diag(P_1, P_2, ..., P_n)$. Noting that $a_{ji}^{-1} - 1 = \gamma_j B_{ji} P_i / P_j > 0$, we have

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} \leq [\mathbf{P^{-1}\Gamma BP}]_{(\{1,2,3\})} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}. \tag{8}$$

Since $\mathbf{I_3} + [\mathbf{P^{-1}\Gamma BP}]_{(\{1,2,3\})} \succ \mathbf{0}$ (where $\mathbf{I_3}$ is a 3×3 identity matrix), $[\mathbf{P^{-1}\Gamma BP}]_{(\{1,2,3\})}$ is irreducible (from Theorem 1.3 in [16]). According to Lemma 3, a solution $[w_1, w_2, w_3]^T$ ($w_1 + w_2 + w_3 = 1$) of (8) exists if and only if $\rho([\mathbf{P^{-1}\Gamma BP}]_{(\{1,2,3\})}) = \rho([\mathbf{\Gamma B}]_{(\{1,2,3\})}) \geq 1$ (where the equality follows from the fact that similar matrices share the eigenvalues). Therefore, if S= {1, 2, 3} is an extreme-point schedule, we must have $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$. ■

A classical application of Perron-Frobenius theorem is the following well-known theorem: In the **non-fading** scenario with normalized cross gain matrix $\mathbf{B}$, the SINR requirements specified in the diagonal matrix $\mathbf{\Gamma}$ can be fulfilled via power allocation among a set of simultaneously transmitting links S if and only if $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$ [17].

Surprisingly, the same inequality appears in our key theorem under **Rayleigh fading** as a necessary condition for an extreme-point schedule. The fundamental cause is subject to the following interpretation. The magnitude of the spectral radius $\rho([\mathbf{\Gamma B}]_{(S)})$ is a reflection of the severity of the mutual interferences among links relative to the SINR requirements of the links. Thus, it is desirable for $\rho([\mathbf{\Gamma B}]_{(S)})$ to be small in both cases.

The necessary condition $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$ can be used to eliminate a large number of (although not all) non-extreme-point schedules. We discuss this elimination process under the context of **Possibly-Extreme-Point Approximation** below. We will demonstrate by simulation experiments that the elimination process eliminates most of the non-extreme points.

Possibly-Extreme-Point Approximation

*Definition 1 (**Possibly-Extreme-Point Schedule**):* A basic schedule S is said to be a possibly-extreme-point schedule if (i) $S \in \{\{1\}, \{2\}..., \{n\}, \phi\}$; or

(ii) $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$, when $S \notin \{\{1\}, \{2\}..., \{n\}, \phi\}$.

Let $\{S_{pep}\}$ denote the set of possibly-extreme-point schedules. Since $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$ is a necessary but not sufficient condition for extreme-point identification, $\{S_{pep}\}$ is in general a superset of $\{S_{ep}\}$. However, we will show that $\{S_{pep}\}$ has two nice properties:

(A) Its identification is rather efficient.

(B) It well approximates $\{S_{ep}\}$.

First, there is a very effective way to check whether $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$, thanks to Lemma 4 below:

Lemma 4 (Theorem 2.2 in [18]*):* $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$ if and only if all ***leading principal minors*** of matrix $[\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)}$ are positive, where $\mathbf{I_n}$ is an $n \times n$ identity matrix.

Lemma 4 is a pure mathematical theorem. Our contribution is our discovery of its application in $\{S_{pep}\}$ identification. Particularly, we develop simple criteria and an efficient algorithm to identify $\{S_{pep}\}$ in a recursive manner, as follows. Let $S = \{i_1, i_2, ..., i_{L-1}, i_L\}$. Then from Lemma 4, we have

$$\begin{aligned} & \rho([\mathbf{\Gamma B}]_{(S)}) < 1 \\ \Leftrightarrow\ & \text{all leading principal minors of } [\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)} \text{ are positive} \\ \Leftrightarrow\ & \text{all leading principal minors of } [\mathbf{I_n} - \mathbf{\Gamma B}]_{(\{i_1, i_2, ..., i_{L-1}\})} \text{ are} \\ & \text{positive and } \det[\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)} > 0 \\ \Leftrightarrow\ & \rho([\mathbf{\Gamma B}]_{(\{i_1, i_2, ..., i_{L-1}\})}) < 1 \text{ and } \det[\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)} > 0. \end{aligned}$$

Noting that the permutation of links in S is immaterial to the value of $\rho([\mathbf{\Gamma B}]_{(S)})$ and $\det[\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)}$, we have

$$\rho([\mathbf{\Gamma B}]_{(S)}) < 1 \Leftrightarrow \rho([\mathbf{\Gamma B}]_{(S')}) < 1 \text{ and } \det[\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)} > 0,$$

where S' is an arbitrary $(L-1)$-link subset schedule of S. This result leads to the following criteria:

Consider a basic schedule S with $L > 1$ links.

