# Efficient Broadcasting and Gathering in Wireless Ad-Hoc Networks

Melih Onus and Andréa Richa *
Computer Science and Engineering Department
Arizona State University
PO Box 875406
Tempe, AZ 85287, USA
{Melih.Onus,aricha}@asu.edu

Kishore Kothapalli and Christian Scheideler †
Department of Computer Science
Johns Hopkins University
3400 N. Charles Street
Baltimore, MD 21218, USA
{kishore, scheideler}@cs.jhu.edu

## Abstract

*This paper considers the problem of broadcasting and information gathering in wireless ad-hoc networks, i.e. in wireless networks without any infrastructure in addition to the mobile hosts. Broadcasting is the problem of sending a packet from a source node in the network to all other nodes in the network. Information gathering is the problem of sending one packet each from a subset of the nodes to a single sink node in the network. Most of the proposed theoretical wireless network models oversimplify wireless communication properties. We will use a model that takes into account that nodes have different transmission and interference ranges, and we propose algorithms in this model that achieve a high time and work-efficiency. We present algorithms for broadcasting a single or multiple message(s), and for information gathering. Our algorithms have the advantage that they are very simple and self-stabilizing, and would therefore even work in a dynamic environment. Also, our algorithms require only a constant amount of storage at any host. Thus, our algorithms can be used in wireless systems with very simple devices, such as sensors.*

## 1. Introduction

In this paper we consider the problem of broadcasting and gathering messages in wireless ad-hoc networks. Broadcasting is a basic communication primitive for wireless networks, and it has therefore been heavily studied both in the systems and in the theory community. Though broadcasting itself appears to be an easy problem, it is actually quite hard to realize in an efficient and reliable way in a mobile ad-hoc network. The main problem concerning theoretical investigations is that mobile ad-hoc networks have many features that are hard to model in a clean way. Major challenges are how to model wireless communication and how to model mobility. Here, theoretical work is still rare. So far, people in the theory area have mostly looked at static wireless systems (i.e. the wireless units are always available and do not move). Wireless communication is usually modeled using the packet radio network model. In this model, the wireless units, or nodes, are represented by a graph, and two nodes are connected by an edge if they are within transmission range of each other. Transmissions of messages *interfere* at a node if at least two of its neighbors transmit a message at the same time. A node can only receive a message if it does not interfere with any other message(s).

The packet radio network model is a simple and clean model that allows one to design and analyze broadcast algorithms with a reasonable amount of effort. However, since it is a high-level model, it does have some serious problems with certain scenarios in practice. For example, in reality it is not true that the transmission range of a node is the same as its interference range. Instead, the interference range of a node is usually at least twice as large as its transmission range. Not taking this into account may result in broadcasting algorithms that cannot handle certain scenarios well although efficient on paper. In fact, it is not difficult to construct examples (see [16]), where most existing protocols for broadcasting require $\Omega(n)$ rounds even in expectation when we consider the situation that the interference range is bigger than the transmission range.

Thus it is necessary that algorithms for broadcasting in wireless networks consider problems due to interference. There is a limited number of papers that use a model that differentiates between the transmission range and interference range [1, 7, 8], but they assume that nodes are distributed in an ideal space so that the transmission range and interference range of every node can be specified in terms of Euclidean distance.

Another serious limitation in most of the existing algorithms is the assumption that the size of the network, or at

* Supported by NSF CAREER grant CCR-9985284.
† Supported by NSF grants CCR-0311121 and CCR-0311795.


1087-4089/05 $20.00 © 2005 IEEE
IEEE COMPUTER SOCIETY
Authorized licensed use limited to: IEEE Xplore. Downloaded on December 19, 2008 at 00:55 from IEEE Xplore. Restrictions apply.

least a linear estimate of the size of the network, is available to all of the nodes in the network. Without an estimate of the size of the network it was shown in [11] that in an $n$ node network, $\Omega(n)$ time units are required in expectation for a single message to be sent successfully, if physical carrier sensing is not available.

