MAC Layer Support For Group Communication in Wireless Sensor Networks

Rong Zheng
Department of Computer Science
University of Houston
Houston, TX 77204
E-mail: rzheng@cs.uh.edu

Lui Sha
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
E-mail: lrs@cs.uiuc.edu

Department of Computer Science
University of Houston
Houston, TX, 77204, USA
http://www.cs.uh.edu

Technical Report Number UH-CS-05-14

July 21, 2005

Keywords: Group communication, WSNs, atomicity

Abstract

In this paper, we investigate the problem of providing efficient communication primitives across domains of wireless sensor network (WSN) applications. We argue both qualitative and quantitatively that group communication among sensors of geographic proximity is one of the basic building blocks of many WSN applications. Furthermore, group communication awareness needs to be embedded and implemented at the MAC layer due to the broadcast nature of wireless medium. We devise a MAC protocol, called LGC-MAC to enable efficient single-hop *one-to-many* and *many-to-one* communication. We present case studies of two example applications, acoustic target tracking and propagation of information with feedback using LGC-MAC and demonstrate that LGC-MAC can improve the response time, alleviate channel contention and provide better fault tolerance to packet collisions and wireless errors.

MAC Layer Support For Group Communication in Wireless Sensor Networks

Rong Zheng

Department of Computer Science
University of Houston
Houston, TX 77204
E-mail: rzheng@cs.uh.edu

Lui Sha

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
E-mail: lrs@cs.uiuc.edu

Abstract

In this paper, we investigate the problem of providing efficient communication primitives across domains of wireless sensor network (WSN) applications. We argue both qualitative and quantitatively that group communication among sensors of geographic proximity is one of the basic building blocks of many WSN applications. Furthermore, group communication awareness needs to be embedded and implemented at the MAC layer due to the broadcast nature of wireless medium. We devise a MAC protocol, called LGC-MAC to enable efficient single-hop *one-to-many* and *many-to-one* communication. We present case studies of two example applications, acoustic target tracking and propagation of information with feedback using LGC-MAC and demonstrate that LGC-MAC can improve the response time, alleviate channel contention and provide better fault tolerance to packet collisions and wireless errors.

Index Terms

Group communication, WSNs, atomicity

I. Introduction

Wireless sensor networks (WSNs) face challenges of constrained energy resource, limited communication and computational capability. Unlike general purpose data communication networks, a WSN is typically designed for a specific domain of applications. On one hand, application domain invariants allow us to co-design the protocol, sensor structure and the network architecture in a way that optimizes resource usage while achieving desirable features, such as robustness, timeliness and energy efficiency. On the other hand, layering and modular design are powerful means that allow easy integration of heterogeneous hardware devices as the technology advances and graceful system evolution to accommodate future applications as demonstrated by the success of the Internet. These two seemingly conflicting design philosophies motivate us to identify *reusable components that provide efficient communication primitives across domains of wireless sensor network applications*. Such primitives should be restrictive such that they account for the unique characteristics of WSNs and yet general enough to be applicable to various WSN applications.

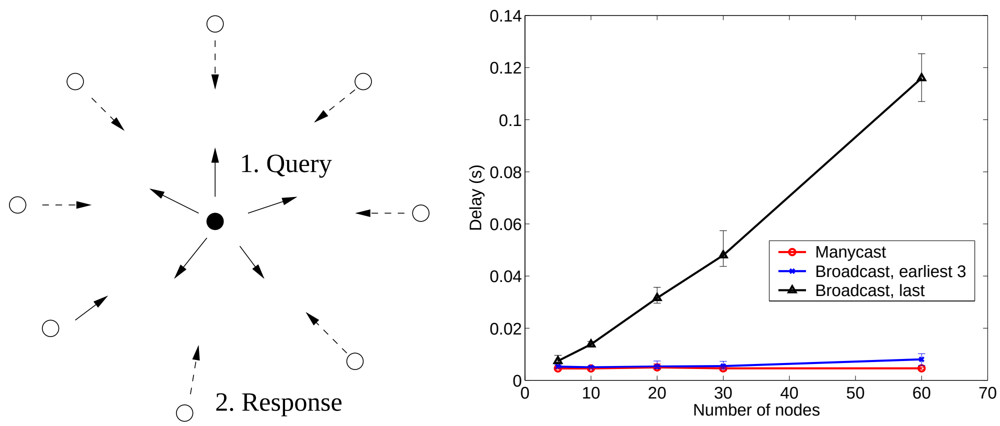
Individual sensors are constrained in computation power and storage space. Each sensor may only have partial information about its physical environment. However, when sensors are inter-connected, networks of sensors can potentially carry out sophisticated tasks such as environmental monitoring, intrusion detection, vehicle tracking, tele-monitoring of human physiological data. To self-organize the network in presence of node addition, failure, drainage of battery power and other unexpected environmental changes, *local group communication* among sensors in geographical proximity is of particular importance. For example, neighboring nodes can communicate with each other to select a leader [8] or distributedly determine their geographical locations with respect to anchor nodes with known locations [9]. Local group communication also makes possible the aggregation and compression of locally generated sensing data before it being transported to remote sinks.

In this paper, we propose an efficient MAC layer solution to support local group communication. Motivated by a case study on acoustic target tracking in WSN which can be decomposed into multiple instances of group communication with reliability requirements, we identify four atomic communication primitives that are common to many WSN applications: *1-to-null communication*, *1-to-1 communication*, *1-to-m communication* and *m-to-1* *communication*. We devise a MAC layer protocol, called LGC-MAC, for the later two primitives and demonstrate their advantages over traditional approach through both theoretical analysis and simulation study in *ns-2*. We further show two seemingly very different sensor network applications, namely, acoustic target tracking and propagation of information with feedback can both benefit from the richer set of primitives.

