

# Distributed Broadcast Scheduling in Mobile Ad Hoc Networks with Unknown Topologies

Guang Tan, Stephen A. Jarvis, James W. J. Xue, and Simon D. Hammond  
Department of Computer Science, University of Warwick,  
Coventry, CV4 7AL, United Kingdom  
{gtan, saj, xuewj2, sdh}@dcs.warwick.ac.uk

## Abstract

*Broadcasting is a fundamental communication task in mobile ad hoc networks, and minimizing broadcasting time (or latency) is crucial to the performance of many applications. Extensive studies have been conducted on the minimization of broadcasting time in the context of radio networks, which are usually modeled as general graphs. In this paper, we consider how to achieve this goal with distributed algorithms based on a more realistic (and restricted) network model. We propose a randomized algorithm that completes broadcasting in  $O(D \log(n/D) + \log^2 n)$  time, where  $n$  is the number of nodes in the network and  $D$  the eccentricity (maximum distance from the source node to any other node). Compared with a previous optimal algorithm that achieves the same result for general networks, our algorithm obviates the need to know the network eccentricity  $D$  beforehand. We also propose a deterministic broadcasting algorithm that works in  $O(n)$  time, which is in contrast with the best known result of  $O(n \log^2 D)$  for general networks.*

## 1 Introduction

In a mobile ad hoc network (MANET), a set of wireless mobile nodes communicate with each other using radio transmission. Without relying on any pre-existing infrastructure, nodes in the networks self-organize into a network for communication. The self-organizing and wireless nature of ad hoc networks introduces a number of challenging research issues in the design of network protocols. One of the major challenges comes from the potential interference caused by the simultaneous transmission of nearby nodes: if two nodes transmit a message to some common receiving node at the same time, the message will be corrupted or lost. In such a case, we say that a *collision* has occurred

at the receiving node. This characteristic gives rise to many interesting problems for some fundamental tasks of network communication.

In this paper we study the distributed broadcast scheduling problem in ad hoc networks. In broadcasting, a source node sends a message to all the other nodes in the network. We want to minimize the broadcast latency, which is defined as the time taken by the message to reach all the nodes in the network. A small broadcasting latency is crucial to network performance in many aspects and is required by a variety of applications, such as real-time multimedia broadcasting and military communications based on ad hoc sensor networks. We assume that individual nodes have no *a priori* knowledge about network topology, as is often the case in practical self-organized networks.

There have been extensive studies on minimizing broadcasting latency in the context of radio packet networks. Most previous work employs a general graph model, based on which complexity issues are explored and efficient algorithms are designed. However, as highlighted by [13], the general graph model is not an accurate description of networks that may arise in real-world settings; the authors of [13] propose a restricted class of graphs, called *planar point graphs*, that prove to be a more accurate model of ad hoc networks. A planar point graph consists of a set of points on a plane; each of these points is associated with a transmission range, and a *directed* edge exists between two nodes if the Euclidean distance between the two nodes is less than or equal to the transmission range of the source node. As a result of this restriction, some complexity results concerning the broadcast time on general graphs do not necessarily hold for the planar point graphs. In this paper we apply a further restriction to the planar point graphs that reflects the physical characteristics of wireless devices. In our network model, called the *MANET graph*, there exists a maximum transmission range  $r_{max}$  and a minimum transmission range  $r_{min}$  among all the nodes in the network. The constant  $r_{max}$  is determined by the upper limit of all nodes' power levels and is therefore greater than zero. Likewise,



an ordinate pair  $(x_u, y_u) \in \mathbf{R}^2$ , and a maximum transmission range  $r_u$ . A directed edge  $(u, v) \in E$  if and only if  $v$  is within the  $r_u$  of node  $u$ . We assume that for all  $u \in V$ ,  $r_u \in [r_{min}, r_{max}]$ , where  $r_{min} \in \mathbf{R}^+$  and  $r_{max} \in \mathbf{R}^+$  are the minimum and maximum transmission ranges, respectively. We denote by  $D$  the *eccentricity* of  $G$ , which is equal to the maximum distance from the source to any other node. Initially, a node does not have any prior knowledge about the network topology or its immediate neighbors; it only knows its coordinates, the transmission range limits  $r_{min}$  and  $r_{max}$ , and the network size  $n$  (or a linear upper bound for  $n$ ). Notice that our deterministic algorithm does not need any knowledge of  $n$ . For notational convenience, we assume, without loss of generality, that  $n$  is a power of 2; moreover, when we use the expression  $O(\log(N_1/N_2))$ , we mean  $O(\max(\log(N_1/N_2), 1))$ , to avoid the case  $N_1 = N_2$ .

