Volume 5, Issue 1, January 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Minimizes the Energy Consumption in Wireless Sensor Networks

G. Ravi Chandra Reddy, Dr. B.TarakeswaraRao, B.Satyanarayana Reddy
M.Tech Student, Dept of CSE, Professor, Dept of CSE,Assoc.Professor, Dept of CSE
KHIT, JNTUK, Guntur,Ap, India.KHIT, JNTUK, Guntur,Ap, India.KHIT, JNTUK, Guntur, Ap, India

Abstract--WSN are increasingly used in different types of data-intensive applications scenarios such as micro-climate monitoring,precision agriculture, and audio/video surveillance. The sensor nodes are tiny and limited in power. Sensor types vary according to the application of WSN. Our study focused on the problem of Optimal Mobile Relay Configuration (OMRC) in data-intensive WSNs. The concept of mobile relay is that the mobile nodes change their locations so as to minimize the total energy consumed by both wireless transmission and locomotion. 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. For reduce the energy consumption we propose 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. Third algorithm improves the routing tree by relocating its nodes without changing its topology.

Keywords--WSN, Energy optimization, Relay, Routing tree, mobile nodes.

I. INTRODUCTION

A wireless sensor network (WSN) consists of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as heat, vibration , pressure,humidity,motion or pollutants and to cooperatively pass their data through the network to a main location.The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control,machine health monitoring, and so on.

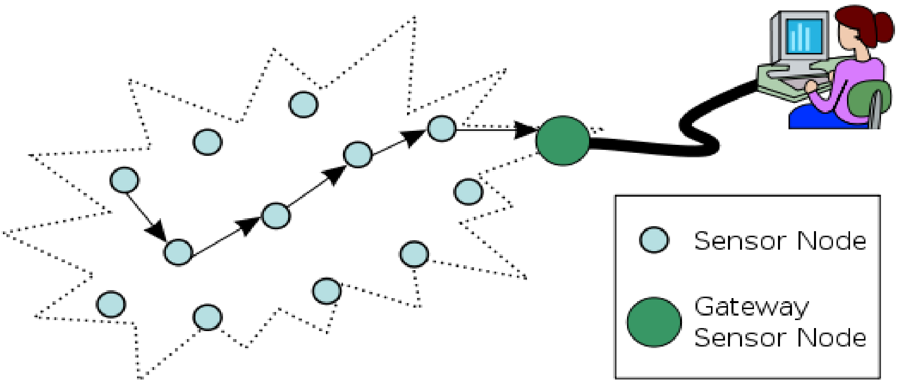


Figure1:Typical multi-hop wireless sensor network architecture

Recent advancement in mobile sensor platform technology has been taken into attention that mobile elements are utilized to improve the Actor network performances such as coverage, connectivity, reliability and energy efficiency. The concept of mobile relay is that the mobile nodes change their locations so as to minimize the total energy consumed by both wireless transmission and locomotion. The conventional methods, however, do not take into account the energy level, and as a result they do not always prolong the network lifetime.A use of Petersen-size WSAN can gather up to 1 Gb/year from a biological habitat [1].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 using of some adjust the max life time power supplies such as batteries or small solar panels(for external use of these both). 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 to destination.Several different approaches have been proposed to significantly low the energy cost of WSNs by using the mobility of nodes.In [2], [3], [4], mobile nodes need to repeatedly compute possible to 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.

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 don't send the data to source; 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.Com-pared 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.

II. METHODS TO REDUCE ENERGY CONSUMPTION

We review three different approaches, mobile base stations, data mules, and mobile relays,that 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 [4]. 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.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. These approaches incur high latencies due to the low to moderate speed, e.g., 0.1-1 m/s [5], of mobile base stations.

B. Data Mules

Data mules are similar to the second form of mobile base stations [6]. They pick up data from the sensors and transport it to the sink. In [7], the data mule visits all the sources to collect data, transports data over some distance, and then transmits it to the static base station through the network. The goal is to find a movement path that minimizes both communication and mobility energy consumption. Similar to mobile base stations, data mules introduce large delays since sensors have to wait for a mule to pass by before starting their transmission.

