Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2007, Article ID 48984, 10 pages
doi:10.1155/2007/48984

Research Article

An Energy-Efficient Framework for Multirate Query in Wireless Sensor Networks

Yingwen Chen,[1] Ming Xu,[1] Huai-min Wang,[1] Hong Va Leong,[2] Jiannong Cao,[2] Keith C. C. Chan,[2] and Alvin T. S. Chan[2]

[1] *School of Computer, National University of Defense Technology, Changsha 410073, Hunan, China*
[2] *Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong*

Received 30 September 2006; Revised 14 March 2007; Accepted 6 April 2007

Recommended by Mischa Dohler

Minimizing the communication overhead is always a hot topic in wireless sensor networks. In a multirate query system, data sources disseminate the data streams to users at the frequency they request. However, sending data in different frequencies to individual users is very costly. We address this problem by broadcasting a single consolidated data stream, aiming at reducing the amount of transmitted data. Taking into account the data correlation, we can reconstruct the data streams at lower frequencies from the consolidated stream at a higher frequency. In this paper, we propose an energy-efficient framework to process multirate queries and investigate the path-sharing routing tree construction method together with the rate conversion mechanism. We evaluate both the accuracy and energy efficiency by simulation. Simulation results indicate that with a reasonable level of tolerance, the performance gain is significant. As far as we know, this is the first energy-efficient solution for multirate query in wireless sensor networks.

1. INTRODUCTION

A wireless sensor network consists of a collection of communicating nodes, each incorporated with sensors collecting real-time data to the sink node. Sensor nodes are battery-powered and energy is the most crucial resource. Many existing research works address the problem of minimizing energy consumption by minimizing the communication overhead, such as adopting data aggregation to reduce data transmission, using data replicas to shorten the data delivery path.

In a multirate query system, a data source serving multiple sink nodes with queries demanding varying data rates needs to send data in different frequencies to individual nodes. This is costly, since the sink nodes in general consume data at different moments and most of the data sent by the data source could not be shared across the sink nodes. This new problem is different from the one addressed in data aggregation and data replication. Observing the correlation among data streams from the same data source to different sinks, it is possible to construct a consolidated stream to represent those multiple data streams. We address this interesting problem by broadcasting the single consolidated streaming data series, aiming at reducing the amount of transmitted data, and hence energy consumption.

The contribution of the paper is threefold. First, we describe the multirate query problem in WSNs. Second, we propose an energy-efficient framework to process multirate queries and investigate rate conversion mechanism between arbitrary frequencies. Third, we analyze analytically the performance on communication cost with our energy-efficient strategy and conduct simulation studies to evaluate the energy efficiency and accuracy of our strategy. Our simulation results indicate that we can achieve an average saving of up to 50% $\sim$ 55% of communication cost, at an average relative error below 5%.

The rest of this paper is organized as follows. Section 2 presents some of the research work related to ours. Section 3 introduces the multirate query problem. In Section 4, we propose our energy-efficient framework including the query frequency registration, path-sharing routing tree construction, data stream dissemination, and data stream frequency conversion. Section 5 presents both analytical and simulation results on the query strategies. Finally, we conclude the paper briefly.

2. RELATED WORK

Because of the energy constraint of wireless sensor networks and relatively expensive communication cost, two types of methods have been proposed to reduce the transmitted data: one is in-network data processing and data aggregation, the other is data replication. This section briefly reviews these methods and provides the motivation for our work.

2.1. In-network data processing and data aggregation

Measurements suggest that sending one bit is equivalent to executing approximately 1000 CPU instructions [1]. Thus, part of the computation can be off-loaded from the sink node and performed inside the network, such as eliminating irrelevant records and aggregating raw data, which is referred to as in-network data processing and data aggregation. Since the placement of the data processing function and operators dominate the energy consumption of in-network data processing, literature [2–4] discussed operator placement strategies for hierarchical and nonhierarchical cases. Literature [5] proved that finding the optimal routing tree to support data aggregation can be shown to be equivalent to finding the minimum Steiner tree, an NP-hard problem. Greedy Incremental tree was employed to improve path sharing so as to reduce transmission energy. Considering the data correlations of different source nodes, literature [6] proposed some efficient, scalable, and distributed heuristic approximation algorithms for solving the new NP-hard problem.

All these in-network data processing and data aggregation research works only deal with the case that there is only one sink node. However, in a real system there might be multiple users. This is the reason we take multiple sink nodes into consideration.

2.2. Data replication

In distributed environments that collect or monitor data, useful data might be spread to multiple users. One of the most useful ways to reduce data transmission is to maintain copies of data objects of interest using replication, which can help to reduce the average length of the routing path. Literature [7] discussed data dissemination in a scenario of multiple mobile sink nodes. In order to feed the sink nodes with minimal energy consumption, a *GateReplicaSearch* algorithm together with a *ReplicaPlacement* algorithm are proposed. Literature [8] considered the problem of optimizing the number of replicas for event information in wireless sensor networks, when queries are disseminated using expanding rings. The authors also derived the replication strategies that minimize the expected total energy cost consisting of search and replication costs.

Current data replication deals with the case that the queries issued by multiple sink nodes are the same. However, if multiple sink nodes issue the queries with different frequencies, how can they share the bandwidth, leading to savings of the transmission energy consumption? This is the main purpose of our work.

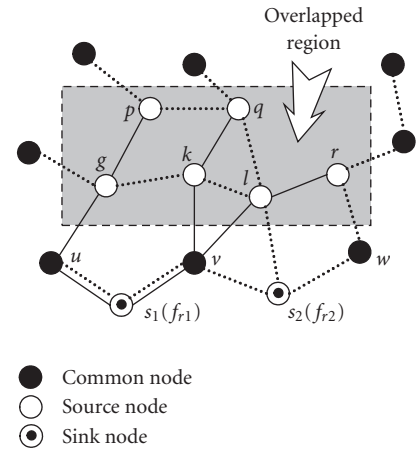

Figure 1: Multirate query example in WSN.

