Analysis and Design of a Novel Randomized Broadcast Algorithm for Scalable Wireless Networks in the Interference Channels

Scott C.-H. Huang, *Member, IEEE*, Shih Yu Chang, *Member, IEEE*, Hsiao-Chun Wu, *Senior Member, IEEE*, and Peng-Jun Wan

***Abstract*—In this paper, we study the minimum-latency broadcast scheduling problem in the probabilistic model. We establish an explicit relationship between the tolerated transmission-failure probability and the latency of the corresponding broadcast schedule. Such a tolerated transmission-failure probability is calculated in the strict sense that the failure to receive the message at any single node will lead to the entire broadcast failure and only if all nodes have successfully received the message do we consider it a success. We design a novel broadcast scheduling algorithm such that the broadcast latency is evaluated under such a strict definition of failure. The latency bound we derive is a strong result in the sense that our algorithm achieves a low broadcast latency under this rather strict broadcast-failure definition. Simulation results are also provided to justify our derived theoretical latency bound.**

***Index Terms*—Wireless networks, scheduling, randomized algorithm.**

I. Introduction

BROADCAST is a classical problem that arises in many applications of communications. For multi-hop wireless networks, in particular, broadcast is a very time-consuming operation because it involves tedious contention, collision, and congestion. The *latency* of a broadcast algorithm is its executing time, as defined in [1] or [2]. How to reduce the broadcast latency can be deemed quite challenging. There exist many different approaches to reduce the latency. Scheduling is one of the most effective approaches. By carefully scheduling each node's message transmission, one can often avoid both interference and collision.

Broadcast has been extensively studied by researchers. Empirical studies regarding the effectiveness of the broadcasting schemes, in terms of collision-free delivery, number of retransmissions, and latency, are presented in [3]. The multi-input multi-output (MIMO) system was also utilized to enhance the broadcast efficiency [4]. A novel and simple efficient broadcast protocol, namely *Activecast*, could operate in the absence of knowledge about neighborhood nodes [5].

There are a lot of theoretical studies regarding broadcast latency since it is a crucial measure for evaluating the wireless broadcast networks. *Minimum-latency broadcast scheduling methods* have been extensively studied in the existing literature. Although there are already a plethora of algorithms aimed at reducing the broadcast latency (such as [1], [2], [6]–[14]) and the upper- or lower-bounds of the latency have been studied as well (such as [15]–[18]), the network models therein are not practical. They all adopt the *deterministic* model as follows. The network topology is modeled as a graph. The vertices represent the nodes/devices in the network. An edge exists between two nodes if and only if there is a communication link between them. If two or more nodes transmit simultaneously to a single node, then collision happens and the message will be corrupted. However, if one node transmits to another node without collision, then the reception is guaranteed. The deterministic model does not reflect the probabilistic behavior of wireless communication in the real world. In these works, they all assume similar network models and one improves another by more and more elaborate mathematical analyses. Although these works already achieved very good latency bounds (such as [11]), the aforementioned problem of the adopted impractical network models still remains unsolved. Apart from those deterministic algorithms in [1], [2], [6]–[18], a randomized broadcast algorithm was proposed in [19], which essentially achieved the same latency bound presented in [8]. However, the channel interference level has not been addressed and evaluated in [19] and hence the probabilistic behavior of the successful message reception at each node is not considered therein.

The objective of this work is to study the minimum latency problem using a practical network model, in which the message transmission failure is quantified in a probabilistic manner. In this paper, we make a new attempt to design the effective scheduling scheme for practical wireless networks in the interference environments. The SINR (signal-to-interference-and-noise ratio) model was used for the topology control [20] and the data aggregation [21] and is more realistic

Manuscript received December 7, 2008; revised July 26, 2009 and November 30, 2009; accepted January 27, 2010. The associate editor coordinating the review of this paper and approving it for publication was I. Habib.

This work was supported by Information Technology Research Award for National Priorities (NSF-ECCS 0426644) from the National Science Foundation, the National Science Foundation-Louisiana Pilot Fund, and the Office of Naval Research-DEPSCoR grant.

S. C.-H. Huang is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: shuang@cs.cityu.edu.hk).

H.-C. Wu is with the Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA 70803, USA (e-mail: wu@ece.lsu.edu).

S. Y. Chang is with the Department of Computer Science, National Tsing Hua University, Taiwan (e-mail: shihyuch@cs.nthu.edu.tw).

P.-J. Wan is with the Department of Computer Science, Illinois Institute of Technology, Chicago, Illinois, USA (e-mail: wan@cs.iit.edu).

Digital Object Identifier 10.1109/TWC.2010.07.081579

than the aforementioned deterministic model. Throughout our discussion, the success probability of the message reception at any node is considered and such an probability depends on the interference level in the adopted SINR model. The higher the SINR, the higher this success probability. Consequently, an explicit relationship between the tolerated transmission-failure probability and its latency bound of the corresponding broadcast schedule can be established. We present this relationship in Theorem 1. The tolerated transmission-failure probability here is calculated in the strict sense that even a single message-transmission failure will result in the entire broadcast failure. Only if all nodes have received the message successfully do we call it a successful broadcast. Our novel scheduling algorithm is designed in a very careful way that, even under such a strict definition of failure, a low broadcast latency can still be achieved thereby.

The rest of this paper is organized as follows. In Section II, we present the network model, the crucial parameters and the assumptions to be used in later sections, particularly the tessellation and the coloring techniques. We present our novel randomized broadcast scheduling algorithm in Section III. A concrete example is given in Section IV. In Section V, we focus on a very important parameter γ, defined in Section II, and discuss how to appropriately select it to make a fully-connected network regime. Numerical results and simulation outcomes are given in Section VI to evaluate our proposed method. Concluding remarks are drawn in Section VII. All proofs for the underlying theorems in this paper are collected in the appendix.

Notations: The sets of all real numbers and all natural numbers are denoted by $\mathbb{R}$ and $\mathbb{N}$, respectively.

II. Network Model and Problem Statement

Let V be the set of nodes within the network of interest in a two-dimensional Euclidean space, and each node is associated with an identical transmission power level P. According to physics, we know that if a node $u \in V$ (transmitter) is transmitting with power level P, the theoretical received-signal strength P_v at another node $v \in V$ (receiver) is given by

$$P_v = \frac{P}{d(u,v)^\alpha}, \tag{1}$$

where $d(u,v)$ is the distance between u, v and α is a constant called the *path-loss exponent*. A commonly assumed path-loss exponent α is greater than **two** [22]. To some extent, such an assumption is facilitated for a static network. In fact, this static assumption can be relaxed a bit as follows. Our proposed algorithm can still work as long as $a \le \alpha \le b$ for the bounds $a, b > 2$. Thus, the theoretical interference strength I_v is

$$I_v = \sum_{w \in T - \{u\}} \frac{P}{d(v,w)^\alpha}. \tag{2}$$

In Eq. (2), $v \notin V$ and $T \subset V$ is the set of the nodes scheduled to transmit in the current time slot, in which only u alone is the transmitter and all other nodes are interferers. The SINR at v is thus given by

$$SINR_v = \frac{P_v}{N + I_v}, \tag{3}$$

where N is the background noise power. The probability $Pr[v]$ characterizes that a node v receives a message successfully in a time slot such that

$$Pr[v] = 1 - Ae^{-B \cdot SINR_v}, \tag{4}$$

where A, B are both positive constants dependent on the real environment. Also, if $Pr[v]$ is too small (i.e. smaller than a threshold value p_κ), we regard it as transmission failure. Here p_κ is called the *transmission failure threshold probability*, which manifests the probabilistic nature of the successful broadcast task.

