Implementation of Wireless Sensor Networks for Mobile Relay Configuration in Data-Intensive Applications

SAILESH KUMAR¹, JAYAKANTH², SAMSONPAUL³

¹PG Scholar, Dept of CSE, Stanley Stephen College of Engineering and Technology, India, Email: saileshkumarj@gmail.com.
²Dept of CSE, Stanley Stephen College of Engineering and Technology, India.
³Dept of CSE, Stanley Stephen College of Engineering and Technology, India.

Abstract: Wireless Sensor Networks (WSNs) are increasingly used in data-intensive applications such as microclimate monitoring, precision agriculture, and audio/video surveillance. A key challenge faced by data-intensive WSNs is to transmit all the data generated within an application’s lifetime to the base station despite the fact that sensor nodes have limited power supplies. We propose using low cost disposable mobile relays to reduce the energy consumption of data-intensive WSNs. Our approach differs from previous work in two main aspects. First, it does not require complex motion planning of mobile nodes, so it can be implemented on a number of low-cost mobile sensor platforms. Second, we integrate the energy consumption due to both mobility and wireless transmissions into a holistic optimization framework. Our framework consists of three main algorithms. The first algorithm computes an optimal routing tree assuming no nodes can move. The second algorithm improves the topology of the routing tree by greedily adding new nodes exploiting the newly added nodes. The third algorithm improves the routing tree by relocating its nodes without changing its topology. This iterative algorithm converges to the optimal position for each node given the constraint that the routing tree topology does not change. We present efficient distributed implementations for each algorithm that require only limited, localized synchronization. Because we do not necessarily compute an optimal topology, our final routing tree is not necessarily optimal. However, our simulation results show that our algorithms significantly outperform the best existing solutions.

Keywords: Wireless Sensor Networks, Energy Optimization, Mobile Nodes, Wireless Routing.

I. INTRODUCTION

WSNs have been deployed in a variety of data-intensive applications including microclimate and habitat monitoring [2], precision agriculture, and audio/video surveillance [3]. A moderate-size WSN can gather up to 1 Gb/year from a biological habitat [4]. Due to the limited storage capacity of sensor nodes, most data must be transmitted to the base station for archiving and analysis. However, sensor nodes must operate on limited power supplies such as batteries or small solar panels. Therefore, a key challenge faced by data-intensive WSNs is to minimize the energy consumption of sensor nodes so that all the data generated within the lifetime of the application can be transmitted to the base station. Several different approaches have been proposed to significantly reduce the energy cost of WSNs by using the mobility of nodes. A robotic unit may move around the network and collect data from static nodes through one-hop or multi-hop transmissions [5], [6], [7], [8], [9]. The mobile node may serve as the base station or a “data mule” that transports data between static nodes and the base station [10], [11], [12]. Mobile nodes may also be used as relays that forward data from source nodes to the base station.

Several movement strategies for mobile relays have been studied. Although the effectiveness of mobility in energy conservation is demonstrated by previous studies, the following key issues have not been collectively addressed. First, the movement cost of mobile nodes is not accounted for in the total network energy consumption. Instead, mobile nodes are often assumed to have replenished an able energy supply [8] which is not always feasible due to the constraints of the physical environment. Second, complex motion planning of mobile nodes is often assumed in existing solutions which introduces significant design complexity and manufacturing costs. In [8], [9] mobile nodes need to repeatedly compute optimal motion paths and change their location, their orientation and/or speed of movement. Such capabilities are usually not supported by existing low-cost mobile sensor platforms. For instance, Robomote nodes are designed using 8-bit CPUs and small batteries that only last for about 25 minutes in full motion.

In this paper, we use low-cost disposable mobile relays to reduce the total energy consumption of data-intensive WSNs. Different from mobile base station or data mules, mobile relays do not transport data; instead, they move to different locations and then remain stationary to forward
SAILESH KUMAR, JAYAKANTH, SAMSONPAUL

data along the paths from the sources to the base station. Thus, the communication delays can be significantly reduced compared with using mobile sinks or data mules. Moreover, each mobile node performs a single relocation unlike other approaches which require repeated relocations.

