

## MIT Open Access Articles

### *Distributed Throughput Maximization in Wireless Networks via Random Power Allocation*

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

**Citation:** Hyang-Won Lee, et al. "Distributed Throughput Maximization in Wireless Networks via Random Power Allocation." IEEE Transactions on Mobile Computing, vol. 11, no. 4, Apr. 2012, pp. 577–90.

**As Published:** <http://dx.doi.org/10.1109/TMC.2011.58>

**Publisher:** Institute of Electrical and Electronics Engineers (IEEE)

**Persistent URL:** <http://hdl.handle.net/1721.1/112926>

**Version:** Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

**Terms of use:** Creative Commons Attribution-Noncommercial-Share Alike



# Distributed Throughput Maximization in Wireless Networks via Random Power Allocation

Hyang-Won Lee, *Member, IEEE*, Eytan Modiano, *Senior Member, IEEE*, Long Bao Le, *Member, IEEE*

**Abstract**—We develop a distributed throughput-optimal power allocation algorithm in wireless networks. The study of this problem has been limited due to the non-convexity of the underlying optimization problems, that prohibits an efficient solution even in a centralized setting. By generalizing the randomization framework originally proposed for input queued switches to SINR rate-based interference model, we characterize the throughput-optimality conditions that enable efficient and distributed implementation. Using gossiping algorithm, we develop a distributed power allocation algorithm that satisfies the optimality conditions, thereby achieving (nearly) 100% throughput. We illustrate the performance of our power allocation solution through numerical simulation.

**Index Terms**—Throughput-optimal power allocation, randomization framework, SINR-based interference model.

## 1 INTRODUCTION

Resource allocation in multihop wireless networks involves solving a joint link scheduling and power allocation problem which is very difficult in general [2], [3]. Due to this difficulty, most of the existing works in the literature consider a simple setting where all nodes in the network use fixed transmission power levels and the resource allocation problem degenerates into simply a link scheduling problem [4], [5], [6], [7]. Furthermore, the link scheduling problem has been mostly studied assuming a simplistic graph-based interference model.

In fact, the resource allocation problem has been considered mainly in two different network settings. The first setting is a static one which does not take randomness in the traffic arrival processes into consideration. In particular, it is usually assumed users either have unlimited amount of traffic to transmit or have predetermined traffic demands. Here, resource allocation aims at achieving fair share of resource among competing traffic flows or developing resource allocation algorithms which have nice performance properties (e.g., constructing minimum length schedule to support a traffic demands) [8], [9], [10], [11], [12]. The second setting assumes random arrival traffic and one of the main objectives of the resource allocation problem is to maximize the average arrival

rates which can be supported while maintaining network stability.

In the seminal work of [13], Tassiulas and Ephremides introduce the concept of *stability region*, defined as the set of all arrival rate vectors that can be stably supported. They also propose a joint routing and scheduling policy that achieves 100% throughput, meaning that it stabilizes the network whenever the arrival rate vector is in the stability region. More recently, this throughput-optimal policy has been extended to wireless networks with power control [14], [15] and for the scenario where arrival rates lie outside the capacity region [16], [17], [18].

All these resource allocation algorithms, however, require repeatedly solving a global optimization problem which is NP-hard in general [17], [3]. Hence, in multi-hop wireless networks, it may be impractical to find its solution in every time slot due to limited computation capability, and the need for distributed operation. As an alternative, distributed greedy scheduling has been proposed and analyzed [7], [17], [19], [20], [21]. However, most of the existing works in this context adopt the *graph-based interference models*, where transmissions on any two links in the network are assumed to be either in conflict or conflict-free. Moreover, the use of greedy scheduling typically results in throughput reduction by factor of up to 2 under the primary interference model [17], [19] and  $\frac{(2K+1)^2}{\lfloor K/2 \rfloor}$  in  $K$ -hop interference model for  $K \geq 2$  [22].

It has been recognized that graph-based interference models may be overly simplistic because they ignore the cumulative effect of wireless interference. However, going beyond these simplistic interference models is challenging. In fact, the power allocation

*The authors are with the Massachusetts Institute of Technology, Cambridge, MA 02139. (e-mail: {hwlee, modiano, longble}@mit.edu)*  
*This work was partly supported by NSF grant CNS-0915988 and by ARO Muri grant number W911NF-08-1-0238. Hyang-Won Lee was supported by the Korea Research Foundation Grant funded by the Korean Government(MOEHRD)(KRF-2007-357-D00164). Long Bao Le was partially supported by NSERC Postdoctoral Fellowship.*  
*This work was presented in part at the WiOpt conference, June 2009 [1].*

problem under the *SINR rate-based interference model* is non-convex; therefore, obtaining a global optimal power allocation even in a centralized manner is not practical. This non-convexity issue in the power allocation problem has been addressed by several papers [8], [10] considering either the high or low SINR regimes. Recently, it was shown that this problem is NP-hard [23], [24], where the optimality conditions for sum rate maximization are extensively studied.

In this paper, we develop a distributed throughput-optimal power allocation algorithm under the SINR rate-based interference model. As mentioned above, the previously known condition for throughput-optimal power allocation under this model requires solving a non-convex optimization problem for every time slot. Hence, its distributed implementation may be prohibitive in practice. We take a randomization approach to develop the optimality conditions that enable distributed power allocation algorithms. The randomization technique was originally developed for input queued switches [25], and later extended for multi-hop wireless networks assuming the graph-based primary and secondary interference models [4], [5]. Its key feature is that it does not seek to find an optimal schedule in every time slot, and consequently, solving a difficult scheduling problem can be avoided. Motivated by this observation, our work attempts to alleviate the difficulty in solving the non-convex optimization problem involved in optimal power allocation, using randomization.

As mentioned above, the throughput optimal scheduling problem under the graph-based interference model has been relatively well understood. In particular, the randomization has been successfully applied for developing efficient throughput optimal scheduling algorithms [4], [5]. On the other hand, there are few results that deal with the throughput optimal power control problem under the SINR-based interference model in which the amount of interference and noise is explicitly taken into account. In [26], [27], [28], [29], optimal scheduling problems are considered assuming that every transmitting node uses fixed power levels and the success or failure of a transmission is determined by certain SINR threshold.

In contrast, we assume a SINR rate-based interference model where the transmission rate of a link is given as a continuous function of its SINR. In [30], the throughput optimal power control problem was considered under this model, however the performance of the proposed power allocation algorithm is not guaranteed. To the best of our knowledge, there is no known work that assumes the SINR rate-based interference model and solves the throughput optimal power control problem in the stability framework of [13]. As mentioned above, the problem needed to be solved in each time slot was shown to be NP-hard in [23]. Hence, achieving throughput optimality under the SINR rate-based interference model is likely to

be a hard problem. To circumvent this difficulty, we develop new tractable throughput optimality conditions by extending the randomization framework, and develop a distributed power allocation algorithm that satisfies the new optimality conditions.

## 2 MODEL AND PROBLEM DESCRIPTION

We consider a multi-hop wireless network modeled by a graph  $G = (V, E)$ , where  $V$  is the set of nodes and  $E$  is the set of links. Let  $N$  be the number of nodes, i.e.,  $N = |V|$ . It is assumed that there is a link between two neighboring nodes if they want to communicate with each other. We assume that time is slotted and a time slot interval is of unit length. Let  $V(a)$  be the set of node  $a$ 's neighbors, i.e.,  $V(a) = \{b \in V : (a, b) \in E\}$ . We assume bidirectional links, hence link  $(a, b)$  exists whenever  $(b, a)$  does. For simplicity of exposition, we start by assuming that there is only single-hop traffic and single channel available in the network. Extension to the case of multi-hop traffic and multi-channels can be found in [31]. Node  $a$  maintains a data buffer for each outgoing link  $(a, b)$ , and its backlog at time  $t$  is denoted by  $q_{ab}(t)$ .

Denote by  $p_{ab}$  the transmit power allocated to link  $(a, b)$ . Each node  $a$  has a limited power budget  $P_a^{\max}$ , and the total transmit power constraint can be written as  $\sum_{b \in V(a)} p_{ab} \leq P_a^{\max}$ . We assume SINR rate-based interference model. That is, under a power allocation vector  $p = [p_{ab}, \forall (a, b) \in E]$ , link  $(a, b)$ 's rate  $r_{ab}(p)$  is given by

$$r_{ab}(p) = \log \left( 1 + \frac{g_{ab} p_{ab}}{n_b + \sum_{i \in V(a) \setminus \{b\}} \frac{g_{ab} p_{ai}}{g_{ab} p_{ai} + \sum_{i \neq a} g_{ib} \sum_{j \in V(i)} p_{ij}} \right), \quad (1)$$

where  $n_b$  is the noise power, and  $g_{ab}$  is the channel gain from node  $a$  to  $b$ . It is assumed  $g_{ab} = \infty$  if  $a = b$ . Since the nodes are static, the channel gains are assumed to be fixed over time. Note that the second term in the denominator of (1) is self-interference, and the third is mutual interference.

Let  $A_{ab}(t)$  represent the amount of exogenous data that arrive to the buffer at the source of link  $(a, b)$  during slot  $t$ , and  $p(t)$  the power allocation vector for slot  $t$ . Then, the backlog  $q_{ab}(t)$  evolves according to the following dynamics:

$$q_{ab}(t+1) = \max[0, q_{ab}(t) - r_{ab}(p(t))] + A_{ab}(t). \quad (2)$$

The arrival process  $A_{ab}(t)$  is assumed to be i.i.d. over time with average  $\lambda_{ab}$ , i.e.,  $E[A_{ab}(t)] = \lambda_{ab}, \forall t$ . We assume that all arrival processes  $A_{ab}(t)$  have bounded second moments and they are upper-bounded by  $A_{\max}$  (i.e.,  $A_{ab}(t) \leq A_{\max}, \forall (a, b) \in E$ ). Now, we define the network stability.

*Definition 1:* A queue  $q_{ab}(t)$  is called *strongly stable* if

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} E \{q_{ab}(\tau)\} < \infty. \quad (3)$$

A network of queues is called strongly stable if all individual queues are strongly stable.

For convenience, we will instead use the term *stable* to represent the term strongly stable.

Let us drop the indices of a variable to denote its vector form, for example,  $q(t) = [q_{ab}(t), \forall (a, b) \in E]$ . Define the stability region, denoted by  $\Lambda$ , to be the union of arrival rate vectors  $\lambda = (\lambda_{ab}, (a, b) \in E)$  such that there exists a scheduling policy which stabilizes the network queues. In [14], the stability region for wireless networks with power control was characterized. Let  $F$  be the feasible region of transmit power vectors, i.e.,  $F = \{p \geq 0 : \sum_{b \in V(a)} p_{ab} \leq P_a^{\max}, \forall a \in V\}$  where  $p \geq 0$  is component-wise inequality. The stability region  $\Lambda$  consists of all arrival rate vectors  $\lambda = (\lambda_{ab}, (a, b) \in E)$  such that

$$\lambda \in \text{Convex\_Hull}\{r(p) : p \in F\}. \quad (4)$$

Note that it is the convex hull of all the feasible link rate vectors. In [14], it was shown that if in each time slot  $t$ , power is allocated according to the following max-weight rule, then the network will be stable for all arrival rates within the stability region.

$$p^*(t) = \arg \max_{p \in F} \sum_{(a,b)} q_{ab}(t) r_{ab}(p). \quad (5)$$

The optimal solution  $p^*(t)$  may not be unique, but in the case of multiple optimal solutions, our randomization framework performs better. Hence, assuming unique  $q^*(t)$  will give a lower bound on the performance of our randomization framework. Note that in the graph-based interference model, link rates are fixed and the resource allocation problem degenerates into the link scheduling problem; where the max-weight scheduling policy which returns a feasible schedule achieving the maximum weight in each time slot is throughput-optimal.

