Throughput and Energy Efficiency in Wireless Ad Hoc Networks With Gaussian Channels

Hanan Shpungin, *Member, IEEE*, and Zongpeng Li

***Abstract*—This paper studies the bottleneck link capacity under the Gaussian channel model in strongly connected random wireless ad hoc networks, with n nodes independently and uniformly distributed in a unit square. We assume that each node is equipped with two transceivers (one for transmission and one for reception) and allow all nodes to transmit *simultaneously*. We draw lower and upper bounds, in terms of bottleneck link capacity, for homogeneous networks (all nodes have the same transmission power level) and propose an energy-efficient power assignment algorithm (CBPA) for heterogeneous networks (nodes may have different power levels), with a provable bottleneck link capacity guarantee of $\Omega(B\log(1+1/\sqrt{n}\log^2 n))$, where B is the channel bandwidth. In addition, we develop a distributed implementation of CBPA with $O(n^2)$ message complexity and provide extensive simulation results.**

***Index Terms*—Algorithm design and analysis, approximation algorithms, channel capacity, distributed algorithms, network topology, wireless communication.**

I. Introduction

THE TECHNOLOGICAL and theoretical advances in the study of wireless communications lead to a rapid introduction of wireless ad hoc networks to a wide spectrum of applications, from scientific monitoring to military and rescue operations. The temporary physical topology of the network is determined by the distribution of the wireless devices (nodes), as well as the transmission range of each node. The ranges determine a directed communication graph, in which nodes correspond to the transceivers and edges correspond to the communication links. Thus, the assignment of transmission powers and the relative node disposition constitute the communication backbone of the network.

A central issue in wireless networks is that of *network capacity* (or throughput). Roughly speaking, network capacity defines the amount of data that can be transported during a given period of time. One of the main factors affecting the network capacity is radio interference caused by simultaneous transmissions. In fact, it has been shown [1] that the capacity of a network with n nodes is $\Theta(B\sqrt{n})$ bits per second (b/s), where B is the bandwidth of the communication channel. This bound holds even when the nodes are optimally placed in a disk of unit area, the transmissions are optimally assigned, and traffic patterns are optimally chosen. Thus, if the capacity is equally divided between the nodes, the per-node throughput scales as $\Theta(B/\sqrt{n})$, i.e., it decreases at the rate of $1/\sqrt{n}$ as the number of nodes, n, increases. This surprising result emphasizes the destructive impact radio interference has on the network capacity.

While there has been a sizable amount of work in the literature on network capacity, almost all previous works [1]–[9] have assumed some form of scheduling is allowed. That is, nodes are assumed to use synchronized channel access methods such as the TDMA protocol; the time is divided into time slots, and each communication link is scheduled during one of the time slots in such a way that the mutual interference of simultaneously transmitting nodes is bounded. These schemes draw theoretical upper bounds on the achievable throughput rates in perfect conditions as they typically rely on global knowledge of the communication links.

In contrast to previous work, in this paper we explore the *worst-case scenario*—that of when all the communication links are scheduled *simultaneously* (the scenario of simultaneous transmissions is discussed in detail in Section II-A). We adopt the *Gaussian channel* model, which determines the link capacity as a continuous function of signal-to-interference-plus-noise ratio (SINR) on the receiver's side. The Gaussian channel better characterizes the physical layer of wireless networks than the more common, *threshold-based* models [1]. A formal definition of the Gaussian channel model appears in Section II-A.

The capacity bounds in this setting are of fundamental interest, both from a *theoretical* and *practical* perspective. Such bounds represent the capacity achievable in the worst-case scenario when node transmissions take place in a distributed fashion without a centralized scheduler. More precisely, the achievable capacity in this setting represents a *lower bound* on the achievable capacity in scenarios where *local collision avoidance* protocols are used. Examples of such protocols include the widely used IEEE 802.11 specified CSMA/CA protocol, as well as TDMA-like protocols with distributed local slot-selection mechanism, such as TRAMA [10] or LMAC [11]. Additionally, we believe that these bounds can shed light on the quantitative and qualitative improvement of capacity possible under scheduling. This is an important question, as scheduling adds complexity to the operation of a wireless network, and the tradeoff between this complexity and the improved capacity is of considerable interest to the network designer. Finally, the bounds derived in this paper serve as a guarantee on the minimum throughput achievable when centralized scheduling in a network is infeasible.

This paper also addresses the critical issue of energy efficiency in wireless networks. Unlike nodes in wired networks,

Manuscript received May 28, 2010; revised January 17, 2011 and March 18, 2011; accepted April 26, 2011; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor A. Capone. Date of publication June 27, 2011; date of current version February 15, 2012. A preliminary version of this paper appeared at the IEEE Communications Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks (SECON), Boston, MA, June 27–30, 2010.

The authors are with the Department of Computer Science, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: hanan.shpungin@ucalgary.ca; zongpeng@ucalgary.ca).

Digital Object Identifier 10.1109/TNET.2011.2158237

1063-6692/$26.00 © 2011 IEEE

wireless devices are typically equipped with limited energy supplies, which makes energy efficiency one of the primary objectives in wireless network design [12], especially when battery replacement is infeasible. We evaluate energy efficiency through two measures: total energy consumption and network lifetime, which is the time until the first battery charge depletion. Note that there is a strong correlation between the two measures.

Our main contribution in this paper is in the study of bottleneck link capacity in *strongly connected*[1] random wireless networks. We provide theoretical bounds for the the minimum bottleneck link capacity among all pairs of nodes. We consider a network where n wireless nodes are randomly and independently deployed in a unit square under uniform distribution, and each is assigned a certain transmission power such that the induced communication graph is strongly connected. Our results can be summarized as follows.

- *Homogeneous networks:* All nodes are assigned the same transmission power. We study the lower and upper bounds of the bottleneck link capacity and show that any pair of nodes can achieve a bottleneck capacity of at least $\Omega\left(B\log\left(1+\frac{1}{n\phi(n)\log^2 n}\right)\right)$, and that there are at least $2n-4$ node pairs that cannot achieve a better bottleneck capacity than $O\left(B\log\left(1+\frac{\phi^3(n)}{n}\right)\right)$, where B is the channel bandwidth and $\phi(n)$ is any function with $\lim_{n\to\infty}\phi(n) = \infty$. We also demonstrate that for a nonrandom node distribution, the bottleneck capacity can be arbitrarily small.
- *Heterogeneous networks:* The nodes can be assigned different power levels. We propose an energy-efficient power assignment algorithm (CBPA) such that for any pair of nodes there is a path in the induced communication graph with a bottleneck capacity of $\Omega\left(B\log\left(1+\frac{1}{\sqrt{n}\log^2 n}\right)\right)$, which is an improvement over the upper bound of homogeneous networks. We also show that the total energy consumption and network lifetime of the power assignment is within a constant factor, in both measures, from the best possible for *any* strongly connected topology.
- We develop an asynchronous distributed implementation of CBPA with $O(n^2)$ message complexity.
- We perform extensive simulations of the CBPA algorithm and compare its performance to several power assignments that induce strongly connected communication graphs. Our measurements show that CBPA outperforms other algorithms in terms of minimum link capacity and is very energy-efficient.

To the best of our knowledge, these are the first provable bounds for link capacity in wireless networks, when nodes are allowed to transmit simultaneously.

The rest of the paper is organized as follows. In Section II, we outline our wireless network model, formulate the problem, and remark on our probabilistic model. Then, in Section III, we introduce some definitions and statements that we use throughout the paper. Sections IV and V study the homogeneous and heterogeneous cases, respectively. This is followed by Section VI, which provides the distributed implementation of CBPA. Numerical results from our simulations appear in Section VII, and related work is presented in Section VIII. Finally, we conclude and discuss possible future directions in Section IX.

II. System Settings and Problem Formulation

In this section, we first describe our wireless network model, then formulate the problem, and finally remark on the probabilistic nature of our model.

A. Wireless Network Model

Let V be the n wireless nodes randomly, uniformly and independently distributed in a unit square. It is customary to assume that the power required to transmit to distance d is proportional to d^κ, where κ is the *distance-power gradient*, i.e., the signal strength of a transmission should exceed a certain threshold. In perfect conditions $\kappa = 2$, however, in more realistic settings (in presence of obstructions) it can have a value between 2 and 4 (see [14]). In this paper, we assume $\kappa = 2$ for simplicity, although our results could be easily generalized for any $\kappa \geq 2$.

To adhere to the scheme of simultaneous transmissions, we assume a full-duplex wireless link model, where a node can transmit and receive simultaneously at the same time, over the same wireless channel. We assume that each node is equipped with separate antennas for transmitting and receiving purposes. Since the node knows its own signal of transmission, it is possible to subtract the transmitted signal from the received signal, which is the combination of the intended signal from a neighbor interleaved with its own transmission.

We wish to point out, however, such a full-duplex link model, although conceivable and sound in theory, has challenges for immediate real-world implementations, given current hardware technologies in wireless radio antennas. The main obstacle lies in the fact that the desired signal from a neighbor node is much weaker (up to 60 dB) than the local transmission. New engineering design in either the analog domain (for circuit noise cancellation) or the digital domain (for digital noise cancellation after ADC sampling) is necessary [15]. A recent work of Choi *et al.* [16] studies a new way of implementing a full-duplex wireless link with off-the-shelf hardware only. Their implementation employs two transmission antennas and one receiving antenna per node. The basic idea, referred to as "antenna cancellation," is to strategically place the three antennas, so that the two transmission antennas have their signals (almost) canceled at the receiving antenna. Experiment results demonstrate 84% throughput gains over half-duplex transmissions.

A power assignment is a function $p: V \to \mathbb{R}^+$; the transmission possibilities that result from a power assignment p induce a directed communication graph $H_p = (V, E_p)$, where

$$E_p = \{(u,v) : p(u) \geq d(u,v)^\kappa\}$$

is a set of directed edges. The graph H_p is strongly connected iff for every pair of nodes $u, v \in V$ there exists a directed path from u to v in H_p. The total energy consumption, also referred to as the cost, of the power assignment is given by $c(p) = \sum_{u\in V} p(u)$.

Each node u has some initial battery charge $b(u)$, which is sufficient for a limited amount of time, proportional to the power assignment $p(u)$. It is common to take the lifetime of a wireless node v to be $l(u) = b(u)/p(u)$. The network lifetime is defined as the time it takes the first node to run out of its battery charge.

[1] A graph is strongly connected if there is a path between any pair of nodes. Many applications in civilian, industrial, and military areas require a strongly connected underlying topology to carry out different networking tasks [13].

For a power assignment p and initial battery charges b, the network lifetime is defined as $l(p) = \min_{u \in V} l(u)$. In this paper, we assume unit initial battery charges $b \equiv 1$, that is $b(u) = 1$, for every $u \in V$.

According to the Gaussian channel model, node v can successfully receive a transmission from u over a wireless communication link (u, v) at a data rate

$$\mathrm{Cap}(u, v) = B \log(1 + \mathrm{SINR}(u, v))$$

bits per second, where B is the channel bandwidth, and

$$\mathrm{SINR}(u, v) = \frac{p(u)/d(u, v)^{\kappa}}{N_0 + \sum\limits_{w \in V \setminus \{u,v\}} p(w)/d(w, v)^{\kappa}},$$

with N_0 being the ambient noise power. That is, the receiver achieves the Shannon's capacity for a wireless channel with additive Gaussian noise [17]. A closer look at the expression $\mathrm{SINR}(u, v)$ reveals that it consists of two parts: the signal strength in the numerator, and the interference in the denominator. In our scenario, the ambient noise, N_0, is negligible compared to the signal strength of the link and the interference caused by other transmitting nodes, so we set $N_0 = 0$ throughout the rest of the paper. In practice, such noise will have to be considered in addition to interference caused by other nodes. Let the interference from all the other nodes at v (except for u) be

$$I(u, v) = \sum_{w \in V \setminus \{u,v\}} p(w)/d(w, v)^{\kappa}.$$

As the expression for $\mathrm{SINR}(u, v)$ is the only variable that affects the capacity of the link (u, v), most of the paper is dedicated to its analysis, and in particular to analyzing $I(u, v)$.

