A Quadtree-Based Data Dissemination Protocol for Wireless Sensor Networks with Mobile Sinks[*]

Zeeshan Hameed Mir and Young-Bae Ko

Graduate School of Information and Communication,
Ajou University, Suwon, Republic of Korea
{zhmir, youngko}@ajou.ac.kr

Abstract. The envisioned sensor network architecture where some of the nodes may be mobile poses several new challenges to this special type of ad hoc wireless network. Recently, researchers have proposed several data dissemination protocols based on some hierarchical structure mainly constructed by a source node to support mobile sinks. However, such a source-initiated hierarchical structure results in significant resource consumption as the number of source-sink pairs are increased. Additionally, stimulus mobility aggravates the situation, where several sources may build a separate data forwarding hierarchy along the stimulus moving path. In this paper, we propose a new data dissemination protocol that exploits "Quadtree-based network space partitioning" to provide more efficient routing among multiple *mobile* stimuli and sink nodes. Simulation results show that our work significantly reduces average energy consumption while maintaining comparably higher data delivery ratio.

Keywords: Wireless Sensor Networks, Mobility, Quadtree-based scheme.

1   Introduction

A wireless sensor network (WSN) consists of a number of tiny sensors that are densely deployed to monitor and interact with the physical world [1] [2]. Since each sensor can partially observe the large terrain, they must collaborate for efficient and reliable delivery of sensory data to the users (i.e., sinks). Sensor network systems spur an immense research potential for a wide range of new applications. One of the applications would be an agriculture production [3], for which the opportunity for sensor networks is explored to aid in a mobile work environment. Here, a mobile worker might be a robot or human equipped with sensor(s) collecting and transmitting data to the sink. Alternatively, static nodes can collectively track any moving target (i.e., stimulus) they detect and forward data through multi-hop communication towards the mobile sink.

Until recently, most of the research on data dissemination in WSN has focused on delivering data to *stationary* sinks. Sink mobility have been considered as a source of adversary to large-scale sensor networking, because the sink nodes are required to propagate their current position continuously so that future reports can be forwarded accordingly. In gradient-based routing protocols such as Directed Diffusion [4] and its variants, reverse paths are established among all the source-sink pairs based on some reverse path vector. This is achieved by periodically flooding control packets by sink nodes, which may cause significant overhead. Moreover, the reverse path vector changes with sink mobility which often causes more frequent flooding. In order to support sink mobility [5] and [6] proposed source-initiated hierarchal structure to the number of stationary or mobile sinks. However, the overhead of per source node grid construction and maintenance make these solutions unsuitable in presence of mobile stimulus.

In our proposed scheme, named as "Quadtree-based Data Dissemination (QDD)", a common hierarchy of data forwarding nodes is created by Quadtree-based partitioning of physical space into successive quadrants. Earlier, this approach has been utilized for addressing the sensor network field using location-based bit strings that represent the Quadtree address of the node [7]. In our approach, upon detecting a mobile stimulus, a source node calculates a set of rendezvous points by successively partitioning the sensor network space into four equally sized logical quadrants, and sends data packets to the nodes closer to the centroid of each successive partition. The mobile sink follows the same strategy for the data query packet dissemination. It starts from querying the immediate rendezvous node and continues until it finds the required data report. Since this procedure results in selecting same static sensor nodes, they form a common hierarchy for information forwarding and therefore results in lower overhead. The simulation results confirm that the cost of using separate hierarchy is considerably higher than our approach in terms of energy consumption, while the data delivery ratio is comparable for both approaches.

The remainder of the paper is organized as follows. Related work on data dissemination in WSN is covered in Section 2. Section 3 introduces our scheme, followed by simulation study in Section 4. In Section 5 we comment on certain design issues and future work. Finally, conclusion is provided in Section 6.

2   Related Work

Efficient and reliable collaboration among sensor nodes is a key to success for all types of applications in large-scale wireless sensor networks. The decision about how these sensor nodes communicate has a significant impact on the energy and bandwidth consumption [1] [2]. Furthermore, a particular architecture of mobile sensor networks will pose several new challenges.