Network Model: Given a set of nodes V and the system parameters A, B, P, N, α, p_κ, we define the *relaxed threshold radius* r_κ as

$$r_\kappa = \sqrt[\alpha]{\frac{PB}{(1+\gamma)N \ln \frac{A}{1-p_\kappa}}}, \tag{5}$$

where $\gamma > 0$ is a constant called the *relaxation factor*[1]. The motivation for defining r_κ is as follows. Let v be a receiver and suppose $I_v = 0$. Then the maximum distance between any transmitter and v is $r_{\max} = \sqrt[\alpha]{\frac{PB}{N \ln \frac{A}{1-p_\kappa}}}$ according to Eqs. (1), (3), (4). Thus, r_κ is actually $r_{\max}$ relaxed by a factor $(1+\gamma)$. We define the *transmission graph* G_T as $G_T = \big(V, E_T(r_\kappa)\big)$ where $E_T(r_\kappa) = \{(u,v) | \overline{uv} < r_\kappa\}$. Note that the relaxed threshold radius r_κ as well as the edge set $E_T(r_\kappa)$ depend on the relaxation factor γ. We assume that G_T is fully connected by carefully choosing γ. Justifications for this assumption as well as how to choose γ are given in Section V.

A. Problem Formulation

Given a set of nodes V, a source $s \in V$, and system parameters A, B, P, N, α, p_κ, we suppose that the graph G_T (which is only related to the system parameters A, B, P, N, α, p_κ) is fully connected by properly selecting γ and every node knows its own location. Here we consider a simple traffic model, where a packet is generated at source s, and s broadcasts the generated packet to the entire network through all neighboring nodes in the transmission graph G_T. More complex traffic models may be considered for our future work. Time is assumed to be discrete and we use time slots to represent it throughout this paper. Each node in the network is equipped with a clock and the clocks of all nodes are synchronized. Each node is able to read a variable, denoted by $Time$, representing its clock value. An *admissible broadcast schedule* can be represented as a collection of the subsets $\{U_1, U_2, \ldots\}$ satisfying the following requirements: (i) for all i, $U_i \subset V$ represents the set of nodes scheduled to transmit in time slot i; (ii) a node cannot be scheduled to transmit unless it has already received successfully from a neighboring node in G_T in an earlier time slot; (iii) in the end, all nodes in V receive the broadcasted message successfully at least once. The *latency* of an admissible broadcast schedule is the first

[1] γ can be determined according to other given parameters. How to determine γ is quite complicated, so it is not presented here. The selection method for γ will be presented in Section V.

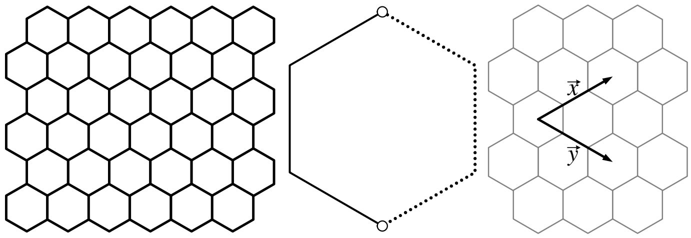


Fig. 1. (a) Hexagonal tessellation; (b) one hexagon; (c) $\vec{x}$ and $\vec{y}$.

time slot for (iii) to hold, i.e. the first time slot that all nodes receive the broadcasted message at least once. Obviously, there will be different latencies when different admissible broadcast schedules are employed. The objective of the *minimum-latency broadcast scheduling* (MLBS) problem is *to find an admissible broadcast schedule minimizing its latency.*

In order to facilitate the solution to this MLBS problem, now we introduce the important terms, concepts, and methods that will be extensively used throughout this paper.

B. Underlying Terms, Concepts, and Methods

Maximal Independent Sets (MIS): A subset $S \subset V$ is an *independent set* of G if the vertices in S are pairwise non-adjacent, and a *maximal independent set* (MIS) S of G is an independent set of G while no proper superset containing S is an independent set of G. Any vertex ordering $v_1, v_2, \ldots, v_n$ of V induces an MIS S in the following first-fit manner. Initially, $S = \{v_1\}$. For $i = 2$ up to $i = n$, add v_i to S if v_i is not adjacent to any vertex in S. Details of MIS can be found in [23].

Connected Dominating Sets: A subset $S \subset V$ is a *dominating set* of V if each vertex in V is adjacent to at least one vertex in S. If $v \in V$ is adjacent to $u \in S$, then v is *dominated by* u or equivalently u is a *dominator* of v. Note that for any set V, its MIS is naturally a dominating set of V. If a dominating set is connected, then it is a *connected dominating set*.

Hexagonal Tessellation and Colorings: A tessellation of the plane is a way of partitioning it into identical (or similar) pieces. A regular hexagonal tessellation is partitioning the entire plane into hexagons, as shown in Fig. 1 (a). Each hexagon is half open, half closed, without both the topmost and the bottommost points, as shown in Fig. 1 (b). We can color this tessellation in various ways. Without loss of generality, in this paper, we will choose the following coloring method.

Method 1: (**Coloring Method**) We introduce a new coloring method here for the future use in broadcast scheduling. Given a hexagonal tessellation and a natural number k, let r denote the radius of a hexagon. Define the vectors $\vec{x} = (3\sqrt{3}r/2, 3r/2)$ and $\vec{y} = (3\sqrt{3}r/2, -3r/2)$ as shown in Fig. 1 (c). The lengths of $\vec{x}$ and $\vec{y}$ are both $3r$. Repeat the following process for all $1 \leq i \leq 3k^2$. Randomly pick an uncolored hexagon whose center is located at $\vec{h}$. Color all the hexagons with color i whose centers are located at $\vec{h} + ak\vec{x} + bk\vec{y}$ for some $a, b \in \mathbb{Z}$.

An Example of Method 1: Take $k = 3$ for example. Suppose that we randomly pick up a hexagon H_0 and color it as $i = 1$. According to our coloring method, we should color the hexagons whose centers are located at $\vec{h} + 3a\vec{x} + 3b\vec{y}$ for all $a, b \in \mathbb{Z}$. We repeat this coloring task until $i = 3k^2 = 27$, by which we can color all hexagons. We add a lemma for Method 1 as follows.

Lemma 1: Method 1 results in a $3k^2$-coloring. Hexagons attributed by the same color are separated by at least $(3k-2)r$.

(*Proof.*) This follows directly from the coloring method. The distance between the centers of any two hexagons with the same color is at least $3kr$ by definition. It immediately arises from that the distance between any two hexagons with the same color is $3kr - 2r = (3k - 2)r$. □

Note that the procedure of Method 1 is not unique. There are still many different ways to color these hexagons, and we may just consider one of them without loss of generality. For more details, see [24].

III. Novel Randomized Broadcast Scheduling Algorithm

In order to combat the minimum-latency broadcast scheduling problem and to provide a low-latency solution, we propose a novel randomized broadcast scheduling algorithm here. Our proposed algorithm involves two phases, namely (1) *virtual backbone tree construction* and (2) *broadcast scheduling*. Intuitively, in Phase (1), in addition to the broadcast tree construction, we try to separate all transmission nodes using coloring techniques so that the interference can be controlled. In Phase (2), we schedule transmissions according to the broadcast tree containing the coloring information acquired in Phase (1). The details of Phase (1) are described in Algorithm 1.

Algorithm 1 Virtual Backbone Tree Construction (Phase 1)

Require: Connected input graph $G_T = (V_T, E_T)$

Ensure: Virtual Backbone Tree $VBT(G_T) = (V_{VBT}, E_{VBT})$

Fix $V_{VBT} = V_T$. Set $E_{VBT} = \emptyset$.
Construct the Breadth First Search (BFS) tree $BFS(G_T)$ for G_T by applying [25].
Sort all nodes $v \in V_T$ according to their hop-distances to s in ascending order. Divide V_T into layers $L_0, L_1, \ldots, L_R$, where L_i contains those which are i-hop from s and R is the radius of G_T with respect to s.
Let $BLACK$ denote the MIS of G_T induced by such a node ordering. The nodes in $BLACK$ are referred to as the *black nodes* or the *dominators*. ($BLACK$ is also a dominating set of G_T). The nodes not belonging to $BLACK$ are called *white nodes* instead.
for $i \leftarrow 1$ to $R - 1$ **do**
    For each $v \in L_{i+1} \cap BLACK$, find its parent $p(v)$ in $BFS(G_T)$. Color $p(v)$ blue.
    Find $p(v)$'s dominator $d_{p(v)}$ at either layer i or layer $i - 1$.
    Add $\big(p(v), v\big)$ and $\big(d_{p(v)}, p(v)\big)$ to E_{VBT}.
end for
for all remaining white nodes u **do**
    Find u's dominator d_u and add $\big(u, d_u\big)$ to E_{VBT}.
end for

Note that the layers of the BFS tree and those of the virtual backbone tree may be different. The employment of the minimum connected dominating sets as the virtual backbone has been adopted extensively in the literature (cf. [2], [26]). Hence, we adopt this standard approach in Phase (1).

