Energy-Efficient Dissemination in Sensor Networks: Reactive Event Flow Shaping

Damla Turgut, Onur Ozyer, Kien Hua, and Ladislau Bölöni
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 32816-2450
{turgut,oozyer,hua,lboloni}@cs.ucf.edu

***Abstract*— Sensor networks provide sensing capability in a wide range of application domains such as healthcare, home, military, commercial and so on. Typically, energy is much more limited in sensor networks than in other wireless networks. This is due to the intrinsic nature of sensing devices and the difficulty in recharging their batteries. The nodes in the network usually dissipate energy by transmitting redundant packets. Additionally, in multi-event scenarios, the nodes fill up their caches with various events, instead of a specific set of the events, decreasing the in-network data aggregation efficiency. In this paper, we propose Reactive Event Flow Shaping (REFS), an energy-efficient data dissemination protocol for wireless sensor networks. REFS selectively prunes links and narrows the event flows according to the heuristics described. From our simulation, we have observed that 40% savings is achieved in terms of the number of transmissions and cache performance compared to flooding.**

I. INTRODUCTION

Techniques for miniaturization allowed the creation of networks of lightweight, low-power, inexpensive nodes capable of sensing a variety of phenomena. These nodes are able to process the sensed data and to transmit it to the external nodes through wireless links. Sensor networks have a variety of applications ranging from sensing toxic chemicals to monitoring patients in a hospital. The peculiarities of sensor networks lead to new challenges in the wireless protocols deployed [2], [6]. Sensor network are deployed in a dense network, leading to more frequent packet collisions [6]. A significant number of collisions is not only detrimental for the bandwidth, but also increases the energy consumption of nodes, because of the multiple retransmissions. As sensor nodes are relying on batery power and are typically not serviceable in the field, this leads to an overall decrease in the lifetime of the network. This situation is even worse when a sensor network is monitoring multiple events for an extended period of time. The network performance is diminished due to the high number of packet collisions and low aggregation.

Let us first consider data transmission using a plain flooding algorithm. Although flooding in general is a very bad algorithm for delivering point to point traffic, it is widely used in sensor networks. One of the reasons is that the redundancy increases the robustness of the forwarding. Another consideration is that for data sensed about an event the wide distribution of the packets allows for significant data fusion and compression to take place. This is facilitated by the fact that the observations concerning a single event from multiple sensors are strongly correlated. This correlation, however, does not exist for multiple independent events.

On the other hand, even with data fusion taken into consideration, the flooding algorithm leads to much more retransmissions than justified by robustness considerations. This is especially problematic when traffic flows representing multiple independent events overlap: in these situations there is no possibility to reduce the number of packets through data fusion.

In the paper we introduce Reactive Event Flow Shaping (REFS), a protocol designed to improve the energy efficiency of data dissemination in sensor networks. REFS starts from the plain forwarding algorithm and selectively shapes the *event data flows* by notifying certain nodes to stop forwarding packets for particular events. Protocol packets carrying *stop forwarding* messages are initiated by the sink and propagate all the way back to the source. The narrowing of event flows means less interference, i.e., fewer collisions, with the other event flows. In addition to this, by reducing the number of independent event data flows handled by a given node, it increases the correlation between the packets forwarded by the node. This leads to a significant improvement in the performance of data aggregation and compression algorithms.

The rest of the paper is organized as follows. Section 2 discusses the previous work. The details of REFS protocol is given in section 3. The simulation study is presented in section 4 and the paper is concluded in section 5.

II. RELATED WORK

Data management in wireless sensor networks can be achieved one of the two ways: one is local storage meaning that store event locally and query to all the nodes and the other is the data-centric storage in which case, the events are still stored locally, but the data is queried by the name. Distributed Index for Features in Sensor Networks (DIFS) [4] is designed along the lines of the latter approach to provide the search efficiency of a quad tree in a manner that balances the communication load across the index.