Directed Diffusion [4] has been proposed as a distributed event detection mechanism for sensor networks. Motivated by the fact that the majority of data transmissions are destined in the direction of a sink, each sensor node sets up a *gradient* i.e., a direction state, towards its neighboring nodes in response to

sink's periodic flooding of interest packets. The sink node can be reached by traversing a high quality reverse path, selected among possibly multiple paths as the result of path reinforcement mechanism. Gradient-based Routing (GBR) [8] and Gradient Broadcast (GRAB) [9] took advantage of the freedom Directed Diffusion paradigm offers by attaching different semantics to the value of gradient. A common assumption each of these gradient-based data dissemination protocols make is the diffusion of gradient value throughout the sensor network, which costs significantly high in terms of overall routing overhead. Sink mobility adds further challenges, where it is essential for reliable data dissemination to maintain correct value of the gradient by a sink.

In TTDD [5], each source node constructs a uniform grid structure throughout the sensor field. A sink collects the reports on stimulus by first flooding its query within the local grid cell. The query packet is then traversed along the grid until either it reaches a source or any node that has yet received data from the source. While the query is disseminated over the grid, a reverse path is established towards the sink. In presence of mobile stimulus several sources may build a separate grid along the stimulus path. This situation can lead to excessive energy drain and increased packet collisions. Similar in spirit of the TTDD, Hierarchical Data Dissemination Scheme (HDDS) [6] is based on constructing a different hierarchy of data dissemination nodes from each source to potentially multiple sinks. In their work, a data forwarding node performs a load balancing by making another level of forwarding nodes. However, a scenario with mobile stimulus faces the limitations similar to the TTDD model.

The problem of sensor network space partitioning has been exploited for efficient execution of spatial queries in sensor networks. In [10], a distributed indexing structure based on R-tree spatial data structure is presented that enables querying of information related to a region of interest . However, in our proposed scheme we have utilized Quadtree-based partitioning to provide an efficient solution to the mobile stimulus and sink problem.

3   Quadtree-Based Data Dissemination (QDD)

The main goal of our scheme is to implement an efficient data dissemination protocol that supports both stimulus and sink mobility, by exploiting Quadtree-based sensor network space partitioning. Following are the assumptions our protocol design is based on:

- Sensor nodes are stationary and location aware, however some or all of the stimuli and sink nodes may be mobile. Since, sensor nodes will remain static a GPS-free localization [11] method can be utilized during the network initialization phase, where a node can also learn its one-hop neighbor location.
- Each sensor node knows the total sensor network space area N, given as $2^k \times 2^k$; where k = $\log_2$(N).
- Greedy geographical forwarding is used for both data and query packet dissemination.
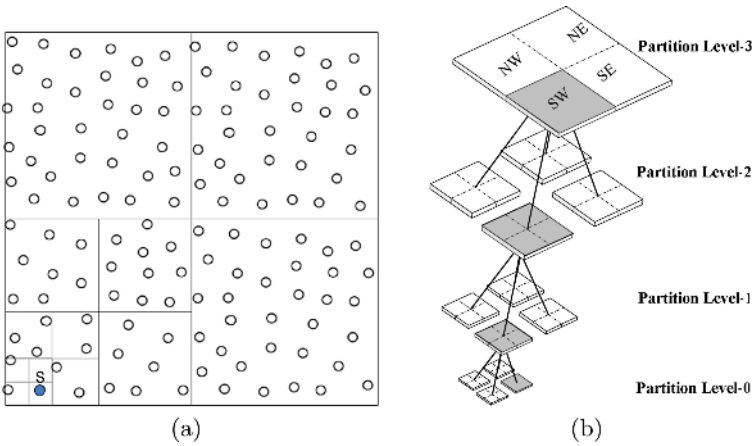
Fig. 1. (a) Sensor network space N partitioning. (b) Quadtree representation.