In order to present our proposed randomized broadcast scheduling algorithm later on, we need to introduce several crucial parameters here. Define

$$\begin{cases} r_1 = \sqrt[\alpha]{\frac{8P}{\gamma N}\left(\frac{2}{\alpha-2}+\frac{1}{\alpha-1}+3\right)} \\ r_2 = \max\left(2r_\kappa,\ \sqrt[\alpha]{\frac{24P}{\gamma N}\left(\frac{2}{\alpha-2}+\frac{1}{\alpha-1}+3\right)}\right) \\ \Pi_1 = 3\left\lceil\frac{2}{3}\left(\frac{r_1}{r_\kappa}+2\right)\right\rceil^2, \quad \Pi_2 = 3\left\lceil\frac{2}{3}\left(\frac{r_2}{r_\kappa}+2\right)\right\rceil^2, \\ \Pi = \Pi_1 + \Pi_2 \end{cases} \tag{6}$$

Accordingly, we can tessellate the plane into half-open half-closed hexagons of radius $\frac{r_\kappa}{2}$, and apply Method 1 to carry out a Π_1-coloring with $k = \left\lceil\frac{2}{3}\left(\frac{r_1}{r_\kappa}+2\right)\right\rceil$. We use C_1 to denote this coloring. Note that C_1 maps a hexagon to an integer that represents a color. However, since each point must be located in a hexagon, we can also view C_1 as a mapping from a point on the plane to an integer by attributing a point v to the color of the hexagon containing v. C_1 can therefore be characterized as $C_1 : \mathbb{R}^2 \rightarrow \mathbb{N}$. Then, we apply Method 1 again to carry out another Π_2-coloring with $k = \left\lceil\frac{2}{3}\left(\frac{r_2}{r_\kappa}+2\right)\right\rceil$. We use C_2 to denote this coloring.

Consequently, we can undertake Phase (2) in our proposed scheme now. The broadcast scheduling algorithm based on the constructed virtual backbone tree is described in Algorithm 2.

Algorithm 2 Randomized Broadcast Scheduling (Phase 2)

```
repeat the following ⌈ln(n/ε) / ln(1/(1−p_κ^2))⌉ times in parallel for
  each node v that either is the source or has just successfully received the message.
    if v is black then
        Wait until Time mod Π ≡ 0.
        Schedule v to transmit to all of its child(ren) at Time mod Π ≡ C_1(v).
    end if
    if v is blue then
        Wait until Time mod Π ≡ Π_1.
        For each black child w of v, v transmits to w in the virtual backbone tree at Time mod Π ≡ Π_1 + C_2(w).
    end if
until done
```

The latency (time complexity) associated with our proposed algorithm (Algorithm 2) can be evaluated using the following new theorem.

Theorem 1: In Algorithm 2 (we refer to as Alg. 2 in brief), the probability that all nodes have successfully received the message by time

$$\frac{\Pi}{p_\kappa^2}\left[R + \ln(n/\epsilon) + \sqrt{2R\ln(n/\epsilon) + \ln^2(n/\epsilon)}\right] \tag{7}$$

is at least $1 - 2\epsilon$, where R is the hop-distance from the source s to the farthest node in the network. □

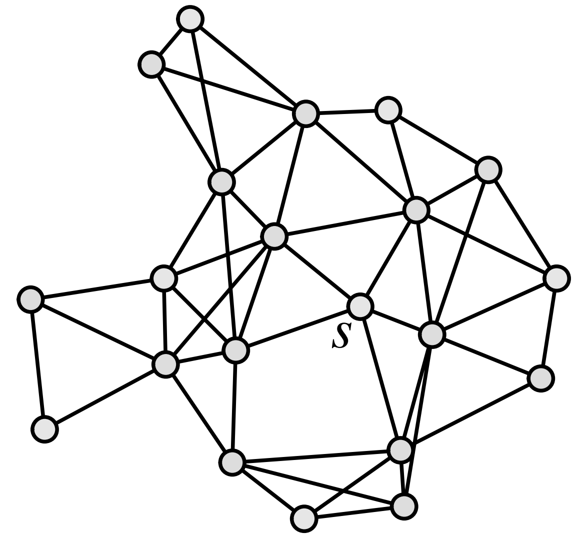


Fig. 2. Topology of G_T.

Note that if we replace Π in Eq. (7) by $\Pi_1 + \Pi_2$ given by Eq. (6), we obtain a latency expression in terms of parameters $A, B, P, N, n, R, \alpha, \epsilon, p_\kappa, \gamma$ only. Theorem 1 establishes an explicit relationship between the tolerated transmission-failure probability 2ϵ and the latency of the corresponding broadcast schedule we introduce in this section. Note that the tolerated transmission-failure probability is calculated in the strict sense that even a message transmission-failure at any single node is regarded as a whole broadcast failure. Only if all nodes have successfully received the message do we call it a success. Theorem 1 tells us that this probability is at least $1 - 2\epsilon$. The complete proof for Theorem 1 is provided in the appendix.

IV. Illustration of Our Proposed Algorithm

We present an example here to illustrate the detailed procedure of our proposed algorithm given in Section III. Suppose that the node set V_T, the source node $s \in V_T$, and the system parameters $A, B, P, N, \alpha, p_\kappa$ are all given and we have already chosen γ properly such that G_T is fully connected as depicted in Fig. 2. According to Algorithm 1 (Phase 1), we need to construct $VBT(G_T)$. First, we start to construct $BFS(G_T)$ by applying the standard BFS algorithm given by [25]. Now we sort V_T according to their hop-distances to s in ascending order. Then, we construct the MIS, called $BLACK$, with this node ordering as follows. We start from layer L_0, which contains s only. We add s to $BLACK$ and move on to layer 1. Since all nodes at layer 1 are adjacent to s, none of them can be added to $BLACK$ and layer 1 is done. Likewise, now we work on layer 2. In a similar manner, we then work on layer 3. $BLACK$ thus contains seven black nodes as depicted in Fig. 3 (a). Those nodes which are not labeled black are white. We omit some graphical illustrations and the associated details due to the figure limitation.

Based on $BLACK$, we may embark on constructing the virtual backbone tree. We start from layer 2 since layer 1 does not have any black node. For each black node at layer 2, we find its parent node at layer 1 in the BFS tree, color it as blue, and connect them. For each blue node, we find a black

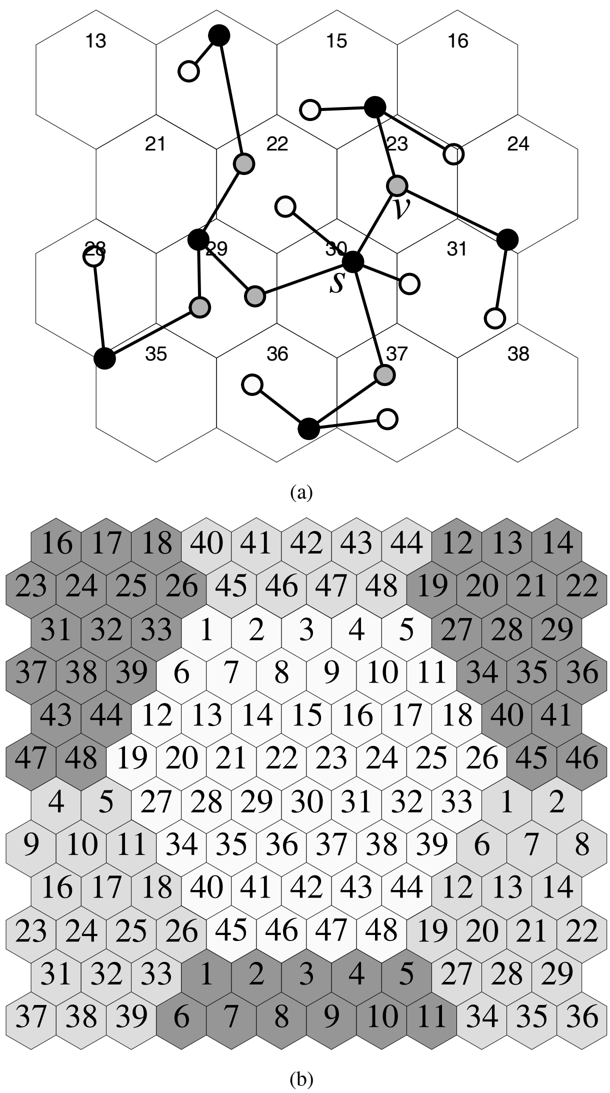


Fig. 3. C_1: 48-coloring ($\Pi_1 = 48$). Actual nodes are shown in (a) only and the overall coloring is shown in (b).