It is our belief that the proposed MAC layer support for local group communication can alleviate application developers and protocol designers from concerns about wireless link characteristics such as channel contention and interference while taking advantage of the broadcast nature of the wireless medium. Our key contributions in this paper are therefore best summarized as i) to identify the sets of building blocks for local group communication and ii) to devise MAC layer protocols to effectively support these primitives.

The rest of the paper is organized as follows. In Section II, we first motivate the need for new communication primitives using an example application in WSNs. The details of the protocol to support the proposed communication primitives are presented in Section III. A simple analytical model for the protocol is given in Section IV. In Section V, we evaluate LGC-MAC in ns-2 simulations. We conclude the paper with a review of related work in Section VI and summarize our findings in Section VII.

II. Motivation



(a) A query-response scenario (b) Comparison of response time

Fig. 1. A motivating example of a query-response process

To motivate our design, let us consider an acoustic tracking system in sensor networks developed in [14]. It consists of a number of sensor devices with low computation power as "slave" sensors and a few pc/104 single board computers acting as leaders. The leaders are responsible for carrying out computation-intensive signal processing algorithms for acoustic recognition and coordinating the sensors in locating interested acoustic signal. Upon detection of a sound of interest (e.g. sniper shot), a leader node broadcasts a query message to its slave sensors to solicit amplitude or delay measurements. The query message also serves the purpose of self-synchronization which triggers the slave sensors to process the most recent acoustic onsets and time-stamp their query responses accordingly. Triangulation is performed at the leader only when enough number (≥ 3) of responses have been received and thus the location of the sound source can be decided. The leader then transmits the localization results to a remote sink through multihop relays. Further details of this protocol can be found in [14].

The above localization process can be decomposed into three stages, i) a query broadcast from the leader to its one-hop slave sensors, ii) multiple responses from sensors to the leader and iii) hop-by-hop forwarding from the leader to the sink. Several observations can be made. First, to perform the triangulation, at least three distinctive responses are required. Moreover, the success of the second stage is contingent upon that of the first stage since if less than three sensors received the query, subsequent responses would be useless. This implies the need for reliable single-hop communication from one to multiple endpoints. Secondly, transmissions within the second stage are *fate-sharing* as opposed to contentious in nature, i.e., it is unimportant which of the slave sensors transmit its measurement first as long as at least three measurements are received at the end the stage. As far as localization is concerned, the order of the responses and the identity of sensors are irrelevant.

Based on these observations, we formally define local group communication as follows.

Definition 1 (Local Group Communication): Fate-sharing transmissions among nodes within the wireless transmission range that collectively perform a common function.

The group of fate-sharing transmissions are called a *transaction*. Group memberships are determined by the initiator of the transaction. In the acoustic localization example, multiple query responses constitute a single transaction.

To this end, we decompose communication in WSNs into a combination of the following building blocks,

- **1-to-null communication:** A single transmission from a node. Due to the broadcast nature of wireless medium, a single transmission may reach multiple nodes. However, as packets may get corrupted or fail to be received due to interference or capture effect, there is no guarantee that any of the neighboring nodes can correctly receive this packet (and thus the term "null").
- **1-to-1 communication:** A reliable transmission to a designated one-hop neighbor, which usually involves data-ACK handshakes.
- **1-to-m communication:** Reliable delivery of a single message to any or a predefined set of m neighbors.
- **m-to-1 communication:** Reliable delivery of messages from any or a predefined set of m neighbors to a common receiver.

Reliability in this context means, the sender has the knowledge that a message has been or not been received by the intended receiver(s). Since wireless transmissions are subject to collisions and errors, MAC layer protocols can only "best-effortly" deliver messages to other nodes within finite number of transmissions. The major challenge is to efficiently implement these primitives with low latency and minimal resource usage. To see why existing MAC protocols such as IEEE 802.11 can be a poor fit for local group communication, let us consider an example as shown in Figure 1 that models after the acoustic target tracking application.

In Figure 1(a), the center node broadcasts a query message to its neighbors. Upon receiving the query message, nodes send response messages subject to random backoffs as specified by IEEE 802.11 MAC. Figure 1(b) shows the delay to receive the query responses as the number of neighbors increases. The top two curves give the delay to receive all and first three responses at the center node, respectively. The bottom curve shows the result from using the local group communication-aware MAC protocol devised later in this paper. In this set of simulation, there are no other concurrent transmissions. As the number of neighbors grows, the latency increases in the broadcast case. Even though the latency to transmit the first three responses is comparable to that of the proposed protocol initially, as the number of nodes increase, due to channel contention, we observe an increasing trend for the former while the later remains the same. Longer response time is not only undesirable for applications that have real-time requirements, it may also result in higher energy cost. Notice that the "earliest 3" is just a hypothetical scheme introduced for comparison purposes. With current IEEE 802.11 MAC, the center node cannot stop the neighbor nodes from transmitting their responses right after enough responses have been received.

III. DESIGN OF MAC LAYER PROTOCOL TO SUPPORT LOCAL GROUP COMMUTATION

In this section, we present the detail of the local group communication-aware MAC protocol (LGC-MAC) for the afore-mentioned communication primitives in WSNs. Since *1-to-null* and *1-to-1* primitives can be implemented using broadcast and direct unicast with ACK respectively, we focus on the *1-to-m* and *m-to-1* primitives.

A. Preliminary

We consider a single channel WSNs employing slotted CSMA/CA type of medium access control scheme. The sensor RF circuits are capable of performing carrier sensing. Before any attempt for transmissions, a sensor first senses the medium and defers for a random period of time if the channel is detected idle. Similar to IEEE 802.11 MAC, in addition to physical carrier sensing, we use virtual carrier sensing to alleviate hidden terminal problems. In virtual carrier sensing, each data and control packet carry a duration field in their MAC headers. Upon overhearing the duration information of an ongoing transmission, a node defers its transmission by the designated amount of time regardless of the physical state of the wireless channel.

Message format: To support local group communication, we modify the data and ACK frame types to include the following fields,

- *mData:* a traffic indication map (TIM) is included in the data message header.