3. MULTIRATE QUERY IN WSNS

In WSNs, the sink nodes may query the data at different frequencies according to different requirements. Thus, a simplest two-rate querying system can be illustrated in Figure 1. Sink node s_1 requests the data from all the nodes in the grey region at the frequency of f_{r1}. At the same time, sink node s_2 requests the data from all the nodes in the grey region at the frequency of f_{r2}. Without loss of generality, we can always find an appropriate time unit such that all frequencies can be represented as integers unless the frequencies are irrational numbers.

Example 1. If the WSN is used for collecting the temperature of the environment, sink node s_1 might need the newest temperature every 2 minutes, and sink node s_2 might need the newest temperature every 3 minutes, supposing these two queries are issued at time 0, this will result in multirate queries in WSN, for which there are two queries, demanding data at times 2, 3, 4, 6, 8, 9, 10, 12, and so on. Selecting the time unit as 6 minutes, we have $f_{r1} = 3$, $f_{r2} = 2$.

Generally, the sink node initiates the data query by sending out a query request to the data sources. The transmission of the query request may naively be flooding or it may follow some logic that the intermediate sensor nodes apply [9]. Finally, when the query request is routed to proper source nodes (i.e., sensors within the queried regions or satisfying some query conditions), the source nodes will start sending data back to the sink node along the corresponding routing tree.

When there are multiple sink nodes, the foregoing process repeats until all the queries have been satisfied. As a result, the whole sensor network will construct multiple routing trees rooted at multiple sink nodes. However, when some sink nodes share some of the source nodes, every overlapped source node belongs to multiple routing trees rooted at different sink nodes.

Example 2. In Figure 1, all the source nodes in the overlapped region are covered by the routing tree rooted at sink node s_1 (solid line) and the routing tree rooted at sink node s_2 (dashed line). Therefore, reducing the total communication cost of the multirate query system asymptotically equals reducing the redundant data forwarding among intermediate nodes from each overlapped source node to all known sink nodes. For this reason, in the following part, we will describe in details how to minimize the transmission cost for an individual source node to report the data periodically to multiple sink nodes according to the path overlapping.

Suppose a multirate querying system in which there are m sink nodes s_i $(i = 1 \cdots m)$ requesting the streaming data series from the same source node d at different frequencies f_{ri} $(i = 1 \cdots m)$. Intuitively, the source node d disseminates the data along the routing trees to each sink node at the corresponding frequency separately. We call this kind of data dissemination strategy the *native strategy* (or *N-strategy*). Theorems 1 to 3 present some properties of the N-strategy. The proofs of these theorems are listed in the appendix.

Theorem 1. *Using N-strategy, the upper bound of the consolidated data dissemination frequency f_{up} of source node is $\sum_{i=1}^{m} f_{ri}$, where f_{ri} $(i = 1 \cdots m)$ are the requested frequencies of all the sink nodes. This upper bound is attained if and only if for any pair of data series in the request, there is no point of intersection along their time axes.*

Example 3. If all the two queries in Example 1 are issued at times 0, 0.5 separately, that is, the data are demanded at times 2, 3.5, 4, 6, 6.5, 8, 9.5, 10, 12, 12.5, and so on, as a result, the upper bound of the consolidated data dissemination frequency f_{up} is achieved as $2 + 3 = 5$.

Theorem 2. *Using N-strategy, the lower bound of the consolidated data dissemination frequency f_{low} of the source node can be calculated by*

$$\sum_{k=1}^{m} \left((-1)^{k-1} \cdot \sum_{\{F_j\}_{j=1}^{k} \subseteq \{f_{ri}\}_{i=1}^{m}} gcd(\{F_j\}_{j=1}^{k}) \right), \tag{1}$$

where $\{F_j\}_{j=1}^{k}$ means the set of all the combinations of k frequencies selected in all m frequencies. This holds if and only if for any pair of data series in the request, they have points of intersection along their time axes. Example 1 satisfies the lower bound condition, as a result $f_{low} = 2 + 3 - gcd(2, 3) = 4$.

Theorem 3. *Given m frequencies $f_{r1} \le f_{r2} \le \cdots \le f_{rm}$, the lower bound of the consolidated data dissemination frequency f_{low} of source node in N-strategy satisfies $f_{low} \ge \max\{f_{ri}\}_{i=1}^{m}$. The equation is achieved if and only if for all $j \ge i$, $f_{ri} \mid f_{ri}$, $1 \le i \le j \le m$, notation "$a \mid b$" means that b is exactly divided by a.*

Example 4. suppose three queries, by which sink node 1 needs the newest temperature every 8 minutes, and sink node 2 needs the newest temperature every 4 minutes, and sink node 3 needs the newest temperature every 2 minutes. All these three queries are issued at time 0, and data are demanded at times 2, 4, 6, 8, 10, 12, 14, 16, and so on. Selecting the time unit as 8 minutes, we have $f_{r1} = 1$, $f_{r2} = 2$, and $f_{r3} = 4$. Because $f_{r1} \mid f_{r2}$, and $f_{r2} \mid f_{r3}$, we have $f_{\text{low}} = \max(f_{r1}, f_{r2}, f_{r3}) = 4$.

From Theorems 1 to 3, we can conclude that N-strategy can reduce the consolidated data dissemination frequency when the requested data series have points of intersection along their time axes, and when the requested frequencies are mutually multiple and submultiple. But in a real application, it is hard to fulfill this kind of requirement. We need an enhanced strategy to reduce the consolidated data dissemination frequency, so as to reduce the summation of the energy consumption.

