



Path-Constrained Data Gathering Scheme

By Khaled Almiani, Ahmad A. Twaissi, Mohammed A. Abuhelaleh,
Bassam A. Alqaralleh & Albara Awajan

Al-Hussein Bin Talal University, Jordan

Abstract - Several studies in recent years have considered the use of mobile elements for data gathering in wireless sensor networks, so as to reduce the need for multi-hop forwarding among the sensor nodes and thereby prolong the network lifetime. Since, typically, practical constraints preclude a mobile element from visiting all nodes in the sensor network, the solution must involve a combination of a mobile element visiting a subset of the nodes (cache points), while other nodes communicate their data to the cache points wirelessly. This leads to the optimization problem of minimizing the communication distance of the sensor nodes, while keeping the tour length of the mobile element below a given constraint. In this paper, we investigate the problem of designing the mobile elements tours such that the length of each tour is below a per-determined length and the number of hops between the tours and the nodes not included in the tour is minimized. To address this problem, we present an algorithmic solution that consider the distribution of the nodes during the process of building the tours. We compare the resulting performance of our algorithm with the best known comparable schemes in the literature.

GJCST-E Classification : G.2.2



PATH-CONSTRAINED DATA GATHERING SCHEME

Strictly as per the compliance and regulations of:



RESEARCH | DIVERSITY | ETHICS

Path-Constrained Data Gathering Scheme

Khaled Almiani ^α, Ahmad A. Twaissi ^σ, Mohammed A. Abuhelaleh ^ρ, Bassam A. Alqaralleh ^ω
& Albara Awajan^{*}

Abstract - Several studies in recent years have considered the use of mobile elements for data gathering in wireless sensor networks, so as to reduce the need for multi-hop forwarding among the sensor nodes and thereby prolong the network lifetime. Since, typically, practical constraints preclude a mobile element from visiting all nodes in the sensor network, the solution must involve a combination of a mobile element visiting a subset of the nodes (cache points), while other nodes communicate their data to the cache points wirelessly. This leads to the optimization problem of minimizing the communication distance of the sensor nodes, while keeping the tour length of the mobile element below a given constraint. In this paper, we investigate the problem of designing the mobile elements tours such that the length of each tour is below a per-determined length and the routing trees size is minimized. To address this problem, we present an algorithmic solution that consider the distribution of the nodes during the process of building the tours. We compare the resulting performance of our algorithm with the best known comparable schemes in the literature.

I. INTRODUCTION

Many typical applications of wireless sensor networks (WSNs) involve the collection of data obtained by sensor nodes at a pre-defined sink. This is normally achieved by wireless transmission of the data, possibly over several hops (especially in applications where the sensors are deployed in a hostile or hard-to-access environment). In many cases, the wireless communication results in a major energy expenditure that limits the operational lifetime of the network. Even worse, in multi-hop scenarios, the depletion of the sensors' energy sources (such as batteries) is non-uniform, as nodes that are close to the sink are required to forward all the data traffic and are likely to be the first to run out of energy. Once these sensors fail, other nodes can no longer reach the sink, and the network ceases to operate even though ample energy remains at nodes further away from the sink. This common problem occurs largely independently of the communication protocols used in the network.

In general, the use of Mobile Elements (MEs) [1], [2], [3] can significantly increases the lifetime of the network. Mobile element roams in the network and collects data from sensors via short range communication, the energy cost of which is

considerably lower. Thus, the lifetime of the network increases by avoiding multi-hop communication. The main drawback for this approach is the increased latency of the data collection. Typically, the speed of mobile element can be about 0.1-2 m/s [4], [5], resulting in substantial traveling time for the ME and, correspondingly, delay in gathering the sensors' data. In practice, often the ME tour length is bounded by a predetermined time deadline, either due to timeliness constraints on the sensor data or a limit on the amount of energy available to the ME itself. A possible solution is to employ more than one ME; however, this solution is often impractical due to the high cost of MEs, and may not in fact help at all if some sensors are beyond reach due to ME battery limitations in the first place.

