

Efficient Distributed Multiple-Message Broadcasting in Unstructured Wireless Networks

Dongxiao Yu*, Qiang-Sheng Hua†, Yuexuan Wang†, Jiguo Yu‡ and Francis C.M. Lau*

*Department of Computer Science, The University of Hong Kong, Hong Kong, P. R. China

†Institute for Theoretical Computer Science, Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, 100084, P. R. China

‡School of Computer Science, Qufu Normal University, Shandong, P. R. China

Abstract—Multiple-message broadcast is a generalization of the traditional broadcast problem. It is to disseminate k distinct ($1 \leq k \leq n$) messages stored at k arbitrary nodes to the entire network with the fewest timeslots. In this paper, we study this basic communication primitive in *unstructured* wireless networks under the physical interference model (also known as the SINR model). The *unstructured* wireless network assumes unknown network topology, no collision detection and *asynchronous* communications. Our proposed randomized distributed algorithm can accomplish multiple-message broadcast in $O((D + k)\log n + \log^2 n)$ timeslots with high probability, where D is the network diameter and n is the number of nodes in the network. To our best knowledge, this work is the first one to consider distributively implementing multiple-message broadcasting in *unstructured* wireless networks under a global interference model, which may shed some light on how to efficiently solve in general a “global” problem in a “local” fashion with “global” interference constraints in *asynchronous* wireless ad hoc networks. Apart from the algorithm, we also show an $\Omega(D+k+\log n)$ lower bound for randomized distributed multiple message broadcast algorithms under the assumed network model.

I. INTRODUCTION

Multiple-message broadcast is a basic operation in many applications in wireless networks, such as updating of routing tables, topology learning of the underlying network, and many kinds of data aggregation functions in sensor networks. Due to the fundamental importance of this problem, many efficient distributed protocols and algorithms have been proposed in the literature. However, most previous work assumed a simplified environment without some of the more realistic constraints. In this paper, we study the multiple-message broadcast problem under the **unstructured** wireless network model [15] which is more accurate in representing real wireless ad-hoc and sensor networks than other simplified models. The unstructured wireless network model assumes **asynchronous** communications, no predefined structure, no collision detection, and that nodes have no knowledge of the network topology such as neighborhood information and the network diameter. To deal with interferences, our work employs the widely-used Signal-to-Interference-plus-Noise-Ratio (SINR) model (also known as the physical interference model), which takes into account the interferences from all simultaneously transmitting nodes. The SINR model assumes that a signal fades with the distance to

the power of some path-loss exponent α , and a signal can be successfully decoded at the receiver if and only if (iff) the ratio of the received signal strength and the accumulation of the interference caused by the other simultaneously sending nodes plus noise is above a hardware-defined threshold β . The constraints of the unstructured wireless network model and the global nature of the SINR model together pose a challenge to the design of efficient distributed protocols.

In practice, synchronization incurs extra time and message overhead, which may be unaffordable to tiny sensors. So it is meaningful to design efficient algorithms for asynchronous environments. It is also well-known that in real distributed systems, where all messages arrive after an unknown and variable delay, nodes have no current information about the other nodes’ clock values. It is therefore impossible to synchronize the clocks perfectly. As pointed out in [16], even if the message delays were always the same and the nodes knew this value exactly, the clocks still could not be synchronized perfectly because of the variable hardware clock drifts. So far, results on distributed algorithm design under asynchronous communication environments have been rather limited [1], [2], [3], [4], [10], [11], [13], [20].

In this paper, we propose an **asynchronous** randomized distributed multiple-message broadcast algorithm for unstructured wireless networks under the SINR model. We show that the proposed algorithm can disseminate all messages to all nodes in $O((D + k)\log n + \log^2 n)$ timeslots with high probability, where D is the network diameter (cf. the definition in Section III), k is the number of messages to be disseminated, and n is the number of nodes in the network. In addition, we also derive an $\Omega(D + k + \log n)$ lower bound for randomized distributed algorithms solving the problem. The proposed algorithm consists of three processes: Leader election, Local information collection and Broadcast. The leader election process constructs a connected backbone network to be used by the broadcast process, which takes place after the elected leaders have collected the stored messages from neighboring nodes in the local information collection process. Here we emphasize that, under the *asynchronous* circumstance, we can not make use of a commonly adopted TDMA (Time-Division Multiple-Access) scheme to coordinate the transmissions of neighboring nodes. Also because of the

asynchronous environment, these three processes can not be divided into phases. Instead, all the three processes may be executed concurrently by different nodes in the network. So the challenge is **how to guarantee successful transmissions only based on local information when communications are asynchronous**. Our solution is by appropriately adjusting the nodes' transmission probabilities to bound the interference at a receiver caused by "far-away" nodes. Using a novel transmission probability updating strategy, we show that the sum of transmission probabilities of nodes in any local region can be bounded. As a result, the interference at a receiver caused by far-away nodes can also be bounded by a constant during the execution of the algorithm, which translates into a sufficient condition for locally determining whether a transmission is successful or not under the SINR model. We then show that such a strategy can guarantee that a message can be broadcast to the whole network within the stated time bound. In the algorithm, each node executes the program only based on its own clock.

Although there is a recently proposed randomized $O(D + k + \log^2 n)$ time distributed multiple-message broadcast algorithm [20] under the SINR model, it relies heavily on **synchronous** communications. With a global clock and synchronized communications, their algorithm can adopt a TDMA-like scheme to achieve efficient message delivery. This renders the algorithm inapplicable in **asynchronous** environments. Furthermore, comparing with the best known $\min\{O(k \log n \log \Delta + (D + n/\log n) \log n \log \Delta), O((k\Delta \log n + D) \log \Delta)\}$ time results [1], [11] under the graph based model, which assumes local interference, synchronous communications and some prior information about the neighborhood (e.g., the maximum degree Δ), our proposed algorithm is faster in spite of the fact that we adopt a much harsher communication model.

II. RELATED WORK

To our best knowledge, the unstructured wireless network model was first formally proposed in [15]. Due to the conceived practicality of this model, it has been widely adopted in designing efficient distributed protocols for many important problems in wireless networks, such as the network initialization problem [15], the maximal independent set problem [17], and the coloring problem [19]. Different from our paper here, all the above algorithms were proposed under the graph based interference model which naturally and artificially suits the design of distributed algorithms.

The SINR interference model has been shown to be superior in terms of increased network throughput as compared to the graph based models [7]. However, the global interference feature of the SINR model makes designing local or distributed algorithms difficult, and there have been only a handful of recent studies [6], [23], [21], [22] assuming the SINR model, let alone for unstructured wireless networks.