We will use a much more general wireless communication model that recently appeared in [17]. In this work we present self-stabilizing algorithms for broadcasting and information gathering in wireless overlay networks. To keep this paper at a reasonable length, we do not give a detailed motivation for the model adopted but instead refer the interested reader to [17].

### 1.1. Wireless communication model

We assume that we are given a set $V$ of mobile stations, or *nodes*, that are distributed in an arbitrary way in a 2-dimensional Euclidean space. For any two nodes $v, w \in V$ let $d(v, w)$ be the Euclidean distance between $v$ and $w$. Furthermore, consider any cost function $c$ with the property that there is a fixed constant $\delta \in [0, 1)$ so that for all $v, w \in V$,

- $c(v, w) \in [(1 - \delta) \cdot d(v, w),\ (1 + \delta) \cdot d(v, w)]$ and
- $c(v, w) = c(w, v)$, i.e. $c$ is symmetric.

$c$ determines the transmission and interference behavior of the nodes and $\delta$ bounds the non-uniformity of the environment. Notice that we do not require $c$ to be monotonic in the distance or to satisfy the triangle inequality. This makes sure that our model even applies to highly irregular environments.

We assume that the nodes use some fixed-rate power-controlled communication mechanism over a single frequency band. When using a transmission power of $P$, there is a transmission range $r_t(P)$ and an interference range $r_i(P) > r_t(P)$ that grow monotonically with $P$. The interference range has the property that every node $v \in V$ can only cause interference at nodes $w$ with $c(v, w) \leq r_i(P)$, and the transmission range has the property that for every two nodes $v, w \in V$ with $c(v, w) \leq r_t(P)$, $v$ is guaranteed to receive a message from $w$ sent out with a power of $P$ (with high probability) as long as there is no other node $v' \in V$ with $c(v, v') \leq r_i(P')$ that transmits a message at the same time with a power of $P'$.

For simplicity, we assume that the ratio $r_i(P)/r_t(P)$ is a fixed constant greater than 1 for all relevant values of $P$. This is not a restriction because we do not assume anything about what happens if a message is sent from a node $v$ to a node $w$ within $v$'s transmission range but another node $u$ is transmitting a message at the same time with $w$ in its interference range. In this case, $w$ may or may not be able to receive the message from $v$, so any worst case may be assumed in the analysis. The only restriction we need, which is important for any overlay network algorithm to eventually stabilize, is that the transmission range is a sharp threshold. That is, beyond the transmission range a message cannot be received any more (with high probability).

Nodes can not only send and receive messages but also perform physical carrier sensing, which has not been considered before in models proposed in the algorithms community. Given some sensing threshold $T$ (that can be flexibly set by a node) and a transmission power $P$, there is a *carrier sense transmission (CST) range* $r_{st}(T, P)$ and a *carrier sense interference (CSI) range* $r_{si}(T, P)$ that grow monotonically with $T$ and $P$. The range $r_{st}(T, P)$ has the property that if a node $v$ transmits a message with power $P$ and a node $w$ with $c(v, w) \leq r_{st}(T, P)$ is currently sensing the carrier with threshold $T$, then $w$ senses a message transmission (with high probability). The range $r_{si}(T, P)$ has the property that if a node $v$ senses a message transmission with threshold $T$, then there was at least one node $w$ with $c(v, w) \leq r_{si}(T, P)$ that transmitted a message with power $P$ (with high probability). More precisely, we assume that the monotonicity property holds. That is, if transmissions from a set $U$ of nodes within the $r_{si}(T, P)$ range cause $v$ to sense a transmission, then any superset of $U$ will also do so. For simplicity, we will assume in the following that for the carrier sense ranges, $r_{si}(T, P)/r_{st}(T, P) = r_i(P)/r_t(P)$ for all relevant values of $T$.

### 1.2. Related work

Broadcasting in wireless ad-hoc networks has been extensively studied in the literature, especially in the more applied ad-hoc networking community. See [18] for a survey. To the best of our knowledge, our paper is the first work that formally develops and analyzes broadcast algorithms under a model with separate transmission and interference ranges.