PEP-Criterion 1: S is a possibly-extreme-point schedule if and only if it has an associated $(L-1)$-link subset schedule that is a possibly-extreme-point schedule and $\det[\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)} > 0$.

PEP-Criterion 2: If S is a possibly-extreme-point schedule, then each and every of its non-trivial-subset schedules is a possibly-extreme-point schedule. If S is not a possibly-extreme-point schedule, then all of its strict-superset schedules are not possibly-extreme-point schedules.

Criterion 1 enables us to verify possibly-extreme-point schedules in a recursive manner by determinant computation. Criterion 2 has the potential to significantly reduce the number of determinant computation. Based on these two criteria, we develop the following algorithm to identify $\{S_{pep}\}$:

IPEP: Algorithm for Identifying a Set of Possibly-Extreme-Point Schedules

```
Let $\{S_{ud}\}$ and $\{S_{npep}\} \triangleq 2^{\{1,...,n\}} \backslash \{S_{pep}\}$ be a set of
undetermined schedules and a set of
non-possibly-extreme-point schedules respectively;
Input: $\mathbf{\Gamma}$, $\mathbf{B}$, $2^{\{1,...,n\}}$
Initialization: $\{S_{pep}\} = \{\{1\},...,\{n\},\phi\}$, $\{S_{npep}\} = \phi$,
$\{S_{ud}\} = 2^{\{1,...,n\}} \backslash \{\{1\},...,\{n\},\phi\}$;
while $\{S_{ud}\} \neq \phi$ do
    Find a $S \in \{S_{ud}\}$ with minimum number of links;
    if $\det [\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)} \leq 0$ then
        move all superset schedules of $S$ from $\{S_{ud}\}$ to
        $\{S_{npep}\}$;
    else
        move $S$ from $\{S_{ud}\}$ to $\{S_{pep}\}$;
    end
end
Output: $\{S_{pep}\}$
```

IPEP demonstrates very high computational efficiency in our simulation experiments. For example, in a 15-link network, it takes **21**μs on average to identify $\{S_{pep}\}$. Based on this quick identification, the average time to solve the optimal scheduling problem reduces from 15.5s to 93.6ms (see Table IV and the discussion in Section III.C for details). It is difficult to derive the theoretical complexity bound for IPEP, but we could still make some simple analysis. As can be readily seen, the computation of $\det [\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)}$ dominates the complexity of IPEP, which can be conducted by Gaussian elimination in $O(L^2)^{\dagger\dagger}$ floating-point operations for any L-link schedule S. For each S, at most one calculation of $\det [\mathbf{I_n} - \mathbf{\Gamma B}]_{(S)}$ is required. Hence, an upper bound of the complexity is $O(n^2 2^n)$. However, this bound is quite loose, because many basic schedules are removed from consideration with the help of criterion 2. An $O(n^2 |\{S_{pep}\}|)$ empirical complexity bound is observed from simulation experiments, as depicted in Table I. The parameter settings of simulation experiments are detailed at the end of this subsection.

We next argue that $\{S_{pep}\}$ is a good approximation of $\{S_{ep}\}$, and therefore further reducing the size of $\{S_{pep}\}$ to $\{S_{ep}\}$ is not worthwhile. We note that $\{\{1\}, \{2\}..., \{n\}, \phi\} \subseteq \{S_{ep}\} \subseteq \{S_{pep}\} \subseteq 2^{\{1,...,n\}}$. To see $|\{S_{pep}\}| \approx |\{S_{ep}\}|$, consider the **penalty factor** defined as $f_{penalty} \triangleq |\{S_{pep}\}| / |\{S_{ep}\}|$. The penalty factor characterizes the computational penalty incurred by the possibly-extreme-point approximation in solving the optimal scheduling problem (the ideal, but not achievable, complexity being that in which the extreme-point oracle is available). From the simulation results of IPEP in Table II, one can check that $f_{penalty}$ is very close to 1. Specifically, for networks with 9 to 15 links, the largest $f_{penalty}$ observed is 1.0635. These numerical results strongly indicate that $\{S_{pep}\}$ well approximates $\{S_{ep}\}$.