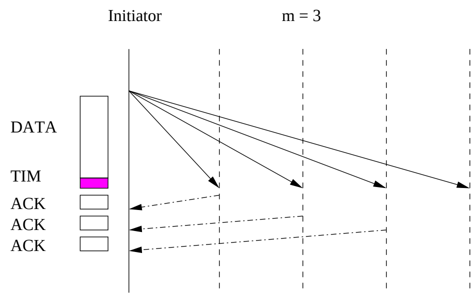


Fig. 2. Illustration of a successful *1-to-m* communication

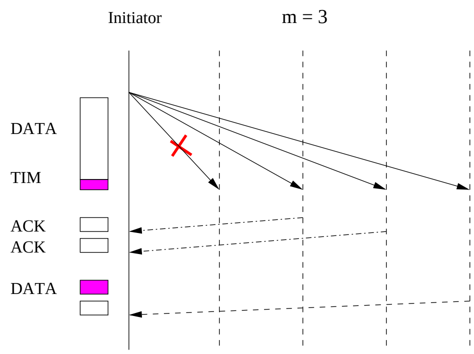


Fig. 3. Illustration of a scenario where some sensors failed to receive the data message. Loss is detected at the initiator. A data frame with no payload is transmitted to solicit further ACKs.

- *mACK:* a "next-to-send" field in the ACK frame indicates the next node to transmit.

We assume the network is semi-static, i.e., with low rate of mobility and failure. Therefore, each node has the knowledge of its neighboring set and its neighbors' neighboring sets in steady states (for example, via *1-to-null* local broadcast). Each node sorts its neighbors in ascending order of their IDs and assigns each neighbor an association identifier (NAID). The list of NAIDs are also exchanged along with the neighboring set.

A TIM in the data message is a tuple $(shift, mask)$. A mask is a bit vector consisting of 0s or 1s. The ith bit corresponds to the neighbor with $NAID = i + shift$ of the sender of the data message. A shift gives the offset of the first neighbor recorded in the mask. The combination of *shift* and *mask* allows grouping the neighbor set into blocks and selectively polling a subset of neighbors. For example, let node v's neighboring set be $\{8, 15, 21, 68, 74\}$. A TIM $(0, 10101_b)$ corresponds to node 8, node 21 and node 74 and a TIM of $(2, 11_b)$ corresponds to node 21 and node 74. TIM can also be used to indicate the ordering of expected responses as will become clearer later.

Addressing: To identify transmissions that belong to the same group, e.g., one of the *m* messages in the *m-to-1* communication, a transaction ID (TID) is generated by the initiator of the transaction. Uniqueness of the TID should be ensured locally. Furthermore, we propose to use multicast addresses (beginning with 0xF2) to distinguish local group communication from direct unicast or broadcast communication. The TID is ORed with the multicast address 0xF2000000 and used as the destination MAC address. Broadcast messages use the broadcast address 0xFFFFFFFF.

B. 1-to-m *Communication*

The goal of *1-to-m* is to transmit a message to any m nodes in a sender node i's neighbor set (of size d, $m \leq d$). We assume the group membership is determined by the sender (also called initiator) of the message either randomly or based on some priori knowledge of the corresponding neighbors (e.g., the link quality).

Next we describe the procedure for *1-to-m* communication. Upon receiving an application layer message with reliability requirements (e.g., how many responses are needed) and a locally generated TID, node i broadcasts an

Following the convention in IEEE 802.11 MAC, the MAC address is 48-bits long.

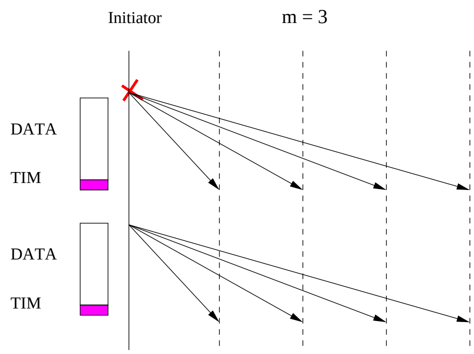


Fig. 4. Illustration of a scenario where the initiator failed to transmit the data message at its network interface. In this case, the initiator needs to retransmit the data frame.

$mData$ message with TIM field set to indicate the desired receivers. The duration field of the mData message is set to that of $T = DATA + m \cdot ACK$ (together with proper inter-frame spacings). Sensors that overhear the message but are not included in the TIM remain silent for T. Sensors who are the intended receivers (as indicated in the TIM) cache the $(TID, sender)$ and take turns to transmit $mACK$ frames (with *next-to-send* field set to the next neighbor on the TIM list) to node i. Before finishing transmitting its $mACK$ frame, a sensor cannot transmit/receive new data messages. The TID information is extracted from the MAC address of $mData$ at the receivers and passed to the application layer to assist future communications.

There can be three possible outcomes of the above procedure,

- *Case 1:* (Figure 2) By the end of T, all m $mACK$s have been successfully received.
- *Case 2:* (Figure 3) The data packet arrives only at k neighbors in the TIM list or only k out of m $mACK$s from neighbors on the TIM list have been received at the initiator, where $0 < k < m$. In both situations, since some $mACK$s have been received, the initiator knows that it has successfully transmitted the data packet on its network interface. However, some neighbors may fail to receive the data message or transmit an $mACK$ due to channel contention or wireless errors. If $d - m \geq m - k$, the initiator would transmit an mData packet with no payload with the corresponding TIM set to $m - k$ out of the remaining neighbors, where d is the node degree of the initiator. Duplicated data message can be detected using the sequence number in the packet header. Optionally, the sender can poll its neighbors in a probabilistic fashion to check if they have received the $mData$ message.
- *Case 3:* (Figure 4) If no ACK has been received by the end of T interval or there are not enough neighbors left to get m $mACK$s (Fig. 4), the sender retransmits the original data message. The TIM list in the message header can be the same or different as the previous one. The rationale for the later choice is, it is possible that sensors in the original TIM are in regions with poor reception.