All of the work on the broadcast problem cited below assume a static network scenario where the transmission and interference ranges of a node are the same. In an early work, Chlamtac and Weinstein [3] presented a deterministic centralized broadcast protocol which assumes complete knowledge of the network topology and which runs in $O(D \log^2 n)$ time, where $n$ is the number of nodes and $D$ is diameter of the network. Bar-Yehuda et al. [2] were the first to present a distributed algorithm for the broadcasting problem in ad-hoc wireless networks without assuming any topological knowledge, except immediate neighborhood, of the network. Their algorithm had expected $O(D \log n + \log^2 n)$ time. In [14] a lower bound of $\Omega(D \log{(n/D)})$ is shown for any randomized broadcast protocol. In [13, 4] randomized protocols with expected runtime of $O(D \log{(n/D)} + \log^2 n)$ are presented — in [13] the underlying topology is assumed to be symmetric, while in [4] this assumption is dropped.

Adler and Scheideler [1] present approximation algorithms for the unicast problem in wireless ad-hoc networks

Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'05)
1087-4089/05 $20.00 © 2005 IEEE

IEEE COMPUTER SOCIETY
Authorized licensed use limited to: IEEE Xplore. Downloaded on December 19, 2008 at 00:55 from IEEE Xplore. Restrictions apply.

under the assumption that the transmission and interference ranges are not the same. However they still assume a simplified disk model based on Euclidean distances. Moreover, their unicast algorithm does not translate directly into an efficient broadcasting algorithm.

The problem of information gathering in wireless networks is studied mostly in the context of wireless sensor networks. The authors of [10] construct a tree on which gathering and aggregation can be performed. However, they do not deal with inherent problems such as channel contention and interference and also do not provide theoretical bounds on time and work. Information gathering and aggregation have also been studied in [5, 12, 15, 19, 9] but a rigorous formal analysis for wireless ad hoc networks has not been performed prior to this work.

### 1.3. Our results

We consider two important communication problems in wireless ad hoc networks, namely, broadcasting and information gathering.

The problem of broadcasting can be described as follows. Given a static connected wireless network of $n$ nodes, minimize the total time and work to send $m \geq 1$ broadcast messages originating from a source node $s$ to all nodes in the network. In Section 3, we consider the simple case where a single node $s$ is the source of a single broadcast message, i.e., $m = 1$. In Section 4, we extend our algorithm to handle the case that node $s$ is the source of multiple broadcast messages.

Information gathering is another important communication primitive in wireless networks. The problem has applications in many scenarios in sensor networks [10, 19, 6], and maintaining connectivity with base stations in a multihop wireless network. The problem of information gathering can be described as follows. Given a static connected wireless network of $n$ nodes among which $m$ packets are arbitrarily distributed and a sink node $s$ in the network, minimize the total time and work required for sending the $m$ packets to the sink node. In Section 5, we present and analyze a simple strategy for information gathering.

Our algorithms are self-stabilizing (i.e., can start in an arbitrary state) and can therefore adapt to changes in a wireless ad-hoc network. Our algorithms do not require any knowledge of the size of the network. For our algorithms to work correctly, it suffices that the nodes in the network have identifiers that are locally different. We only require that the nodes synchronize locally into rounds up to some reasonably small time difference, which can be easily accomplished using GPS signals or any form of beacons. Another important feature of our algorithms is that a constant amount of storage at any node suffices even in the case of gathering. The above properties make our algorithms applicable to sensor networks without any modifications.

The proofs are omitted in this version but can be found in [16]. Before proceeding further, in Section 2 we present some preliminary definitions and assumptions used in the paper.

