Adaptive Multi-pattern Reuse in Multi-cell Networks

Kyuho Son, Yung Yi and Song Chong
School of Electrical Engineering and Computer Science, KAIST
E-mail: skio@netsys.kaist.ac.kr, {yiyung, song}@ee.kaist.ac.kr

***Abstract*—Achieving sufficient spatial capacity gain by having small cells requires careful treatment of inter-cell interference (ICI) management via BS power coordination coupled with user scheduling inside cells. Optimal algorithms have been known to be hard to implement due to high computation and signaling overheads. We propose joint pattern-based ICI management and user scheduling algorithms that are practically implementable. The basic idea is to decompose the original problem into two sub-problems, where we run ICI management at a slower time scale than user scheduling. We empirically show that even with such a slow tracking of system dynamics at the ICI management part, the decomposed approach achieves high performance increase, compared to a conventional universal reuse scheme.**

I. Introduction

To achieve high spatial capacity, wireless cellular networks consider the dense deployment of base stations (BSs) that cover small cells. As a consequence, inter-cell interference (ICI) from neighboring BSs becomes a major source of performance degradation and the portion of users whose capacity is naturally limited by ICI grows. In order to fully attain the potential gain of multi-cell networks, the coordination of transmissions among BSs which can effectively manage ICI is essential. The key intuition of BS coordination is that the achievable rates, which depend on the amount of ICI, can be increased by turning off some of neighboring BSs. Thus there are cases when the increment of achievable rates preponderates the sacrifice of taking away transmission opportunities at the neighboring BSs. In particular, this usually happens to users at cell edges severely suffering from the ICI since the increment of achievable rates may be sufficiently large.

A brute-force approach for mitigating ICI is the use of traditional reuse scheme in time and/or frequency domain. However, this may waste precious radio resource since users at different geographical locations inside cells prefer different reuse schemes. Several schemes, e.g., fractional frequency reuse (FFR) [1] in Mobile WiMAX, have been proposed to accommodate users in different channel conditions with different reuse factors. However, these priori hand-crafted schemes are still far from optimal in the sense that they do not adapt to dynamic network environments, e.g., time-varying user loads/locations. In addition, opportunistic user scheduling based on their perceived time-varying channels, needs to be jointly considered with ICI management to achieve a high performance gain.

In this paper, we aim at (i) studying coupling dynamics of inter-cell ICI management and intra-cell user scheduling, and (ii) proposing practically implementable joint algorithms that achieve a significant performance gain. To that end, we first propose a pattern-based optimal algorithm that tracks time-varying channel conditions (that runs at short time scales) at both user scheduling and ICI management, where 'pattern' corresponds to a combination of BS ON/OFF activities. Then, we show that the proposed optimal algorithm is hard to implement due to high complexity. The key bottleneck lies in the ICI management part that requires collecting excessive amount of feedback information from all users and also needs complex operations to make decisions on BS coordination at every time slot. To overcome such complexity, we decompose the original optimization problem into two sub-problems (user scheduling and pattern-based ICI management), and solve them with different time scales, whose complexity becomes much lower than that of the optimal algorithm.

The algorithm based on time-scale decomposition stems from a design rationale that ICI management may not have to track fast dynamics, e.g., fast fading channel condition. Instead, it may suffice to run the ICI management following only macroscopic network changes, e.g., user loads/locations, and the average channel conditions of users. In spite of such slow tracking of system dynamics in ICI management, we empirically show that with our decomposed algorithms, the performance increase amounts to about 6∼20% (compared to conventional universal reuse scheme), corresponding to 1/2∼2/3 of the optimal algorithm (that is almost impossible to implement).

The research on mitigating ICI have recently received a lot of attentions [2]–[8]. Optimal binary power control (BPC) for sum rate maximization has been considered in [2]. In [3], [4], optimal joint ICI management (similar BPC) and user scheduling algorithms that operate slot-by-slot and require heavy computation overheads, have been considered in slightly different systems. The authors there presented an idea of using clustering only neighboring BSs [3] or considering only neighboring BSs [4] to reduce complexity. However, it still requires centralized coordination and complex operations per-slot basis, which hinders practical implementation.

To make algorithms practical, there have been recent approaches [6]–[8], based on a slightly different time-scale separation approach from ours. In [6], the authors abstract users that share similar traffic loads and channel environments into classes, and perform ICI management on a very long time scale (e.g., hours) without explicit consideration of intra-cell user scheduling. They basically design ICI management that tracks system dynamics at a very macroscopic level. Our

This research was partly supported in part by the Ministry of Knowledge Economy, Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2009-C1090-0902-0037). This work was also partly supported by the IT R&D program of MKE/IITA [2009-F-045-01, Ultra Small Cell Based Autonomic Wireless Network].

approach differs from [6] that user scheduling is explicitly considered, and also our ICI management runs much faster (e.g., seconds) than that in [6]. The work with a similar time-scale separation to ours has been proposed in [7], [8] for different systems, i.e., OFDMA systems, where they periodically updates the transmit power level for different subbands for ICI management. Due to the difference in system model, we use a different mechanism that updates patterns not powers, leading to a different style of algorithms and analysis. We additionally study the performance gap between the optimal and the decomposed algorithms.

Related work also includes the examination of potential capacity gains (from the perspective of flow-level performance) by BS coordination [9]. Another important issue in multi-cell networks is to resolve load imbalance problem between cells. The authors in [10], [11] *explicitly* balance the load by changing user associations from the BS in hot-spot cells to the adjacent less-crowded BS. Sang et al. [10] proposed an integrated framework consisting of a MAC-layer cell breathing technique and load-aware handover/cell-site selection to deal with load balancing. Bu et al. [11] were the first to rigorously consider a formulation of network-wide proportional fairness (PF) [12] in a multi-cell network where associations between users and BSs are decision variables. Although we assume that user association is fixed, we later argue and empirically show that ICI management is able to *implicitly* resolve the load imbalance, and the performance gain by controlling user association may be small.

The remainder of this paper is organized as follows. In Section II, we present our system model and problem definition. In Section III, we propose a joint pattern selection and user scheduling algorithm to solve this problem. Although this joint algorithm is optimal, it has some implementation difficulties. In order to take into account practical concerns, we design two algorithms using time-scale decomposition that run at different time-scales in Section IV. In Section V, we demonstrate the performance of proposed algorithms, and conclude the paper in Section VI.

II. System Model and Problem Definition

A. Network Model

We consider a wireless cellular network consisting of multiple cells. Denote by $\mathcal{N} \doteq \{1, \ldots, N\}$ and $\mathcal{K} \doteq \{1, \ldots, K\}$ a set of BSs and MSs (or users), respectively. A user $k \in \mathcal{K}$ is associated with a single BS $n \in \mathcal{N}$, which means that data intended for the user k is served only by the BS n. Define $a(\cdot) : \mathcal{K} \to \mathcal{N}$ to be the association function, e.g., $a(k) = n$ if the user k is associated with the BS n. We further denote by $\mathcal{K}_n$ the set of users associated with the BS n. Assume that BSs transmit data with either its given maximum power or 0, which we simply denote by 'ON' or 'OFF' states[1]. We assume that a same frequency band (or channel in short) with bandwidth W in all cells, and consider only downlink transmissions in the time-slotted system indexed by $t = 0, 1, \ldots$. At each slot, a BS can select only one user for its data transmission. Channels may be time-varying, modeled by some stationary, ergodic random process with the finite state index set $\mathcal{I}$ and the stationary distribution $\boldsymbol{\theta} = (\theta^{(i)},\ i \in \mathcal{I})$.