In the broadcasting problem, we assume that all nodes in the network are reachable from the source node. In the gossiping problem, we assume that the graph is strongly connected.

Time is divided into discrete time steps. All nodes start simultaneously and have access to a globally synchronized clock. A node can be in either transmitting or receiving mode, but not both. A message is successfully received by a node  $v$  at time step  $t$  (also referred to as node  $v$  becoming *active*) if and only if exactly one of  $v$ 's in-neighbors transmits at  $t$ ; if more than one in-neighbors of  $v$  transmit a message simultaneously, then a *collision* occurs and  $v$  receives nothing. As assumed in most models, a node cannot distinguish between the case of collision and the case of no in-neighbors transmitting at all.

We partition the plane into equal-sized *squares* with side length equal to  $kr_{min}/\sqrt{2}$ , where  $k$  is an integer constant such that  $kr_{min}/\sqrt{2} \geq 2r_{max}$ ; each such square further consists of  $k^2$  equal-sized *cells*, whose side length is  $r_{min}/\sqrt{2}$ . Within each square, the cells are consecutively numbered as  $1, 2, \dots, k^2$  from bottom left to top right (see Figure 1 for an example). A node  $v$  belongs to a certain cell can calculate its cell number as:  $cell(v) = k \lfloor \sqrt{2}y_v/r_{min} \rfloor + \lfloor \sqrt{2}x_v/r_{min} \rfloor$ , where  $x' = x_v - \lfloor \sqrt{2}x_v/r_{min} \rfloor kr_{min}/\sqrt{2}$  and  $y' = y_v - \lfloor \sqrt{2}y_v/r_{min} \rfloor kr_{min}/\sqrt{2}$ . Let the set of nodes within cell  $i$  be  $C_i$ .

In our broadcasting algorithms, node  $v$  is allowed to transmit only at time  $k^2i + cell(v)$  for  $i = 0, 1, \dots, n$ . This rule, together with the properties of wireless transmission, implies that (1) two nodes in different cells within the same square are always collision-free; (2) two nodes with the same cell numbers but in different squares are collision-free (because their distance is greater than  $r_{max}$ ); and (3) nodes in the same cell form a complete subgraph  $K_{|C_i|}$  and collisions only occur among these nodes.

Notice that with the above plane partitioning, there generally exists an idle interval of length at most  $k^2$  between a node's consecutive actions, including receiving and transmitting. With this idle interval, the total transmission time from the source to some destination node can be at most  $k^2$  multiplied by the transmission time in a network without the space partitioning.

## 4 Randomized broadcasting

In this section, we first develop a method to estimate the number of nodes in each cell in  $O(\log^2 n)$  time. The knowledge of  $|C_i|$  will help us reduce the broadcast time from  $O(D \log n + \log^2 n)$ , the running time of the original randomized algorithm for general graphs, to  $O(D \log(n/D) + \log^2 n)$ . For succinctness of discussion, in this section we do not consider the constant factor  $k^2$  when calculating broadcasting time. When we refer to a time step in the context of a certain cell, we mean  $k^2$  time steps in the system – the omitted factor  $k^2$  does not affect the complexity results.

### 4.1 $O(\log^2 n)$ -time approximation algorithm for $|C_i|$

Since nodes in different cells have different transmission times, the approximation of  $|C_i|$  can be done independently within each cell. We first introduce a simple randomized algorithm to elect a leader in  $C_i$ , which will serve as a coordinator for the estimating procedure. This algorithm essentially uses the idea of Bar-Yehuda et al.'s decay procedure [2]: in  $\log n + 1$  consecutive time steps, which we call a *decay round*, nodes in the network transmit with probability  $1, 2^{-1}, 2^{-2}, \dots, 2^{-\log n}$  at the step  $0, 1, \dots, \log n$ , respectively. It can be proved that a decay round yields a successful transmission with a constant probability. Using this procedure as a building block, Algorithm 1 runs in  $O(\log^2 n)$  time to produce a successful transmission with probability at least  $1 - n^{-1}$ .