## 2. Preliminaries

Our results build on top of a distributed algorithm for organizing the wireless nodes into a constant density spanner, proposed recently in [17]. A constant density spanner is defined as follows. Given an undirected graph $G = (V, E)$, a subset $U \subseteq V$ is called a *dominating set* if all nodes $v \in V$ are either in $U$ or have an edge to a node in $U$. A dominating set $U$ is called *connected* if $U$ forms a connected component in $G$. The *density* of a dominating set is the maximum over all nodes $v \in U$ of the number of neighbors that $v$ has in $U$. In our context, *constant density spanner* is a connected dominating set $U$ of constant density with the property that for any two nodes $v, w \in V$ there are two nodes $v', w' \in U$ with $\{v, v'\} \in E$, $\{w, w'\} \in E$, and a path $p$ from $v'$ to $w'$ along nodes in $U$ so that the length of $p$ is at most a constant factor larger than the distance between $v$ and $w$ in $G$.

Let $V$ be the set of nodes in the network. For any transmission range $r$, let the graph $G_r = (V, E)$ denote the graph containing all edges $\{v, w\}$ with $c(v, w) \leq r$. Throughout this paper, $r_t$ denotes the transmission range and $d(u, v)$ denotes the shortest distance between $u$ and $v$ in $G_{r_t}$. Furthermore, let $D(s) = \max_{v \in V} \ d(s, v)$.

The spanner protocol for $G_{r_t}$ presented in [17] consists of three phases that are continuously repeated in rounds as shown in Figure 1. The task of Phase I is to obtain a set $U \subseteq V$ of *active* nodes so that $U$ forms a constant density dominating set in $G_{r_t}$. As $U$ may not be connected, additional phases are required to arrive at the constant density spanner. The task of Phase II is to arrange nodes in $U$ into color classes that keep nodes with the same color sufficiently far apart from each other. Only a constant number of different colors is needed for this, where the constant depends on $\delta$ as defined in Section 1.1. Every node organizes its rounds into time frames consisting of as many rounds as there are colors, and a node in $U$ only becomes active in Phase III in the round corresponding to its color, also referred to as the round *owned* by that active node. The task of Phase III is to interconnect nodes in $u, v \in U$ such that $d(u, v) \leq 3$ via a set $\mathcal{G}$ of *gateway nodes*. (See Figure 2). Each phase has a constant number of time slots associated with it, where each time slot represents a communication step as shown in Figure 1. To achieve interference free communication among the active nodes, the coloring obtained in Phase II is used. Nodes in $V \setminus (U \cup \mathcal{G})$ are referred to as *inactive* nodes and the constant $\hat{d}$ refers to the number of active nodes that are within the interference range of any node.

Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'05)
1087-4089/05 $20.00 © 2005 IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on December 19, 2008 at 00:55 from IEEE Xplore. Restrictions apply.

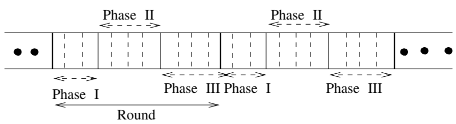


**Figure 1. Timeline of the spanner protocol.**

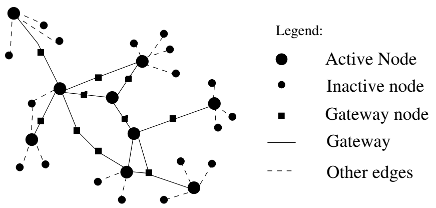


**Figure 2. Constant density spanner**

In [17], it was shown that such a spanner can be constructed in $O(\Delta \log \Delta \log n + \log^4 n)$ time steps, with high probability, where $\Delta$ is the maximum number of nodes that are within the transmission range of a node.

## 3. Isolated Broadcasting

In this section we consider the problem of broadcasting a single message. Let node $s$ be the source of the broadcast message. Since $s$ has a maximum distance of $D(s)$ to any node in $G_{r_t}$, $D(s)$ is a lower bound on the time an optimal offline algorithm needs to broadcast a message from $s$ to all nodes. Our goal is to come up with a broadcast scheme so that the time needed by the broadcast message to reach all nodes is as close to $D(s)$ as possible. We use the constant density spanner construction of [17] as the basis. If $s$ is not an active node, i.e., $s \notin U$, then let $\ell$ be some active node that is within the transmission range of $s$. Then $s$ first sends the message to $\ell$. The broadcast scheme then proceeds in rounds that are synchronized among the nodes. In the broadcast scheme below, $\ell$ refers to the ID of an active node that owns the current slot. Every item below is a separate time step.