From the basic rule of information theory, the total amount of information is proportional to the number of samples and the number of bits coding the sample [10] . Under the same coding system, a data series at higher frequency (with smaller intervals) contains more information than the one at lower frequency. Taking advantage of the data correlation between data series at different frequencies, data series at lower frequency could be constructed from data series at higher frequency. It is obvious that N-strategy is inefficient because the source node propagates the data series regardless of the data correlation between them. Since wireless communication in WSNs is of a broadcast nature, transmitting data at a consolidated frequency can potentially cut down the total amount of transmitted data, leading to savings in energy consumption. Taking Figure 1 as an example, if data series at frequency f_{r2} can be reconstructed from data series at frequency f_{r1} within acceptable error, source node l only needs to disseminate the data to s_1 at frequency f_{r1}. When node 1 forwards the data to v at frequency f_{r1}, node s_2 can also receive the data at frequency f_{r1}. Node s_2 can then reconstruct the data series at frequency f_{r2} from the received data series. As a result, the transmission overhead of source node l is reduced by avoiding sending the data series individually to s_1 and s_2. Likewise, in a multirate query system, the total amount of data transmitted across intermediate nodes can also be reduced. We call our strategy the *E-strategy* in contrast to the intuitive *N-strategy*. In E-strategy, if data streams with different frequencies share the same path, only the data stream with the highest frequency needs to be transmitted, and other data streams can be reconstructed from it. This leads to reduction of the transmission energy consumption.

There are three problems that need to be addressed when considering data correlation between data series at different frequencies in a multirate query system. The first one is how to find new routing paths to all the sink nodes in order to take the full advantage of bandwidth sharing. The second one is how to organize the sensor node activity to generate a consolidated data stream, with the aim of reducing the amount of transmitted data, hence bandwidth requirement and energy consumption. The last one is how to reconstruct the data streams at the desired frequency from the consolidated stream at a different frequency. We will present the solutions in the subsequent sections.

4. ENERGY-EFFICIENT FRAMEWORK

Our energy-efficient framework for multirate query in WSNs is built upon a number of components, including query frequency registration, path-sharing routing tree construction, data stream consolidated dissemination, and data stream frequency conversion. Query frequency registration allows data sinks to pose their querying requirement to the data source. With the historical path information of the query requests from sink nodes to source node, the source node can construct a path-sharing routing tree, which shares the bandwidth for data transmission. From the query frequencies registered along the route, every intermediate node determines the frequency on which the data stream should be generated and then disseminated. By adopting the data dissemination process, the data streams are transmitted to their designated destination. Staying in the core is the frequency conversion mechanism, which allows data streams to be converted from one frequency to another. In the midst of data dissemination, forwarding nodes may need to perform frequency conversion when necessarily in order to make use of the path-sharing property.

4.1. Query frequency registration

N-strategy is inefficient because it does not take advantage of the data correlation between data series, even though the data series are transmitted along the same path. In order to make use of the data correlation between data series, we need the information about the query frequencies on the intermediate node along the path from the source node to the sink nodes. We maintain a list, called *RequestList*, on every node in the network. The list contains the frequencies of all requests passing through that particular node.

When the sink node generates a query at a certain frequency, as it is explained in Section 3, it adopts the directed diffusion routing algorithm [9] to deliver the query request to the corresponding source nodes. The details about the process can be described as follows. (1) The sink broadcasts a query request for the source to its neighbors. (2) After receiving the request message for the first time, a node n adds the frequency of the request in the *RequestList* and decides whether to forward the message. If the message comes from its only neighbor, it would not forward the message; otherwise, it broadcasts the message to other neighbors. If it is not the first time for n to receive the request message, n will refrain from doing anything. This process is repeated until the query request finally reaches all the source nodes.

In the query frequency registration process, every node in the network forwards the query request at most once. Supposing each bypassing node is added in the payload of the query request, every node can learn the path from the sink to itself. Assuming that the time to transmit packets between neighboring nodes is approximately the same, the query frequency registration process becomes similar to a breadth-first search, and the paths from each sink node to every sensor node would be those with minimal number of hops. Since every sink node delivers the query request by adopting directed diffusion routing algorithm, all sensor nodes can buffer the minimal-hop path to each sink node in a short time interval. We will explain the details about how to construct the routing tree with maximal path sharing in the following part.

4.2. Path-sharing routing tree construction

The basic idea of our E-strategy is to make full use of the potential bandwidth sharing of all the routes from an individual source to multiple sinks. As a result, maximizing the path-sharing property leads to lowest energy consumption by adopting the E-strategy. On the other hand, maximizing the path sharing equals to finding the *minimal Steiner tree* problem, which can be defined as follows.

Given an undirected graph $G = \langle V, E\rangle$ and a node set, $U \subseteq V$ a *minimal Steiner tree* for U in G is a minimum-size subset $T \subseteq E$ with the least number of edges such that $\langle V(T), T\rangle$ contains a path from s to t for all $s, t \in U$, where $V(T)$ denotes the set of nodes incident to an edge in T.

Since the *minimal Steiner tree* problem is known to be NP-hard, we propose a heuristic method to get an approximation, in which all the sink nodes are incrementally connected to the routing tree by minimal-hop path. In order to shorten the path for disseminating the data stream with larger frequency, the sink node with larger query frequency has higher priority to be added to the existing routing tree. Since there is no global information, we need a decentralized greedy process to implement this kind of heuristic method.

The source node orders all the sink nodes by their request data frequencies descendingly. In Section 4.1, we explain that each node has buffered the minimal-hop paths from all the sink nodes to itself. So the source node can select the shortest path to the first sink node as the original routing tree T_1. In order to connect the ith $(i > 1)$ sink node to the existing routing tree T_{i-1} by minimal-hop path, the source node needs to send an $(i-1)$th *explorer message* along the existing routing tree to find the joint u, which has shorter minimal-hop path to the ith $(i > 1)$ sink node than its neighbors. This process is similar as the decentralized neighbor exploration strategy discussed in [3], in which the cost is defined as the hop count to the sink node. Note that in the neighbor exploration strategy, the *explorer message* is always unicast to the neighbor node that has the minimal hop count to the sink node. Therefore, the forwarding times of each *explorer message* are no greater than the diameter of the WSNs. In another word, the transmission consumption of each *explorer message* is small and tolerable.