As a basic broadcast primitive, the multiple-message broadcast problem has been extensively studied by the distributed

computing community since the 1990s. It appears that all previous work except [20] adopted the graph based radio network model. In the radio network model, synchronous communications are assumed and a transmission can be successful if and only if there is only one neighbor transmitting a message to the receiver. The synchronized communication and local interference as defined in graph based models simplify the design of distributed protocols. In addition, it was commonly assumed that nodes know some or all the network parameters, e.g., Δ and D . Under this model, the best known randomized distributed results have time complexity $\min\{O(k \log n \log \Delta + (D + n/\log n) \log n \log \Delta), O((k\Delta \log n + D) \log \Delta)\}$ [1], [11]. The best known lower bound under the graph-based radio network model is $\Omega(k + D \log(n/D))$ in expectation [4]. Furthermore, under the radio network model, how to use network coding to accelerate multiple-message broadcast was studied. In [10], by combining randomized techniques and simple coding, Khabbazi and Kowalski presented a faster $O(k \log \Delta + (D + \log n) \log n \log \Delta)$ time randomized algorithm. In [12], based on a proposed Analog Network Coding based algorithm that implements an abstract MAC layer service, Khabbazi et al. also gave a randomized multiple message broadcast algorithm of time complexity $O((D+k \log(\frac{nk}{\epsilon})) \log \Delta + k(\Delta + \Delta \log(\frac{\Delta nk}{\epsilon})))$ with probability guarantee $1 - \epsilon$. The $O(D+k+\log^2 n)$ time randomized algorithm in [20] under the SINR model breaks the lower bound for graph based interference models. However, the algorithm in that paper can only work in a synchronous environment. Regarding deterministic distributed solutions, the state-of-the-art result was given in a recent paper [3] which completes multiple-message broadcast in $O(k \log^3 n + n \log^4 n)$ time. The highest lower bound for the time complexity of a deterministic solution is $\Omega(k+n \log n)$ [4]. Furthermore, In [13], by assuming the existence of an abstract MAC layer, Kuhn et al. presented a multiple-message broadcast protocol for regional networks with running time $O((D+k)F_{prog} + (k-1)F_{ack})$, where F_{prog} and F_{ack} are progress and acknowledgement bounds, respectively. The multiple-message broadcast problem was also studied in networks with multiple channels [8] and in dynamic networks [14].

III. DEFINITIONS AND MODEL

Given a network consisting of n nodes arbitrarily distributed on the plane and an arbitrary subset of $k \leq n$ nodes, each of which has been given a distinct message, the multiple-message broadcast problem is to disseminate all these k messages to every node in the network. We assume that the nodes have no knowledge of both k and which k nodes are in the subset.

Let $d(u, v)$ denote the Euclidean distance between two nodes u, v . We say two nodes are independent in terms of a given distance d if $d(u, v) > d$. An independent set I in terms of d is defined as a set of nodes where every pair of nodes are independent. Then if the independent set I satisfies the condition that for each node v , either $v \in I$, or there is a node $u \in I$ such that $d(u, v) \leq d$, I is called a maximal independent set in terms of d . We define a dominating set S

as a set of nodes in which for any node v , either $v \in S$, or there is a node in S which is within distance d from v . Let $G = (V, E)$, where V is the set of nodes, and E is the set of links that connect every pair of nodes within distance d . Then a dominating set S is said to be connected in terms of d if the subgraph $G[S]$ of G induced by S is connected.

In order to make sure that all nodes can receive all k messages, we assume that the nodes are already woken up before the algorithm starts. At the beginning, the nodes have no knowledge about neighborhood, i.e., they do not know their neighbors, not even their number. The nodes also do not have any estimate of the maximum network degree (the maximum number of neighbors of any node). The only prior knowledge given to the nodes is an estimate of the number of nodes in the network. A polynomial estimate n is enough, the bias of which will only affect the performance of our algorithm by a constant factor. Although the exact number of nodes may be difficult to predict in practice, a polynomial estimate can be easily provided [5]. We assume each node has a unique ID, which does not need to be in the interval $[1, n]$. We will not perform any special computation using the IDs. They are only used by the receiver to identify the senders and tell them apart. There is no collision detection mechanism assumed in the nodes, i.e., nodes can not distinguish between a collision and the case where there is no transmission.

We assume that the time is divided into timeslots. Note however that the proposed algorithm itself does *not* rely on synchronous communications in any way. The sole purpose of assuming slotted channels is to ease the algorithm analysis. As shown in [18], [15] where the slotted and unslotted ALOHA systems are compared, the difference is just a factor of two if one analyzes the algorithm in a synchronized timeslot setting as opposed to the realistic unslotted setting.

We employ the SINR interference model (also called the physical interference model). In the SINR model, the signal strength fades with the distance according to some path-loss exponent α , and a successful transmission occurs iff the SINR value at the receiver of the link is above a hardware related threshold β . Formally, a message sent by node u to node v can be correctly received at v iff

$$\frac{P_u/d(u,v)^\alpha}{N + \sum_{w \in V \setminus \{u,v\}} P_w/d(w,v)^\alpha} \geq \beta, \quad (1)$$

where P_u (P_w) is the transmission power for node u (w); α is the path-loss exponent whose value is normally between 2 and 6; β is a hardware determined threshold which is greater than 1; N is the ambient noise, $d(u,v)$ denotes the Euclidean distance between u, v and $\sum_{w \in V \setminus \{u,v\}} \frac{P_w}{d(w,v)^\alpha}$ is the interference experienced by the receiver v caused by all simultaneously transmitting nodes in the network.

Given a transmission power P for a node v , the transmission range R of v is defined as the maximum distance at which a node u can receive a clear transmission from v ($\text{SINR} \geq \beta$) when there are no other simultaneous transmissions in the network. By the SINR constraint (1), we define

$R = (P/cN\beta)^{1/\alpha}$, where $c > 1$ is a constant determined by the environment. Based on the transmission ranges of nodes, we define a communication graph $G = (V, E)$, where V is the set of nodes in the network, and a link (u, v) exists in E if and only if the distance from u to v is not larger than the transmission range of u . Furthermore, if all nodes have the same transmission range R_T , the obtained communication graph is denoted as G_{R_T} . Obviously, in this case, G_{R_T} can be seen as an undirected graph. We say a network is connected in terms of d if the communication graph G_d is connected. Let P_M and R_M be the maximum transmission power and the corresponding maximum transmission range of nodes, respectively. By the above definitions, $R_M = (P_M/cN\beta)^{1/\alpha}$. We denote D as the diameter of the communication graph G_{R_M} .