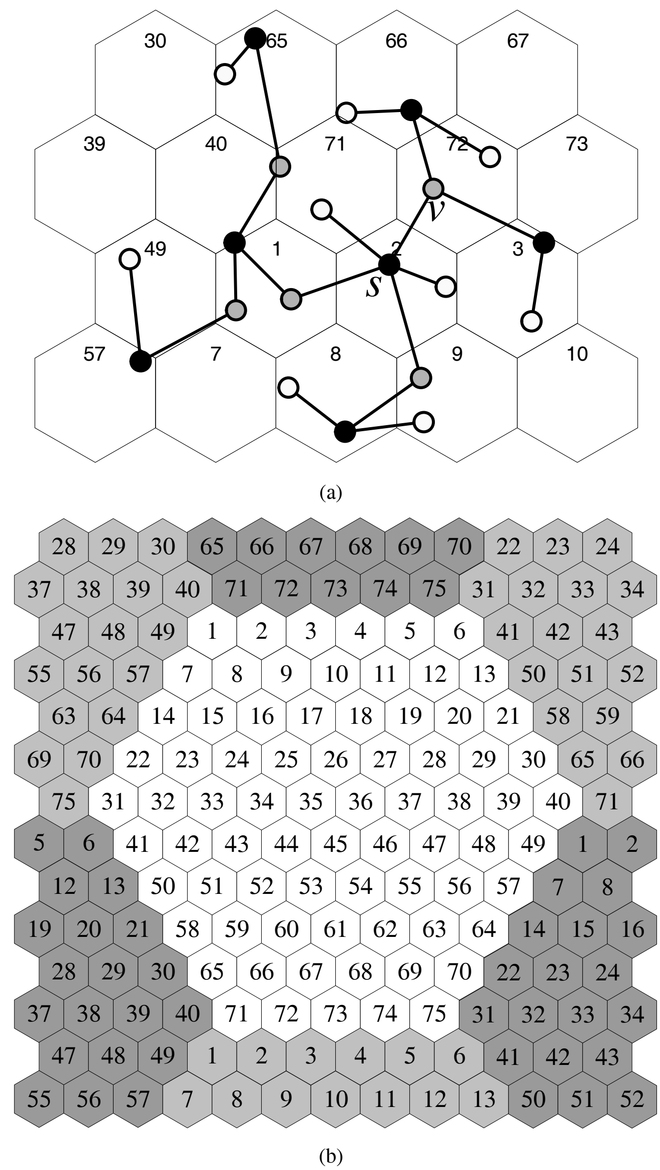


Fig. 4. C_2: 75-coloring ($\Pi_2 = 75$). Actual nodes are shown in (a) only and the overall coloring is shown in (b).

node in either L_1 or L_0 (in this case L_0) in the BFS tree. We repeat this process at layer 3, find the corresponding blue nodes at layer 2 in the BFS tree, and connect them. We repeat this procedure until all layers in the BFS tree have been visited (only up to layer 3 in this example). Finally we connect the remaining white nodes. Ultimately, the virtual backbone tree is thus constructed, as depicted in Fig. 3 (a). We omit some figures and certain details about the intermediate results due to the figure limitation.

Now, according to the definitions of Π_1 and Π_2 in Section III, we calculate

$$\left\lceil \frac{2}{3}\left(\frac{r_1}{r_\kappa}+2\right)\right\rceil = 4, \quad \left\lceil \frac{2}{3}\left(\frac{r_2}{r_\kappa}+2\right)\right\rceil = 5.$$

Therefore, $\Pi_1 = 48$, $\Pi_2 = 75$, and $\Pi = 123$. The colorings C_1, C_2 are shown in Fig. 3 and Fig. 4, respectively. In each figure, part (a) shows the positions of the nodes while part (b) shows the overall coloring. We do not show the nodes in part (b) simply to maintain the legibility. Note that C_1 and C_2 are constructed independently and their colors have nothing to do with each other. Take s for example; $C_1(s) = 30$ while $C_2(s) = 2$. Now, for Phase (2), according to Algorithm 2, we group 123 time slots altogether as a unit and all black nodes (that have successfully received the message) are scheduled to transmit according to their C_1-colors. In this example, there are 7 black nodes with C_1-colors 14, 15, 24, 29, 30, 35, 36. Therefore, they should transmit in these time slots (colors) repeatedly for every period consisting of 123 time slots. Blue nodes are scheduled to transmit according to their black child(ren)'s C_2-color(s). In this example, there are 5 blue nodes. Take node v for example; v has two black children with C_2-colors 3 and 66, respectively; v should therefore transmit its successfully received message in the 51^{st} and 114^{th} time slots (3+48 and 66+48) repeatedly for every period consisting of 123 time slots. Each black or blue node should start the transmission once it has successfully received the message, and repeat such transmissions for $\lceil \frac{\ln(n/\epsilon)}{\ln(1/(1-p_\kappa^2))} \rceil$ times.

V. Appropriate Selection of the Relaxation Factor γ

As discussed in Section II, the relaxation factor γ plays an important role in our assumption of conditionally-full connection for any wireless network. We assume that we can always make G_T fully connected by choosing γ appropriately. Here we will explain why this assumption actually makes sense in the MLBS problem. Let us first revisit Eq. (4) in Section II and find the minimum SINR to make the probability of successful reception exceed the threshold probability. In other words, find $SINR_v$ such that $Pr[v] \geq p_\kappa$. According to Eq. (4), it yields

$$1 - Ae^{-B \cdot SINR_v} \geq p_\kappa. \tag{8}$$

Therefore, $e^{-B \cdot SINR_v} \leq \frac{1-p_\kappa}{A}$, and $SINR_v \geq \frac{1}{B} \ln \frac{1-p_\kappa}{A}$. Since both P and N are assumed to be constants, if there is no interference involved at all, $SINR_v$ only depends on the transmission distance r from the transmitting node to the receiving node. From Eq. (8), we have

$$SINR_v = \frac{P}{r^\alpha N} \geq \frac{1}{B} \ln \frac{1-p_\kappa}{A}, \text{ and } r \geq \sqrt[\alpha]{\frac{PB}{N \ln \frac{A}{1-p_\kappa}}}. \tag{9}$$

Thereby, we define the *threshold radius* $r_{\kappa 0}$ as

$$r_{\kappa 0} = \sqrt[\alpha]{\frac{PB}{N \ln \frac{A}{1-p_\kappa}}}. \tag{10}$$

According to Eqs. (9) and (10), in order to make the reception successful, the transmission distance must be less than or equal to $r_{\kappa 0}$. Note that Eq. (10) is derived upon when there is no interference. In the MLBS problem, it means that no concurrent transmission is allowed. Therefore, it becomes a trivial problem. In order to make this problem non-trivial, we must accommodate the concurrent transmissions to some extent by *relaxing* the threshold radius a little more. We define the relaxed threshold radius as *the maximum radius that makes $P_r[v]$ greater than the threshold probability, provided that the overall interference is γN*. In other words, we can tolerate up to γN interference totally and still guarantee our scheduling algorithm's effectiveness. The above reasons lead us to define the relaxed threshold radius in Eq. (5). The assumption that G_T is conditionally-fully connected is actually very reasonable for the following reasons. If G_T is not fully connected for any $\gamma > 0$, then no algorithm that allows concurrent transmissions can lead to an admissible broadcast schedule. However, within the connected sub-graph containing the source node, the MLBS problem can still be pursued and our algorithm can be effective.