Topology-Divided Dynamic Event Scheduling (TD-DES) [3] is another dissemination protocol that organizes the network into a mult-hop tree. The root of the tree creates a data dissemination schedule and propagates it through the tree. The usage of data/event model is effective because it allows for flexibility in the ordering of events by the event scheduling algorithm where different event scheduling criterias can be used such as network specific parameters as well as application specific semantic types in which each node can choose to subscribe.

The Scalable Energy-Efficient Asynchronous Disseminaton (SEAD) [7] protocol achieves savings in the power consumption expended on communication to mobile sinks at the expense of an increase in the path delay. It constructs and updates the d-tree dynamically and composed of two main functions, namely, adding/removing a node to/from the tree. The sources send the data to multiple mobile sinks.

In directed diffusion [5], the authors proactively construct a network where the sink initiates the event queries or the interests. While the interests are propagated over the nodes, they set gradients, footprints to the immediate neighbor nodes from which interests come from. Hence, data finds its way to the sink following the gradients back to the sink. Although the scheme supports multiple-sink and multiple-event scenarios, the data dissemination needs to be triggered by the interests. REFS, in contrast, was designed to reactively construct event data paths. No queries from the sink are needed. In addition, the interests in directed diffusion are flooded in the network while REFS does not flood any control packets. Instead, control packets are sent to the neighboring nodes. Directed diffusion uses negative reinforcement to restrict event flows which is similar to REFS mechanism to stop unnecessary flows.

Stream Enabled Routing (SER) [1] focuses on quality of service. QoS issues include delivering data in the shortest time or selecting routes that will conserve the most energy. Routes are set up dynamically when sensed information is requested from the sources. S-messages are broadcast from the sink. When sources hear S-messages, they establish streams based on QoS requirements specified in the S-message. The source can establish a basic level-1 stream or a level-2 stream to provide a better QoS. While REFS does not have the extensive QoS choices provided by SER, it nonetheless provides basic QoS guarantees based on event delivery speed. Again, this is because slower flows are more likely to be stopped.

In both directed diffusion and SER, data dissemination is proactive. A recent approach, Event-to-Sink Reliable Transport (ESRT) [11], has also reactive mechanism, however their motivation is different from our approach. In this method, sink maintains the data rate of the events. Based on the data rate values, the sink sends feedback to the sources to regulate the reporting rate of the sensors to avoid congestion and power-down sensors.

Our data dissemination protocol, REFS, is a fully reactive protocol, built on the robustness of flooding, where event flows are narrowed in order to reduce collisions between different flows; therefore resulting of the increased network lifetime. REFS also boasts simplicity and low control overhead.

III. REACTIVE EVENT FLOW SHAPING (REFS)

A. Preliminaries

Our protocol assumes a scenario in which a series of sensor nodes are deployed in a geographic area.

The nodes are using their sensors to detect the occurence of any of a set of pre-defined *events*.

The nodes are continuosly monitoring the environment, but they are transmitting their observations only if one of the predefined set of circumstances arise. These *events* are defined as a boolean expression over the sensor readings, and they are either pre-programmed in the sensor nodes firmware, or are set by queries previously originated in a sink. An example of such a scenario is a forest monitoring network, where the event can be, for instance a forest fire. Once an event is detected, the sensor starts sending data about the event to the *sink*. Thus, the transmission is initiated by the nodes (potentially, as a result of an earlier query); the role of the sink is purely reactive.

We will an *event data flow* the collection of packets which represent data about a given event. All packets of the flow originate at a set of sensors in the geographical vicinity of the event and terminate at the sink. The transmission of the data happens through hop-by-hop communication, every sensor node acting as a retransmission and, potentially, data fusion and compression engine. One advantage of this method is that the nodes need to use a limited transmission range, which conserves energy. As the flows are narrowing towards the sink, the buffers of the intermediate nodes will contain more and more packets from the same flow [8], [9]. In REFS the sink can stop the forwarding of certain flows at selected intermediate nodes. This way the buffers of the forwarding nodes can store more packets of the remaining flow, increasing the ability of a node to perform data compression. Finally, by selectively pruning the flows with lower data rates reactive event flow shaping improves the quality of service of the sensor network.