To address this problem, several proposals presented a hybrid approach, which combines multi-hop forwarding with the use of mobile elements. In this approach, mobile element visits subset of the nodes termed as caching points. These caching points stores the data of the nodes that are not included in the tour of the mobile element. Once a mobile element become within the transmission range of a caching point, the caching point transmit its data to the mobile element. By adopting such an approach, the mobile element gather the data of the entire without the need of visiting each node physically. In this direction, we investigate the problem of designing the tours for the mobile elements and the data forwarding trees, with the objective of minimizing the distance between nodes not included in the tour and the tour itself. We propose a heuristic-based solution that creates its solution by partition the network into clusters. The in each cluster a tour will be constructed to satisfy the objective. The results show that our scheme significantly outperforms the best comparable scheme in the literature.

The rest of the paper is organized as follows. Section 2 presents the related work in this research area. Section 3 presents the Problem definition. In Section 4, we present the details of our algorithmic solution. Section 5 presents the evaluation. Finally, Section 6 concludes the paper.

II. RELATED WORK

There have been many proposals in recent literature that studied using mobile element(s) to prolong the lifetime of the network. Based on the categorization given in [3], we review three major approaches.

Authors α σ ρ ω : Al-Hussein Bin Talal University, Jordan.
E-mails: twaissi@ahu.edu.jo, mabuhela@ahu.edu.jo,
alqaralleh@ahu.edu.jo, k.almiani@ahu.edu.jo.
*Author **: Albalqa Applied University, Jordan.
E-mail : a.awajan@bau.edu.jo

In a typical flat-topology network, the nodes around the sink are likely to be the first to die due to having to forward the data traffic from all other sensors. Based on this observation, several proposals [6], [7] have investigated using mobile sink(s) to reduce the energy consumption in the network. By varying the path to the sink(s), the residual energy in the nodes becomes more evenly balanced throughout the network, leading to a higher network's lifetime. However, in order to be effective, this technique requires the sink location (and routes thereto) to change regularly, which places a potentially prohibitive overhead on the nodes due to the frequent re-computation of the routes. Zhao et al [8], [9] investigated the problem of maximizing the overall network utility. Accordingly, they presented two distributed algorithms for data gathering where the mobile sink stays at each anchor point (gathering point) for a period of sojourn time and collects data from nearby sensors via multi-hop communications. They considered the cases where the sojourn time is fixed as well as variable.

In the second approach, mobile elements travel across the network and gather each sensors data via single-hop, short-range communication. In this scenario, the problem of computing the ME tours is exactly the Traveling Salesman Problem (TSP) [10], with the possibility of adding additional constraints to capture the limitations of the nodes buffer size. In [11], [12], [13], [14], [15] several heuristics have been proposed to that effect, so that each sensor is visited before its buffer is full. Although this approach substantially reduces the energy consumption by avoiding multi-hop communications, it incurs a high delay when the network area is large, because of the requirement that the MEs physically visit all sensor nodes.

Finally, the third approach is a hybrid that combines multihop forwarding with data collection by mobile elements. Our work falls into this category. Some earlier works, e.g. [16], [17], [18], assumed the mobile route to be predetermined and were mainly concerned with the timing of transmissions, aiming to minimize the need for in-network caching by timing the transmissions to coincide with the passing of the tour. In [19], [20], the minimum-energy Rendezvous Planning Problem (RPP) is introduced. This problem deals with determining the set of rendezvous points constructing the ME tour. In RPP, the goal is to minimize the Euclidean distance between the source nodes and the tour. Path finding algorithms based on maxflow computations have been considered by [21]. In that work, the authors use a standard maxflow formulation to represent the sensor network. However the problem they consider is finding a path through anywhere in the network area, which does not need to move from a sensor location to another sensor location.