B. Problem Definition

The capacity of a path P in a communication graph H is defined as the capacity of the minimum capacity link in P, that is $\mathrm{Cap}(P) = \min_{(x,y) \in P} \mathrm{Cap}(x, y)$. For a pair of nodes $u, v \in V$, we define the feasible throughput between u and v in a H as $\mathrm{Cap}(H, u, v) = \max\{\mathrm{Cap}(P), P \text{ is a path from } u \text{ to } v \text{ in } H\}$. Finally, the capacity of a communication graph H is defined as the feasible throughput between a pair of nodes with the minimum feasible throughput

$$\mathrm{Cap}(H) = \min_{u,v \in V} \mathrm{Cap}(H, u, v).$$

Formally, this paper addresses the following problem.

Problem 1:

Input: A set of wireless nodes V, randomly, uniformly, and independently distributed in a unit square.

Output: A power assignment p, such that H_p is strongly connected.

Objective: Maximize $\mathrm{Cap}(H_p)$.

In addition to the primary optimization objective of capacity maximization we also look to make the power assignments as energy-efficient as possible, in terms of both, total energy consumption and network lifetime.

C. Probabilistic Nature of Results

This paper considers a random wireless network where n nodes are randomly and independently placed in unit square under the uniform distribution. As a result, the majority of statements in this paper bears a probabilistic nature, i.e., the probability of a statement converges to 1 as the number of network nodes, n, increases. In common literature, this is denoted as *with high probability* (a.k.a., w.h.p.) or *almost always* (a.k.a., a.a.). For simplicity of notation, we avoid the use of either *w.h.p.* or *a.a.* since it is clear from the context.

III. Preliminaries

In this section, we present some definitions and statements that are used throughout the rest of the paper.

Let $G_V = (V, E_V)$ be a complete directed graph, where V is a set of the wireless nodes. We define the weight function, $w : E_V \to \mathbb{R}^+$, on the edge set E_V as $w(u, v) = d(u, v)^2$, where $d(u, v)$ is the Euclidean distance between u and v. Note that the weight of an edge (u, v) matches the amount of energy that is required to transmit from u to v. Let MST_V be the minimum weight spanning tree of the undirected version of G_V (which is obtained easily by omitting the edge directions). Let e^* be the maximum length edge in MST_V. For every subgraph H of G_V, let $E(H)$ be the edge set of H. The weight of H is given by $w(H) = \sum_{e \in E(H)} w(e)$. For any edge $e = (u, v)$, its length is denoted by $|e| = d(u, v)$.

Let p^* be the minimum cost power assignment so that H_{p^*} is strongly connected. Chen and Huang [18], and later Kirousis *et al.* [19] made the following statement, which is a common folklore in the study of wireless networks.

Theorem 2.1 [18]: $c(p^*) \geq w(\mathrm{MST}_V)$.

Let p^{**} be the maximum lifetime power assignment so that $H_{p^{**}}$ is strongly connected. In [20], the authors showed the following lemma for the case that all the initial battery charges are equal ($\forall\ u \in V, b(u) = 1$).

Theorem 2.2 [20]: $l(p^{**}) \leq \frac{1}{w(e^*)}$.

We make use of several relevant theoretical results, which apply to the random distribution. The following corollary was derived in [21].

Corollary 2.3 [21]: $w(e^*) = \Theta\left(\frac{\log n}{n}\right)$.

Zhang and Hou in [22] derived a lower bound on the cost of a power assignment required to induce a k-fault resistant strongly connected communication graph ($k = 1$ in our case) under the assumption that the nodes form a homogeneous Poisson point process with density λ. According to [23], a random, uniform, and independent n-point process in a unit square is essentially a Poisson process with $\lambda = n$, for large values of n. In the next theorem, we bring the main result of [22] adapted to the case of $k = 1$ and our point process.

Theorem 2.4 [22]: $c(p^*) = \Omega(1)$.

For any $u \in V$, let $d_1(u) = \min_{v \in V \setminus \{u\} d(u,v)}$ be the distance from u to its nearest neighbor. Berend *et al.* [20] bounded the distance to the kth nearest neighbor from any node. The theorem below is adapted to the case of $k = 1$.

Theorem 2.5 [20]: For every node $v \in V$

$$\sqrt{\frac{1}{2\pi(n-1)n\phi(n)}} \leq d_1(v) \leq 2\sqrt{\frac{2\log n}{\pi(n-1)}}$$

where $\phi(n)$ is any function with $\lim_{n\to\infty} \phi(n) = \infty$.

Note that $\phi(n)$ can be any function as long as $\lim_{n\to\infty} \phi(n) = \infty$, e.g., $\log n$, $\sqrt{\log n}$, or $\log^* n$. The choice

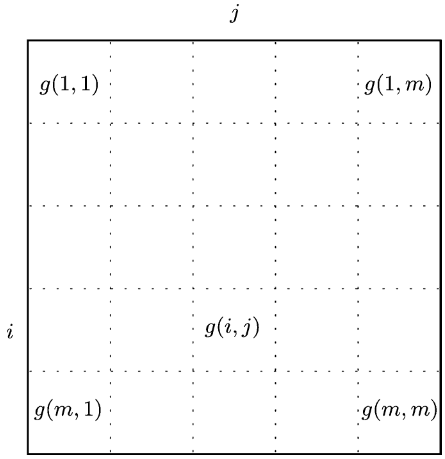


Fig. 1. Grid cells in a unit square $\left(m=\sqrt{\frac{n}{16\log n}}\right)$.

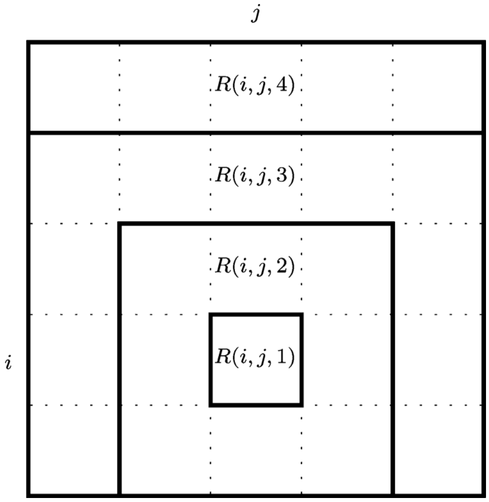


Fig. 2. Rings around grid cell $g(i,j)$.

of $\phi(n)$ affects only the convergence rate of the probability of the statement.

We divide the unit square into $\frac{n}{16\log n}$ grid cells,[2] each of size $\sqrt{\frac{16\log n}{n}}\times\sqrt{\frac{16\log n}{n}}$. Let $g(1,1)$ and $g\left(\sqrt{\frac{n}{16\log n}},\sqrt{\frac{n}{16\log n}}\right)$ be the leftmost top and rightmost bottom cells, respectively. The rest of the cells are indexed as depicted in Fig. 1. Let $N(i,j)$ be the set of nodes in a grid cell $g(i,j)$, $1\le i,j\le\sqrt{\frac{n}{16\log n}}$. The next lemma analyzes the number of nodes in each cell.

Lemma 2.6: There are $\Theta(\log n)$ nodes in each cell.

Proof: The process of random, uniform, and independent node placement in a unit square can be viewed as Bernoulli trials with respect to grid cell $g(1,1)$. Let $X_1,X_2,\ldots,X_n$ be independent Bernoulli trials, where $X_i=1$ if the ith node is placed inside $g(1,1)$, and $X_i=0$ otherwise. Clearly, $\Pr[X_i=1]=\frac{16\log n}{n}$ for every i, $1\le i\le n$. Let $X=\sum_{i=1}^n X_i$. We make use of the well-known Chernoff bounds to bound X. The expected value of X is given by $E[X]=16\log n$. First, we compute the lower bound of X. For any $\delta\in(0,1]$

$$\Pr[X<(1-\delta)E[X]]<\exp(-E[X]\delta^2/2).$$

By setting $\delta=1/2$, we obtain $\Pr[X<8\log n]<1/n^2$. The upper bound is computed in a similar manner. For any $\delta>0$

$$\Pr[X>(1+\delta)E[X]]<\left(\frac{e^\delta}{(1+\delta)^{\delta+1}}\right)^{E[X]}.$$

By setting $\delta=e-1$ we obtain $\Pr[X>e\cdot E[X]]<\exp(-16\log n)=1/n^{16}$. As the total number of cells is $\frac{n}{16\log n}$, by applying the union bound, we conclude that there are $\Theta(\log n)$ nodes in each cell. ■

Later in this paper, we use the notion of rings of cells, which are defined "around" a certain grid cell $g(i,j)$. The nodes that reside in the kth ring, $k\ge 1$, around $g(i,j)$ are defined as

$$R(i,j,k)=\{v:v\in N(l,m),\max\{|i-l|,|j-m|\}=k-1\}.$$

It is possible that some rings have cells that "fall out" of the unit square. To simplify the notation, we define $N(i,j)=\emptyset$ if $\min\{i,j\}\le 0$ or $\max\{i,j\}>\sqrt{\frac{n}{16\log n}}$. An example of the rings is given in Fig. 2 (note that, starting from the third ring, some of the cells are "outside" the unit square).

IV. Homogeneous Power Assignment

In this section, we consider the case when all the nodes are assigned the same power level. First, we prove lower and upper bounds on the throughput in any strongly connected topology. Then, we shortly discuss how to compute an optimum routing that maximizes the throughput between any pair of nodes. Finally, we demonstrate that for a nonrandom node distribution, the network capacity can be arbitrarily small.

A. Throughput Bounds

Suppose that the nodes are assigned a transmission power level of γ, i.e., $p(u)=\gamma$, for every $u\in V$. Then

$$\mathrm{SINR}(u,v)=\frac{\gamma\cdot d(u,v)^{-2}}{\gamma D(u,v)}$$

where $(u,v)\in E_p$ and $D(u,v)=\sum_{w\in V\setminus\{u,v\}}d(w,v)^{-2}$. This results in the following observation.

Observation 3.1: If p is a homogeneous power assignment, then $\mathrm{SINR}(u,v)=\Theta(d(u,v)^{-2}/D(u,v))$, for every $(u,v)\in E_p$.

In our analysis, we focus on the SINR in the worst possible case. The following two lemmas derive bounds for $D(u,v)$. Note that Lemma 3.2 (which derives a lower bound on $D(u,v)$) does not have a direct effect on the capacity upper bound analysis in Theorem 3.5. However, it contributes to the understanding of minimum interference in a wireless environment as we remark toward the end of this section.

Lemma 3.2: For any $u,v\in V$, $D(u,v)=\Omega(n\log n)$.