Here we present how to choose γ appropriately. We choose γ subject to the following criteria: (1) G_T is fully connected; (2) the overall latency is minimized.

Theorem 1 tells us that the latency is of order $O(\Pi R)$, in which $\Pi = O\big((\frac{r_1}{r_\kappa} + 2)^2 + (\frac{r_2}{r_\kappa} + 2)^2\big)$. Moreover, $r_1 = O(\gamma^{-\frac{1}{\alpha}})$, $r_2 = O(\gamma^{-\frac{1}{\alpha}})$, and $r_\kappa = O\big((1+\gamma)^{-\frac{1}{\alpha}}\big)$. Note that R may be influenced by γ as well. Although there is no explicit relationship between them, generally speaking, R is proportional to $\frac{1}{r_\kappa}$ if nodes are distributed evenly, and the latency is therefore $O\big((1+\gamma)^{\frac{1}{\alpha}}\big)$. Consequently,

$$\Pi = O\left(\left(\frac{1+\gamma}{\gamma}\right)^{\frac{1}{\alpha}}\right)(1+\gamma)^{\frac{1}{\alpha}} = O\left(\left(1+\frac{1}{\gamma}\right)^{\frac{1}{\alpha}}(1+\gamma)^{\frac{1}{\alpha}}\right). \tag{11}$$

We can see that the latency tends to infinity when γ tends to either 0 or ∞. The minimum latency value can therefore be determined according to elementary calculus as follows. First we determine the range along the real line such that G_T is fully connected in this range. We then express the latency as a function of γ and seek its minimum within this range.

VI. Numerical Evaluation of the Proposed Scheme

In this section, we evince the numerical results according to our latency formula addressed in Sections III and V where we demonstrated the relationship between the broadcasting latency and the system parameters A, B, p_κ and ϵ. In our numerical evaluations, we fix $\frac{P}{N} = 5$ and $\gamma = 0.5$ (except for Fig. 6 (b)). Crucial parameters are listed in (12) on the next page for different simulation results. Two sets of results are demonstrated here, namely the theoretical latency upper-bounds given by Eq. (7) and the actual latency values from the simulations. Note that each latency value from the simulation is obtained from the average over 50 random network topologies subject to the same number of nodes.

In Fig. 6(a), we delineate the transmission latency for different p_κ, subject to the requirements that the number of viable nodes in the network ranges from 200 to 1200 ($A = 0.75, B = 0.5, \gamma = 0.5, \epsilon = 0.1$). When p_κ is higher, the latency is relatively lower. The reason is as follows. When p_κ is larger, there are fewer edges (communication links) in G_T (network topology). Therefore, according to our algorithm, the number of concurrent transmissions increases. As a result, the latency becomes lower.

We also compare our proposed algorithm with the *leveled probabilistic broadcasting* (LPB) method in [27] for $p_\kappa = 0.3$ and $p_\kappa = 0.7$. According to [27], we select the numerical parameters as $p_1 = 1.0$, $p_2 = 0.5$, $p_3 = 0.2$, $p_4 = 0$, where p_1, p_2, p_3, p_4 are the forwarding probabilities for the four groups classified in [27] subject to the node connectivity degree. The reason why we choose this set of parameters is to achieve the shortest delay as illustrated by Fig. 11 of [27]. The latency results including our derived theoretical upper-bounds, the actual latencies from the simulation using our proposed algorithm, and the actual latencies from the simulation using the LPB method are all depicted in Fig. 5. Based on Fig. 5, our algorithm exhibits a lower latency than the LPB scheme in the same condition when the interference effect of wireless channels is considered (this effect was completely ignored by [27]). It is observed that the smaller p_κ is (corresponding to the smaller SINR), the larger latency performance margin of our proposed algorithm will be over the LPB method. The simulation results justify that our algorithm is more robust than the LPB method especially when the channel is noisy.

Figure 6(b) illustrates the relationship between the latency and the relaxation factor γ. The number of nodes still ranges from 200 to 1200 ($A = 0.75$, $B = 0.5$, $p_\kappa = 0.7$, $\epsilon = 0.1$). Note that the latency decreases with the increase of γ. The reason for this fact is that the number of nodes which could be

Crucial System Parameters

parameter	description
A	system parameter defined in Eq. (4)
B	system parameter defined in Eq. (4)
p_κ	transmission failure threshold probability
ϵ	half of the failure probability defined in Theorem 1

(12)

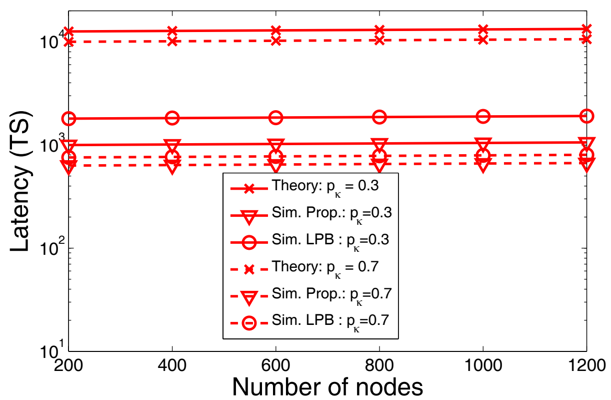
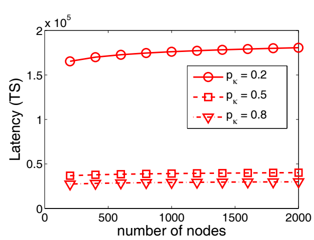
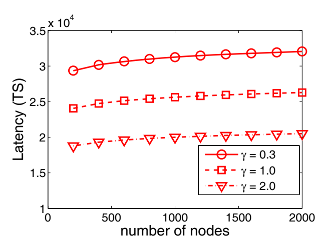


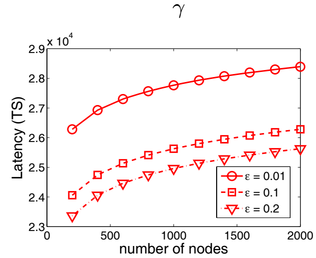
Fig. 5. The latency performance comparison among our derived theoretical upper-bounds (denoted by "Theory"), our proposed randomized broadcast method (denoted by "Sim. Prop.") and the leveled probabilistic broadcasting (LPB) method (denoted by "Sim. LPB") for $p_\kappa = 0.3$ and $p_\kappa = 0.7$.



(a) Transmission latency for different p_κ



(b) Transmission latency for different γ



(c) Transmission latency for different ϵ

Fig. 6. Transmission latency versus its system parameters.

covered by a broadcasting action becomes larger if the value of γ increases. Hence, the latency to accomplish the broadcasting task is getting smaller when the value of γ becomes larger.

Figure 6(c) illustrates the relationship between the latency and the broadcast failure ratio ϵ. The number of nodes ranges from 200 to 1200 ($A = 0.75$, $B = 0.5$, $p_\kappa = 0.7$, $\gamma = 0.5$). Note that ϵ is evaluated in a very strict manner that even if a single node fails to receive the message, the whole broadcast is considered being failed. As can be seen in Fig. 6(c), the lower ϵ, the larger the theoretical transmission latency upper-bound. The increase of ϵ reduces the probability of successful reception for each node. However, in our simulation, failure cases are not counted towards the latency. For this reason, in our simulation, the latencies for different ϵ values are almost the same and the three curves for different ϵ values almost completely overlap as shown in Fig. 6(c).

VII. Conclusion

In this paper, we study the minimum-latency broadcast scheduling problem in the realistic probabilistic model and establish a new explicit relationship between the tolerated transmission-failure probability and the overall latency of the broadcast schedule. We also design a novel algorithm which can dynamically adjust the latency subject to the condition of the interference channel. Our algorithm and analysis could be deemed as the first attempt to combat the low-latency broadcast problem for the scalable wireless networks in the interference channels.

Appendix

In this appendix, we will prove Theorem 1. Before the complete proof of Theorem 1 can be achieved, we facilitate some useful lemmas at first.

