Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 763015, 10 pages
doi:10.1155/2012/763015

*Research Article*

# A Virtual-Ring-Based Data Storage and Retrieval Scheme in Wireless Sensor Networks

**Xingpo Ma, Jianliang Gao, Weiping Wang, and Jianxin Wang**

*School of Information Science and Engineering, Central South University, Hunan, Changsha 410083, China*

Correspondence should be addressed to Jianliang Gao, gaojianliang@csu.edu.cn and Weiping Wang, wpwang@csu.edu.cn

Received 21 July 2012; Accepted 29 September 2012

Academic Editor: Limin Sun



In-network data storage and retrieval is one of the most important problems in wireless sensor networks. Many schemes have been proposed to solve this problem. However, most of them do not take the frequencies of event and query into consideration. In fact, the frequencies of event and query are very important factors in real applications of in-network data storage and retrieval. In this paper, we introduce a virtual-ring-based data storage and retrieval scheme, which is called VRS, to solve the problem. VRS divides the whole sensor network field into some virtual rings. According to the frequencies, one of the virtual rings is selected out as the rendezvous ring, which plays the role of a bridge between the information consumers and the information producers. Extensive experiments have been done to evaluate the performance of the proposed scheme, VRS. Simulation results show that VRS outperforms the existing work in load balance, delay of data storage and retrieval, and the lifetime of the sensor networks.

## 1. Introduction

In wireless sensor networks, full data collection [1, 2] is a traditional method for information consumers to get the useful information. However, this method triggers at least two problems. On the one hand, not all the data collected by the sink node are needed by the users, and it is a waste of energy to send the useless event data to the sink node. On the other hand, the method leads to bad load balance performance, because sensor nodes around the sink node have to take much more load than other sensor nodes.

In this paper, we focus on the in-network data storage and retrieval problem in wireless sensor networks, where information consumers (sinks, users) search for the desired event data produced by information producers (sources). There are many such applications. For example, a sensor network might be deployed in a battlefield for the soldiers to know the battlefield situation better. A soldier in the battlefield might be interested in whether there are some hostile tanks nearby, but it is hard for the soldier to be aware of when visibility is low (either due to smoke or at night). However, the sensor nodes that detect the tanks can transmit the information to the soldier. The problem is how can the sensors that detect the tanks send the information to the soldier without knowing where the soldier is. There are mainly two challenges for the problem of in-network data storage and retrieval. First, neither information producer nor information consumer knows where the others are. Second, the sensor nodes are resource constrained and their communication radii are short.

Recently, a lot of in-network data storage and retrieval schemes [3–13] have been proposed. In most of the existing schemes, the frequencies of event and query are less considered, while in fact they play an important role on the efficiency of the schemes. Intuitively, if the frequency of event is high and the frequency of query is low, the data replicas of event should be stored near the information producers. On the contrary, the data replicas of event should be stored near the consumers. Recently, Yu et al. [3] has proposed an in-network storage and retrieval scheme, in which the frequencies of event and query are taken into consideration when selecting the optimal storage node. However, there is only one storage node in the scheme, which leads unbalanced load distribution and can shrink the lifetime of the sensor networks.

There are mainly three contributions in the paper. Firstly, we propose a virtual ring model for the wireless sensor networks. The virtual ring model can be formed in a totally distributed way. Secondly, we propose a novel in-network data storage and retrieval scheme based on the virtual ring model for wireless sensor networks, where the frequencies of event and query are considered. Thirdly, we have done a lot of experiments to evaluate the performance of our scheme and compare it with the scheme presented in [3], which also considers the event and query frequencies. Simulation results show that VRS outperforms the existing work in load balance, delay of data storage and retrieval, and the lifetime of the sensor networks. Besides, the results also show that VRS can ensure stable lifetime of a sensor network as the sensor nodes increase when the effective time of event data is short. The lifetime of the sensor networks in this paper is defined as the time before the first node dies.

The rest of this paper is arranged as follows. In Section 2, we introduce the related work. Section 3 presents the sensor network model as well as the assumptions of our work. In Section 4, we provide our in-network data storage and retrieval scheme named VRS in detail. The simulation results and analysis are given in Section 5. Section 6 makes a conclusion.

## 2. Related Work

Existing in-network data storage and retrieval schemes mainly follow three basic approaches. The first one is based on the transmission trail [4–7], the second one is based on the deterministic storage node/nodes [3, 8–13], and the last one is the combination of the former two approaches [14–16].

The main idea of the transmission-trail-based schemes is as follows. Find two intersecting trails: one is for data replication and the other one is for transmitting the queries. Thus the desired data can be retrieved at the crossover point/points of the two trails if they intersect with each other. Rumor routing [4] is an early work of such kind of schemes. Rumor routing creates paths leading to each event. When a query is generated, it will be transmitted in the manner of random walk until it finds the event path. Rumor routing cannot guarantee good successful retrieval rate and is not energy effective, since the directions of the forwarding route are random and dynamic. In double ruling [5], the data replication trail as well as the query transmitting trail is a closed curve, which is projected from a circle on a sphere with the same center as the sphere to a plane. It is obvious that any two circles on the sphere with the same center must intersect, and their projections also intersect. In GLIDER [6], all sensor nodes are divided into tiles using a number of landmarks. The event data are first hashed into a tile, and then they are replicated along three curves towards three neighboring landmarks. The event data in a tile are retrieved by transmitting a query either following a zigzag curve that visits the tile's boundary with each guide or toward the next landmark on the globally planned combinatorial path [6]. In RDRIB [7], a cycle of four axes as the virtual boundary is constructed. All event data are firstly transmitted into the interior of the virtual boundary, and then the data are replicated and retrieved along the path towards the axes using the double ruling technique.

The main idea of the schemes is straight-forward. In those schemes, one or more sensor nodes is/are selected as the storage node/nodes, which is/are known by all the consumers and producers. Thus, data storage and retrieval can be achieved on the storage node/nodes. GHTs [8] are a classic work of this category. One of the shortcomings in GHTs is that it brings a hotspot problem [9]. To solve the hotspot problem, a lot of works were done later. DBAS [10] and DCAAR [11] hash an event type or an attribute of the event type to a grid region, instead of a point. EHS [12] hashes an event type to a point in a face, and the nodes in the face are responsible for storing the event data of that type. Another problem in GHTs is that data retrieval may fail when the nodes nearest to the hashed points move. To solve this problem, C-DCS [13] proposed a cluster-based approach.