B. Network Resource and Allocation Schemes

The time-varying network resources at slot t are represented by a finite set $\mathcal{R}(t)$ of the K-dimensional feasible rate (bits/slot) vectors over users. A resource allocation scheme then chooses a feasible rate vector in $\mathcal{R}(t)$ at each slot and serves a subset of users with the chosen rate vector. A feasible rate vector in $\mathcal{R}(t)$ is determined by the following two factors: (i) *which BSs are activated* and (ii) *which users are selected in cells for data transmission*.

To formally discuss (i), we define *reuse pattern* (or simply *pattern*) p to be a combination of ON/OFF activities of BSs, which determines inter-cell interference to the corresponding scheduled users in cells. Denote by $\mathcal{P}$ the set of patterns. A pattern p is said to *activate* a BS n, if the activity of the BS n is ON under pattern p. Denote by $\mathcal{N}_p \subset \mathcal{N}$ the set of all BSs activated by the pattern p. In parallel, we denote by $\mathcal{P}_n \subset \mathcal{P}$ the set of patterns that activate the BS n. Define *reuse factor* of a pattern p to be $\chi_p \doteq \frac{|\mathcal{N}_p|}{N} \leq 1$, i.e., the ratio of the number of BSs which use a pattern p to the total number of BSs. Denote by $X_p(t)$ the *pattern selection indicator* for the pattern p, i.e., $X_p(t) = 1$ when the pattern p is used at slot t and 0 otherwise. Then, since only one pattern is used per one slot, we should have:

$$\sum_{p \in \mathcal{P}} X_p(t) = 1. \tag{1}$$

In regard to (ii), define *user scheduling indicator* at slot t by $I_k(t)$, i.e., $I_k(t) = 1$, when the user k is scheduled in its cell, and 0 otherwise. Reflecting the constraint that only one user can be selected in each cell, we should have:

$$\sum_{k \in \mathcal{K}_n} I_k(t) \begin{cases} \leq 1, & \text{if } X_p(t) = 1 \text{ and } n \in \mathcal{N}_p\ , \\ = 0, & \text{otherwise.} \end{cases} \tag{2}$$

Then, a resource allocation scheme incorporates *pattern selection* and *user scheduling* that can be regarded as choosing a sequence of $((I_k(t) : k \in \mathcal{K}), (X_p(t) : p \in \mathcal{P}))_{t=0}^{\infty}$ satisfying the constraints (1) and (2).

We now define the transmission rates of users provided by a resource allocation scheme, depending on the choice of patterns. Let $G_{n,k}(t)$ represent the time-varying channel gain from BS n to user k at slot t. The channel gain may take into account path loss, log-normal shadowing, fast fading and etc. The received SINR for user k at slot t when pattern p is selected and user k is served by its associated BS, can be written as:

$$\Gamma_{kp}(t) = \begin{cases} \dfrac{G_{a(k),k}(t) P_n^{max}}{N_0 W + \sum_{m \in \mathcal{N}_p, m \neq a(k)} G_{m,k}(t) P_m^{max}}, & \text{if } p \in \mathcal{P}_{a(k)}, \\ 0, & \text{otherwise}, \end{cases}$$

where N_0 is the noise spectral density. Here, the noise spectral density is assumed to be equivalent over all users for simple presentation. Following the Shannon's formula, the data rate for user k on reuse pattern p at slot t is given by:

$$r_{kp}(t) = W \log_2 \left(1 + \Gamma_{kp}(t)\right).$$

[1] All discussions in this paper can be readily extended to the case where BSs can transmit data with a finite number of discrete power levels.

TABLE I
SUMMARY OF NOTATIONS

$\mathcal{N}$	set of BSs, $\mathcal{N} = \{1, \dots, N\}$
$\mathcal{K}$	set of users, $\mathcal{K} = \{1, \dots, K\}$
$a(\cdot)$	association function from $\mathcal{K}$ to $\mathcal{N}$
$\mathcal{K}_n$	set of users associated with BS n
$\mathcal{P}$	set of patterns, $\mathcal{P} \doteq \{1, \dots, P\}$
$\mathcal{P}_n$	set of patterns that can be used by BS n
$\mathcal{N}_p$	set of BSs allowed to use the pattern p
χ_p	reuse factor (PF) of pattern p
$G_{n,k}(t)$	channel gain between BS n and user k at slot t
P_n^{max}	transmit power of BS n
N_0	noise spectral density
W	system bandwidth
$\Gamma_{kp}(t)$	received SINR of user k on pattern p at slot t
$r_{kp}(t)$	instantaneous data rate of user k on pattern p at slot t
π_p	fraction of time for pattern p
π_{kp}	fraction of time that user k is served with pattern p
U	system (network-wide) utility
$U^{(n)}$	utility of BS n
$U_k(\cdot)$	utility function of user k's average throughput
$\mathcal{R}$	achievable rate region
$\bar{\mathbf{R}}$	vector of long-term user throughputs, $\bar{\mathbf{R}} = (\bar{R}_1, \cdots, \bar{R}_K)$
$\mathbf{X}(t)$	pattern selection indicator at slot t
$\mathbf{I}(t)$	user scheduling indicator at slot t

Note that $r_{kp}(t) = 0, \forall p \notin \mathcal{P}_{a(k)}$, i.e., user k cannot receive any data rate if its associated BS $a(k)$ is not activated by the pattern p. Also notice that $r_{kp}(t)$ is the *potential data rate* when the user k is scheduled, i.e., its actual data rate becomes 0, when other user, say k', associated with the BS $a(k)$, is scheduled for service. We assume that each BS n knows instantaneous achievable data rates for all its associated users through channel feedbacks. We further assume that BSs have infinite amount of data to be destined to users.

C. Problem Definition

In this paper, we aim at proposing the joint pattern selection and user scheduling that maximizes the long-term network-wide utility whenever possible, i.e., solves the following optimization problem **Q**:

$$\textbf{Q:} \qquad \max \quad U = \sum_{n\in\mathcal{N}} U^{(n)} = \sum_{k\in\mathcal{K}} U_k(\bar{R}_k)$$
$$\text{subject to} \quad \bar{\mathbf{R}} \in \mathcal{R},$$

where $\bar{\mathbf{R}} = (\bar{R}_k, k \in \mathcal{K})$ is the vector of long-term user throughputs; the network-wide utility U is just the summation of utilities of all BSs ($U^{(n)}, n \in \mathcal{N}$), or of utilities of all users ($U_k, k \in \mathcal{K}$). Assume the standard condition of differentiability and strictly increasing concavity of U_k. The set $\mathcal{R} \in \mathbb{R}_+^K$, the set of all achievable rate vectors over long-term, is shown to be a closed bounded convex set. Denote by $\pi_p^{(i)}$ the portion of pattern p for the i-th channel state. By further denoting by $\pi_{kp}^{(i)}$ and $r_{kp}^{(i)}$ the fraction of time that is served by user k and the data rate of user k (if scheduled) on the pattern p and for i-th channel state, respectively, we can characterize $\mathcal{R}$ by:

$$\mathcal{R} = \Big\{ \bar{\mathbf{R}} = (\bar{R}_k : k \in \mathcal{K}) \mid \bar{R}_k = \sum_{i\in\mathcal{I}} \sum_{p\in\mathcal{P}} \theta^{(i)} \pi_{kp}^{(i)} r_{kp}^{(i)} , \sum_{k\in\mathcal{K}_n} \pi_{kp}^{(i)} \le \pi_p^{(i)}, \forall i, \forall p, \forall n, \sum_{p\in\mathcal{P}} \pi_p^{(i)} \le 1, \forall i \Big\}.$$

III. Optimal Algorithm

In this section, we first study the structure of optimal solutions analytically for simple scenarios to gain insights. and then describe a optimal pattern selection and user scheduling algorithm that converges to the optimal solution of **Q**.