Proof: Denote $D(v)=\sum_{w\in V\setminus\{v\}}d(w,v)^{-2}$. We first show that $D(v)=\Omega(n\log n)$, and then conclude the same for $D(u,v)$. Recall the grid defined in Section III. Suppose that $v\in N(1,1)$. We divide the nodes into $\sqrt{\frac{n}{16\log n}}$, rings around $g(1,1)$. For simplicity, let $R_k=R(1,1,k)$,

[2] For convenience, we omit the use of floors and ceilings, which does not affect our analysis.

$k \in \left\{1,\ldots,\sqrt{\frac{n}{16\log n}}\right\}$. Clearly, for any $w \in R_k$, $d(w,v) \le k\sqrt{32\log n/n}$. Note that in the kth ring, R_k, there are $2k-1$ grid cells. Combining with Lemma 2.6, we can conclude that the number of nodes in R_k is at least $\Omega(k\log n)$. In fact, from the proof of Lemma 2.6, it follows $|R_k| \ge 8k\log n$. Therefore

$$\begin{aligned} D(v) &= \sum_{k=1}^{\sqrt{\frac{n}{16\log n}}} \sum_{w\in R_k} d(w,v)^{-2} \\ &\ge \sum_{k=1}^{\sqrt{\frac{n}{16\log n}}} \frac{|R_k|n}{32k^2\log n} \ge n\sum_{k=1}^{\sqrt{\frac{n}{16\log n}}} \frac{1}{4k} \\ &= \frac{1}{4}nH_{\sqrt{\frac{n}{16\log n}}} = \Omega(n\log n) \end{aligned}$$

where H_m is the mth harmonic number. In our analysis, we assumed that every node $w \in R_k$ is at a maximum possible distance from v. As a result, for any $u \in V$, $D(u,v) \ge \Omega(n\log n) - \frac{n}{32\log n} = \Omega(n\log n)$. It is easy to verify that the lower bound computed for $v \in N(1,1)$ holds for any node as well, as the worst case is obtained when v is in one of the corners of the unit square and all the other nodes are positioned in the most distant corner of their grid cell. ■

Lemma 3.3: For any $u, v \in V$, $D(u,v) = O(n^2\phi(n)\log n)$, where $\phi(n)$ is any function with $\lim_{n\to\infty}\phi(n) = \infty$.

Proof: The proof resembles the one of Lemma 3.2. Again, let $D(v) = \sum_{w\in V\setminus\{v\}} d(w,v)^{-2}$. Note that for any pair of nodes, u, $v \in V$, $D(u,v) \le D(v)$. This time we compute the upper bound for a node in the center grid cell, that is $v \in N\left(\sqrt{\frac{n}{64\log n}}, \sqrt{\frac{n}{64\log n}}\right)$. We divide the nodes into $\sqrt{\frac{n}{64\log n}}$ rings around $g\left(\sqrt{\frac{n}{64\log n}}, \sqrt{\frac{n}{64\log n}}\right)$. For simplicity, let $R_k = R\left(\sqrt{\frac{n}{64\log n}}, \sqrt{\frac{n}{64\log n}}, k\right)$, $k \in \left\{1,\ldots,\sqrt{\frac{n}{64\log n}}\right\}$.

Due to Theorem 2.5, for any $w \in V$, $d(w,v) \ge \frac{1}{n\phi(n)}$, where $\phi(n)$ is any function with $\lim_{n\to\infty}\phi(n) = \infty$. In particular, this holds for nodes in $R_1 \cup R_2$. For any $w \in R_k$, $k > 2$, it holds $d(w,v) \ge (k-2)\sqrt{\frac{16\log n}{n}}$. Note that there are nine grid cells covered by R_1 and R_2, and at most $8(k-1)$ grid cells in R_k, $k > 2$. Combining with Lemma 2.6, there are $\Theta(k\log n)$ nodes in R_k. As a result

$$\begin{aligned} D(v) &= \sum_{k=1}^{\sqrt{\frac{n}{64\log n}}} \sum_{w\in R_k} d(w,v)^{-2} \\ &= \sum_{w\in R_1\cup R_2} d(w,v)^{-2} + \sum_{k=3}^{\sqrt{\frac{n}{64\log n}}} \sum_{w\in R_k} d(w,v)^{-2} \\ &\le O(n^2\phi(n)\log n) + O(n\log n). \end{aligned}$$

The bound for $\sum_{k=3}^{\sqrt{\frac{n}{64\log n}}} \sum_{w\in R_k} d(w,v)^{-2}$ is obtained similarly to the proof of Lemma 3.2. Note that for every $w \ne v$, both v and w were considered to be on the boundaries of their grid cells, which is the worst case possible. By taking v to be in the center grid cell, we minimized the pairwise distances between cell boundaries. Thus, we can conclude that the upper bound holds for all the nodes, in every grid cell. ■

Before we proceed to present the main result of this section, we derive the following technical lemma that shows the existence of an "isolated" pair of nodes. A pair of nodes u, $v \in V$ is (ε,δ)-*isolated*, $\varepsilon \le \delta$, if $d(u,v) \le \varepsilon$ and for every $w \in V \setminus \{u,v\}$, $\min\{d(u,w), d(v,w)\} \ge \delta$.

Lemma 3.4: There exists at least one (ε,δ)-isolated pair, where $\varepsilon = \frac{\phi(n)}{n}$, $\delta = \frac{1}{\sqrt{n\phi(n)}}$, and $\phi(n)$ is any function with $\lim_{n\to\infty}\phi(n) = \infty$ such that $\varepsilon \le \delta$.

Proof: Our proof resembles the technique used in [24]. For convenience, we will omit the notation of (ε,δ) and simply say that a pair is isolated. For any $u \in V$, let A_u be an event that u belongs to an isolated pair. Let X_u be an indicator random variable such that $X_u = 1$ if A_u occurs, and $X_u = 0$ otherwise. Let $X = \sum_{u\in V} X_u$. We use the second moment method [25] to show that $X > 0$, which proves the lemma.

First, we derive an upper bound on the probability of A_u to occur. A node $u \in V$ is a part of an isolated pair iff there is another node within distance ε and all the other nodes are at a distance of at least δ from both nodes. There are $n-1$ possibilities to choose a node v, which together with u forms an isolated pair and therefore the probability that there is a close node to u is at most $(n-1)\pi\varepsilon^2$. To compute the probability that u and v are isolated, we distinguish between two cases of the location of u. If u is within a distance δ of the unit square boundary, then the probability that u and v are isolated is at most $(1-\pi\delta^2/4)^{n-2}$, whereas if u is far from the boundary (at a greater distance than δ), the probability is at most $(1-\pi\delta^2)^{n-2}$. Therefore, according to the total probability theorem for any $u \in V$, we have

$$\begin{aligned} \Pr[A_u] \le{}& (n-1)\pi\varepsilon^2(4\delta - 4\delta^2)\left(1 - \frac{\pi\delta^2}{4}\right)^{n-2} \\ &+ (n-1)\pi\varepsilon^2(1 - 4\delta + 4\delta^2)(1-\pi\delta^2)^{n-2}. \end{aligned}$$

Similarly, we compute the lower bound of $\Pr[A_u]$. Again, we distinguish between two cases. If u is within ε from the unit square boundary, then the probability that $d(u,v) \le \varepsilon$ is at least $\pi\varepsilon^2/4$, otherwise it is at least $\pi\varepsilon^2$. As a result for any $u \in V$

$$\begin{aligned} \Pr[A_u] &\ge (n-1)\left(4\varepsilon\cdot\frac{\pi\varepsilon^2}{4} + (1-4\varepsilon)\pi\varepsilon^2\right)(1-\pi\delta^2)^{n-2} \\ &\ge (n-1)(1-4\varepsilon)\pi\varepsilon^2(1-\pi\delta^2)^{n-2}. \end{aligned}$$

Note that $\lim_{n\to\infty}\left(1-\pi\delta^2\right)^{n-2} = e^{-\pi\delta^2(n-2)}$ when $\delta \to 0$. Thus, by linearity of expectation

$$E[X] = \sum_{u\in V}\Pr[A_u] = \Theta(\phi(n)^2).$$

According to [25, Corollary 4.3.3] to prove that $X = \Theta(\mathrm{E}[X])$, it is sufficient to show that $\mathrm{Var}[X] = o(\mathrm{E}[X]^2)$. By definition

$$\begin{aligned} \mathrm{Var}[X] &= \sum_{z\in V}\mathrm{Var}[X_z] + \sum_{\substack{y,z\in V\\ y\ne z}}\mathrm{Cov}[X_y,X_z] \\ &\le E[X] + n^2\mathrm{Cov}[X_u,X_v] \end{aligned}$$

where u, $v \in V$ are any two fixed nodes, and

$$\begin{aligned} \mathrm{Cov}[X_u,X_v] &\le E[X_uX_v] - E[X_u]E[X_v] \\ &= \Pr[A_u \cap A_v] - \Pr[A_v]\Pr[A_v]. \end{aligned}$$

Next, we analyze $\Pr[A_u \cap A_v]$. We consider three cases.

Case 1) If $d(u,v) \leq \varepsilon$, then both A_u and A_v occur only if u and v are an isolated pair, and therefore

$$\Pr[A_u \cap A_v | d(u,v) \leq \varepsilon] \leq (1 - \pi\delta^2/4)^{n-2}.$$

Case 2) If $\varepsilon < d(u,v) < \delta$, then neither A_u or A_v can ever occur, and thus

$$\Pr[A_u \cap A_v | \varepsilon < d(u,v) < \delta] = 0.$$

Case 3) If $\delta < d(u,v)$, then u and v must be part of two different isolated pairs. The probability that there exist two other nodes y, z within distance ε from u and v, respectively, is at most $2\binom{n-2}{2}(\pi\varepsilon^2)^2$. The probability that u, y and v, z are isolated is at most $(1 - \pi\delta^2/2)^{n-4}$ when both u and v are within distance δ from the square boundary, and at most $(1-\pi\delta^2)^{n-4}$ otherwise. As a result

$$\begin{aligned}&\Pr[A_u \cap A_v | \delta < d(u,v)]\\ &\leq 2\binom{n-2}{2}(\pi\varepsilon^2)^2 \left((4\delta)^2 + (1-(4\delta - 4\delta^2)^2)(1 - \pi\delta^2)^{n-4}\right).\end{aligned}$$

Note that the probability that the conditions of the first and third cases occur is at most $\pi\varepsilon^2$ and $(1 - \pi\delta^2/4)$, respectively. By combining all three cases, we obtain an upper bound on $\Pr[A_u \cap A_v]$ (some obvious steps are omitted for simplicity)

$$\Pr[A_u \cap A_v] \leq \pi\varepsilon^2 + 2\binom{n-2}{2}(\pi\varepsilon^2)^2(16\delta^2 + 1).$$

Finally, we have (recall the lower bound for $\Pr[A_u]$)

$$\begin{aligned}\mathrm{Cov}[X_u, X_b] = &\Pr\left[A_u \cap A_v\right] - \Pr[A_u]\Pr[A_v]\\ \leq &\pi\varepsilon^2 + 2\binom{n-2}{2}(\pi\varepsilon^2)^2\left(16\delta^2 + 1\right)\\ &- (n-1)^2(\pi\varepsilon^2)^2(1 - 4\varepsilon)^2(1 - \pi\delta)^{2n-4}\\ \leq &\pi\varepsilon^2 + (n-1)^2(\pi\varepsilon^2)^2(16\delta^2 + 1\\ &- (1 - 4\varepsilon)^2(1 - \pi\delta)^n)\\ = &\pi\varepsilon^2 + (n-1)^2(\pi\varepsilon^2)^2 o(1).\end{aligned}$$

The last equality is obtained for sufficiently small δ. Recall that $\varepsilon = \frac{\phi(n)}{n}$ and $\delta = \frac{1}{\sqrt{n\phi(n)}}$. We can conclude that

$$\begin{aligned}\mathrm{Var}[X] &\leq E[X] + n^2(\pi\varepsilon^2 + (n-1)^2(\pi\varepsilon^2)^2 o(1))\\ &\leq o(\phi(n)^4) = o(E[X]^2).\end{aligned}$$

This concludes our proof. ■

We are ready to present the main theorem of this section.