IV. ALGORITHM

A. Algorithm Description

In this section, we describe our randomized distributed multiple-message broadcast algorithm. As briefly mentioned in Section I, the algorithm is divided into three processes: Leader election, Local information collection and Broadcast. Note that although this general framework of using the three processes has appeared before under different settings [11], [13], [20], [21], what remains tricky and difficult is how to implement each process efficiently under a particular network and interference model. Here we briefly outline how the three processes are implemented under the unstructured wireless network model and the physical interference model. At first, a leader election protocol is executed to construct a connected dominating set in terms of range R_M . The nodes in the connected dominating set are called leaders; the others are non-leaders, each of which chooses one of the neighboring leaders as its leader. A cluster consists of a leader v and non-leaders that have chosen v as their leader. After that, all non-leaders transmit their stored messages to their leaders. Finally, all the leaders which comprise a connected backbone network are responsible for broadcasting the messages. In order to compute the connected dominating set, we execute the MIS (Maximal Independent Set) algorithm in [22] to obtain a maximal independent set in terms of $R_M/3$.¹ Obviously, when nodes adopt the maximum transmission power whose corresponding transmission range is R_M , the MIS nodes form a connected dominating set in terms of R_M according to the fact that a connected dominating set can be constructed by connecting each pair of MIS nodes within three hops [2]. We introduce the protocols for completing the local information collection and broadcast in the next paragraph. During the local information collection and broadcast processes, all nodes use the maximum transmission power P_M . As mentioned before, **due to the asynchronous communications, these three processes may be executed concurrently by different nodes in the network**. Next we describe the algorithm from the view of a node.

¹We assume that the network is connected in terms of $R_M/3$.

Each node v is assigned a queuing set M_v storing the received messages. Furthermore, for each leader v , it is assigned another set Q_v to store the IDs of non-leaders in its cluster that have sent messages to v . After starting the algorithm, a node v first executes the MIS algorithm to decide whether joining the MIS and becoming a leader. After that, if v becomes a leader, it starts executing the local information collection process as given in Algorithm 1. During this process, if v receives a message from a non-leader u in its cluster, it adds the received message to M_v and adds u 's ID to Q_v . If Q_v is not empty, v will transmit an $Ack_v(u)$ message for $\mu \log n$ timeslots with a constant probability for the first node u in Q_v , by which v informs u that it has received u 's message. Then v deletes u from Q_v . We set two counters s_v and t_v to decide when v stops the information collection process. Specifically, s_v is used to count the number of timeslots that v has not received a new message from a non-leader in its cluster, and t_v is used to count the number of Ack messages that have been transmitted by v . If the quitting condition of $s_v > t_v \cdot \mu \log n + 2\mu \log^2 n + 3\mu \log n$ is satisfied, v stops the local information collection process and starts the broadcast process as given in Algorithm 2. In the analysis, we will show that with high probability, all non-leaders in v ' cluster have sent their initially stored messages to v before the quitting condition is satisfied. In the broadcast process, v locally broadcasts each message stored in M_v to all its neighbors by transmitting the message with a constant probability for $\Theta(\log n)$ timeslots.

If v becomes a non-leader after executing the MIS algorithm, it chooses the first node that sent a dominating message to it as the leader, as shown in the MIS algorithm [22]. If initially, v has a message that it wants to share in the network, it starts the local information collection process as shown in Algorithm 3. Otherwise, it just listens. During the local information collection process, by continuously transmitting a combined message including the stored information and the IDs of v and its leader with a specified transmission probability, v endeavors to send the stored message to its leader. Due to the lack of an accurate upper bound on the number of nodes in a node's transmission range, we can not use this parameter to assign an appropriate transmission probability. So we set a non-leader v 's initial transmission probability as a very small value $\Theta(\frac{1}{n})$ determined by n . v doubles its transmission probability every $2\mu \log n$ (μ is a constant determined later) timeslots if it does not receive Ack messages from its leader. In order to bound the interference at a receiver caused by far-away nodes, as shown in the analysis, we need to ensure that in any local region, the sum of transmission probabilities of nodes should be upper bounded by a constant. For this purpose, if v received an Ack message from its leader which is not for v , it stops increasing the transmission probability for $2\mu \log n$ timeslots. Finally, after receiving an $Ack_u(v)$ message from v 's leader u , v quits the information collection process. From then on, it listens to messages sent by leaders.

We set the constant parameters used in the algorithm

as follows to guarantee that our algorithm can accomplish multiple-message broadcast with high probability: $\lambda = \frac{1-\phi}{3}$, $\mu = \frac{48 \cdot 4^{\chi(R_I + R_M, 0.5R_M)}}{\lambda(1-1/\rho)}$. In the above definitions, ϕ is the upper bound of the transmission probability sum of nodes in any disk with radius $R_M/2$ that are executing the MIS algorithm. In [22], it has been proved that ϕ is a known constant less than $\frac{1}{3}$ with probability $1 - O(n^{-1})$. $\chi(R_1, R_2)$ denotes the number of disks with radius R_2 needed to cover a disk with radius R_1 , where $R_1 > R_2$. And $R_I = R_M \left(\frac{360\rho\beta}{1-1/c} \cdot \frac{\alpha-1}{\alpha-2} \right)^{1/(\alpha-2)}$ which is used to define far-away nodes in the analysis, where ρ is a constant larger than 1 chosen such that $R_I > 2R_M$.