3.1   Quadtree-Based Network Space Partitioning

In this subsection we describe how the network space partitioning based on the concept of Quadtree is done in our QDD protocol. A sensor node S with location (X_s, Y_s), takes the complete sensor network space N as the root of a Quadtree, and logically partitions N into four equal sized quadrants. Each of these four quadrants North West (NW), South West (SW), North East (NE) and South East (SE) corresponds to a child of N, respectively, such that:

- The root N represents the entire network space, specified by $N.X_{LB} = 1$, $N.Y_{LB} = 1$, $N.X_{UB} = 2^k$ and $N.Y_{UB} = 2^k$; where $(N.X_{LB}, N.Y_{LB})$ are coordinates for lower left corner (lower bound) and $(N.X_{UB}, N.Y_{UB})$ are coordinates for upper right corner (upper bound) of a square, respectively.
- If P is the parent of child quadrant C, then values for $C.X_{LB}$, $C.Y_{LB}$, $C.X_{UB}$ and $C.Y_{UB}$, depends upon whether C is the NW, SW, NE, or SE child of P.

Next, each quadrant is considered as a separate parent and divided into further four sub-quadrants. Given the current location of node S (X_s, Y_s), this process is repeated for each quadrant, until node S remains the only node in a sub-quadrant (the leaf cell). This method requires a comparison at each partition level, to check if the current sub-quadrant is the leaf cell [7]. For example, if node S is in the NW quadrant of parent P (i.e., C = P.NW), then:

$$\{S \in P.NW:(C.X_{LB} \le X_s \le C.X_{UB}) \text{ and } (C.Y_{LB} \le Y_s \le C.Y_{UB})\}$$

Fig. 1(a) and (b) illustrate the network space partitioning into sub-quadrants by some node S and the corresponding Quadtree representation.
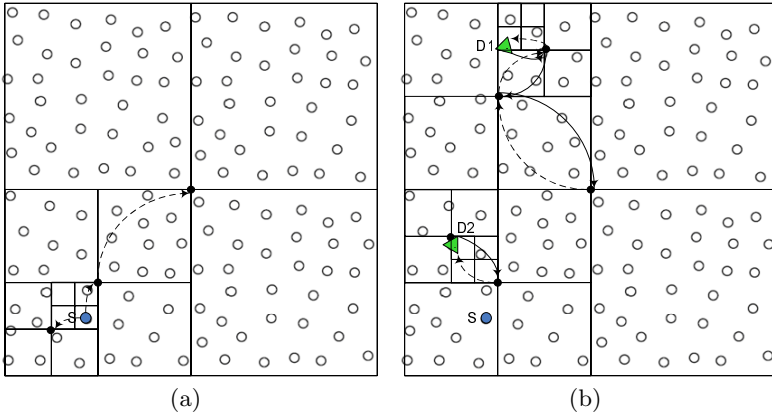
Fig. 2. (a) Rendezvous points and data packet forwarding by source node S (dashed lines). (b) Query packet forwarding by sink nodes D1 and D2 (solid lines).

3.2   Data Forwarding

The simplicity of our scheme is in the way a source node disseminates data. Upon detecting a stimulus, source node S performs a logical partitioning of sensor network field as explained in the previous section. For each partition level i represented by a square with lower corner and upper corner values set to $(i.X_{LB}, i.Y_{LB})$ and $(i.X_{UB}, i.Y_{UB})$ respectively, it calculates a list of central points called *rendezvous points* RP (x_i, y_i), given as:

$$x_i = i.X_{LB} + (i.X_{UB} - i.X_{LB})/2$$
$$y_i = i.Y_{LB} + (i.Y_{UB} - i.Y_{LB})/2 \ (0 < i \le k)$$

Fig. 2(a) shows the rendezvous points calculated by node S and the data forwarding process. It starts from its current location as the first rendezvous point (0th level) and forwards data packet to the *immediate rendezvous point* (1st partition level) using geographical greedy forwarding. If S is not itself the closest node to the immediate rendezvous point it looks into its neighbor table for a neighbor that is closest to that point and forward packet to it. Each node in turn repeats this process, until a node finds that no other node in its neighborhood is closer than itself. Now this node becomes the *rendezvous node*.