C. *Mobile Relays*

In the third approach, 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. Goldenberg et al. [8] 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 [9], mobile nodes decide to move only when moving is beneficial, but the only position considered is the midpoint of neighbors.

Unlike mobile base stations and data mules, our OMRC problem considers the energy consumption of both mobility and transmission. Our approach also relocates each mobile relay only once immediately after deployment. Unlike previous mobile relay schemes [8] and [9], we consider all possible locations as possible target locations for a mobile node instead of just the midpoint of its neighbors

III. ALGORITHMS

A. Static Tree Construction

Different applications may apply different constraints on the routing tree. When only optimizing energy consumption, a shortest path strategy (as discussed below) yields an optimal routing tree given no mobility of nodes. However, in some applications, we do not have the freedom of selecting the routes. Instead, they are predetermined according to some other factors (such as delay, capacity, etc.). In other less stringent cases, we may be able to update the given routes provided we keep the main structure of the tree. Depending on the route constraints dictated by the application, we start our solution at different phases of the algorithm. In the unrestricted case, we start at the first step of constructing the tree. When the given tree must be loosely preserved, we start with the relay insertion step. Finally, with fixed routes, we apply directly our tree optimization algorithm.We construct the tree for our starting configuration using a shortest path strategy. We first define a weight function w specific to our communication energy model. For each pair of nodes si and sj in the network, we define the weight of edge sisj as: w(si, sj) = a + b||oi − oj ||2 where oi and oj are the original positions of nodes si and sj and a and b are the energy parameters.

We observe that using this weight function, the optimal tree in a static environment coincides with the shortest path tree rooted at the sink. So we apply Dijkstra's shortest path algorithm starting at the sink to all the source nodes to obtain our initial topology. We improve the routing tree by greedily adding nodes to the routing tree exploiting the mobility of the inserted nodes. For each node sout that is not in the tree and each tree edge sisj , we compute the reduction (or increase) in the total cost along with the optimal position of sout if sout joins the tree such that data is routed from si to sout to sj instead of directly from si to sj using the LocalPos algorithm described in algorithm 1. We repeatedly insert the outside node with the highest reduction value modifying the topology to include the selected node at its optimal position, though the node will not actually move until the completion of the tree optimization phase. After each node insertion occurs, we compute the reduction in total cost and optimal position for each remaining outside node for the two newly added edges (and remove this information for the edge that no longer exists in the tree). At the end of this step, the topology of the routing tree is fixed and its mobile nodes can start the tree optimization phase to relocate to their optimal positions.

Algorithm 1:

▻**function**LOCALPOS(o_i, u_i, u_{i-1}, u_{i+1})
  Consider case s_i *moves right*
  valid ← FALSE;
$x_i \leftarrow \frac{1}{2}(x_{i-1} + x_{i+1}) - Y_i$;
if$x_i > p_i$ **then**
 valid ← TRUE;
 else
 ▻*Consider case* s_i *moves left*
 $x_i \leftarrow \frac{1}{2}(x_{i-1} + x_{i+1}) + Y_i$;
 if$x_i < p_i$ **then**
 valid ← TRUE;
 end if
end if
▻*Record if new position is different from previous one*
ifvalid **then**
 $y_i \leftarrow \frac{(x_{i-1}+x_{i+1}-2p_i)}{(y_{i-1}+y_{i+1}-2q_i)}(x_i - p_i) + q_i$;
 $u' = (x_i, y_i)$;
 if $\|u'_i - u_i\|$ > threshold **then**
 return(u_i, TRUE);
 end if
end if
 ▻*not beneficial to move, stay at original position*
  return(o_i, FALSE);
end function

Fig 2. Algorithm to compute the optimal position of a relay node that receives data from a single node and transmits the data to a single node.