After the execution of Algorithm 1, with probability at least  $1 - n^{-1}$ , some node  $v$  has either received a message containing its own coordinates or has received no message at all, while each of the other nodes has received a message with different content from its own coordinates. Therefore, an agreement can be reached among all nodes that  $v$  is the leader of  $C_i$ .

Next, we present an algorithm (Algorithm 2) for the approximation of  $|C_i|$ . With the leader elected, we need to estimate the number of remaining nodes  $n_i = |C_i| - 1$ . For notational convenience, we assume that  $n_i = 0$  or  $n_i = 2^m$ , ( $m \geq 1$ ). (The trivial case of  $n_i = 1$  can be easily identified by the leader.)

**Theorem 1.** *Algorithm 2 runs in  $O(\log^2 n)$  time and gives an approximation for  $n_i$  that satisfies  $|n_i| < \hat{n}_i \leq 16n_i$ .*

---

**Algorithm 1** Simple Randomized Leader Election Algorithm in Cell  $i$ 


---

**Input:** Complete graph  $K_{|C_i|}$ .  
 /\*  $\log n$  decay rounds \*/  
**for**  $dr = 1$  to  $\log n$  **do**  
   **for**  $j = 0$  to  $\log n$  **do**  
 Each node  $v \in C_i$  executes the following procedure independently with probability  $2^{-j}$   
**if**  $v$  has never received a message **then**  
    $v$  transmits  $(x_v, y_v)$ ;  
**else if**  $v$  has already received some message  $(x', y')$  ( $\neq (x, y)$ ) **then**  
    $v$  transmits  $(x', y')$ .  
**end if**  
**end for**  
**end for**

---

*Proof.* From Algorithm 2 we can see that a total of  $300 \log n$  decay rounds are run, requiring  $300 \log^2 n$  time. Now suppose that at a certain time step, the leader orders the nodes in  $C_i$  to transmit with probability  $1/s$ . Let  $S$  denote the event that exactly one node transmits. Then we have  $\Pr(S) = \frac{n_i}{s} (1 - \frac{1}{s})^{n_i - 1}$ , ( $n_i \geq 2, s \geq 2$ ). Now,

$$\begin{aligned} \Pr\left(S \mid \frac{n_i}{16} < s \leq n_i\right) &= \Pr\left(S \mid \frac{n_i}{16} < s \leq \frac{n_i}{2}\right) \\ &+ \Pr\left(S \mid \frac{n_i}{2} < s \leq n_i\right) \\ &> 0 + \left(1 - \frac{1}{n_i/2}\right)^{n_i - 1} \\ &> 0.1. \end{aligned}$$

$$\begin{aligned} \Pr\left(S \mid s \leq \frac{n_i}{16}\right) &= \Pr\left(S \mid \frac{n_i}{32} < s \leq \frac{n_i}{16}\right) \\ &+ \Pr\left(S \mid \frac{n_i}{64} < s \leq \frac{n_i}{32}\right) + \dots \\ &< 16 \left(1 - \frac{1}{n_i/8}\right)^{n_i - 1} \\ &+ 32 \left(1 - \frac{1}{n_i/16}\right)^{n_i - 1} + \dots \\ &< \frac{16}{e^8} + \frac{32}{e^{16}} + \dots \\ &< 0.01. \end{aligned}$$

Let random number  $X_{(j_1, j_2]}$  denote the number of successful transmissions for  $2^{-j_1} < s \leq 2^{-j_2}$  during the  $300 \log n$  decay rounds. Then the second inequality above implies  $E[X_{(0, \log n_i - 4]}] < 300 \log n \times 0.01 = 3 \log n$ . Using the Chernoff bound, we have

$$\Pr\left(E[X_{(0, \log n_i - 4]}] \leq 6 \log n\right) > 1 - e^{-\log n} = 1 - n^{-1}.$$