Theorem 3.5: Let p be a homogeneous power assignment so that H_p is strongly connected. Then, $\mathrm{Cap}(H_p) = O\left(B\log\left(1 + \frac{\phi^3(n)}{n}\right)\right)$ and $\mathrm{Cap}(H_p) = \Omega\left(B\log\left(1 + \frac{1}{n\phi(n)\log^2 n}\right)\right)$, where $\phi(n)$ is any function with $\lim_{n\to\infty}\phi(n) = \infty$.

Proof: We prove the upper bound first. Let u, $v \in V$ be an (ε, δ)-isolated pair as defined in Lemma 3.4. For any node $w \in V \setminus \{u, v\}$, let $P^*_{w,u}$ be a path from w to u such that $\mathrm{Cap}(P^*_{w,v})$ is maximized. Let $x \in V \setminus \{u,v\}$ and $y \in \{u,v\}$ be two nodes such that $(x,y) \in P^*_{w,v}$ (it easy to verify that such an edge exists). Then, $\mathrm{SINR}(x,y) \leq \frac{1/\delta^2}{1/\varepsilon^2} = \frac{\phi(n)^3}{n}$, and therefore $\mathrm{Cap}(H_p) = O\left(B\log\left(1 + \frac{\phi^3(n)}{n}\right)\right)$. Note that there are at least $2n-4$ node pairs (every pair with a destination either u or v) that experience the same level of bottleneck SINR.

Next, we prove the lower bound. From Corollary 2.3, it follows that if H_p is strongly connected, then $p(u) = \Omega(\log n/n)$, $\forall\, u \in V$, and therefore the directed version of MST_V (obtained by replacing each edge with two edges in opposite directions) is a subgraph in H_p (note that all links in H_p are bidirectional as all the nodes are assigned the same power level). As a result, for any pair of nodes $x, y \in V$, there is a path in H_p that uses edges with length at most $O(\sqrt{\log n/n})$. Denote such a path by $P_{x,y}$. Combining with Lemma 3.3, for any edge $(u,v) \in P_{x,y}$ it holds $\mathrm{SINR}(u,v) = \Omega\left(\frac{n/\log n}{n^2\phi(n)\log n}\right)$, where $\phi(n)$ is any function with $\lim_{n\to\infty}\phi(n) = \infty$. As a result

$$\mathrm{Cap}(H_p, x, y) = \Omega\left(B\log\left(1 + \frac{1}{n\phi(n)\log^2 n}\right)\right)$$

for any $x, y \in V$. ■

Remark 1: Theorem 3.5 analyzes the capacity of the worst link in the network. However, some nodes may have a routing with links of substantially higher capacity. It is possible to find such a routing for every source node $u \in V$ by constructing a maximum bottleneck spanning tree of H_p rooted at u. A maximum bottleneck spanning tree is a spanning tree that maximizes the minimum cost edge. In our case, the cost of an edge $(u,v) \in E_p$ can be defined as $\mathrm{SINR}(u,v)$. It is possible to compute the above tree efficiently even with an additional requirement of a hop-count [26].

Remark 2: Although many links might have a very high capacity, there will also be a large number of node pairs with a limited link capacity, as follows from Lemma 3.2 and the following reasoning. Let e^* be the maximum length edge of MST_V, and p be a homogeneous power assignment such that H_p is strongly connected. Let H'_p be a graph obtained by removing all edges with length equal or greater than $|e^*|$ from H_p. Due to the fact that the longest edge of any spanning tree of G_V has length of at least $|e^*|$, H'_p has at least two strongly connected components. Let x, $y \in V$ be two nodes from different components. Let (u,v) be one of the edges removed from H_p with the maximum $\mathrm{SINR}(u,v)$. According to Corollary 2.3, $d(u,v) = \Omega\left(\sqrt{\frac{\log n}{n}}\right)$. From Observation 3.1 and Lemma 3.2, $\mathrm{SINR}(u,v) = O(1/\log^2 n)$. As every path from x to y in H_p uses one of the edges that were removed, we obtain: $\mathrm{Cap}(H_p, x, y) = O\left(B\log\left(1 + \frac{1}{\log^2 n}\right)\right)$. Note that the number of such pairs x, y is at least $\Omega(n)$.

B. Arbitrarily Small Throughput in a Nonrandom Case

As stated above, the location of nodes has a significant impact on the network capacity. We demonstrate that if the nodes

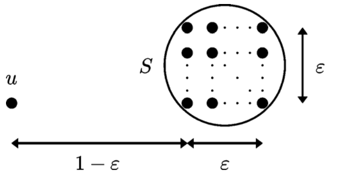


Fig. 3. Arbitrarily small throughput in a nonrandom case.

are placed in a unit square, without any assumption on their distribution, the throughput may be arbitrarily small.

An example for such a network is given in Fig. 3. There are two groups of nodes: an isolated node u and a dense uniform grid of $n-1$ nodes (denote this set by S), such that the distance between every two nodes in the grid is at least $\varepsilon/\sqrt{n-1}$, where ε is an arbitrarily small parameter. The next theorem shows that the throughput of any strongly connected communication graph induced by a power assignment for the nodes in Fig. 3 can be arbitrarily small.

Theorem 3.6: For any power assignment p such that the network in Fig. 3 is strongly connected, $\mathrm{Cap}(H_p) = \Omega\left(B\log\left(1+\varepsilon^2/n\right)\right)$.

Proof: Since H_p is strongly connected, every strongly connected spanning subgraph of H_p contains an edge (u,v), where v is some node in S. We show an upper bound on $\mathrm{SINR}(u,v)$. Note that for every $z \in S$, it holds $d(v,z) \le \sqrt{2}\varepsilon$, and hence $D(u,v) \ge (n-2)/(2\varepsilon^2)$. Following Observation 3.1, we can conclude that

$$\mathrm{SINR} \le \frac{1/(1-\varepsilon)^2}{(n-2)/(2\varepsilon^2)} = O\left(\varepsilon^2/n\right).$$

As ε can be arbitrarily small, the throughput can be arbitrarily small as well. ■

V. Heterogeneous Power Assignment

In this section, we allow nodes to have distinct power levels. We develop a power assignment algorithm that produces p such that $c(p) = O(1)\mathrm{OPT}_\mathrm{c}$, $l(p) = \Omega(1)\mathrm{OPT}_\mathrm{l}$, and $\mathrm{Cap}(H_p) = \Omega\left(B\log\left(1+\frac{1}{\sqrt{n}\log^2 n}\right)\right)$, where OPT_c and OPT_l are the minimum possible cost and maximum possible lifetime, respectively, of a power assignment that induces a strongly connected communication graph. Then, we show that this algorithm can be implemented in a distributed way.

A. Power Assignment Algorithm

From the analysis in the previous section, we can intuitively see that the closest nodes have the most impact on interference levels. The main idea of the algorithm is to create clusters of nodes with low transmission power, and consequently low interference, then connect the clusters, allowing only one node to transmit at high power level. By clustering, we wish to avoid multiple nodes, within a small distance from each other, transmitting at high power levels.

The algorithm Cluster-Based-Power-Assignment (CBPA) works as follows. The algorithm has two phases. In the first phase of the algorithm (lines 4–17), the clusters are created. In the second phase (lines 18–32), the nodes are assigned a transmission power. We now describe each phase in detail.

Cluster-Based-Power-Assignment (CBPA)

```
 1 $U \leftarrow V$
 2 foreach $u \in V$ do
 3     $\mathrm{type}[u] \leftarrow \mathrm{nil}; \mathrm{head}[u] \leftarrow \mathrm{nil}; \mathrm{members}[u] \leftarrow \mathrm{nil}$
 4 while $U \neq \emptyset$ do
 5     take any $u \in U$
 6     $M \leftarrow \left\{v \in V : d(u,v) \le \frac{1}{n^{3/4}}\right\}$
 7     $\mathrm{type}[u] \leftarrow \mathrm{HEAD}; \mathrm{head}[u] \leftarrow u; \mathrm{members}[u] \leftarrow M$
 8     remove $u$ and $M$ from $U$
 9     foreach $v \in M$ do
10         if $\mathrm{type}[v] == \mathrm{MEMBER}$ then
11             if $d(v, \mathrm{head}[v]) < d(u,v)$ then
12                 remove $v$ from $\mathrm{members}[u]$
13             else
14                 remove $v$ from $\mathrm{members}[\mathrm{head}[v]]$
15                 $\mathrm{head}[v] \leftarrow u$
16         else
17             $\mathrm{head}[v] \leftarrow u$
18 foreach $u \in V$ do
19     compute $\mathrm{MST}_V = (V, E)$
20     if $\mathrm{type}[u] == \mathrm{MEMBER}$ then
21         $p(u) \leftarrow d(u, \mathrm{head}[u])^2$
22     else
23         $p(u) \leftarrow 0$
24         foreach $v \in \mathrm{members}[u]$ do
25             if $d(u,v)^2 > p(u)$ then
26                 $p(u) \leftarrow d(u,v)^2$
27             foreach $(v,w) \in E$ do
28                 if $d(u, \mathrm{head}[w])^2 > p(u)$ then
29                     $p(u) \leftarrow d(u, \mathrm{head}[w])^2$
30         foreach $(u,x) \in E$ do
31             if $d(u, \mathrm{head}[x])^2 > p(u)$ then
32                 $p(u) \leftarrow d(u, \mathrm{head}[x])^2$
```

Phase I: The clusters are created iteratively. Each node should belong to some cluster and can either be a member of some cluster or its head. We use a temporary set U, which at first is initialized to be all the nodes in V. Then, while U is not empty, we pick an arbitrary node $u \in U$ (line 5). This is our new cluster head. The cluster is formed (lines 6–8) from all the nodes (denoted as M) that are within a distance of $\frac{1}{n^{3/4}}$ from u, and u itself. The nodes in M are defined as the cluster members of u (line 7). Then, we check for every node $v \in M$ if it is already defined as a member of some other cluster. If not, it stores u as its cluster head (line 17). Otherwise, v chooses to be the member of the cluster with the closest cluster head. Note that v cannot be a cluster head, as otherwise u would have been marked as a cluster member (by v or some other node) and removed from U (line 8). This process continues until U becomes empty.

