

# On-Demand Data Dissemination in Large Wireless Sensor Networks

Ramdharna Reddy Kalva, Jong-Hoon Youn and Chulho Won

Dept. of Computer Science  
University of Nebraska at Omaha  
Omaha, NE 68182  
{rreddy, jyoun, cwon}@mail.unomaha.edu

**Abstract:** *In this paper, we present a new data dissemination protocol that conserves battery power by reducing the amount of communication required to disseminate sensing data from sources to mobile sinks. The proposed scheme, called on-demand data dissemination (ODDD), focuses on the elimination of resource waste in large-scale dense sensor networks caused by virtual grid construction and maintenance. The virtual grid-based approaches in the literature build grids over the entire network including the areas where a data collector never roams into. However, in the proposed scheme, a source constructs a dissemination line along the X-axis only.*

*To evaluate the effectiveness of the proposed scheme, we have conducted a comprehensive simulation study using the ns2 simulator. Our simulation results show that ODDD outperforms other existing schemes in terms of packet delivery ratio and delay. ODDD also achieves significant energy savings by reducing the overhead caused by grid constructions and frequent local flooding.*

**Keywords-** *wireless sensor networks; data dissemination*

## I. INTRODUCTION

The reduction in the development costs of sensors and wireless communication devices has made rapid ad-hoc deployment of wireless sensor networks for a wide range of applications.

A wireless sensor network (WSN) is a self-organized and self-configured network consisting of hundreds or thousands of battery-powered sensors with very limited processing and communications capabilities. Resource scarcity is the major design challenge in wireless sensor networks. For example, the lifetime of batteries dictates the lifetime of the entire network since sensors are battery operated devices and once the battery is depleted mostly it is not possible to renew the battery. On research and development of sensor networks, design features like low power, low cost, small size and fault tolerance pose numerous challenges. Especially, low energy consumption is one of the major bottlenecks in the design of sensor networks. Therefore, in this paper, we present a new data dissemination protocol for large sensor networks with mobile data collectors. The goal of the proposed scheme is to minimize the energy consumption of sensors, and hence prolongs the lifetime of the entire network.

## II. RELATED WORKS

In large-scale sensor networks, scalable low-power data dissemination schemes are very critical for achieving high levels of efficiency [1, 2]. This is, because the exchange of data between nodes strongly dominates other functions such as sensing and computing [2]. Consequently, a number of researchers have focused on developing energy efficient

communication protocols for wireless sensor networks [3, 4, 6, 7, 8, 9, 12, 13, 14]. However, many protocols in the literature do not scale well to large sensor networks due to their global flooding strategy. Furthermore, if a network has mobile data collectors, each of data collectors need to periodically report its current position or path in order to keep receiving sensed data from a source. These frequent location updates consume limited communication resources such as bandwidth and battery power.

In order to prevent the explosion of messages due to frequent message flooding in large scale sensor networks, Two-Tier Data Dissemination (TTDD) was proposed in [13]. TTDD constructs and maintains a uniform grid structure by dividing the field into small grid cells, and confines flooding within local grid cells only. Using the ability to localize query flooding, TTDD demonstrates the feasibility of building a virtual grid structure to support efficient data dissemination in large-scale wireless sensor networks. However, TTDD has several problems that should be addressed. First, the grid construction and maintenance required per source leads to considerable overhead. Secondly, uniform grid construction leads to inefficient use of bandwidth and energy, because grid points have to be maintained even in the regions where no sink is present. Thirdly, local flooding still wastes a significant amount of energy and the amount of energy consumption increases with the density of network and the speed of mobile sinks.

To overcome these problems in TTDD, another grid-based data dissemination protocol, called Efficient Data Dissemination and Aggregation (EDDA), was introduced in [14]. In EDDA, sources with same data type share a single grid structure to disseminate their sensing data. This grid sharing also increases the chance of data aggregation. Furthermore, the local flooding is substituted by unicasting messages. In TTDD, in order to obtain the immediate dissemination node, the sink floods in the  $1.3\alpha$  range. In EDDA, a sink obtains the position of a nearby immediate dissemination node by using a hash function [10]. The hash function gives one output,  $(x, y)$ , where location  $(x, y)$  lies within 0, cell size  $(\alpha)$ . Then, it sends a unicasting query to the point. However, EDDA still constructs and maintains the grid uniformly all over the sensing field. In large scale wireless sensor networks, the grid construction and maintenance in the regions that a sink never enter wastes considerably network resources.