Xu et al [22] also proposed a tour finding algorithm in which nodes away from the caching points

send their data to the caching points using multi-hop communication. The main concept of their algorithm is to find a tour that satisfies the transit constraint such that the depth of the routing trees connected to this tour are bounded by pre-determined variable h . this algorithm starts by setting the value of h to 1 and increasing it gradually, until such a tour is found. By bounding the depth of the routing tress, the algorithm aims to reduce the energy consumption due to multi-hop forwarding. However, important factors in determining the lifetime of the network such as structure of routing tress, the energy level of the nodes, and the distribution of the caching points were not mainly considered in this algorithm. Liang et al [23] investigated similar problem where the depth of the routing trees were bounded by pre-determined variable. The problem presented in this work shares some similarities with the Vehicle Routing Problem (VRP) [24]. Given a fleet of vehicles assigned to a depot, VRP deals with determining the fleet routes to deliver goods from a depot to customers while minimizing the vehicles' total travel cost.

III. PROBLEM DEFINITION

We are given an undirected graph $G = (V, E)$, where V is the set of vertices representing the locations of the sensors in the network, and E is the set of edges that represents the communication network topology, i.e. $(v_i, v_j) \in E$ if and only if v_i and v_j are within each others communication range. In addition, we are given k , that represents the number of tours need to be constructed. The complete graph $G' = (V, E')$, where $E' = V \times V$, represents the possible movements of the mobile elements. Each edge $(v_i, v_j) \in E'$ has a length r_{ij} , which represents the time needed by a mobile element to travel between sensor v_i and v_j . The data of all sensors must be uploaded to a mobile element periodically at least once in L time units, where L is determined from the application requirements and the sensors buffer size. In other words, we assume that each mobile element conducts its tour periodically, with L being a constraint on the maximum tour length. In this paper, for simplicity, we assume that the mobile element travels at constant speed, and that, therefore, the travelling times between sensors (r_{ij}) correspond directly to their respective Euclidean distances; however, this assumption is not essential to our algorithms and can be easily dropped if necessary.

In our problem, we seek to find the k tours, where the length of the tour is bounded by L , such that the number of hops between any node and its caching point is minimized.

IV. TOURS AND FORWARDING TREES

The goal of our algorithm is to find the tours for the given k mobile elements such that distance between the nodes not included in the tours and the tours are

minimized. In this direction, we first partition the network into k partitions. The underlying goal for the partitioning processes is to minimize the distance between nodes belong to the same cluster. By adopting such a process, we aim to minimize the distance between the nodes and the tours in each cluster. Once the clusters are constructed, the tour building step works to create a tour in each cluster with the aim of minimizing the total distance of the routing trees.

a) Clustering Step

The clustering step attempts to find a given number of clusters such that the sum of hop-distances

among nodes belonging to the same cluster is minimized. The clustering process start by selecting a node randomly as a cluster centroid. Once a centroid node is identified it will be added to list termed R . Then, the process works by identifying $k-1$ cluster centroids, where in each iteration, a node is identify as centroid if it has the maximum total hop-distance to all nodes stored in R . Once all k clusters are identified, each node not chosen

	Input: G (topology graph), c (number of clusters to be established)
	Output: a set of clusters
1	Randomly choose a centroid centers and add it to R
2	Do
3	for all nodes in G
4	calculate the distance to all nodes in R (in terms of hop-distance)
5	Select the node with the maximum distance as new centroid
6	Add the select node to R
7	Until k centroid is selected
8	assign each node to its nearest centroid
9	identify each centroid and the nodes assigned to it as a cluster

Figure 1 : The clustering step

a centroid will be assign to its nearest cluster centroid. By adopting such a mechanism, we aim to direct the partitioning to group the nodes into clusters based on their distribution. Figure 1 outlines the process of this step.