Phase II: The power assignment phase is divided into cases. Each node that is a cluster member is assigned with a transmission power that is just enough to reach its cluster head (lines 20–21). Determining the power assignment for the cluster head is more complicated. A cluster head has

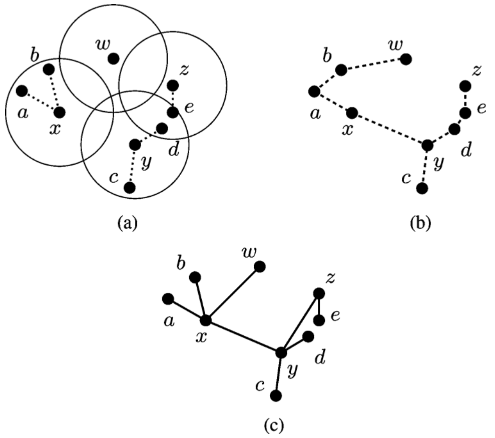


Fig. 4. Exposition of algorithm CBPA. (a) Clusters with w, x, y, and z as cluster heads. (b) Minimum weight spanning tree. (c) Induced topology.

to satisfy three requirements: 1) reach all its cluster members (lines 24–26); 2) reach all cluster heads of its neighbors in MST_V (lines 30–32); 3) reach all cluster heads of the nodes that are neighbors in MST_V of its cluster members (lines 27–29). It is easy to verify that all the edges that result from these requirements are bidirectional.

An example of four clusters with cluster heads w, x, y, and z is given in Fig. 4. The disks outline the cluster range of $1/n^{3/4}$, while the dotted lines identify the members of each cluster [Fig. 4(a)]. Note that d and e are both within the cluster range from two cluster heads, y and z; their membership is determined by the closest cluster head. The second phase of the algorithm induces a topology based on clusters and the minimum weight spanning tree [Fig. 4(b)]. The induced topology [Fig. 4(c)] has to fulfill several requirements as mentioned above: 1) cluster head connects to its members and the members connect to their cluster head [the edges (x,a), (x,b), (y,c), (y,d), (z,e)]; 2) cluster head reaches the cluster heads of its neighbors in the minimum spanning tree [the edges (x,y), (x,w)]; 3) cluster head reaches the cluster heads of its cluster members' neighbors in the minimum spanning tree [the edge (y,z)].

B. Analysis

First, we argue that H_p is strongly connected. Intuitively, we keep the connectivity of MST_V as cluster heads are able to use the edges of MST_V and cluster members do the same with the help of cluster heads. The next lemma proves that H_p is strongly connected.

Lemma 4.1: H_p is strongly connected.

Proof: For any pair of nodes $u, v \in V$, we show there is a path in H_p from u to v. Let $P = \langle u = z_1, z_2, \ldots, z_k = v\rangle$ be a path from u to v in MST_V. We construct a path in H_p by converting each edge $e \in P$ to a path in H_p. For each edge (z_i, z_{i+1}), $1 \le i \le k-1$, there are four cases to consider.

Case 1) z_i and z_{i+1} are both cluster heads. Therefore, it follows easily that $\langle z_i, z_{i+1}\rangle \in H_p$.

Case 2) z_i is a cluster head, and z_{i+1} is a cluster member. If they both belong to the same cluster, then clearly $\langle z_i, z_{i+1}\rangle \in H_p$. Otherwise, $\langle z_i, \mathrm{head}[z_{i+1}], z_{i+1}\rangle \in H_p$, as z_i reaches the cluster head of z_{i+1}, which in turn reaches z_{i+1}.

Case 3) z_i is a cluster member, and z_{i+1} is a cluster head. If they both belong to the same cluster, then clearly $\langle z_i, z_{i+1}\rangle \in H_p$. Otherwise, z_i reaches its cluster head, $\mathrm{head}[z_i]$, which in turn reaches z_{i+1}, and therefore $\langle z_i, \mathrm{head}[z_i], z_{i+1}\rangle \in H_p$.

Case 4) z_i and z_{i+1} are both cluster members. If they share the same cluster head, then $\langle z_i, \mathrm{head}[z_i], z_{i+1}\rangle \in H_p$. Otherwise, z_i reaches its cluster head, which in turn reaches the cluster head of z_{i+1}, which eventually reaches z_{i+1}. We conclude that $\langle z_i, \mathrm{head}[z_i], \mathrm{head}[z_{i+1}], z_{i+1}\rangle \in H_p$.

We showed how each edge in P can be converted into a path in H_p. Hence, there exists a path between any pair of nodes in H_p. ■

Following the path conversion described in Lemma 4.1 we can see that for any pair of nodes $u, v \in V$, there exists a path in H_p with two types of edges only: 1) between cluster heads; 2) between cluster members and their respective cluster heads. This gives us the next important observation.

Observation 4.2: For any pair of nodes $u, v \in V$, there exists a path P in H_p such that for any $(x,y) \in P$ one of the following conditions holds.

1) Either $\mathrm{head}[x] = y$ or $\mathrm{head}[y] = x$.
2) x and y are both cluster heads.

The next lemma analyzes the cost of the power assignment and the network lifetime.

Lemma 4.3: Let $\mathrm{OPT_c}$ and $\mathrm{OPT_l}$ be the minimum possible cost and maximum possible lifetime, respectively, of a power assignment that induces a strongly connected communication graph. Then, $c(p) = O(1)\mathrm{OPT_c}$, $l(p) = \Omega(1)\mathrm{OPT_l}$.

Proof: The lower bound for the network lifetime is based on the observation that for every $u \in V$, $p(u) = O(\log n/n)$. To prove that, we take a closer look at CBPA. The power is assigned in four different places:

Lines 21, 26: From the construction of M, it follows that $d(u, \mathrm{head}[u])$, $d(u,v) \le 1/n^{3/4}$.

Line 29: From Corollary 2.3, $d(v,w) = O\left(\sqrt{\frac{\log n}{n}}\right)$. Note that $d(u,v), d(w, \mathrm{head}[w]) \le 1/n^{3/4}$. As a result

$$\begin{aligned} d(u, \mathrm{head}[w]) &\le d(u,v) + d(v,w) + d(w, \mathrm{head}[w]) \\ &= O\left(\sqrt{\frac{\log n}{n}}\right). \end{aligned}$$

Line 32: Following a similar reasoning

$$d(u, \mathrm{head}[x]) \le d(u,x) + d(x, \mathrm{head}[x]) = O\left(\sqrt{\frac{\log n}{n}}\right).$$

The network lifetime is therefore $l(p) = \min_{u \in V} \frac{1}{p(u)} = \Omega\left(\frac{n}{\log n}\right)$. From Theorem 2.2 and Corollary 2.3, it easily follows that $l(p) = \Omega(1)\mathrm{OPT_l}$.

To show an upper bound on the cost of the power assignment, we distinguish between cluster members and cluster heads. Denote by M and H the sets of cluster members and cluster heads, respectively. According to CBPA, the power assignment of each cluster member is upper bounded by $\frac{1}{n^{3/2}}$ (lines 20–21). As a result, $\sum_{u \in M} \le 1/\sqrt{n}$.

Let u be a cluster head. Recall that the power assignment of a cluster head is based on three requirements: 1) reach all its cluster members (lines 24–26); 2) reach all cluster heads of its neighbors in MST_V (lines 30–32); 3) reach all cluster heads of the nodes that are neighbors in MST_V of its cluster members (lines 27–29).

Intuitively, most of the energy is consumed by connecting to other cluster heads. We denote by $E(u)$ the MST_V edges that are either adjacent to u or to one of its cluster members. By combining the three requirements above, we derive the following bound on $p(u)$: $p(u) \le \max_{e\in E(u)} w(e) + \frac{2}{n^{3/2}}$. Therefore

$$\sum_{u\in H} p(u) \le \sum_{u\in H}\left(\max_{e\in E(u)} w(e) + \frac{2}{n^{3/2}}\right) \le \sum_{u\in H}\max_{e\in E(u)} w(e) + 2/\sqrt{n}.$$

Note that each MST_V edge (x,y) appears only in $E(\mathrm{head}[x])$ and $E(\mathrm{head}[y])$. That is, for any $z\in H$, such that $z \ne \mathrm{head}[x]$ and $z \ne \mathrm{head}[y]$, $(x,y)\notin E(z)$. As a result

$$\sum_{u\in H}\max_{e\in E(u)} w(e) \le 2\sum_{e\in E(\mathrm{MST}_V)} w(e) = 2w(\mathrm{MST}_V).$$

The cost of p is bounded by combining the power consumption of both cluster members and heads

$$c(p) = \sum_{u\in M} p(u) + \sum_{u\in H} p(u) \le 3/\sqrt{n} + 2w(\mathrm{MST}_V).$$

Finally, due to Theorems 2.1 and 2.4, we obtain the required bound, $c(p) = O(1)\mathrm{OPT}_c$. ■

We now prove the main theorem of this section.

Theorem 4.4: $\mathrm{Cap}(H_p) = \Omega\left(B\log\left(1+\frac{1}{\sqrt{n}\log^2 n}\right)\right)$.

Proof: According to Observation 4.2, for each pair of nodes u, $v\in V$, there exists a path P in H_p with only two types of edges. We show that for each $(x,y)\in P$

$$\mathrm{SINR}(x,y) = \frac{p(x,y)\cdot d(x,y)^{-2}}{I(x,y)} = \Omega\left(\frac{1}{\sqrt{n}\log^2 n}\right).$$

By definition, for each edge $(x,y)\in P$, $p(x)\ge d(x,y)^2$. We now focus our analysis on $I(x,y)$.

Recall the grid constructed in Section III, and suppose $y\in N(i,j)$. Let $R = R(i,j,1)\cup R(i,j,2)$. We stated in the proof of Lemma 3.3 that $|R| = O(\log n)$. We distinguish between two cases.

Case 1) y is a cluster member. Then, by Observation 4.2 $x = \mathrm{head}[y]$. Let M be all the cluster members in R, and let H be all the cluster heads in R. Note that $y\in R(i,j,1)\subset R$ by definition, whereas x is restricted to $R(i,j,1)\cup R(i,j,2) = R$ due to the fact that $d(x,y)\le 1/n^{3/4}$. Hence, the interference sensed over the link (x,y) can be expressed as

$$I(x,y) = \sum_{z\in M\setminus\{y\}}\frac{p(z)}{d(z,y)^2} + \sum_{z\in H\setminus\{x\}}\frac{p(z)}{d(z,y)^2} + \sum_{z\in V\setminus R}\frac{p(z)}{d(z,y)^2}.$$

From Theorem 2.5, the distance between any pair of nodes is at least $\Omega\left(\frac{1}{n\phi(n)}\right)$, where $\phi(n)$ is any function with $\lim_{n\to\infty}\phi(n)=\infty$. From the algorithm CBPA, it follows that if z is a cluster member, then $p(z)\le\frac{1}{n^{3/2}}$. Therefore

$$\sum_{z\in M\setminus\{y\}}\frac{p(z)}{d(z,y)^2} \le O\left(|M|\cdot\frac{n^2\phi^2(n)}{n^{3/2}}\right) = O(\sqrt{n}\phi^2(n)\log n).$$

The distance between y and any cluster head, which is not x, is at least $\frac{1}{2n^{3/4}}$, as each node belongs to a cluster of the closest head (lines 10–15). In the proof of Lemma 4.3, we showed that $p(z) = O(\log n/n)$ for any $z\in V$. As a result

$$\sum_{z\in H\setminus\{x\}}\frac{p(z)}{d(z,y)^2} \le O\left(|H|\cdot\frac{\log n\cdot n\sqrt{n}}{n}\right) = O(\sqrt{n}\log^2 n).$$

An upper bound of $\log^2 n$ for the third addend is achieved very similarly to the proof of Lemma 3.3. Again, we use the fact that all the nodes are assigned a transmission power that is $O(\log n/n)$. Let $R_k = R(i,j,k)$, $k\in\left\{3,\ldots,\sqrt{\frac{n}{16\log n}}\right\}$. Eventually

$$\begin{aligned}\sum_{z\in V\setminus R}\frac{p(z)}{d(z,y)^2} &= \sum_{k=3}^{\sqrt{\frac{n}{16\log n}}}\sum_{z\in R_k}\frac{p(z)}{d(z,y)^2}\\ &= O\left(\sum_{k=3}^{\sqrt{\frac{n}{16\log n}}}|R_k|\cdot\frac{\log n}{n}\cdot\frac{n}{k^2\log n}\right)\\ &= O\left(\sum_{k=3}^{\sqrt{\frac{n}{16\log n}}}\frac{\log n}{k}\right) = O(\log^2 n).\end{aligned}$$

We conclude that $I(x,y) = O(\sqrt{n}\log^2 n)$.