Some other schemes were proposed combining the former two basic ideas. A data storage method based on rings was proposed in [14]. Although the load among sensor nodes is balanced using this method, it cannot be independent of time synchronization technique. Besides, multiple sink nodes need to be deployed, which increases the cost of sensor networks. Xie et al. [15] proposed a ring-based multiresolution data storage scheme, where $n$ index trees were built over the virtual rings, and each tree kept several replicas of each data item with different data resolution. It is evident that the cost of replicating data items is huge if data generating frequencies are high. The idea of hashing an event type to a point was also used in [16]. The difference between the scheme proposed in [16] and others of the same kind is that the index instead of the event data is stored on the nodes in the virtual ring around the hashing point. It is easy to see that if the data generating rate is high and the query rate is low, the scheme in [16] is efficient. However, if the data generating rate is low and the query rate is high, the scheme in [16] is not efficient any longer. This is because event data are stored near the detecting nodes, and each query must be sent to the index nodes first and then sent to the storage nodes to get the event data.

The literatures mentioned above do not consider the frequencies of event and query. Yu et al. [3] proposed an adaptive data storage and retrieval scheme, where the frequencies of event and query were considered. However, the scheme only chooses one node as the storage node, thus the hotspot problem still exists.

## 3. Network Model and Assumptions

In our sensor network model, $N$ sensor nodes as well as one sink node are deployed randomly in a nonbrand field. We assume that the outline of the sensor network field is convex polygon, and the other kind of field shapes is beyond the scope of this paper. We also assume that the approximate location of the central point of the field and the approximate shortest distance from the central point to the boundary of the field are preloaded on each sensor node. We hold that it is a reasonable assumption, because those data can be got by measuring the field before the sensor nodes are deployed. Let $L$ denote the approximate shortest distance from the central point to the boundary of the sensor network field in the following of this paper.

Packet loss is allowed in our network model. We assume that sensor nodes know their geographic locations. This can be achieved by means of the GPS [17] or some other location service methods [18, 19]. We assume that every sensor node has the same communication radius $R$ and the network is connective. The whole sensor network formulates a connected unidirectional graph $G = \{V, E\}$, in which $V$ is the set of nodes and $E$ is the set of edges. $V$ and $E$ can be described as follows:

$$V = \{v_0, v_1, \ldots, v_N\},$$
$$E = \left\{\left(v_i, v_j\right)\right\} \quad \left(\mathrm{DIS}\left(v_i, v_j\right) < R\right), \tag{1}$$

where $|V| = N + 1$, $v_0$ is the sink node, and $\mathrm{DIS}(v_i, v_j)$ denotes the distance between $v_i$ and $v_j$. We assume that any sensor node can be a producer or a consumer, and the role of a sensor node can convert between a producer and a consumer. In the following sections, $f_d^i$ denotes the event data packet generating frequency of producer $i$, and $f_q^i$ denotes the query frequency of consumer $i$.

## 4. Virtual-Ring-Based In-Network Data Storage and Retrieval Scheme

In this section, we present a novel in-network data storage and retrieval scheme, named VRS, in detail.

*4.1. Overview of VRS.* The main idea of VRS is given as follow. As shown in Figure 1, the sensor network field is divided into many virtual rings, one of which is selected out as the rendezvous ring to mainly take the responsibility of data storage and retrieval. Specifically, the event data will be sent to the rendezvous ring for storing if they are generated outside the rendezvous ring, and the event data will be stored locally if they are produced inside the rendezvous ring. Data retrieval can be achieved by querying the nodes in the rendezvous ring.

The virtual rings can be formed in a totally distributed way. For ease of presentation, we give each of the virtual rings an ID. From the central point of the sensor field, the farther the virtual ring is, the larger the ID of the virtual ring is. ID of the innermost virtual ring is set to zero, and the value deviation between the IDs of two adjacent virtual rings is one. For each node, it first computes the straight-line distance from itself to the central point of the sensor network field, and then it can decide which virtual ring it belongs to by simple calculation. The method for a sensor node $S$ to determine the ID of the virtual ring which it belongs to is as follows.

(i) If the distance from $S$ to the central point of the field is not bigger than $L - L\%R$, $S$ ascertains itself in the virtual ring with ID $k$ if the following inequality holds:

$$R^* k < \mathrm{DIS}(S, \mathrm{CENTER}) < R^*(k+1) \quad (k \geq 0,\ k \in Z), \tag{2}$$

where $\mathrm{CENTER} = (x_{\mathrm{center}}, y_{\mathrm{center}})$ denotes the location of the central point in the sensor network field.

(ii) If the distance from $S$ to the central point of the field is bigger than $L - L\%R$, the virtual ring ID that $S$ locates in is $\lfloor L/R \rfloor$.

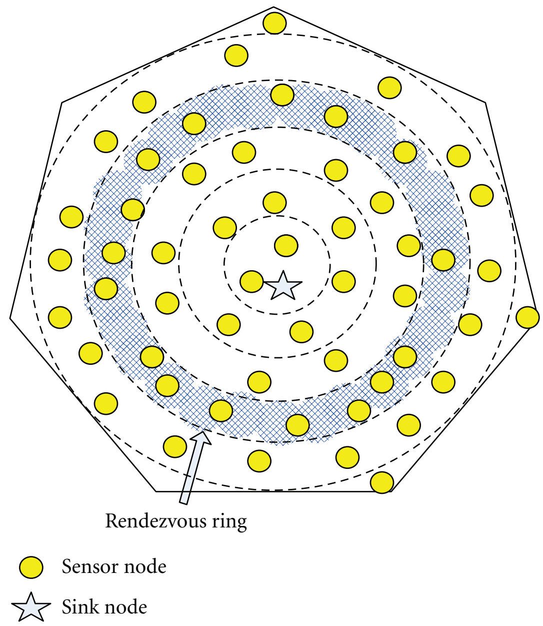


Figure 1: Example of the rendezvous ring in the virtual ring model. (The colored one is the rendezvous ring).