## III. ON-DEMAND DATA DISSEMINATION

Our scheme is designed with the following set of assumptions:

- Sensors are randomly and densely scattered over the field.
- Sensors are stationary after deployment but data gathering nodes, called *sinks*, may be mobile. When a sensor detects an event of interest, it becomes a source node and collects data from the sensing field. Mobile sinks aggregate the received data, and decide whether there are any unusual occurrences in the area of interest.
- $\leftarrow$  All sensors are aware of their location and use greedy forwarding [5] to deliver packets to a destination, which forwards packets to a neighboring node that is always progressively closer to the destination. Whenever greedy forwarding fails, a sensor uses local flooding to bypass topology holes.

Sensor networks with mobile sinks are useful in many applications. For example, in a military application, seismic sensors could be scattered across a battlefield in order to detect and monitor the deployment of enemy tanks. A group of soldiers may carry a wireless communication device to collect information regarding the location of enemy tanks.

The proposed protocol, called *on-demand data dissemination* (ODDD), is similar to EDDA. However, it differs from EDDA in grid construction. In the proposed scheme, a source does not proactively construct a virtual grid. Instead, a source sends a data announce message along the X-axis only. Therefore, ODDD reduces the amount of communication overhead for creating and maintaining virtual grid structures over the entire network including the areas where a data collector never roams into. The details of the proposed scheme are given below.

**Data Announcement:** Assume a source  $S$  detects an event and generates sensing data of type  $d$ . In order to disseminate the data announcements, the source first figures out the location of a grid point on a cell using a hashing function  $H(d, t)$  where  $d$  is a data type and  $t$  is a current unit time. The output of  $H(d, t)$  is the position of a grid point on a cell  $(x_i, y_i)$ , where  $0 \leq x_i, y_i \leq \alpha$  ( $\alpha$  is a cell size). In order to randomly distribute communication load, the output of  $H(d, t)$  needs to be changed from time to time. So, the hashing function takes the current unit time as a parameter. The unit time increases after a predefined period; in our simulation study, we set this period to 50 seconds.

Once a source figures out the position of the nearby grid point, it sends a data announcement message towards the grid point using greedy geographical forwarding. The data announcement message contains the cell ID and data lifetime; the cell ID indicates where the source currently resides, and data lifetime is the expected period that the detected event lasts. This message moves toward a sensor node that is closest to the grid point; let us call this node  $G_i$ . After receiving the announcement message, the receiver ( $G_i$ ) communicates with its neighbors in order to check whether there is any established dissemination lines of data type  $d$ . If there exists another grid point serving for data type  $d$  (constructed by another source with same data type), the receiver simply forwards the received announcement to the existing grid point, and the existing grid structure will be used to propagate the announcement message from the new source  $S$ . Otherwise, the receiver becomes a *grid node* of data type

$d$ . ODDD enables sharing of the existing grid nodes to disseminate data of same type, and gives an opportunity for data aggregation.

If there is no existing grid nodes,  $G_i$  calculates the locations of its two neighboring grid points,  $(x_i + \alpha, y_i)$ ,  $(x_i - \alpha, y_i)$ , and then propagates the received data announcement message to each grid point using greedy forwarding. The node closest to the grid point receives this message and becomes a grid node. The new grid node forwards the received data announcement message to its neighboring grid points again, and so the data announcement message is recursively forwarded until it reaches the network boundary. When a grid node receives a data announcement, it keeps the received cell ID in its announcement list until the advertised lifetime expires.