Lemma 2: The distance between any two concurrently transmitting black nodes is at least $r_\kappa + r_1$ while the distance between any two concurrently transmitting blue nodes is at least $r_2 - r_\kappa$.

(*Proof.*) According to the C_1-coloring and Lemma 1, the distance between any two hexagons of the same color is at least $2r_1/r_\kappa + 2$ times as large as the radius of a hexagon, which is $r_\kappa/2$. It means that any two hexagons of the same color are separated by at least $r_\kappa + r_1$. Since each hexagon can contain at most one black node and the concurrently transmitting black nodes must have the same color, we have already shown that the distance between any two concurrently transmitting black nodes is at least $r_\kappa + r_1$. For blue nodes, according to the C_2-coloring and Lemma 1, the distance between any two hexagons of the same color is at least $2r_2/r_\kappa + 2$ times as large as the radius of a hexagon, which is $r_\kappa/2$. It results in that any two hexagons of the same color are separated by at least $r_\kappa + r_2$. Since blue nodes are scheduled subject to their black children's colors, and it essentially means that the distance between their black children is at least $r_\kappa + r_2$. Moreover, the distance between a blue node and its children is at most r_κ. Therefore, we conclude that the distance between the concurrently transmitting blue nodes is at least $r_\kappa + r_2 - 2r_2 = r_\kappa - r_2$. □

Lemma 3: The total interference experienced at any receiving node at any time is at most γN.

(*Proof.*) Since black and blue nodes will not be scheduled for concurrent transmission, we will prove this lemma by distinguishing the following two cases.

(Case 1: *Total interference at a blue or white receiving node - black nodes as senders*) According to Lemma 2, we know that at any time the distance between any two simultaneously transmitting black nodes is at least $r_\kappa + r_1$. Moreover, let u be a black sender and v be its targeted receiver, and thus there will be no other concurrent sender whose distance to v is less than r_1. Now, let us pick up a targeted receiver v and consider its concentric circles of radii $r_1, 2r_1, 3r_1, \ldots$. Here we use $A(r_a, r_b)$ to denote the annulus between two concentric circles of radii r_a and r_b ($r_a < r_b$). We define $A(r_a, r_b)$ to be an inner-closed and outer-open ring (i.e. $A(r_a, r_b)$ contains the circle of radius r_a but does not contain the circle of radius r_b). Now we focus on the region $A\big((i-1)r_1, ir_1\big)$ for all $i \geq 2$ and consider the senders scheduled to transmit simultaneously therein at a fixed time. Let M_i be the number of these senders in $A\big((i-1)r_1, ir_1\big)$. We know that the distance between any two black nodes is at least $r_\kappa + r_1$. Therefore, according to the geometry, we get

$$M_i < \frac{4(2i-1)r_1(r_\kappa + 2r_1)}{(r_\kappa + r_1)^2}. \tag{13}$$

Since the distance between v and any point within $A\big((i-1)r_1, ir_1\big)$ is at least $(i-1)r_1$, the cumulative interference caused by other senders in $A\big((i-1)r_1, ir_1\big)$ is thus bounded by $M_i \frac{P}{\left((i-1)r_1\right)^\alpha}$ and the overall interference I_{total} experienced at v caused by all other concurrent black senders on the entire plane is bounded by $I_{total} \leq \sum_{i=2}^{\infty} M_i \frac{P}{\left((i-1)r_1\right)^\alpha}$. Here i starts from 2 because, except for the intended sender, no other interfering senders are within the disk centered at v with radius r_1. Invoking Eq. (13), we obtain

$$I_{total} \leq \sum_{i=2}^{\infty} \frac{4(2i-1)r_1(r_\kappa + 2r_1)}{(r_\kappa + r_1)^2} \frac{P}{\left((i-1)r_1\right)^\alpha}. \tag{14}$$

Now, let q be $q = \frac{r_1}{r_\kappa}$, in which case Eq. (14) becomes

$$\begin{aligned} I_{total} &< \sum_{i=2}^{\infty} \frac{4(2i-1)q(2q+1)}{(q+1)^2} \cdot \frac{P}{(i-1)^\alpha r_1^\alpha} \\ &= \frac{4q(2q+1)}{(q+1)^2} \cdot \frac{P}{r_1^\alpha} \sum_{i=2}^{\infty} \frac{2i-1}{(i-1)^\alpha}. \end{aligned} \tag{15}$$

It can be easily verified that $\sum_{i=2}^{\infty} \frac{2i-1}{(i-1)^\alpha} = \sum_{i=2}^{\infty} \left[\frac{2(i-1)}{(i-1)^\alpha} + \frac{1}{(i-1)^\alpha}\right] = 2\sum_{i=2}^{\infty} \frac{1}{(i-1)^{\alpha-1}} + \sum_{i=2}^{\infty} \frac{1}{(i-1)^\alpha} = 2\sum_{j=1}^{\infty} \frac{1}{j^{\alpha-1}} + \sum_{j=1}^{\infty} \frac{1}{j^\alpha}$. We obtain by elementary calculus that

$$\sum_{j=1}^{\infty} \frac{1}{j^\alpha} \leq \frac{1}{\alpha-1} + 1 \Rightarrow \sum_{i=2}^{\infty} \frac{2i-1}{(i-1)^\alpha} \leq \frac{2}{\alpha-2} + \frac{1}{\alpha-1} + 3. \tag{16}$$

Invoking Eq. (16) and $r_1 = \sqrt[\alpha]{\frac{8P}{\gamma N}\left(\frac{2}{\alpha-2} + \frac{1}{\alpha-1} + 3\right)}$ given by Eq. (15), we thus get

$$I_{total} < \frac{4q(2q+1)}{(q+1)^2} \cdot \frac{\gamma N}{8}.$$

Besides, we find

$$\frac{4q(2q+1)}{(q+1)^2} = \frac{8q^2+4q+4}{(q+1)^2} < \frac{8q^2+16q+16}{(q+1)^2} = 8.$$

Finally, we can conclude that $I_{total} < \gamma N$ as desired for this case.

(Case 2: *Total interference at a black receiving node - blue nodes as senders*) Similarly, according to Lemma 2, we know that at any time the distance between any two simultaneously transmitting blue nodes is at least $r_2 - r_\kappa$. Let u be a blue sender and v be its targeted receiver, and there will be no other concurrent blue sender whose distance to v is less than r_2. Now, let us pick up a targeted receiver v and consider its concentric circles of radii $r_2, 2r_2, 3r_2, \ldots$. Let M'_i be the number of these senders belonged to $A\big((i-1)r_2, ir_2\big)$. We know that the distance between any two blue nodes is at least $r_2 - r_\kappa$. We may apply simple geometry to get $M'_i < 4\frac{(4i-2)r_2^2 - (2i-1)r_2 r_\kappa}{(r_2 - r_\kappa)^2}$. Similarly, it yields

$$\begin{aligned} I_{total} &\leq \sum_{i=2}^{\infty} M'_i \frac{P}{\left((i-1)r_2\right)^\alpha} I_{total} \\ &\leq \sum_{i=2}^{\infty} 4\frac{(4i-2)r_2^2 - (2i-1)r_2 r_\kappa}{(r_2 - r_\kappa)^2} \cdot \frac{P}{\left((i-1)r_2\right)^\alpha}. \end{aligned} \tag{17}$$

Let q' be $q' = \frac{r_2}{r_\kappa}$ and the above expression (17) becomes

$$\begin{aligned} I_{total} &< 4\frac{(2q'-1)q'}{(q'-1)^2} \cdot \frac{P}{r_2^\alpha} \sum_{i=2}^{\infty} \frac{2i-1}{(i-1)^\alpha} \\ &\leq 4\frac{(2q'-1)q'}{(q'-1)^2} \cdot \frac{P}{r_2^\alpha}\Big(\frac{2}{\alpha-2} + \frac{1}{\alpha-1} + 3\Big), \end{aligned} \tag{18}$$

where

$$r_2 \overset{\text{def}}{=} \max\left(2r_\kappa, \sqrt[\alpha]{\frac{24P}{\gamma N}\Big(\frac{2}{\alpha-2} + \frac{1}{\alpha-1} + 3\Big)}\right).$$

It actually implies that $q' \geq 2$ and $\frac{P}{r_2^\alpha}\left(\frac{2}{\alpha-2} + \frac{1}{\alpha-1} + 3\right) \leq \frac{\gamma N}{24}$. Therefore, we get $I_{total} < 4\frac{(2q'-1)q'}{(q'-1)^2} \times \frac{\gamma N}{24}$. Observe that $\frac{(2q'-1)q'}{(q'-1)^2}$ is monotonically decreasing in $(1, \infty)$. Since $q' \leq 2$, we conclude that $\frac{(2q'-1)q'}{(q'-1)^2} \leq 6$ and therefore

$$I_{total} < 4 \cdot 6 \cdot \frac{\gamma N}{24} = \gamma N.$$

□

Lemma 4: At any moment, the probability of successful message reception of any targeted receiver is at least p_κ.