B. Tree Optimization Algorithm

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. Let the network consists of multiple sources, one relay node and one sink such that data is transmitted from each source to the relay node and then to the sink. We modify our solution as follows. Let si be the mobile relay node, S(si) the set of source nodes transmitting to si and sdi the sink collecting nodes from si. The cost incurred by si in this configuration U is:

$$ci(U) = k\|ui - oi\| + ami + bmi\|ud - ui\|2$$

$$x_i = p_i + \frac{-Bx(\sqrt{pB^2_x + B^2_y} \pm k)}{A\sqrt{pB^2_x + B^2_y}}$$

$$y_i = q_i + \frac{-By(\sqrt{pB^2_x + B^2_y} \pm k)}{A\sqrt{pB^2_x + B^2_y}}$$

Where

$$A = mi + \sum_{sl \in S(si)} ml$$

$$Bx = mixd + \sum_{sl \in S(si)} mlxl + Api$$

$$By = miyd + \sum_{sl \in S(si)} mlyl + Aqi$$

These values correspond to two candidate points moving in each direction (left/right). The optimal position is the valid value yielding the minimum cost. Our algorithm starts by an odd/even labeling step followed by a weighting step. To obtain consistent labels for nodes, we start the labeling process from the root using a breadth first traversal of the tree. The root gets labeled as even. Each of its children gets labeled as odd. Each subsequent child is then given the opposite label of its parent. We define mi, the weight of a node si, to be the sum of message lengths over all paths passing throughsi. This computation starts from the sources or leaves of our routing tree. Initially, we know mi = Mi for each source leafnode si. For each intermediate node si, we compute its weight as the sum of the weights of its children. Once each node gets a weight and a label, we start our iterative scheme. In odd iterations j, the algorithm computes a position uj i for each odd-labeled node si that minimizes Ci(Uj) assuming that uj i−1 = uj−1i−1 and uji+1 = uj−1i+1 ; that is, node si's even numbered neighboring nodes remain in place in configuration Uj . In even-numbered iterations, the controller does the same for even-labeled nodes. The algorithm behaves this way because the optimization of uj i requires a fixed location for the child nodes and the parent of si.By alternating between optimizing for odd and even labeled nodes, the algorithm guarantees that the node si is always making progress towards the optimal position ui. Our iterative algorithm is shown in algorithm.

Algorithm 2 :

procedureOPTIMALPOSITIONS(U^0)
converged ← false;
j ← 0;
repeat
anymove ← false;
j⊳← j + 1;
Start an even iteration followed by an odd iteration
foridx = 2 to 3 **do**
fori = idx to n by 2 **do**
(u^j_i ,moved) ← LOCALPOS(o_i, S(s_i), s^d_i);
anymove ← anymove OR moved
end for
end for
converged ← NOT anymove
untilconverged
end procedure

Fig. 3. Centralized algorithm to compute the optimal positions in a given tree.

IV. IMPLEMENTATION

We now present a centralized approach to solve OMRC that breaks the problem into three distinct steps: initial tree construction, node insertions, and tree optimization. For each step, we present an algorithm to solve the correspond-ingsub problem. Our algorithm for initial tree construction is optimal for the static environment where nodes cannot move. However, we can effectively apply the later algorithms if we must start with a different topology. Our greedy heuristic for improving the routing tree topology by adding nodes exploits the mobility of the newly added nodes. Our tree optimization algorithm improves the routing tree by relocating its nodes without changing its topology. This iterative algorithm converges on the optimal position for each node given the constraint that the routing tree topology is fixed. Our node insertion and tree optimization algorithms use the LocalPos algorithm we propose in Fig. 2 that optimally solves the simplest case of the mobile relay configuration problem where there is a single source, a single sink, and a single relay node.