---

**Algorithm 2** Approximation Algorithm for  $n_i$ 


---

1: **Input:** Complete graph  $K_{|C_i|}$  and leader node  $v$ ; array  $A[\log n]$  with all elements initialized to 0.  
 2: /\*  $300 \log n$  decay rounds \*/  
 3: **for**  $dr = 1$  to  $300 \log n$  **do**  
 4:   **for**  $j = 0$  to  $\log n$  **do**  
 5:      $v$  orders each other node to transmit with probability  $2^{-j}$   
 6:     **if** transmission succeeded **then**  
 7:        $A[j] = A[j] + 1$ ;  
 8:     **end if**  
 9:   **end for**  
 10: **end for**  
 11: **for**  $j = 0$  to  $\log n$  **do**  
 12:   **if**  $A[j] + A[j + 1] + A[j + 2] + A[j + 3] > 6 \log n$  **then**  
 13:     return  $2^{j+7}$  as the approximation of  $n_i$ .  
 14:   **end if**  
 15: **end for**  
 16: **if** all elements in  $A$  are 0 **then**  
 17:   return 0 as the approximation of  $n_i$ .  
 18: **end if**

---

On the other hand,  $E[X_{(\log n_i - 4, \log n_i)}] > 300 \log n \times 0.1 = 30 \log n$ , and hence

$$\begin{aligned} \Pr\left(E[X_{(\log n_i - 4, \log n_i)}] > 6 \log n_i\right) &> 1 - e^{-9.6 \log n_i} \\ &= 1 - n_i^{-9.6}. \end{aligned}$$

This means that in the loop of Line 11, Algorithm 2, if the sum of four consecutive  $A[j]$ 's is for the first time greater than  $6 \log n$ , then with a high probability,  $j + 3$  falls between  $(\log n_i - 4, \log n_i]$ . It follows that  $\log n_i < j + 7 \leq \log n_i + 4$ ; taking  $\hat{n}_i = 2^{j+7}$  as the approximation of  $n_i$  leads to  $n_i < \hat{n}_i \leq 16n_i$ , as desired.  $\square$

After the execution of Algorithm 2, the leader notifies all the other nodes in  $C_i$  of  $\hat{n}_i$ , and then all nodes in  $C_i$  obtain an approximation  $|\widehat{C}_i|$  for  $|C_i|$  that satisfies  $|\widehat{C}_i| \leq 16|C_i| + 1 \leq 32|C_i|$ .

#### 4.2 An $O(D \log(n/D) + \log^2 n)$ -time broadcast algorithm

Since nodes in each cell transmit independently, we only have to look at the transmission scheduling within a single cell. Upon becoming active, a node  $v$  in cell  $i$  waits until the next time step  $t_{next} = a \log |\widehat{C}_i| + i$  ( $a \in \mathbb{N}$ ), when it executes a decay round. The procedure is shown in Algorithm 3.

---

**Algorithm 3** Randomized Broadcasting Algorithm for node  $v$  in Cell  $i$ 


---

**if** node  $v$  becomes active **then**  
 $v$  waits until the next time step  $t_{next} = a \log \widehat{C}_i + i$ ,  
where  $a \in \mathbf{N}$ .  
**for**  $dr = 0$  to  $\log \widehat{C}_i$  **do**  
**for**  $j = 0$  to  $\log \widehat{C}_i$  **do**  
 $v$  transmits independently with probability  $2^{-j}$ .  
**end for**  
**end for**  
**end if**

---

**Theorem 2.** Let  $G = (V, E)$  be an  $n$ -node MANET network of eccentricity  $D$ , then with probability at least  $1 - n^{-1}$ , Algorithm 3 completes broadcasting in  $G$  in  $O(D \log(n/D) + \log^2 n)$  time.

*Proof.* We only have to show that for any single node  $v \in V$ , if we take a shortest path  $P = \langle v_0, v_1, \dots, v_l \rangle$  ( $l \leq D$ ) from the source node to  $v$ , then node  $v$  will be active after  $O(D \log(n/D))$  time with probability at least  $1 - n^{-2}$ . Since there are  $n$  nodes in the network, the union bound will imply the theorem:  $\Pr(\text{all nodes in } G \text{ become active after } O(D \log(n/D)) \text{ time}) \geq 1 - n \cdot n^{-2} = 1 - n^{-1}$ .

Before we proceed, we first introduce a lemma that gives some properties of the path and the nodes on it.