While forwarding data packets, each rendezvous node maintains a local table, including the source node's current location and identity, data message M, the previous rendezvous node's location, packet type and the sequence number, so that the duplicate entries related to same data packet can be identified and subsequently dropped. In addition, each table entry includes an expire field, that determines how long that entry would remain valid before it is discarded from the table.

3.3   Query Forwarding

Consider an application that tracks location and type of activity in a mobile work environment. When a stationary or mobile sink requires sensory data it calculates a list of rendezvous points and sends a query packet towards its immediate rendezvous point. On receiving a query packet, the immediate rendezvous node checks its local table for a valid data (that meets the criteria specified within the query packet and that is not expired yet). If this lookup for requested data fails, the query is propagated one level up in the hierarchy until it reaches either an intermediate rendezvous node that has the requested data or the kth partition level rendezvous node.

Fig. 2(b) exemplifies the query forwarding process, where two sink nodes D1 and D2 query for the data generated by source node S. This figure shows two different scenarios for query forwarding. Each sink sends query packets towards their respective immediate rendezvous point. As each rendezvous node receives the query packet, it checks for any valid data sink nodes are looking for. Based on its findings, it either sends the data packet towards the rendezvous point that is known to be closest to the sink (sink D2 in this case) or forwards the query packet one level up in the hierarchy, where it can find the requested data at the kth level of the hierarchy (sink D1, for example). As soon as a query matches with the required data in the local table, data is forwarded towards the sink by traversing the reverse path established during the query dissemination process.

Due to a random deployment of sensor nodes, there might exist two different nodes: one for holding data packets and the other for query packets. This situation can occur due to the following two reasons. For the first case, when there are several nodes close to the rendezvous point and therefore more than one node become candidates for keeping information, we have set a minimum distance threshold value to tell if it is the same location. Secondly, a network partitioning also results in this situation. Depending on the network neighborhood density, we set a maximum distance threshold for a node to become a rendezvous node. While forwarding packets towards the rendezvous point RP, if a node finds none of its neighbors closer than itself and its distance from RP is greater than this value it simply performs one-hop flooding.

Generally, there are two possible choices for a sink to receive data packets, either a single event or multiple continuous events. In the former case each query by a sink is acknowledged with the requested data only once whereas in the later case a sink receives multiple data continuously in response to its query. In order to receive continuous data packets, rendezvous nodes also maintain the query entries in their local table. The purpose behind storing these entries is twofold. First, while the query packet traverses the hierarchy of rendezvous points, it establishes a backward path towards the sink and secondly to reduce the amount of time, sink node has to wait for newly announced data. Upon receiving a new data packet from the source, a rendezvous node immediately forwards it towards the sink by following the path stored in local table. The degree of how long these entries remain persistent is controlled by the frequency with which sink nodes
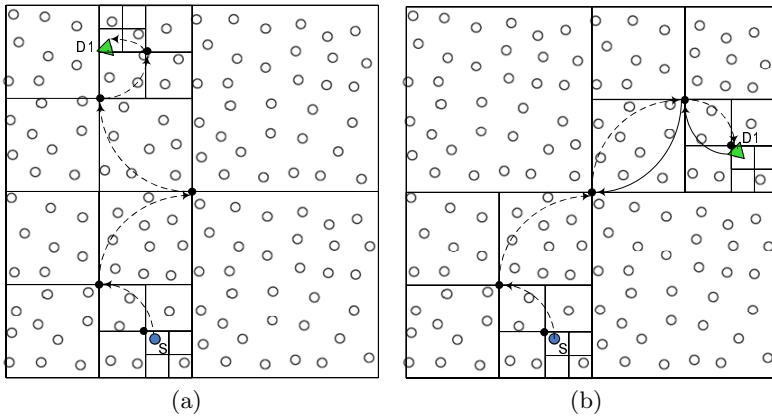
Fig. 3. (a) Stimulus mobility scenario (b) Sink D1 mobility scenario

transmit query packets. Each query packet refreshes the query entries in table with the location of the last rendezvous point that is serving the sink and the amount of time those entries will remain valid (for example, a query might have a DURATION clause that specifies query lifetime).