1. If $\ell$ received the broadcast message in the previous round and it is the first time it received the broadcast message, $\ell$ sends out the broadcast message.
2. If $v$ is a gateway node and has already received the broadcast message, then $v$ sends out an RTS (Request-To-Send) signal with probability $p$.
3. If $v$ is a gateway node and decided not to send out an RTS signal or $v$ is an active node, then $v$ checks if it correctly received an RTS signal. If so, and $v$ has not received the broadcast message yet, $v$ sends out a CTS (Clear-To-Send) signal.
4. If $v$ is a gateway node and sent out an RTS signal, then $v$ checks if it sensed a CTS signal. If so, $v$ sends out the broadcast message.

Notice that inactive nodes just need to listen to the wireless channel in order to receive the broadcast message eventually. This is because our spanner algorithm [17] makes sure that message transmissions of active nodes in step 1 above never interfere at an inactive node. The following theorems demonstrate that the above protocol has a high time and work efficiency. We neglect the cost for sending and sensing the RTS/CTS signals in arriving at the work bound.

**Theorem 3.1** *Given the constant density spanner of $G_{r_t}$ as in [17], the broadcast algorithm with $p = 1/\hat{d}$ needs $O(D(s) + \log n)$ rounds, with high probability, to deliver the broadcast message to all nodes.*

**Theorem 3.2** *Given the constant density spanner of $G_{r_t}$ as in [17], the broadcast algorithm needs $O(W(s))$ work, where $W(s)$ is the optimal work required to send a broadcast message from $s$ to all nodes.*

The broadcast algorithm can also be made to self-stabilize by making simple changes to the algorithm above as shown in [16].

## 4. Broadcasting Multiple Messages

Next we look at the case that the source $s$ wants to send out multiple broadcast messages instead of just one. Then $s$ attaches continuous sequence numbers to the messages, starting with 1.

The broadcast scheme proceeds in rounds that are synchronized among the nodes. Each active or gateway node $v$ keeps track of two numbers, $i_v$ and $j_v$. Number $i_v$ denotes the minimum message number $v$ has not received so far and number $j_v$ denotes the minimum message number ($v$ knows about since its last successful transmission attempt) a node of distance at most $r_t$ from $v$ has not received so far. In the broadcast scheme below, $\ell$ refers to the ID of an active node that owns the current slot. Initially, for each gateway and active node $v$, $i_v$=$j_v$=1. In each round, every node $v \neq s$ does the following. Each item below represents a separate time step.

1. If $\ell$ received the broadcast message with sequence number $i' = i_\ell$ in the previous round, then it sets $i_\ell = i_\ell + 1$ and sends out the broadcast message with sequence number $i'$.

   If $v$ is a gateway node and received a broadcast message with sequence number $i' = i_v$, then it sets $i_v = i_v + 1$.
2. If $v$ is an active or gateway node, then it sends out an (RTR, $i_v$) message (RTR means "ready-to-receive") with probability $p$. If $v$ decides not to send out an RTR message, it checks whether it is able to receive an (RTR, $i'$) message. If so, it sets $j_v = \min\{j_v, i'\}$.



IEEE COMPUTER SOCIETY

Authorized licensed use limited to: IEEE Xplore. Downloaded on December 19, 2008 at 00:55 from IEEE Xplore. Restrictions apply.

3. If $v$ is a gateway node and $i_v > j_v$, then it sends out an (RTS, $j_v$) message with probability $p$.
   If $v$ is a gateway node and decided not to send out an RTS message or $v$ is an active node, then $v$ checks if it correctly received an (RTS, $j'$) message with $j' = i_v$. If so, $v$ sends out a CTS signal.
4. If $v$ is a gateway node and sent out an (RTS, $j_v$) message, then $v$ checks if it sensed a CTS signal. If so, $v$ sends out the broadcast message with sequence number $j_v$. Afterwards, $v$ sets $j_v = \min\{j_v + 1, i_v - 1\}$.
   If $v$ is a gateway node and did not send a message but received a broadcast message with sequence number $i' = i_v$, then it sets $i_v = i_v + 1$.