Algorithm 1 Local Information Collection for a Leader v

Initially, $Q_v = \emptyset; M_v = \emptyset; s_v = 0; t_v = 0$

- 1: **if** $s_v > t_v \cdot \mu \log n + 2\mu \log^2 n + 3\mu \log n$ **then**
- 2: start the broadcast process as shown in Algorithm 2
- 3: **else if** Q_v is not empty **then**
- 4: **for** $\mu \log n$ timeslots **do** delete the first node u from Q_v and transmit $Ack_v(u)$ with probability $\lambda/16$; $s_v = s_v + 1$;
- 5: **end for**
- 6: $t_v = t_v + 1$;
- 7: **else** $s_v = s_v + 1$;
- 8: **end if**
- 9: **end if**

Message Received

- 1: **if** v received a message from another leader u that has not been received before **then** add the message into M_v **end if**
 - 2: **if** v received a message from a non-leader u in its cluster that has not been received before **then** add the message into M_v and add u into Q_v ; $s_v = 0$. **end if**
-

Algorithm 2 Broadcast for a Leader v

- 1: **if** M_v is not empty **then**
- 2: **for** $\mu \log n$ timeslots **do** delete the first message from M_v and transmit this message with probability $\lambda/16$
- 3: **end for**
- 4: **else** listen;
- 5: **end if**

Message Received

- 1: **if** v received a message from another leader u that has not been received **then** add the message into M_v **end if**
-

B. Analysis

In this section, we show that with probability at least $1 - O(n^{-1})$, all messages can be delivered to the whole network in $O((D+k) \log n + \log^2 n)$ timeslots. We use D_v , T_v and I_v to denote the disks of radii $R_M/2$, R_M and R_I centered at node

Algorithm 3 Local Information Collection for a Non-Leader v

Initially, $p_v = \frac{\lambda}{16n}$; $d_v = 0$;
1: $d_v = d_v + 1$
2: **if** v initially stores a message and the message $Ack_u(v)$ from its leader u has not been received **then** transmit the message with probability p_v ;
3: **else** listen
4: **end if**
Update p_v and d_v
5: **if** $d_v > 2\mu \log n$
 then $p_v = 2p_v$; $d_v = 0$.
 end if
6: **if** received Ack_w from its leader for some node w that has not been received before
 then $d_v = 0$;
 end if

v respectively. The notation E_v^d denotes the disk of radius d centered at v . Without confusion, we also use these notations to denote the nodes located in the corresponding disks.

To prove the correctness and efficiency of the algorithm, the basic idea is to show that under the condition that the sum of transmission probabilities of nodes in any local region is bounded by a constant, a sufficient condition for successful transmissions can be obtained. Based on this, it can be shown that all three processes are correctly executed in the stated time bound. The condition on the transmission probability sum, as will be shown, can be guaranteed by the transmission probability adjustment strategy used in the algorithm.

We first give a property which states that the number of leaders and the transmission probability sum of non-leaders in any disk D_v are bounded by constants. The following *Property 1 (i)* can be obtained using a standard area argument, and *Property 1 (ii)* is guaranteed by the adopted probability adjustment strategy in Algorithm 3. In Lemma 8, *Property 1* will be shown to be correct with probability $1 - O(n^{-1})$.

Property 1: For any disk D_v and in any timeslot t throughout the execution of the algorithm,

- (i) there are at most 16 leaders in D_v ;
- (ii) the sum of transmission probabilities of non-leaders is at most 2λ .

Based on the above property and the transmission probability for leaders, the transmission probability sum of nodes in any disk D_v that are executing the local information collection and broadcast processes is at most $2\lambda + 16 \times \frac{\lambda}{16} = 3\lambda$.

In [22], it is shown that as long as the transmission probability sum in any local region can be bounded by a constant, the MIS algorithm is correct even if there are some other algorithms executing concurrently in the network. The following Lemma 1 has been proved and *Property 2* is also shown to be correct with probability at least $1 - O(n^{-1})$ in [22].

Lemma 1: After executing the MIS algorithm for $O(\log^2 n)$ timeslots, a maximal independent set can be correctly com-

puted with probability at least $1 - O(n^{-1})$.

Property 2: During the execution of the MIS algorithm, in any timeslot and for any disk D_v , the transmission probability sum of nodes is upper bounded by $\sum_{u \in D_v} p_u \leq \phi$, where ϕ is a known constant less than $1/3$.

Based on above *Property 1* and *Property 2*, we can bound the transmission probability sum of nodes in any disk D_v , as shown in the following Lemma 2.

Lemma 2: Assume that Property 1 and Property 2 hold. For any disk D_v and in any timeslot t during the algorithm execution, the sum of transmission probabilities of nodes in D_v is at most 1.

Now we are ready to derive a sufficient condition for successful transmissions. For two disks D_1 and D_2 with radii R_1 and R_2 respectively, where $R_1 > R_2$, recall that $\chi(R_1, R_2)$ is the number of disks D_2 needed to cover D_1 . Let v be the center of D_1 . Due to the limit of the ratio of the area of D_1 to the area of smaller disks D_2 is $\frac{3\sqrt{3}}{2\pi}$ [9] and all disks D_2 that intersect D_1 is contained in the region $E_v^{R_1+2R_2}$, we can easily obtain the following lemma.

$$\text{Lemma 3: ([6]) } \chi(R_1, R_2) \leq \frac{2\pi}{3\sqrt{3}} \cdot \frac{(R_1+2R_2)^2}{R_2^2}.$$

We define the *probabilistic interference* at a receiver as the expected interference experienced by the receiver. Formally, for a node $v \in V$, the *probabilistic interference* at v in a certain timeslot t , is defined as: $\Psi_v = \sum_{u \in V \setminus \{v\}} \frac{P_u p_u}{d(u,v)^\alpha}$, where P_u is the transmission power and p_u is the sending probability of node u in timeslot t .

Benefiting from Lemma 2, the probabilistic interference at a receiver from far-away nodes (with distance larger than R_I) can be bounded by a constant, which is used to obtain a sufficient condition for successful transmissions, as shown in the following Lemma 4.

Lemma 4: Assume that Property 1 and Property 2 hold. If v is the only transmitting node in $E_v^{R_I+R_M}$, then v can successfully transmit its message to all nodes in T_v with probability $1 - \frac{1}{\rho}$.

Proof: We first prove a claim which states that the probabilistic interference from far-away nodes can be bounded by a constant.

Claim 1: For every node u , the probabilistic interference caused by nodes outside I_u can be bounded as: $\Psi_u^{v \notin I_u} \leq \frac{(1-1/c)P_M}{\rho\beta R_M^\alpha}$.