**Query forwarding:** When a sink wants to receive data of type  $d$ , it applies the same hash function  $H(d, t)$ , which gives the position of a nearby grid point, *i.e.*  $(x, y)$ . We call this coordinate as an *immediate dissemination point* to the sink. The sink then sends a query towards the immediate dissemination point as shown in Fig. 1. A query message may contain a list of cell IDs that a sink wants to monitor. When the query reaches a node nearest to the immediate dissemination point, it sends an acknowledgment to the sink and forwards the query along the Y axis. Let us assume the location of the nearby node is  $(x_j, y_j)$ . Then, it propagates the received query to two neighboring points  $(x_j, y_j + \alpha)$  and  $(x_j, y_j - \alpha)$ . This message is recursively forwarded along the Y axis.

In ODDD, all sensors and sinks should use the same hash function  $H(d, t)$ . As long as the two parameters of the hash function,  $d$  and  $t$ , are same, the output of the function is same. Therefore, for any given data type, all sources and sinks would work with the same coordinate  $(x, y)$ . Accordingly, this scheme guarantees that there will be a dissemination node that receives both query and data announcement messages.

**Data forwarding:** If a grid node receives a query, it checks with its announcement list to see whether the list includes any data announcements from the cell IDs specified in the query. If so, the grid node forwards the query to the corresponding source along the reverse path of the data announcement message. Once the source receives the query, it starts sending data back to the sink along the reverse path of the query as shown in Figure 1. Therefore, data move toward a sink along the  $x$ -axis and then the  $y$ -axis. An immediate dissemination point sends the data to a sink using Trajectory forwarding [13].

An example of the proposed dissemination scheme is shown in Figure 1. When a source detects an event of interest, it initiates a data announcement message and it is propagated horizontally. Later, two sinks interested in receiving the data initiate a query that travels along the Y axis. Once this query reaches a dissemination node that receives a data announcement, the dissemination node forwards the query to the neighboring dissemination node from which this dissemination node receives data announcements. Upon receiving the query, the source sends data to the sinks along the reverse path of the query. Therefore, a sink can collect all desired sensing data of type  $d$  currently available in the network by initiating a single query message of type  $d$ .

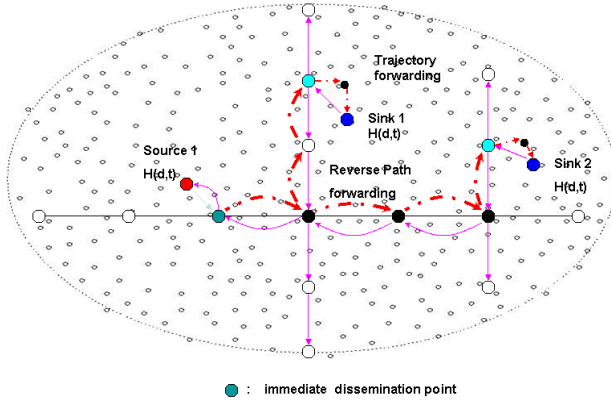


Figure 1: Data forwarding in ODDD: Data announcement (on X-axis) and Query forwarding (on Y-axis)

#### IV. SIMULATION STUDY

To compare the performance of ODDD with both EDDA and TTDD, a simulation study has been conducted using various network configurations and traffic patterns. We have implemented EDDA and ODDD in ns-2.1b8a by extending the original simulation code developed for TTDD[11, 13].

##### A. Simulation Model

The default simulation scenario has 200 sensors randomly distributed in a  $1200\text{m} \times 1200\text{m}$  field. Among 200 nodes, 4 sources are randomly selected and 4 sinks (also, called as *data collectors*) are randomly deployed in the  $1200\text{m} \times 1200\text{m}$ . A source generates a report every second. A sink moves according to the random waypoint model with a maximum velocity of 10 m/sec. Upon arriving at the chosen destination, sinks pause for 20 seconds. We assume the size of grid construction packet, query packet and data packet are 8, 36 and 64 respectively. The default cell size ( $\alpha$ ) is set to 400 meters.

We use a greedy geographical forwarding to deliver packets. The 802.11 DCF protocol is used as a MAC protocol. To measure the energy consumption, we define three different energy modes: receiving, transmitting and sleep. The energy consumptions of transmitting, receiving and idle modes are 0.66W, 0.395W, and 0.035W, respectively. Each simulation run lasts for 200 seconds, and each result is averaged over five random network topologies.