Case 2) y is a cluster head. As in the first case, let M and H be cluster members and cluster heads in R. Then

$$I(x,y) \le \sum_{z\in M}\frac{p(z)}{d(z,y)^2} + \sum_{z\in H\setminus\{y\}}\frac{p(z)}{d(z,y)^2} + \sum_{z\in V\setminus R}\frac{p(z)}{d(z,y)^2}.$$

Similarly to the first case, we can upper-bound the first addend by using Theorem 2.5 and the fact that each cluster member is assigned a power of at most $\frac{1}{n^{3/2}}$

$$\sum_{z\in M}\frac{p(z)}{d(z,y)^2} = O\left(|M|\cdot\frac{n^2\phi^2(n)}{n^{3/2}}\right) = O(\sqrt{n}\phi^2(n)\log n).$$

From the construction, it follows that all the cluster heads are separated by a distance of at least $1/n^{3/4}$. Combining with the bound of $O(\log n/n)$ on the transmission power, we obtain

$$\sum_{z\in H}\frac{p(z)}{d(z,y)^2} = O\left(|H|\cdot\frac{n^{3/2}\log n}{n}\right) = O(\sqrt{n}\log^2 n).$$

Finally, the third addend is upper-bounded exactly as in the first case. We conclude $I(x,y) = O(\sqrt{n}\log^2 n)$ in the second case as well.

Note that both cases cover the conditions of Observation 4.2, which rests our proof. ■

VI. Distributed Implementation of CBPA

Sometimes, especially in wireless deployments, it is impossible to have a central entity that coordinates the network. In this section, we present a distributed asynchronous implementation of the CBPA algorithm that is called Distributed-CBPA and is executed by each node $u \in V$. The algorithm requires each node to have a unique ID, the knowledge of the total number of nodes, n, and the size of the deployment area. The distributed algorithm is composed of four phases (see pseudocode): (a) MST_V computation; (b) clustering; (c) verification; and (d) power assignment. We assume that the size of the deployment area is 1. In what follows, we discuss each of the phases in detail.

Distributed-CBPA(u)

```
    // Phase (a)
1   $N \leftarrow \mathrm{nil}; r \leftarrow \sqrt{\frac{8\log n}{\pi(n-1}} + \frac{2}{n^{3/4}}$
2   send HELLO$\langle u\rangle$ in range $r$
3   while no timeout do
4       on receive HELLO$\langle v\rangle$ do
5           add $v$ to $N$; estimate the distance $d(u,v)$ by using [27]
6   Compute a minimum spanning tree (MST by using [28], where $N$ are the neighbors of $u$ and $d(u,v)^2, v \in N$, is the weight function of the edge $(u,v)$
    // Phase (b)
7   Send the information about MST edges adjacent to $u$ to all other nodes in the network over the edges of $\mathrm{MST}_V$; Wait to receive all $n-1$ messages from other nodes before continuing; Construct a local view of $\mathrm{MST}_V = (V,E)$
8   $U \leftarrow V$; type $\leftarrow$ NONE; potential_heads $\leftarrow 0$
9   while $U \neq \emptyset$ do
10      if $u \in U$ and $u$ has the minimum ID in $U$ then
11          type $\leftarrow$ HEAD;
            $M \leftarrow \{v : d(u,v) \leq 1/n^{3/4}, v \in N\}$
12          remove $u$ and $M$ from $U$
13          broadcast CLUSTER$\langle u, M\rangle$ over $\mathrm{MST}_V$
14      on receive CLUSTER$\langle v, M(v)\rangle$ do
15          remove $v$ and $M(v)$ from $V$
16          if $u \in M(v)$ then
17              potential_heads $\leftarrow$ potential_heads $+ 1$
18              if type == NONE then
19                  type $\leftarrow$ MEMBER; head $\leftarrow v$
20              if type == MEMBER and $d(u,v) < d(u,\mathrm{head})$ then
21                  head $\leftarrow v$
    // Phase (c)
22  if type == HEAD then
23      $\forall v \in M$ send VERIFY$\langle u\rangle$ to $v$
24      $S \leftarrow M$
25      while $S \neq \emptyset$ do
26          on receive KEEP$\langle v\rangle$ do
27              remove $v$ from $S$
28          on receive REMOVE$\langle v\rangle$ do
29              remove $v$ from $S$
30              remove $v$ from $M$
31  else
32      while potential_heads $> 0$ do
33          on receive VERIFY$\langle v\rangle$ do
34              potential_heads $\leftarrow$ potential_heads $- 1$
35              if head == $v$ then
36                  send KEEP$\langle u\rangle$ to $v$
37              else
38                  send REMOVE$\langle u\rangle$ to $v$
    // Phase (d)
39  if type == HEAD then
40      broadcast HEAD$\langle u, u\rangle$ over $\mathrm{MST}_V$
41      received_count $\leftarrow 0$
42      while received_count $< n-1$ do
43          on receive HEAD$\langle v, \mathrm{head}(v)\rangle$ do
44              received_count $\leftarrow$ received_count $+ 1$
45              store information about the cluster head of $v$
46      compute $p(u)$ as in CBPA (lines 23–32)
47  else
48      HEAD$\langle u, \mathrm{head}\rangle$ over $\mathrm{MST}_V$
49      $p(u) \leftarrow d(u,\mathrm{head})^2$
```

A. MST_V Computation [Phase (a), Lines 1–6]

First, each node $u \in V$ transmits its ID to all close nodes by sending HELLO$\langle u\rangle$ in the range $r = \sqrt{\frac{8\log n}{\pi(n-1)}} + \frac{2}{n^{3/4}}$. Recall that n is known to all nodes, and the area size is 1, which can be easily adapted to any area size by multiplying the transmission range by the actual size of the deployment area. Next, u waits for HELLO$\langle v\rangle$ messages from all the other nodes within range r. Every time a node receives a HELLO$\langle v\rangle$ message, it adds v to its neighbor list N. Using the standard methods described in [27], it is possible to estimate all the distances $d(u,v)$, $v \in N$ (we skip the discussion about correct reception of all these transmissions). We can now compute the MST_V by using the Distributed Algorithm for Minimum-Weight Spanning Trees (DMST) [28], where the initial set of neighbors of u is N and the weight of the edge (u,v), $v \in N$ is $d(u,v)^2$.

To show the correctness of MST_V computation, we need the following technical lemma from [29].

Lemma 5.1 ([29]): For n points placed uniformly in the unit square, let M_n (respectively, M'_n) be the longest edge length of the nearest-neighbor graph (respectively, the minimum spanning tree) on these points. Then, $\lim_{n\to\infty} \Pr[M_n = M'_n] = 1$.

Due to the upper bound of Theorem 2.5 and Lemma 5.1, we can easily see that the subset of nodes N is a superset of the neighbors of u in MST_V, i.e., MST_V is a subgraph of H_p, where $p(u) = r^2$, for every $u \in V$. This fact proves the correctness of MST_V computation.

B. Clustering [Phase (b), Lines 7–21]

Before clustering can take place, every node needs to acquire the IDs of the other nodes in the network. Each node sends a message that contains its ID and the IDs of the adjacent nodes in MST_V to all the other nodes in the network over the edges of MST_V. Essentially, these are n tree broadcasts that happen simultaneously.[3] Each node waits until it has a full knowledge of all the other nodes in the networks and the edges of MST_V (recall that each node holds information about the values of n so it waits to receive $n-1$ messages). After this step, each node

[3] Broadcast is an operation of sending a message to all the other nodes in the network.

can construct a local view of $\mathrm{MST}_V = (V, E)$. Any subsequent broadcasts are carried over the MST_V as well.

Next, the actual clustering takes place, which is a distributed implementation of the first phase of CBPA. The set U holds the subset of nodes that are still not classified as a cluster head or cluster member as perceived by node u. The while-loop is executed while U is not empty. If u is still in U and has the lowest ID in U, it associates itself as a cluster head, compiles a subset of nodes M as its cluster members, and broadcasts (over the edges of MST_V) the $\texttt{CLUSTER}\langle u, M\rangle$ message that holds the information that u has associated itself as a cluster head and M as its cluster members. In the next lemma, we argue that a node cannot receive a CLUSTER message that identifies it as a cluster member if it already associated itself as a cluster head.

Lemma 5.2: When u receives $\texttt{CLUSTER}\langle v, M(v)\rangle$ and $\mathrm{type} == \mathrm{HEAD}$, then $u \notin M(v)$.

Proof: Suppose by contradiction that $\mathrm{type} == \mathrm{HEAD}$ and u receives a message $\texttt{CLUSTER}\langle v, M(v)\rangle$ so that $u \in M(v)$. If the ID of u is greater than the ID of v, then u could not change type to HEAD prior to receiving the message. Otherwise, if the ID of u is less than the ID of v, then u must have broadcasted the appropriate CLUSTER message that had to be received by v before it could generate its CLUSTER message. Since $u \in M(v)$, then $d(u, v) \leq 1/n^{3/4}$ and therefore v must have been classified as a member of the cluster constructed by u, and therefore the $\texttt{CLUSTER}\langle v, M(v)\rangle$ message could not be generated—a contradiction. ■

When node u receives a $\texttt{CLUSTER}\langle v, M(v)\rangle$ message, it immediately removes v and $M(v)$ from U. If type is NONE and $u \in M(v)$, then u associates itself as a cluster member and sets v as its cluster head. If type is MEMBER and $u \in M(v)$, then u considers v as a potential cluster head and switches to v if it is closer than its current head (similar to the first phase of CBPA). According to Lemma 5.2 it cannot happen that $\mathrm{type} == \mathrm{HEAD}$ and $u \in M(v)$.

Note the $\mathrm{potential\_heads}$ counter that counts the number of cluster heads that make a claim for u to be their cluster member. We discuss this variable in the next phase.

C. Verification (Phase (c), Lines 22–32]

The third phase is the verification of the clustering algorithm. Note that after the second phase, a single node can be classified as a cluster member of more than two clusters since, in order to keep the algorithm asynchronous, they never reported to their current cluster heads of a possible cluster change.

If u is a cluster head, it initiates VERIFY messages to all the nodes in M (which it considers to be in its cluster) and waits for the responses that indicate whether the node has to be removed from a cluster or kept. In case of a $\texttt{REMOVE}\langle v\rangle$ response, the node v is removed from M; in case of $\texttt{KEPT}\langle v\rangle$, it remains in M. The cluster head uses a set of nodes S (initialized to M) to track the progress of responses. Only when S becomes $\emptyset$ does the algorithm move on to the next phase.

If u is a cluster member, it waits for $\mathrm{potential\_heads}$ $\texttt{VERIFY}\langle v\rangle$ messages as u is considered by $\mathrm{potential\_heads}$ cluster heads to be in their cluster. It sends back a $\texttt{KEEP}(u)$ in case $\mathrm{head} == v$ and $\texttt{REMOVE}\langle u\rangle$ otherwise. With every VERIFY message, $\mathrm{potential\_heads}$ is decreased by 1; the algorithm continues to the next phase when it becomes 0.