B. Proposed Technique

Our main design goal is to prune the event flows based on the observations on the data rates of the event flows coming from different nodes. In the current version of REFS, the sink is the only node which can initiate pruning. A lighter version of pruning criteria may be incorporated in the immediate nodes in the future work.

The operation of the REFS protocol is based on two tables maintained by the sensor nodes:

- The *event-data rate table* is stored at all the nodes including the sink. It stores the state information about the events and the data rates. At the sink, it is used to compute which flow to be kept whereas at an intermediate node, it is used to check whether it receives a particular event from a node that sent an event advertisement. The attributes of the Event-data rate table are given in I.
- The *stop transmission table* is stored at all intermediate nodes. Together with event-data rate table it is used to determine whether it should forward or drop the data. It

TABLE I

THE STRUCTURE OF THE EVENT-DATA RATE TABLE

D_{jk}	Data rate for event k from node j
D_{avg}	Avg. incoming data rate for event flows
D_{sumj}	Sum of data rate from node j
E_j	No. of events received from node j
E_{avg}	Avg. no. of incoming events from upstream nodes

contains the list of events for which the node should drop the packets.

The novelty of the REFS lies on how the sink chooses which flow to keep. If D_{jk} is greater than D_{avg} then the flow is kept; however, if it is less, it may be because of the upstream node (node j) condition. Hence, if the rate is actually considered unacceptable compared to the average rate from the upstream node (D_{sumj}), it is dropped. Else, we check whether the number of events from the upstream node (E_j) is greater than average number of events from any upstream node (E_{avg}). If so, it is dropped since the aim is to ease the congestion at the upstream node j. Below is the formal representation of the selection algorithm.

```
Bool KeepThisStream = true
If($D_{jk} \leq D_{avg}$) Then
  If($D_{jk} \leq \frac{(D_{sumj}}{E_j}$))or($E_{avg} \leq E_j$) Then
    KeepThisStream = false
```

In REFS, unless the data rate is better than average, it must be at least equal to the average data rate from the particular upstream node and the number of streams from the node must be at most the average number of streams from any node.

There are three protocol packet types: *stop, event advertise*, and *continue*. In Figure 1, the data is flooded from each node to its downstream neighbor one hop away. The arrows show the direction of the data flow. Each node has an event-data rate table to collect the data rate from each upstream node and event pair. Once the sink decides to drop a connection, for instance event 1 from node 2, it sends a *stop* message to its upstream nodes. Upon the specified node receiving the stop request, it starts a timer for this event. It has to ensure that no other node is expecting this event before dropping the flow within a timeout value. It then sends the *event advertise* to all immediate downstream nodes. The downstream nodes of this node, in this case node 1, decides whether to respond to the message or let the sending node really stop sending the stream. If none of the downstream nodes send *continue* message, the timeout for this event expires. Consequently, the event is stopped resulting that any incoming packet regarding this event is to be dropped regardless of the upstream node or the source. The *continue* message requires the source attribute since only downstream node knows its upstream neighbor, the source attribute allows the upstream node to determine if the continue packet is targeted for itself.

REFS has low control overhead, as it does not flood the network with control messages. In addition, since REFS is reactive, there are no setup packets required. Specific nodes that are part of event flows are told to cease forwarding when necessary. In addition to low control overhead, REFS attempts to balance the load of forwarding events among the nodes. This is accomplished as part of the sink's heuristics. Distribution of load means that energy consumption is even among the nodes resulting the prolonged overall network lifetime.