##### B. Performance metrics

We choose three metrics to compare the performance of our proposed approach with TTDD and EDDA. They are: success ratio, packet delay and total energy consumption. The *success ratio* is defined as the ratio of the number of successfully received packets over the total number of transmitted packets, averaged over all source-sink pairs. The *packet delay* is the average time interval between the packet being ready and the packet arriving at the destination, averaged over all source-sink pairs. The *energy consumption* is defined as the sum of each node's energy consumption for transmitting and receiving only. We do not take the energy consumption in idle mode into account, since it largely depends on the data generation interval.

##### C. Simulation Results

We adjust the following parameters in the simulation: speed of mobile sinks, cell size ( $\alpha$ ) and node failure ratio.

**Impact of sink mobility:** We study the impact of mobile sinks on the on-demand scheme in comparison with the EDDA and TTDD schemes. In the default simulation setup, we varied the maximum speed of sinks from 0, 5, 10, to 20 m/s. All other simulation parameters are as specified in the default simulation scenario.

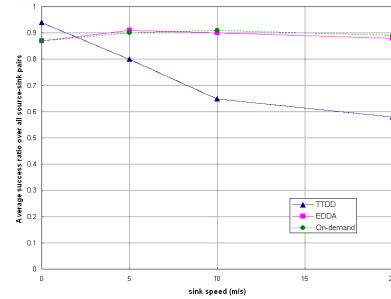


Figure 2: Success ratio for sink mobility

Figure 2 gives the success ratio of the ODDD scheme with mobile sinks, in comparison with the EDDA, TTDD schemes. Both on-demand and EDDA have a higher success ratio than TTDD because both eliminate expensive and frequent local flooding by unicasting messages. On the other hand, on-demand has a slightly higher success ratio than EDDA since it has slightly less communication overhead than others due to non-uniform grid construction.

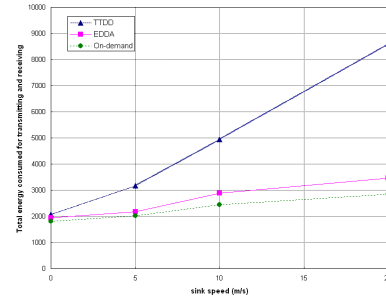


Figure 3: Total energy for sink mobility

Figure 3 plots the total energy consumption as the sinks' moving speed changes. As a sink moves with high speed, it needs a new immediate dissemination node more frequently. In TTDD, a sink floods a local query within the  $1.3\alpha$  region each time to find a new nearby dissemination node, whereas in EDDA and ODDD schemes, a sink uses unicasting. Therefore, the impact of sink mobility is severe on TTDD due to frequent local flooding queries. However, in both EDDA and ODDD, the slope of the curve decreases because the mobile sink does not significantly affect the high-tier data forwarding.

Figure 4 depicts the packet delay in data delivery of ODDD in comparison with EDDA and TTDD. We can make two observations from Figure 4. First, in the static sinks scenario, TTDD has slightly lower delay than other two schemes since TTDD takes comparatively shorter paths than EDDA and ODDD (both use X-Y routing). Second, as the

sink speed increases, all protocols generate more query packets which may cause congestion in the network. So, the delay increases in all three cases, but the impact is greatest in the case of TTDD.

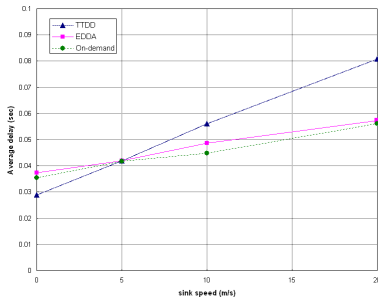


Figure 4: Delay for sink mobility

**Cell size ( $\alpha$ ):** We further study the impact of cell size on the on-demand scheme in comparison with other schemes. In the simulations, the cell size ( $\alpha$ ) varies from 200m to 600m. All other simulation parameters are same as specified in the default simulation scenario.