b) Caching point identification step

Now, in each cluster subset of nodes will be selected as caching points. These caching points will store the other nodes data and will be used to construct the mobile element tour. In each cluster, this process works by first identifying the center node. The center node is defined as the node that has the minimum total hop-distance to all other nodes in the cluster. Then, it works to constructs Minimum Spanning Tree (MST) rooted at the cluster center node. Once this MST is constructed the process proceeds to traverse the constructed tree using BST mechanism. This traversing stops once the total distance of the visited edges reach $(L \cdot 2/3)$. As we will see in the tours constructing step, this condition depends on the employed TSP algorithm. The last step in this process is to identify the visited nodes in the traversing mechanism as caching points.

c) Constructing the tour

The tour construction phase uses the nodes identified as caching points in the previous step and can be based on any TSP algorithm or heuristic. We use the Christofides approximation algorithm here, as it is known algorithm with $2/3$ approximation factor.

V. EXPERIMENTAL EVALUATION

To evaluate the presented algorithm's performance, we conducted an extensive set of experiments using the J-sim simulator [25]. We used the following parameters:

- (1) The area of the network is $250,000\text{m}^2$.
- (2) The tour length constraint L is set to $0:05 \cdot s \cdot T_L$, where $s = 1 \text{ m/s}$ is the speed of the mobile element, and T_L is the total length of the edges in the minimum spanning tree that connects all nodes, for 500 nodes in the network.
- (3) The starting value of M is chosen to be $0:5 \cdot T_H$, where T_H is the number of hops between the farthest nodes in the network.
- (4) The radio parameters are set according to the MICAz data sheet [26], namely: the radio bandwidth is 250 Kbps, the transmission power is 21 mW, the receiving power is 15 mW, and the initial battery power is 10 Joules. For simplicity, we only account for the radio receiving and transmitting energy.

We are particularly interested in investigating the performance of the presented algorithm in terms of the lifetime of the network and the total distance of the routing trees.

We consider the following deployment scenarios:

1. Uniform density deployment: in this scenario, we assume that the nodes are uniformly deployed in a square area of $500 \times 500\text{m}^2$.
2. Variable density deployment: in this scenario, we divide the network into a 10×10 grid of squares,

where each square is $50 \times 50 \text{m}^2$. We randomly choose 30 of the squares, and in each one of those we fix the node density to be x times the density in the remaining squares. x is a density parameter, which in most experiments (unless mentioned otherwise) is set to $x = 5$.

We compare our algorithm to a modified version of the FFT algorithm [27], we refer to this version as V-FFT. In the original FFT algorithm, a new node is added to the tour based on benefit function. The benefit value of each node depends on the distance between this node and the currently constructed tour as well as the number of nodes it covers. The number of nodes that are covered by any tour node is controlled by the parameter $h \geq 1$. This parameter refers to the maximum number of hops allowed between any two nodes: a tour node can cover any node at most h hops away. Initially, $h = 1$ and in each round the value of h is incremented by one until a tour that satisfies the transit constraint is determined. In the modified version of the FFT algorithm (V-FFT), we fix h to be 0.5 the maximum distance between any two nodes inside any cluster obtained by our heuristic. Then, we start by constructing the first tour. Each selected caching point and the neighbor of this caching points, will be removed from consideration at later stage. This tour will be extended, based on the given cost function, until the constructing tour cannot be extended without violating the transit constraint. Once such a tour is obtained, a new tour will be constructed using the same mechanism.

We evaluate the impact of the number of nodes on the lifetime of the network and the total size of the routing trees each algorithm obtains. Figure 2, Figure 3, Figure 5 and Figure 5 show the results for deployment scenarios. In the

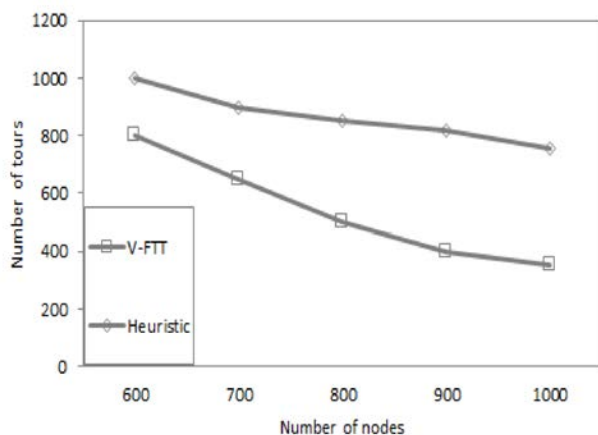


Figure 2 : Network lifetime against the number of nodes, for the uniform density deployment scenario