The optimization problem (5) is nonconvex in  $p$ , and hence, it may not be possible to find an optimal power vector for every time slot  $t$ , even in a centralized manner. We address this issue by using *randomization*, originally proposed for input queued switches [25] and wireless networks under graph-based interference models [4], [5].

### 3 RANDOMIZATION FRAMEWORK

#### 3.1 Background on Randomization Framework

The randomization approach was first developed for scheduling in input queued switches [25], and extended for distributed operations in multi-hop wireless networks [4], [5]. Recall that under these settings, a feasible schedule is to be found in each time slot. The key feature of the randomization approach is that it does not seek to find an optimal schedule in every slot, and hence, it can significantly reduce the computation overhead. In every time slot, the randomization framework does the following:

---

#### Algorithm 1 Randomized Power Control Framework (for each time slot $t$ )

---

1. RAND-POW: Generate a new random power allocation vector  $\tilde{p}(t)$  in a distributed manner.
  2. DECIDE: Determine the current power allocation  $p(t)$  by comparing the previous power allocation  $p(t-1)$  and the new power allocation  $\tilde{p}(t)$ , and selecting the one with higher weight in (5).
- 
- (i) RAND-SCH: generate a new random schedule,
  - (ii) DECIDE: decide on the current schedule by comparing and selecting the better of the new and old schedules (i.e., the one with higher weight in (5)).

*Lemma 1* ([25]): Under the condition that the newly generated schedule in RAND-SCH is optimal with positive probability, the randomization framework achieves 100% throughput.

Note that in an input queued switch the number of possible activations is finite. Hence, it is trivial to develop a random algorithm to satisfy the condition in Lemma 1. Moreover, the comparison in a switch can be done in a centralized manner. However, in multi-hop wireless networks, the DECIDE step is challenging because each node must compare the network-wide weighted sum rates achieved by the two schedules in a distributed manner. In [4], this comparison is localized over connected subgraphs consisting of old and new link activations; where the decisions in one subgraph do not affect the decisions at other subgraphs. The communication overhead can be substantially reduced using this localization.

#### 3.2 Extension to SINR Rate-Based Model

Our work is motivated by the intuition that the difficulty due to the non-convexity in (5) can also be alleviated using this randomization technique. For notational convenience, let  $q(t)^T r(p)$  be the objective value in (5). A natural extension of the randomization framework to SINR rate-based interference model will be as follows. First, in each time slot  $t$ , the nodes generate a new random power allocation vector, denoted by  $\tilde{p}(t)$ , in a distributed manner. Second, the current power vector  $p(t)$  is selected by comparing the new power vector  $\tilde{p}(t)$  and the previous one  $p(t-1)$ ; namely,  $p(t) = \tilde{p}(t)$  if  $q(t)^T r(\tilde{p}(t)) > q(t)^T r(p(t-1))$  and  $p(t) = p(t-1)$  otherwise. These two steps are summarized in Algorithm 1. The key challenge in this setting is that it may not be possible to devise a power allocation policy RAND-POW that has a positive probability of being optimal since the optimal power allocation takes on real-values. Consequently, the randomization approach to the power allocation problem will not be able to achieve 100% throughput as in the case of the graph-based interference model. We address this issue by generalizing the condition

on RAND-SCH in the graph-based interference model; namely, the newly generated power vector is not required to be optimal, but is required to be within a small factor of optimal.

Another challenge lies in the DECIDE part, as the localized comparison in the graph-based interference model cannot work in our setting. With the SINR rate-based interference model, the interference level experienced at a node is affected by all the other nodes in the network. Hence, the localized comparison may lead to a wrong decision, and a network-wide comparison will be inevitable. To resolve this problem, we will use randomized gossiping [32].

We first present new conditions for RAND-POW and DECIDE, that will be used to characterize the performance of randomization framework.

*Condition 1 (C1):* For every time slot  $t$ ,

$$\Pr [q(t)^T r(\tilde{p}(t)) \geq (1 - \gamma_1)q(t)^T r(p^*(t))] \geq \delta_1 > 0,$$

where  $\gamma_1$  and  $\delta_1$  are some positive constants, and  $\tilde{p}(t)$   $p^*(t)$  are the new random power vector and optimal power vector, respectively.

Condition C1 allows for the possibility that the new random power allocation is within a factor of the optimal. Notice that when  $\gamma_1 = 0$ , C1 becomes the condition on RAND-SCH in [4], [25] which requires the new scheduling to be optimal with positive probability. This generalization is the key to dealing with the power control problem (5) using the randomization approach, and the *optimality loss* under this condition will be characterized in Theorem 1.

The following is the condition on DECIDE adopted from [4].

*Condition 2 (C2, [4]):* For every time slot  $t$ ,

$$q(t)^T r(p(t)) \geq (1 - \gamma_2) \max\{q(t)^T r(p(t-1)), q(t)^T r(\tilde{p}(t))\}$$

with probability at least  $1 - \delta_2$ , where  $\gamma_2$  and  $\delta_2 (\ll \delta_1)$  are some positive constants.

Condition C2 requires that the weight attained by the chosen power vector  $p(t)$  should not be less than some factor of the maximum of the weights obtained by  $\tilde{p}(t)$  and  $p(t-1)$ . This condition was considered in [4] to account for imperfect comparison in multi-hop networks. In Section 5, we discuss a distributed implementation of the DECIDE step that satisfies C2.

The achievable stability region under our randomization framework can be characterized as follows:

*Theorem 1:* If RAND-POW and DECIDE in Algorithm 1 satisfy C1 and C2, then it stabilizes the network for any arrival rate vector in  $\rho\Lambda$  where  $\rho < 1 - (\gamma_1 + (1 - \gamma_1)\gamma_2) - 2\sqrt{\frac{\delta_2}{\delta_1}}$ .

*Proof:* Here, we briefly prove the theorem. More detailed version of the proof can be found in [31]. Consider the following Lyapunov function

$$L(q(t)) := \sum_{(a,b) \in E} q_{ab}(t)^2. \quad (6)$$

Then, the expected conditional  $T$ -step Lyapunov drift is bounded as

$$\begin{aligned} \Delta_T(t) &= E \{L(q(t+T)) - L(q(t)|q(t))\} \\ &\leq 2 \sum_{\tau=0}^{T-1} E \{q(t+\tau)^T \lambda - q(t+\tau)^T r^*(t+\tau)|q(t)\} \quad (7) \end{aligned}$$

$$+ B_1 + 2 \sum_{\tau=0}^{T-1} E \{\Psi(t+\tau)|q(t)\} \quad (8)$$

where  $B_1$  is a finite constant,

$$\Psi(t) := q(t)^T r^*(t) - q(t)^T r(t), \quad (9)$$

and  $r^*(t)$  is the optimal rate which corresponds to the optimal power allocation given the queue length vector  $q(t)$  at time  $t$  (i.e., it achieves the maximum weight).

Let  $W^*(t) = q(t)^T r^*(t)$ . Then, using Conditions C1 and C2, we can show

$$\begin{aligned} \sum_{\tau=0}^{T-1} E \{\Psi(t+\tau)|q(t)\} &\leq T \left( \gamma_1 + (1 - \gamma_1)\gamma_2 + \frac{1}{\delta_1 T} \right. \\ &\quad \left. + \delta_2 T \right) W^*(t) + B_2, \quad (10) \end{aligned}$$

where  $B_2$  is a finite constant. The first term in (7) is bounded as

$$\begin{aligned} \sum_{\tau=0}^{T-1} \{q(t+\tau)^T \lambda - q(t+\tau)^T r^*(t+\tau)\} \\ \leq T [q(t)^T \lambda - q(t)^T r^*(t)] + B_3 \quad (11) \end{aligned}$$

where  $B_3$  is a finite number. Now, consider any arrival rate vector  $\lambda$  which lies inside the  $\rho$ -scaled stability region (i.e., inside  $\rho\Lambda$ ) for  $0 < \rho < 1$ . Then, we have  $\frac{\lambda}{\rho} \in \Lambda$  and hence, the scaled rate vector  $\frac{\lambda}{\rho}$  can be represented as a convex combination of feasible rate vectors:

$$\frac{\lambda}{\rho} = \sum_i \alpha_i r(p^i), \quad (12)$$

where  $\sum_i \alpha_i = 1$  and  $\alpha_i \geq 0, \forall i$ , and  $p^i$  is a feasible power vector. Multiplying both side of (12) by  $q^T/\rho$  yields

$$q^T \lambda = \rho \sum_i \alpha_i q^T r(p^i) \leq \rho \sum_i \alpha_i q^T r^* = \rho q^T r^*, \quad (13)$$

where  $r^*$  is the optimal rate which achieves the maximum weight (i.e.,  $q^T r^* = \max_{p \in F} q^T r(p)$ ). The above inequality also implies

$$q^T \lambda \leq q^T r^* = W^*(t) \quad (14)$$

for any  $\lambda$  lying inside  $\rho\Lambda$ .

Using (13), the bound in (11) can be rewritten as

$$\begin{aligned} \sum_{\tau=0}^{T-1} \{q(t+\tau)^T \lambda - q(t+\tau)^T r^*(t+\tau)\} \\ \leq -T(1 - \rho)W^*(t) + B_3. \quad (15) \end{aligned}$$

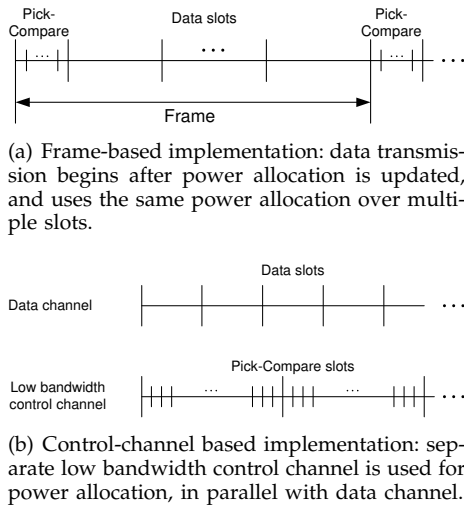


Fig. 1. Frame-based and control-channel based implementations

Using (10) and (15), the conditional expectation of  $T$ -step Lyapunov drift in (7)-(8) can be bounded as

$$\Delta_T(t) \leq -2T \times \left\{ 1 - \rho - \gamma_1 - (1 - \gamma_1)\gamma_2 - \frac{1}{\delta_1 T} - \delta_2 T \right\} W^*(t) + B_4$$

where  $B_4 = B_1 + 2B_3 + 2B_2$  is a finite number. Now, the proof can be completed by choosing  $T = \sqrt{1/(\delta_1 \delta_2)}$ , and applying the inequality (14) and the condition  $\rho < 1 - \gamma_1 - (1 - \gamma_1)\gamma_2 - 2\sqrt{\frac{\delta_2}{\delta_1}}$ .  $\square$

When  $\gamma_1$  is 0, i.e., when a new power vector is optimal with probability  $\delta_1$ , the obtained throughput mainly depends on the comparison performance ( $\gamma_2$ ). However, the throughput loss increases as  $\gamma_1$  increases. In case of perfect comparison (i.e.,  $\gamma_2 = 0$  and  $\delta_2 = 0$ ), the throughput loss depends only on the optimality loss in the random power allocation. In brief, our randomized power control framework can achieve nearly 100% throughput if we can develop a power allocation policy (RAND-POW) and a comparison algorithm (DECIDE) satisfying conditions  $C1$  and  $C2$  with small  $\gamma_1$ ,  $\gamma_2$  and  $\delta_2$ . In the rest of the paper, we focus on developing such algorithms. In particular, in Section 4 we develop a random power allocation policy that satisfies  $C1$  and in Section 5 we develop a comparison algorithm that satisfies  $C2$ .