Our approach is motivated by the current state of mobile sensor platform technology. On the one hand, numerous low-cost mobile sensor prototypes such as their manufacturing cost is comparable to that of typical static sensor platforms. As a result, they can be massively deployed in a network and used in a disposable manner. Our approach takes advantage of this capability by assuming that we have a large number of mobile relay nodes. On the other hand, due to low manufacturing cost, existing mobile sensor platforms are typically powered by batteries and only capable of limited mobility. Consistent with this constraint, our approach only requires one-shot relocation to designated positions after deployment. Compared with our approach, existing mobility approaches typically assume a small number of powerful mobile nodes, which does not exploit the availability of many low-cost mobile nodes. We make the following contributions in this paper:

1. We formulate the problem of Optimal Mobile Relay Configuration (OMRC) in data-intensive WSNs. Our objective of energy conservation is holistic in that the total energy consumed by both mobility of relays and wireless transmissions is minimized, which is in contrast to existing mobility approaches that only minimize the transmission energy consumption. The tradeoff in energy consumption between mobility and transmission is exploited by configuring the positions of mobile relays.
2. We study the effect of the initial configuration on the final result. We compare different initial tree building strategies and propose an optimal tree construction strategy for static nodes with no mobility.
3. We develop two algorithms that iteratively refine the configuration of mobile relays. The first improves the tree topology by adding new nodes. It is not guaranteed to find the optimal topology. The second improves the routing tree by relocating nodes without changing the tree topology. It converges to the optimal node positions for the given topology. Our algorithms have efficient distributed implementations that require only limited, localized synchronization.
4. We conduct extensive simulations based on realistic energy models obtained from existing mobile and static sensor platforms. Our results show that our algorithms can reduce energy consumption by up to 45 percent compared to the best existing solutions.

The rest of the paper is organized as follows: Section II Existing System and Proposed Systems. In Section III, Related Works. Section IV describes our simulation results and Section V concludes this paper.

II. EXISTING SYSTEM AND PROPOSED SYSTEMS

A. Existing System

Wireless Multimedia Sensor Networks (WMSNs) has many challenges such as nature of wireless media and multimedia information transmission. Consequently traditional mechanisms for network layers are no longer acceptable or applicable for these networks. A key challenge faced by data-intensive WSNs is to transmit all the data generated within an application’s lifetime to the base station despite the fact that sensor nodes have limited power supplies.

1. Disadvantages:
   1. Require complex motion planning of mobile nodes.
   2. Energy consumption.

B. Proposed System

We use low-cost disposable mobile relays to reduce the total energy consumption of data intensive WSNs. Different from mobile base station or data mules, mobile relays do not transport data; instead, they move to different locations and then remain stationary to forward data along the paths from the sources to the base station. Thus, the communication delays can be significantly reduced compared with using mobile sinks or data mules. Moreover, each mobile node performs a single relocation unlike other approaches which require repeated relocations (fig 1).

1. Advantages
   1. We use low-cost disposable mobile relays to reduce the total energy consumption of data intensive WSNs.
   2. The total energy consumed by both mobility of relays and wireless transmissions is minimized.

Architecture:

Fig.1. System Architecture.

C. Implementation

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and it’s constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.
Mobile Relay Configuration in Data-Intensive Wireless Sensor Networks

D. Problem Statement
Several different approaches have been proposed to significantly reduce the energy cost of WSNs by using the mobility of nodes. A robotic unit may move around the network and collect data from static nodes through one-hop or multi-hop transmissions. The mobile node may serve as the base station or a “data mule” that transports data between static nodes and the base station. Mobile nodes may also be used as relays that forward data from source nodes to the base station. Although the effectiveness of mobility in energy conservation is demonstrated by previous studies, the following key issues have not been collectively addressed. First, the movement cost of mobile nodes is not accounted for in the total network energy consumption. Instead, mobile nodes are often assumed to have replenished an able energy supply which is not always feasible due to the constraints of the physical environment. Second, complex motion planning of mobile nodes is often assumed in existing solutions which introduces significant design complexity and manufacturing costs. In mobile nodes need to repeatedly compute optimal motion paths and change their location, their orientation and/or speed of movement. Such capabilities are usually not supported by existing low-cost mobile sensor platforms.