The source node $s$ uses the same protocol as above with the only difference that it only executes the first step. The inactive nodes just need to listen to the wireless channel in order to receive the broadcast messages eventually. The following theorems demonstrate that this protocol has a high time and work efficiency.

**Theorem 4.1** *Given the constant density spanner of $G_{r_t}$ as in [17], the concurrent broadcast algorithm with $p = 1/2\hat{d}$ needs $O(D(s) + m + \log n)$ rounds, with high probability, to deliver $m$ broadcast messages to all nodes.*

**Theorem 4.2** *Given the constant density spanner of $G_{r_t}$ as in [17], the broadcast algorithm needs $O(W(s,m))$ work, where $W(s,m)$ is the optimal work required to send $m$ broadcast messages from $s$ to all nodes.*

The above protocol also can be made to self-stabilize and the details can be found in [16].

## 5. Information Gathering

We now consider the situation where a total of $m$ packets distributed in an arbitrary way among the nodes in the wireless network are to be delivered to a sink node $s$ in the network. Firstly, note that $\Omega(m + D(s))$ is a lower bound on any solution for the information gathering problem. In the following, we describe a 2-stage protocol to perform information gathering efficiently. Each stage has a constant number of reserved time slots, 4 slots for stage 1 and 4 slots for stage 2.

### 5.1. Stage 1: Building Gathering Tree $T(s)$

We first show how to build the gathering tree rooted at $s$. All internal nodes in this tree will belong to $U \cup \mathcal{G}$. The sink node $s$, if it is not in $U \cup \mathcal{G}$, selects an active node $\ell$ such that $d(\ell, s) = 1$ and sends a route packet to $\ell$ with sequence number of 0 of the form $\langle \text{ROUTE}, s, 0\rangle$. The rest of the nodes do the following. Initially, $d'(s,v) = \infty, \pi(v) = \text{NULL}\ \forall v \in U \cup \mathcal{G}$. Let $d'(s,s) = 0$ and $\pi(s) = s$.

1. If $u \in U \cup \mathcal{G}$ receives a message $\langle \text{ROUTE}, v, d'(s,v)\rangle$ from $v$ with a sequence number of $d'(s,v)$ and if $d'(s,v) + 1 < d'(s,u)$, then $u$ sets $\pi(u) = v$ and $d'(s,u) = d'(s,v) + 1$. Node $u$ also sets flag$(u) = 1$ in this case, indicating that $u$ has to send a route message since $u$ updated its predecessor. If $u \notin U \cup \mathcal{G}$ and $d'(s,v) + 1 < d'(s,u)$ and $v \in U$, then $u$ updates $\pi(u) = v$ and $d'(s,u) = d'(s,v) + 1$.
2. If $u \in U \cup \mathcal{G}$ and $d'(s,u) \neq \infty$ and flag$(u) = 1$ then $u$ sends an RTS signal with probability $p$ to be determined later.
3. If $u \in U \cup \mathcal{G}$ and $u$ received an RTS signal then $u$ sends a CTS signal.
4. If $v \in U \cup \mathcal{G}$ and $v$ sent an RTS signal and receives a CTS signal then $v$ sends a route packet $\langle \text{ROUTE}, v, d'(s,v)\rangle$ and sets flag$(v) = 0$ signifying that the update has been notified.

Set $T(s) = (V_T, E_T)$ with $V_T = V$ and $E_T = \{(v, \pi(v)) | v \in V_T\}$ where $\pi(v)$ is set as in step (1) of the above protocol. Due to the properties of the spanner [17], for the above construction it holds that $\max_{v \in V} d_{T(s)}(s,v) \leq 5 \max_{v \in V} d(s,v)$. The following lemma can be shown [16].