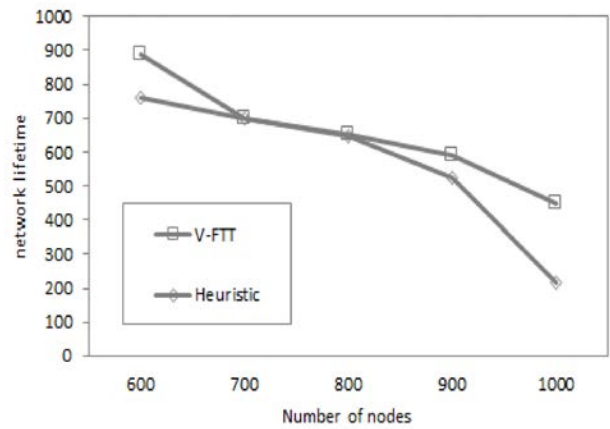


Figure 3 : Network lifetime against the number of nodes, for the variable density deployment scenario

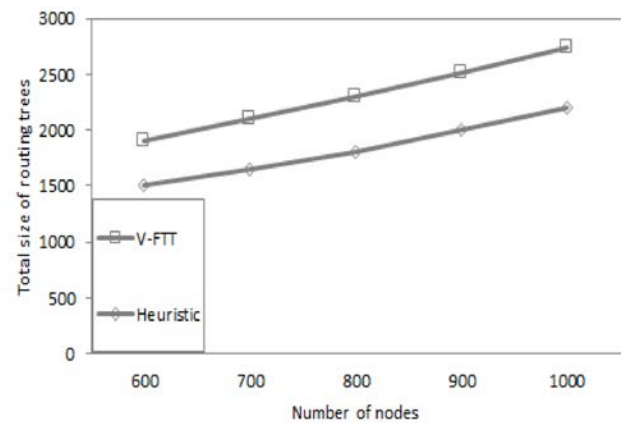


Figure 4 : Total size of routing trees against the number of nodes, for the uniform density deployment scenario

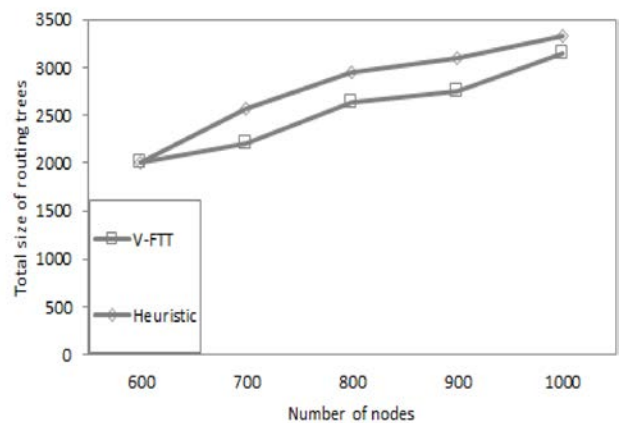


Figure 5 : Total size of routing trees against the number of nodes, for the variable density deployment scenario

uniform deployment scenario, we can see that increasing the number of nodes increases the gap between the algorithm performances. In this deployment

scenario, since we deal with uniformly deployment scenario, the locality of each cluster obtained by the proposed algorithm is expected to be the same. Such behavior results in creating a relatively linear relationship between network lifetime and the number of node. This can be noticed in the results of the routing trees size experiments. In the V-FFT algorithm, the benefit function takes into account the number of nodes covered by the considered nodes as well as the distance between this node and the current constructed tour. And considering the number of covered-nodes as well as the stochastic behavior of such benefit function are the factor behind the seen performance.