In Case 2 or Case 3, retransmissions are performed up to a limited number of times before failure is declared and informed to the upper layer. To enhance the robustness of the protocol, each node can optionally summarize and piggyback the ID of sensors they have overheard $mACK$s from. Note that for $m = 1$, the protocol behaves exactly the same as a unicast single-hop communication. If $m = 0$, no acknowledgment is needed.

C. m-to-1 *Communication*

As defined in Section II, *m-to-1* communication involves delivery of messages from any or a pre-defined set of m neighbors to a common receiver, called a terminator. There are two possibilities. First, membership information has been established by the TIM field of an earlier message using *1-to-m* communication described in the previous section. The order of responses is also defined. Second, a node can solicit responses from its neighbor using a regular broadcast message (*1-to-null*). The broadcast message carries the TID information, which is later used to identify the grouping among multiple response messages. Clearly, the second case is more general and requires less state information at the neighboring sensors. In what follows, we consider the general case where the desired number of responses, membership and ordering information are not known ahead of time at the sensors. We assume

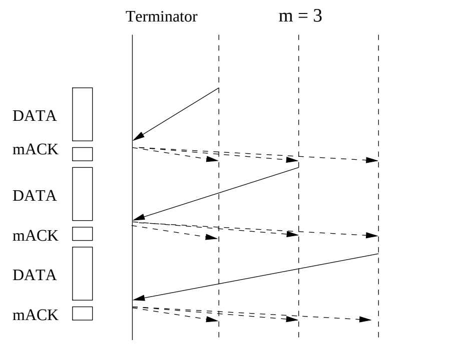


Fig. 5. Illustration of a successful *m-to-1* communication where m = 3.

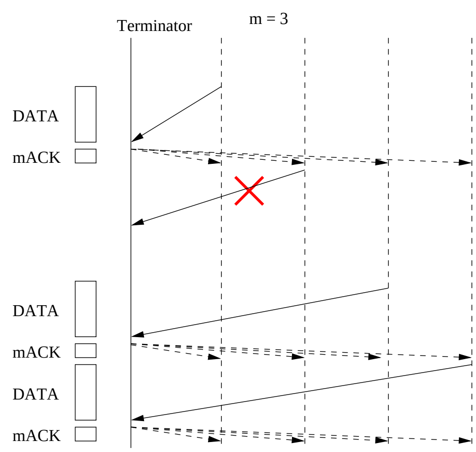


Fig. 6. Transmission from the second node is lost. The remaining nodes need to contend for the channel and re-establish the chain of data/mACK transmissions.

the processing time on sensor devices are similar and thus the responses are available roughly at the same time among all the nodes though the ordering and starting point are not know a priori.

Any sensor that has a response message ready to transmit participates in the channel contention. Upon the reception of the first data message from a neighbor, the terminator transmits an $mACK$ message to acknowledge reception of the message and the ID of the next sensor to transmit. This effectively polls the intended sensors one by one and synchronizes the transmissions. In the $mACK$ message, the duration field specifies the time duration to transmit the data frame and the subsequent $mACK$. On reception of such an $mACK$ message, the corresponding sensor cancels its backoff timer and transmits immediately. The process continues till the terminator receives m messages and indicates in the last $mACK$ that no more replies are needed. The process is illustrated in Fig. 5.

In the cases that sensors polled by the terminator do not have the interested data or their data frames are not received, ACK cannot be sent. Self-synchronization needs to be re-established. The remaining sensors need to restart the process by contending for the channel again. After the first successful delivered message, subsequent transmissions can be treated the same way as before (Fig. 6). A timer at the terminator is set to determine if the current procedure fails.

D. Implementation Considerations

The previous two sections define the message format and timing logic to support two types of local group communication. To make the protocol more resilient to channel contentions, several enhancements can be borrowed from the widely adopted IEEE 802.11 MAC. First, RTS-CTS can be used in precedence of transmissions of *mData* messages. Second, shorter inter-frame spaces for acknowledgment frames is desirable to reduce the collision probability. It should be noted, however, that LGC-MAC can be implemented in conjunction with other CSMA access schemes as well.

IV. Protocol Analysis

In this section, we develop a simple analytical model to study the performance of LGC-MAC in lossy wireless channels. For ease of presentation, we assume the p-persistent CSMA scheme is used for medium access, i.e., in a slotted system, a backlogged sender transmits with probability p in each slot. In [1], Cali *et al.* analyzes the throughput of IEEE 802.11 DCF. It is argued that p-persistent CSMA can approximate the behavior of contention window-based mechanism in IEEE 802.11 DCF with an appropriate choice of p.

A. 1-to-m *Communication*

To see the benefit of LGC-MAC, let us compare it against a baseline where all the acknowledgments are transmitted as direct unicast frames of length $\overline{S}$.

No wireless error: First, suppose there are no other sensors contending the medium and no wireless errors. From the results in [1], if there are M contending sensors in a single hop network and assume that each sensor chooses a random backoff window from a geometric distribution with parameter p, the average time to successfully transmit a frame of length $\overline{S}$ in the IEEE 802.11 MAC is given by Eq. (1).

$$t(M,p) = \frac{1-p}{Mp} t_{slot} + \frac{1-(1-p)^M - Mp(1-p)^{M-1}}{Mp(1-p)^{M-1}} \overline{S}, \tag{1}$$

where t_{slot} is the length of each slot.

Therefore, the total time to transmit the *mData* frame of length $\overline{L}$ and m acknowledgments (using regular data frames without ACKs) from d neighbors is,

$$T^{base}_{1\_to\_m} = \overline{L} + \sum_{k=1}^{k=m} t(d-k+1,p), \tag{2}$$

The second term is due to the fact that the number of backlogged sensors reduces by 1 once one has finished its transmission.