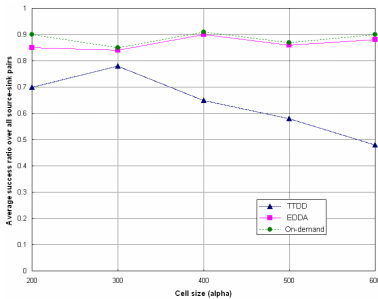


Figure 5: Success ratio for different cell sizes

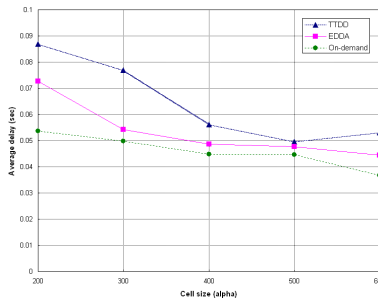


Figure 6: Delay for different cell sizes

Figure 5 and Figure 6 plot the success ratio and the average delay of all three approaches with respect to the variations of the cell size ( $\alpha$ ), respectively. In all cases the success ratio of ODDD and EDDA is greater than 85 %. The success ratio of TTDD increases first as the cell size increases. This is because the larger cell size results in less grid construction overhead. However, it decreases again from the cell size of 400m $\times$ 400m since the larger cell size increases the scope of local flooding, and causes congestion, collisions and packet drops. In terms of delay, ODDD outperforms EDDA and TTDD in all scenarios.

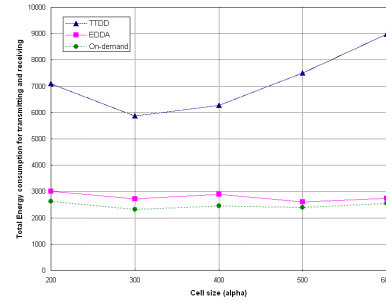


Figure 7: Energy for different cell sizes

Figure 7 plots the energy consumption with respect to variations in cell size ( $\alpha$ ). We can make two observations. First, in all cases, the energy consumption of ODDD is less than EDDA and TTDD because ODDD constructs grid only on X-axis and eliminates local flooding with unicasting. Second, in all simulation cases TTDD has higher energy consumption than others. In TTDD, the energy consumption first decreases and then increases because in the case of lower cell sizes, the overhead caused by grid construction takes lot of energy. In the case of higher cell sizes, local flooding consumes a significant amount of energy.

From the simulation results, we can conclude that cell size has little impact on the average energy consumption per packet in the case of ODDD. In all scenarios, ODDD has slightly less average energy consumption than EDDA due to non-uniform grid construction.

**Resilience to node failures:** We also study the impact of node failures on the proposed protocol. In order to study the impact of node failures, we allow 0 to 10 % of randomly chosen nodes out of 200 nodes to experience sudden failures at  $t = 20$  sec. The nodes failed at  $t = 20$  sec may include some grid nodes.

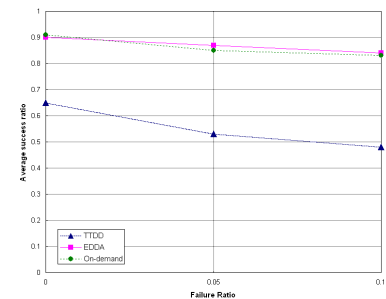


Figure 8: Success rate for sensor node failures

Figure 8 plots the success ratio of all three schemes. The success ratio of ODDD and EDDA decreases slightly as the node failure ratio increases. This is because the faulty dissemination node stops downstream data delivery, and so the success ratio drops. The success ratio of TTDD decreases significantly because of flood messages within the  $1.3\alpha$  to obtain a new immediate dissemination node.

Figure 9 plots the energy consumption with respect to the variation of the failure ratios. In the ODDD and EDDA schemes, the energy consumption decreases slightly as the failure ratio increases due to the data packet drops in the network. However, in TTDD, energy consumption increases because of frequent local flooding. In TTDD, when a grid

node stops the downstream data flow towards the sink, sink floods within the  $1.3\alpha$  range after waiting for a threshold amount of time for the data, which increases the energy consumption.