**Lemma 1.** Let  $\mathcal{C}_P = \{C_{cell(v_i)} : v_i \in P\}$  be the set of cells associated with all the nodes on  $P$ , then

- (1)  $\sum_{C \in \mathcal{C}_P} |C| < n$ ;
- (2) any  $C \in \mathcal{C}_P$  contains at most two nodes on  $P$ .

Consider a node  $v_i$  ( $1 \leq i < l$ ) on path  $P$ , and let the random variable  $Y_i$  denote the time between the activation time of  $v_i$ 's preceding node,  $v_{i-1}$ , and the activation time of  $v_i$ . Clearly,  $Y_1, Y_2, \dots, Y_l$  are independent of each other. So the time it takes to successfully deliver the message from the source to node  $v_l$  is  $Z = \sum_1^l Y_i$ . From the proof of Theorem 1, a decay round yields a successful transmission with probability at least 0.1, so for a single time step at  $v_{i-1}$ , the probability of successful transmission is at least  $\rho_i = 1/(10 \log |C_{cell(v_{i-1})}|)$ . We consider two cases separately:  $D = \Omega(\log^4 n)$  and  $D = O(\log^4 n)$ .

**Case 1.**  $D = \Omega(\log^4 n)$  It can be seen that the probability that the transmission from  $v_{i-1}$  succeeds after exactly  $k$  time steps,  $\Pr(Y_i = k) \geq \frac{1}{\rho_i} (1 - \frac{1}{\rho_i})^{k-1}$ . Let  $Y_i'$  be a random variable having geometric distribution with parameter  $\rho_i$ , then we have  $\Pr(Y_i = k) \geq \Pr(Y_i' = k)$ . Define  $Z' = \sum_1^l Y_i'$  and we have  $\Pr(v_1 \text{ becomes active after time step } t) = \Pr(Z \leq t) \geq$

$\Pr(Z' \leq t)$ . Now, if we can show that with a high probability,  $Z' < cD \log(n/D)$  for some constant  $c$ , we can establish the theorem. To do so, we use the following result that characterizes the tail distribution of the sum of geometrical random variables [7]:

**Lemma 2.** [7] Let  $X_1, \dots, X_l$  be a sequence of independent integer-valued random variables, each  $X_i$  being geometrically distributed with a parameter  $\rho_i$ ,  $0 < \rho_i < 1$ . For every  $1 \leq i \leq l$ , let  $\mu_i = 1/\rho_i$ , and assume that all  $\mu_i$  are from a set  $\Delta$ , that is,  $\Delta = \{\mu_i : 1 \leq i \leq l\}$ . If  $\sum_{i=1}^l \mu_i \leq N$ , then for every positive real number  $\beta$ ,

$$\Pr\left(\sum_{i=1}^l X_i \leq 2N + 8 \ln(|\Delta|/\beta) \sum_{z \in \Delta} z\right) \geq 1 - \beta.$$

First we establish an upper bound for  $E[Z']$ . Using the properties of geometric random variables, we have

$$\begin{aligned} E[Z'] &= \sum_1^l E[Y_i'] = \sum_{i=1}^l \frac{1}{\rho_i} \\ &= 10 \sum_1^l \log |C_{cell(v_i)}| \\ &< 50 \sum_{i=1}^l \log |C_{cell(v_i)}| \quad (\text{Theorem 1}) \\ &\leq 100 \sum_{C \in \mathcal{C}_P} \log |C| \quad (\text{Lemma 1}) \\ &= 100 \left( \sum_{C \in \mathcal{C}_P} \log |C| + \sum_{i=|C_P|+1}^D \log 1 \right) \\ &\leq 100D \log \frac{\sum_{C \in \mathcal{C}_P} |C| + (D - |C_P|)}{D} \\ &\leq 100D \log \frac{n + (D - |C_P|)}{D} \quad (\text{Lemma 1}) \\ &< 200D \log \frac{n}{D}. \end{aligned}$$

Next, we want to show that  $Z'_i$  is highly concentrated around its mean using lemma 2. Let  $X_i \equiv Y_i$ ,  $\Delta = \{1, 2, \dots, \log n\}$ ,  $\beta = n^{-2}$ , and  $N = 200D \log \frac{n}{D}$ . It is easy to see that  $\sum_{z \in \Delta} z = O(\log^2 n)$ . Applying lemma 2 yields for  $D = \Omega(\log^4 n)$  and some constant  $c$ ,

$$\begin{aligned} \Pr(v_1 \text{ becomes active after } t) &\geq \Pr\left(Z' < cD \log \frac{n}{D}\right) \\ &\geq 1 - n^{-2}. \end{aligned}$$

This inequality, combined with our argument at the beginning of the proof, proves the theorem for the case  $D = \Omega(\log^4 n)$ .