### 3.3 Frame-based Implementation

In this section, we discuss some issues arising in the implementation of our randomization framework. First, the RAND-POW step can be easily implemented as it is easy to generate a random power vector in a distributed manner, as we demonstrate in Section 4. For the DECIDE phase, each node has to estimate the *global* weights  $q^T r$  in order to make the same decision on the selection of the current power allocation. In small networks, a centralized entity may exist (e.g., base station in cellular networks), hence

comparison and decision can be implemented in a centralized manner. In large networks, however, such a centralized comparison is prohibitive and we adopt gossiping for distributed comparison.

In the proof of Theorem 1, we assumed for simplicity that the power allocation is updated for each data transmission slot by running the RAND-POW and DECIDE steps. However, it may not be practical to run these two steps on a slot-by-slot basis because the DECIDE step may require a significant amount of communications. In fact, this assumption can be easily relaxed by running the RAND-POW and DECIDE on a frame basis as shown in Fig. 1(a); where they are performed for every multiple data transmission slots so that the same power allocation is kept for multiple data slots. By doing so, the control overhead can be significantly reduced. Moreover, it was shown in [4] that this frame-based scheduling still achieves throughput optimality as long as the RAND-POW and DECIDE steps are performed at regular intervals. Alternatively, the power control algorithm can be done on a separate low bandwidth control channel, in parallel with data transmission, as shown in Fig. 1(b). Again, throughput optimality can be achieved as long as a new power allocation is generated at regular (finite duration) intervals. The advantage of this implementation over the frame-based is clear, that is, the data transmission does not need to wait until the update of power allocation is finished, and consequently it will achieve better performance.

## 4 RANDOMIZED POWER ALLOCATION

We present a power allocation policy RAND-POW that satisfies  $C1$ , i.e., finds with positive probability a power vector within a small factor of the optimal value in (5). The problem (5) is to maximize

$$p^* = \arg \max_{p \in F} \sum_{a \in V} \sum_{b \in V(a)} q_{ab} \times \log \left( 1 + \frac{g_{ab} p_{ab}}{n_b + g_{ab} \sum_{i \in V(a) \setminus b} p_{ai} + \sum_{i \neq a} g_{ib} \sum_{j \in V(i)} p_{ij}} \right), \quad (16)$$

where  $F = \{p \geq 0 : \sum_{b \in V(a)} p_{ab} \leq P_a^{\max}, \forall a \in V\}$ . Clearly, the new power vector  $\tilde{p}$  in RAND-POW is desired to be as close to  $p^*$  as possible, and hence, identifying the optimality properties of (16) would be helpful for generating such  $\tilde{p}$ . The following lemma characterizes some useful properties of  $p^*$ .

*Lemma 2:* Under the optimal power allocation  $p^*$ ,

- (i) A node does not transmit while receiving, and vice versa,
- (ii) A node transmits to at most one of its neighbors.

*Proof:* Recall the assumption  $g_{aa} = \infty, \forall a$ . Under this assumption, if a node transmits while trying to receive, it will achieve zero rate due to infinite interference. Hence, at optimal  $p^*$ , case (i) does not happen.

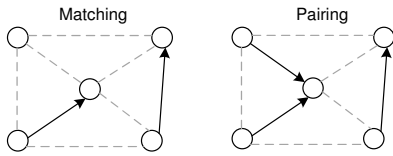


Fig. 2. *Matching*: either (single) transmission or reception is allowed for each node. *Pairing*: node can receive from multiple neighbors but transmitting to multiple neighbors are not allowed.

To prove (ii), let  $p_a^* = \sum_{b \in V(a)} p_{ab}^*$ , i.e.,  $p_a^*$  is the total power transmitted by node  $a$  at the optimal allocation. It is obvious that solving the problem (16) with the additional constraints  $\sum_{b \in V(a)} p_{ab} = p_a^*, \forall a$  will result in the same optimal solution. Hence, the objective function in (16) can be written as

$$\sum_a \sum_{b \in V(a)} q_{ab} \log \left( 1 + \frac{g_{ab} p_{ab}}{n_b + g_{ab}(p_a^* - p_{ab}) + \sum_{i \neq a} g_{ib} p_i^*} \right).$$

Clearly, changing transmit power  $p_{ab}, \forall b \in V(a)$  under fixed total power does not affect the mutual interference, but only changes the self-interference. Hence, the new optimization problem can be solved separately with respect to each node, i.e., for each  $a$ , we only need to maximize

$$\sum_{b \in V(a)} q_{ab} \log \left( 1 + \frac{g_{ab} p_{ab}}{n_b + g_{ab}(p_a^* - p_{ab}) + \sum_{i \neq a} g_{ib} p_i^*} \right)$$

subject to  $p \geq 0$  and  $\sum_{b \in V(a)} p_{ab} = p_a^*$ . Since this function is *strictly convex* in  $[p_{ab}, \forall b \in V(a)]$ , it is maximized at a corner point, i.e.,  $p_{ab} = p_a^*$  for some  $b \in V(a)$  and zero for all others. This shows that it is optimal for each node to transmit to at most one neighbor.  $\square$

According to Lemma 2, at an optimal allocation, a node is not allowed to transmit to multiple neighbors, and to be a transmitter and receiver simultaneously. Note, however, that it is possible for a node to receive from multiple transmitters. This is in contrast to a *matching* in which a node cannot be shared by multiple edges. For ease of exposition, we define the notion of a *pairing* as follows:

*Definition 2*: Assume that the tail and the head of a directed edge denote a transmitter and a receiver respectively. A directed subgraph of  $G$  is called a *pairing* if it satisfies (i) and (ii) in Lemma 2.

Note that a pairing is different from a matching because it allows a node to be shared by multiple edges. Figure 2 shows an example of matching and pairing

#### 4.1 Transmitter-Receiver Pairing

From Lemma 2, it is clear that finding a power allocation can be decomposed into two steps. First, find a pairing, and then select the transmit power levels

---

#### Algorithm 2 RAND-PAIR

---

- 1: Each node  $a$  decides to be a transmitter w.p.  $1/2$  and a receiver w.p.  $1/2$ , and initializes  $I_{ab} = 0, \forall b \in V(a)$ .
  - 2: Each *transmitting node*  $a$  sends a pair-request message (PQM) to one of its neighbors in  $V(a)$  uniformly at random.
  - 3: If node  $b$  receives a PQM, one of the following happens:
    - (i) If node  $b$  is a receiver, then it accepts the request and sends a pair-request-accepted message (PAM) to node  $a$ .
    - (ii) Otherwise, ignore the PQM and nothing happens for node  $a$ .
  - 4: If node  $a$  receives a PAM from node  $b$ , set  $I_{ab} = 1$ , meaning that node  $b$  is a receiver of node  $a$ .
- 

for the given pairing. Since there is a finite number of pairings, and at least one of them is optimal, it is easy to generate an optimal pairing with positive probability. One such algorithm is given by RAND-PAIR (see Algorithm 2). Let  $I_{ab} = 1$  if node  $a$  transmits to its neighbor  $b$ , and 0 otherwise. The goal of RAND-PAIR is to generate a vector  $I = [I_{ab}, b \in V(a), a \in V]$  satisfying the pairing constraints. To do this, each node  $a$  decides to be a transmitter with probability (w.p.)  $\frac{1}{2}$  and a receiver w.p.  $\frac{1}{2}$ . Then, each transmitting node  $a$  sends a pair-request message to one of its neighbors  $b$ . If  $b$  decided to be a receiver, it accepts the request and sends an acceptance message. Otherwise, it is ignored and nothing happens. Once node  $a$  receives the acceptance message, it updates  $I_{ab}$  to  $I_{ab} = 1$ . This algorithm has  $O(1)$  computation and communication complexity, and will find an optimal pairing with positive probability, as stated in the following lemma.

*Lemma 3*: Algorithm RAND-PAIR finds an optimal pairing with probability at least  $(4N)^{-N}$ .

*Proof*: See [31].  $\square$

Note that in the interference graph model, a new scheduling should be a max-weight matching (or independent set in general) with positive probability. Because the max-weight matching is one of maximal matchings, and such a probability can be increased by performing multiple iterations until the obtained matching becomes maximal. However, in our case, maximal pairing<sup>1</sup> may not be always optimal. Hence, performing multiple iterations does not necessarily enhance the probability of being optimal, and further it may not guarantee that the obtained pairing has a positive probability of being optimal.

1. A pairing  $I$  is maximal if adding a link (not in  $I$ ) to  $I$  makes it no longer a pairing

**Algorithm 3** RAND-PSEL (for given pairing  $I$ )

- 
- 1: Each node  $a$  initializes  $\tilde{p}_{ab} = 0, \forall b \in V(a)$ .
  - 2: Every *paired transmitting node*  $a$  does the following:
    - (i) Select a number, say  $u$ , from  $[0, P_a^{\max}]$  uniformly at random, and set  $\tilde{p}_{ab} = u$  for  $b$  such that  $I_{ab} = 1$ .
- 

**4.2 Power Level Selection**

Now what remains is to select a power level which together with RAND-PAIR satisfies  $C1$ . Recall that RAND-PAIR generates a pairing  $I = [I_{ab}, b \in V(a), a \in V]$ . Given this pairing, the problem (16) is rewritten by:

$$p^*(I) = \arg \max_{p \in F} \sum_{a \in V} \sum_{b: I_{ab}=1} q_{ab} \log \left( 1 + \frac{g_{ab} p_{ab}}{n_b + \sum_{\substack{i \neq a \\ j: I_{ij}=1}} g_{ib} p_{ij}} \right). \quad (17)$$

Notice that the self-interference has been removed and the mutual interference has been simplified due to the constraints (i) and (ii) in Lemma 2. Since the pairing  $I$  found by RAND-PAIR has a positive probability of being optimal, the condition  $C1$  can be satisfied if a power level is selected such that it is within a factor of the objective in (17) with positive probability. To meet this requirement, Algorithm RAND-PSEL simply chooses power levels uniformly at random. In particular, each *transmitting node*  $a$  randomly selects its transmit power from the feasible region, i.e.,  $[0, P_a^{\max}]$ . This random power selection meets the requirement as shown in the following lemma. Assume  $P_a^{\max} = 1, \forall a$ .