TABLE I
NUMBER OF FLOATING-POINT OPERATIONS OF IPEP (#OP) VERSUS NUMBER OF LINKS

n	**9**	**11**	**13**	**15**
$\overline{\lvert\{S_{pep}\}\rvert}$	50.95	96.12	167.254	292.462
$\max \lvert\{S_{pep}\}\rvert$	102	203	334	606
$\lvert\{S_{pep}\}\rvert_{99\%}$	88	165	299	534
$\overline{\#OP}$	1274.5	3689.9	9014.3	21287
max#OP	2872	9140	20488	49628
$\#OP_{99\%}$	2482	7210	17732	43044
$\frac{\overline{\#OP}}{n^2 \overline{\lvert\{S_{pep}\}\rvert}}$	0.3088	0.3173	0.3189	0.3235
$\frac{\max \#OP}{n^2 \max \lvert\{S_{pep}\}\rvert}$	0.3476	0.3721	0.363	0.364
$\frac{\#OP_{99\%}}{n^2 \lvert\{S_{pep}\}\rvert_{99\%}}$	0.3482	0.3611	0.3509	0.3583

We show that $\#OP = O(n^2 |\{S_{pep}\}|)$ from a simulation experiment with 500 parallel runs. Statistics $\overline{X}$, $\max X$ and $X_{99\%}$ denote the mean, maximum and 99^{th} percentile of samples from random variable X, respectively.

TABLE II
PENALTY FACTOR OF IPEP $f_{penalty}$ VERSUS NUMBER OF LINKS

n	**9**	**11**	**13**	**15**
$\overline{f_{penalty}}$	1.0041	1.0072	1.0100	1.0131
$\max f_{penalty}$	1.0606	1.0635	1.0411	1.0421
$f_{penalty\_99\%}$	1.0500	1.0400	1.0361	1.0388
$\mathrm{std} f_{penalty}$	0.0098	0.0097	0.0091	0.0085

We show that $f_{penalty}$ is close to 1 from a simulation experiment with 500 parallel runs. The statistic $\mathrm{std} f_{penalty}$ denotes the standard deviation of random variables $f_{penalty}$.

Parameter Settings in Simulation Experiments

We demonstrate by simulation experiments that IPEP is an effective algorithm for the elimination of redundant schedules. We first describe the settings for the simulation experiments that yield Tables I and II, and the settings for additional experiments.

For Tables I and II, the path loss exponent $\alpha = 2$, the log-normal shadowing gain is fixed at a unit constant, $\beta_{ji}^S = 1$, and Rayleigh fading is assumed. All transmitters transmit at unit power. All receivers set the SINR threshold to 10 and see a noise with 0.01 power unit. The transmitters are placed

[††] It is well known that the determinant-calculation of an $L \times L$ matrix normally requires $O(L^3)$ floating-point operations using Gaussian-elimination-based algorithms (say, using the LU-decomposition method). The complexity reduces to $O(L^2)$, however, if one has already performed LU-decomposition to any of its $(L-1) \times (L-1)$ principal minors and stored the results. This is a reason to start IPEP with a basic schedule with the minimum number of links.

randomly within a unit square, according to i.i.d. uniform distribution. The receiver of each transmitter is placed randomly on a circle of radius $r = 0.25\sqrt{\ln(n)/n}$ centered on the transmitter. Note that $\kappa_0\sqrt{\ln(n)/n}$ is the transmission radius that can guarantee connectivity with high probability in various random networks [12], where the constant leading coefficient κ_0 depends on node distribution. For $r = \kappa\sqrt{\ln(n)/n}$, larger κ results in better connectivity, larger mutual interferences, and smaller $|\{S_{ep}\}|$.We choose $\kappa = 0.25$ so that $|\{S_{ep}\}|$ falls between the two trivial extreme cases of $|\{S_{ep}\}| = n+1$ and $|\{S_{ep}\}| = 2^n$.