In the following parts, we will introduce the way to determine the rendezvous ring and the procedure of data storage and retrieval in detail.

*4.2. Rendezvous Ring Determination.* Before determining the rendezvous ring, the sink node first collects some information of each sensor node from the sensor network. The information includes node locations, the event frequencies, and the query frequencies. Then the sink node determines the rendezvous ring in a centralized way. Finally, the sink node broadcasts the ID of the rendezvous ring to all the sensor nodes in the sensor network. The strategy for the sink node to determine the rendezvous ring is as follows. The virtual ring with ID $k$ is determined as the rendezvous ring if it satisfies the following equation:

$$\frac{E_{\mathrm{total}}^k}{n^k} = \min\left\{\frac{E_{\mathrm{total}}^i}{n^i},\ 0 \leq i < \left\lfloor \frac{L}{R} \right\rfloor,\ i \in \mathrm{Z}\right\}, \tag{3}$$

where $E_{\mathrm{total}}^i$ denotes the total cost of data storage and retrieval for all the sensor nodes in a certain time period $T$ when the ring with ID $i$ is selected as the rendezvous ring and $n^i$ denotes the number of nodes in the ring whose ID is $i$. Obviously, two metrics have been considered by using this strategy: the total energy consumption and the load balance.

To calculate $E_{\mathrm{total}}^i$, we have

$$E_{\mathrm{total}}^i = \sum_{j=1}^{N} E_j^i \quad \left(0 \leq i < \frac{(L - L\%R)}{R},\ i \in \mathrm{Z}\right), \tag{4}$$

where $E_j^i$ denotes the energy consumption of data storage and retrieval for node $j$ in time period $T$ when the virtual ring with ID $i$ is selected as the rendezvous ring. Thus, the main job to calculate $E_{\text{total}}^i$ is to calculate $E_j^i$, $(0 \leq i < (L - L\%R)/R,\ 0 < j \leq N,\ i, j \in Z)$.

Since we assume that a sensor node can be an information producer or an information consumer at different time, $E_j^i$ must contain two aspects: $E_{j\text{-storage}}^i$ and $E_{j\text{-retrieval}}^i$, where $E_{j\text{-storage}}^i$ denotes the energy consumption for node $j$ to transmit the event data to ring $i$ when node $j$ plays the role of an information producer during time period $T$, and $E_{j\text{-retrieval}}^i$ denotes the energy consumption for node $j$ as a consumer to retrieve the desired event data during time period $T$. Thus we have the following equation:

$$E_j^i = E_{j\text{-storage}}^i + E_{j\text{-retrieval}}^i. \tag{5}$$

$E_{j\text{-retrieval}}^i$ also contains two aspects, which are $E_{j\text{-query}}^i$ and $E_{j\text{-result}}^i$. $E_{j\text{-query}}^i$ denotes the energy consumption of query transmission, and $E_{j\text{-result}}^i$ denotes the energy consumption of transmitting the query results to node $j$. Then the following equation holds:

$$E_j^i = E_{j\text{-storage}}^i + E_{j\text{-query}}^i + E_{j\text{-result}}^i. \tag{6}$$

In Section 4.4, we will present the way to calculate $E_{j\text{-storage}}^i$, $E_{j\text{-query}}^i$, and $E_{j\text{-result}}^i$ in detail.

*4.3. Event Data Storage.* In our data storage strategy, at least one copy of each event data should be stored by the nodes in the rendezvous ring. To implement this, all the event data produced by the producers outside the rendezvous ring should be sent to the rendezvous ring. Besides, any sensor node which the event data pass by keeps a replica of the event data. If a producer locates in the rendezvous ring, it just needs to save the data in its own cache.

Now we pay attention to the event data that are produced by producers outside the rendezvous ring. Since we use GPSR [20] as our routing protocol, the destinations of the event data packets should be carefully designed. In this paper, the destinations are designed as follows. If the ID of the ring which any producer is located in is bigger than the ID of the rendezvous ring, the destination of each event data packet should be the central location of the sensor network field. Otherwise, the destination should satisfy the following condition $\Omega$:

$$\text{condition } \Omega\text{: } \text{DIS}(\text{DEST}, \text{CENTER}) > \text{ID}_r * R, \tag{7}$$

where DEST denotes the destination of the event data packet, DIS(DEST, CENTER) denotes the distance between DEST and the central point of the sensor network field, and $\text{ID}_r$ denotes the ID of the rendezvous ring. The event data packet must be intercepted by the nodes in the rendezvous ring in this way. In VRS, to reduce and balance the energy consumption, the destination of an event data packet generated in the virtual ring whose ID is smaller than $\text{ID}_r$ should satisfy another two conditions as follows.

*Condition* $\Psi$. It should be in the same line with the producer and the central point of the sensor network field.

*Condition* $\mathfrak{R}$. The central point of the sensor network field should not be between the destination and the producer in the line.

When the destination of the event data packet is worked out, the producer sends the event data packet to the destination using GPSR routing protocol, and the event data packet must pass the rendezvous ring. When the event data packet is received by the first node in the rendezvous ring, it will be stored by the node, and stop being transmitted.

*4.4. Event Data Retrieval.* In the paper, we are interested in real-time event information discovery in wireless sensor networks. If there are the desired event data in a sensor network, they can be found in the rendezvous ring, because at least one replica of each event data is kept on some node/nodes in the rendezvous ring.

Two cases should be considered when executing the query procedure. One case is that the consumers are located outside the rendezvous ring, while the other case is that the consumers are located inside the rendezvous ring. In fact, if the query procedure in the former case is known, it is also clear in the latter case. We only present the query procedure in the former case because of the limited space.

The query procedure for the consumers who are located outside the rendezvous ring consists of three stages. The first stage is to transmit queries towards the rendezvous ring, the second stage is to process the queries in and/or outside the rendezvous ring, and the third stage is to send query results back to the consumers. The way to transmit queries to the rendezvous ring is just the same as the way to transmit the event data to the rendezvous ring, so we will not repeat it. Here, we mainly answer the question how the queries are processed in and outside the rendezvous ring.