Here it is noteworthy that how this scheme accommodates stimulus and sink mobility without any associated overhead and delays. Each time a mobile sink has to inquire about any new updates; it just needs to send its query packet up to that level where the latest updates are available. On the other hand, a source node sends data to the set of calculated rendezvous points. Since the sensor nodes are immobile the information flows through same intermediate rendezvous nodes. Fig. 3(a) and Fig. 3(b) show two different network setups as a result of stimulus and sink mobility.

4   Performance Evaluation

For the purpose of performance evaluation, we have performed extensive simulations. A comparative study is carried out with TTDD model by varying the number of source-sink pairs. Then, the impact of stimulus mobility on the proposed scheme is shown as a function of maximum stimulus speed. The set of experiments presented here describe results obtained by evaluating our scheme in the presence of multiple mobile stimuli and sink nodes.

4.1   Simulation Environments

We have implemented and tested our protocol performance in ns-2 [12]. In order to ensure a fair comparison with the TTDD model, we set simulation parameters comparable to those used in [5]. This includes simulation of IEEE MAC 802.11 DCF and node energy model which consumes 0.395W, 0.660W and 0.035W per
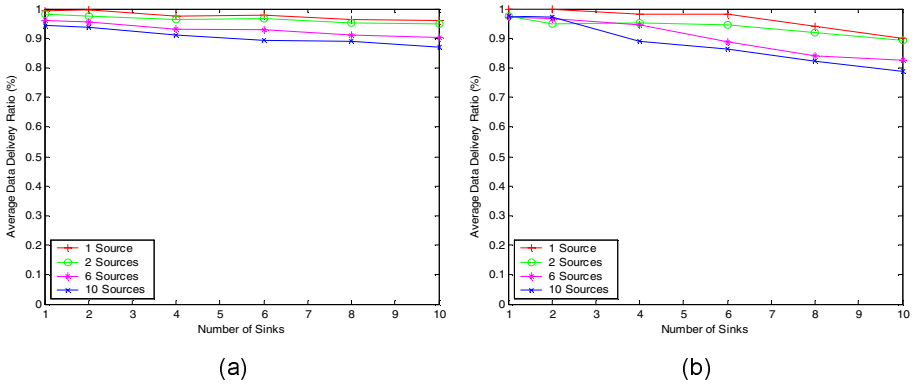
(a)

(b)

Fig. 4. Comparison in terms of Data Delivery Ratio: (a)QDD and (b)TTDD
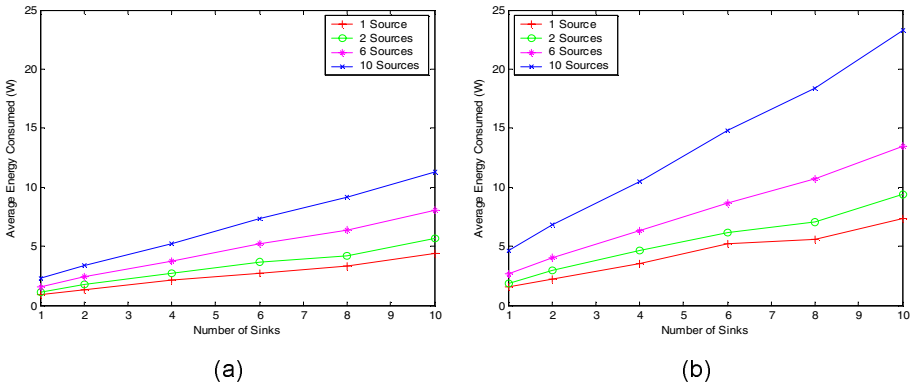

(a)

(b)