On the other hand, using LGC-MAC, we have

$$T^{proposed}_{1\_to\_m} = \overline{L} + m \cdot \overline{S}. \tag{3}$$

Clearly, $T^{base}_{1\_to\_m}$ is straightly greater than $T^{proposed}_{1\_to\_m}$. Note that the analysis errors on the pessimistic side since in the baseline protocol, neighbors continue transmitting the acknowledgment even though only m acknowledgments are needed.

With Bernoulli wireless error: The presence of wireless error will affect both protocols negatively. Consider a simple Bernoulli error model with probability q, i.e., all message transmissions are likely to be erroneous with probability q. For tractability of the analysis, we assume that messages are either successfully received or corrupted at all nodes. Suppose in the baseline, the initiator detects the loss of its data frame using a timeout τ. After τ, if no acknowledgment is received and the medium is idle, the initiator retransmits the data message and solicits acknowledgments again.

The conditional probability that less than *m* acknowledgments are received given that the data frame has been correctly received by all sensors is given by, $p_f = \sum_{l=0}^{m-1} C^d_l (1-q)^l q^{d-l}$, where C^d_l is the combinatoric number. The total time to transmit d acknowledgments (some of which are corrupted) is, $\sum_{k=1}^{k=d} t(d-k+1,p)$.

Therefore, Eq. (2) and Eq. (3) can be modified as,

$$\begin{aligned} T^{base}_{1\_to\_m}(p) &= \frac{1}{1-p_f}[\frac{\overline{L}+\tau}{1-q} \\ &+ \sum_{l=m}^{d} C^{l-1}_{m-1}(1-q)^m q^{l-m} \sum_{k=1}^{k=l} t(d-k+1,p) \\ &+ p_f \sum_{k=1}^{k=d} t(d-k+1,p)] \end{aligned} \tag{4}$$

and

$$T^{proposed}_{1\_to\_m} = \frac{\overline{L}+m\overline{S}}{(1-q)^{m+1}} \tag{5}$$

The first term in the parenthesis in Eq. (4) is due to the fact on average it takes $1/q$ (re)transmissions to send the data message successfully.

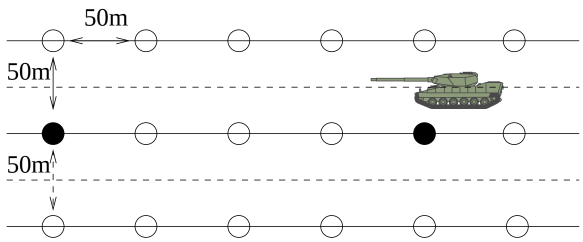


Fig. 7. Simulation Scenario for Acoustic Target Tracking. Leaders are marked solid.

B. m-to-1 *Communication*

No wireless error: Due to the use of virtual carrier sensing, collisions rarely happen in the middle of the process. Among M contending stations, the average time to transmit a response of length $\overline{L}$ and acknowledgment $\overline{S}$ is,

$$t(M,p) = \frac{1-p}{Mp} t_{slot} + \frac{1-(1-p)^M - Mp(1-p)^{M-1}}{Mp(1-p)^{M-1}} \overline{L} + \overline{S}. \tag{6}$$

Therefore, the total time to transmit m data frames among d neighbors is,

$$T^{base}_{m\_to\_1} = \sum_{k=1}^{k=m} t(d-k+1,p) \tag{7}$$

On the other hand, using the proposed protocol, we have

$$T^{proposed}_{m\_to\_1} = t(d,p) + (m-1)(\overline{L}+\overline{S}) \tag{8}$$

Clearly, $T^{proposed}_{m\_to\_1}$ is straightly less than $T^{base}_{m\_to\_1}$.

With Bernoulli wireless error: In the baseline, we have

$$T^{base}_{m\_to\_1} = \sum_{k=1}^{k=m} \frac{t(d-k+1,p)}{(1-q)^2}, \tag{9}$$

since retransmission of a data frame is needed either because of corruption of data or ACK.

For LGC-MAC, the response time needs to be computed recursively. Let $T(n,k)$ be the average time to successfully receive k distinctive responses from a total of n nodes under LGC-MAC. We have,

$$\begin{aligned} T(n,k) &= T(n,1) \\ &+ \sum_{l=1}^{l=k} (1-q)^{2(l-1)}(2q-q^2)[l(\overline{L}+\overline{S}) + T(n-l,k-l)] \end{aligned}$$

with $T(n,1) = \frac{t(n,p)}{(1-q)^2}$. Finally, $T^{proposed}_{m\_to\_1} = T(d,m)$.

V. Performance Evaluation

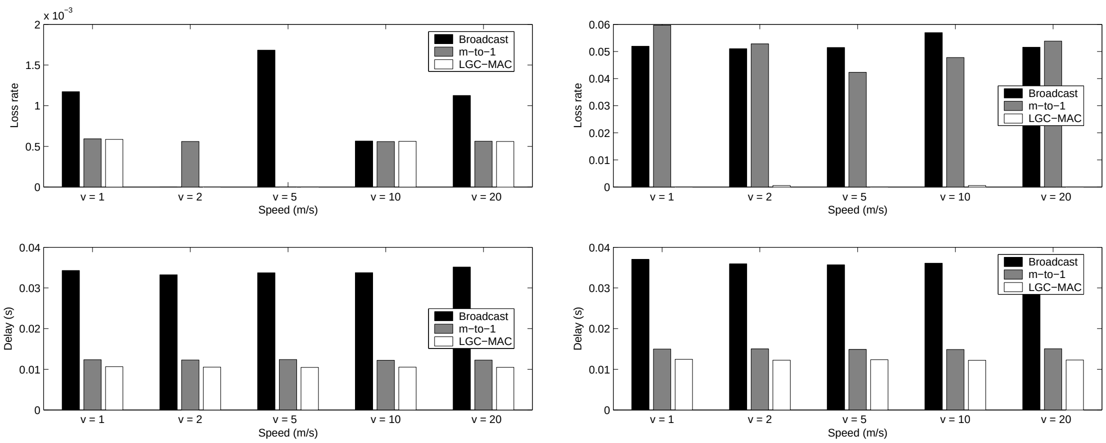
In this section, we study the performance of LGC-MAC using *ns-2* simulations. We demonstrate that two seemingly very different applications, namely, acoustic tracking and propagation of information with feedback (PIF) can be effectively implemented with the proposed primitives.