The query procedure outside the rendezvous ring is not complicated. When a node outside of the rendezvous ring receives a query, it only needs to check whether the desired event data in its own cache exist. If there are, it sends the event data back to the consumer. Otherwise, the node continues routing the query to the rendezvous ring.

The query procedure inside the rendezvous ring is a little complex. To process queries inside the rendezvous ring efficiently, we have established a polar coordinate system in the sensor field. The polar coordinate system takes the center of the sensor field as the pole, takes the ray, which originates from the center and points to the right-hand side horizontally, as the polaraxis, and takes a counterclockwise direction as the positive direction of the polar angle.

Now, we describe how the queries are processed in the rendezvous ring. Every node in the rendezvous ring has two roles. One role is a leading node and the other role is an ordinary node. These two roles can exchange with each other. Suppose a consumer $C$ who is located outside the rendezvous ring creates a query QRY. $C$ sends QRY towards the rendezvous ring. Suppose that node $U$ in the rendezvous ring receives QRY. Then node $U$ becomes a leading node, and it checks its own cache to see whether there are event data that meet the requirement of the query. If there are such event data, node $U$ sends the event data to the consumer using GPSR routing protocol [14]. If there are no such event data, $U$ generates a copy of QRY, and broadcasts the copy to its one hop neighbors. Suppose that a node $V$, which is $U$'s neighbor in the rendezvous ring, receives the copy. It checks its own cache to see whether there are the event data that satisfy the requirement of the copy. If there are such event data, node $V$ sends the event data to node $U$. Otherwise, $V$ sends a null message to $U$. We can make the nodes in the rendezvous ring distinguish the query QRY with its copies by setting different message types.

If node $U$ discovers the desired event data from its one hop neighbors, it sends them to the consumer. Otherwise, node $U$ chooses one of its one-hop neighbors as the next hop of the query QRY and sends QRY to the next hop. Suppose that node $W$ is the next hop of QRY. To improve the efficiency of query processing, we specify that node $W$ should satisfy the following two conditions:

(i) it should be in the rendezvous ring;

(ii) it should be as far as possible to node $U$ counterclockwise.

We can select out the node that satisfies the second condition with the help of the polar coordinate system. Node $W$ then becomes a leading node and does the same thing as node $U$. This procedure continues until the desired event data are discovered or all the nodes in the rendezvous ring are accessed. Figure 2 illustrates the procedure of event data retrieval in VRS.

*4.5. Detailed Energy Calculation.* We have explained the way to determine the rendezvous ring in Section 4.1. The key step in the procedure of determining the rendezvous ring is to calculate $E_j^i$, which consisted of $E_{j\text{-storage}}^i$, $E_{j\text{-query}}^i$, and $E_{j\text{-result}}^i$. Suppose that the time period $T$ is a unit time interval. Then $E_{j\text{-storage}}^i$ can be calculated as follows:

$$E_{j\text{-storage}}^i = f_d^j \times s_d \times abs\left(\mathrm{ID}_{\text{ring}}^j - i\right) \times (\delta_{\text{send}} + \delta_{\text{receive}}), \tag{8}$$

where $\mathrm{ID}_{\text{ring}}^j$ denotes the ID of the ring in which node $j$ is located, $\delta_{\text{send}}$ denotes the energy consumption when sending one byte of data, and $\delta_{\text{receive}}$ denotes the energy consumption when receiving one byte of data.

To calculate $E_{j\text{-query}}^i$, we need to estimate the average hop count $H_{\text{query}}$ for the queries to travel in the rendezvous ring. This depends on where the desired event data will be discovered. Let $\theta$ denote the average angle that the queries have rotated around the center of the field in the rendezvous ring before meeting the desired event data. $H_{\text{query}}$ can be calculated as follows:

$$H_{\text{query}} = \theta \times \left(i + \frac{1}{2}\right) \times \frac{R}{R} = \theta \times \left(i + \frac{1}{2}\right). \tag{9}$$

We assume that each event data item has an effective time interval $\tau$ when it is stored on a node. When the effective time expires, the event data item will be cleared. Let $P^l$ denote the probability that there are some effective event data on node $l$. Then we have

$$P^l = \begin{cases} f_d^l \times \tau, & \left(f_d^l \times \tau < 1\right), \\ 1, & \left(f_d^l \times \tau \geq 1\right). \end{cases} \tag{10}$$

The average probability that there are some effective event data on a node is

$$P_{\text{average}} = \frac{\sum_{l=1}^{N} P^l}{N}. \tag{11}$$

Let $P$ denote the probability that there are some effective event data among $n$ nodes, and the following equations hold:

$$P = 1 - \left(1 - P_{\text{average}}\right)^n, \qquad n = \left\lceil \log_{1-P_{\text{average}}}^{1-P} \right\rceil. \tag{12}$$

Let $P = P'$, where $1 - P' < \varepsilon$ ($\varepsilon$ is an arbitrarily small positive integer). Then we can obtain an integer $n'$:

$$n' = \left\lceil \log_{1-P_{\text{average}}}^{1-P'} \right\rceil. \tag{13}$$

Thus we can obtain the approximate value of $\theta$ as follows:

$$\theta \approx \frac{2\pi n'}{N}. \tag{14}$$

Based on Equations (9) and (14), the value of $H_{\text{query}}$ can be worked out.

Then $E_{j\text{-query}}^i$ can be worked out using the following equation:

$$\begin{aligned} E_{j\text{-query}}^i = {} & f_q^j \times s_q \times \left(abs\left(id_{\text{ring}}^j - i\right) + H_{\text{query}}\right) \\ & \times (\delta_{\text{send}} + \delta_{\text{receive}}) + \left(H_{\text{query}} + 1\right) \\ & \times f_q^j \times s_q \times \left(\delta_{\text{send}} + nb_{\text{avg}} \times \delta_{\text{receive}}\right), \end{aligned} \tag{15}$$