E. Module Description
1. Mobile Relays
The network consists of mobile relay nodes along with static base station and data sources. Relay nodes do not transport data; instead, they move to different locations to decrease the transmission costs. We use the mobile relay approach in this work showed that an iterative mobility algorithm where each relay node moves to the midpoint of its neighbors converges on the optimal solution for a single routing path. However, they do not account for the cost of moving the relay nodes. In mobile nodes decide to move only when moving is beneficial, but the only position considered is the midpoint of neighbors.

2. Sink
The sink is the point of contact for users of the sensor network. Each time the sink receives a question from a user, it first translates the question into multiple queries and then disseminates the queries to the corresponding mobile relay, which process the queries based on their data and return the query results to the sink. The sink unifies the query results from multiple storage nodes into the final answer and sends it back to the user.

3. Source Nodes
The source nodes in our problem formulation serve as storage points which cache the data gathered by other nodes and periodically transmit to the sink, in response to user queries. Such network architecture is consistent with the design of storage centric sensor networks. Our problem formulation also considers the initial positions of nodes and the amount of data that needs to be transmitted from each storage node to the sink.

4. Tree Optimization
We consider the sub problem of finding the optimal positions of relay nodes for a routing tree given that the topology is fixed. We assume the topology is a directed tree in which the leaves are sources and the root is the sink. We also assume that separate messages cannot be compressed or merged; that is, if two distinct messages of lengths m1 and m2 use the same link (si, sj ) on the path from a source to a sink, the total number of bits that must traverse link (si, sj ) is m1 + m2.

II. RELATED WORK
Analyzing the three different approaches: Mobile base stations, data mules and mobile relays all the three approaches use mobility to reduce energy consumption in wireless sensor networks

A. Mobile Base Station
A mobile base station is a sensor node collects the data by moving around the network from the nodes. In some work, in order to balance the transmission load, all nodes are performing multiple hop transmissions to the base station. The goal is to rotate the nodes which are close to the base station. Before the nodes suffer buffer overflows, the base station computes the mobility path to collect data from the visited nodes. Several rendezvous based data collection algorithms are proposed, where the mobile base station only visits a selected set of nodes referred to as rendezvous points within a deadline and the rendezvous points buffer the data from sources. High data traffic towards the base station is always a threat to the networks life time. The battery life of the base station gets depleted very quickly due to the sensor nodes which are located near to the base station relay data for large part of the network. The proposed solution includes the mobility of the base station such that nodes located near base station changes over time. All the above approaches incur high latency due to the low to moderate speed of mobile base stations. Fig.2 shows Mobile base station

Fig.2. Mobile base station.

B. Data Mules
Data mules are another form of base stations. They gather data from the sensors and carry it to the sink. The data mule collects the data by visiting all the sources and

International Journal of Advanced Technology and Innovative Research
Volume. 06, IssueNo.12, December-2014, Pages: 1403-1410
then transmits it to the static base station through the network. In order to minimize the communication and mobility energy consumption the mobility paths are determined. In paper the author analyses an architecture based on mobility to address the energy efficient data collection problem in a sensor network. This approach utilizes the mobile nodes as forwarding agents. As a mobile node moves in close propinquity to sensors, data is transmitted to the mobile node for later dumps at the destination. In the MULE architecture sensors transmit data only over a short range that requires less transmission power. However, latency is increased because a sensor has to wait for a mule before its data can be delivered. Fig.3 the three tiers of the MULE architecture. The Mule architecture has high latency and this limits its applicability to real time applications (although this can be mitigated by collapsing the MULE and access point tiers). The system requires sufficient mobility. For example, mules may not arrive at a sensor or after picking the data may not reach near an access-point to deliver it. Also, data may be lost because of radio-communication errors or mules crashing. To improve data delivery, higher-level protocols need to be incorporated in the MULE architecture. Data mules also introduce large delays like base stations since sensors have to wait for a mule to pass by before initiating their transmission.