A. Structure of optimal solution for symmetric networks with static channels

For general networks, it is quite difficult to characterize the optimal fractions of time for user-patterns $(\pi_{kp}^{(i)} : k \in \mathcal{K}, p \in \mathcal{P}, i \in \mathcal{I})$. However, we will show that it is indeed possible to explicitly characterize them for symmetric networks with static channels. Here, a network is said to be symmetric if all BSs have the same number of users whose channel characteristics are equivalent each other. Fig. 1 depicts an illustrative example of a linear two-cell network having three patterns where $(\chi_1, \chi_2, \chi_3) = (1, 0.5, 0.5)$. Recall that χ_p, the reuse factor of pattern p, is equal to the ratio of the number of BSs which use a pattern p to the total number of BSs. Since the network is symmetric, it is enough to analyze the following optimization problem **Q-symmetric** for a reference BS only:

Q-symmetric:

$$\max_{(\pi_{kp}: k\in\mathcal{K}, p\in\mathcal{P})} \quad U^{(1)} = \sum_{k\in\mathcal{K}_1} U_k(\bar{R}_k) \tag{3}$$
$$\text{subject to} \quad \sum_{p\in\mathcal{P}_1} \sum_{k\in\mathcal{K}_1} \frac{\pi_{kp}}{\chi_p} \le 1, \tag{4}$$
$$\pi_{kp} \ge 0, \quad \forall k \in \mathcal{K}_1, \forall p \in \mathcal{P}_1, \tag{5}$$
$$\bar{R}_k = \sum_{p\in\mathcal{P}_1} \pi_{kp} r_{kp}, \quad \forall k \in \mathcal{K}_1. \tag{6}$$

Here, we derive the constraint (4) for the above two-cell example. However, this can be readily extended to general symmetric networks.

$$\begin{aligned} 1 = &\ \textstyle\sum_{p\in\mathcal{P}} \pi_p = \pi_1 + \pi_2 + \pi_3 \\ = &\ \pi_1 + 2\pi_2 = \pi_1/\chi_1 + \pi_2/\chi_2 \ \ (\because \pi_2 = \pi_3 \text{ by symmetry}) \\ \ge &\ \textstyle\sum_{k\in\mathcal{K}_1} \frac{\pi_{kp}}{\chi_1} + \sum_{k\in\mathcal{K}_1} \frac{\pi_{kp}}{\chi_2} = \sum_{p\in\mathcal{P}_1} \sum_{k\in\mathcal{K}_1} \frac{\pi_{kp}}{\chi_p}. \end{aligned}$$

We have found that the problem **Q-symmetric** has an interesting structure of optimal solution described by the Lemmas 3.1 and 3.2. Let $\chi_p r_{kp}$ be the *effective rate* on pattern p for user k, which is the normalized data rate w.r.t. χ_p. Note that there is a trade-off between the reuse factor χ_p and the data rate r_{kp}. If the user k chooses the pattern p with the lower value χ_p, then the less BSs are active in the network, and accordingly the higher data rate r_{kp} is expected, and vice versa. And we adopt the generalized (w, α)-fair utility function in [13] where $U_k(\bar{R}_k)$ is given by:

$$U_k(\bar{R}_k) = \begin{cases} w_k \log \bar{R}_k, & \text{if } \alpha = 1, \\ w_k(1-\alpha)^{-1} \bar{R}_k^{1-\alpha}, & \text{otherwise,} \end{cases} \tag{7}$$

where α is nonnegative and w_k is positive.

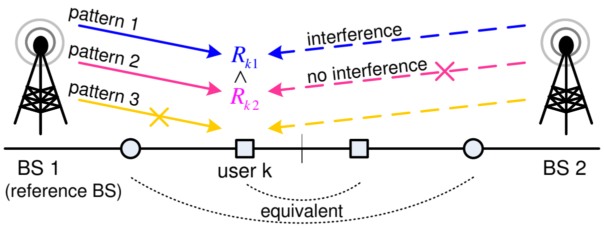


Fig. 1. Example of a linear two-cell symmetric network

Lemma 3.1: For symmetric networks with static channels, the objective (3) is maximized if and only if

$$\pi_{kp} \begin{cases} \geq 0, & \text{if } p = p^*(k), \\ = 0, & \text{otherwise}, \end{cases} \quad \text{where} \;\; p^*(k) = \arg\max_p \chi_p r_{kp}.^2$$

This implies that each user, if served, only utilizes its pattern having the largest effective rate.

Lemma 3.2: For the generalized (w, α)-fair utility function, the optimal fractions of time for user-patterns is given by:

$$\begin{aligned} &\pi_{kp^*(k)} = \left(w_k \chi_{p^*(k)} r_{kp^*(k)}^{1-\alpha} / \lambda_0\right)^{1/\alpha}, \\ &\text{where} \;\; \lambda_0 = \left(\textstyle\sum_{p \in \mathcal{P}_1} \sum_{k \in \mathcal{K}_{1p}} w_k^{1/\alpha} \chi_{p^*(k)}^{\frac{1-\alpha}{\alpha}} r_{kp^*(k)}^{\frac{1-\alpha}{\alpha}}\right)^{\alpha}, \end{aligned} \tag{8}$$

where $\mathcal{K}_{1p}$ is the set of users whose most effective pattern is p, i.e., $p^*(k) = p$ if $k \in \mathcal{K}_{1p}$.

Please refer to our technical report [14] for proofs. Now, we give a numerical example to illustrates the property of the optimal solution.

Example 1: Consider the example of the linear two-cell symmetric network in the Fig. 1. In this example, we have three patterns $p \in \mathcal{P} = \{1, 2, 3\}$ where $\mathcal{N}_1 = \{1, 2\}, \mathcal{N}_2 = \{1\}$, $\mathcal{N}_3 = \{2\}$ and $(\chi_1, \chi_2, \chi_3) = (1, 0.5, 0.5)$. Suppose that all users have the same utility function with $(w, \alpha) = (1, 1)$. By (8), we can obtain $\lambda_0 = |\mathcal{K}_1|$ and the optimal time fractions of user-patterns is given by

$$\pi_{kp^*(k)} = \begin{cases} |\mathcal{K}_1|^{-1}, & \text{if } k \in \mathcal{K}_{11}, \\ (2|\mathcal{K}_1|)^{-1}, & \text{if } k \in \mathcal{K}_{12}, \end{cases} \tag{9}$$

where $|\mathcal{K}_{11}|$ is the set of users such that $R_{k1} \geq 2R_{k2}$, i.e., the set of center users, and $|\mathcal{K}_{12}|$ is the set of users such that $R_{k1} < 2R_{k2}$, i.e., the set of edge users. Thus, the optimal portion for each pattern (π_1, π_2, π_3) is given by

$$\begin{aligned} \pi_1 &= \sum_{k \in \mathcal{K}_{11}} \pi_{kp^*(k)} = |\mathcal{K}_{11}| / |\mathcal{K}_1| \quad \text{and} \\ \pi_2 = \pi_3 &= \sum_{k \in \mathcal{K}_{12}} \pi_{kp^*(k)} = |\mathcal{K}_{12}| / (2|\mathcal{K}_1|) \,. \end{aligned} \tag{10}$$

Note that in the case of proportional fair ($\alpha = 1$) the optimal portion of each pattern depends only on and is proportional to the number of users in the sets of center and edge users. However, for general cases ($\alpha \neq 1$), its closed form is very complex because the optimal portion of each pattern depends on the data rate $r_{kp^*(k)}$ for all users due to (8). Thus, we rely on numerical computations for $\alpha \geq 0, \alpha \neq 1$.