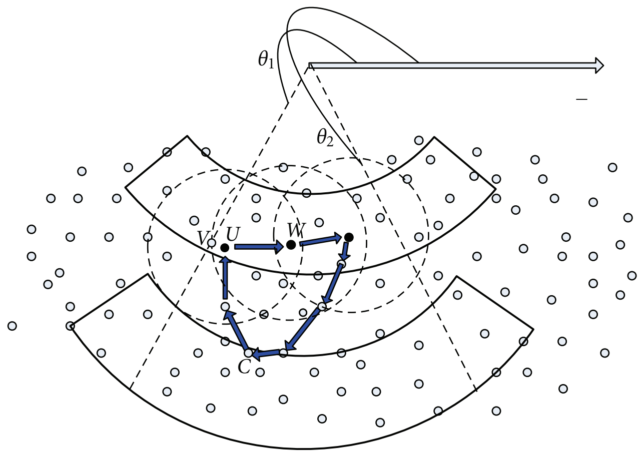


Figure 2: Example of the procedure of data retrieval. $C$ is a consumer. The black dots denote the leading nodes, and other dots denote the ordinary nodes.

where $nb_{avg}$ denotes the average number of one hop neighbors of each node. Suppose that the polar angle of node $j$ is $\theta_j$, and the approximate polar coordinates of the point where the query meets the event data are $(\theta_{mt}, (i + 1/2)R)$. Then we have

$$\theta_{mt} = \begin{cases} \theta_j + \theta, & \left(\theta_j + \theta < 2\pi\right), \\ \theta_j + \theta - 2\pi, & \left(\theta_j + \theta \geq 2\pi\right). \end{cases} \tag{16}$$

Let the Cartesian coordinates of the point where the query meets the event data is approximately $(x_{mt}, y_{mt})$. Then we can make a transformation between the polar coordinates and the Cartesian coordinates:

$$\begin{aligned} x_{mt} &= \left(i + \frac{1}{2}\right) \times R \times \cos(\theta_{mt}) + x_{\text{center}}, \\ y_{mt} &= \left(i + \frac{1}{2}\right) \times R \times \sin(\theta_{mt}) + y_{\text{center}}, \end{aligned} \tag{17}$$

where $(x_{\text{center}}, y_{\text{center}})$ denotes the Cartesian coordinates of the central point of the sensor network field. The approximate hop count $H_{\text{result}}$ between the meetingpoint and node $j$ is as follows:

$$H_{\text{result}} = \frac{\sqrt{\left(x_{mt} - x_j\right)^2 + \left(y_{mt} - y_j\right)^2}}{R}, \tag{18}$$

where $(x_j, y_j)$ denotes the Cartesian coordinates of node $j$.

We can work out $E^i_{j\text{-result}}$ With $H_{\text{result}}$ and some other parameters as follows:

$$E^i_{j\text{-result}} = f^j_q \times s_d \times \alpha \times (H_{\text{result}} + 1) \times (\delta_{\text{send}} + \delta_{\text{receive}}), \tag{19}$$

where $\alpha$ is the data compression ratio. When $E^i_{j\text{-storage}}$, $E^i_{j\text{-query}}$, and $E^i_{j\text{-result}}$ are all worked out, $E^i_j$ in Formula (6) can be worked out.

## 5. Simulation Results and Analysis

We use OMNET++ as our simulator to evaluate the performance of VRS on the application of real-time event data discovery. We compare VRS with ODS [3], which also takes the frequencies of event and query into consideration, mainly on the metrics of network lifetime, load balance, and the delay of data storage and retrieval. In ODS, data are stored and retrieved on the optimal storage node.

*5.1. Simulation Scenario.* The test scenario is a 400 m × 400 m square field, in which 400–900 sensor nodes are deployed randomly. Other parameters of our simulation are listed in Table 1. Each node chooses two numbers from range (0, $f_{\max}$) randomly as its event frequency and query frequency, respectively.

Table 1: The simulation parameter settings.

| Parameter | Value |
|---|---|
| Size of an event data package ($s_d$) | 20 byte |
| Size of a query package ($s_q$) | 10 byte |
| Event data compression ratio ($\alpha$) | 0.5 |
| Maximum packet generating frequency ($f_{\max}$) | 20 packets/100 s |
| Radio transmission range ($R$) | 40 m |
| Energy cost of sending one byte data | 0.0144 mJ/byte |
| Energy cost of receiving one byte data | 0.00864 mJ/byte |

*5.2. Evaluation the Algorithm of Selecting the Rendezvous Ring in VRS.* We evaluate the algorithm of selecting the rendezvous ring in VRS by simulation. The initial energy is set to 1 J, and the node number is configured to 500. We take each of the virtual rings as the rendezvous ring, respectively, and test the value of $EN = E^i_{\text{total}}/n^i$, where $0 \leq i < (L - L\%R)/R$ and the lifetime of the network, respectively.

Figure 3 shows the value of $E^i_{\text{total}}/n^i$ when taking different virtual rings as the rendezvous ring, respectively. From Figure 3, we can see that the values of $EN$ decrease as the ring IDs increase. This is because the nodes are deployed randomly and a virtual ring may contain more nodes as the virtual ring becomes farther to the center of the sensor field. Thus, if the value of $E^i_{\text{total}}$ does not change too much, the values of $E^i_{\text{total}}/n^i$ must decrease as the ring IDs increase.

Figure 4 shows the simulation result of the lifetime of the sensor network when taking different virtual rings as the rendezvous ring. It is clear that the lifetime of the sensor network increases as the ring IDs increase. From Figures 3 and 4, we can see that the smaller the value of $EN$ is, the longer is the lifetime of the sensor network. This reflects that our method on selecting the rendezvous ring is correct to a certain degree.

*5.3. Lifetime Evaluation.* In this section, we show the simulation results of the lifetime of the sensor network in ODS and VRS. In this experiment, the initial energy is set to 1 J, and the number of nodes is varied from 500 to 900.