Fig. 5. Comparison in terms of Energy consumed: (a)QDD and (b)TTDD

receiving, transmitting and idle modes, respectively. In our simulation model, 200 sensor nodes are randomly placed on a 2048 x 2048m^2 grid. For different set of simulations, speed and pause time of stimuli and sink nodes vary accordingly with Random Waypoint model used as the mobility model. Total simulation duration is 200 seconds. During simulations the data rate is 1 packet per second, so there are total 200 data packets sent. All the simulation results are averaged over six different scenarios. Following metrics are used to evaluate our proposed scheme. (a) *Average Data Delivery Ratio*, obtained by comparing total number of data reports received by a sink with total packets generated by a source, which is further averaged over total number of source-sink pairs. (b) *Average Energy Consumed*, represented as the ratio between the total energy consumed during a simulation run and the total number of sensor nodes. (c) *Average Delay*, defined as the total time elapsed between the data generation by a source node and reception by a sink over total number of data packets received per source-sink pairs.
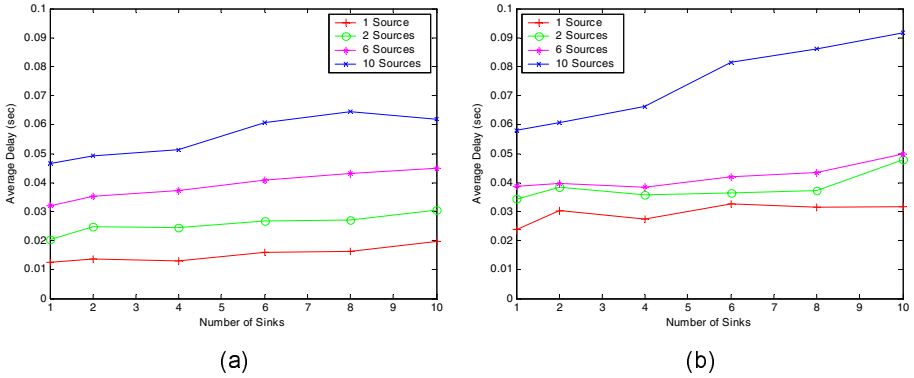
Fig. 6. Comparison in terms of Delay: (a)QDD and (b)TTDD

4.2 Simulation Results

Experiments with Stationary Stimuli and Mobile Sinks: For the first set of simulation results, number of source-sink pairs are varied to make a performance comparison with TTDD scheme in presence of mobile sinks. Mobile sink nodes could attain a maximum speed up to 10m/s with 5 seconds pause time. The stimuli remain static throughout the simulation time. Beginning from the average data delivery ratio Fig. 4(a) and Fig. 4(b), show that for both protocols most of the data packets are delivered. Even though the results are quite comparable, the delivery ratio for TTDD scheme falls more consistently as the number of source-sink pairs grows.

Fig. 5(a) and Fig. 5(b) compare the energy consumption overhead. Here, it demonstrates that the energy requirements of TTDD is substantially higher than that of our scheme. However, for the proposed scheme it increases linearly with the increase in number of sources. In TTDD every source sends data packets to four different corners of each grid cell recursively to build a network wide virtual grid. This practice is in contrast to the way data is disseminated in our scheme, where data packets are required to send to only one rendezvous node per partition level. As the number of source nodes increases, a separate grid structure construction and maintenance on per source basis results in higher cost, both in terms of packet overhead and energy consumption.

Fig. 6 presents the average delay. We can see that in both protocols, the delay increases with the increase in number of source-sink pairs. Once again the reason behind TTDD's delay is the excessive number of data packets sent by source nodes throughout the sensor network and local query packets flooded by the sink nodes. In our case, since all the source and sink nodes forward data along the common hierarchy, incurred delay increase as the number of sources increases. However, the proposed scheme performs better because there is no extra delay associated with tracking mobile sinks.
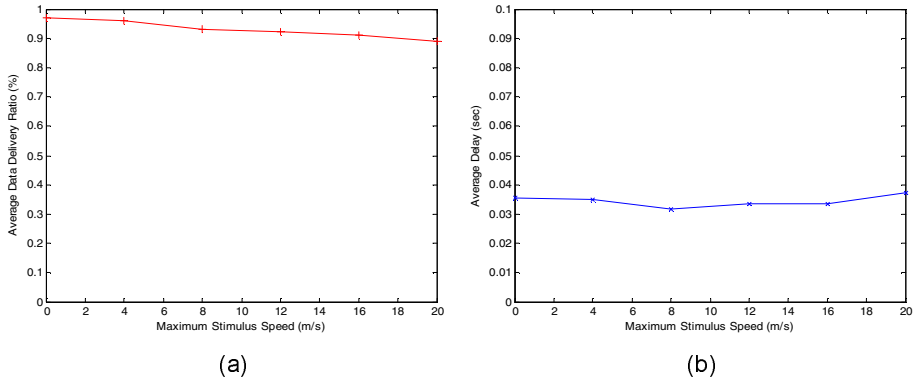
Fig. 7. Impact of stimuli mobility on QDD: (a)Data Delivery Ratio (b)Delay