Proof: We use an area argument. Denote $R_l = \{v \in V : lR_I \leq d(u,v) \leq (l+1)R_I\}$ and let S be a maximum independent set in terms of R_M whose nodes are in R_l . Note that S is also a dominating set, which means that $\sum_{v \in S} T_v$ covers all nodes of R_l . Note also that all disks D_v for $v \in S$ are mutually disjoint and all nodes in these disks are contained in $R_l^+ = \{v \in V : lR_I - \frac{R_M}{2} \leq d(u,v) \leq (l+1)R_I + \frac{R_M}{2}\}$. So $|S| \leq \frac{\text{Area}(R_l^+)}{\text{Area}(\text{disk}(R_M/2))}$. Then we can upper bound the probabilistic interference caused by nodes in R_l as follows,

$$\begin{aligned}
\Psi_u^{R_l} &= \sum_{v \in R_l} \Psi_u^v \\
&\leq \frac{\text{Area}(R_l^+)}{\text{Area}(\text{disk}(R_M/2))} \cdot \max_{i \in S} \left\{ \sum_{v \in T_i \cap R_l} \frac{P_M \cdot p_v}{(lR_l)^\alpha} \right\} \\
&= \frac{\pi((l+1)R_l + R_M/2)^2 - (lR_l - R_M/2)^2}{\pi(R_M/2)^2} \\
&\quad \cdot \max_{i \in S} \left\{ \sum_{v \in T_i \cap R_l} \frac{P_M \cdot p_v}{(lR_l)^\alpha} \right\} \\
&= \frac{4(2l+1)(R_l^2 + R_l R_M)}{R_M^2} \cdot \frac{P_M}{(lR_l)^\alpha} \cdot \max_{i \in S} \left\{ \sum_{v \in T_i \cap R_l} p_v \right\} \quad (2)
\end{aligned}$$

By Lemma 2 and Lemma 3, the sum of transmission probabilities of nodes in each T_i can be bounded as follows:

$$\sum_{v \in T_i} p_v \leq \frac{2\pi}{3\sqrt{3}} \cdot \frac{(R_M + 2 \cdot \frac{R_M}{2})^2}{(\frac{R_M}{2})^2} \cdot \sum_{v \in D_w} p_v \leq 20 \quad (3)$$

Note that $R_M < \frac{R_l}{2}$. Then we have

$$\begin{aligned}
\Psi_u^{v \notin I_u} &= \sum_{l=1}^{\infty} \Psi_u^{R_l} \\
&\leq \sum_{l=1}^{\infty} \frac{4(2l+1)(R_l^2 + R_l R_M)}{R_M^2} \cdot \frac{P_M}{(lR_l)^\alpha} \cdot 20 \\
&\leq \sum_{l=1}^{\infty} \frac{120P_M R_l^2 (2l+1)}{R_M^2 R_l^\alpha l^\alpha} \\
&\leq \sum_{l=1}^{\infty} \frac{360P_M R_l^2}{R_M^2 R_l^\alpha (l-1)^\alpha} \\
&\leq \frac{360P_M R_l^2}{R_M^2 R_l^\alpha} \cdot \frac{\alpha-1}{\alpha-2} \\
&\leq \frac{(1-1/c)P_M}{\rho\beta R_M^\alpha} \quad (4)
\end{aligned}$$

We are now ready to prove the lemma. By the above claim and using Markov inequality, with probability at least $1-1/\rho$, the interference at node u that is caused by nodes outside I_u is at most $\frac{(1-1/c)P_M}{\beta R_M^\alpha}$. Then if v is the only transmitting node, with probability $1-1/\rho$, the SINR at any node u in T_v can be lower bounded as

$$\begin{aligned}
\text{SINR} &\geq \frac{\frac{P_M}{d(u,v)^\alpha}}{N + \frac{(1-1/c)P_M}{\beta R_M^\alpha}} \\
&\geq \frac{\frac{P_M}{R_M^\alpha}}{\frac{P_M}{c\beta R_M^\alpha} + \frac{(1-1/c)P_M}{\beta R_M^\alpha}} \\
&= \beta \quad (5)
\end{aligned}$$

So by the the SINR constraint (1), u can successfully receive the message sent from v .

With the sufficient condition given in Lemma 4, we next show that a leader can send a message to all its neighbors in $\mu \log n$ timeslots with high probability. ■

Lemma 5: Assume that Property 1 and Property 2 hold. Then a leader v can successfully send a message to all its neighbors in $\mu \log n$ timeslots with probability at least $1 - O(n^{-3})$.

Proof: Denote P_{only} as the probability that v is the only transmitting node in $E_v^{R_l+R_M}$. We can lower bound P_{only} as follows.

$$\begin{aligned}
P_{\text{only}} &= p_v \prod_{u \in E_v^{R_l+R_M} \setminus \{v\}} (1-p_u) \\
&\geq \frac{\lambda}{16} \prod_{u \in E_v^{R_l+R_M}} (1-p_u) \\
&\geq \frac{\lambda}{16} \cdot \left(\frac{1}{4}\right)^{\sum_{u \in E_v^{R_l+R_M}} p_u} \\
&\geq \frac{\lambda}{16} \cdot \left(\frac{1}{4}\right)^{\chi(R_l+R_M, 0.5R_M)} \quad (6)
\end{aligned}$$

The last inequality is by Lemma 3 and Lemma 2. Then by Lemma 4, the probability P_{no} that v fails to send a message to all nodes in T_v is at most

$$\begin{aligned}
P_{\text{no}} &\leq (1 - (1-1/\rho) \frac{\lambda}{16} \cdot \left(\frac{1}{4}\right)^{\chi(R_l+R_M, 0.5R_M)})^{\mu \log n} \\
&\leq e^{-(1-1/\rho) \frac{\lambda}{16} \cdot \mu \log n \cdot \left(\frac{1}{4}\right)^{\chi(R_l+R_M, 0.5R_M)}} \\
&\leq n^{-3} \quad (7)
\end{aligned}$$

In order to ensure the correctness of our multiple-message broadcast algorithm, we need to show that all messages are collected by the leaders. This requires that any leader can not quit the local information collection process until it receives all the messages initially stored at the non-leaders in its cluster, which is guaranteed by the following Lemma 6. ■

Lemma 6: Assume that Property 1 and Property 2 hold. With probability at least $1 - O(n^{-2})$, a leader v will not quit the local information collection process until all non-leaders in its cluster have sent their stored messages to v .