We modified the IEEE 802.11 MAC code in *ns-2* to implement LGC-MAC. The MAC header is enhanced with a TIM. Multicast addresses beginning with $0xF2$ are used to distinguish the two primitives from direct unicast (*1-to-1*) or broadcast (*1-to-null*) messages.

TABLE I

PARAMETERS USED IN THE SIMULATION

Wireless transmission range	250m
Number of background CBR connections	6
Rate of each CBR connections	0.1 packet/s
Packet size	128Byte
Spatial resolution	5m
Number of sensors	60
Number of leaders	5



(a) no wireless error (b) wireless error rate = 0.05

Fig. 8. Acoustic target tracking with and without wireless errors in 60x3 mesh networks

A. A Case Study for Acoustic Target Tracking

We implemented in ns-2 the acoustic tracking protocol proposed in [14]. The setup of the simulation is as follows. Sensors are placed along and in the middle of a two-way road. For simplicity, this is approximated by a M-by-3 mesh network (Figure 7). Adjacent sensors are separated by 50m. Leader nodes are placed in the middle of the road every 200m apart. As a target moves along the road, the leader node closest to the target broadcasts a query message to its neighbor sensors to gather acoustic readings. Once enough query responses (≥ 3) have been received, a leader node computes the location of the target and sends the result along a multi-hop path to the sink at the far end down the road. In the simulations, static routing is assumed as there is no mobility in this network. In addition to messages for localization results, background traffic in the network is simulated using several low bit-rate CBR connections. This can be used to model periodic monitoring events in the networks (e.g., temperature measurements, sensor management messages etc). The sampling rate or the frequency to broadcast query messages at the leaders is determined by the required spatial resolution and the speed of the targets. For example, if a target moves at the speed of v m/s and the required spatial resolution is l m, the sampling frequency should be at least l/v. The setting of parameters in the simulation is summarized in Table I.

We compare the performance of three schemes,

- **Broadcast query with 1-to-1 response:** (short as “broadcast”) the query message is transmitted using broadcast and there is no acknowledgment to determine whether the message has been successfully received. Query responses are transmitted using direct unicast messages.
- **Broadcast with m-to-1 response:** (short as “m-to-1”) The key difference is that only m responses are needed and are transmitted in an orderly fashion illustrated in Section III.
- **1-to-m query with m-to-1 responses:** (short as “LGC-MAC”) Query messages are acknowledged to ensure successful delivery to at least m neighbors.

Figure 8 gives the percentage of un-reported localization events at the sink and the time from onset of the acoustic event to receiving a report at the sink. The speed of the target varies from 1m/s to 20m/s. We simulate scenarios without and with 5% Bernoulli wireless errors respectively. The results shown are averages of ten simulation runs.

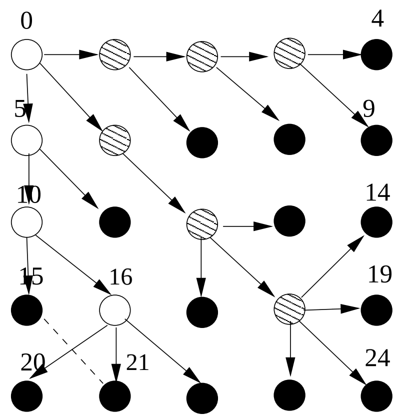
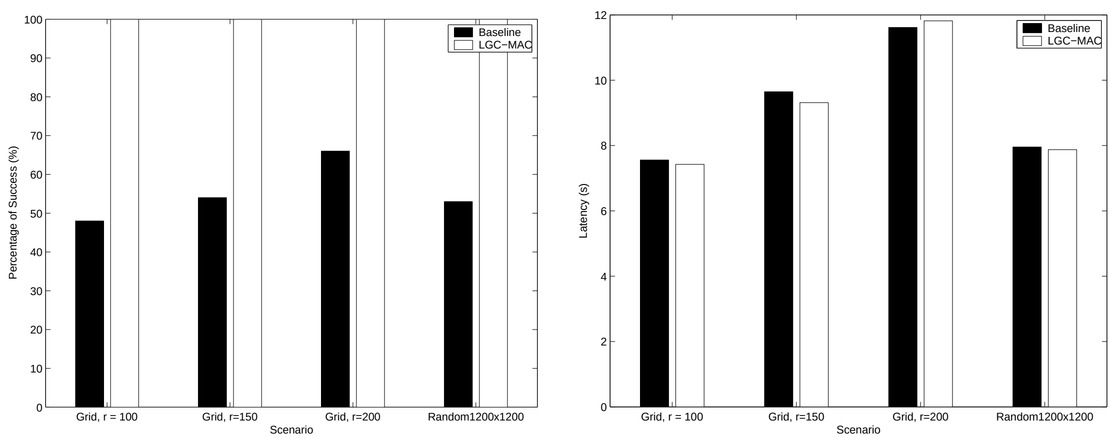


Fig. 9. An instance of Propagation of Information with Feedback. Node 0 fails to determine the termination of procedure although all nodes have received the message. Solid dark nodes are leaf nodes. Nodes that have sent a feedback message to their parents are shaded. The dashed link indicates collision of broadcast messages from node 15.



(a) Percentage of successful runs

(b) Latency

Fig. 10. Propagation of Information with Feedback. Numb. of nodes = 100, wireless transmission range = 250

In the simulations, losses of localization results come from three sources, i) failure to broadcast the query to enough number of sensors, ii) failure to gather enough responses from sensors and iii) failure to deliver the localization results to the sink.