D. Power Assignment [Phase (d), Lines 39–49]

To execute the final phase properly, every cluster head has to acquire information about the associations between cluster members and their cluster heads. The phase starts with every node broadcasting a HEAD message with its association information to the whole network over MST_V.

If u is a cluster member, then it chooses a power level of $d(u, \mathrm{head})^2$. It can decide on the power level since the value of $d(u, \mathrm{head})$ was estimated in the first phase. A cluster member can simply ignore the HEAD messages (of course, it still forwards them, as required, to their destinations).

If u is a cluster head, then it waits to receive all $n-1$ HEAD messages from the other nodes and execute the second phase of the CBPA algorithm locally as they have the knowledge of their cluster members, MST_V edges E, and the cluster head of every other node in the network. Note that for every node v that u has to reach according to the second phase of CBPA, it holds $v \in N$ due to our choice of r, and therefore $d(u, v)$ was estimated in the first phase.

E. Message Complexity and Remarks

Note that our algorithm is completely asynchronous [except for network discovery in Phase (a)]. Every node advances to the next phase after completing the previous one. Next, we analyze the message complexity of each of the phases.

The first phase consists of network discovery (HELLO message) and the execution of DMST [28]. There are n Hello messages as each node transmits only one. The message complexity of DMST is $5n \log n + |E'|$, where E' are the edges of the initial graph. Based on Lemma 2.6, it is easy to see that for each node u, the size of N (the number of nodes within distance r) is $|N| = O(\log n)$, and therefore the message complexity of DMST (and of the whole phase) is $O(n \log n)$.

The second phase is composed of initial n broadcasts to disseminate the MST_V information, and then at most another n broadcasts of the CLUSTER messages (we refer the reader to some standard techniques of data dissemination [30] that address various challenges in the wireless domain). Since each broadcast is routed along the edges of MST_V, the message complexity of a single broadcast is $O(n)$, and therefore the total message complexity of this phase is $O(n^2)$.

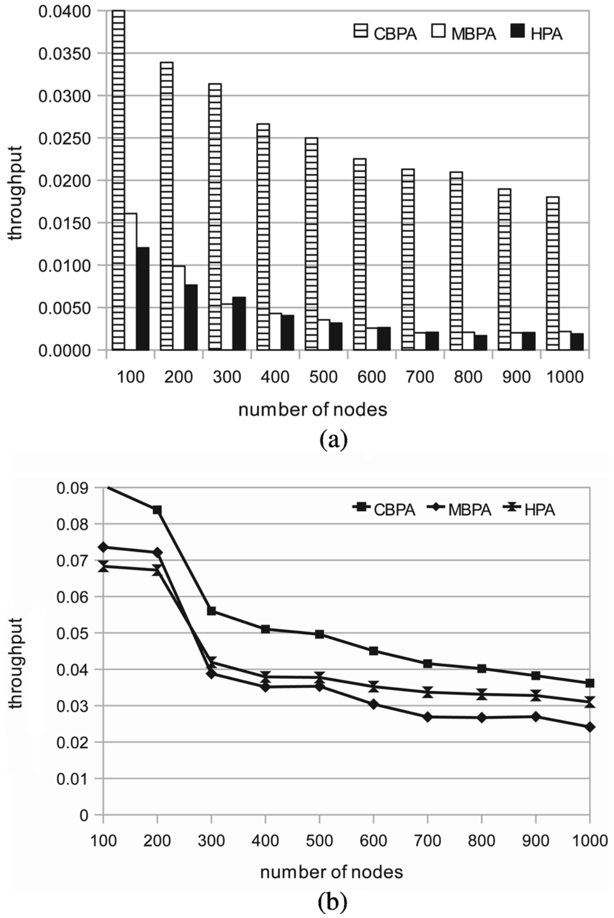
In the third phase, there are three types of messages: VERIFY, KEEP, and REMOVE. We note that for every VERIFY message, there is only one, either KEEP or REMOVE, response. Again, relying on Lemma 2.6, there are at most $O(\log n)$ possible cluster member candidates in M for each cluster head u, and therefore the total message complexity of this stage is $O(n \log n)$.

The fourth, and final, stage is composed of at most n broadcast HEAD messages with a total message complexity of $O(n^2)$.

As a result, the overall message complexity of Distributed-CBPA is $O(n^2)$. We observe that it is possible to improve the message complexity by avoiding the expensive broadcasts since each node has to have knowledge of its close environment, which is sufficient for clustering. This, however, is outside the scope of this paper.

VII. Numerical Results

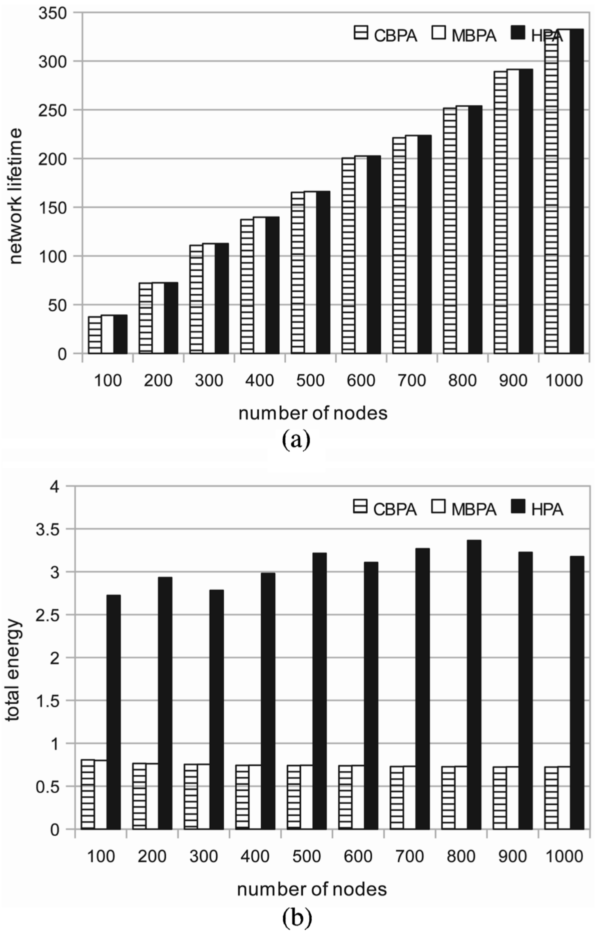
To the best of our knowledge, we are the first to study the throughput when all the nodes transmit simultaneously. As a result, in our simulations we compare the performance of the



(a)

(b)

Fig. 5. Throughput comparison between CBPA, MBPA, and HPA. (a) Minimum throughput. (b) Average throughput.



(a)

(b)

Fig. 6. Energy efficiency comparison among CBPA, MBPA, and HPA (a) Network lifetime. (b) Total energy consumption.

CBPA algorithm to two naive methods: minimum weight spanning tree-based power assignment (MBPA) (introduced in [18]) and homogeneous power assignment (HPA). In MBPA, each node is assigned to reach its farthest neighbor in MST_V, thus forming a strongly connected graph. We choose this power assignment due to its energy efficiency, as its cost is at most twice the cost of an optimal one [18]. In HPA, each node $u \in V$ is assigned a power of $p(u) = |e^*|^2$, where e^* is the maximum length edge in MST_V, which results in a strongly connected communication graph.

We compare the power assignments through two measures: *throughput* and *energy efficiency*. The simulations have been carried out for values of n ranging from 100 to 1000 with steps of 100. Each point in the plot is an average of 50 tries.

A. Throughput

To compute the capacity of the minimum capacity link, for each pair of nodes we find the maximum bottleneck path in terms of SINR and then compute the capacity (according to the definition in Section II with a channel bandwidth of $B = 1$) of the minimum SINR link in the path. We compare the minimum (among all pairs of nodes) and the average capacity for the three power assignments as depicted in Fig. 5.

We can see that CBPA achieves a much higher minimum throughput than MBPA and HPA [Fig. 5(a)]. More specifically, the ratio between CBPA and MBPA grows from 2.7 to 8.6, while for HPA it varies from 4.4 $(n = 800)$ to 11.5 $(n = 600)$. The differences for the average throughput are more subtle as shown in Fig. 5(b), with CBPA performing consistently better than MBPA and HPA.

B. Energy Efficiency

Energy efficiency was measured by evaluating the network lifetime and total energy consumption of the power assignments (Fig. 6).

The network lifetime of CBPA, MBPA, and HPA is presented in Fig. 6(a). The figure shows the value of $\min_{u\in V} b(u)/p(u)$, where p is either CBPA, MBPA, or HPA. We assume that the initial battery charge is $b \equiv 1$ for all nodes. It is easy to observe that both MBPA and HPA have the same network lifetime (as they share the same maximum power assigned), which is also the best possible (Theorem 2.2). It can be clearly seen that the network lifetime of CBPA is almost optimal as well.

In terms of total energy consumption, HPA appears to consume significantly more energy than CBPA and MBPA [Fig. 6(b)], which is quite surprising due to its optimality in terms of network lifetime. The reason for this discrepancy lies in the fact that the network lifetime reflects the moment that the *first* node runs out of its battery charge, and therefore even if all the nodes share the same transmission range, as in the case of HPA, the network lifetime will not be affected at all, while the total energy consumption will deteriorate considerably. It

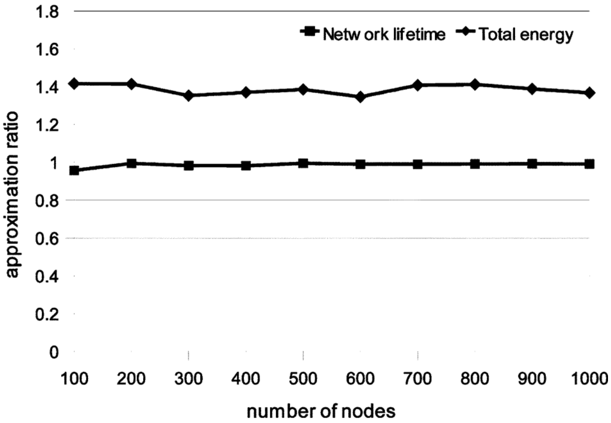


Fig. 7. Approximation ratios of CBPA for network lifetime and total energy consumption.

is also interesting to note that the total energy consumption of CBPA is almost identical to MBPA, which is at most two times higher than optimal possible.

Fig. 7 shows the approximation ratios, in terms of energy efficiency, for the power assignment produced by CBPA. We compare the network lifetime and total energy consumption with the respective, upper and lower, optimal bounds according to Theorems 2.2 and 2.1. It appears that the network lifetime is almost optimal, while the total energy consumption is within a factor of ≈1.4 from the best possible one.

VIII. Related Work

This paper combines several different areas of research in wireless networks, which can be roughly divided into *capacity in wireless networks* and *energy-efficient topology control*. We briefly sketch current developments in each of the areas.

1) Capacity in Wireless Networks: The asymptotic capacity for wireless ad hoc networks has been intensively studied under different channel models. The existing research can be divided into two main channel models: the threshold-based channel model and the Gaussian channel model.