Figure 5 shows the lifetime of the sensor network in ODS and VRS with different data effective time. We can see that the lifetime of both two schemes decreases as the node number increases in the most cases. As the number of nodes in the sensor network increases, the load of the storage node in ODS and that of the nodes in the rendezvous ring in VRS increase. The node dying first is often the storage node in ODS or a node in the rendezvous ring in VRS. However, we also note that the lifetime of the sensor network in VRS does not change too much as the node number increases when the effective time is set to 0.2 s. The reason is that, when the data effective time is small enough, almost all the queries meet the desired event data in the rendezvous ring. As the number of nodes increases, the probability for a query to meet the desired event data with a small hop count in the rendezvous ring increases, which can decrease the energy consumption on query processing in the rendezvous ring. However, the cost of transmitting the event data to the rendezvous ring increases as the node number increases. Thus the total energy

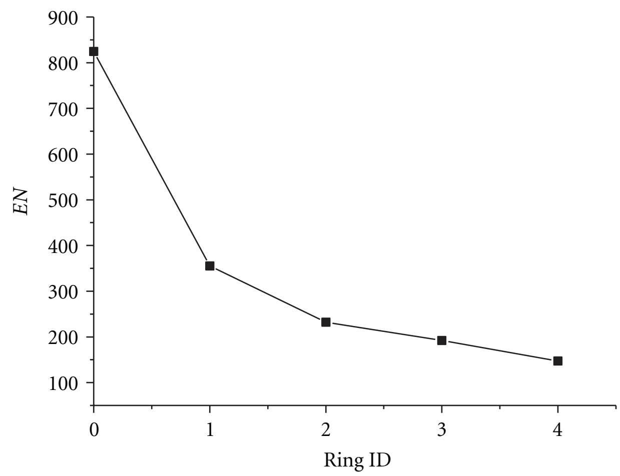


FIGURE 3: The value of $EN$, which is equal to $E^i_{\text{total}}/n^i$, when taking different virtual rings as the rendezvous ring, respectively.

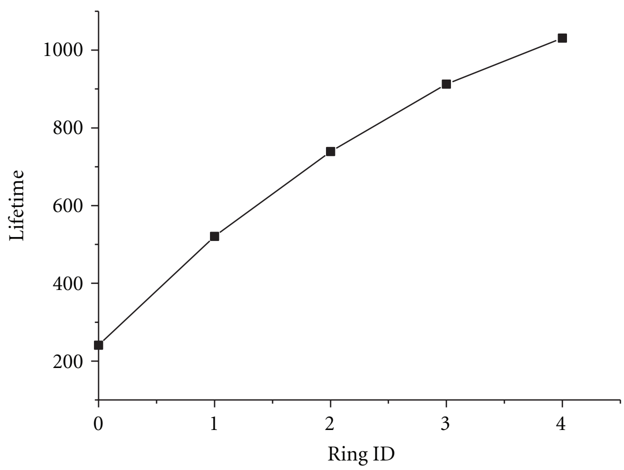


FIGURE 4: The lifetime of the sensor network when taking different virtual rings as the rendezvous ring, respectively.

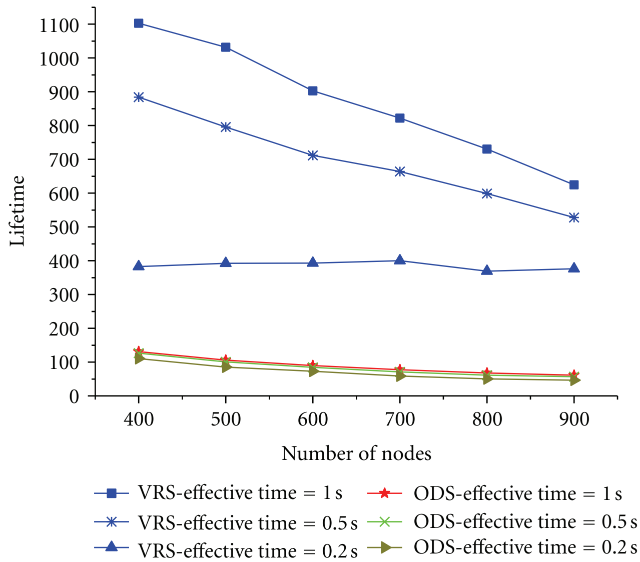


FIGURE 5: Network lifetime in ODS and VRS with different effective time.

consumption of the sensor nodes in the rendezvous ring may not change too much, which also makes the lifetime of the sensor network not change too much in VRS.

We observe that, as the data effective time decreases, the lifetime of the sensor network in both two schemes decreases. As the data effective time decreases, the queries need to go much farther to meet the desired event data, which increases the energy consumption of data retrieval. However, the energy consumption of data storage does not change as the data effective time changes. Thus, as the data effective time decreases, the total energy consumption increases, and the lifetime of the sensor network becomes shorter.

It is clear that the lifetime of the sensor network in VRS is much longer than that in ODS, because VRS can balance the load much better than ODS. Later on, we will show the load distribution of the sensor nodes in the two schemes, respectively.

*5.4. Delay of Data Storage and Retrieval.* In this section, we test the performance of VRS on the average delay of data storage and retrieval. We take the hop count that the data have traveled before they rest on a node in the rendezvous ring as the delay of data storage. The delay of data retrieval is represented as the hop count that the query travels from the consumer to the meeting point, plus the hop count that the query result travels from the meeting point to the consumer. In this test, we set the initial energy to 3 J, and set the simulation time to 100 s.

Figure 6 shows the average delay of data storage in ODS and VRS when the data effective time is set to 0.5 s. In fact, when testing the performance on the delay of data storage, it does not matter what value the data effective time is set to, because the data effective time does not affect the track of the event data at all. We can see that the average delay of data storage in VRS is much less than that in ODS. In VRS, the event data only need to be routed into the rendezvous ring, which is much easier than routing the event data to the storage node using GPSR routing protocol. Besides, there are many nodes located in the rendezvous ring, and they just need to store the data generated by themselves locally, which can greatly decrease the average delay of data storage.

Figure 7 gives the average delay of data retrieval in ODS and VRS with different data effective time. As shown in Figure 7, the average delay of data retrieval in VRS is less than that in ODS. In VRS, there are many nodes located in the rendezvous ring, and they discover the desired event data by broadcasting queries to their neighbors. The probability to meet the desired event data in a short hop count is very high in such a way. Besides, the nodes that are located outside the rendezvous ring are much easier to route queries to the rendezvous ring in VRS than most of the nodes in ODS to route the queries to the rendezvous nodes.