**Case 2.**  $D = O(\log^4 n)$  For networks of “small” eccentricities, we can prove the theorem by showing that our algorithm can achieve  $O(D \log n + \log^2 n)$  broadcasting time (the same as that of the algorithm in [2]), which is asymptotically equal to  $O(D \log(n/D) + \log^2 n)$ .

The analysis is similar to the first case. Let  $\rho_0 = \log n$ , then for every  $1 \leq i \leq l$ ,  $\rho_i \geq \rho_0$ . Let  $Y'_1, Y'_2, \dots, Y'_l$  be random variables having identical geometric distribution with parameter  $\rho_0$ , then we have  $\Pr(Y_i = k) \geq \Pr(Y'_i = k)$ . Define  $Z' = \sum_1^l Y'_i$  and we have  $\Pr(Z \leq t) \geq \Pr(Z' \leq t)$ . Using lemma 2 and noting that  $E[Z'] = D \log n$ , we can obtain that with probability at least  $1 - n^{-2}$ ,  $Z' = O(D \log n + \log^2 n)$ .

Combining the two above cases yields the proof of theorem 2.  $\square$

Finally, the results of theorems 1 and 2 indicate that our broadcast scheme comprising the two stages (cell size approximation and broadcasting) requires a total time of  $O(D \log(n/D) + \log^2 n)$ .

## 5 Deterministic broadcasting and gossiping

In this section, we first assign distinct labels  $1, 2, \dots, |C_i|$  to the nodes in cell  $i$ . The labels help to serialize the transmissions within a cell; coupled with the time division mechanism of the grids, the distributed broadcast can be easily scheduled. We then extend the broadcast algorithm to a simple gossiping algorithm that works in time  $O(n)$ , which is optimal.

The problem of assigning labels to nodes in a cell has been investigated in [12] in terms of network initialization for single-hop radio networks. In their work, Nakano et al. assign IDs to nodes by gradually partitioning the node set to singletons. More specifically, their protocol partitions the original node set  $C_i$  into non-empty subsets  $C_i^1$  and  $C_i^2$ . In turn,  $C_i^2$  is partitioned into two non-empty subsets  $C_i^2$  and  $C_i^3$ . In general,  $C_i^{j-1}$  is partitioned into non-empty subsets  $C_i^{j-1}$  and  $C_i^j$ . This procedure is repeated until, at some stage, some  $C_i^l$  contains a single node. This node is assigned the ID of 1 and quits the protocol. After that, the same partitioning procedure is applied to  $C_i^{l-1}$  and so on. This is repeated until all nodes have been assigned IDs from  $\{1, 2, \dots, |C_i|\}$ . According to the analysis in [12], the process of assigning labels succeeds in  $O(n)$  time with high probability.

Let the label of a node  $v$  in cell  $i$  be  $label(v)$ . Node  $v$  can determine its transmission time as follows: upon receiving the message for the first time,  $v$  waits until the next time step  $t_{next} = (a \cdot |C_i| + label(v))k^2 + i$ , where  $a \in \mathbf{N}$ , and then transmit the message. This way, the transmissions of nodes within a cell will not interfere with each other;

moreover, due to the time division mechanism, all transmissions in the network are collision-free. Since it takes at most  $2|C_i|k^2$  for the nodes in cell  $i$  to complete transmission, the total broadcasting time in the network is at most  $\sum 2|C_i|k^2 = O(n)$ . Adding the time for label assigning to the broadcasting time, we conclude that the total time to schedule the broadcast is  $O(n)$  with probability at least  $1 - n^{-1}$ .