Proof: Otherwise, assume that v quits the local information collection process in timeslot t when there are still some non-leaders in its cluster that fail to send their messages to v . Denote the set of these non-leaders as S , where $|S| > 0$. By Algorithm 1, v quits only if the condition $s_v > t_v \cdot \mu \log n + 2\mu \log^2 n + 3\mu \log n$ is satisfied, which means that from $t - (t_v \cdot \mu \log n + 2\mu \log^2 n + 3\mu \log n) - 1$, v does not receive any new message from non-leaders in its cluster. Note that v transmits every *Ack* message for $\mu \log n$ timeslots. Thus, v has completed the transmissions of all t_v *Ack* messages by the timeslot $t - (t_v \cdot \mu \log n + 2\mu \log^2 n + 3\mu \log n) + t_v \mu \log n - 1$. From then on, these non-leaders do not receive any *Ack* message from v before t . Furthermore, by Algorithm 3, from $t - (t_v \cdot 3\mu \log n + 2\mu \log^2 n + \mu \log n) + t_v \mu \log n - 1$, for each

node $u \in S$, it stops increasing its transmission probability for at most $2\mu \log n$ timeslots. Then each node in S will double its transmission probability every $2\mu \log n$ timeslots. By the timeslot $t - (t_v \cdot \mu \log n + 2\mu \log^2 n + 3\mu \log n) + t_v \mu \log n + 2\mu \log n + 2\mu \log^2 n - 1 = t - \mu \log n - 1$, each node in S has a constant transmission probability $\lambda/16$. Using a similar argument as in proving Lemma 5, we have that with probability at least $1 - O(n^{-2})$, each node in S has successfully sent their messages to v by $t - 1$, which makes s_v be reset as 0. Thus, v does not quit the local information collection process in timeslot t with probability $1 - O(n^{-2})$. This contradiction completes the proof. ■

For a node v , denote Δ_k^v as the number of nodes within distance R_M from v that initially store a message. Let $\Delta_k = \max\{\Delta_k^v\}$. Clearly, $\Delta_k \leq k$, since there are totally k messages. In the following lemma, we bound the number of timeslots that a leader spends in executing the local information process.

Lemma 7: Assume that Property 1 and Property 2 hold. With probability at least $1 - O(n^{-2})$, for a leader v , the local information collection process takes $O(\Delta_k \log n + \log^2 n)$ timeslots.

Proof: Denote by S the set of non-leaders in v 's cluster that take part in the local information collection process. In Lemma 6, it has been shown that with probability $1 - O(n^{-2})$, the information collection process will end after v has collected all information from non-leaders in S for $O(\Delta_k \log n + \log^2 n)$ timeslots, since $t_v \leq \Delta_k$. So we only need to show that each non-leader in S takes at most $O(\Delta_k \log n + \log^2 n)$ timeslots to successfully send the stored message to v . By Algorithm 1, each node $u \in S$ doubles its transmission probability every $2\mu \log n$ timeslots if it does not receive an *Ack* message from v . Thus after at most $(\Delta_k - 1 + \log n) \cdot 2\mu \log n$ timeslots, either u receives an *Ack_v*(u) message from v which means that v has received the message transmitted by u , or u has a constant transmission probability of $\lambda/16$, since u can receive at most $\Delta_k - 1$ *Ack* message from v that are not for u , and each of these *Ack* message can make u stop increasing the transmission probability for at most $2\mu \log n$ timeslots. Then using a similar argument as in the proof of Lemma 5, it can be shown that with probability $1 - O(n^{-3})$, v will successfully receive the message transmitted by u in the subsequent $\mu \log n$ timeslots. Thus, after transmitting for $O(\Delta_k \log n + \log^2 n)$ timeslots, each node in S can successfully send a message to v with probability $1 - O(n^{-2})$. Combining all together, the lemma is proved. ■

Before the proof of the main theorem, we show the correctness of Property 1 in the following Lemma 8.

Lemma 8: Property 1 is correct with probability at least $1 - O(n^{-1})$.

Proof: (i) Using an area argument as in the proof of Lemma 4, this property can be easily obtained as long as the MIS is correctly computed as shown in Lemma 1.

(ii) For a leader u , denote S_u to be the set of non-leaders in u 's cluster. We first prove the following claim.

Claim 2: In any timeslot during the local information collection process, for a leader u , with probability at least $1 - O(n^{-2})$, $\sum_{w \in S_u} p_w(t) \leq \frac{\lambda}{8}$, where $p_w(t)$ is the transmission probability of node w in timeslot t .

Proof: Otherwise, assume that t^* is the first violating timeslot. By Algorithm 1, a non-leader doubles its transmission probability at most once in $2\mu \log n$ timeslots. Thus, during the interval $I = [t^* - 2\mu \log n, t^*)$, $\frac{\lambda}{16} < \sum_{w \in S_u} p_w(t) \leq \frac{\lambda}{8}$. Obviously, before any violating timeslot, there exists such an interval I . Next we show that in timeslot t^* , with probability at least $1 - O(n^{-3})$, $\sum_{w \in S_u} p_w(t^*)$ will not exceed $\frac{\lambda}{8}$. This contradiction will help complete the proof of the claim.

First, we prove that at least one node in S_u can successfully send a message to u during the interval $I_1 = [t^* - 2\mu \log n, t^* - \mu \log n)$. Using an argument similar to that in proving Lemma 4, it can be shown that if a node w is the only transmitting node in $E_u^{R_I}$, u can successfully receive the message sent by w with probability at least $1 - \frac{1}{\rho}$. Let P_{only} denote the probability that a node $w \in S_u$ is the only transmitting node in $E_u^{R_I}$. Let C be a cover of $E_u^{R_I}$ using the minimum number of disks with radius $R_M/2$. Then we can lower bound P_{only} as follows.

$$\begin{aligned}
P_{only} &= \sum_{w \in S_u} p_w \prod_{w' \in E_u^{R_I} \setminus \{w\}} (1 - p_{w'}) \\
&\geq \sum_{w \in S_u} p_w \prod_{D_j \in C} \prod_{w' \in D_j} (1 - p_{w'}) \\
&\geq \sum_{w \in S_u} p_w \prod_{D_j \in C} \left(\frac{1}{4}\right)^{\sum_{w' \in D_j} p_{w'}} \\
&\geq \sum_{w \in S_u} p_w \left(\frac{1}{4}\right)^{\chi(R_I, R_M/2) \sum_{w' \in D_j} p_{w'}} \\
&\geq \frac{\lambda}{16} \cdot \left(\frac{1}{4}\right)^{\chi(R_I, R_M/2)}
\end{aligned} \tag{8}$$

So during I_1 , the probability P_T that there is no node in S_u successfully transmitting a message to u is at most

$$P_T \leq \left(1 - \left(1 - \frac{1}{\rho}\right) \cdot \frac{\lambda}{16} \left(\frac{1}{4}\right)^{\chi(R_I, R_M/2)}\right)^{\mu \log n} \leq n^{-3} \tag{9}$$