From Figure 8(a), we can see that when there is no wireless error, the performance of “m-to-1” and LGC-MAC are comparable. Overall, the broadcast scheme incurs the highest loss rate and longest delay since multiple responses (often more than 3) for each query lead to high channel contention in the network. When there is no wireless error, most of the packet loss is due to failure to deliver the localization results to the sink. The latency of LGC-MAC is slightly smaller than “m-to-1”. This is because in “m-to-1”, all neighboring sensors need to contend the medium to send the first response message, while in LGC-MAC, group membership is announced in the “1-to-m” query message and thus less sensors contend to transmit the first response message.

In Figure 8(b), with wireless error rate of 0.05, both the “broadcast” and “m-to-1” scheme experience high loss rate (greater than 0.05) as query messages fail to reach enough number of sensors. In contrast, LGC-MAC has significantly less losses due to the use of reliable *1-to-m* communication. Due of retransmission of reports on the multihop path to the sink, all schemes incur a slightly higher delay compared to the case without wireless errors.

B. Propagation of Information with Feedback (PIF)

Next, we study the problem of an arbitrary node that has a message it wants to transmit to all the nodes in the network. In addition, the initiator node should be provided the knowledge of the propagation termination, i.e., it is ensured that all the nodes in the network have received the message.

PIF is a basic building block for distributed algorithms such as construction of spanning tree, leader election, construction of dominating set etc. Moreover, it can be used to implement code disseminating for software update, counting number of nodes in a distributed network.

The PIF algorithm described in [6] is as follows. A message is of the form: $MSG(target, l, parent)$, where $target$ specifies the target node or nodes. A null value in the target field indicates a broadcast to all neighboring nodes, and is used when broadcasting the message. The $parent$ field specifies the parent of the node that sends the message. The l field determines the sender's identity. The initiator, called the source node, broadcasts a message, thus starting the propagation. Each node, upon receiving the message for the first time, stores the identity of the sender from which it got the message as its parent and broadcasts the message. The feedback process starts at the leaf nodes, which are childless nodes on the virtual tree spanned by the PIF algorithm. A leaf node that is participating in the propagation from the source, and has received the message from all of its neighboring nodes, sends back an acknowledgment message, called a feedback message, which is directed to its parent node. A node that got feedback messages from all of its child nodes, and has received the broadcast message from all of its neighboring nodes sends the feedback message to its parent. The algorithm terminates when the source node gets broadcast messages from all of it neighboring nodes, and feedback messages from all of its child neighboring nodes. It can be trivially proved that, each node in the network sends one message during the propagation, and one message during the acknowledgment, to the total of two messages under the assumption of no message collision/losses.

In the PIF algorithm, the propagation stage constructs a spanning tree rooted at the source node. The set of leaf nodes may differ depending on the delay of (single hop) message transmission and channel contention in the network. Feedback messages travel and aggregate on the reverse path to the source node. Failure to receive a broadcast message from nodes other than one's children results in failure of the protocol. On the other hand, since feedback messages use direct unicast, they are less likely to get lost. Therefore, this protocol is sensitive to broadcast message collisions.

As an example, consider a 5x5 grid. The edge length of each grid are 200 meters. Given a wireless transmission range of 250 meters, each node has 4 neighbors except those at the corners. We implemented the above PIF algorithm in the *ns-2* simulator. Broadcast messages are jittered with a random delay uniformly distributed between 0 and 1 second before transmission. This is important to alleviate the broadcast storm problem [10]. Figure 9 gives the result of a single execution of the PIF algorithm with IEEE 802.11 DCF. In Figure 9, leaf node 21 failed to receive the broadcast message from node 15. Therefore, no acknowledgment message is transmitted to node 21's parent (node 16), which in turn cannot send acknowledgment message to node 10. In the end, the branch routed at node 5 fails produce an acknowledgment and thus the node 0 cannot determine the termination of the process.

Next, we apply the *1-to-m* primitive in the propagation stage. m is set to be a node's degree minus 1. All neighbors other than a node's parent need to acknowledge the message. Figure 10 compares the success rate of LGC-MAC and the baseline IEEE 802.11 DCF under four different topologies, 10x10 grid with edge length 200 meters, 150 meters, 100 meters and a random topology of size 1200mx1200m. The number of nodes in each scenario is set to be 100. Figure 10 (a) and (b) give, respectively, the percentage of successful runs and the latency to obtain the acknowledgment message at the source for each topology. The latency for IEEE 802.11 DCF is measured for successful runs only. As shown in Fig. 10 (a), the success rate in the baseline IEEE 802.11 DCF varies with the density of the network. The denser the network (e.g., with grid edge length of 100m, the maximum degree is 20), the more likely broadcast messages will be lost. In contrast, among all 100 runs, LGC-MAC always manage to get the feedback message successfully. This significant improvement in success rate is achieved with comparable delay performance as shown in Fig. 10 (b). It should be noted that the delay is dominated by the jitter (0.5s on average) to transmit the broadcast message at each hop. Delay incurred by transmitting additional ACKs in the proposed scheme is negligible.

We can envision many other distributed algorithms that have reliability requirements for single hop communication. Although PIF algorithm, or a distributed algorithm in general can be designed more resilient to errors, we believe LGC-MAC makes channel contention and wireless error "transparent" to upper layer. With LGC-MAC, it is reasonable to assume that with high probability, (broadcast) message can reliably reach intended receiver in finite time.

One comment is in order about the end-to-end argument. The end-to-end argument states that "many functions can only be completely implemented at the end points of the network, so any attempt to build features in the network to support particular applications must be viewed as a trade-off". In the PIF example, reliability cannot be fully implemented at the MAC layer. Retransmission at the source is still needed in cases when an acknowledge failed to arrive in a prolonged period of time. However, as demonstrated in Figure 10, by incorporating better reliability at MAC layer, the need for retransmission at end points of the network can be greatly reduced (and thus a desirable trade-off between MAC layer complexity and performance).

VI. Related Work

To the best of our knowledge, this is the first work that identify the common communication primitives among WSN applications and proposes a MAC layer solution to local group communication. However, the idea of providing multicast and manycast services in multihop wireless networks have been studied in literature under different contexts.