**Fig.3. The three tiers of the MULE architecture.**

**C. Mobile Relay**

In this approach, the network consists of three nodes such as mobile relay nodes along with static base station and data sources. To reduce the transmission cost relay nodes do not transport data rather it will move to different locations. We use the mobile relay approach in this work. In author showed that an iterative mobility algorithm where each relay node moves to the midpoint of its neighbors converges on the optimal solution for a single routing path this paper presents mobility control scheme for improving communication performance in WSN. The objectives of the paper are 1) Analyze when controlled mobility can improve fundamental networking performance metrics such as power efficiency and robustness of communications 2) Provide initial design for such networks. Mobile nodes move to midpoint of the neighbors only when movement is beneficial. Unlike mobile base stations and data mules, our approach reduces the energy consumption of both mobility and transmission. Our approach also relocates each mobile relay only once immediately after deployment. The paper study the energy optimization problem that accounts for energy costs associated with both communication and physical node movement. Unlike previous mobile relay schemes the proposed solution consider all possible locations as possible target locations for a mobile node instead of just the midpoint of its neighbors.

**IV. SIMULATIONS**

We carried out simulations on 100 randomly generated initial topologies, each of which has 100 nodes placed uniformly at random within a 150 m by 150 m area. We used these initial topologies to generate two subsequent sets of complete topologies with established sources and sinks. We used the first set to study the effectiveness of our algorithms as the amount of data transferred to the sink varies and the second set to study the effectiveness of our algorithms for different numbers of sources. In the first set, we selected sources and sinks uniformly at random from these 100 nodes. We varied the number of sources from 4 to 12, by increments of two, and used each number of sources for 20 initial topologies. For each resulting topology, we created many separate input instances by varying the data chunk size from 1 to 150 MB where the data chunk size for an input instance is the common amount of data to be transferred from each source to the sink. In the second set, for each initial topology, we generated 10 different complete topologies by starting with two randomly selected sources, and adding two new sources to the previous set at each step.

We used the following settings to model the transmission and mobility costs of our nodes. For transmission, we use $a = 0.6 \times 10^{-7}$ and $b = 4 \times 10^{-10}$ as the standard setting which is consistent with the empirical measurements on CC2420 motes. For mobility, we used different settings in each of our two sets. In the first set, we used $k = 2$ as the standard setting because it models several platforms such as Robomote. In the second set, we set $k$ to be 1, 2, and 4 since we additionally use that set to study the effect of different mobility costs on the energy reduction. Furthermore, we set the maximum communication distance of a node to be 30 m, which was shown to result in a high packet reception ratio for the CC2420 radio. We ran simulations using different values for the convergence threshold. We obtained similar gains for values less than or equal to 0.01. In the following simulations, we set the threshold to 0.01.

Our algorithmic framework starts with an initial routing tree. In the centralized setting, we construct this initial routing tree using the following three widely used routing algorithms: power-based routing, hop-based routing, and greedy geographic routing. Power-based routing computes a shortest path from the sink to each
source with each edge weight being the square of the distance between the two corresponding nodes plus some constant value to represent the energy consumed $a + bd^2$ to transmit each byte of data over that edge. Hop-based routing minimizes the number of hops between each source and the sink and is the base of several widely used algorithms in wireless networks (e.g., AODV). Given our maximum communication range of 30 m, we do not have any links with poor quality which is a common concern with hop-based routing.