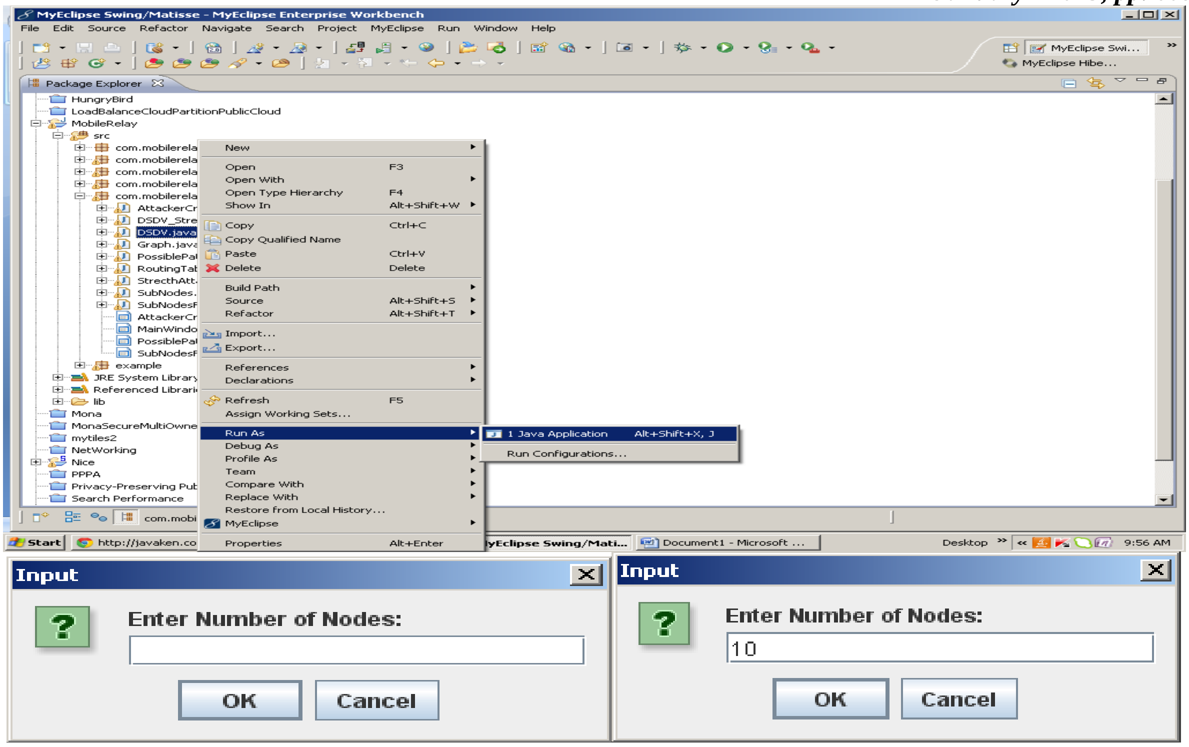
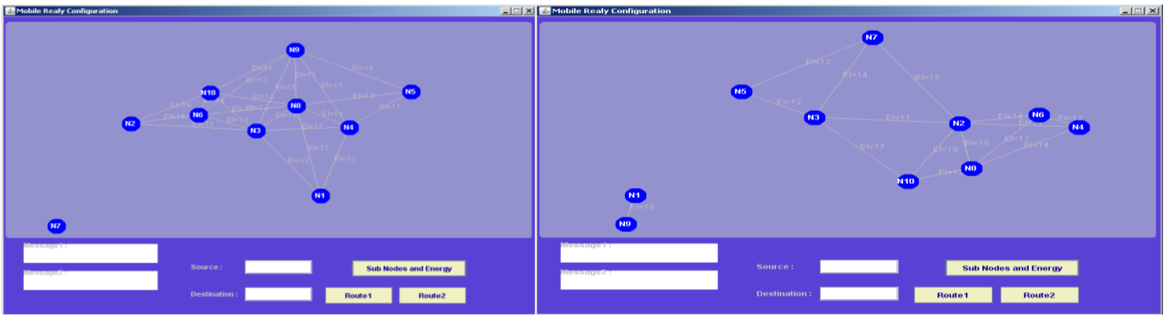
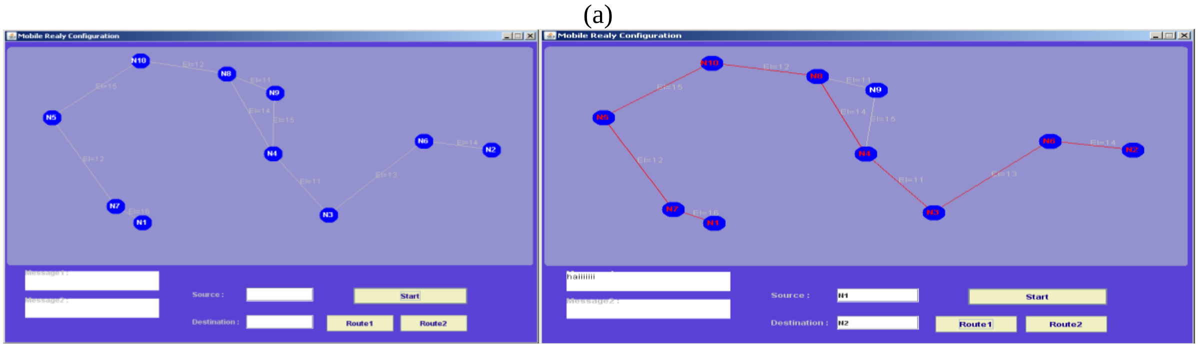