Several multicast routing protocols [3], [4], [11] have been proposed in literature to support one to multi-point communication across the networks. Reliable multicast is a much harder problem. The main challenges are handling of ACK explosion, mobility and temporary partition in the network. In [7], a hierarchical system based on a tree structure, Family ACK Tree (FAT), is proposed. Each node on the tree is responsible for the reliable transmission of packets to its downstream nodes so that the reliability charge is distributed. Each node on the tree temporarily caches the packets and keeps track of on-going traffic. In [2], manycast is defined as a group communication scheme that enables communication with an arbitrary (client specified) number of group members. The authors argue that manycast should be a network layer service. Several network-layer solutions to support manycast are investigated in [2], including scoped flood, unicast, small group multicast (SGM) and SGM-Broadcast. Both the network layer manycast and reliable multicast can potentially benefit from the new MAC layer primitives proposed in this paper. As the later provides reliability in single-hop transmissions to multiple end points.

A MAC layer anycast protocol is proposed in [5]. The key idea is to use MAC layer knowledge about the channel condition to break ties among “equally” good routes (from the perspective of network layers). The underlying assumption is forwarding a packet to any of the next hops is accepted to the network layer (and thus the name “anycast”). Anycast is essentially implemented using a single-hop unicast with its destination picked by the MAC layer. There does not exist “cooperative” relationship among neighboring nodes in forwarding messages as only one copy of message is forwarded. Instead, fate-sharing and cooperativeness are explored in the design of our group communication-aware MAC protocols. In LGC-MAC, selections of many-to-one senders and one-to-many receivers can also incorporate knowledge of link characteristics.

Two protocols to support reliable broadcast and manycast in ad hoc networks, Autograph and Scribble have been proposed in [12], [13]. The Autograph protocol achieves a similar goal as the PIF protocol in Section V-B with the exception that it is designed for a potentially more dynamic network with constrained node mobility. In Autograph, a node which has received the broadcast message periodically broadcasts the message until it learns that all nodes have received the message. Termination condition is determined by piggybacking a bit vector for all known nodes that have received the broadcast message. The Autograph protocol has better fault tolerance compared to the original PIF protocol in [6]. The message complexity result is not available for the Autograph protocol. We believe that the proposed MAC layer primitives is complementary (though not limited) to the Autograph protocol for the reliable broadcast problem and can reduce the number of transmitted messages. For static environments, as demonstrated in Section V, the PIF algorithm in conjunction with the LGC-MAC protocol has low complexity and roughly 100% success rate.

VII. Conclusion and Future Work

In this paper, we present LGC-MAC, an efficient MAC layer solution to support reliable one-to-many and many-to-one single-hop communication in WSNs. Fate-sharing transmissions that contend for a common wireless channel are grouped to transactions, which are scheduled in a coordinated manner. Both analysis and simulation studies show that the proposed primitives can indeed reduce the resource usage, shorten the response time and are more tolerant to wireless errors. Two example applications, though very different at first sight, demonstrate the power of proposed primitives.

As part of ongoing work, we are developing APIs for WSN applications to utilize these primitives. We are also interested in providing tools to automate implementation of distributed algorithms in WSNs.

References

[1] F. Cali, M. Conti, and E. Gregori. Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit. *IEEE/ACM Transactions on Networking*, 8(6), December 2000.

[2] C. Carter, S. Yi, P. Ratanchandani, and R. Kravets. Manycast: exploring the space between anycast and multicast in ad hoc networks. In *Proceedings of the 9th international conference on Mobile computing and networking*, pages 273–285. ACM Press, 2003.

[3] K. Chen and K. Nahrstedt. Effective location-guided tree construction algorithms for small group multicast in manet. In *Proc. of IEEE INFOCOM*, pages 1180–1189, 2002.

[4] C.-C. Chiang and M. Gerla. On-demand multicast in mobile wireless networks. In *Proc. of ICNP*, 1998.

[5] R. R. Choudhury and N. H. Vaidya. Mac-layer anycasting in ad hoc networks. *SIGCOMM Comput. Commun. Rev.*, 34(1):75–80, 2004.

[6] I. Cidon and O. Mokryn. Propagation and leader election in multihop broadcast environment. In *Proc. of 12th International Symposium on DIStributed Computing, (DISC98)*, pages 104–119, September 1998.

[7] M.-Y. Jiang and W. Liao. Family ack tree (fat) protocol for reliable multicast in mobile ad hoc networks. In *Proc. IEEE ICC*, 2002.

[8] F. Kuhn, R. Wattenhofer, Y. Zhang, and A. Zollinger. Constant-Time Distributed Dominating Set Approximation. In *Proc. 22nd ACM Int. Symposium on the Principles of Distributed Computing (PODC)*, 2003.

[9] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Comput. Networks*, 43(4):499–518, 2003.

[10] S. Y. Ni, Y. C. Tseng, Y. S. Chen, and J. P. Sheu. The broadcast storm problem in a mobile ad hoc network. In *Proceedings of the fifth annual ACM/IEEE international conference on Mobile computing and networking*, 1999.

[11] E. M. Royer and C. E. Perkins. Multicast ad hoc on-demand distance vector (maodv) routing. draft-ietf-manet-maodv-00, 2000.

[12] E. Vollset and P. Ezhilchelvan. An efficient reliable broadcast protocol for mobile ad-hoc networks. Technical Report CS-TR-822, University of Newcastle upon Tyne, 2003.

[13] E. Vollset and P. Ezhilchelvan. Design and evaluation of an efficient reliable manycast protocol for ad-hoc networks. Technical Report CS-TR:838, University of Newcastle upon Tyne, 2004.

[14] Q. Wang, W.-P. Chen, R. Zheng, K. Lee, and L. Sha. Acoustic target tracking using tiny wireless sensor devices. In *Proc. of the 2nd International Workshop on Information Processing in Sensor Networks (IPSN'03)*, 2003.