For node u, if its minimal-hop path to the ith sink node is noted as $P(u, s_i)$,[1] we have $T_i = T_{i-1} \cup P(u, s)$. Because the $(i-1)$th *explorer message* must be sent along the tree T_{i-1}, we should insert a time slot ΔT between any two explorer messages. In fact, all *explorer messages* are initially sent by the source node. The $(i-1)$th *explorer message* is always in front of the ith one. So the time slot ΔT is no need to be very large.

[1] Because there is no global information, $P(u, si)$ is still a local minimum.

In this manner, we can reduce the latency induced by the localized and decentralized greedy processes, which is just like a pipelining.

4.3. Data stream consolidation and dissemination

Since all the frequencies of the requested queries are registered in *RequestList* of each intermediate node along the routing path, it is easy for the intermediate node to determine whether there is bandwidth sharing. In fact, bandwidth sharing happens in those nodes with *RequestList* containing at least two frequencies. As a result, each node can cut down the communication cost by choosing the largest frequency from *RequestList* as the frequency of its consolidated data stream.

Algorithm 1 describes the algorithm for data consolidation and dissemination. We can see that the source node simply broadcasts the data at the *largest frequency* of all the queries. However, for other nodes, there may be the case that the frequency of the data series received, *ReceivedF*, is larger than the largest frequency in *RequestList*, *RequestF*, meaning that the incoming data is more than enough. The frequency conversion function is invoked to reconstruct the data series at frequency *RequestF* from the data series at frequency *ReceivedF*. The frequency conversion mechanism is discussed next.

4.4. Frequency conversion

Frequency conversion is concerned with the problem that given a data series X at frequency f_1, how to determine the value of an unknown data series Y at frequency f_2? The frequency conversion problem is similar in nature with the interpolation problem, which is constructing new data points from a discrete set of known data points.

We adopt interpolation techniques to achieve simple frequency conversion. There are many interpolation algorithms such as linear interpolation, quadratic interpolation, cubic-spline interpolation. We choose linear interpolation based on two reasons: first, it is the simplest interpolation method, with the least computation overhead and the smallest window size; second, our preliminary simulation results show that its accuracy is acceptable, and that the advantage of a few other interpolation mechanisms is not very significant.

In linear interpolation, the values interpolated between two consecutive data samples lie on a straight line connecting them and we can estimate the values $\hat{Y}$ of data series Y by

$$\hat{y}[i] = (x[\lfloor z_i \rfloor + 1] - x[\lfloor z_i \rfloor]) \cdot (z_i - \lfloor z_i \rfloor) + x[\lfloor z_i \rfloor], \quad (2)$$

where $z_i = (i \cdot f_1)/f_2$, and $\lfloor z \rfloor$ is the floor function, returning the largest integer no larger than z.

If we know the true value of Y, we can use the average relative error (ARE) metric to evaluate the accuracy of interpolation. For a series of length len, ARE is defined as

$$\mathrm{ARE}(Y, \hat{Y}) = \left(\sum_{i=0}^{\mathrm{len}} \frac{|\, y[i] - \hat{y}[i] \,|}{y[i]} \right) \Big/ (\mathrm{len} + 1). \quad (3)$$

4.5. Pragmatic consideration

From (2), we can observe that if we want to get the ith value of $\hat{Y}$, we need the $\lfloor z_i \rfloor$th and $(\lfloor z_i \rfloor + 1)$th values of X.

Since $\lfloor z_i \rfloor \cdot 1/f_1 \le i/f_2 < (\lfloor z_i \rfloor + 1) \cdot 1/f_1$, we need *future value* of X to estimate the current value of Y. This is only possible in a historical system, but not in a real-time system like most sensor network applications. Fortunately, we can still attempt to predict the required future value of X from the historical information of data series X. In particular, we employ the following prediction method for a future value of X:

$$x[\lfloor z_i \rfloor + 1] = \alpha \cdot x[\lfloor z_i \rfloor] + (1 - \alpha) \cdot x[\lfloor z_i \rfloor - 1]. \quad (4)$$

Using the frequency conversion mechanism, we can convert the data series between arbitrary frequencies. However, converting data series at lower frequency to higher frequency brings in a relatively large ARE than the more natural downsampling operation. That is the reason why we choose the largest frequency to be the frequency of the consolidated broadcasting stream in E-strategy, in order to reduce the ARE when the intermediate and sink nodes reconstruct the data series at lower frequency.

5. PERFORMANCE ANALYSIS

We first give the analytical bound on the energy consumption of N-strategy and E-strategy, and then conduct the simulation studies to make further evaluations. The greatest performance gain from E-strategy is due to the ability of sharing the bandwidth as much as possible along the path when disseminating the data series, thereby reducing the energy consumed.

5.1. Analytical result

Theorem 4. *In the case that all the nodes except the source node in the WSNs query the same data source. The upper bound of the total communication overhead in one time unit for N-strategy is $O(D \cdot (N-1))$, while that of E-strategy is $O(N-1)$, where D is the diameter of the sensor network and N is the number of sensor nodes.*

Proof. By applying Theorem 1, in N-strategy, the upper bound of the total communication cost is $\sum_{i=1}^{N-1} f_i d_i$, where d_i is the number of hops from the sink nodes to source node. Since $d_i \le D$, the expression can be simplified as

$$\sum_{i=1}^{N-1} f_i d_i \le f_{\max} \cdot \sum_{i=1}^{N-1} d_i \le f_{\max} \cdot D \cdot (N-1) \cong O(D \cdot (N-1)). \quad (5)$$

In E-strategy, because all the query results can be constructed from the data series with the largest frequency, the upper bound of the total communication cost is materialized when all the nodes forward the data series at $f_{\max}$ to the farthest sink nodes and it can be calculated by $f_{\max} \cdot (N-1)$, which is $O(N-1)$. □

```
DataDissemination(MyID)
begin
    RequestF ⟵ FindMax (RequestList);
    if (MyID = SourceID) then
        broadcast (Data, RequestF); // broadcast at the requested frequency
    else
        receive(Data);
        ReceivedF ⟵ GetFrequency (Data);
        if (RequestF < ReceivedF) then
            convertFrequency (Data, ReceivedF, RequestF); // do downsampling
            SendF ⟵ RequestF;
        else SendF ⟵ ReceivedF;
        if (myID = SinkID) then
            toApplication (Data);
        else broadcast (Data, SendF);
    end if;
```

Algorithm 1: Data consolidation and dissemination.