*Lemma 4:* For any  $\epsilon \in (0, 1)$ , Algorithm RAND-PSEL generates a power vector  $\tilde{p}$  such that  $\tilde{p} \in B(p^*(I), \epsilon)$  with probability at least  $\left(\frac{\epsilon}{\sqrt{N}}\right)^N$ , where  $B(p^*(I), \epsilon) = \{p \in F : \|p - p^*(I)\|_2 \leq \epsilon\}$ .

*Proof:* See [31].  $\square$

Note that this lemma can be easily extended to the case of general  $P_a^{\max}$ . Combining Lemmas 3 and 4, we can show that Condition  $C1$  can be satisfied by RAND-PAIR and RAND-PSEL.

*Theorem 2:* Choosing a power allocation according to RAND-PAIR and RAND-PSEL satisfies  $C1$  with arbitrarily small  $\gamma_1 > 0$  and positive  $\delta_1$  which is a function of  $\gamma_1$ .

*Proof:* Let  $f(p)$  be the objective function in (17), and consider an arbitrary  $\gamma_1 \in (0, 1)$ . Due to the continuity of  $f(p)$ , there exists  $\epsilon > 0$  such that  $f(p) \geq (1 - \gamma_1)f(p^*(I))$  for any feasible  $p$  such that  $\|p - p^*(I)\| \leq \epsilon$ . Let  $I$  be a pairing generated by RAND-PAIR and  $\tilde{p}$  be a power vector obtained through RAND-PSEL, given pairing  $I$ . Let  $I^*$  be an optimal

pairing. Then, it follows that

$$\begin{aligned} & \Pr [q^T r(\tilde{p}) \geq (1 - \gamma_1)q^T r(p^*)] \\ &= \Pr [f(\tilde{p}) \geq (1 - \gamma_1)f(p^*(I)) | I = I^*] \Pr [I = I^*] \\ &\geq \Pr [\|\tilde{p} - p^*(I)\|_2 \leq \epsilon | I = I^*] \Pr [I = I^*] \\ &\geq \left(\frac{\epsilon}{\sqrt{N}}\right)^N \cdot (4N)^{-N} = \left(\frac{\epsilon}{4N^{3/2}}\right)^N, \end{aligned}$$

where the last inequality is due to Lemmas 3 and 4. Therefore, the power allocation obtained by RAND-PAIR and RAND-PSEL achieves at least  $(1 - \gamma_1)$  fraction of optimal value of problem (16) with probability at least  $\delta_1 = \left(\frac{\epsilon}{4N^{3/2}}\right)^N > 0$ , satisfying Condition  $C1$ .  $\square$  Therefore, the optimality loss  $\gamma_1$  can be arbitrarily small (with small enough  $\epsilon$  and thus small probability  $\delta_1$ ). According to Theorem 1, the throughput loss due to this optimality loss ( $\gamma_1$ ) under our power allocation is negligible, as long as  $\delta_2 \ll \delta_1$ .

*Remark:* Theorem 2 implies that the random power allocation hits a near optimal solution in every  $\left(\frac{4N^{3/2}}{\epsilon}\right)^N$  slots (in average sense). As a consequence it can experience large delay or network backlog, although our work in this paper focuses on the long-term throughput performance. In fact, recent results in [33] show that there may not exist a polynomial time (deterministic or randomized) throughput optimal policy for NP-hard scheduling problem such that it achieves polynomial network backlog. The power allocation problem (5) contains a maximum weight independent set problem [24]. Hence, any polynomial time power allocation policy that takes on the problem (5) will experience large delay, as our random power allocation algorithm will. This implies that our random power allocation algorithm may not scale very well as the network size grows. As mentioned above, in this paper, we focus on the throughput performance, and we leave this delay and scalability issue as future research.

**5 COMPARISON AND AGREEMENT**

The goal of the DECIDE step is to choose a power allocation  $p(t)$  by selecting one of the two power allocations  $p(t-1)$  and  $\tilde{p}(t)$ , so that Condition  $C2$  can be satisfied. Such a selection is easy in a centralized setting; namely, a central entity can compare  $q(t)^T r(p(t-1))$  and  $q(t)^T r(\tilde{p}(t))$ , pick the one having larger value, and disseminate the selection to every node. In small networks, such a centralized comparison might be possible, or a spanning tree could be computed in a distributed manner and used for the comparison [5]. However, in large networks, such a centralized computation is prohibitive. For this reason, we develop a distributed DECIDE policy by using randomized gossiping [32]. It consists of two procedures, COMPARE and AGREE. The COMPARE procedure estimates the objective values achieved under the new and old power allocations, and the AGREE procedure uses these estimates to make a



unanimous decision on the selection of current power allocation.

Let  $x_b^{\text{new}}(0)$  and  $x_b^{\text{old}}(0)$  be the weighted (receiving) rates at node  $b$  under the new power  $\tilde{p}(t)$  and old power  $p(t-1)$ , respectively. Then, they can be expressed as  $x_b^{\text{new}}(0) = \sum_{a \in V} q_{ab}(t)r_{ab}(\tilde{p}(t))$  and  $x_b^{\text{old}}(0) = \sum_{a \in V} q_{ab}(t)r_{ab}(p(t-1))$ . Let  $X_{\text{new}} = \sum_{a \in V} x_a^{\text{new}}(0)$  and  $X_{\text{old}} = \sum_{a \in V} x_a^{\text{old}}(0)$ , i.e.,  $X_{\text{new}}$  and  $X_{\text{old}}$  are the objective function values under the newly generated power vector and the old power vector, respectively. The DECIDE step must choose the new power allocation if  $X_{\text{new}} > X_{\text{old}}$ , and the old one if  $X_{\text{new}} \leq X_{\text{old}}$ . This can also be accomplished using the average values  $\bar{X}_{\text{new}}$  and  $\bar{X}_{\text{old}}$  instead of  $X_{\text{new}}$  and  $X_{\text{old}}$ , where  $\bar{X}_{\text{new}} = X_{\text{new}}/N$  and  $\bar{X}_{\text{old}} = X_{\text{old}}/N$ . Therefore, if every node can compute an accurate estimate of  $\bar{X}_{\text{new}}$  and  $\bar{X}_{\text{old}}$ , they will be able to make a decision leading to C2. A randomized gossiping algorithm is used to estimate  $\bar{X}_{\text{new}}$  and  $\bar{X}_{\text{old}}$ . Note that gossiping has been used extensively for computing averages (See [32], [34] and references therein).

Typically, gossiping generates a matching for each iteration. Let  $x_a(k)$  be the value at node  $a$  after iteration  $k$ . The initial value is thus  $x_a(0)$  and the global average is  $\sum_a x_a(0)/N$ . If any two nodes  $a$  and  $b$  share the same link under the current matching, then they update their values to their average, i.e.,  $x_a(k+1) = x_b(k+1) = \frac{x_a(k)+x_b(k)}{2}$ . Gossiping keeps generating a random matching for this averaging operation, and every node eventually obtains an estimate of the global average  $\sum_a x_a(0)/N$ . In this paper, we use a random matching policy in [32] that works as follows. Let  $d(a)$  be the degree of node  $a$ , i.e.,  $d(a) = |V(a)|$  and  $d^*$  be the maximum node degree, i.e.,  $d^* = \max_{a \in V} d(a)$ . Each node  $a$  decides to be active with probability (w.p.)  $\frac{1}{2}$  and inactive w.p.  $\frac{1}{2}$ . An active node  $a$  does nothing w.p.  $1 - \frac{d(a)}{d^*}$ , and randomly contacts one of its neighbors w.p.  $\frac{d(a)}{d^*} \frac{1}{2}$ . Consider an inactive node  $b$ . If  $b$  is contacted by node  $a$  while it has not been contacted by any other, then nodes  $a$  and  $b$  average their values w.p.  $(1 - \frac{1}{2d^*})^{d^* - d(b)}$ . Otherwise, nothing happens for  $a$ .

## 5.1 COMPARE and AGREE

The COMPARE procedure estimates the averages  $\bar{X}_{\text{new}}$  and  $\bar{X}_{\text{old}}$  using the gossiping described above, and is shown in Algorithm COMPARE. Note that at each iteration, a matching is generated and any two nodes sharing a link in that matching average their values. Note also that the same matching is used for new and

2. Under the algorithm COMPARE, each active node  $a$  has  $\frac{1}{2}d(a)$  inactive neighbors in average. Hence for better chance of matching, it is desirable for an active node with high degree to make an attempt to match with high probability, while a node with low degree is desired to attempt with low probability. This is why the contact probability is proportional to the node degree.

---

### Algorithm 4 COMPARE

---

1: For iteration  $k = 1, \dots, K$ , do the following:

- (i) Each node  $a$  updates  $x_a^{\text{new}}(k) = x_a^{\text{new}}(k-1)$  and  $x_a^{\text{old}}(k) = x_a^{\text{old}}(k-1)$ .
  - (ii) Each node decides to be active w.p.  $1/2$  and inactive w.p.  $1/2$ . An active node  $a$  does nothing w.p.  $1 - \frac{d(a)}{d^*}$ , and contacts one of its neighbors uniformly at random (i.e., with equal probability  $\frac{1}{d^*}$ ).
  - (iii) If node  $b$  is contacted, one of the following happens:
    - (b) If  $b$  is inactive and has not been contacted, they average as  $x_a^{\text{new}}(k) = x_b^{\text{new}}(k) = \frac{x_a^{\text{new}}(k-1) + x_b^{\text{new}}(k-1)}{2}$  and  $x_a^{\text{old}}(k) = x_b^{\text{old}}(k) = \frac{x_a^{\text{old}}(k-1) + x_b^{\text{old}}(k-1)}{2}$  w.p.  $(1 - \frac{1}{2d^*})^{d^* - d(b)}$ .
    - (c) Otherwise,  $b$  ignores the contact and nothing happens for  $a$ .
- 

---

### Algorithm 5 AGREE

---

- 1: Run gossiping for  $\tilde{K}$  iterations to estimate the average  $\sum_{a \in V} z_a(0)/N$ .
  - 2: Each node  $a$  selects the new power if  $z_a(\tilde{K}) \geq 1$ , and the old one otherwise.
- 

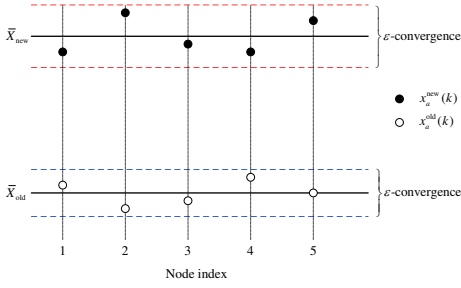
old values. After  $K$  iterations, every node will obtain the estimates of new and old average values  $x_a^{\text{new}}(K)$  and  $x_a^{\text{old}}(K)$ .

If the estimates are exact, a unanimous decision satisfying C2 can be easily made since every node  $a$  will have  $x_a^{\text{new}}(K) > x_a^{\text{old}}(K)$  (or  $x_a^{\text{new}}(K) \leq x_a^{\text{old}}(K)$ ). Such a unanimous decision can also be guaranteed if the estimates are highly accurate, provided that the difference  $|\bar{X}_{\text{new}} - \bar{X}_{\text{old}}|$  is sufficiently large. However, in the case of small difference, decisions can be mixed even under highly accurate estimation (See Fig. 3), which can lead to the violation of C2. An additional procedure is thus needed to ensure that every node makes the same and right decision.