Fig 4.



(a)



(b)

Fig. 4.a,b Example of optimal configurations as a function of amount of data to be transferred. In each part, source nodes n_1 and n_2 must send m bits of data to the sink n_4.

V. FUTURE ENHANCEMENT

As future work, particular cases may be studied by the consideration of additional features related to the link level, routing scheme or application layer in the constraints of the optimization problem. This may lead to a more reduced solution space, reaching optimal results and narrowing the gap to the simulation results obtained by our study. On the other hand, feasibility/compatibility studies between our OPT-PRIM proposal and WSN routing algorithms, such as ALBA-R (Adaptive Load-Balanced Algorithm, Rainbow version)] or XLP (Cross Layer Protocol), might mean the enhancement of network lifetime and other network performance figures, such as end-to-end delay or packet delivery ratio.

VI. CONCLUSION

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.

REFERENCES

[1] D. Ganesan, B. Greenstein, D. Perelyubskiy, D. Estrin, and J. Heidemann, “An Evaluation of Multi-Resolution Storage for Sensor Networks,” Proc. First Int’l Conf. Embedded Networked Sensor Systems (SenSys), 2003.

[2] A. Kansal, D.D. Jea, D. Estrin, and M.B. Srivastava, “Controllably Mobile Infrastructure for Low Energy Embedded Networks,” IEEE Trans. Mobile Computing, vol. 5, no. 8, pp. 958-973, Aug. 2006.

[3] G. Xing, T. Wang, W. Jia, and M. Li, “Rendezvous Design Algorithms for Wireless Sensor Networks with a Mobile Base Station,” Proc. ACM MobiHoc, pp. 231-240, 2008.

[4] A.A. Somasundara, A. Ramamoorthy, and M.B. Srivastava, “Mobile Element Scheduling with Dynamic Deadlines,” IEEE Trans. Mobile Computing, vol. 6, no. 4, pp. 395-410, Apr. 2007.

[5] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G.S. Sukhatme, “Robomote: Enabling Mobility in Sensor Networks,”Proc. Fourth Int’l Conf. Information Processing in Sensor Networks (IPSN), 2005.

[6] D.Jea, A.A. Somasundara, and M.B. Srivastava, “Multiple Controlled Mobile Elements (Data Mules) for Data Collection in Sensor Networks,” Proc. IEEE First Int’l Conf. Distributed Comput-ing in Sensor Systems (DCOSS), 2005.

[7] C.-C. Ooi and C. Schindelhauer, “Minimal Energy Path Planning for Wireless Robots,” Proc.First Int’l Conf. Robot Comm. and Coordination (ROBOCOMM), p. 2, 2007.

[8] D.K. Goldenberg, J. Lin, and A.S. Morse, “Towards Mobility as a Network Control Primitive,” Proc. ACM MobiHoc, pp. 163-174, 2004.

[9] C. Tang and P.K. McKinley, “Energy Optimization under Informed Mobility,” IEEE Trans. Parallel and Distributed Systems, vol. 17, no. 9, pp. 947-962, Sept. 2006.