Experiments with Mobile Stimuli and Sinks: TTDD model has its main focus on scenarios with sink mobility only. In this subsection we describe the impact of stimuli mobility on our protocol as the function of maximum stimulus speed. In these simulation scenarios, stimuli speed varies from 0 to 20m/s where as the sink node speed is chosen randomly between 0 and 10m/s with 5 second pause time for both. For a total of 5 stimuli and 5 sinks deployed, Fig. 7(a) shows the average data delivery ratio. As the stimulus speed increases the delivery ratio decreases gradually. At higher speed stimuli tend to change there immediate quadrants more frequently, however since the higher order partition levels remains the same, therefore most of the data packets are delivered to sink nodes. The average energy consumption statistics remains stagnate at 5W over the changing stimuli speed because of two observations. First of all, the inherent support for stimuli mobility Quadtree-based partitioning offers by making the data delivery to sink nodes independent of the current position of the stimuli and secondly there is no overhead associated with tracking mobile sink. In Fig. 7(b), the average delay increases slightly with speed mainly because the situations where stimuli change quadrants relatively fast. This situation often results in frequent change of lower level rendezvous points and therefore sink queries have to traverse up towards higher levels in hierarchy to fetch data.

5   Discussions

5.1   Distribution of Total Energy Consumed Among Sensor Nodes

Selecting rendezvous nodes based on the geometric argument results in using the same nodes, if needed, to forward information between different source-sink pairs. In order to reduce the risk of exhausting the batteries of these nodes, the responsibility of forwarding data and query packets can be rotated among nodes that are within their close vicinity. Specifically, a rendezvous node can delegate any of its neighboring nodes to handle future traffic for certain period of time and
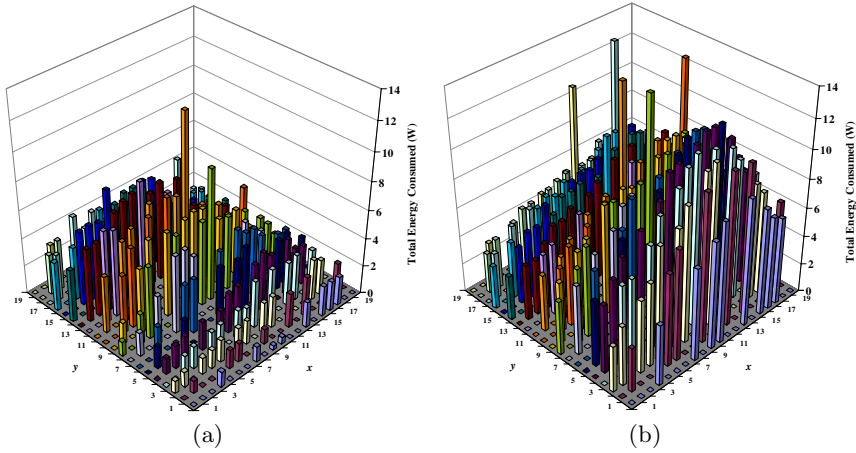
Fig. 8. Distribution of total energy consumed among sensor nodes: (a)QDD (b)TTDD

locally floods its decision to one-hop neighbors. In this way, a packet forwarding node can then use the new designated node for storing data and control entries.