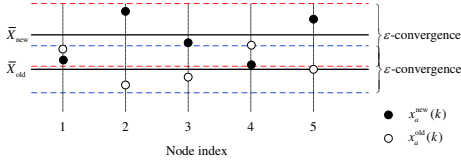
The AGREE procedure keeps the decision made by COMPARE if it is unanimous. Otherwise, it keeps the old power allocation. Note that in the case of small difference, this selection policy will not incur big losses in throughput because there is only a small difference between selecting either the new or the old power allocation. To do this, the AGREE procedure uses the estimates  $x_a^{\text{new}}(K)$  and  $x_a^{\text{old}}(K)$  as follows. Each node  $a$  initiates a variable  $z_a(0)$  as follows:

$$z_a(0) = \begin{cases} 1 & \text{if } x_a^{\text{new}}(K) > x_a^{\text{old}}(K) \\ 0 & \text{if } x_a^{\text{new}}(K) \leq x_a^{\text{old}}(K). \end{cases}$$

Namely,  $z_a(0)$  is equal to 1 if node  $a$  prefers the new power allocation and 0 otherwise. It runs gossiping for  $\tilde{K}$  iterations (as in COMPARE) to estimate the average  $\bar{Z} = \sum_{a \in V} z_a(0)/N$ . After that, each node  $a$



(a) Large difference:  $\epsilon$ -convergence may result in unanimous decision



(b) Small difference: decisions can be mixed even under  $\epsilon$ -convergence

Fig. 3. Impact of difference  $|\bar{X}_{new} - \bar{X}_{old}|$  on unanimous decisions

decides to use the new power if  $z_a(\tilde{K}) = 1$  and the old one otherwise, where  $z_a(k)$  is the value at node  $a$  after iteration  $k$ . Note that if  $z_a(0)$ 's are all zero or all one, then the convergence and unanimous decision is guaranteed immediately. We will show that this is the right decision. If there is a mixture of decisions at the end of COMPARE, the AGREE procedure tries to keep the old power allocation. The following lemmas show a unanimous decision is the right decision; hence justifying the AGREE.

*Lemma 5:* Suppose that there was an agreement after COMPARE, i.e.,  $z_a(0)$ 's are all zero or all one. Then, it is the right decision regardless of the values of  $\bar{X}_{new}$  and  $\bar{X}_{old}$  in that the power allocation selected based on  $z_a(0)$ 's achieves the objective value of  $\max\{X_{new}, X_{old}\}$ . As a consequence, *unanimous wrong* decisions cannot happen after COMPARE.

*Proof:* To prove Lemma 5, we need the following lemma. Again, let  $x_a$  be  $x_a^{new}$ ,  $x_a^{old}$  or  $z_a$ .

*Lemma 6:* For every  $k \geq 0$ , the sum is conserved as

$$\sum_{a \in V} x_a(k) = \sum_{a \in V} x_a(0).$$

*Proof:* Let  $x_a(k)$  and  $x(k)$  be the value of node  $a$  after iteration  $k$  and its vector, respectively. Denote by  $M(k)$  the matching found in iteration  $k$ . The update of node values can be expressed as a linear equation by

$$x(k+1) = W(k)x(k), \quad (18)$$

where  $W(k)$  is an  $N \times N$  matrix given by

$$W(k) = I - \sum_{(a,b) \in M(k)} \frac{(e_a - e_b)(e_a - e_b)^T}{2}, \quad (19)$$

where  $I$  is the identity matrix and  $e_i$  is the  $i$ -th element vector whose  $i$ -th coordinate is 1 and all

others are zero. The first term in  $W(k)$  corresponds to the original value of each node and the second term describes the change from the original value. For example, when node  $a$  averages with  $b$ , its new value becomes  $\frac{1}{2}x_a(k) + \frac{1}{2}x_b(k) = x_a(k) - \frac{1}{2}x_a(k) + \frac{1}{2}x_b(k)$ , where the first term corresponds to  $I$  and the last two terms correspond to the second term in (19). Note that the matrix  $W(k)$  is doubly stochastic, and as a consequence, the following holds:

$$\begin{aligned} \sum_{a \in V} x_a(k+1) &= \bar{1}^T \cdot x(k+1) \\ &= \bar{1}^T W(k)x(k) \\ &= \bar{1}^T \cdot x(k) = \sum_{a \in V} x_a(k). \end{aligned}$$

The third equality is due to the fact that  $W(k)$  is doubly stochastic. This proves the lemma.  $\square$

We now prove Lemma 5. Under the assumption of unanimous decisions after COMPARE, there can be only two cases including (i)  $x_a^{new}(K) > x_a^{old}(K), \forall a$  or (ii)  $x_a^{new}(K) \leq x_a^{old}(K), \forall a$ . Suppose case (i), in which case every node selects the new power. Then, it follows that

$$\begin{aligned} \sum_a x_a^{new}(K) &> \sum_a x_a^{old}(K) \\ \Rightarrow \sum_a x_a^{new}(0) &> \sum_a x_a^{old}(0), \end{aligned}$$

where the second line is due to Lemma 6. Therefore, selecting the new power is the right decision. Case (ii) can be proved similarly.  $\square$

Therefore, it is desirable to keep any unanimous decision made after COMPARE, because the better power allocation is always selected under such a decision. This justifies the AGREE procedure that always keeps unanimous decisions made after COMPARE.

We now analyze and prove that the combination of COMPARE and AGREE can satisfy Condition C2. For the proof, we need to define some parameters. Let  $x(k)$  be the vector of  $x_a(k)$ 's and  $\bar{X} = \sum_a x_a(0)/N$ , where  $x_a$  can be  $x_a^{new}$ ,  $x_a^{old}$  or  $z_a$ .

*Definition 3 ( $\epsilon$ -convergence time, [32]):* For given  $\delta > 0$ , the  $\epsilon$ -convergence time  $K(\epsilon, \delta)$  is defined by

$$K(\epsilon, \delta) = \sup_{x(0)} \inf \left\{ k : \Pr \left[ \frac{\|x(k) - \bar{X}\bar{1}\|}{\|x(0)\|} \leq \epsilon \right] \geq 1 - \delta \right\} \quad (20)$$

where  $\|\cdot\|$  is  $l_2$ -norm.

Briefly, the  $\epsilon$ -convergence time is the time until the estimation vector  $x(k)$  falls into the  $\epsilon$ -neighborhood (in relative sense) of the average vector  $\bar{X}\bar{1}$  with high probability.

*Assumption 1:* Fix arbitrary  $\gamma_2, \delta_2 \in (0, 1)$ . Consider positive constants  $\hat{\epsilon}, \epsilon, \delta, \bar{\epsilon}, \bar{\delta}$  and assume the following:

$$\begin{aligned} 0 < \hat{\epsilon} &\leq \frac{\gamma_2}{2-\gamma_2}, \epsilon = \frac{\hat{\epsilon}}{N}, 0 < \delta \leq \frac{\delta_2}{2} \\ 0 < \bar{\epsilon} &< \frac{1}{N-1}, \bar{\delta} = \frac{\hat{\epsilon}}{N}. \end{aligned}$$

Assume further that  $K = K(\epsilon, \delta)$  in COMPARE and  $\bar{K} = K(\bar{\epsilon}, \bar{\delta})$  in AGREE.

Let  $\bar{X}_{agr}$  be the average objective value achieved by the above described DECIDE algorithm that runs

COMPARE and then AGREE. It can be proved that this policy satisfies C2 as shown in Theorem 3.

*Theorem 3:* Consider any  $\gamma_2, \delta_2 \in (0, 1)$ . Under Assumption 1, the DECIDE algorithm (COMPARE and AGREE) achieves

$$\Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2) \max\{\bar{X}_{\text{new}}, \bar{X}_{\text{old}}\}] \geq 1 - \delta_2.$$

*Proof:* See Section 5.2.  $\square$

**Remark:** As seen above, the  $\epsilon$ -convergence time  $K(\epsilon, \delta)$  is a critical parameter because Condition C2 can be guaranteed after  $\epsilon$ -convergence time in COMPARE and AGREE. It is known that in a line or ring topology, it is given by  $\Theta(-N^2 \log(\epsilon\delta))$  [35]. Moreover, in a complete graph, it is given by  $K(\epsilon, \delta) = \Theta(-\log(\epsilon\delta))$  [32]. In wireless networks, the topology can be controlled by adjusting the coding and transmission rate. That is, if a strong coding is used with low transmission rate, then the communication range can be increased (for the purpose of control signalling only). This will make the topology closer to a complete graph. In particular, a small network could be made a complete graph. Hence, if this is used for gossiping, the  $\epsilon$ -convergence time will be substantially improved<sup>3</sup>. The convergence time can be further improved by exploiting the geographic information. In [37], [38], geographic gossip algorithms were developed such that their convergence time is  $O(N)$ . Clearly, this is order optimal for network-wide averaging, and therefore, the gossiping-based comparison can be a practical solution in real wireless networks.

**Remark:** We briefly discuss the total overhead of our algorithm. Recall that our algorithm consists of RAND-POW and DECIDE. In RAND-POW,  $O(N)$  and  $O(1)$  computations are needed respectively for RAND-PAIR, and RAND-PSEL. The DECIDE step runs two rounds of gossip algorithm which requires  $O(N^3)$  computations in the worst case [32]. Therefore, our algorithm requires  $O(N^3)$  computations in total.

## 5.2 Proof of Theorem 3

The theorem is proved in three steps. First, in Section 5.2.1, we analyze the case of large difference  $|\bar{X}_{\text{new}} - \bar{X}_{\text{old}}|$ . In particular, we show that a unanimous decision can be easily made, given that every has obtained a good estimate ( $\epsilon$ -convergence) of averages. Since any unanimous decision made after COMPARE is the right decision, the probability of right decision is the probability of  $\epsilon$ -convergence in COMPARE. We show that this probability is high. Second, in Section 5.2.2, we deal with the case of small difference, where even a good estimate can possibly result in mixed decisions. The key to dealing with this case is that

3. In fact, the techniques used in [35], [36] for analyzing convergence time show that as the number of disjoint paths increases, the convergence speed increases. Hence, such a topology control will enhance the convergence speed. More details can be found in [31].

selecting either the new or old power allocation is not a bad choice due to  $|\bar{X}_{\text{new}} \approx \bar{X}_{\text{old}}|$ . We show that the AGREE procedure attains a unanimous decision with probability, which in this case implies a fairly good choice. Finally, Section 5.2.3 combines these two results to show that COMPARE and AGREE will select the power allocation that achieves almost the maximum of new and old objective values with high probability.