We also performed simulation experiments with other parameter settings. Specifically, we try various scenarios with different node distributions and topologies (e.g., nodes arranged in a grid, a chain, a ring, etc), different numbers of links n, different path loss exponents α, different transmit-power-allocation schemes $\{P_i\}_{1\le i\le n}$, different SINR thresholds $\{\gamma_i\}_{1\le i\le n}$, different noise powers $\{N_i\}_{1\le i\le n}$, etc. Interestingly, similar results are observed and they are rather insensitive to the parameter settings. Table III compares the penalty factors of some of these scenarios.

TABLE III
PENALTY FACTORS OF IPEP $f_{penalty}$ FOR VARIOUS SCENARIOS WHERE 9 LINKS ARE ASSUMED

scenarios	a	b	c	d
$\overline{f_{penalty}}$	1.0041	1.0057	1.0056	1
$\max f_{penalty}$	1.0606	1.0993	1.0638	1
$f_{penalty\_99\%}$	1.0500	1.0684	1.0500	1
$\text{std}f_{penalty}$	0.0098	0.0154	0.0110	0

We show that $f_{penalty}$ is still close to 1.

Scenarios:
a. The same case of n = 9 as in Table II.
b. Same as scenario *a* except that the path loss exponent is 4.
c. Same as scenario *a* except that the transmit powers are not fixed to 1, but are i.i.d. uniformly distributed within $[1, 2]$.
d. Same as scenario *a* except that the radius of the circle for placement of receiver is doubled.

C. Fast Scheduling Algorithm and Performance Evaluation

Building on IPEP, we propose the following algorithm to solve the optimal scheduling problem:

SPEP: Algorithm of Optimal Scheduling Based on Possibly Extreme Points:
Step 1 (Preprocessing): Run IPEP algorithm to quickly identify $\{S_{pep}\}$;
Step 2 (Throughput computation): Compute $\{\mathbf{T}^{(S_{pep})}\}$ according to (3);
Step 3 (LP optimization): Solve the following LP:

$$\begin{aligned} \text{minimize:} \quad & w_{sum} = \sum\nolimits_{S_b\in\{S_{pep}\}} w^{(S_b)} \\ \text{subject to:} \quad & \sum\nolimits_{S_b\in\{S_{pep}\}} w^{(S_b)}\mathbf{T}^{(S_b)} = \mathbf{T}^* \qquad (9) \\ & w^{(S_b)} \ge 0, \forall S_b \in \{S_{pep}\}. \end{aligned}$$

If (9) returns $w_{sum}^{opt} \le 1$, then $\mathbf{T}^*$ is feasible and can be achieved by a mixed schedule with total airtime w_{sum}^{opt}; otherwise, $\mathbf{T}^*$ is infeasible.

Compared with the algorithm that solves the LP in (6) directly, SPEP contains an additional preprocessing step to quickly remove non extreme points. From the previous discussion, all three steps in our algorithm share the same complexity order: $O(n^2|\{S_{pep}\}|)$. Therefore, the overall complexity is also $O(n^2|\{S_{pep}\}|)$. Note that we have shown that $|\{S_{pep}\}| \approx |\{S_{ep}\}|$ or $f_{penalty} \approx 1$, so that the complexity is practically $O(n^2|\{S_{ep}\}|)$, i.e., our algorithm achieves almost the same computational efficiency as that achieved by a fictitious extreme-point oracle, beating the direct LP algorithm by an improvement factor of $O(2^n/|\{S_{ep}\}|)$. As observed in most of our simulation results, $|\{S_{ep}\}| = O(c^n)$, where $c < 2$ is a constant depending on the system settings (e.g., we observe $|\{S_{ep}\}| \approx |\{S_{pep}\}| = O(1.33^n)$ in Table I). Although $|\{S_{ep}\}|$ is exponential in n, the improvement factor $O([2/c]^n)$ is still very significant.

Additional numerical results are given in Table IV. The system settings in Table IV are the same as those in Tables I and II. In Table IV, t_{pp1}, t_{tpc1} and t_{lpo1} denote the runtimes of preprocessing, throughput computation, and LP optimization in SPEP respectively; while t_{tpc2} and t_{lpo2} denote the runtimes of throughput computation and LP optimization in the direct-LP algorithm respectively. The total runtimes of two algorithms are t_{tot1} and t_{tot2}. We consider their sample means and 99th percentiles. We can see that the improvement factor grows very fast as n increases. The preprocessing procedure consumes only a tiny portion of the overall runtime in our algorithm.