Table 1: Parameters of query and sensor network.

Parameter	Symbol	Default value
Coverage of sensor network	δ	300 by 300
Number of sensor nodes	N	420
Transmission range	ρ	30
Number of sink nodes	m	6
Frequency of the query	f	1–20
Query distance	H	6 hops

It is obvious that E-strategy always outperforms N-strategy in terms of communication cost. If the multirate queries in the network share more paths, there is a greater savings in communication overhead using E-strategy. Theorem 4 specifies an extreme case that E-strategy can take full advantage of path sharing, yielding a theoretically perfect performance over N-strategy.

5.2. Simulation studies

In this section, we present the results of our simulation studies. We evaluated the communication cost and accuracy of E-strategy and made a comparison with N-strategy. We also investigated the effects of the sensor network and query parameters on the performance of E-strategy.

In our simulation, the sensor nodes are distributed in a region δ, according to the uniform distribution. A communication graph is generated under the assumption that all the nodes have the same transmission range ρ. A summary of the query and sensor network parameters and their default values is presented in Table 1.

In order to ensure that the simulation experiments are repeatable, we use synthetic data. We generate the data source time series with a function of the random-walk series, defined as [11]

$$x[i] = 100 * \left(\sin\left(0.1 * \text{RandomWalk}[i]\right) + 1 + \frac{i}{R} \right), \tag{6}$$

where $i = 0, \ldots, R-1$; $RandomWalk\ [0 \cdots R-1]$ is a random-walk series; and R is the range of the walk, with a value of 100 000. The time unit is chosen as the least common multiplier of all frequencies of the queries launched by the sink nodes, so as to keep the time intervals of all sampled data series integers.

The sink nodes and source node are chosen randomly. Each sink node launches a query to the same source node with an integer frequency. We use both direct diffusion [9] routing protocol to find the shortest-path routing tree (SPT) and our heuristic method to find the path-sharing routing tree (PST) for data dissemination. The communication cost is evaluated by the number of data packets sent per time unit including the packets amount for constructing the routing tree, and the accuracy is evaluated by the mean of the ARE of all sink nodes.

We generate 100 connected network instances for each simulation and spawn multirate queries in each network instance for 100 times. The average performance for the queries in each network topology is measured and the overall performance is obtained as an average over all the 100 topologies. The confidence level is chosen as 95%.

5.2.1. Impact of query distance

The first set of simulated experiments aims at evaluating the communication cost and accuracy with a different query distance H. The query distance reflects how far it is from the sink node to the source node. It is the number of hops between the sink node and the source node. In this experiment, we fix the number of sensors N to 420. The results are depicted in Figures 2 and 3.

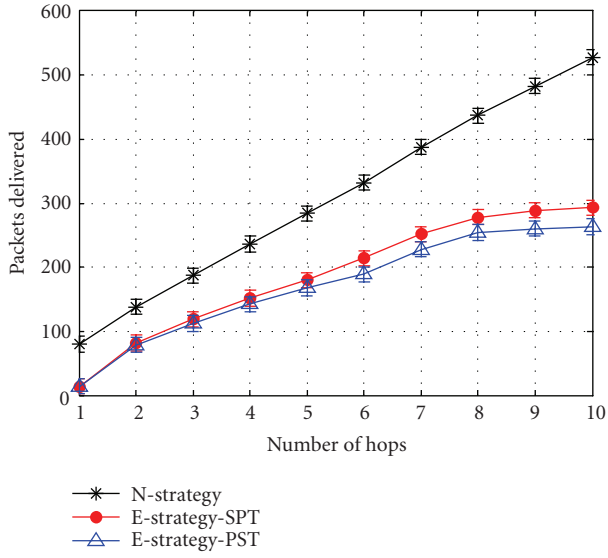

Figure 2: Cost versus query distance.

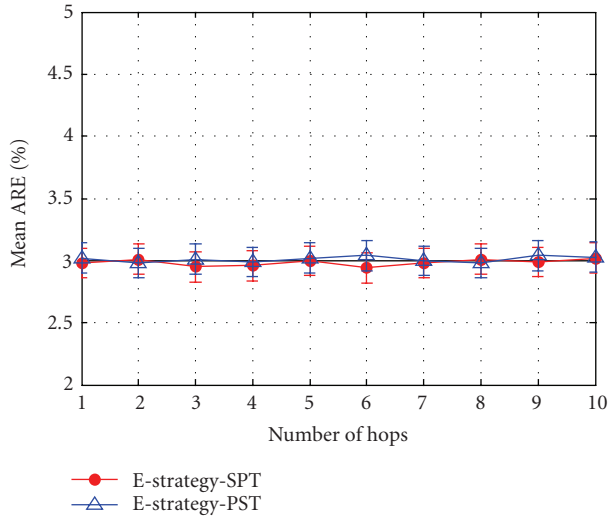

Figure 3: Accuracy versus query distance.