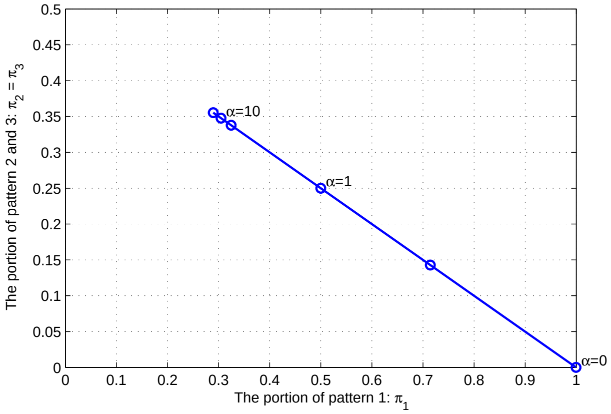


Fig. 2. Numerical example of the linear two-cell symmetric network where each BS has two users; user 1 is in the inner region of the cell and user 2 is in the edge of the cell, whose instantaneous data rate vectors are given by $(r_{11}, r_{12}) = (10, 11)$ and $(r_{21}, r_{22}) = (3, 8)$.

Fig. 2 depicts the optimal portion of patterns with respect to the fairness criterion α. We fix the number of users as shown in the Fig. 1, that is, each BS has are two users: one is in the center and the other in the edge of the cell, $|\mathcal{K}_1| = 2, |\mathcal{K}_{11}| = 1, |\mathcal{K}_{12}| = 1$. When $\alpha = 1$, the optimal portion of patterns can be given by (10): $(\pi_1, \pi_2, \pi_3) = (1/2, 1/4, 1/4)$. Accordingly, user throughputs can be easily calculated: $(R_1, R_2) = (\pi_{11} r_{11}, \pi_{22} r_{22}) = (\pi_1 r_{11}, \pi_2 r_{22}) = (5, 2)$. When we decrease α, the portion of pattern 1 increases as expected. In the extreme case, throughput maximization ($\alpha = 0$), only user 1 having a better channel is always served with pattern 1, and user 2 cannot be served at all, i.e., $(\pi_1, \pi_2) = (1, 0)$. On the other extreme case ($\alpha \to \infty$), max-min fairness is achieved such that the throughputs of user 1 and user 2 become identical.

B. Joint pattern selection and user scheduling algorithm

We now present an optimal joint pattern selection and user scheduling algorithm. To that end, we use a standard gradient-based algorithm, e.g., Stolyar's gradient algorithm [15], that selects the achievable rate vector maximizing the sum of weighted rates where the weights are marginal utilities at each slot. Then, it suffices to solve the following problem at each slot, which jointly determines the pattern selection $\mathbf{X}(t) = (X_p(t) : p \in \mathcal{P})$ and user scheduling $\mathbf{I}(t) = (I_k(t) : k \in \mathcal{K})$:

Q-joint:

$$\max_{\mathbf{X}(t), \mathbf{I}(t)} \quad \Delta U(t) = \sum_{k \in \mathcal{K}} U_k'(\bar{R}_k(t-1)) r_k(t) \tag{11}$$

$$\text{subject to} \;\; \sum_{p \in \mathcal{P}} X_p(t) = 1, \tag{12}$$

$$\sum_{k \in \mathcal{K}_n} I_k(t) \begin{cases} \leq 1, & \text{if } X_p(t) = 1 \text{ and } n \in \mathcal{N}_p, \\ = 0, & \text{otherwise}, \end{cases} \tag{13}$$

where $r_k(t) = \sum_{p \in P} X_p(t) I_k(t) r_{kp}(t)$ is the data rate assigned to user k at slot t and $\bar{R}_k(t) = \frac{1}{t} \sum_{\tau=1}^{t} r_k(\tau)$ $= \bar{R}_k(t-1) + \epsilon_t \left[r_k(t) - \bar{R}_k(t-1)\right]$ (by letting $\epsilon_t = 1/t$) is the long-term throughput for user k up to slot t.

[2] For simplicity, we ignore the case when more than two patterns achieve the same largest value.

Remark 3.3: If we fix the user scheduling $\mathbf{I}(t)$ and choose utility function as $U_k(\bar{R}_k) = \bar{R}_k$ in **Q-joint**, then this problem is reduced to binary power control (BPC) problem for sum-rate maximization in [2].

The optimization problem **Q-joint** is an integer programming. A naive approach is the exhaustive search of all possible combinations of pattern selections and user schedulings. With the help of Lemma 3.4 telling us the nice property of the problem, we can develop the joint optimal pattern selection and user scheduling algorithm that requires lower complexity than the exhaustive search.

Lemma 3.4: For a *fixed* pattern p, then the problem **Q-joint** can be decomposed into the following $|\mathcal{N}_p|$ independent intra-cell user scheduling problems:

$$k_n^*(t) = \arg\max_{k \in \mathcal{K}_n} U_k'(\bar{R}_k(t-1)) r_{kp}(t), \quad \forall n \in \mathcal{N}_p. \qquad (14)$$

Remark 3.5: A similar argument has been made in a difference setting [4], but we present this Lemma for completeness. Please refer to our technical report [14] for a proof.

Joint pattern selection and user scheduling algorithm

$$p^*(t) = \arg\max_{p \in \mathcal{P}} \sum_{n \in \mathcal{N}_p} \Big[\max_{k \in \mathcal{K}_n} U_k'(\bar{R}_k(t-1)) r_{kp}(t) \Big],$$
$$k_n^*(t) = \arg\max_{k \in \mathcal{K}_n} U_k'(\bar{R}_k(t-1)) r_{kp^*}(t), \ \ \forall n \in \mathcal{N}_{p^*}.$$

Note that the total number of combinations for our joint algorithm is polynomial $O(P \cdot K)$ while the that of the naive exhaustive search is $O(P \cdot K^N)$. For each pattern p, we select the best user having the largest value of $U_k'(\bar{R}_k(t-1)) r_{kp}(t)$ from (14) and then the value of the selected user is used in the pattern selection algorithm. We then find the best pattern $p^*(t)$ that maximizes the sum of weighted rate $U_k'(\bar{R}_k(t-1)) r_{kp^*}(t)$ of the scheduled users. The proof of convergence to the optimal solution is a slight extension to [15], [16] that studied only user scheduling for a fixed pattern. We skip the proof.

This joint algorithm requires instantaneous channel feedback from all users in the network. We assume that at each slot t, user k estimates its own SINR for all patterns $p \in \mathcal{P}_{a(k)}$ upon listening to pilot signals, calculates the instantaneous data rate $r_{kp}(t)$ and then reports this information to the cental coordinator through its associated BS.

However, this joint pattern selection and user scheduling algorithm still has implementation difficulties. Apart from the computational complexity of this algorithm, the central coordinator running the algorithm needs to collect the following information from each BS $n \in \mathcal{N}$: instantaneous data rate $r_{kp}(t)$ of all its associated users $k \in \mathcal{K}_n$ on its available patterns $p \in \mathcal{P}_n$. The total amount of feedbacks is quite large, i.e, $(\sum_{n \in \mathcal{N}} |\mathcal{K}_n||\mathcal{P}_n|)$, though they may be delivered along with high speed wired links. Furthermore, a series of tasks, including information feedback from BSs to the central coordinator as well as the computation and the distribution of central coordinator's decision, should be performed in one slot.

IV. Time-Scale Decomposed Algorithm

A. Algorithm Description

In contrast to the centralized joint pattern selection and user scheduling algorithm in Section III, user scheduling in practice is typically undertaken by individual BSs independently without any coordination and information exchange with other BSs. In this section, in order to take into account such autonomous feature in user scheduling as well as overcome high computation and feedback overheads in the optimal algorithm, we run user scheduling at every slot, but pattern portion change less frequently, say, every $T_p >> 1$ slots. We first describe our algorithm (see Fig. 3 for a pictorial description), and then explain the rationale behind it.