TABLE IV
SIMULATION RUNTIME VERSUS NUMBER OF LINKS

n	9	11	13	15
$\overline{t_{pp1}}$	$1.2\mu s$	$3.7\mu s$	$9.0\mu s$	$21\mu s$
$\overline{t_{tpc1}}$	$3.13ms$	$6.16ms$	$11.4ms$	$23.7ms$
$\overline{t_{lpo1}}$	$38.0ms$	$42.3ms$	$55.3ms$	$69.9ms$
$\overline{t_{tot1}}$	$41.1ms$	$48.4ms$	$66.7ms$	$93.6ms$
$\overline{t_{tpc2}}$	$68.0ms$	$267ms$	$1.26s$	$6.43s$
$\overline{t_{lpo2}}$	$94.4ms$	$274ms$	$1.38s$	$9.06s$
$\overline{t_{tot2}}$	$163ms$	$541ms$	$2.64s$	$15.5s$
$\overline{t_{tot2}}/\overline{t_{tot1}}$	3.95	11.2	39.6	166
$t_{tot1\_99\%}$	$78.1ms$	$78.1ms$	$125ms$	$172ms$
$t_{tot2\_99\%}$	$219ms$	$703ms$	$3.31s$	$21.1s$
$t_{tot2\_99\%}/t_{tot1\_99\%}$	2.80	9.00	26.5	123

We show that SPEP (labeled by 1 in subscripts of parameters) yields significant improvement in computational efficiency over the direct-LP algorithm (labeled by 2).

D. Scheduling with General Objectives

The analysis so far has focused on developing fast scheduling algorithm to minimize airtime usage. In general, "**optimal STDMA scheduling**" could be associated with various optimization objectives. In this subsection, we briefly discuss the generalization of SPEP.

Conventionally, "**optimal STDMA scheduling**" deals with the following two types of problems:
I. Given a target throughput vector $\mathbf{T}^*$, determine whether and how it can be achieved by scheduling with certain objective (note: a special case, airtime minimization, was chosen as a focus in the preceding sections);
II. Given an objective that is a function of the throughput vector (e.g., total system throughput, proportional fairness utility, etc), find the optimal mixed schedule.

The optimization variables for I and II are the same: weights (airtimes) of basic schedules. Thus, for both I and II, complexity can be reduced by cutting down the number of optimization variables based on $\{S_{pep}\}$ identification. For instance, consider II. Let $f(\mathbf{T}) : \mathbb{R}^n \to \mathbb{R}$ be the objective function to be maximized. The scheduling problem can be formulated as

$$\text{maximize: } g(\{w^{(S_b)}\}_{S_b \in 2^{\{1,\dots,n\}}}) \triangleq f(\sum_{S_b \in 2^{\{1,\dots,n\}}} w^{(S_b)} \mathbf{T}^{(S_b)})$$

$$\text{subject to: } \sum\nolimits_{S_b \in 2^{\{1,\dots,n\}}} w^{(S_b)} = 1$$
$$w^{(S_b)} \geq 0, \forall S_b \in 2^{\{1,\dots,n\}}.$$

With IPEP, one can identify $\{S_{pep}\}$ and the above scheduling problem can be simplified to

$$\text{maximize: } g_1(\{w^{(S_b)}\}_{S_b \in \{S_{pep}\}}) \triangleq f(\sum_{S_b \in \{S_{pep}\}} w^{(S_b)} \mathbf{T}^{(S_b)})$$

$$\text{subject to: } \sum\nolimits_{S_b \in \{S_{pep}\}} w^{(S_b)} = 1$$
$$w^{(S_b)} \geq 0, \forall S_b \in \{S_{pep}\}.$$

We can then adapt SPEP to solve the above optimization problem (details omitted here to conserve space). Note that the computational complexities of most optimization problems are at least linear in the number of optimization variables. Therefore, the improvement factors of SPEP are linear or more than linear in $2^n/|\{S_{pep}\}|$.