Threshold Model: The threshold-based model is subdivided into two interference models—the protocol model (PR) and the physical model (PH). In the former, a transmission by node u is successfully received by a target node v iff node v is sufficiently apart from the source of any other simultaneous transmission. In the latter, a transmission from u is received at v if $\mathrm{SINR}(u,v)$ is above a certain threshold. These models were introduced in the groundbreaking work of Gupta and Kumar [1], where they studied the asymptotic capacity of the network for unicast multihop sessions, under the PR and PH models, where each node chooses a random destination and is transmitting at B bits per second. They showed that a transport capacity of $\Theta(B/\sqrt{n})$ per node is feasible for arbitrary placement of nodes. They also considered random homogeneous networks and showed that a throughput of $O(B/\sqrt{n \log n})$ is feasible. Grossglauser and Tse [2] demonstrated how mobility can be used to increase the throughput. Gastpar and Vetterli [31] addressed the problem of only one active source/destination pair, while all other nodes assist this transmission, for both PR and PH models. Moscibroda and Wattenhofer [3] studied the scheduling complexity of connectivity, i.e., the minimal amount of time required until a connected structure can be scheduled under the PH model, and presented an $O(\log^4 n)$ scheduling algorithm. See additional results in [4] and [5].

Gaussian Channel Model: Much less research was done in the context of Gaussian channels. Franceschetti *et al.* [6] show that a throughput of $\Omega(1/\sqrt{n})$ can be achieved for random networks and random unicast sessions. Zheng [7] studied the data dissemination capacity in power-constrained networks. The author showed that the total broadcast capacity is $\Theta(P/\log n)$, when each node transmits at a power P. Keshavarz-Haddad and Riedi [32] study static wireless networks with the goal of assessing the impact of topology and traffic pattern on capacity. For additional results see [8] and [9].

The results above use channel access methods, such as the TDMA scheme to schedule node transmissions, which requires *synchronization* and does not allow simultaneous transmissions of *all* nodes simultaneously.

2) Energy-Efficient Topology Control: The first to initiate the study of topology control through varying the power assignment of wireless nodes were Chen and Huang [18]. They addressed the problem of *minimizing the total energy* of a strongly connected graph and gave a 2-approximation algorithm based on finding the minimum spanning tree and showed that the problem is NP-hard. Kirousis *et al.* showed the problem is NP-hard for the three-dimensional Euclidean space for any value of κ. The NP-hardness for the two-dimensional Euclidean space for any value of κ was proven in [33]. An excellent survey can be found in [34]. *Maximizing the network lifetime* in the case of uniform battery charges is equivalent to minimizing the maximum power level assigned to any node. The first to study this problem were Ramanathan and Hain [35], who provided an optimal polynomial time algorithm for this problem under the strong connectivity property. A general approach, which leads to polynomial-time algorithms was developed in [36]. In [37], a PTAS for the problem under various network tasks was developed by devising an LP formulation for the problem. For additional results, see [38] and [39].

IX. Conclusion and Future Work

This paper considers the problem of bottleneck link capacity under the Gaussian channel model in strongly connected random wireless ad hoc networks, with n nodes independently and uniformly distributed in a unit square. The nodes are assumed to have two transceivers (one for transmission and one for reception), which allows all nodes to transmit *simultaneously*.

We addressed two network types. For homogeneous networks, we drew theoretical lower and upper bounds on the achievable bottleneck link capacity and demonstrated that for a nonrandom node distribution, the network throughput can be arbitrarily small. In the case of heterogeneous networks, we developed a cluster-based power assignment algorithm (CBPA), with a provable lower bound on the bottleneck link capacity. We also showed that CBPA achieves a constant factor approximation for both total energy consumption and network lifetime, relative to *any* strongly connected topology. In addition, we developed a distributed implementation of CBPA with $O(n^2)$ message complexity.

In our simulations, we evaluated the performance of CBPA and compared it to two other power assignment algorithms (MBPA and HPA). The measurements showed that CBPA outperforms other algorithms in terms of throughput and is also very energy-efficient. It appears that the network lifetime of CBPA is almost optimal, while the total energy consumption is within a factor of $\approx$1.4 from the best possible.

As possible future directions, it would be of great interest to improve the capacity bound in the heterogeneous case and study nonrandom node deployments.

REFERENCES

[1] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[2] M. Grossglauser and D. N. C. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 477–486, Aug. 2002.

[3] T. Moscibroda and R. Wattenhofer, "The complexity of connectivity in wireless networks," in *Proc. IEEE INFOCOM*, 2006, pp. 1–13.

[4] H. R. Sadjadpour, Z. Wang, and J. J. Garcia-Luna-Aceves, "The capacity of wireless ad hoc networks with multi-packet reception," *IEEE Trans. Commun.*, vol. 58, no. 2, pp. 600–610, Feb. 2010.

[5] Y. Wang, J. C. S. Lui, and D.-M. Chiu, "Understanding the paradoxical effects of power control on the capacity of wireless networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 1, pp. 406–413, Jan. 2009.

[6] M. Franceschetti, O. Dousse, D. N. C. Tse, and P. Thiran, "Closing the gap in the capacity of wireless networks via percolation theory," *IEEE Trans. Inf. Theory*, vol. 53, no. 3, pp. 1009–1018, Mar. 2007.

[7] R. Zheng, "Asymptotic bounds of information dissemination in power-constrained wireless networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 1, pp. 251–259, Jan. 2008.

[8] C. Wang, S. Tang, X.-Y. Li, C. Jiang, and Y. Liu, "Multicast throughput of hybrid wireless networks under Gaussian channel model," in *Proc. ICDCS*, 2009, pp. 111–118.

[9] S. Li, Y. Liu, and X.-Y. Li, "Capacity of large scale wireless networks under Gaussian channel model," in *Proc. ACM MobiCom*, 2008, pp. 140–151.

[10] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-efficient collision-free medium access control for wireless sensor networks," in *Proc. SenSys*, 2003, pp. 181–192.

[11] L. van Hoesel and P. Havinga, "A lightweight medium access protocol (LMAC) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches," in *Proc. INSS*, 2004, pp. 205–208.

[12] A. Chandrakasan, R. Amirtharajah, S. Cho, J. Goodman, G. Konduri, J. Kulik, W. Rabiner, and A. Wang, "Design considerations for distributed microsensor systems," in *Proc. CICC*, 1999, pp. 279–286.

[13] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Commun.*, vol. 11, no. 6, pp. 54–61, Dec. 2004.

[14] K. Pahlavan and A. H. Levesque, *Wireless Information Networks*. New York: Wiley-Interscience, 1995.

[15] A. Farina and L. Ortenzi, "Effect of ADC and receiver saturation on adaptive spatial filtering of directional interference," *J. Signal Process.*, vol. 83, pp. 1065–1078, 2003.

[16] J. I. Choi, M. Jain, K. Srinivasan, P. Levis, and S. Katti, "Achieving single channel, full duplex wireless communication," in *Proc. ACM MobiCom*, 2010, pp. 1–12.

[17] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ: Wiley-Interscience, 2006, vol. 2.

[18] W.-T. Chen and N.-F. Huang, "The strongly connecting problem on multihop packet radio networks," *IEEE Trans. Commun.*, vol. 37, no. 3, pp. 293–295, Mar. 1989.

[19] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc, "Power consumption in packet radio networks," *Theor. Comput. Sci.*, vol. 243, no. 1–2, pp. 289–305, 2000.

[20] D. Berend, M. Segal, and H. Shpungin, "Energy and lifetime efficient connectivity in wireless ad-hoc networks," *Ad Hoc Sensor Wireless Netw.*, vol. 10, no. 1, pp. 61–87, 2010.

[21] H. Shpungin, "Energy efficient online routing in wireless ad hoc networks," in *Proc. IEEE SECON*, 2011, pp. 64–72.

[22] H. Zhang and J. C. Hou, "Asymptotic critical total power for k-connectivity of wireless networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, pp. 347–358, Apr. 2008.

[23] P. Hall, *Introduction to the Theory of Coverage Processes*. New York: Wiley, 1988.

[24] R. B. Ellis, X. Jia, and C. Yan, "On random points in the unit disk," *Random Struct. Algor.*, vol. 29, pp. 14–25, 2006.

[25] N. Along and J. H. Spencer, *The Probabilistic Method*. Hoboken, NJ: Wiley-Interscience, 2008.

[26] R. Guérin and A. Orda, "Computing shortest paths for any number of hops," *IEEE/ACM Trans. Netw.*, vol. 10, no. 5, pp. 613–620, Oct. 2002.

[27] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," *IEEE Pers. Commun.*, vol. 7, no. 5, pp. 28–34, Oct. 2000.

[28] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A distributed algorithm for minimum-weight spanning trees," *Trans. Program. Lang. Syst.*, vol. 5, no. 1, pp. 66–77, 1983.

[29] M. D. Penrose, "The longest edge of the random minimal spanning tree," *Ann. Appl. Probab.*, vol. 7, no. 2, pp. 340–361, 1997.

[30] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proc. ACM MobiCom*, 1999, pp. 174–185.

[31] M. Gastpar and M. Vetterli, "On the capacity of wireless networks: The relay case," in *Proc. IEEE INFOCOM*, 2002, vol. 3, pp. 1577–1586.

[32] A. Keshavarz-Haddad and R. H. Riedi, "Bounds for the capacity of wireless multihop networks imposed by topology and demand," in *Proc. ACM MobiHoc*, 2007, pp. 256–265.

[33] A. E. F. Clementi, P. Penna, and R. Silvestri, "Hardness results for the power range assignmet problem in packet radio networks," in *Proc. RANDOM-APPROX*, 1999, pp. 197–208.

[34] A. E. F. Clementi, G. Huiban, P. Penna, G. Rossi, and Y. Verhoeven, "Some recent theoretical advances and open questions on energy consumption in ad-hoc wireless networks," in *Proc. ARACNE*, 2002, pp. 23–38.

[35] R. Ramanathan and R. Hain, "Topology control of multihop wireless networks using transmit power adjustment," in *Proc. IEEE INFOCOM*, 2000, pp. 404–413.

[36] E. L. Lloyd, R. Liu, M. V. Marathe, R. Ramanathan, and S. S. Ravi, "Algorithmic aspects of topology control problems for ad hoc networks," *Mobile Netw. Appl.*, vol. 10, no. 1–2, pp. 19–34, 2005.

[37] G. Calinescu, S. Kapoor, A. Olshevsky, and A. Zelikovsky, "Network lifetime and power assignment in ad hoc wireless networks," in *Proc. ESA*, 2003, pp. 114–126.

[38] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *Proc. IEEE INFOCOM*, 2000, pp. 22–31.

[39] I. Kang and R. Poovendran, "Maximizing network lifetime of broadcasting over wireless stationary ad hoc networks," *Mobile Netw. Appl.*, vol. 10, no. 6, pp. 879–896, 2005.

Hanan Shpungin (M'09) received the B.Sc., M.Sc., and Ph.D. degrees in computer science from Ben-Gurion University of the Negev, Beer-Sheva, Israel, in 2000, 2005, and 2010, respectively.

Currently, he is a Post-Doctoral Fellow with the Networks Research Group, Department of Computer Science, University of Calgary, Calgary, AB, Canada. His research interests revolve around combinatorial optimization in wireless ad hoc networks. He is particularly interested in topology control, fundamental capacity bounds, distributed beamforming, and scheduling.

Zongpeng Li received the B.E. degree in computer science and technology from Tsinghua University, Beijing, China, in 1999, the M.S. degree in computer science and Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2001 and 2005, respectively.

Since August 2005, he has been with the Department of Computer Science, University of Calgary, Calgary, AB, Canada. His research interests are in computer networks, particularly in network optimization, multicast algorithm design, network game theory, and network coding.