Pattern portion change algorithm

For every T_p slots, each BS $n \in \mathcal{N}$ computes the partial derivative $D_p^{(n)}$ and sends it to the central coordinator,

$$D_p^{(n)} = \sum_{k \in \mathcal{K}_n} U_k'(\bar{R}_k) \cdot \Big(\frac{\bar{\pi}_{kp}}{\pi_p} \bar{r}_{kp} \Big) \ , \quad p \in \mathcal{P}_n.$$

Then, the central coordinator calculates the gradient vector $\mathbf{D} = (D_1, D_2, \cdots, D_P)$ by collecting $D_p^{(n)}$ from all BSs,

$$D_p = \sum_{n \in \mathcal{N}} D_p^{(n)} \ , \quad p \in \mathcal{P},$$

and updates the pattern portion vector $\boldsymbol{\pi}$ as follows,

$$\boldsymbol{\pi} \leftarrow Proj_{\sum_{p \in \mathcal{P}} \pi_p = 1}\, (\boldsymbol{\pi} + \gamma \mathbf{D}) \, ,$$

where $Proj_A(\cdot)$ denotes an orthogonal projection on a set A.

User scheduling algorithm

At each slot t, each BS $n \in \mathcal{N}_{p(t)}$ activated by pattern $p(t)$ selects the user $k_n^*(t)$, i.e., $I_{k_n^* p}(t) = 1$,

$$k_n^*(t) = \arg\max_{k \in \mathcal{K}_n} U_k'(\bar{R}_k(t-1)) r_{kp}(t),$$

and updates the following variables for all users $k \in \mathcal{K}_n$ with some constants $0 < \beta_1, \beta_2, \beta_3 < 1$:

$$\begin{aligned} \bar{R}_k(t) &= (1-\beta_1)\bar{R}_k(t-1) + \beta_1 I_k(t) r_{kp}(t) \ , \\ \bar{\pi}_{kp}(t) &= (1-\beta_2)\bar{\pi}_{kp}(t-1) + \beta_2 I_k(t) \ , \\ \bar{r}_{kp}(t) &= \begin{cases} (1-\beta_3)\bar{r}_{kp}(t-1) + \beta_3 r_{kp}(t), & \text{if } I_k(t) = 1 \ , \\ \bar{r}_{kp}(t-1), & \text{otherwise,} \end{cases} \end{aligned}$$

Two algorithms with different time scales interact with each other as follows: The pattern portion change algorithm adjusts the portion of reuse patterns $\boldsymbol{\pi}$ for every T_p slots, using the variables $\bar{R}_k(t), \bar{\pi}_{kp}(t), \bar{r}_{kp}(t)$. These variables essentially correspond to the long-term averages of $I_k(t) r_{kp}(t)$, $\pi_{kp}(t)$, and $r_{kp}(t)$ which are progressively updated at every slot by the user scheduling algorithm. This time-scale decomposition and the way of interaction between two algorithms implies that we design and operate the pattern portion algorithm to let it tract just *average* interference levels and channel conditions, not fast time-varying ones like the joint optimal algorithm in Section III. Remarking that user scheduling algorithm can be carried out autonomously, we can significantly reduce the actual (amortized) complexity per slot, which makes our

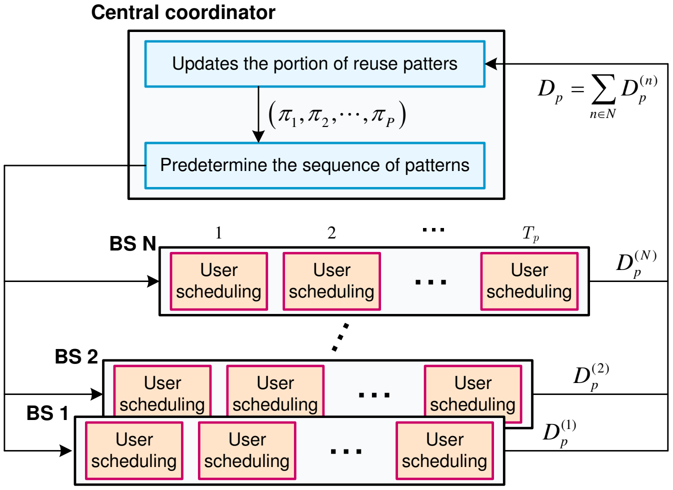


Fig. 3. Proposed time-scale decomposed algorithms

algorithms much more implementable. We will discuss the price of such complexity reduction, i.e., utility gperformance gap with the optimal algorithm in the Subsection IV-C.

B. Rationale of Time-scale Decomposed Algorithms

The pattern portion change algorithm is a standard gradient projection algorithm for the following problem **Q-pattern**:

Q-pattern:

$$\max_{\boldsymbol{\pi}} \sum_{k \in \mathcal{K}} U_k(\bar{R}_k) = \sum_{k \in \mathcal{K}} U_k\Big(\sum_{p \in \mathcal{P}} \phi_{kp} \pi_p \bar{r}_{kp}\Big)$$
$$\text{subject to} \quad \sum_{p \in \mathcal{P}} \pi_p = 1,$$

where $\phi_{kp} \in [0,1]$ is the probability that the user k is scheduled when pattern p is selected, i.e., $\phi_{kp} \cdot \pi_p = \bar{\pi}_{kp}$

For each of the pattern portion update epoch, i.e., every T_p slots, each BS n needs to calculate the partial derivative $D_p^{(n)} \doteq \partial U^{(n)} / \partial \pi_p$ of per-cell utility $U^{(n)}$ with respect to the portion of pattern p and send these information to the central coordinator.

$$D_p^{(n)} \doteq \frac{\partial U^{(n)}}{\partial \pi_p} = \sum_{k \in \mathcal{K}_n} U_k'(\bar{R}_k) \cdot \frac{\partial \bar{R}_k}{\partial \pi_p}, \tag{15}$$

where[3]

$$\frac{\partial \bar{R}_k}{\partial \pi_p} = \phi_{kp} \bar{r}_{kp} = \frac{\bar{\pi}_{kp}}{\pi_p} \bar{r}_{kp}. \tag{16}$$

Note that three parameters ($\bar{R}_k$, $\bar{\pi}_{kp}$ and $\bar{r}_{kp}$) required to run this pattern portion update algorithm can be attained by attained by the user scheduling algorithm over long time. And then the central coordinator gathers information from all BSs and calculates the partial derivative of the network utility $D_p \doteq \partial U / \partial \pi_p$ by aggregating these partial derivatives of the local utility,

$$D_p \doteq \frac{\partial U}{\partial \pi_p} = \sum_{n \in \mathcal{N}} D_p^{(n)}, \quad p \in \mathcal{P}, \tag{17}$$

and updates the portion of reuse patterns following the increasing direction of network utility.

$$\boldsymbol{\pi} \leftarrow Proj_{\sum_{p \in \mathcal{P}} \pi_p = 1}, (\boldsymbol{\pi} + \gamma \mathbf{D}). \tag{18}$$

Based on the updated portion of patterns, the central coordinator predetermines the sequence of patterns for next T_p slots that satisfies:

$$\text{(the total number of pattern } p\text{) } / \; T_p \approx \pi_p, \quad \forall p \in \mathcal{P}.$$

While there may be many strategies, a nice candidate is a random strategy. The central coordinator sequentially determines the sequence of patterns by rolling a P-dimensional die T_p times with probability of the pattern p being π_p.