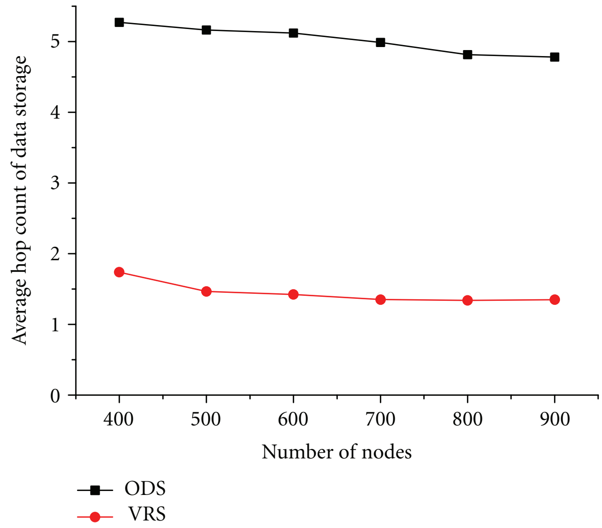


Figure 6: Delay of data storage.

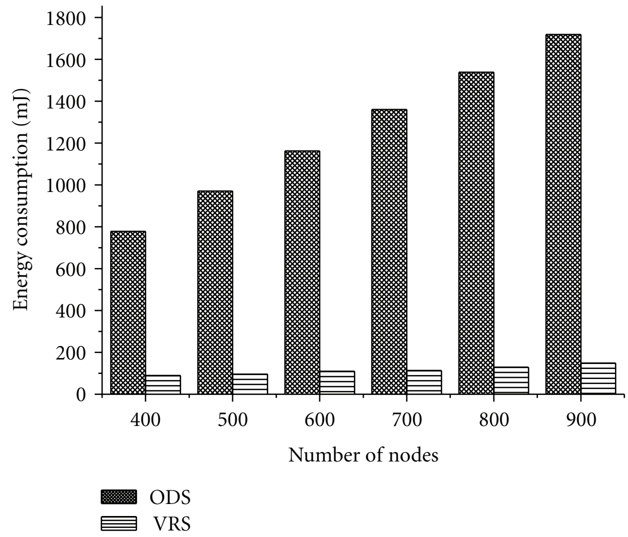


Figure 8: Energy consumption of the node that is the biggest energy consumer.

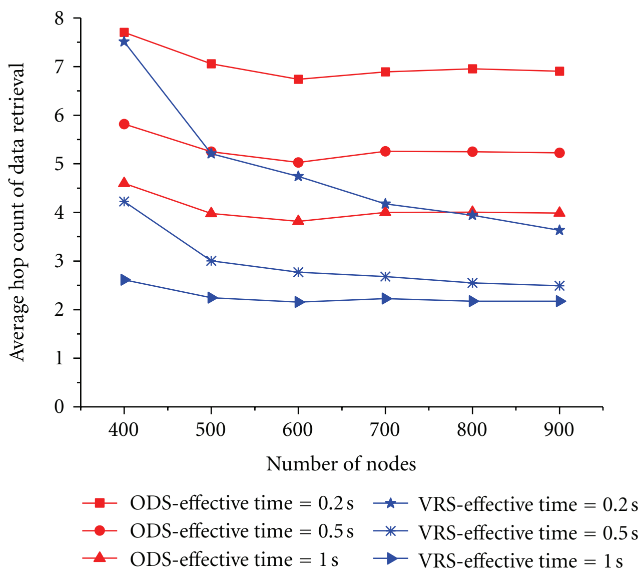


Figure 7: Delay of data retrieval with different effective time.

In Figure 7, as the data effective time decreases, the average delay of data retrieval increases both in ODS and VRS. As the data effective time decreases, the probability for a query to meet the desired data in a short hop count decreases, and the query needs to go much farther to hit the desired event data both in ODS and VRS.

*5.5. Load Balancing Evaluation.* This experiment tests the performance of ODS and VRS on load balance. The initial energy of each sensor node is set to 3 J, the simulation time is set to 100 s, and the data effective time is set to 1 s.

Figure 8 shows the energy consumption of the node that consumes the most energy among all the nodes in the sensor network. The energy consumption by the node that consumes the most energy in ODS is much higher than that in VRS. Theoretically speaking, the node that consumes the most energy must be the storage node in ODS or a node in the rendezvous ring in VRS. Because there is only one storage node in ODS, the load of the storage node in ODS must be much higher than that of any node in the rendezvous ring in VRS. Thus, the energy consumption of the storage node in ODS must be much higher than that of any node in VRS.

Figure 9 shows the distribution of the energy consumption of all the sensor nodes in ODS and VRS, respectively, when the node number is set to 500. It is obvious that VRS can balance the load much better than ODS.

## 6. Conclusion

In this paper, we propose a novel in-network data storage and retrieval scheme, named VRS, in wireless sensor networks. VRS divides the whole sensor network field into virtual rings and selects the rendezvous ring from all the virtual rings to mainly take responsibility of event data storage and query processing. VRS takes the frequencies of event and query into consideration when determining the rendezvous ring, which makes it more efficient. We have done a lot of experiments to evaluate the performance of VRS, and compare it with another existing scheme called ODS. The simulation results show that VRS not only outperforms ODS in terms of load balance and network lifetime but also exhibits comparable latency to ODS.

With regards to future works, we plan to design a distributed algorithm of the rendezvous ring selection and make VRS a totally distributed scheme. Another direction is to combine VRS with the virtual coordinate mechanism [21], which can make VRS a GPS-free scheme.