### 5.2.1 The Case of Large Difference

Let us first delineate between large and small differences. Consider an arbitrary  $\hat{\epsilon} \in (0, 1)$ , and let  $\epsilon = \hat{\epsilon}/N$ . Recall the definition of  $K(\epsilon, \delta)$  in (20). It can be easily shown that under Algorithm COMPARE, for any  $k \geq K(\epsilon, \delta)$ ,

$$\begin{aligned} |x_a^{\text{new}}(k) - \bar{X}_{\text{new}}| &\leq \hat{\epsilon} \bar{X}_{\text{new}}, \forall a \in V \\ |x_a^{\text{old}}(k) - \bar{X}_{\text{old}}| &\leq \hat{\epsilon} \bar{X}_{\text{old}}, \forall a \in V \end{aligned} \quad (21)$$

with probability at least  $1 - \delta$ . Define  $\mathcal{E}_1$  as the event that (21) is satisfied, under the assumption that  $K = K(\epsilon, \delta)$  in Algorithm COMPARE. Then, it is obvious that  $\Pr[\mathcal{E}_1] \geq 1 - \delta$ . Define  $\mathcal{E}_2$  as the event<sup>4</sup> that  $\bar{X}_{\text{new}} > \frac{1+\hat{\epsilon}}{1-\hat{\epsilon}} \bar{X}_{\text{old}}$  or  $\bar{X}_{\text{new}} \leq \frac{1-\hat{\epsilon}}{1+\hat{\epsilon}} \bar{X}_{\text{old}}$ . Then, its complement  $\mathcal{E}_2^C$  is the event that  $\frac{1-\hat{\epsilon}}{1+\hat{\epsilon}} \bar{X}_{\text{old}} < \bar{X}_{\text{new}} \leq \frac{1+\hat{\epsilon}}{1-\hat{\epsilon}} \bar{X}_{\text{old}}$ . Note that the event  $\mathcal{E}_2$  basically indicates that the difference between the old and new average values is relatively large, whereas  $\mathcal{E}_2^C$  indicates that they are fairly close. These two events  $\mathcal{E}_2$  and  $\mathcal{E}_2^C$  respectively define large and small differences. In the following, we will see how these two events affect the performance of our decision policy.

Consider a naive policy  $\Pi$  such that each node  $a$  decides its power based on its own estimates obtained by running COMPARE, that is, it switches to the new power if  $x_a^{\text{new}}(K) > x_a^{\text{old}}(K)$  and keeps the old one otherwise.

*Lemma 7:* Assume  $K = K(\epsilon, \delta)$  in Algorithm COMPARE. Then, the policy  $\Pi$

$$\Pr[\bar{X}_{\Pi} \geq \max\{\bar{X}_{\text{new}}, \bar{X}_{\text{old}}\} | \mathcal{E}_1, \mathcal{E}_2] = 1,$$

where  $\bar{X}_{\Pi}$  is the average objective value of the power vector selected under the policy  $\Pi$ .

*Proof:* Given  $\mathcal{E}_1$ , (21) holds, and consequently it follows that for all  $a$ ,

$$\begin{aligned} (1 - \hat{\epsilon}) \bar{X}_{\text{new}} - (1 + \hat{\epsilon}) \bar{X}_{\text{old}} \\ \leq x_a^{\text{new}}(K) - x_a^{\text{old}}(K) \leq (1 + \hat{\epsilon}) \bar{X}_{\text{new}} - (1 - \hat{\epsilon}) \bar{X}_{\text{old}}. \end{aligned}$$

Further, given  $\mathcal{E}_2$ , we have  $\bar{X}_{\text{new}} > \frac{1+\hat{\epsilon}}{1-\hat{\epsilon}} \bar{X}_{\text{old}}$  or  $\bar{X}_{\text{new}} \leq \frac{1-\hat{\epsilon}}{1+\hat{\epsilon}} \bar{X}_{\text{old}}$ . In the first case, the above sandwich inequality implies  $x_a^{\text{new}}(K) > x_a^{\text{old}}(K), \forall a$ . Consequently, every node will select the new power under the policy  $\Pi$  so that  $\bar{X}_{\Pi} = \bar{X}_{\text{new}}$ . Note that this is the right decision because  $\bar{X}_{\text{new}} > \bar{X}_{\text{old}}$  in this case. The second case can be proved similarly.  $\square$

4. Note that  $\bar{X}_{\text{new}}$  is a random variable.

The following is a consequence of Lemma 7.

*Corollary 1:* Given  $\mathcal{E}_2^C$ , i.e., if  $\frac{1-\tilde{\epsilon}}{1+\tilde{\epsilon}}\bar{X}_{\text{old}} < \bar{X}_{\text{new}} \leq \frac{1+\tilde{\epsilon}}{1-\tilde{\epsilon}}\bar{X}_{\text{old}}$ , the policy  $\Pi$  can result in mixed decisions even after  $\epsilon$ -convergence.

Lemma 7 shows that when the difference between  $\bar{X}_{\text{new}}$  and  $\bar{X}_{\text{old}}$  is sufficiently large, the desired selection can be made easily based solely on the COMPARE procedure. On the other hand, according to the above corollary, if they are too close, the decisions can be mixed, that can possibly lead to the violation of  $C_2$ . Note also that as the accuracy of estimation increases (i.e.,  $\tilde{\epsilon}$  decreases), the region of mixed decisions diminishes.

### 5.2.2 The Case of Small Difference

So far we have seen that when the difference is large (i.e., given  $\mathcal{E}_2$ ), a unanimous decision can be easily made right after COMPARE. Moreover, any unanimous decision is kept by AGREE, and hence, Lemma 7 also hold for AGREE. This subsection studies the case where the decisions are mixed after COMPARE, in particular when the difference is small.

Recall that if  $z_a(0)$ 's are all zero or all one, then the convergence and the right decision are guaranteed immediately under the decision algorithm AGREE (See Lemma 5). Hence, assume that such a case has not happened, so that there is a mixture of nodes with  $z_a(0) = 0$  and  $z_a(0) = 1$ . Consider any  $\tilde{\epsilon} \in (0, 1/(N-1))$  and let  $\bar{\epsilon} = \tilde{\epsilon}/N$ . Then, as argued in (21), after  $\tilde{K} = K(\bar{\epsilon}, \delta)$  iterations in AGREE, every node  $a$  will obtain

$$(1 - \tilde{\epsilon})\bar{Z} \leq z_a(\tilde{K}) \leq (1 + \tilde{\epsilon})\bar{Z} \quad (22)$$

with probability at least  $1 - \delta$ . Consequently, the above inequality implies  $0 < z_a(\tilde{K}) < 1, \forall a$  since  $1/N \leq \bar{Z} \leq (N-1)/N$  and  $\tilde{\epsilon} < 1/(N-1)$ . Hence, every node will acknowledge that there are mixed decisions, and hence they will decide to use the old power.

Let  $\mathcal{E}_3$  denote this event, i.e.,  $\mathcal{E}_3$  is the event that every node obtains  $\tilde{\epsilon}$  approximation of  $\bar{Z}$  (as in (22)) under the assumption that  $\tilde{K} = K(\bar{\epsilon}, \delta)$ . Note that  $\Pr[\mathcal{E}_3] \geq 1 - \delta$ . The following lemma shows the performance of AGREE under some conditions.

*Lemma 8:* Let  $\bar{X}_{\text{agr}}$  be the average objective value achieved by AGREE. Then,

$$\Pr[\bar{X}_{\text{agr}} \geq (1 - 2\tilde{\epsilon}/(1 + \tilde{\epsilon})) \max\{\bar{X}_{\text{new}}, \bar{X}_{\text{old}}\} | \mathcal{E}_2^C, \mathcal{E}_3] = 1.$$

*Proof:* Note first that given  $\mathcal{E}_3$ , the AGREE policy will result in agreed decisions. If  $z_a(0)$ 's were all zero or one, then it follows from Lemma 5 that  $\bar{X}_{\text{agr}} = \max\{\bar{X}_{\text{new}}, \bar{X}_{\text{old}}\}$ . If this was not the case, then every node will select the old power given  $\mathcal{E}_3$ ; so that  $\bar{X}_{\text{agr}} = \bar{X}_{\text{old}}$ . Further, given  $\mathcal{E}_2^C$ , we have  $\frac{1-\tilde{\epsilon}}{1+\tilde{\epsilon}}\bar{X}_{\text{old}} < \bar{X}_{\text{new}} \leq \frac{1+\tilde{\epsilon}}{1-\tilde{\epsilon}}\bar{X}_{\text{old}}$ . Consequently, it follows that  $\bar{X}_{\text{agr}} = \bar{X}_{\text{old}} \geq \frac{1-\tilde{\epsilon}}{1+\tilde{\epsilon}} \max\{\bar{X}_{\text{new}}, \bar{X}_{\text{old}}\}$ . Therefore, the agreed decisions are made achieving at least  $1 - \frac{2\tilde{\epsilon}}{1+\tilde{\epsilon}}$  of the maximum of old and new values.  $\square$

Lemma 8 implies that when the case of small difference can be addressed by the AGREE procedure. That is, if the decisions were unanimous after COMPARE, they are right decisions and kept by AGREE. Even if the decisions are mixed, the AGREE procedure can guarantee *almost* the maximum of new and old objective values with high probability.

### 5.2.3 Combining All The Results

We now combine all the above results to complete the proof. For any trivial event  $A$  (i.e., event having zero probability measure), assume the convention  $P(\cdot|A) = 0$  where  $P(\cdot)$  is probability measure. We will use the following relationship for any events  $A, B, C$ :

$$P(A|B) = P(A|B, C)P(C|B) + P(A|B, C^C)P(C^C|B). \quad (23)$$

Recall  $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$  are respectively the events of  $\epsilon$ -convergence in COMPARE, relatively large difference between  $\bar{X}_{\text{new}}$  and  $\bar{X}_{\text{old}}$ , and  $\bar{\epsilon}$ -convergence in AGREE.

Let  $\bar{X}_{\text{max}} = \max\{\bar{X}_{\text{new}}, \bar{X}_{\text{old}}\}$ . First, note that

$$\begin{aligned} \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}}] &= \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1] \cdot \Pr[\mathcal{E}_1] \\ &\quad + \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1^C] \cdot \Pr[\mathcal{E}_1^C] \\ &\geq (1 - \delta) \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1], \end{aligned}$$

where the inequality follows from the facts that the second term is nonnegative and  $\Pr[\mathcal{E}_1] \geq 1 - \delta$ . Using the relationship (23), the last line can be rewritten as

$$\begin{aligned} &= (1 - \delta) \{ \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1, \mathcal{E}_2] \cdot \Pr[\mathcal{E}_2 | \mathcal{E}_1] \\ &\quad + \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1, \mathcal{E}_2^C] \cdot \Pr[\mathcal{E}_2^C | \mathcal{E}_1] \} \\ &\geq (1 - \delta) \{ \Pr[\mathcal{E}_2 | \mathcal{E}_1] \\ &\quad + \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1, \mathcal{E}_2^C] \cdot \Pr[\mathcal{E}_2^C | \mathcal{E}_1] \} \\ &\geq (1 - \delta) \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1, \mathcal{E}_2^C], \end{aligned}$$

The second inequality follows from Lemma 7, and the last inequality follows from the fact  $\Pr[\mathcal{E}_2 | \mathcal{E}_1] + \Pr[\mathcal{E}_2^C | \mathcal{E}_1] = 1$ . Similarly to the above (where (23) was applied and the second term was removed for proceeding the inequality), the last line can be rewritten as

$$\geq (1 - \delta) \Pr[\bar{X}_{\text{agr}} \geq (1 - \gamma_2)\bar{X}_{\text{max}} | \mathcal{E}_1, \mathcal{E}_2^C, \mathcal{E}_3] \cdot \Pr[\mathcal{E}_3 | \mathcal{E}_1, \mathcal{E}_2^C].$$

Recall  $\Pr[\mathcal{E}_3] \geq 1 - \delta$ , and this is true regardless of the initial value  $z(0)$ . The events  $\mathcal{E}_1$  and  $\mathcal{E}_2^C$  only affect the initial value, hence the conditional probability  $\Pr[\mathcal{E}_3 | \mathcal{E}_1, \mathcal{E}_2^C]$  is also no less than  $1 - \delta$ . By Assumption 1, we also have  $(1 - \gamma_2) \leq (1 - \frac{2\tilde{\epsilon}}{1+\tilde{\epsilon}})$ . The above inequality is then rewritten as

$$\geq (1 - \delta)^2 \Pr[\bar{X}_{\text{agr}} \geq (1 - \frac{2\tilde{\epsilon}}{1+\tilde{\epsilon}})\bar{X}_{\text{max}} | \mathcal{E}_1, \mathcal{E}_2^C, \mathcal{E}_3].$$

The proof is completed by noting that Lemma 8 holds even if it is additionally conditioned on  $\mathcal{E}_1$  and using  $\delta \leq \frac{\delta_2}{2}$  in Assumption 1.