(*Proof.*) According to Lemma 3, the total interference at any time at any node v is at most γN. Since the sender is at most r_κ away from v, the received signal power at node v is at least

$$\frac{P}{r_\kappa^\alpha} = \frac{(1+\gamma)N}{B} \ln \frac{A}{1-p_\kappa}. \tag{19}$$

The interference plus noise at node v is at most $\gamma N + N = (1+\gamma)N$. Therefore, $SINR_v \geq \frac{1}{B} \ln \frac{A}{1-p_\kappa}$. Since $Pr[v] = 1 - Ae^{-B \cdot SINR_v}$, we conclude that $Pr[v] \geq p_\kappa$. □

Lemma 5: When Alg. 2 finishes, the probability that all nodes have received the message successfully is at least $1-\epsilon$.

(*Proof.*) In the **repeat** loop of Alg. 2, each node successfully receives the message with probability at least p_κ. It means that each black node successfully receives the message from its grand-parent with probability at least p_κ^2. Therefore, $Pr($not all nodes received successfully) $= Pr(\exists v \neq s$ s.t. v didn't receive successfully and at least one of v's neighbors received successfully) $\leq \sum_{v\neq s} Pr(v$ didn't receive successfully and at least one of v's neighbors received successfully) $\leq \sum_{v\neq s} Pr(v$ didn't receive successfully $|$ at least one of v's neighbors received successfully) $\leq n(1-p_\kappa^2)^{\lceil \frac{\ln(n/\epsilon)}{\ln(1/(1-p_\kappa^2))} \rceil} \leq \epsilon$. Therefore, Pr(all nodes received successfully) $\geq 1-\epsilon$. □

Lemma 6: We modify the loop in Alg. 2 such that each node repeats forever instead of just $\lceil \frac{\ln(n/\epsilon)}{\ln(1/(1-p_\kappa^2))} \rceil$ times. We call this modified algorithm as *Alg. 2**. Then, if we run Alg. 2* for time

$$\frac{\Pi}{p_\kappa^2}\Big[R + \ln(n/\epsilon) + \sqrt{2R\ln(n/\epsilon) + \ln^2(n/\epsilon)}\Big],$$

the probability that all nodes have successfully received the message is at least $1-\epsilon$.

(*Proof.*) Let us define $T(\epsilon)$ as

$$T(\epsilon) = \frac{1}{p_\kappa^2}\bigg(R + \ln\frac{n}{\epsilon} + \sqrt{2R\ln\frac{n}{\epsilon} + \ln^2\frac{n}{\epsilon}}\bigg). \tag{20}$$

The ultimate goal of this proof is to show that $Pr\big($all nodes received successfully at $\Pi T(\epsilon)\big) > 1-\epsilon$ in Alg. 2*.

Let T_v be a random variable denoting the time by which node v has successfully received the message. We group Π time slots altogether as a unit starting from $Time = 0$ and call it a *round*. First, we will show that

$$Pr\big[T_v > \Pi T(\epsilon)\big] < \epsilon/n. \tag{21}$$

For a node $v \in V$, let D_j be a random variable representing the length of a shortest path from v to the set of the informed nodes (the nodes having successfully received the message) at round j. Initially, we have $D_0 \leq R$. From Lemma 4, we get

$$Pr\big[D_j - D_{j+1} = 1 | D_j \neq 0\big] \geq 1 - p_\kappa^2. \tag{22}$$

Now, on the other hand, $Pr[D_{T(\epsilon)} > 0]$ is the probability that v has not received the message by time $T(\epsilon)\Pi$. Therefore,

$$\begin{aligned} Pr\big[T_v > \Pi T(\epsilon)\big] &= Pr\big[D_{T(\epsilon)} > 0\big] \\ &= Pr\left[\sum_{j=0}^{T(\epsilon)-1}(D_j - D_{j+1}) < D_0\right] \\ &\leq Pr\left[\sum_{j=0}^{T(\epsilon)-1}(D_j - D_{j+1}) < R\right]. \end{aligned} \tag{23}$$

Define a binary random variable $\chi_j = D_j - D_{j+1}$. By Lemma 4, we get $Pr[\chi_j = 1 | D_j \neq 0] \geq 1 - p_\kappa^2$. Thus, the above expression (23) corresponds to the probability that the sum of such $T(\epsilon)$ variables does not exceed R. Using the Chernoff bound (pp. 18 in [28]) with $\mu = p_\kappa^2 T(\epsilon)$ and $\delta = 1 - \frac{R}{p_\kappa^2 T(\epsilon)}$, we get

$$\begin{aligned} Pr\big[D_j - D_{j+1} = 1 | D_j \neq 0\big] &< e^{-\frac{\mu\delta^2}{2}} \\ &< \exp\Big[-\frac{1}{2}p_\kappa^2 T(\epsilon)\cdot\Big(1 - \frac{R}{p_\kappa^2 T(\epsilon)}\Big)^2\Big]. \end{aligned} \tag{24}$$

From Eq. (20), we can get

$$\Big(1 - \frac{R}{p_\kappa^2 T(\epsilon)}\Big)^2 = \frac{2}{p_\kappa^2 T(\epsilon)}\ln\frac{n}{\epsilon}. \tag{25}$$

Therefore, (24) becomes $\exp\Big[-\frac{1}{2}p_\kappa^2 T(\epsilon)\cdot\Big(1 - \frac{R}{p_\kappa^2 T(\epsilon)}\Big)^2\Big] = \exp\Big[-\frac{1}{2}p_\kappa^2 T(\epsilon)\frac{2}{p_\kappa^2 T(\epsilon)}\ln\frac{n}{\epsilon}\Big] = \frac{\epsilon}{n}$, and thus we have proved Eq. (21). Consequently, we get

$$\begin{aligned} &Pr\Big(\text{in Alg. 2*, not all nodes received successfully} \\ &\quad \text{by } \Pi T(\epsilon)\Big) \\ &\quad = Pr\Big[\max_v T_v > \Pi T(\epsilon)\Big] \leq \sum_v Pr\big[T_v > \Pi T(\epsilon)\big] \\ &\quad < n\cdot\frac{\epsilon}{n} = \epsilon. \end{aligned}$$

Finally, we have proved

$$\begin{aligned} &Pr\Big(\text{in Alg. 2*, all nodes received successfully} \\ &\quad \text{by } \Pi T(\epsilon)\Big) \\ &\quad > 1 - \epsilon. \end{aligned}$$

□

Proof of Theorem 1. We combine Lemmas 5 and 6 to achieve

$$\begin{aligned} &Pr\Big(\text{in Alg.2, not all nodes received successfully by } \Pi T(\epsilon)\Big) \\ &\quad \leq Pr\Big(\text{in Alg. 2, not all nodes received successfully}\Big) + \\ &\quad Pr\Big(\text{in Alg.2*, not all nodes received successfully} \\ &\qquad \text{by } \Pi T(\epsilon)\Big) \\ &\quad < \epsilon + \epsilon = 2\epsilon. \end{aligned}$$

Therefore, $Pr\Big($in Alg. 2, all nodes received successfully by $\Pi T(\epsilon)\Big) > 1 - 2\epsilon$.

□

REFERENCES

[1] I. Chlamtac and O. Weinstein, "The wave expansion approach to broadcasting in multihop radio networks," *IEEE Trans. Commun.*, vol. 39, pp. 426–433, 1991.