In the variable deployment scenario, we can see that increasing the number of nodes results in slightly reducing the gap between the algorithms performances. To understand this behavior, let us discuss the main mechanism behind each algorithm performance. In the proposed algorithm, the obtained clusters is expected to be centralized at the dense grids. This is expected to significantly reduce the lifetime of the network, since in each cluster the tour will be saturated with nodes very closed to each other. This become more obvious while increasing the number of tours. In the V-FFT algorithm, as we mentioned, the benefit function take into consideration the number of neighboring nodes covered by a node as well as the distance between the node and the current constructed tour. In this deployment scenario, considering the number of neighboring nodes during the construction of the tours is expected to improve the V-FFT performance, since it will avoid adding caching node that is very closed to the current constructed tour. These can mainly describe the seen performance.

VI. CONCLUSIONS

In this paper, we consider the problem of designing the mobile elements tours such that total size of the routing trees is minimized. In this work, we present an algorithmic solution that create its solution by partitioning the network, then in each clusters, a tour will be constructed based on the distribution of the nodes.

An interesting open problem would be to consider application scenarios where the data gathering latency requirements vary in the network. For example, some areas in the network need to send data more frequently than others. In this case the tour length constraints would be different for different areas.

REFERENCES RÉFÉRENCES REFERENCIAS

1. J. Butler, "Robotics and Microelectronics: Mobile Robots as Gateways into Wireless Sensor Networks," *Technology@Intel Magazine*, 2003.
2. A. LaMarca, W. Brunette, D. Koizumi, M. Lease, S. Sigurdsson, K. Sikorski, D. Fox, and G. Borriello, "Making sensor networks practical with robots," *Lecture notes in computer science*, pp. 152–166, 2002. [Online]. Available: <http://www.springerlink.com/index/9BUYG5K8BEW05CRX.pdf>
3. E. Ekici, Y. Gu, and D. Bozdag, "Mobility-based communication in wireless sensor networks," *IEEE Communications Magazine*, vol. 44, no. 7, pp. 56–62, 2006.
4. K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. Sukhatme, "Robomote: enabling mobility in sensor networks," in *Proceedings of the 4th international symposium on Information processing in sensor networks (IPSN)*. IEEE Press, 2005, pp. 404–409. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1147685.1147751>
5. R. Pon, M. Batalin, J. Gordon, A. Kansal, D. Liu, M. Rahimi, L. Shirachi, Y. Yu, M. Hansen, W. Kaiser, and Others, "Networked infomechanical systems: a mobile embedded networked sensor platform," in *Proceedings of the 4th international symposium on Information processing in sensor networks*. IEEE Press, 2005, pp. 376–381. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1147685.1147746>
6. S. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations," in *Proceedings of IEEE Globecom*, vol. 1. IEEE, 2003, pp. 377–381.
7. Z. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli, "Exploiting sink mobility for maximizing sensor networks lifetime," in *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS)*, vol. 9, 2005, pp. 03–06. [Online]. Available: <http://doi.ieeecomputersociety.org/10.1109/HICSS.2005.259>
8. M. Zhao and Y. Yang, "Optimization-Based Distributed Algorithms for Mobile Data Gathering in Wireless Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 10, pp. 1464–1477, 2012.
9. —, "Efficient data gathering with mobile collectors and space-division multiple access technique in wireless sensor networks," *IEEE Transactions on Computers*, vol. 60, no. 3, pp. 400–417, 2011.
10. N. Christofides, "Worst-case analysis of a new heuristic for the traveling salesman problem," 1976.
11. A. Somasundara, A. Ramamoorthy, and M. Srivastava, "Mobile element scheduling with dynamic deadlines," *IEEE transactions on Mobile Computing*, vol. 6, no. 4, pp. 395–410, 2007. [Online]. Available: http://www.ece.iastate.edu/~adityar/adi_docs/04116703.pdf
12. A. Somasundara, A. Ramamoorthy, and B. Srivastava, "Mobile Element Scheduling for Efficient Data Collection in Wireless Sensor Networks with Dynamic Deadlines," in *Real-Time Systems Symposium, 2004. Proceedings. 25th IEEE International*. IEEE Press, 2004, pp. 296–305.