From Figure 2, it is obvious that we can benefit a lot in communication cost by adopting E-strategy, especially by using the path-sharing routing tree. As the query distance H increases, the cost of N-strategy grows almost linearly with H, faster than that of E-strategy. That is because the cost of N-strategy reflects the cumulative overhead of all queries, while the cost of E-strategy is only a part of that, owing to its bandwidth sharing property. E-strategy with PST outperforms E-strategy with SPT, because the bandwidth is only shared by chance in the latter one. When the average hop of the query distance is getting to 10, E-strategy with PST leads to a saving of about 50% of communication cost over N-strategy.

Figure 3 indicates the tradeoff in accuracy. We can see that using the linear interpolation to convert the frequency

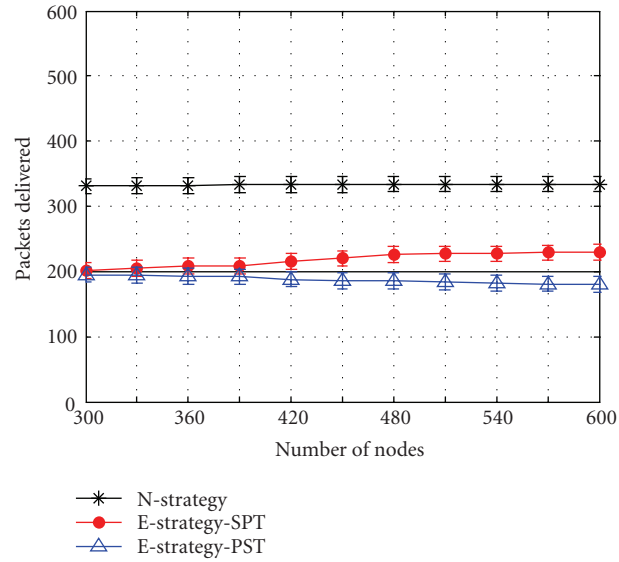

Figure 4: Cost versus node density.

generates a very tolerable mean ARE, which is only about 3% of the actual sensor data value. Furthermore, this imprecision is relatively independent of the query distance.

5.2.2. *Impact of node density*

Since the topology of the sensor network is affected greatly by the node density, we investigate how the node density will affect the performance of the query strategies. In this experiment, we fix the number of hops of the query H to 6 and vary the number of nodes N, and hence node density. The results are depicted in Figures 4 and 5.

From Figure 4, it is obvious that E-strategy outperforms N-strategy in terms of communication cost. Both the communication costs of N-strategy and E-strategy with PST decrease slightly as the node density increases. This is because when there are more sensor nodes, each node may have more neighbors, which help to further shorten the shortest paths from the sink nodes to the source node, leading to reduction of the communication cost. However, we can see that the communication cost of E-strategy with SPT increases slightly as the node density increases. That is because even though more neighbors of each node might shorten the shortest paths from the sink nodes to the source node, they also reduce the chance for different sink nodes to share the same path. This phenomenon shows that the path-sharing property is more important than the short-path property according to the E-strategy.

When accuracy is concerned, Figure 5 indicates that the mean ARE is again maintained at a comfortable level of about 3%, and is relatively independent of node density.

5.2.3. *Impact of number of sink nodes*

The communication cost is closely related to the number of sink nodes, and hence the number of queries. Thus, we

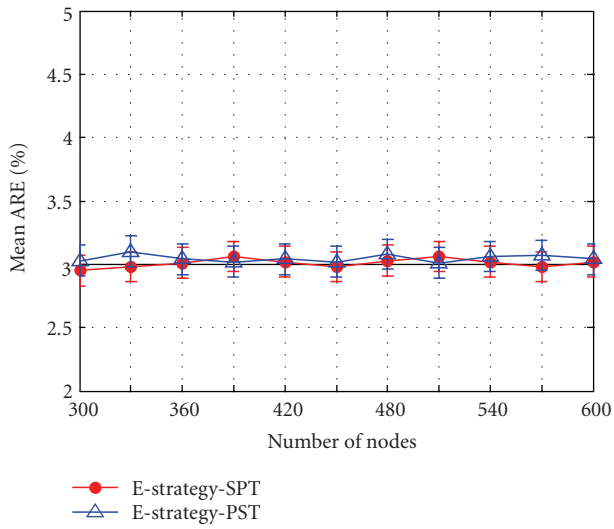

Figure 5: Accuracy versus node density.

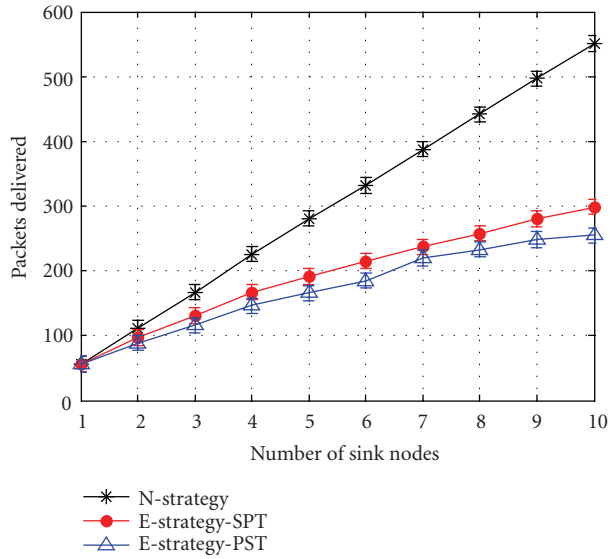

Figure 6: Cost versus number of sink nodes.

measure the performance of N-strategy and E-strategy with respect to number of sink nodes. In this set of experiments, we fix the number of sensors N to 420 and the query distance H to 6, and we vary the number of sink nodes from 1 to 10. The results are depicted in Figures 6 and 7.