[2] R. Gandhi, S. Parthasarathy, and A. Mishra, "Minimizing broadcast latency and redundancy in ad hoc networks," in *Proc. 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'03)*, pp. 222–232.

[3] J.-P. Sheu, P.-K. Hung, and C.-S. Hsu, "Scheduling of broadcasts in multihop wireless networks," in *The Handbook of Ad Hoc Wireless Networks*. Boca Raton, FL: CRC Press, Inc., 2003, pp. 483–495.

[4] L. C. Wang and C. J. Yeh, "On the performance of zero-forcing beamforming for MIMO broadcast systems," in *Proc. IEEE International Symposium on Antennas and Propagation*, 2007, pp. 4108–4111.

[5] V. Paruchuri and A. Durresi, "Broadcast protocol for energy-constrained networks," *IEEE Trans. Broadcast.*, vol. 53, no. 3, pp. 112–119, 2007.

[6] S. Basagni, I. Chlamtac, and D. Bruschi, "A mobility-transparent deterministic broadcast mechanism for ad hoc networks," *IEEE/ACM Trans. Networking*, vol. 7, no. 6, pp. 799–807, 1999.

[7] I. Chlamtac and S. Kutten, "On broadcasting in radio networks–problem analysis and protocol design," *IEEE Trans. Commun.*, vol. 33, pp. 1240–1246, 1985.

[8] D. R. Kowalski and A. Pelc, "Centralized deterministic broadcasting in undirected multi-hop radio networks," in *Proc. 7th Int'l Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX-RANDOM'04)*, pp. 171–182.

[9] I. Gaber and Y. Mansour, "Centralized broadcast in multihop radio networks," *J. Algorithms*, vol. 46, no. 1, pp. 1–20, 2003.

[10] M. Elkin and G. Kortsarz, "An improved algorithm for radio networks," 2005, an earlier version appeared in *SODA'05*.

[11] L. Gąsieniec, D. Peleg, and Q. Xin, "Faster communication in known topology radio networks," in *Proc. 24th Annual ACM Symposium on Principles of Distributed Computing (PODC'05)*, pp. 129–137.
[12] D. R. Kowalski and A. Pelc, "Optimal deterministic broadcasting in known topology radio networks," to appear in *Distributed Computing*.
[13] A. Dessmark and A. Pelc, "Tradeoffs between knowledge and time of communication in geometric radio networks," in *Proc. 13th annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'01)*, pp. 59–66.
[14] D. Bruschi and M. Del Pinto, "Lower bounds for the broadcast problem in mobile radio networks," *Distributed Computing*, vol. 10, no. 3, pp. 129–135, 1997.
[15] M. Elkin and G. Kortsarz, "Polylogarithmic additive inapproximability of the radio broadcast problem," in *Proc. 7th Int'l Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX-RANDOM'04)*, 2004.
[16] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg, "A lower bound for radio broadcast," *J. Computer and System Sciences*, vol. 43, no. 2, pp. 290–298, 1991.
[17] E. Kushilevitz and Y. Mansour, "An $\Omega(D\log(N/D))$ lower bound for broadcast in radio networks," *SIAM J. Computing*, vol. 27, pp. 702–712, 1998.
[18] B. S. Chlebus, L. Gąsieniec, A. Gibbons, A. Pelc, and W. Rytter, "Deterministic broadcasting in unknown radio networks," in *Proc. Symposium on Discrete Algorithms*, 2000, pp. 861–870.
[19] R. Bar-Yehuda, O. Goldreich, and A. Itai, "On the time-complexity of broadcast in multi-hop radio networks: an exponential gap between determinism and randomization," *J. Computer and System Sciences*, vol. 45, no. 1, pp. 104–126, 1992.
[20] T. Moscibroda, R. Wattenhofer, and A. Zollinger, "Topology control meets SINR: the scheduling complexity of arbitrary topologies," in *Proc. 7th ACM Int'l Symposium on Mobile Ad Hoc Networking and Computing*, 2006, pp. 310–321.
[21] R. Zheng and R. Barton, "Toward optimal data aggregation in random wireless sensor networks," in *Proc. 26th IEEE Int'l Conference on Computer Communications (Infocom'07)*, pp. 249–257.
[22] P. Gupta and P. R. Kumar, "Capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, 2000.
[23] D. B. West, *Introduction to Graph Theory*, 2nd edition. Upper Saddle River, NJ: Prentice Hall, 2001.
[24] R. A. Leese, "A unified approach to the assignment of radio channels on a regularhexagonal grid," *IEEE Trans. Veh. Technol.*, vol. 46, no. 4, pp. 968–980, 1997.
[25] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd edition. Cambridge, MA: MIT Press/McGraw Hill, 2003.
[26] S. C.-H. Huang, P.-J. Wan, X. Jia, and H. Du, "Low-latency broadcast scheduling schemes in ad hoc networks," in *1st Int'l Conference on Wireless Algorithms, Systems, and Applications (WASA'06)*, Aug. 2006.
[27] Q. Zhang and D. P. Agrawal, "Performance evaluation of leveled probabilistic broadcasting in MANETs and wireless sensor networks," *Simulation*, vol. 81, no. 8, pp. 533–545, 2005.
[28] P. Erdős and J. Spenser, *Probabilistic Methods in Combinatorics*. Academic Press, 1974.

Scott Chih-Hao Huang (M'09) received his BS degree from National Taiwan University, and his PhD degree from the University of Minnesota, Twin Cities. He is currently with the Computer Science Department, City University of Hong Kong. His research interests include wireless ad hoc/sensor network, security, communication theory, algorithms, as well as combinatorial optimization.

Shih Yu Chang (M'06) received the B.S.E.E. degree from National Taiwan University, Taipei, Taiwan, R.O.C., in 1998 and the M.S.E.E. degree from University of Southern California in 2001 and the Ph.D. degree in electrical and computer engineering from the University of Michigan at Ann Arbor in 2006.

Since August 2006, he has been a faculty member of the Department of Computer Science, National Tsing Hua University, Taiwan. His research interests include wireless communications and wireless networks. Dr. Chang is very active in IEEE COMSOC society. He has been on the technical program committee for ICC 2008, GLOBECOM 2008-2009, IWCMC 2008. He also constantly participates in technical paper reviews for numerous IEEE journals in communications, computer science and networking, etc.

Hsiao-Chun Wu (M'00-SM'05) received a B. S. E. E. degree from National Cheng Kung University, Taiwan, in 1990, and the M. S. and Ph. D. degrees in electrical and computer engineering from University of Florida, Gainesville, in 1993 and 1999 respectively. From March 1999 to January 2001, he had worked for Motorola Personal Communications Sector Research Labs as a Senior Electrical Engineer. Since January 2001, he has joined the faculty in Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, Louisiana, USA. From July to August 2007, Dr. Wu had been a visiting assistant professor at Television and Networks Transmission Group, Communications Research Centre, Ottawa, Canada. From August to December 2008, he was a visiting associate professor at Department of Electrical Engineering, Stanford University, California, USA.

Dr. Wu has published more than 120 peer-refereed technical journal and conference articles in electrical and computer engineering. His research interests include the areas of wireless communications and signal processing. Dr. Wu is an IEEE Senior Member and an IEEE Distinguished Lecturer. He currently serves as an Associate Editor for IEEE TRANSACTIONS ON BROADCASTING, IEEE SIGNAL PROCESSING LETTERS, IEEE COMMUNICATIONS MAGAZINE, INTERNATIONAL JOURNAL OF COMPUTERS AND ELECTRICAL ENGINEERING, JOURNAL OF INFORMATION PROCESSING SYSTEMS, PHYSICAL COMMUNICATION, and JOURNAL OF THE FRANKLIN INSTITUTE. He used to serve as an Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He has also served for numerous textbooks, IEEE/ACM conferences and journals as the technical committee, symposium chair, track chair, or the reviewer in signal processing, communications, circuits and computers.

Peng-Jun Wan received the BS degree from Tsinghua University, the MS degree from the Chinese Academy of Science, and the PhD degree from the University of Minnesota. He is currently a professor of computer science in the Department of Computer Science, Illinois Institute of Technology, Chicago. His research interests include wireless networks, and algorithm design and analysis.