13. Y. Gu, D. Bozdag, E. Ekici, F. Ozguner, and C. Lee, "Partitioning based mobile element scheduling in wireless sensor networks," in *Sensor and Ad Hoc Communications and Networks, 2005. IEEE SECON 2005. 2005 Second Annual IEEE Communications Society Conference on*, 2005, pp. 386–395.
14. K. Almi'ani, S. Selvadurai, and A. Viglas, "Periodic Mobile Multi-Gateway Scheduling," in *Proceedings of the Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*. IEEE, 2008, pp. 195–202. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4710981>
15. M. Ma and Y. Yang, "Data gathering in wireless sensor networks with mobile collectors," in *2008 IEEE International Symposium on Parallel and Distributed Processing*. IEEE, Apr. 2008, pp. 1–9. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4536269>
16. A. Somasundara, A. Kansal, D. Jea, D. Estrin, and M. Srivastava, "Controllably mobile infrastructure for low energy embedded networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 8, pp. 958–973, 2006.
17. R. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: Modeling a Three-tier Architecture for Sparse Sensor Networks," in *Proceedings of the First IEEE Workshop on Sensor Network Protocols and Applications*, 2003, pp. 30–41. [Online]. Available: <http://www.comp.nus.edu.sg/~tayyc/6282/forPresentation/SensorNet.pdf>
18. D. Jea, A. Somasundara, and M. Srivastava, "Multiple controlled mobile elements (data mules) for data collection in sensor networks," *Lecture Notes in Computer Science*, vol. 3560, pp. 244–257, 2005. [Online]. Available: <http://www.springerlink.com/index/4617cluarmakx5x.pdf>
19. G. Xing, T. Wang, Z. Xie, and W. Jia, "Rendezvous planning in wireless sensor networks with mobile elements," *IEEE Transactions on Mobile Computing*, vol. 7, no. 12, pp. 1430–1443, Dec. 2008. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4492781>
20. G. Xing, T. Wang, W. Jia, and M. Li, "Rendezvous design algorithms for wireless sensor networks with a mobile base station," in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*. ACM New York, NY, USA, 2008, pp. 231–240. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1374650>
21. M. Ma and Y. Yang, "SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 10, pp. 1476–1488, Oct. 2007. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4302733>
22. Z. Xu, W. Liang, and Y. Xu, "Network Lifetime Maximization in Delay-Tolerant Sensor Networks With a Mobile Sink," in *Proceedings of the 8th IEEE International Conference on Distributed Computing in Sensor Systems*, 2012, pp. 9–16.
23. W. Liang and J. Luo, "Network Lifetime Maximization in Sensor Networks with Multiple Mobile Sinks," in *Proceedings of the 36th IEEE Conference on Local Computer Networks (LCN)*. IEEE, 2011, pp. 350–357.
24. P. Toth and D. Vigo, "The Vehicle Routing Problem," *Society for Industrial & Applied Mathematics (SIAM)*, 2001.
25. J. Hou, L. Kung, N. Li, H. Zhang, W. Chen, H. Tyan, and H. Lim, "J-Sim: A Simulation and emulation environment for wireless sensor networks," *IEEE Wireless Communications Magazine*, vol. 13, no. 4, pp. 104–119, 2006.
26. J. Hill and D. Culler, "Mica: A wireless platform for deeply embedded networks," *IEEE micro*, vol. 22, no. 6, pp. 12–24, 2002.
27. Z. Xu, W. Liang, and Y. Xu, "Network Lifetime Maximization in Delay-Tolerant Sensor Networks With a Mobile Sink," in *Proceedings of the 8th IEEE International Conference on Distributed Computing in Sensor Systems*, 2012, pp. 9–16.