From Figure 6, it is obvious that we can again benefit a lot in communication cost by adopting E-strategy. As the number of sink nodes m increases, the cost of N-strategy increases almost linearly and much faster than E-strategy. E-strategy with SPT increases faster than E-strategy with PST. That is because more sink nodes intuitively arouse more queries, hence higher communication overhead. By applying E-strategy with PST, the communication overhead can be greatly reduced via bandwidth sharing. When the number

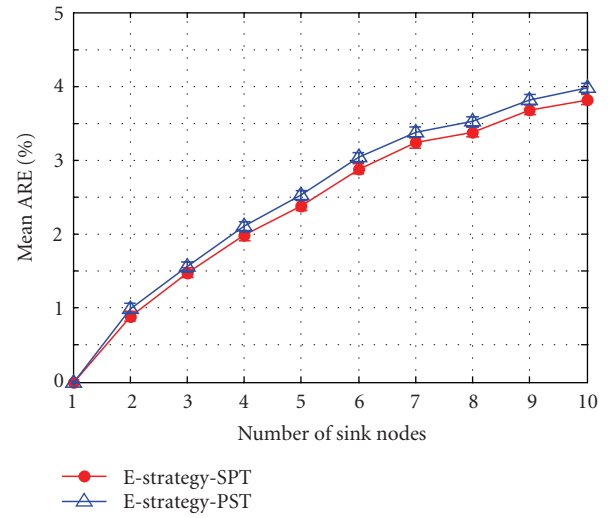

Figure 7: Accuracy versus number of sink nodes.

of sink nodes gets to 10, E-strategy with PST leads to a saving of 55% of communication cost over N-strategy.

Unlike the query distance and node density, the number of sink nodes does pose an impact on the accuracy of the reconstructed data series. As evidenced from Figure 7, the mean ARE increases with increasing number of sink nodes. This is because more sink nodes imply more varying frequencies, as well as the number of times that frequency conversion needs to be performed. Both factors result in larger mean ARE. However, even when the number of sink nodes becomes 10, the mean ARE is still no more than 5%. In other words, even for a good amount of sink nodes, the mean ARE is still tolerable.

6. CONCLUSION

Energy consumption is a crucial factor affecting the application and effectiveness of a wireless sensor network. In this paper, we proposed an energy-efficient framework in coping with multirate queries in WSNs. To the best of our knowledge, this is the first study that leverages existing research work and addresses the issues in this aspect. In summary, our technologies include the following: (1) an energy-efficient framework to process multirate queries; (2) an effective path-sharing routing tree construction method to make full use of the potential bandwidth sharing of all the data streams; and (3) a novel rate conversion mechanism to reconstruct the data stream at the desired frequency from the data stream at a different frequency. Both analytical and simulation results reveal that by tolerating a small degree of imprecision, our E-strategy can lead to a significant amount of communication cost savings, thereby extending the effective lifetime of WSNs.

Our work has broad impacts. With a tremendous spurt in sensor network deployment demanded by sensor network applications, our approach can effectively support generic sensor information query and data dissemination services.

There are several directions to extend our study. First, in the original model, we implicitly assume that the underlying architecture is based on the directed diffusion [9] routing mechanism. Extending our approach so that it can support other routing protocols would be one direction. Second, the rate conversion mechanism is feasible only if the requested sensor values are smoothly changing and can be well fitted by the applied linear interpolation. More accurate and better methodologies need to be explored. Finally, we wish to investigate the functionality of our system in a more dynamic situation, where nodes can join and leave the network frequently.

APPENDIX

Proof of Theorem 1. (1) If there is no point of intersection along the time axes of any pair of data series in the request, then every point of the data series should be collected. As a result, the dissemination frequency f_{up} achieves the upper bound as $\sum_{i=1}^{m} f_{ri}$.

(2) On the other hand, if the dissemination frequency f_d achieves the upper bound as $\sum_{i=1}^{m} f_{ri}$, we can make the proof by contradiction. Assuming at least two data series at frequencies f_{r1} and f_{r2}, respectively, have points of intersection, then the dissemination frequency f_d should be no more than $\sum_{i=1}^{m} f_{ri} - \gcd(f_{r1}, f_{r2})$, where function $\gcd(\cdot)$ means calculating the greatest common division. This contradicts with the precondition. □

Proof of Theorem 2. We can use the similar process to prove that the lower bound of the dissemination frequency f_{low} of each node can be achieved if and only if for any pair of data series in the request, they have points of intersection along their time axes. Next, we use *mathematical induction* to prove that the lower bound of the dissemination frequency f_{low} of each node can be calculated by expression (1).

(1) When m=1, it is obvious that the lower bound of the dissemination frequency $f_{\text{low}} = f_{r1}$. At the same time, expression (1) can be simplified as $(-1)^{1-1} \cdot \gcd(f_{r1}) = f_{r1}$. That is to say, the proposition holds true when $m = 1$. Furthermore, we can make the assumption that the conclusion holds true when $m = N$, where N is a positive integer. We will prove that the conclusion also holds true when $m = N + 1$ in the following part.

(2) When $m = N + 1$, then the lower bound of the dissemination frequency should be calculated as

$$\begin{aligned} f_{\text{low}} + f_{r(N+1)} - \gcd(f_{\text{low}}, f_{r(N+1)}) &= f_{\text{low}} + f_{r(N+1)} \\ &- \sum_{\{F_j\}_{j=1}^{1} \in \{f_{ri}\}_{i=1}^{N}} \gcd(\gcd(\{F_j\}_{j=1}^{1}), f_{r(N+1)}) \\ &+ \sum_{\{F_j\}_{j=1}^{2} \in \{f_{ri}\}_{i=1}^{N}} \gcd(\gcd(\{F_j\}_{j=1}^{2}), f_{r(N+1)}) + \cdots \\ &+ (-1)^{N} \cdot \gcd(\gcd(f_{r1}, f_{r2}, \ldots, f_{rN}), f_{r(N+1)}), \end{aligned} \tag{A.1}$$

where f_{low} is the lower bound of the dissemination frequency of the former N requested frequencies, which can be calculated as

$$f_{\text{low}} = \sum_{k=1}^{N} \left((-1)^{(k-1)} \cdot \sum_{\{F_j\}_{j=1}^{k} \in \{f_{ri}\}_{i=1}^{N}} \gcd(\{F_j\}_{j=1}^{k}) \right). \tag{A.2}$$

By adopting (A.2), expression (A.1) can be simplified as

$$\sum_{k=1}^{N+1} \left((-1)^{(k-1)} \cdot \sum_{\{F_j\}_{j=1}^{k} \in \{f_{ri}\}_{i=1}^{N+1}} \gcd(\{F_j\}_{j=1}^{k}) \right). \tag{A.3}$$

That is to say, the proposition also holds true when $m = N + 1$.