We further studied the distribution of total energy consumed among sensor nodes in the network and contrasted our results with the TTDD model. The simulations are performed with one mobile sink and ten static stimuli, where the itinerary of the sink is randomly selected during the experiments. Sensor network space is divided into 100 x 100m^2 sized non-overlapping blocks for the network size given in the previous section. In Fig. 8(a) and Fig. 8(b), each 3-D bar represents cumulative energy consumed by all the nodes within a block for the proposed scheme and the TTDD model, respectively. Although, energy consumption is highly variable and depends on the current location of the sink, an important observation about our approach is that the nodes that form a common backbone hierarchy experience the highest energy consumption while the energy levels of most of the other nodes remains partially intact. It is therefore, obvious that in our case the network lifetime is defined by a fewer nodes that are along the diagonals of the sensor field. In TTDD, however a separate hierarchy for every source-sink pair is maintained, therefore the energy consumption is equally high throughout the network. This increases the probability to exhaust the battery energy of majority of the nodes, leading to network partitioning and reduced network lifetime.

Moreover, recent studies show that deployment of higher energy and communication capacity nodes can be exploited to leverage the overall system capabilities [13]. The fact that in our scheme the routing decisions are biased to use the common hierarchy can be combined with placement of superior nodes alongside diagonals to further improve the network performance. We leave these design alternatives to be addressed as future work.

5.2   In-network Data Aggregation

Since the forwarding paths along the diagonals of sensor fields are shared among all source-sink pairs, it provides an opportunity for similar data to meet at some common rendezvous point. Data from multiple sources can be aggregated and replaced by a single data packet and forwarded towards the destined sink. Although our proposed scheme can achieve further performance gain by in-network data aggregation, we are intending to explore this avenue in our future work.

6   Conclusion

Environment monitoring application varies greatly with one common goal of detecting and reporting on the phenomena of interest to the sink. In this paper we have proposed an efficient and simple, Quadtree-based data dissemination protocol for large scale wireless sensor networks that supports both stimulus and sink mobility. By making the data dissemination process independent of each others current location, our work provides an efficient solution for mobile stimulus-sink problem. Through simulation results it is shown that a common hierarchy results in reduced communication overhead and significant energy saving against network wide data and control packet transmission overhead.

References

1. D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the World with Wireless Sensor Networks", *Proc. ICASSP'01*, May 2001.
2. D. Estrin and R. Govindan, "Next Century Challenges: Scalable Coordination in Sensor Networks", *Proc. MobiCom'99*, Aug. 1999.
3. J. Burrell, T. Brooke and R. Beckwith, "Vineyard Computing: Sensor Networks in Agricultural Production", *IEEE Pervasive Computing*, 3(1):38-45, Jan. 2004.
4. C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", *Proc. Mobicom'00*, Aug. 2000.
5. F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A Two-Tier Data Dissemination Model for Large-scale Wireless Sensor Networks", *Proc. Mobicom'02*, Sept. 2002.
6. A. Visvanathan, J.H. Youn, and J. Deogun, "Hierarchical Data Dissemination Scheme for Large Scale Sensor Networks", *Proc. ICC'05*, May 2005.
7. C. Cimen, E. Cayirci, and V. Coskun, "Querying Sensor Fields by using Quadtree based Dynamic Cluster and Task Sets", *Proc. IEEE MILCOM'03*, Oct. 2003.
8. C. Schurgers and M. B. Srivastava, "Energy Efficient Routing in Wireless Sensor Networks", *Proc. IEEE MILCOM'01*, Oct. 2001.
9. S. Lu, F. Ye, G. Zhong, and L. Zhang, "Gradient Broadcast: A Robust Data Delivery Protocol for Large-scale Sensor Networks", *Proc. IPSN'03*, April 2003.
10. M. Demirbas and H. Ferhatosmanoglu, "Peer-to-Peer Spatial Queries in Sensor Networks", *Proc. IEEE P2P'03*, Sept. 2003.
11. N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less Low-Cost Outdoor Localization for Very Small Devices", *IEEE Personal Communication*, 7(5):28-34, Oct. 2000.
12. ns-2 network simulator. http://www.isi.edu/nsnam/ns.
13. M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh, "Exploiting Heterogeneity in Sensor Networks", *Proc. IEEE INFOCOM'05*, March 2005.