Now we develop the user scheduling algorithm under the fixed pattern given by the pattern portion change algorithm. From the Lemma 3.4, for the given pattern the network-wide user scheduling problem can be decomposed into independent intra-cell user scheduling problems. Therefore, each BS needs to solve the following problem **Q-scheduling**:

Q-scheduling:

$$\max_{(I_k(t), k \in \mathcal{K}_n)} \sum_{k \in \mathcal{K}_n} U_k'(\bar{R}_k(t-1)) I_k(t) r_{kp}(t)$$
$$\text{subject to} \quad \sum_{k \in \mathcal{K}_n} I_k(t) \le 1.$$

The user scheduling algorithm solving **Q-scheduling** is straightforward. Each BS $n \in \mathcal{N}_p$ allowed to use the pattern p independently chooses the best user $k_n^*(t)$ among it associated user set $\mathcal{K}_n$, i.e., $I_{k_n^*}(t) = 1$:

$$k_n^*(t) = \arg\max_{k \in \mathcal{K}_n} U_k'(\bar{R}_k(t-1)) r_{kp}(t), \quad \forall n \in \mathcal{N}_p, \tag{19}$$

and updates the following variables for the future purpose of the pattern portion change algorithm:

$$\begin{aligned}
\bar{R}_k(t) &= (1-\beta_1)\bar{R}_k(t-1) + \beta_1 I_k(t) r_{kp}(t), \\
\bar{\pi}_{kp}(t) &= (1-\beta_2)\bar{\pi}_{kp}(t-1) + \beta_2 I_k(t), \\
\bar{r}_{kp}(t) &= \begin{cases} (1-\beta_3)\bar{r}_{kp}(t-1) + \beta_3 r_{kp}(t), & \text{if } I_k(t) = 1, \\ \bar{r}_{kp}(t-1), & \text{otherwise,} \end{cases}
\end{aligned}$$

where $\beta_1, \beta_2, \beta_3 > 0$ are small averaging parameters; $\bar{R}_k(t)$, $\bar{\pi}_{kp}(t)$ and $\bar{r}_{kp}(t)$ are the average throughput of user k, the average fraction of time that user k is served with pattern p, and the average instantaneous data rate when the user k is served with pattern p, respectively.

Remark 4.1: There are two key differences between the algorithm in [8] and ours. First, they additionally introduce a *virtual scheduler* to obtain the fraction of time that the scheduler chooses user i for transmission in sub-band j (their notation: ϕ_{ij}). In our algorithm, however, we just obtain the fraction of time that user k is served with pattern p (our notation: $\bar{\pi}_{kp}$) using the *actual scheduler* without any extra algorithm. Second, they do not reflect time-varying nature of the data rate available to user i in sub-band j (their notation: R_{ij}) by assuming this rate does not change with time. In our algorithm, the long-term average of data rate of user k on pattern p (our notation: $\bar{r}_{kp}$) is not just the average of instantaneous data rata. We take the average of instantaneous data rata only if the user k is really served by the scheduler.

[3] While we differentiate $\bar{R}_{kp}$ on π_p, we assume that ϕ_{kp} is constant. Please refer to [8] and its technical report for rigorous proof.

TABLE II

COMPARISON BETWEEN JOINT OPTIMAL ALGORITHM (JOA) AND TIME-SCALE DECOMPOSED ALGORITHM (TDA)

	Joint optimal algorithm	Time-scale decomposed algorithm
Time-scale of algorithms	every slot	every slot (user scheduling) every T_p slot (pattern portion change)
Amount of feedbacks to each BS n at each slot	$\lvert\mathcal{K}_n\rvert\lvert\mathcal{P}_n\rvert$	$\lvert\mathcal{K}_n\rvert$
Amount of feedbacks to the central coordinator	$\sum_{n\in\mathcal{N}}\lvert\mathcal{K}_n\rvert\lvert\mathcal{P}_n\rvert$	$\sum_{n\in\mathcal{N}}\lvert\mathcal{P}_n\rvert$
Period of feedback to the central coordinator	1	T_p
Convergence speed	fast	reasonable speed (depending on T_p)

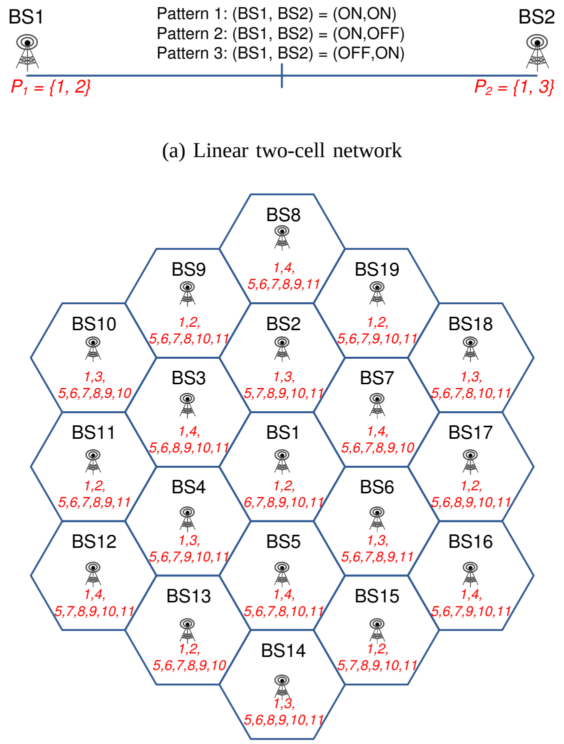
Recall that the opportunistic scheduler likely to serve the user whose current channel quality is high relative to his own rate statistics. In other words, our $\bar{r}_{kp}$ reflects the multi-user diversity gain from exploiting the channel fluctuation.

C. Complexity Reduction and Its Price

Our time-scale decomposed algorithm still involves signalings from BSs to the central coordinator. However, we can significantly reduce feedback overheads because the periodicity of the feedback is stretched from every slot to every T_p slots. Moreover, the amount of feedbacks is reduced from $\left(\sum_{n\in\mathcal{N}}|\mathcal{K}_n||\mathcal{P}_n|\right)$ to $\left(\sum_{n\in\mathcal{N}}|\mathcal{P}_n|\right)$, i.e., requires only the BS-level feedback, not the user-level channel feedback. The amount of feedbacks to each BS n from its associated users at each slot is also reduced from $|\mathcal{K}_n||\mathcal{P}_n|$ and $|\mathcal{K}_n|$ because users need to send channel information only for the predetermined pattern. Table II compares the joint pattern selection and user scheduling algorithm with the proposed algorithms based on time-scale decomposition.

This complexity reduction for implementability comes at the cost of performance gap with the joint optimal algorithm. This is because the ICI management part in the decomposed algorithm cannot fully exploit instantaneous *inter-cell* channel variations, and only *intra-cell* channel variations are opportunistically utilized. Note that in the joint optimal algorithm, both pattern selection and user scheduling fully exploit both inter-cell and intra-cell time-varying channel conditions at a fast time scale.

As an example, consider a two cell network where two users are located at the edge of each cell. Their achievable rates are limited by severe ICI. The decomposed algorithm will find the following TDMA-like solution: BS 1 and BS 2 are exclusively active in order to mitigate the ICI, i.e., the portion for the pattern in which both BSs are active is nearly zero. However, suppose that both (time-varying) inter-cell channel gains from BS 1 (or 2) to the user in BS 2 (or 1) are in deep fading at some time slot. You can imagine this case as if there were a big wall between two cells. Then the user in cell 1 (or 2) is not interfered by the BS transmission in cell 2 (or 1). Therefore, serving two users simultaneously is transiently optimal in this inter-cell deep fading case, whereas the pattern that only one BS is active is the solution of the average ICI mitigation. Joint optimal algorithm can find this optimal solution by tracking this fast fading while the decomposed algorithm cannot. We finally comment that as we will see in the Section V, in absence of fast fading, the performance gap becomes negligible.