### 5.3 Sign-wise Convergence

In order for a unanimous decision to be made after running COMPARE, every node  $a$  has to obtain the estimate such that  $x_a^{\text{new}}(k) > x_a^{\text{old}}(k)$  (or  $x_a^{\text{new}}(k) \leq x_a^{\text{old}}(k)$ ). Without loss of generality, we assume  $\bar{X}_{\text{new}} > \bar{X}_{\text{old}}$ . Obviously, the  $\epsilon$ -convergence is not a necessary condition for  $x_a^{\text{new}}(k) > x_a^{\text{old}}(k), \forall a$ . Therefore, the  $\epsilon$ -convergence time is a *conservative lower bound* in that a unanimous decision can be possibly made without having  $\epsilon$ -approximation of the actual average. This has led us to define a new concept of *sign-wise convergence* (or *s-convergence*).

**Definition 4:** A real-number vector  $x$  is said to be *uniform in sign* (u.i.s.) if  $x > 0$  (or  $x \leq 0$ ) component wise.

**Definition 5 (s-convergence time):** For a sequence of vectors  $\{x(k)\}$ , the *sign-wise convergence time*  $K^s(\delta)$  is defined by

$$K^s(\delta) = \inf \{k \geq 0 : \Pr [x(k) \text{ is u.i.s.}] \geq 1 - \delta\}. \quad (24)$$

Let  $x_a^{\text{dif}}(k) = x_a^{\text{new}}(k) - x_a^{\text{old}}(k)$ , then a unanimous decision can be made when the vector  $x^{\text{dif}}(k)$  is u.i.s. The following result is obvious.

**Lemma 9:** Note that once a sequence  $\{x^{\text{dif}}(k)\}$  generated by Algorithm COMPARE becomes u.i.s., it will remain u.i.s. forever.

*Proof:* Suppose that  $x^{\text{dif}}(k)$  is u.i.s, and consider any two nodes  $a, b$ . Since the vector is u.i.s., we have  $x_a^{\text{new}}(k) > x_a^{\text{old}}(k)$  and  $x_b^{\text{new}}(k) > x_b^{\text{old}}(k)$ . If they average in the next iteration, then

$$x_a^{\text{new}}(k+1) = \frac{x_a^{\text{new}}(k) + x_b^{\text{new}}(k)}{2} > \frac{x_a^{\text{old}}(k) + x_b^{\text{old}}(k)}{2} = x_a^{\text{old}}(k+1),$$

and the same is true for node  $b$ . This completes the proof.  $\square$

As a consequence, after any  $K \geq K^s(\delta_2)$  iterations in COMPARE, a decision can be made such that  $C2$  is satisfied with  $\gamma_2 = 0$ . Now, it remains to identify the value of  $K^s(\delta)$  in terms of network parameters. Let  $\bar{X}_{\text{dif}} = \bar{X}_{\text{new}} - \bar{X}_{\text{old}}$ , then by assumption  $\bar{X}_{\text{dif}} > 0$ .

**Lemma 10:** The following is a sufficient condition for the sign-wise convergence of vector  $x^{\text{dif}}(k)$ :

$$\|x^{\text{dif}}(k) - \bar{X}_{\text{dif}}\vec{1}\| < \sqrt{N}\bar{X}_{\text{dif}} \left( \frac{N}{\alpha^2} - 1 \right)^{1/2}, \quad (25)$$

where  $\alpha$  is a constant in  $(1, \sqrt{N})$ .

*Proof:* For notational convenience, drop all the indices and consider an  $N$ -dimensional vector  $x$  such that  $\sum_i x_i = N\bar{X}$ . We want to find the condition when  $x > 0$  component wise. To get some intuition, we start from the case of  $N = 2$ . In this case, the condition is easily obtained as  $x_1 x_2 > 0$  (Since we are assuming  $\bar{X}_{\text{new}} > \bar{X}_{\text{old}}$ , the case of  $x_1 \leq 0$  &  $x_2 \leq 0$  cannot happen), but this form of condition is not easy to extend to higher dimension. Consider the normalized vector  $\frac{x}{\|x\|}$ , which has exactly the same properties as  $x$  in terms of sign-wise convergence. This vector lies on the unit circle as shown in Fig. 4. Observe that any

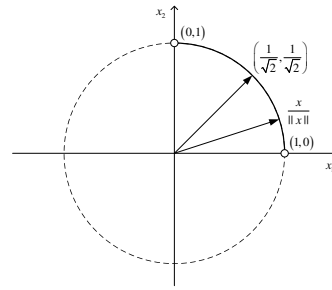


Fig. 4. Sign-wise convergence condition in 2-dimension

u.i.s. vector lies on the solid line, and the vector  $\frac{1}{\sqrt{2}}\vec{1}$  at its center. We call this *standard vector*. It is easy to see that if the inner product of standard vector and  $\frac{x}{\|x\|}$  is greater than certain value, then  $x$  is u.i.s. That value can be easily computed as  $\frac{1}{\sqrt{2}}$ , hence the condition is written as

$$\frac{1}{\sqrt{2}}\vec{1} \cdot \frac{x}{\|x\|} > \frac{1}{\sqrt{2}}. \quad (26)$$

Expanding and rearranging the above condition yields  $x_1 x_2 > 0$ . In this case, this is a necessary and sufficient condition (given that  $x_1 \leq 0$  &  $x_2 \leq 0$  cannot happen).

For higher dimension, we take similar approach. First, the standard vector  $\frac{1}{\sqrt{N}}\vec{1}$  will lie at the center of the space where all the u.i.s. vectors exist. Similarly to the 2-dimensional case, it is obvious that any u.i.s. vector can be described as

$$\frac{1}{\sqrt{N}}\vec{1} \cdot \frac{x}{\|x\|} > \frac{\alpha}{\sqrt{N}} \Rightarrow \alpha \|x\| < \sum_i x_i. \quad (27)$$

Notice that this is a natural generalization of (26), except for  $\alpha$ . For  $N = 2$ ,  $\alpha$  was 1. However, we can show that  $\alpha$  should be greater than 1 as below.

**Lemma 11 ([31]):** Assume  $N \geq 3$ . In order for the condition (27) to describe u.i.s. vectors, it should be  $1 < \alpha < \sqrt{N}$ .

Squaring both side of the inequality (27) and using the condition  $\sum_i x_i = N\bar{X}$  yields

$$\begin{aligned} \|x\|^2 &< \frac{N^2 \bar{X}^2}{\alpha^2} \\ \|x - \bar{X}\vec{1}\| &< \sqrt{N}\bar{X} \left( \frac{N}{\alpha^2} - 1 \right)^{1/2}. \end{aligned}$$

This proves the lemma since the vector  $x^{\text{dif}}(k)$  satisfies all the conditions assumed on  $x$  here.  $\square$

Since the same matching is used for averaging new and old values in COMPARE,  $x^{\text{dif}}(k)$ , defined by  $x^{\text{new}}(k) - x^{\text{old}}(k)$ , will be updated in the same way as  $x^{\text{new}}(k)$  and  $x^{\text{old}}(k)$ . Further, dividing both side of (25) by  $\|x^{\text{dif}}(0)\|$ , we obtain

$$\frac{\|x^{\text{dif}}(k) - \bar{X}_{\text{dif}}\vec{1}\|}{\|x^{\text{dif}}(0)\|} < \frac{\sqrt{N}\bar{X}_{\text{dif}}}{\|x^{\text{dif}}(0)\|} \left( \frac{N}{\alpha^2} - 1 \right)^{1/2}, \quad (28)$$

In comparison with the condition in (20), the only difference is the right hand side (RHS) of the inequality. The RHS constant  $\epsilon$  in (20) is obviously some value between 0 and 1. Moreover, the RHS value in (28)

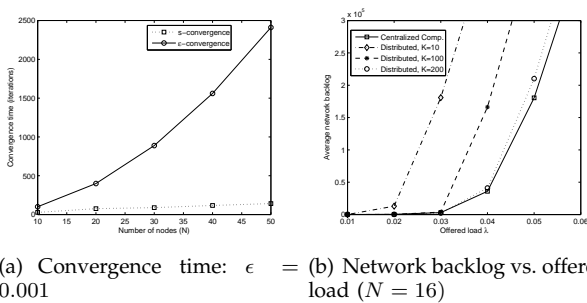


Fig. 5. Convergence time and stability performance

is positive by assumption, and in fact no less than  $(N/\alpha^2 - 1)^{1/2} < 1$ ; this is because  $\|x^{\text{dif}}(0)\|$  is convex and thus minimized when  $x_a^{\text{dif}}(k) = \bar{X}_{\text{dif}}, \forall a$  in which case the RHS becomes  $(N/\alpha^2 - 1)^{1/2} < 1$ . Hence, in the worst case,  $\epsilon$  and  $(N/\alpha^2 - 1)^{1/2}$  are about the same. This implies that the two notions of convergence have the same order of convergence speed.

Nevertheless, we expect that they will show a substantial difference in convergence time. In (20), the constant  $\epsilon$  is required to be very small because for  $C2$  to be satisfied, we need  $\epsilon = \hat{\epsilon}/N$  where  $\hat{\epsilon} \leq \gamma_2/(2 - \gamma_2)$ . In (28), the right hand side value depends on the initial condition. If  $\bar{X}_{\text{dif}}$  is large, then the sign-wise convergence can be substantially faster than the  $\epsilon$ -convergence. On the other hand, if it is very small, the sign-wise convergence time will be relatively the same as the  $\epsilon$ -convergence time. In the next section, we verify through simulations that the sign-wise convergence time is much smaller than the  $\epsilon$ -convergence time.

## 6 SIMULATION RESULTS

We generated a network topology by randomly placing  $N$  nodes in a plane. For each link  $(a, b)$ , packets arrive according to a Poisson arrival process of rate 0.5, with the mean packet size of  $2\lambda$ . The offered load is thus  $\lambda$ , and this parameter will be changed to examine the algorithm performance. Let  $d_{ab}$  be the distance between nodes  $a$  and  $b$ . The channel gain  $g_{ab}$  is fixed to  $1/(1 + d_{ab}^4)$  if  $a \neq b$ , and as assumed in Section 2,  $g_{ab} = \infty$  if  $a = b$ . The noise power and the maximum transmit power are fixed as  $n_a = 0.01$  and  $P_a^{\text{max}} = 1$  for every node  $a$ .