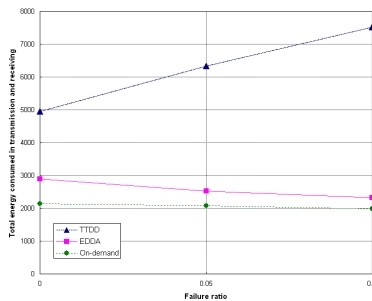


Figure 9: Energy for sensor node failures

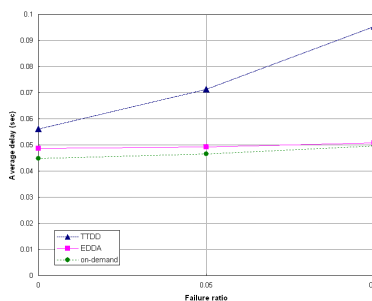


Figure 10: Delay for sensor node failures

Figure 10 gives the average delay of all three dissemination mechanisms. The delay increases mildly in the proposed scheme and EDDA since bypassing failure nodes introduces additional delay. However, the delay increases significantly in the case of TTDD. In TTDD, when there is no data report from an immediate dissemination node, a sink floods in the cell to obtain a new immediate dissemination node, which may cause congestion, collisions and considerable packet delay.

## V. CONCLUSIONS

In this paper, we proposed a new data dissemination scheme to reduce battery waste in the creation and maintenance of virtual grid structures. Therefore, in the proposed scheme, a source builds only a single dissemination line horizontally, and a query message initiated by a sink is propagated vertically.

To evaluate the effectiveness of the proposed scheme, we have conducted a comprehensive simulation study using the ns2 simulator. Our simulation results show that ODDD outperforms other existing dissemination schemes in terms of packet delivery ratio and delay. ODDD also achieves significant energy savings. This is because ODDD almost eliminates frequent and expensive local flooding queries using a hashing function. ODDD also consumes less energy than EDDA since data announcement messages propagate along X-axis only.

## REFERENCES

- [1] D. Culler, D. Estrin and M. Srivastava, "Overview of sensor networks," *IEEE Computer Magazine*, vol. 37, no. 8, pp. 41-49, August 2004
- [2] D. Estrin and R. Govindan, "Next century challenges: scalable coordination in sensor networks," *ACM/IEEE International Conference on Mobile Computing and Networking*, August 1999.
- [3] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Micro sensor Networks," *Proceedings of the Hawaii Conference on System Sciences*, January 2000.
- [4] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, August 2000.
- [5] B. Karp and H. T. Kung, "Greedy perimeter stateless routing (GPSR) for wireless networks," *ACM/IEEE International Conference on Mobile Computing and Networking*, August 2000.
- [6] H. S. Kim, T. Abdelzaher, and W. H. Kwon, "Minimum-Energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks," *ACM SenSys*, Los Angeles, CA, November, 2003.
- [7] J. Kulik, W. Heinzelman, and H. Balakrishnan, "Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks," *Wireless Networks Vol.8*, 2002.
- [8] D. Petrovic, R. C. Shah, K. Ramchandran, and J. Rabaey, "Data Funneling: Routing with Aggregation and Compression for Wireless Sensor Networks," *IEEE International Workshop on Sensor Network Protocols and Applications*, 2003.
- [9] Rahul C. Shah and Jan Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks," *IEEE Wireless Communications and Networking Conference (WCNC)*, March 2002.
- [10] S. Ratnasamy, B. Karp, S. Shenker, D. Estrin, R. Govindan, L. Yin, and F. Yu, "Data-centric storage in sensornets with GHT, a geographic hash table", *ACM Mobile Networks and Applications (MONET)*, Volume 8 Issue 4, August 2003
- [11] The Vint Project, The ns Manual, <http://www.isi.edu/nsnam/ns/>
- [12] Anand Visvanathan, Jong-Hoon Youn and Jitender S. Deogun, "Hierarchical Data Dissemination Schemes for Large Scale Sensor Networks", the 40th annual IEEE International Conference on Communications, May 2005
- [13] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A Two-Tier Data Dissemination Model for Large-scale Wireless Sensor Networks", *ACM Mobicom 2002*.
- [14] J. Youn, R. Kalva, S. Park, "Efficient Data Dissemination and Aggregation in Large Wireless Sensor Networks", *IEEE VTC 2004 Fall*