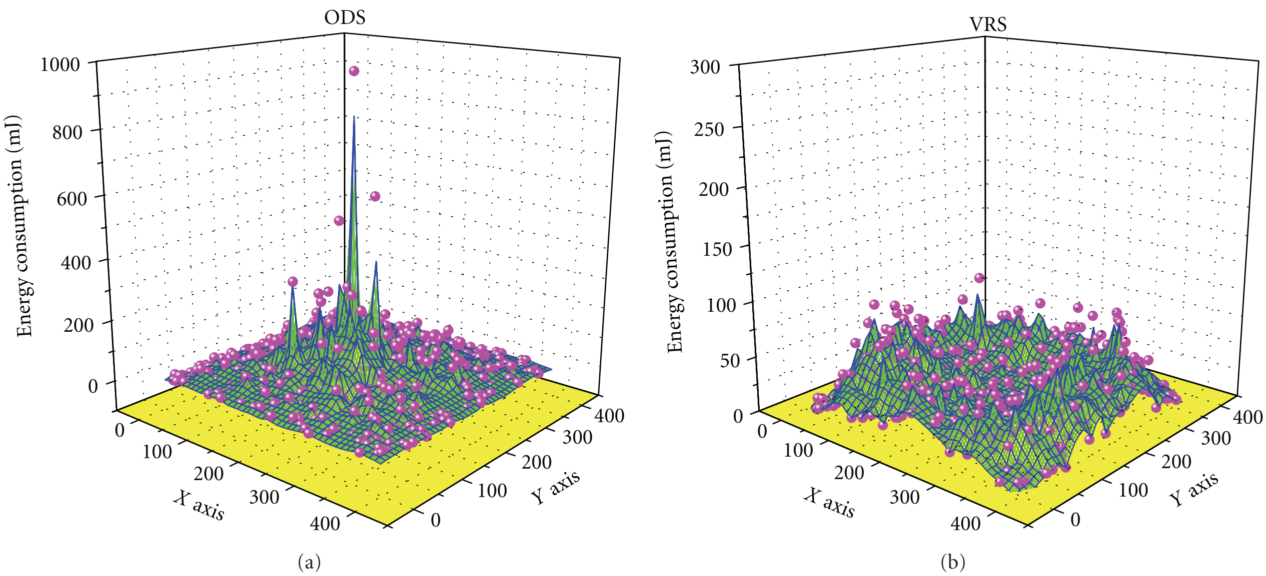


Figure 9: Distribution of energy consumption: (a) ODS and (b) VRS.

## Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grants no. 61173169 and no. 61106036 and the Program for New Century Excellent Talents in University of Ministry of Education under Grant no. NCET-10-0798.

## References

[1] M. Liu, J. N. Cao, G. H. Chen, L. J. Chen, X. M. Wang, and H. G. Gong, "EADEEG: an energy-aware data gathering protocol for wireless sensor networks," *Journal of Software*, vol. 18, no. 5, pp. 1092–1109, 2007.

[2] J. B. Liang, J. X. Wang, T. S. Li, and J. E. Chen, "Maximum lifetime algorithm for precise data gathering based on tree in wireless sensor networks," *Journal of Software*, vol. 21, no. 9, pp. 2289–2303, 2010.

[3] Z. Yu, B. Xiao, and S. Zhou, "Achieving optimal data storage position in wireless sensor networks," *Computer Communications*, vol. 33, no. 1, pp. 92–102, 2010.

[4] D. Braginsky and D. Estrin, "Rumor routing algorithm for sensor networks," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02 )*, pp. 22–31, September 2002.

[5] R. Sarkar, X. Zhu, and J. Gao, "Double rulings for information brokerage in sensor networks," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MobiCom '06)*, pp. 286–297, September 2006.

[6] F. Qing, J. Gao, and J. G. Leonidas, "Landmark-based information storage and retrieval in sensor networks," in *Proceedings of the 25th Conference of the IEEE Communication Society (INFOCOM '06)*, pp. 1–12, April 2006.

[7] C. H. Lin, J. J. Kuo, and M. J. Tsai, "Reliable GPS-free double-ruling-based information brokerage in wireless sensor networks," in *Proceedings of the 29th Conference on Information Communications (INFOCOM '10)*, pp. 1–5, March 2010.

[8] S. Ratnasamy, B. Karp, L. Yin et al., "GHT: a geographic hash table for data-centric storage," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, pp. 78–87, September 2002.

[9] K. T. Ma, K. Y. Cheng, K. S. Lui, and V. Tam, "Improving data centric storage with diffuse caching in wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 9, no. 3, pp. 347–356, 2009.

[10] Y. Lai, H. Chen, and Y. Wang, "Dynamic balanced storage in wireless sensor networks," in *Proceedings of the 4th Workshop on Data Management for Sensor Networks (DMSN '07)*, pp. 7–12, September 2007.

[11] R. Biswas, K. Chowdhury, and D. P. Agrawal, "Attribute allocation and retrieval scheme for large-scale sensor networks," *International Journal of Wireless Information Networks*, vol. 13, no. 4, pp. 303–315, 2006.

[12] W. H. Liao and W. C. Wu, "Effective hotspot storage management schemes in wireless sensor networks," *Computer Communications*, vol. 31, no. 10, pp. 2131–2141, 2008.

[13] H. C. Le, H. Guyennet, and N. Zerhouni, "Mobile effect reduction in data-centric storage for wireless sensor networks," in *Proceedings of the 3rd IET International Conference on Intelligent Environments (IE '07)*, pp. 304–311, September 2007.

[14] G. L. Li and H. Gao, "A load balance data storage method based on ring for sensor networks," *Journal of Software*, vol. 18, no. 5, pp. 1173–1185, 2007.

[15] L. Xie, L. J. Chen, D. X. Chen, and L. Xie, "Ring-based multiresolution data storage for sensor networks," *Journal of Software*, vol. 20, no. 12, pp. 3163–3178, 2009 (Chinese).

[16] W. S. Zhang, G. H. Cao, and T. La Porta, "Data dissemination with ring-based index for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 7, pp. 832–847, 2007.

[17] E. D. Kaplan, *Understanding GPS: Principles and Applications*, Artech House, Boston, Mass, USA, 1996.

[18] B. Li, Y. He, F. Guo, and L. Zuo, "A novel localization algorithm based on isomap and partial least squares for wireless sensor

networks," *IEEE Transactions on Instrumentation and Measurement*, no. 99, pp. 1–11, 2012.

[19] N. Zhang, X. Cheng, M. Song, and D. Chen, "Time-bounded essential localization for wireless sensor networks," *IEEE/ACM Transactions on Networking*, no. 99, pp. 1–13, 2012.

[20] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom '00)*, pp. 243–254, August 2000.

[21] M. J. Tsai, H. Y. Yang, and W. Q. Huang, "Axis-based virtual coordinate assignment protocol and delivery-guaranteed routing protocol in wireless sensor networks," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, pp. 2234–2242, May 2007.