IV. SIMULATION STUDY

A. Simulation setup

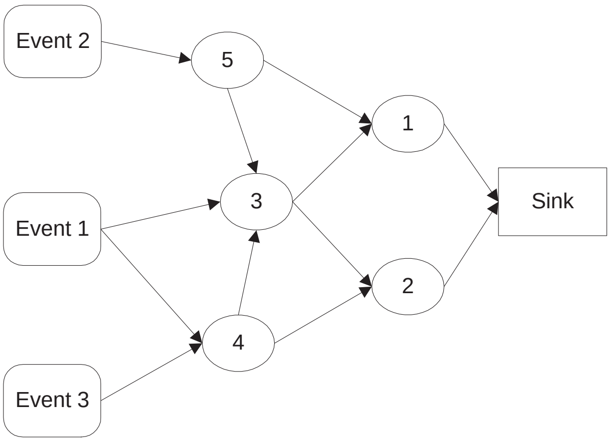
In order to measure the performance of the proposed method we performed several simulations of the sensor network. In this section, we describe the simulation environment and setup, the performance metrics used and give a detailed discussion of the results. We used the Global Mobile Information Systems Simulation Library (GloMoSim) [12] as the simulation tool. GloMoSim is a wireless network simulator designed on top of the discrete-event simulation language Parsec. The simulations were performed on 1.4 GHz Pentium-4 computers with 1GB RAM.

The sensor nodes were deployed in a grid topology, keeping the distance between the nodes at a constant value of 20 meters for each of the configurations. We used three basic simulation configurations with the number of nodes in the network as 16, 25 and 49. We inserted a single sink node in the upper left corner of the grid. The source nodes were selected from the right bottom triangle of the grid randomly. In each simulation setup, we assigned 10 nodes to be the source nodes. The radio bandwidth and the transmission range for the nodes was set to 1.6 Mbit/sec and 40 meters [6] respectively. In order to ensure the connectivity in the simulation, we adjusted the distance between the nodes to 20 meters for all configurations. The source nodes reported the events with a stream of packets with a constant bit-rate of one packet per second. Each event packet has a payload of 64 bytes, which is a common application-layer packet size for sensor nodes. In each configuration we generated 5 independent events which span the entire simulated interval of 240s. The simulation parameters are summarized in Table II.

B. Simulation Metrics

For each simulation ran we have measured the following values:

- **Number of messages transmitted:** Total of number of data packets transmitted successfully in the simulation. This metric includes the redundant transmissions of the same data packet.
- **Same-event overlap:** The number of packets of each event in the node's buffer is calculated each time a data packet is inserted into buffer. The maximum of these numbers is returned and average is determined



Event-Data Rate Table of Sink		
Node	Event	Data Rate
1	1	5
	2	3
	3	6
2	1	2
	2	3
	3	4

Fig. 1. Example Scenario

TABLE II
SIMULATION PARAMETERS

Total number of nodes	16, 25, 49
Number of sink(s)	1
Source nodes	10
Radio bandwidth	1/6 Mbitssec.
Transmission range	40 meters
Event packet payload	64 Bytes
Events generated	5
Simulation duration	240s

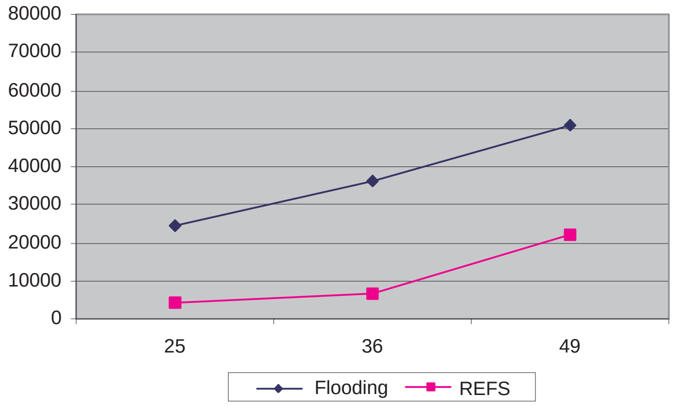


Fig. 2. The number of transmitted packets in function of the number of nodes for plain flooding and the REFS algorithm

over the simulation duration time. Intuitively, the same-event overlap is a measure of the compressibility of the information in the individual nodes.