Greedy geographic routing is a greedy strategy in which each node forwards messages to the reachable node (within the communication range of the node) that is closest to the sink. The first two tree construction approaches require global knowledge of the network whereas the last one is fully localized. For the distributed setting, we construct the initial routing tree using greedy geographic routing because it is fully localized. Of the 100 initial topologies, the distributed routing algorithm resulted in a disconnected path between the sources and the sink in only four networks given our maximum communication distance of 30 m. We study variants of our strategy where we use only one optimization, inserting nodes or optimizing a given tree, to determine the benefit of both optimizations. Specifically, we use TREE to represent the variant where we only construct an initial tree and do no optimizations, TREE+FO to represent the variant where we optimize the initial tree, TREE+INS to represent the variant where we

insert nodes into the initial tree, and TREE+INS + FO to represent the variant where we insert nodes into the initial tree and then optimize the final tree. The three possibilities for TREE are PB, HB, and GG which represent the Power Based, Hop Based, and Greedy Geographic tree construction algorithms, respectively. For each input instance $I$, we let TREE $(I)$ denote the energy consumed by the initial tree constructed by our three tree construction algorithms PB, HB, and GG, and we let TREE+OPT $(I)$ denote the energy consumed by the final optimized tree where TREE can be PB, HB, or GG and OPT can be INS, FO or INS+FO.

The reduction ratio achieved by optimization OPT on input $I$ for tree construction algorithm TREE is $(TREE+OPT (I))/TREE (I)$. We measure the performance of optimization OPT on initial tree strategy TREE by computing the average reduction ratio achieved by OPT over all input instances $I$ of set 1 that have the same data chunk size. Moreover, for each input instance $I$ and each algorithm TREE+OPT, we define the static energy ratio $(TREE + OPT (I)/PB (I)$ where $PB (I)$ is the cost of the power-based tree which is the optimal cost for the static version of this problem where no nodes can move. The static energy ratio measures the benefit of our algorithms which exploit mobility of nodes versus the static optimal configuration. We measure the overall performance of algorithm TREE+OPT by computing the average static energy ratio achieved by TREE+OPT over all input instances $I$ of set 1 that have the same data chunk size. Finally, we measure the performance of optimization INS + FO on initial tree strategy TREE by computing the average reduction ratio achieved by INS+FO over all input instances $I$ of set 2 that have the same number of sources.

A. Centralized Algorithm

We first show the benefit of exploiting the mobility of relay nodes by computing the average static energy consumption ratio of TREE+INS+FO for all data chunk sizes for each of our three tree building strategies PB, HB, and GG as shown in Fig.4. For all three initial tree strategies, we see that the average static energy consumption ratio drops quickly as the data chunk size increases. For HB and GG, the average static energy consumption ratio starts out higher than 100 percent because PB $(I)$, the optimal tree for the static case, is roughly 37 percent lower than HB$(I)$ and GG $(I)$ for any of our input instances. Even given this initial disadvantage of a poor starting tree from an energy consumption perspective, we see that the average static energy consumption ratios of HB+INS+FO and GG+INS+FO drop below 100 percent for data chunk sizes of 12 and 15 MB, respectively. As the data chunk size increases further, HB+INS+FO and GG+INS+FO achieve average static energy consumption ratios of 75 and 60 percent for data chunk sizes of 60 and 150 MB, respectively. The results for PB+INS+FO are even better because we start with the optimal tree for the static case. Thus, the average static energy consumption ratio for PB+INS+FO is always below 100 percent and reaches 55 percent for 150 MB.

We now evaluate the benefit achieved by our optimizations FO, INS, and INS+FO for each of our tree building strategies PB, HB, and GG. We note that in this set of simulations, we used our centralized improvement schemes with the distributed tree building approach GG.