**Lemma 5.1** *Given the constant density spanner of $G_{r_t}$ as in [17], to construct $T(s)$ the protocol given above with $p = 1/\hat{d}$, takes $O(D(s) + \log n)$ time steps w.h.p.*

### 5.2. Stage 2: Gathering on $T(s)$

In the gathering tree, $T(s)$, constructed in stage 1, each node has an unique path to the sink node $s$ via the predecessor pointers $\pi$. Nodes use this path system to eventually deliver packets to $s$.

The active node uses the first time slot to deliver packets and the second and third time slots are used to coordinate the actions of the inactive nodes. Nodes $\ell \in U \cup \mathcal{G}$ have a queue, $Q_\ell$, which can hold a constant number of packets. This queue works as a first-in-first-out list and supports operations *enqueue* and *dequeue* which add a packet and return a packet respectively to $Q_\ell$.

In the following when we refer to inactive nodes, it is implicit that we are referring to those inactive nodes that have a packet to send. Inactive nodes have a state among {`awake`, `asleep`}. Initially all inactive nodes are in the `asleep` state.

1. If $\ell$ is active and has a non-empty queue, then $\ell$ sends the packet *dequeue*$(Q_\ell)$ during the time slot owned by $\ell$. This packet has a destination $\pi(\ell)$ and nodes other than $\pi(\ell)$ discard the packet and $\pi(\ell)$ stores the packet by calling *enqueue* on $Q_{\pi(\ell)}$. In the second time slot, the active nodes listen to the channel.
   If $g$ is a gateway node and has a non-empty queue then $g$ sends an RTS message containing the id of $\pi(g)$ with probability $p$, where $p$ is to be determined later.

Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'05)
1087-4089/05 $20.00 © 2005 IEEE
IEEE COMPUTER SOCIETY
Authorized licensed use limited to: IEEE Xplore. Downloaded on December 19, 2008 at 00:55 from IEEE Xplore. Restrictions apply.

2. If $u \in U \cup \mathcal{G}$ and $Q_u$ is not full and $u$ receives an RTS message containing the id of $u$ then $u$ sends a CTS message.
   If $u$ is inactive and has a packet to send and $u$ is `awake` then $u$ sends an I-RTS *(for Inactive-RTS)* signal to $\pi(u)$ with a probability $1/2$.
3. If $g$ is a gateway node and sent an RTS signal in the previous time step and receives a CTS signal from $\pi(g)$ then $g$ sends the packet *dequeue*$(Q_g)$ to $\pi(g)$. This packet has a destination $\pi(g)$ and other nodes that receive the packet ignore it.
   If $\ell$ is active and $\ell$ receives an I-RTS signal from an inactive node $u$ then $\ell$ sends an I-CTS signal. If $\ell$ senses a busy channel but does not receive any I-RTS signal, then $\ell$ sends a collision message of the form $\langle \ell, \mathrm{COLLIDE} \rangle$. Otherwise if $\ell$ senses a free channel then $\ell$ sends a free message of the form $\langle \ell, \mathrm{FREE} \rangle$.
4. If $u$ is inactive and `asleep` and receives a free message then $u$ becomes `awake`. If $u$ is inactive and decided not to send an I-RTS message in the previous time step and $u$ is `awake` and receives a collision message then $u$ decides to go to `asleep` state with probability $1/2$. If $u$ is inactive and sent an I-RTS in the earlier step and gets an I-CTS then $u$ sends the packet to $\pi(u)$.

Let $\Delta_m$ denote the density of nodes that have a packet to send (note that $\Delta_m \leq m$). The following theorems demonstrate that the gathering protocol described above is efficient in terms of the time and work. The work performed while sending the RTS/CTS signals and the I-RTS/I-CTS signals is ignored while arriving at the work bound.