Thus, by the timeslot $t - \mu \log n - 1$, with probability $1 - O(n^{-3})$, u can receive at least one message from non-leaders in its cluster. By Algorithm 1, as long as Q_u is not empty, u transmits an *Ack* message every $\mu \log n$ timeslots. And a non-leader stops transmitting after receiving the *Ack* message for it. As shown above, Q_u is not empty by the timeslot $t - \mu \log n - 1$. Thus, u starts transmitting a new *Ack_u*(w) message from a timeslot t_1 in the interval $[t^* - 2\mu \log n + 1, t^* - \mu \log n]$. By Lemma 5, with probability $1 - O(n^{-3})$, u can send the *Ack* message to all its neighbors in $\mu \log n$ timeslots. So all nodes received the *Ack_u*(w) message during the interval $[t_1, t_1 + \mu \log n)$. Furthermore, by Algorithm 1, after receiving the *Ack_u*(w) message, each non-leader in S_u stops increasing its transmission probability for $2\mu \log n$ timeslots except the node w that will stop transmitting. Let t' be the first timeslot that all nodes in S_u have received *Ack_u*(w). All nodes

keep their transmission probabilities unchanged in the interval $(t', t' + 2\mu \log n - (t' - t_1)) = (t', t_1 + 2\mu \log n]$. Note that t^* is in this interval. So $\sum_{w \in S_u} p_w(t^*) \leq \sum_{w \in S_u} p_w(t') \leq \frac{\lambda}{8}$, since $t' \in I$. Combing all together, with probability at least $1 - O(n^{-3})$, $\sum_{w \in S_u} p_w(t^*) \leq \frac{\lambda}{8}$, which contradicts with the definition of t^* .

We still need to bound the number of potential violating timeslots for leader u . As shown above, before any potential violating timeslot, there is a new $Ack_u(w)$ message transmitted by u and the node $w \in S_u$ stops transmitting after receiving this message. So there are at most Δ_k potential violating timeslots. Thus, with probability $1 - O(n^{-2})$, there is no violating timeslot for u . The claim is proved. ■

From (i), we know that there are at most 16 leaders in a disk D_v . Based on the above claim, it is easy to get that with probability $1 - O(n^{-2})$, *Property 1 (ii)* is correct for D_v during the execution of the algorithm. So *Property 1 (ii)* is correct for all disks with probability $1 - O(n^{-1})$. Thus, the lemma holds. ■

The following Lemma 9 is given in [13], which analyzes the pipelining effect of the multiple-message broadcast process. Let F_{prog} denote the maximum number of timeslots needed for a successful transmission. For a graph G , define $d_G(u, v)$ as the number of edges in the shortest path from u to v in G .

Lemma 9: Assume that in timeslot t_0 , a node u receives a new message m . Let v be a node at distance $d = d_G(u, v)$ from u . For integers $l \geq 1$, we define $t_{d,l} = t_0 + (d + 2l - 2)F_{prog}$. Then for all integers $l \geq 1$, at least one of the following two statements is true:

- (i) v received the message m by the time $t_{d,l}$;
- (ii) there exists a set M , $|M| = \min\{l, k\}$, such that for every $m' \in M$, v has received m' by the timeslot $t_{d,l}$.

Based on above lemmas, we next prove the main theorem.

Theorem 1: With probability $1 - O(n^{-1})$, all nodes will receive all k messages after executing the multiple-message broadcast algorithm for $O((D + k) \log n + \log^2 n)$ timeslots.

Proof: By Lemma 1 and Lemma 7, the leader election process and the local information collection process are completed by all nodes after executing the algorithm for $O(\Delta_k \log n + \log^2 n)$ timeslots with probability $1 - O(n^{-1})$. Next we analyze the completion time for the broadcast process. Denote by t_b the first timeslot that all nodes have completed the local information collection process. To simplify the analysis, we assume that all leaders synchronously start the broadcast process from the timeslot $t_b + 1$. Clearly, this assumption does not affect the analysis of the completion time of the broadcast process, since some leaders have started the broadcast process before t_b . By Lemma 5, a leader can successfully send a message to all its neighbors in $\mu \log n$ timeslots with probability $1 - O(n^{-3})$. Thus all messages can be successfully transmitted by all leaders to their neighbors with probability $1 - O(n^{-1})$. Denote D_1 as the diameter of the subgraph induced by nodes in the computed connected dominating set. Clearly, $D_1 = O(D)$. Then based on Lemma 9, by the timeslot $t_b + (D_1 + 2k - 2) \cdot \mu \log n$, all nodes in the network have received all k messages with probability $1 - O(n^{-1})$.

Finally note that all the above analyses are based on *Property 1* and *Property 2*. It has been proved that both properties are correct with probability $1 - O(n^{-1})$. Combining everything together and note that $\Delta_k \leq k$, we complete the proof of the theorem. ■

V. LOWER BOUND

In this section, we present Theorem 2 which gives a lower bound on the timeslots needed for accomplishing multiple-message broadcast.

Theorem 2: For any $D > 1$, if all nodes use the same transmission power, there exists an unstructured wireless network with diameter D such that any randomized distributed algorithm needs $\Omega(D + k + \log n)$ timeslots to accomplish multiple-message broadcast with probability at least $1 - \frac{1}{n}$.

Proof: Since $\Omega(D + k)$ is a trivial lower bound, we only need to show that any randomized distributed algorithm needs $\Omega(\log n)$ timeslots to accomplish multiple-message broadcast with probability $1 - \frac{1}{n}$. The lower bound is proved under the synchronous communication circumstance, which of course also holds in asynchronous communication models.

We consider the following network with diameter D : $n - 2$ nodes locate on a line which constitute a connected network with diameter $D - 1$, and the other two nodes u, w are within the transmission range of the first node v of the line but are out of the transmission ranges of the other nodes. Furthermore, the distance between any two nodes of u, v, w is the same. We assume that there are messages initially stored at u and w . So u and w must send their stored messages to v .

We now claim that any node in $\{u, v, w\}$ can successfully receive a message from the other two nodes only if these two nodes do not transmit at the same time. Without loss of generality, we prove this for v . Assume that u and w both transmit in a timeslot. Then for each node of u, w , the SINR at v is at most $\frac{P/r^\alpha}{N+P/r^\alpha} < 1 \leq \beta$, where r is the distance between u, w and v . So v can not receive any message from u, w .