Figure 5(a) compares  $\epsilon$ -convergence time and sign-wise convergence time of the gossiping-based averaging in Algorithm COMPARE. The  $\epsilon$ -convergence time increases quadratically in number of nodes, whereas the  $s$ -convergence time increases linearly. Hence, the gossiping-based decision can satisfy  $C2$  with much less iterations than expected in the  $\epsilon$ -convergence analysis. Figure 5(b) plots the stability performance of distributed comparison (Algorithms RAND-PAIR, RAND-PSEL and DECIDE; denoted by distributed) and centralized comparison (Algorithms RAND-PAIR, RAND-PSEL with centralized comparison; denoted by

centralized comp.). The centralized comparison satisfies  $C2$  with  $\gamma_2 = 0$  and  $\delta_2 = 0$ , and hence by Theorem 1, it achieves nearly 100% throughput. As the number of iterations ( $K$  in Algorithm COMPARE) increases, the performance of distributed comparison approaches that of centralized comparison. This implies that our distributed power control scheme can achieve maximum throughput.

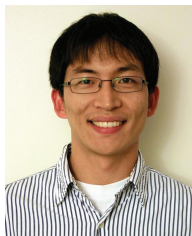
## 7 CONCLUSION

We considered the problem of achieving maximum throughput under SINR rate-based model in multi-hop wireless networks. In particular, we focused on distributed implementation of optimal power allocation algorithm. Typically, this requires repeatedly solving an optimal power allocation problem by taking into account channel conditions and queue backlog information. However, finding such a power allocation for every time slot is impractical due to not only the difficulty of the problem but also the need for distributed operation. By applying randomization approach, we characterized new throughput-optimality conditions that enable distributed implementation. We developed a randomized power allocation that satisfies the new optimality conditions, and a distributed gossip-based comparison mechanism that achieves 100% throughput, together with the randomized power allocation.

## REFERENCES

- [1] H.-W. Lee, E. Modiano, and L. B. Le, "Distributed throughput maximization in wireless networks via random power allocation," in *WiOpt*, 2009.
- [2] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Select. Areas Commun.*, vol. 24, no. 8, pp. 1452–1463, Aug. 2006.
- [3] G. Sharma, N. B. Shroff, and R. R. Mazumdar, "On the complexity of scheduling in wireless networks," in *ACM MOBICOM*, Los Angeles, CA, Sept. 2006.
- [4] E. Modiano, D. Shah, and G. Zussman, "Maximizing throughput in wireless networks via gossiping," in *ACM SIGMETRICS/Performance*, Saint-Malo, France, June 2006.
- [5] A. Eryilmaz, A. Ozdaglar, and E. Modiano, "Polynomial complexity algorithms for full utilization of multi-hop wireless networks," in *IEEE INFOCOM*, Anchorage, AK, May 2007.
- [6] S. Sanghavi, L. Bui, and R. Srikant, "Distributed link scheduling with constant overhead," in *ACM SIGMETRICS*, June 2007.
- [7] A. Gupta, X. Lin, and R. Srikant, "Low-complexity distributed scheduling algorithms for wireless networks," in *IEEE INFOCOM*, Anchorage, AK, May 2007.
- [8] R. Cruz and A. Santhanam, "Optimal routing, link scheduling and power control in multihop wireless networks," in *IEEE INFOCOM*, San Francisco, CA, June 2006.
- [9] T. ElBatt and A. Ephremides, "Joint scheduling and power control for wireless ad hoc networks," *IEEE Trans. Wireless Commun.*, vol. 3, no. 1, pp. 74–85, Jan. 2004.
- [10] M. Chiang, "Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control," *IEEE J. Select. Areas Commun.*, vol. 23, no. 1, pp. 104–116, Jan. 2005.
- [11] B. Radunovic and J. Le Boudec, "Optimal power control, scheduling, and routing in UWB networks," *IEEE J. Select. Areas Commun.*, vol. 22, no. 7, pp. 1252–1270, Sep. 2004.
- [12] S. A. Borbash and A. Ephremides, "Wireless link scheduling with power control and sinr constraints," *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 5106–5111, Nov. 2006.

- [13] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Automat. Contr.*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.
- [14] M. J. Neely, E. Modiano, and C. E. Rohrs, "Power allocation and routing in multibeam satellites with time-varying channels," *IEEE/ACM Trans. Networking*, vol. 11, no. 1, pp. 138–152, 2003.
- [15] M. Neely, E. Modiano, and C. Rohrs, "Dynamic power allocation and routing for time varying wireless networks," *IEEE J. Select. Areas Commun.*, vol. 23, no. 1, pp. 89–103, Jan. 2005.
- [16] M. Neely and E. Modiano, "Fairness and optimal stochastic control for heterogeneous networks," *IEEE/ACM Trans. Networking*, vol. 16, no. 2, pp. 396–409, Apr. 2008.
- [17] X. Lin and N. Shroff, "The impact of imperfect scheduling on cross-layer congestion control in wireless networks," *IEEE/ACM Trans. Networking*, vol. 14, no. 2, pp. 302–315, 2006.
- [18] A. Eryilmaz and R. Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," *IEEE/ACM Trans. Networking*, vol. 15, no. 6, pp. 1333–1344, Dec. 2007.
- [19] P. Charporkar, K. Kar, and S. Sarkar, "Throughput guarantees through maximal scheduling in wireless networks," *IEEE Trans. Inform. Theory*, vol. 54, no. 2, pp. 572–594, Feb. 2008.
- [20] G. Zussman, A. Brzezinski, and E. Modiano, "Multihop local pooling for distributed throughput maximization in wireless networks," in *IEEE INFOCOM*, Phoenix, AZ, April 2008.
- [21] C. Joo, X. Lin, and N. B. Shroff, "Understanding the capacity region of the greedy maximal scheduling algorithm in multihop wireless networks," in *IEEE INFOCOM*, April 2008.
- [22] G. Sharma, N. Shroff, and R. Mazumdar, "Joint congestion control and distributed scheduling for throughput guarantees in wireless networks," in *IEEE INFOCOM*, May 2007.
- [23] S. Hayashi and Z.-Q. Luo, "Spectrum management for interference-limited multi-user communication systems," *IEEE Trans. Inform. Theory*, vol. 55, no. 3, pp. 1153–1175, Mar. 2009.
- [24] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE J. Select. Topics Signal Processing*, vol. 2, no. 1, pp. 57–73, Feb. 2008.
- [25] L. Tassiulas, "Linear complexity algorithms for maximum throughput in radio networks and input queued switches," in *IEEE INFOCOM*, San Francisco, CA, March 1998.
- [26] Y. Yi, G. de Veciana, and S. Shakkottai, "On optimal mac scheduling with physical interference," in *IEEE INFOCOM*, Anchorage, AK, May 2007.
- [27] T. Moscibroda, Y. Oswald, and R. Wattenhofer, "How optimal are wireless scheduling protocols?" in *IEEE INFOCOM*, 2007.
- [28] O. Goussevskaia, R. Wattenhofer, M. Halldorsson, and E. Welzl, "Capacity of arbitrary wireless networks," in *INFOCOM*, 2009.
- [29] M. Andrews and M. Dinitz, "Maximizing capacity in arbitrary wireless networks in the sinr model: Complexity and game theory," in *IEEE INFOCOM*, 2009.
- [30] J. Kim, X. Lin, and N. Shroff, "Locally optimized scheduling and power control algorithms for multi-hop wireless networks under sinr interference models," in *WiOpt*, 2007.
- [31] H.-W. Lee, E. Modiano, and L. B. Le. (2009) Distributed throughput maximization in wireless networks via random power allocation. [Online]. Available: [http://web.mit.edu/hwlee/www/papers/TR1\\_randpow.pdf](http://web.mit.edu/hwlee/www/papers/TR1_randpow.pdf)
- [32] S. Boyd and et al., "Randomized gossip algorithms," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2508–2530, June 2006.
- [33] D. Shah, D. N. C. Tse, and J. N. Tsitsiklis, "Hardness of low delay network scheduling," *Submitted to IEEE Trans. Info. Theory*.
- [34] R. Olfati-Saber, J. A. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.
- [35] A. Olshevsky and J. N. Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 33–55, 2009.
- [36] H. J. Landau and A. M. Odlyzko, "Bounds for eigenvalues of certain stochastic matrices," *Linear Algebra and Its Applications*, vol. 38, pp. 5–15, 1981.
- [37] A. Dimakis, A. Sarwate, and M. Wainwright, "Geographic gossip: Efficient averaging for sensor networks," *IEEE Trans. Signal Processing*, vol. 56, no. 3, pp. 1205–1216, Mar. 2008.
- [38] F. Benezit, A. Dimakis, P. Thiran, and M. Vetterli, "Order-optimal consensus through randomized path averaging," in *Allerton*, 2007.



**Hyang-Won Lee** (S'02-M'08) received his B.S., M.S. and Ph.D. degrees all in Electrical Engineering and Computer Science from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2001, 2003 and 2007, respectively. He is currently a Postdoctoral Research Associate at the Massachusetts Institute of Technology, Cambridge, MA. His research interests are in the areas of congestion control, wireless resource allocation and network survivability.



**Eytan Modiano** (S'90-M'93-SM'00) received his B.S. degree in Electrical Engineering and Computer Science from the University of Connecticut at Storrs in 1986 and his M.S. and PhD degrees, both in Electrical Engineering, from the University of Maryland, College Park, MD, in 1989 and 1992 respectively. He was a Naval Research Laboratory Fellow between 1987 and 1992 and a National Research Council Post Doctoral Fellow during 1992-1993. Between 1993 and 1999

he was with MIT Lincoln Laboratory where he was the project leader for MIT Lincoln Laboratory's Next Generation Internet (NGI) project. Since 1999 he has been on the faculty at MIT; where he is presently an Associate Professor. His research is on communication networks and protocols with emphasis on satellite, wireless, and optical networks.

He is currently an Associate Editor for The IEEE/ACM Transactions on Networking and for The International Journal of Satellite Communications and Networks. He had served as Associate Editor for The IEEE Transactions on Information Theory (communication networks area) and as a guest editor for IEEE JSAC special issue on WDM network architectures; the Computer Networks Journal special issue on Broadband Internet Access; the Journal of Communications and Networks special issue on Wireless Ad-Hoc Networks; and for IEEE Journal of Lightwave Technology special issue on Optical Networks. He was the Technical Program co-chair for Wiopt 2006, IEEE Infocom 2007, and ACM MobiHoc 2007. He is an Associate Fellow of the AIAA.



**Long Bao Le** (S'04-M'07) received the B.Eng. degree with highest distinction from Ho Chi Minh City University of Technology in 1999, the M.Eng. degree from Asian Institute of Technology (AIT) in 2002 and the PhD degree from University of Manitoba in 2007. He is currently an assistant professor at INRS-EMT, University of Quebec, Montreal, Quebec, Canada. Before that, he was a postdoctoral researcher first at University of Waterloo, Canada then at Massachusetts

Institute of Technology, USA. His current research interests include cognitive radio, dynamic spectrum sharing, link and transport layer protocol issues, cooperative diversity and relay networks, stochastic control and cross-layer design for communication networks. He is a Member of the IEEE