- **Overhead:** This is the ratio of the total number of protocol control messages (stop-transmission, event-advertisement and continue-transmission) to the total number of data packets transmitted in the network. This metric gives the cost of the proposed method.

C. Simulation Results

We conducted the simulation in four different configuration files varied with the network size. In order to observe how much of the data transfer is reduced we measured the total number of data transmissions in case of plain flooding and REFS. Figure 2 shows the number of data transmissions for each configuration. REFS is selecting the upstream node for data stream, keeping the higher rate streams. A subset of upstream nodes is chosen using the heuristics described. Since the number of upstream nodes is reduced, this will lead to a decrease in the total number of data messages in the network. We observed about 40% savings in the number of transmissions in the network compared to the plain flooding. In critical systems, some may argue that the redundant packets will be dropped reducing the confidence of the data. However, if the numbers of transmissions are reduced in a controlled manner as in REFS, the data packets will experience less collision. In REFS, an event flowing from an upstream node is dropped only if there is another upstream node sending the same flow with a higher rate. In addition, REFS allows almost identical packets reach to the sink since it prunes only the slower redundant paths. As a result, our approach reduces the collisions in the network and gives the sensed information a better chance to arrive at the sink.

The more packets from the same flow are in the buffer of a node, the higher the efficiency of the data compression. In order to measure this property, we devised a metric, same-event overlap. The same-event overlap of the nodes during the simulations are plotted in Figure 3. The average same-event overlap of nodes in REFS became 25% to 40% higher than in the case of the plain flooding algorithm. Although our simulations do not implement data aggregation due to the large number of possible algorithms which can be chosen, the same-event overlap is a good indicator of what the performance of the aggregation algorithm can be.

Let us now consider the overhead introduced by the reactive

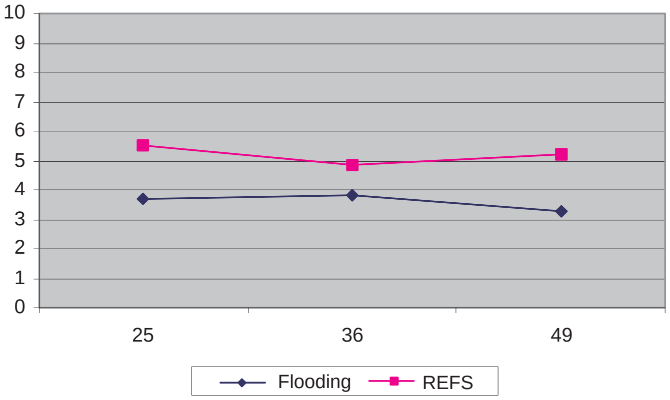


Fig. 3. The average same-event buffer overlap in function of the number of nodes for plain flooding and the REFS algorithm

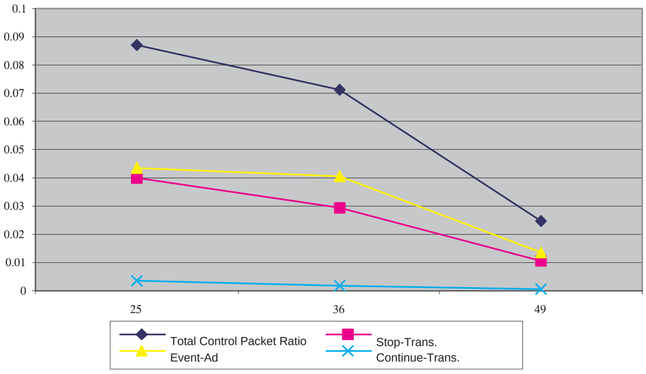


Fig. 4. The overhead of the REFS protocol, expressed as a number of various types of control packets as a percentage of total packets in the network, in function of the number of nodes.