![Comparison to power-based original configuration](image.png)

Fig.4. Graph of the average static energy consumption ratio of TREE + INS + FO as a function of data chunk size for our three tree construction strategies PB, HB, and GG.
The purpose is to test the limits of our optimizations given a non-optimal starting tree. A fully distributed setup is studied later in this section. We start with optimization INS+FO. Fig. 5 plots the average reduction ratio for optimization INS+FO for PB, HB, and GG. In all three cases, we see the same basic trend; the average reduction ratio increases as data chunk size increases. For both HB and GG, the average reduction ratio starts at roughly 25 percent for small data chunk sizes and exceeds 60 percent for large data chunk sizes; for PB the average reduction ratio starts near 0 percent and exceeds 43 percent for large data chunk sizes.

![Graph of the average reduction ratio of optimization INS+FO as a function of data chunk size for our three tree construction strategies PB, HB, and GG.](image1)

The difference in average reduction ratio, in particular for small data chunk sizes, is due to the quality of the initial tree. For PB, the initial tree is good so there is little that our optimization INS+FO can do to improve energy consumption for small data chunk sizes. For HB and GG, the initial tree can be very poor, so INS+FO can provide immediate improvement to the tree to significantly reduce energy consumption by an average of 25 percent for data chunk sizes of only 1 MB. We note that although INS+FO achieves higher reduction ratios for HB and GG than for PB, the total energy consumed by PB+INS+FO is lower than the total energy consumed by HB+INS+FO or GG+INS+FO. We next consider optimization FO alone. Fig. 6 plots the average reduction ratio for optimization FO for PB, HB, and GG. In all three cases, the average reduction ratio starts at 0 percent for small data chunk sizes and increases to roughly 18 percent for HB and GG and 33 percent for PB for large data chunk sizes. It is interesting to note that FO is most effective for PB whereas INS+FO achieved significantly greater reduction ratios for GG and HB for all data chunk sizes.

![Graph of the average reduction ratio of optimization FO as a function of data chunk size for our three tree construction strategies PB, HB, and GG.](image2)

Finally, we consider optimization INS alone. Fig. 7 plots the average reduction ratio for optimization INS for PB, HB, and GG. In all three cases, we see the average reduction ratio of INS alone is comparable to that of INS+FO (within 5-8 percent for data chunk sizes of at least 15 MB). For very small data chunk sizes, the average reduction ratio is constant until a certain threshold is exceeded and then rises significantly. We now evaluate our approach as we vary the number of sources. We used the greedy geographic tree GG as our initial tree and INS+FO as our optimization algorithm. Fig. 8 shows the average reduction ratio as a function of the number of sources. We observe that this ratio remains almost constant for different values of k as the difference in ratios for different number of sources does not exceed 3.5 percent. Fig. 8 also shows the effect of mobility costs on the reduction in energy consumption costs in general. As mobility costs decrease, it becomes more effective for mobile nodes to move over longer distances and reduce the communication consumption further so the reduction in total costs increases as k decreases. Given our simulation results, we draw the following five conclusions. First, we achieve the best results when we use...
the power-based tree PB as our initial tree. Second, if we use the power-based tree, either optimization alone is very effective and both optimizations together achieve the best results. Third, if we start with either the hop-based tree HB or the greedy geographic tree GG, the most effective optimization is the node insertion optimization INS which achieves nearly as good an average reduction ratio as INS+FO. Fourth, if we start with the hop-based or greedy geographic tree, we can achieve a static energy ratio that is close to that achieved by starting with the power-based tree if we apply both optimizations. In particular, the node insertion optimization INS helps alleviate the initial disadvantage by adding a lot of new nodes into the tree. We briefly explain the reason for all of these conclusions. The key observation is that the hop-based and greedy geographic trees HB and GG tend to create initial trees with relatively long edges and relatively few nodes whereas the power-based tree PB tends to create trees with lots of nodes and relatively short edges because of the quadratic cost metric.