As a result, Theorem 2 always holds true when m is a positive integer. □

Proof of Theorem 3. (1) First, we prove $f_{\text{low}} \geq \max\{f_{ri}\}_{i=1}^{m}$.

Supposing $f_{\text{low}} < \max\{f_{ri}\}_{i=1}^{m}$, this conflicts with the N-strategy that the source node will disseminate the data at all the requested frequencies separately, including $\max\{f_{ri}\}_{i=1}^{m}$. As a result, we have $f_{\text{low}} \geq \max\{f_{ri}\}_{i=1}^{m}$.

(2) Now we use *mathematical induction* to prove $f_{\text{low}} \max\{f_{ri}\}_{i=1}^{m}$ if and only if for all $j \geq i$, $f_{ri} \mid f_{rj}$, $1 \leq i \leq j \leq m$.

(a) If $m = 1$, the proposition holds true.

(b) If $m = 2$, and from Theorem 2, we have $f_{\text{low}} f_{r1} + f_{r2} - \gcd(f_{r1}, f_{r2})$. It is obvious that $f_{\text{low}} \max(f_{r1}, f_{r2}) = f_{r2}$ if and only if $f_{r1} \mid f_{r2}$. That is to say, the proposition holds true when $m = N$, where N is a positive integer. We need to prove that the proposition also holds true when $m = N + 1$.

(c) When $m = N + 1$, from Theorem 2, we have

$$\begin{aligned} f'_{\text{low}} &= \sum_{k=1}^{N+1} \left((-1)^{(k-1)} \cdot \sum_{\{F_j\}_{j=1}^{k} \in \{f_{ri}\}_{i=1}^{N+1}} \gcd(\{F_j\}_{j=1}^{k}) \right) \\ &= f_{\text{low}} + f_{r(N+1)} - \gcd(f_{\text{low}}, f_{r(N+1)}) \end{aligned} \tag{A.4}$$

$$\begin{aligned} f'_{\text{low}} = \max\{f_{ri}\}_{i=1}^{N+1} = f_{r(N+1)} &\iff \\ f_{\text{low}} = \gcd(f_{\text{low}}, f_{r(N+1)}) &\iff f_{\text{low}} \mid f_{r(N+1)}. \end{aligned} \tag{A.5}$$

From (b), we know

$$f_{\text{low}} = \max\{f_{ri}\}_{i=1}^{N} = f_{rN} \iff \forall\, j \geq i,\ f_{ri} \mid f_{rj},\ 1 \leq i \leq j \leq N. \tag{A.6}$$

Together with (A.5), we have

$$f'_{\text{low}} = \max\{f_{ri}\}_{i=1}^{N+1} = f_{r(N+1)} \iff \forall\, j \geq i,\ f_{ri} \mid f_{rj},\ 1 \leq i \leq j \leq N+1. \tag{A.7}$$

Thus Theorem 3 holds true when m is a positive integer. □

ACKNOWLEDGMENTS

This research is partially supported by a research grant from the Department of Computing, the Hong Kong Polytechnic University, the Doctoral Foundation of National Education Ministry of China under Grant no.20059998022 and the National High-Tech R&D Program of China under Grant no.2006AA01Z198. The authors would like to express great appreciation to the reviewers of the paper for their valuable comments on improving the quality of this paper.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] U. Srivastaya, K. Munagala, and J. Widom, "Operator placement for in-network stream query processing," in *Proceedings of the 24th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS '05)*, pp. 250–258, Baltimore, Md, USA, June 2005.

[3] B. J. Bonfils and P. Bonnet, "Adaptive and decentralized operator placement for in-network query processing," *Telecommunication Systems*, vol. 26, no. 2–4, pp. 389–409, 2004.

[4] Y. Chen, H. V. Leong, M. Xu, J. Cao, K. C. C. Chan, and A. T. S. Chan, "In-network data processing for wireless sensor networks," in *Proceedings of the 7th International Conference on Mobile Data Management (MDM '06)*, p. 26, Nara, Japan, May 2006.

[5] B. Krishnamachari, D. Estrin, and S. Wicker, "Modelling data-centric routing in wireless sensor networks," in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, pp. 2–14, New York, NY, USA, June 2002.

[6] R. Cristescu, B. Beferull-Lozano, M. Vetterli, and R. Wattenhofer, "Network correlated data gathering with explicit communication: NP-completeness and algorithms," *IEEE/ACM Transactions on Networking*, vol. 14, no. 1, pp. 41–54, 2006.

[7] H. S. Kim, T. F. Abdelzaher, and W. H. Kwon, "Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 193–204, Los Angeles, Calif, USA, November 2003.

[8] B. Krishnamachari and J. Ahn, "Optimizing data replication for expanding ring-based queries in wireless sensor networks," in *Proceedings of the 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt '06)*, pp. 361–370, Boston, Mass, USA, April 2006.

[9] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 56–67, Boston, Mass, USA, August 2000.

[10] J. Lesurf, *Information and Measurement*, Institute of Physics, London, UK, 2002.

[11] L. Gao and X. S. Wang, "Continually evaluating similarity-based pattern queries on a streaming time series," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 370–381, Madison, Wis, USA, June 2002.