Before leaving this part, we note that the set of extreme points plays a critical role in general fast optimal scheduling throughout this paper. It would be helpful to present more details here. Consider the case where $f(\mathbf{T})$ is component-wisely increasing with $\mathbf{T}$, which is the focus of most relevant studies. It is well known that for such objectives, the optimal solution must occur on the **Pareto-efficient frontier** of the feasible rate region; and conversely, for each point $\mathbf{T_o}$ on the Pareto frontier, there exists an $f(\mathbf{T})$ whose optimizer is $\mathbf{T_o}$. Note from Lemma 2 that each and every non-origin extreme point must be located on the Pareto frontier. From Caratheodory's theorem [15], any Pareto-efficient $\mathbf{T_o}$ can be represented by a convex combination of at most n extreme points. Therefore, **the set of extreme points concisely captures both computational efficiency and Pareto efficiency, and is the core of "fast optimal scheduling".**

IV. Concluding Remark

The investigations in this paper have been a first attempt to address the fast optimal STDMA scheduling problem according to the stochastic behavior of fast fading. We realized that the set of extreme-point schedules captures both aspects of computational efficiency and Pareto efficiency; and the key is to find an efficient method to identify this set or a slightly larger superset. We discovered that a simple Perron-Frobenius condition $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$ achieves this goal effectively:

1) It can eliminate a majority of non-extreme-point schedules, regardless of the setting of system parameters.
2) The elimination process is rather efficient and requires only simple channel statistics such as the ratios of stochastic means of the channel gains.

The overall complexity then reduces from $O(n^2 2^n)$ to $O(n^2|\{S_{ep}\}|)$, where n is the number of links and $|\{S_{ep}\}|$ is the number of extreme points.

It has not escaped our attention that $|\{S_{ep}\}|$ varies with system parameters and could be exponential in n, and as such the complexity could still be high. However, in view of the fact that the set of non-trivial extreme points constitute the **bases** of the Pareto frontier, it is doubtful that this complexity can be avoided without additionally exploitable structures. Fortunately, for modest-size networks (say, $n = 20$), online scheduling is very quick with respect to the assumed scheduling period.

For larger-size networks, the findings in this paper could still serve as stepping stones for further investigations. On one hand, our optimal algorithm could serve as a performance benchmark for heuristic algorithms. On the other hand, the interference metric $\rho([\mathbf{\Gamma B}]_{(S)})$ gives us a hint on the design of fast heuristics to strike a proper balance between optimality and complexity. For instance, we may carefully select $\theta_{\min}$ and $\theta_{\max}$ such that $0 \leq \theta_{\min} < \theta_{\max} \leq 1$ and compute the suboptimal solution based on a condition $\rho([\mathbf{\Gamma B}]_{(S)}) \in (\theta_{\min}, \theta_{\max})$ instead of $\rho([\mathbf{\Gamma B}]_{(S)}) < 1$, which yields to a small subset of basic schedules to be considered.

Last but not least, while the results in this paper are based on the Rayleigh fading model, we conjecture that they are still applicable under other small-scale-fading models (such as Rician, Nakagami-m, etc.). This is corroborated by observations from extensive simulation experiments detailed in our technical report [19]. We believe that there is a unified mathematical framework to explain these observations.

Appendix A
Proof of Lemma 2

The proof of Lemma 2 requires the following pre-lemmas:

Pre-lemma 1: For two non-empty basic schedules S_b and $S_b{}'$, if $S_b \subset S_b{}'$, then $[\mathbf{T}^{(S_b{}')}]_{(S_b)} \prec [\mathbf{T}^{(S_b)}]_{(S_b)}$.

Comment: The physical meaning is straightforward: Adding additional transmitting links to a basic schedule S_b will ***strictly decrease*** the throughput of original links.

Proof of Pre-lemma 1: It suffices to show that adding an additional link to a non-empty basic schedule S_b decreases the throughput of original links. With no loss of generality, we consider $S_b = \{1, ..., L-1\}$ with link L being the additional link. Then $S_b{}' = \{1, ..., L\}$. Let $T_{1:L}$ denote $[T_1, T_2, ..., T_L]^T$. From (3), we have $[\mathbf{T}^{(S_b{}')}]_{1:L-1} = diag(a_{1L}, ..., a_{(L-1)L})[\mathbf{T}^{(S_b)}]_{1:L-1}$. Recall that $0 < a_{iL} < 1, \forall 1 \le i \le L-1$ in (4), we have $[\mathbf{T}^{(S_b{}')}]_{(S_b)} = [\mathbf{T}^{(S_b{}')}]_{1:L-1} \prec [\mathbf{T}^{(S_b)}]_{1:L-1} = [\mathbf{T}^{(S_b)}]_{(S_b)}$.