(a) Linear two-cell network

(b) Two-tier multi-cell network composed of 19 cells

Fig. 4. Network configurations.

V. SIMULATION RESULTS

A. Simulation Setup

We consider two cases of network configuration: one is the linear two-cell network and the other is the two-tier multi-cell network composed of 19 cells. In both cases, the distance between BSs is 2km. In the linear two-cell network, there are three patterns $\mathcal{P}=\{1,2,3\}$. Under pattern 1, both BSs are ON, and under pattern 2 (resp. 3), only BS 1 (resp. 2) is ON. In the two-tier multi-cell network, we consider 11 patterns. Under pattern 1, all BSs are ON so that each BS receives all the ICI from all over the network. However, under well-designed patterns 2∼4 or 5∼11 (see Fig. 4 for the pattern design), the BS using these patterns can expect the first-tier ICI mitigation or the mitigation of ICI from one of its neighbors, respectively.

To evaluate the performance under various user distribution scenarios, we introduce a variable, so-called, ‘user distribution offset’ $\rho\in[0,1]$, which adjusts the minimum distance between the BS and the user to $\rho\times$(cell radius). Basically, we randomly distributed users in each cell with this minimum distance restriction. For example, if $\rho = 0$, users are uniformly generated over the cell. On the other hand, if ρ goes to 1, users are only located in the edge of the cell.

The maximum powers of BSs are all the same with 20W. Channel models are implemented following ITU PED-B path loss model [17] and Jakes' Rayleigh fading model. The channel bandwidth is 10MHz, and the time-slot length is 5ms, as specified in the IEEE 802.16e standard. The pattern update period $T_p = 500$, and the step size is chosen to be a typically small value, i.e., $\beta_1 = \beta_2 = \beta_3 = \gamma = 0.001$. For each given parameter set, we ran simulations over 50000 slots.

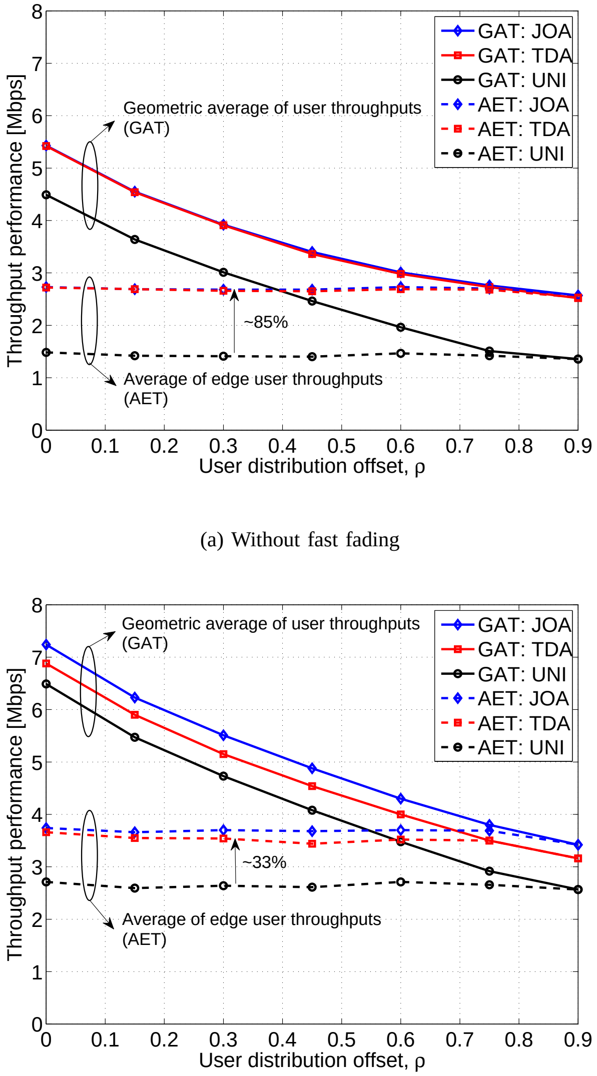
We consider the performance of the (i) *conventional universal reuse scheme* (UNI), in which all BSs in the network are always active without any ICI management, as a baseline and compare the performance of the following two algorithms normalized by UNIVERSAL: (i) *the joint optimal algorithm* (JOA) and (ii) *the algorithm based on time-scale decomposition* (TDA). As performance metrics, the geometric average of user throughputs (GAT) and the average of edge user throughputs (AET) are used. We use GAT since maximizing this is equivalent to the system objective (sum of log throughputs). The AET is the measure of cell edge performance defined as the average throughput of users located at cell edges. In our simulation, we treat 'edge users' as those who are more than 800m away from their associated BSs in our setup.

B. Linear two-cell network case

Fig. 5 shows the GAT and AET performances of three algorithms in the linear two-cell network. In the case without fading, the performances of both JOA and TDA are almost same, where they increase the GAT and the AET by 20∼85% (depending on user distribution) and 85% compared to UNI. We observe a higher performance gain when user distribution offset is larger (i.e., more users are located at cell edges). This is because the ICI management is mainly targeted for the performance improvement of edge users. With fading, however, as discussed in subsection IV-C, there is a performance gap between JOA and TDA due to loss in opportunism of time-scale decomposition. Still, the TDA outperforms the UNI in terms of both the GAT (6∼20% depending on user distribution) and AET (33%). Note that TDA can attain more than 1/2 (at $\rho = 0$) and up to 2/3 (at $\rho = 0.9$) of the GAT performance gain that is achieved by JOA.

C. Two-tier 19-cell network case

In Fig. 6 shows the GAT and AET performances in the two-tier multi-cell network composed of 19 cells. Although the performance gain is a little small compared to the simple linear two-cell network case, trends are similar to those in Fig. 5 as a whole. With fading, the TDA outperforms the UNI in terms of both the GAT (5∼25% depending on user distribution) and AET (25%). Similar to the two-cell case, TDA can still attain 1/2∼2/3 of GAT performance gain that is achieved by JOA.



(a) Without fast fading

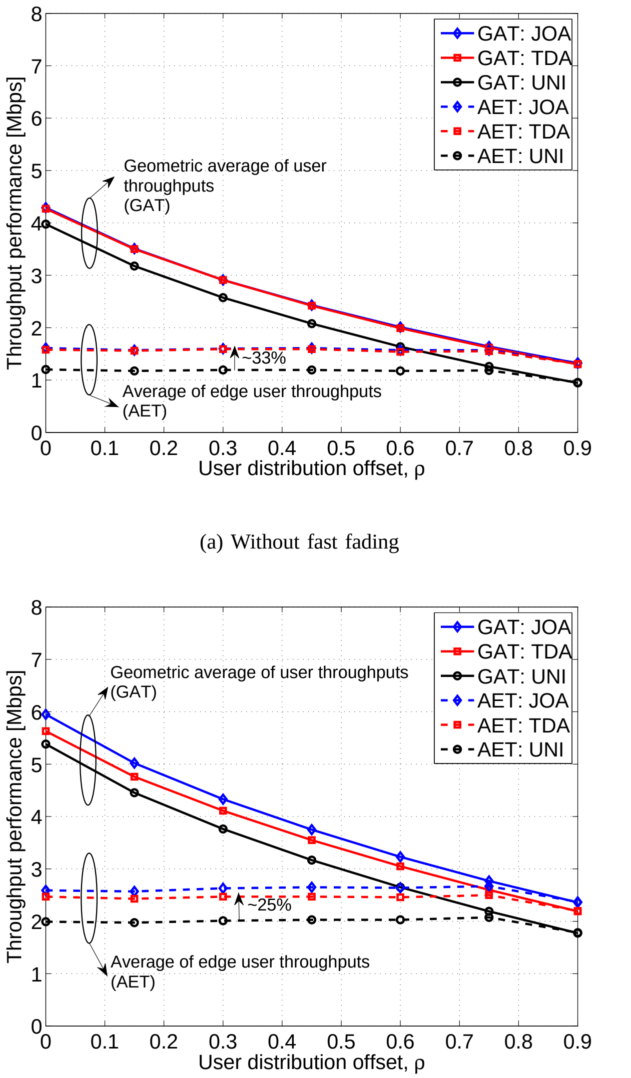
(b) With fast fading

Fig. 5. Throughput performances of three algorithms in two-cell network: joint optimal algorithm (JOA), time-scale decomposed algorithm (TDA) and universal reuse (UNI).