event flow shaping algorithm. First, REFS requires the nodes, to measure and store some attributes of the incoming data in the event-data rate table and the stop transmission table. A timeout mechanism guarantees the expiration of the entries from these tables after the termination of the flow. The size of the table is bound by the number of currently active events. In addition to the cost of maintaining the tables, the protocol introduces additional traffic by the protocol packets which are transmitted by the nodes. In Figure 4 we compare the ratio of the three types of protocol packets (event advertisement, stop flow and continue flow) to the number of data packets in the simulation. We find that the ratio is 9% for a network of 25 nodes, and decreases as the number of nodes in the network increases. We conclude that reactive event flow shaping adds a small overhead to the operation of the sensor network, more than offset by the reduction of the overall number of packets.

V. CONCLUSIONS

In this paper, we described the Reactive Event Flow Shaping technique for data dissemination in sensor networks. The algorithm selectively prunes certain intermediary nodes from the forwarding, decreasing the probability of collisions and increasing the ability of the nodes to perform data fusion and compression. Future work involves the incorporation of more advance heuristics for the pruning of the flows, a stricter load balancing algorithm and the extension of the algorithm to sensor networks with multiple sinks.

REFERENCES

[1] Akyildiz, I. F., and Su, W. "A Stream Enabled Routing (SER) Protocol for Sensor Networks", *Med-hoc-Net 2002*, September 2002.

[2] Akyildiz, I. F., Su, W. Sankarasubramaniam, Y., and E. Cayirci. "A Survey on Sensor Networks", *IEEE Communications Magazine*, Vol. 40, No. 8, pp. 102-114, August 2002.

[3] Cetintemel, U., Flinders, A., and Sun, Y. "Power-Aware Data Dissemination in Wireless Sensor Networks", In *Proceedings of the 3rd ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDEŠ03)*, September 2003.

[4] Greenstein, B., Estrin, D., Govindan, R., Ratnasamy, S., and Shenker, S. "DIFS: A Distributed Index for Features in Sensor Networks", In *Proceedings of First IEEE International Workshop on Sensor Network Protocols and Applications*, May 2003.

[5] Intanagonwiwat, C., Govindan, R., and Estrin D. "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", In *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks*, August 2000.

[6] Kahn, J.M., Katz R.H., and Pister, K.S. "Next Century Challenges: Mobile Networking for Smart Dust", In *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks*, August 1999

[7] Kim, H.S., Abdelzaher, T., and Kwon, W.H. "Minimum-Energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks", In *Proceedings of ACM SenSys*, Los Angeles, CA, November 2003.

[8] Krishanamachari B., Estrin D., and S. Wicker. "The Impact of Data Aggregation in Wireless Sensor Networks", In *International Workshop of Distributed Event Based Systems (DEBS)*, Vienna, Austria, July 2002.

[9] Madden, S., Franklin, M., Hellerstein, and J., Hong, W. "Tag: A tiny aggregation service for ad-hoc sensor networks", *OSDI 2002*, Boston MA, 2002.

[10] Pottie, G., and Kaiser, W. "Integrated Wireless Network Sensors", *Communications of the ACM*, May 2000, 43 (5) pp.51-58.

[11] Sankarasubramaniam, Y., Akan O.B., and Akyildiz, I. F. "ESRT: Event-to-Sink Reliable Transport in Wireless Sensor Networks", *Fourth International Symposium on Mobile Ad Hoc Networking and Computing*, June 2003.

[12] Zeng, X., Bagrodia, R., and Gerla, M. "Glo-MoSim: A library for parallelsimulation of large-scale wireless networks", In *Proceedings of the 12th Workshop on Parallel and Distributed Simulations*, June 2003.