■

Pre-lemma 2: For a pair of throughput vectors $\mathbf{T}'$ and $\mathbf{T}'' \neq \mathbf{0}$, if $\mathbf{T}' < \mathbf{T}''$ and $\mathbf{T}''$ is feasible (i.e., $\mathbf{T}'' \in Co(\{\mathbf{T}^{(S_b)}\}_{S_b \in 2^{\{1,...,n\}}})$), then $\mathbf{T}'$ must also be feasible.

Proof of Pre-lemma 2: This is obvious for $n = 1$. By mathematical induction, suppose that Pre-lemma 2 holds for $n = 1, ..., l-1$. For $n = l$, we only need to consider the case where $\mathbf{T}'' \succ \mathbf{0}$ (Otherwise, if, say, $T_l'' = 0$, then it reduces to the case of $n = l-1$). Define $\eta \triangleq \min\limits_{1 \le i \le l} [T_i''/(T_i'' - T_i')]$ (note: $\eta \ge 1$), and without loss of generality, assume that $l = \arg\limits_i \min\limits_{1 \le i \le l} [T_i''/(T_i'' - T_i')]$. Define $\mathbf{T}''' \triangleq \mathbf{T}'' - \eta(\mathbf{T}'' - \mathbf{T}') < \mathbf{T}''$. It is easy to see that $\mathbf{T}'''$ is the intersection of the ray passing through $\mathbf{T}''$ and $\mathbf{T}'$, and the hyperplane $T_l = 0$ (where the l-dimensional Cartesian coordinate system is represented by an l-tuple $\mathbf{T} = (T_1, ..., T_l)$). Note that $\mathbf{T}'$ is a convex combination of $\mathbf{T}''$ and $\mathbf{T}'''$ (i.e., $\mathbf{T}' = \frac{1}{\eta}\mathbf{T}''' + (1 - \frac{1}{\eta})\mathbf{T}''$), and therefore if $\mathbf{T}''$ and $\mathbf{T}'''$ are feasible, so is $\mathbf{T}'$. Hence, to complete the proof, it suffices to show that $\mathbf{T}''' \in Co(\{\mathbf{T}^{(S_b)}\}_{S_b \in 2^{\{1,...,l\}}})$. Since $T_l''' = 0$, we need only to show that $\mathbf{T}_{1:l-1}'''$ is feasible in the case of $n = l-1$. Note that $\mathbf{0} \le \mathbf{T}_{1:l-1}''' \le \mathbf{T}_{1:l-1}''$ and $\mathbf{T}_{1:l-1}'' \in Co(\{\mathbf{T}_{1:l-1}^{(S_b)}\}_{S_b \in 2^{\{1,...,l\}}})$. From Pre-lemma 1 and the induction hypothesis of $n = l-1$, the $(l-1)$-dimensional convex hull $Co(\{\mathbf{T}_{1:l-1}^{(S_b)}\}_{S_b \in 2^{\{1,...,l-1\}}}) = Co(\{\mathbf{T}_{1:l-1}^{(S_b)}\}_{S_b \in 2^{\{1,...,l\}}})$. Therefore, $\mathbf{T}_{1:l-1}''$ is feasible in the case of $n = l-1$. So is $\mathbf{T}_{1:l-1}'''$, according to the induction hypothesis of $n = l-1$.

■

Proof of Lemma 2: Consider a feasible $\mathbf{T}^{(S)} \neq \mathbf{0}$. We need to show that if it is not Pareto-efficient, then it is not an extreme point. This is obvious for $n = 1$. By mathematical induction, suppose it is valid for $n = 1, ..., l-1$. For $n = l$, we only need to consider the case where $\mathbf{T}^{(S)} \succ \mathbf{0}$. If $\mathbf{T}^{(S)}$ is not Pareto-efficient, there exists $\mathbf{T}^{(S_e)}$ on the Pareto frontier such that $0 \prec \mathbf{T}^{(S)} < \mathbf{T}^{(S_e)}$. Let $\mathbf{T}^{(S_e)}$ and $\mathbf{T}^{(S)}$ be $\mathbf{T}''$ and $\mathbf{T}'$ in the proof of Pre-lemma 2, respectively. Then we can find a feasible $\mathbf{T}'''$ with $T_l''' = 0$ such that $\mathbf{T}^{(S)}$ lies **strictly** inside the segment connecting $\mathbf{T}^{(S_e)}$and $\mathbf{T}'''$. Hence $\mathbf{T}^{(S)}$ could not be an extreme point.