D. Imbalance load: BS coordination vs. association change

We also test the performance in the linear two-cell network having imbalanced loads. We located 2 users 900m away from BS 1 and $2 \times LI$ users 900m away from BS 2, respectively, where LI quantifies the load imbalance. In the network with imbalanced load, we have performed simulation to investigate the amount of additional gain that BS association change can provide. We compare the following two different approaches: (i) *Association change: load-aware handover in [10]* and (ii) *BS coordination: our TDA*. As a baseline, we also plot the performance of UNI.

Originally users are associated with the closest BS offering the best signal strength. In the case of association change algorithm, however, if the expected throughput measure in [10] from the other BS is greater than that from the current BS,



(a) Without fast fading

(b) With fast fading

Fig. 6. Throughput performances of three algorithms in 19-cell network: joint optimal algorithm (JOA), time-scale decomposed algorithm (TDA) and universal reuse (UNI).

then the user changes its association. When the LI is small, users do not change their associations. When we increase LI more than 6, the association change from the hot-spot cell (BS 2) and the under-loaded cell (BS1) happens (moving one, two and three users at LI=6, 8 and 10, respectively) by the load-aware handover in [10]. As can be seen in Fig. 7, however, the gain from the association change is marginal.

On the other hand, using the BS coordination, we can implicitly resolve the load imbalance by preventing the hot-spot cell (BS 2) from being turned off, i.e., provide more transmission chances compare to the BS 1. In brief, the BS coordination originally developed for ICI mitigation can also resolve the load imbalance, and the improvement of than BS coordination is better than that of the association change.

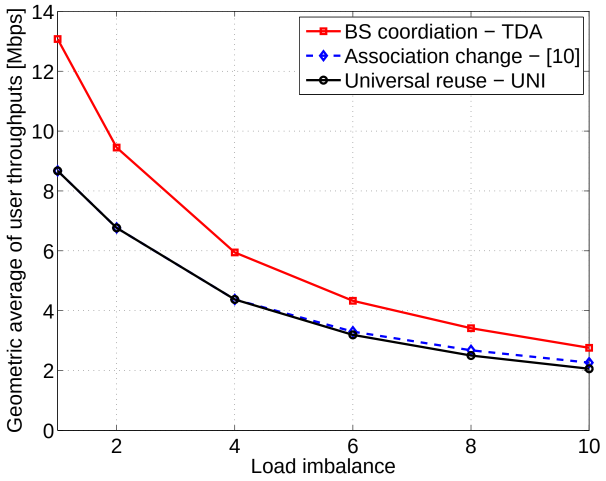


Fig. 7. Association change vs. BS coordination under imbalanced load scenario.

VI. Conclusion

In this paper, we have focused on the problem of joint ICI management and user scheduling in multi-cell networks. We have shown that the joint optimal algorithm is too complex (in terms of computational and signaling overhead) to be implemented in practical systems. To overcome this complexity and make the algorithm practical, we have decomposed the original optimization problem into two sub-problems, where we run ICI management at a slower time scale than user scheduling. This time-scale decomposition stems from a design rationale that ICI management may not have to track fast changing dynamics, and it may suffice to attain much gain just by running it based only on macroscopic network changes. We empirically show that even with such a slow tracking of system dynamics at the ICI management, our algorithm achieves high performance gain compared to a conventional universal reuse scheme, as well as is practically implementable compared to the very complex optimal algorithm.

References

[1] WiMAX Forum, “Mobile WiMAX - Part I: A technical overview and performance evaluation,” Aug. 2006.

[2] A. Gjendemsj, D. Gesbert, G. E. Oien, and S. G. Kiani, “Binary power control for sum rate maximization over multiple interfering links,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 8, pp. 3164–3173, 2008.

[3] S. Das, H. Viswanathan, and G. Rittenhouse, “Dynamic load balancing through coordinated scheduling in packet data systems,” in *Proc. IEEE INFOCOM*, San Francisco, CA, Mar. 2003.

[4] J. Cho, J. Mo, and S. Chong, “Joint network-wide opportunistic scheduling and power control in multi-cell networks,” in *Proc. IEEE WoWMoM*, San Francisco, CA, Jun. 2007.

[5] K. Son, S. Chong, and G. de Veciana, “Dynamic association for load balancing and interference avoidance in multi-cell networks,” in *Proc. WiOpt*, Limassol, Cyprus, Apr. 2007.

[6] B. Rengarajan and G. de Veciana, “Network architecture and abstractions for environment and traffic aware system-level coordination of wireless networks: The downlink case,” in *Proc. IEEE INFOCOM*, Phoeniz, AZ, Apr. 2008.

[7] A. L. Stolyar and H. Viswanathan, “Self-organizing dynamic fractional frequency reuse in ofdma system,” in *Proc. IEEE INFOCOM*, Phoeniz, AZ, Apr. 2008.

[8] ——, “Self-organizing dynamic fractional frequency reuse for best-effort traffic through distributed inter-cell coordination,” to appear in *Proc. IEEE INFOCOM*, 2009.

[9] T. Bonald, S. Borst, and A. Proutière, “Inter-cell scheduling in wireless data networks,” in *Proc. European Wireless*, Cyprus, Greece, Apr. 2005.

[10] A. Sang, X. Wang, M. Madihian, and R. D. Gitlin, “Coordinated load balancing, handoff/cell-site selection, and scheduling in multi-cell packet data systems,” in *Proc. ACM MobiCom*, Philadelphia, PA, Sep. 2004, pp. 302–314.

[11] T. Bu, L. Li, and R. Ramjee, “Generalized proportional fair scheduling in third generation wireless data networks,” in *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr. 2006.

[12] F. Kelly, A. Maullo, and D. Tan, “Rate control in communication networks: shadow prices, proportional fairness and stability,” *Journal of the Operational Research Society*, vol. 49, pp. 237–252, Jul. 1998.

[13] J. Mo and J. Walrand, “Fair end-to-end window-based congestion control,” *IEEE/ACM Trans. Networking*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[14] K. Son, Y. Yi, and S. Chong, “Adaptive multi-pattern reuse in multi-cell networks,” Available at http://netsys.kaist.ac.kr/ ∼skio/multipattern.pdf, *Technical Report*, May 2009.

[15] A. L. Stolyar, “On the asymptotic optimality of the gradient scheduling algorithm for multiuser throughput allocation,” *Operations Research*, vol. 53, no. 1, pp. 12–25, Jan. 2005.

[16] H. J. Kushner and P. A. Whiting, “Convergence of proportional-fair sharing algorithms under general conditions,” *IEEE Trans. Wireless Commun.*, vol. 3, no. 4, pp. 1250–1259, 2004.

[17] ITU, “Recommendation ITU-R M.1225: Guidelines for evaluation of radio transmission technologies for IMT-2000,” 1997.