As a result, for HB and GG, optimization FO alone which rearranges nodes is relatively ineffective as it can only balance the relatively long edges. On the other hand, optimization INS alone can insert new nodes into the tree and thus create a new tree with significantly shorter edges on average given HB or GG as the initial tree. Because PB starts with many more nodes and shorter edges, PB does not benefit as much from node insertion INS as HB and GG do, and PB benefits a lot more from node rearrangement FO than HB and GG do. Fifth, the improvement ratios that we obtain are almost independent of the number of sources in the network. In all our simulation results, the standard deviation varied between 4 and 6.5 percent. We identified six outlier topologies which deviated from the mean by more than 10 percent. In these topologies, the sources were either very close to the sink so there was little room for improvement or very far from the sink so the improvement was much greater than the average case.

B. Distributed Algorithm

We now evaluate how well our distributed implementation works. Our initial tree is the greedy geographic tree GG. We consider four optimizations: the centralized implementation of INS+FO, the distributed implementation of just FO, the distributed implementation of just INS, and the distributed implementation of INS followed by the distributed implementation of FO. For the distributed implementation of INS, we set parameter B to 10 percent (a potential offer must be 10 percent better than the best actual offer to cause a node to wait). Fig.9 shows the average reduction ratio of each of these optimizations. The average reduction ratio for distributed INS+FO starts at 20 percent for small data chunk sizes, reaches 30 percent for data chunk sizes around 20 MB, and exceeds 40 percent for data chunk sizes larger than 75 MB. The gap between the average reduction ratio for centralized INS+FO and distributed INS+FO starts at roughly 5 percent for small data chunk sizes and increases to roughly 15 percent for large data chunk sizes. This gap is due to the lack of global information when performing the insertion step.

Expensive links in the tree that do not have nearby relay nodes are not able to communicate with further but available relay nodes whose help is only offered to cheaper but nearby links. This problem is exacerbated as the data chunk size increases. We varied values for B between 10 and 50 percent and for R between 1 and 3. For all combinations of B and R that we tested, we obtained similar results to those of Fig.9. As in the centralized case, distributed INS is more effective than distributed FO.
However, doing both distributed optimizations does result in roughly a 10 percent improvement compared to only doing the distributed INS optimization for most data chunk sizes. Similar to the centralized implementation, we observe a slow reduction in the improvement ratio as the number of sources increases for $k = 2$ and 4 (Fig. 8). For cheaper mobility cost ($k = 1$), the difference in improvement ratios increases at a faster rate and reaches 9 percent as the number of sources increases from 2 to 20. This is because when mobility is cheaper, in an optimal setting, nodes can move over longer distances to help expensive links. However, as we mentioned earlier, in a distributed setting, mobile nodes are not aware of those distant expensive edges moreover, as the number of sources increases, the number of mobile nodes available to help decreases. Both factors combined make the distributed implementation slightly less effective for a high number of sources.

V. CONCLUSION

In this paper, we proposed a holistic approach to minimize the total energy consumed by both mobility of relays and wireless transmissions. Most previous work ignored the energy consumed by moving mobile relays. When we model both sources of energy consumption, the optimal position of a node that receives data from one or multiple neighbors and transmits it to a single parent is not the midpoint of its neighbors; instead, it converges to this position as the amount of data transmitted goes to infinity. Ideally, we start with the optimal initial routing tree in a static environment where no nodes can move. However, our approach can work with less optimal initial configurations including one generated using only local information such as greedy geographic routing. Our approach improves the initial configuration using two iterative schemes. The first inserts new nodes into the tree. The second computes the optimal positions of relay nodes in the tree given a fixed topology. This algorithm is appropriate for a variety of data-intensive wireless sensor networks. It allows some nodes to move while others do not because any local improvement for a given mobile relay is a global improvement. This allows us to potentially extend our approach to handle additional constraints on individual nodes such as low energy levels or mobility restrictions due to application requirements. Our approach can be implemented in a centralized or distributed fashion. Our simulations show it substantially reduces the energy consumption by up to 45 percent.

VI. REFERENCES