Furthermore, for a given randomized algorithm, the transmission probability of each node in each timeslot can be known before it receives a message from its neighbor. By the pigeon hole principle, there must exist three nodes such that for the first $\frac{\log n}{4}$ timeslots, either all these three nodes have transmission probability at least $\frac{1}{2}$ or less than $\frac{1}{2}$ in each timeslot. We construct the network such that these three nodes are u, v and w . Next we show that with probability at least $\Omega(\frac{1}{\sqrt{n}})$, there are no successful transmissions between any pair of nodes in $\{u, v, w\}$ in the first $\frac{\log n}{4}$ timeslots. We prove this for u and v . In a timeslot, the probability that u can successfully transmit a message to v or the other way around is at most $p_u(1 - p_v)(1 - p_w) + p_v(1 - p_u)(1 - p_w) \leq p_u(1 - p_v) + p_v(1 - p_u) \leq \frac{3}{4}$. Then in the first $\frac{\log n}{4}$ timeslots, the successful probability is at most $1 - (\frac{1}{4})^{\log n/4} = 1 - \frac{1}{\sqrt{n}}$. So after $\frac{\log n}{4}$ timeslots, the probability that there is at least one successful transmission between any pair of nodes in $\{u, v, w\}$ is at most $1 - \Omega(\frac{1}{\sqrt{n}})$, which completes the proof. ■

VI. CONCLUSION

In this paper, we present the first randomized distributed multiple-message broadcast algorithm for *unstructured* wireless networks under the SINR model. Our algorithm can disseminate all messages to the whole network in $O((D+k)\log n + \log^2 n)$ timeslots with high probability. We also give an $\Omega(D+k+\log n)$ lower bound for any randomized algorithm. Although a harsher network model is assumed in this work, our algorithm turns out to be even faster than existing ones designed under the graph interference models [1], [11]. We believe this work can shed some light on how to deal with global interferences and asynchronous communications when designing distributed protocols for unstructured wireless networks.

There are some interesting and meaningful directions for future work. The first direction is to consider the case in which messages arrive at the network at arbitrary times. The second one is to design deterministic distributed protocols for the multiple-message broadcast problem under the SINR model. For the sake of energy efficiency, it will also be interesting to see if we can design distributed multiple-message broadcast algorithms under the SINR model with non-uniform power assignments, i.e., different nodes can use different powers.

VII. ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their very helpful comments. This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00302, the National Natural Science Foundation of China Grant 61073174, 61103186, 61202360, 61033001, 61061130540, and Hong Kong RGC-GRF grants 714009E and 714311.

REFERENCES

- [1] R. Bar-Yehuda, A. Israeli, and A. Itai. Multiple communication in multi-hop radio networks. *SIAM Journal on Computing*, 22:875-887, 1993.
- [2] K. Censor-Hillel, S. Gilbert, F. Kuhn, N.A. Lynch, C.C. Newport. Structuring unreliable radio networks. In *PODC*, 2011.
- [3] B.S. Chlebus, D.R. Kowalski, A. Pelc, M.A. Rokicki. Efficient Distributed Communication in Ad-Hoc Radio Networks. In *ICALP*, 613-624, 2011.
- [4] B. Chlebus, D. Kowalski, and T. Radzik. Many-to-Many Communication in Radio Networks. *Algorithmica*, 54(1):118-139, 2009.
- [5] J. Cichon, J. Lemiesz, M. Zawada. On Cardinality Estimation Protocols for Wireless Sensor Networks. In *ADHOC-NOW*, 2011.
- [6] O. Goussevskaia, T. Moscibroda, R. Wattenhofer. Local broadcasting in the physical interference model. In *DialM-POMC*, 2008.
- [7] Q.-S. Hua, F.C.M. Lau. Joint link scheduling and topology control for wireless sensor networks with SINR constraints. *Handbook of Research on Developments and Trends in Wireless Sensor Networks: From Principle to Practice*, IGI Global, pages 184-208, 2010.
- [8] S. Holzer, Y.A. Pignolet, J. Smula, R. Wattenhofer. Time-optimal information exchange on multiple channels. In *FOMC*, 2011.
- [9] R. M. Karp. Reducibility Among Combinatorial Problems. In *Proc. of a Symposium on the Complexity of Computer Computations*, 85-103, 1972.
- [10] M. Khabbazi, D.R. Kowalski. Time-efficient randomized multiple-message broadcast in radio networks. In *PODC*, 2011.
- [11] M. Khabbazi, F. Kuhn, D.R. Kowalski, N. A. Lynch. Decomposing broadcast algorithms using abstract MAC layers. In *DialM-PODC*, 2010.
- [12] M. Khabbazi, F. Kuhn, N. Lynch, M. Médard, and A. ParandehGheibi. MAC Design for Analog Network Coding. In *FOMC*, 2011.
- [13] F. Kuhn, N.A. Lynch, C.C. Newport. The abstract MAC layer. *Distributed Computing* 24(3-4):187-206, 2011.
- [14] F. Kuhn, N.A. Lynch, R. Oshman. Distributed computation in dynamic networks. In *STOC*, 2010.
- [15] F. Kuhn, T. Moscibroda, and R. Wattenhofer. Initializing Newly Deployed Ad Hoc and Sensor Networks. In *MOBICOM*, 2004.
- [16] C. Lenzen, T. Locher, P. Sommer, R. Wattenhofer. Clock Synchronization: Open Problems in Theory and Practice. In *SOFSEM*, 2010.
- [17] T. Moscibroda and R. Wattenhofer. Maximal Independent Sets in Radio Networks. In *PODC*, 2005.
- [18] L.G. Roberts. Aloha Packet System with and without Slots and Capture. *ACM SIGCOMM Computer Communication Review*, 5(2):28-42, 1975.
- [19] J. Schneider and R. Wattenhofer. Coloring Unstructured Wireless Multi-Hop Networks. In *PODC*, 2009.
- [20] D. Yu, Q.-S. Hua, Y. Wang, H. Tan, F.C.M. Lau. Distributed Multiple-Message Broadcast in Wireless Ad-Hoc Networks under the SINR Model. In *SIROCCO*, 2012.
- [21] D. Yu, Q.-S. Hua, Y. Wang, F.C.M. Lau. An $O(\log n)$ Distributed Approximation Algorithm for Local Broadcasting in Unstructured Wireless Networks. In *DCOSS*, 2012.
- [22] D. Yu, Y. Wang, Q.-S. Hua, F.C.M. Lau. Distributed $(\Delta + 1)$ -Coloring in the Physical Model. In *ALGOSENSORS*, 2011.
- [23] D. Yu, Y. Wang, Q.-S. Hua, and F.C.M. Lau. Distributed local broadcasting algorithms in the physical interference model. In *DCOSS*, 2011.