**Theorem 5.2** *Given the constant density spanner of $G_{r_t}$ as in [17], and a gathering tree $T(s)$ with sink node $s$, the information gathering algorithm presented above with $p = 1/\hat{d}$ needs $O(m + \Delta_m \log n \log \Delta_m + D(s) + \log n)$ time steps w.h.p so that all the $m$ packets reach the sink $s$.*

**Theorem 5.3** *Once a stable gathering tree has been constructed, the gathering protocol described above needs $O(W'(s))$ work, where $W'(s)$ is the optimal work required to send all the $m$ messages to the sink node $s$.*

Further, the algorithms for both the stages can be made to self-stabilize by making necessary changes as shown in [16].

## 6. Conclusions

In this paper, we use the realistic model for wireless communication of [17] to design algorithms for broadcasting and information gathering in wireless ad-hoc networks. The natural next steps would be to directly address node mobility and node faults, and to study more complex communication tasks such as anycasting and multicasting.

## References


[1] M. Adler and C. Scheideler. Efficient communication strategies for ad hoc wireless networks. *Theory of Computing Systems*, 33:337–391, 2000.

[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. *J. of Comp. and System Sci.*, 45:104–126, 1992.

[3] I. Chlamtac and O. Weinstein. The wave expansion approach to broadcasting in multihop radio networks. *IEEE trans. on commun.*, COM-39, 3:426–433, 1991.

[4] A. Czumaj and W. Rytter. Broadcasting algorithms in radio networks with unknown topology. In *IEEE FOCS*, 2003.

[5] E. J. Duarte-Melo and M. Liu. Data-gathering wireless sensor networks: organization and capacity. *Computer Networks*, 43(4):519–537, 2003.

[6] M. Enachescu, A. Goel, R. Govindan, and R. Motwani. Scale free aggregation in sensor networks. In *First Intl. W. on Algorithmic Aspects of Wireless Sensor Networks*, 2004.

[7] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Trans. Information Theory*, IT-46(2):388–404, 2000.

[8] P. Gupta and P.R. Kumar. Internets in the sky: The capacity of three dimensional wireless networks. *Communications in Information and Systems*, 1(1):33–50, 2001.

[9] Q. Han, S. Mehrotra, and N. Venkatasubramanian. Energy efficient data collection in distributed sensor environments. In *ICDCS*, pages 590–597, 2004.

[10] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *Mobile Computing and Networking*, pages 56–67, 2000.

[11] T. Jurdzinski and G. Stachowiak. Probabilistic algorithms for the wakeup problem in single-hop radio networks. In *Proc. of ISSAC*, pages 535–549, 2002.

[12] K. Kothapalli and C. Scheideler. Information gathering in adversarial systems: Lines and cycles. In *ACM SPAA*, 2003.

[13] D. Kowalski and A. Pelc. Broadcasting in undirected ad hoc radio networks. In *ACM PODC*, pages 73–82, 2003.

[14] E. Kushilevitz and Y. Mansour. An $\Omega(D \log(N/D))$ lower bound for broadcast in radio networks. *SICOMP: SIAM Journal on Computing*, 27, 1998.

[15] H. Luo and G. Pottie. Routing explicit side information for data compression in wireless sensor networks. In *DCOSS*, 2005.

[16] M. Onus, A. Richa, K. Kothapalli, and C. Scheideler. Efficient broadcasting and gathering in wireless ad-hoc networks. Available at http://www.cs.jhu.edu/∼kishore.

[17] M. Onus, A. Richa, K. Kothapalli, and C. Scheideler. Constant density spanners for wireless ad-hoc networks. In *ACM SPAA*, 2005.

[18] Brad Williams and Tracy Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In *ACM MOBIHOC*, pages 194–205. ACM Press, 2002.

[19] Y. Yu, B. Krishnamachari, and V. Prasanna. Energy-latency tradeoffs for data gathering in wireless sensor networks. In *IEEE Infocom*, 2004.




1087-4089/05 $20.00 © 2005 **IEEE**



IEEE COMPUTER SOCIETY

Authorized licensed use limited to: IEEE Xplore. Downloaded on December 19, 2008 at 00:55 from IEEE Xplore. Restrictions apply.