■

Appendix B
Proof of Lemma 3

Proof: If-Obvious from the Perron-Frobenius Theorem. Only if-Directly from the second point of Theorem 1.5 (equation (1.20)) and equation (1.33) in [16], stating that $\rho = \rho(\mathbf{M})$ is the largest real number for a solution $\mathbf{x}$ of inequality $\mathbf{Mx} - \rho\mathbf{x} \ge \mathbf{0}$ to exist, where $\mathbf{x} > \mathbf{0}$ and its components are summing to 1. Therefore, if a solution of $\mathbf{Mx} - \mathbf{x} \ge \mathbf{0}$ exists, we must have $\rho(\mathbf{M}) \ge 1$.

■

References

[1] R. Nelson and L. Kleinrock, “Spatial-TDMA: A Collison-free Multihop Channel Access Protocol”, *IEEE Transactions on Communication*, Vol. 33, pp. 934-944, Sept. 1985.

[2] K. Jain, J. Padhye, V. Padmanabhan, L. Qiu, “Impact of Interference on Multi-hop Wireless Network Performance”, ACM MOBICOM 2003.

[3] S. A. Borbash and A. Ephremides, “The Feasibility of Matchings in a Wireless Network”, *IEEE Transactions on Information Theory*, vol. 52, no. 6, June 2006, pp. 2749-2755.

[4] S. Toumpis and A. J. Goldsmith, “Capacity Regions for Wireless Ad Hoc Networks”, *IEEE Transactions on Wireless Communications*, vol. 2, no. 4, July 2003, pp. 736-748.

[5] G. Brar, D. M. Blough and P. Santi, “Computational Efficient Scheduling with the Physical Interference Model for Throughput Improvement in Wireless Mesh Networks”, ACM MOBICOM 2006.

[6] P. Soldati, B. Johansson and M. Johansson, “Proportionally Fair Allocation of End-to-End Bandwidth in STDMA Wireless Networks”, ACM MOBIHOC 2006.

[7] L. Fu, S. C. Liew and J. Huang, “Joint Power Control and Link Scheduling in Wireless Networks for Throughput Optimization”, IEEE ICC 2008.

[8] T. S. Rappaport, *Wireless Communications Principles and Practice.* Prentice Hall, 2nd ed., 2001

[9] Jean-Paul M.G. Linnartz, JPL’s Wireless Communication Reference Website, http://www.wirelesscommunication.nl/

[10] S. Kandukuri and S. Boyd, “Optimal Power Control in Interference-Limited Fading Wireless Channels with Outage-Probability Specifications”, *IEEE Transactions on Wireless Communications*, Vol. 1, No. 1, Jan. 2002.

[11] S. C. Liew and Y. J. Zhang, “Proportional Fairness in Multi-channel Multi-rate Wireless Networks - Part II: The Case of Time-Varying Channels,” to appear in *IEEE Transactions on Communications*.

[12] P. Gupta and P.R. Kumar, “The Capacity of Wireless Networks,” *IEEE Transactions on Information Theory*, Vol.46, No. 2, pp. 388-404, Mar. 2000.

[13] J.-P.M.G. Linnartz, “Slotted ALOHA Land-Mobile Radio Networks with Site Diversity”, IEE PROCEEDINGS-I, Vol. 139, No. I, Feb. 1992.

[14] D. Bertsimas and X. Luo, “On the Worst Case Complexity of Potential Recudtion Algorithms for Linear Programming”, Mathematical Programming, 77(2):321-333, 1997

[15] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004

[16] T. Kaczorek, *Positive 1D and 2D Systems.* Springer-Verlag London Limited, c2002

[17] J. Zander, “Performance of Optimum Transmitter Power Control in Cellular Radio systems”, *IEEE Transactions on Vehicular Technology*, vol. 41, no. 1. pp. 57-62. Feb. 1992.

[18] E. Seneta, *Non-Negative Matrices and Markov Chains.* Springer-Verlap New York, c1981, 2^{nd} ed.

[19] J. Zhang, S. C. Liew and L. Fu, “On Fast Optimal Scheduling over Fading Wireless Channels”, *Technical Report*, available at http://www.ie.cuhk.edu.hk/fileadmin/staff_upload/soung/Journal/ZhangLiewFuTRInfoCom08.pdf.