Gossiping is another classical problem of disseminating information in networks. In gossiping, each node  $v$  in the network initially owns a message  $m_v$ , and we wish to distribute each message  $m_v$  to all nodes in the network. Gossiping is not simply  $n$  simultaneous broadcasts, because nodes can collect many messages and encapsulate them in one big packet that can be sent at a single time step. (It is commonly assumed that the message size is so small that a single packet can contain all  $n$  messages.) The fastest gossiping algorithm for general networks works in  $O(n \log^2 n)$  time [7].

With our model, the procedure of deterministic broadcasting can be directly applied to the gossiping problem. Upon receiving some new message(s),  $v$  waits until the next time step  $t_{next} = (a \cdot |C_i| + label(v))k^2 + i$ , where  $a \in \mathbf{N}$ , and then transmits a packet containing all new messages received during the waiting period. Since each message  $m_v$  can be broadcast in  $O(n)$  time, the gossiping can be completed in  $O(n)$  time.

## 6 Conclusion

In this paper, we investigate distributed broadcast scheduling in mobile ad hoc networks based on a restricted network model. By considering the inherent properties of practical wireless transmission, our model gives a more accurate description of network topologies that may arise from realistic settings compared with a traditional radio network model. The model exhibits a number of properties that yield some interesting results. We propose a randomized algorithm that completes broadcasting in  $O(D \log(n/D) + \log^2 n)$  time. Compared with a previous algorithm that achieves the same result for general networks, our algorithm does not rely on the knowledge of  $D$ ; and the previously fastest eccentricity-ignorant algorithm for general directed networks runs in  $O(n)$  time. We also propose a deterministic broadcasting algorithm that works in  $O(n)$  time, which is in contrast with the best known result of  $O(n \log^2 D)$  for general networks. A simple extension of the deterministic broadcasting algorithm yields an  $O(n)$  time gossiping algorithm.

## Acknowledgments

We are grateful to the anonymous reviewers for their feedback that helps to improve the presentation of the paper. This research was sponsored in part by grants from the NASA AMES Research Center (administrated by US-ARDSG, contract no. N68171-01-C-9012), the EPSRC (contract no. GR/R47424/01) and the EPSRC e-Science Core Programme (contract no. GR/S03058/01).

## References

- [1] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg. A lower bound for radio broadcast. *Journal of Computer and System Sciences*, 43:290–298, 1991.
- [2] 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. *Journal of Computer and System Sciences*, 45:104–126, 1992.
- [3] I. Chlamtac and S. Kutten. On broadcasting in radio networks – problem analysis and protocol design. *IEEE Transactions on Communications*, 33:1240–1246, 1985.
- [4] B. S. Chlebus, L. Gąsieniec, A. Östlin, and J. M. Robson. Deterministic radio broadcasting. In *Proceedings of the 27th Annual International Colloquium on Automata, Languages and Programming (ICALP)*, pages 717–728, 2000.
- [5] M. Chrobak, L. Gąsieniec, and W. Rytter. Fast broadcasting and gossiping in radio networks. *Journal of Algorithms*, 43:177–189, 2002.
- [6] A. E. F. Clementi, A. Monti, and R. Silvestri. Distributed broadcasting in radio networks of unknown topology. *Theoretical Computer Science*, 302:337–364, 2003.
- [7] A. Czumaj and W. Rytter. Broadcasting algorithms in radio networks with unknown topology. In *Proceedings of the 44th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 492–501, 2003.
- [8] R. Elsässer and L. Gąsieniec. Radio communication in random graphs. *Journal of Computer and System Sciences*, 72:490–506, 2006.
- [9] R. Gandhi, S. Parthasarathy, and A. Mishra. Minimizing broadcast latency and redundancy in ad hoc networks. In *Proc. the 4th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, pages 222–232, 2003.
- [10] V. Kawadia and P. Kumar. Power control and clustering in ad hoc networks. In *Proc. IEEE INFOCOM*, 2003.
- [11] E. Kushilevitz and Y. Mansour. An  $\Omega(D \log(N/D))$  lower bound for broadcast in radio networks. *SIAM Journal on Computing*, 27:702–712, 1998.
- [12] K. Nakano and S. Olariu. Randomized initialization protocols for ad-hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 11:749–759, 2000.
- [13] A. Sen and M. L. Huson. A new model for scheduling packet radio networks. *Wireless Networks*, 3:71–82, 1997.