International Journal of Wireless & Mobile Networks (IJWMN) Vol. 6, No. 4, August 2014

DATA-GATHERING IN WIRELESS SENSOR NETWORKS USING MOBILE ELEMENTS

Khaled A. Almi'ani, Ali Al_ghonmein, Khaldun Al-Moghrabi and Muder Almi'ani

Al-Hussein Bin Talal University

ABSTRACT

Using mobile gateway as mechanical data carrier has emerged as a promising approach to prolong the network lifetime and relaying information in partition networks. This mobile gateway periodically travels the network to gather the sensor data, where the gateway tour start and end at the sink. The gateway's tour length must be bounded by pre-defined time constant to avoid buffer overflow. In this paper, we investigate the problem of scheduling the mobile gateway tour in which the tour length satisfies bounding constraints while the sensors lifetime is also increased. We present an algorithmic approach that schedule gateway path by partitioning the network into clusters, so that one node from each cluster must be visited by a mobile gateway. Our experiment results demonstrate that the proposed approach significantly increases the network lifetime compare to networks with static-sink. Also, the quality of the obtained gateway is within 3/2 of the optimal solution tour length.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) have recently witnessed increasing effort to explore applications in various environments [1-3]. A WSN consists of hundred of battery-powered devices deployed in a fly for unattended operations. Because energy is the main concern in WSN, once the network is deployed, recharging the sensors batteries becomes impractical. One of the major energy expenditures is communicating the sensors to deliver their readings to sinks. This communication pattern results in hotspots whereby affected sensors around the sink(s) die earlier than the others as all traffic is funnelled through these sensors.

To address this problem, using mobile gateways has emerged as a promising option [4]. Every mobile gateway travels the network to gather the sensors data and returns to the sink (departure point) to upload the data. By embedding the network with mobile gateways, the sensors data forwarding traffic will be reduced significantly. In addition, the network no longer needs to be connected, since mobile gateways could works as bridges to connect network partitions.

In the literature, many proposals have investigated the beneficial aspects of employing mobile gateways. Based on the gateway's motion strategy, these proposals can be categorized as follows:

- Random motion: Gateways can be mounted on entities that travel the network in an unplanned fashion. For instance in [5, 6], humans or animals act as "data mules" travel the network and opportunistically visit sensors to upload their data. In this motion strategy, the End-to-End delay cannot be bounded as providing a reliable communication is very difficult.
- Fixed motion: Gateways travel the network in a fixed path. For instance in [7], a gateway is mounted on a public bus, which moves in a pre-determined schedule. The

DOI : 10.5121/ijwmn.2014.6409

sensors on the street monitor the bus movements to determine when they can communicate with the gateway.

- Controlled motion: Gateway path is determined based on an objective function. For instance in [8, 9], gateway path is scheduled in which each sensor must be visited before its buffer becomes full.

  In this paper, we investigate the problem of scheduling mobile gateway paths in the controlled motion paradigm. Here, we assumed that sensor network is deployed over a large terrain. Sensors have the same sampling rate with a limited buffer size. The network is then equipped with a mobile gateway that can travel the network to gather the sensors data. The gateway tour start and end at the sink. Sensor data must be uploaded to the sink at a pre-determined rate. This rate is determined based on the sensors buffer size and the end-user interest. Satisfying this frequency constraint results in bounding the gateway tour length, and therefore limits the number of sensors that the gateway can visit in its tour. A problem that naturally crops up is to determine which sensors the gateway must visit in its path, and how the sensors data should be routed and stored in the sensors the gateway will visit during its tour.We refer to this problem as the Mobile Gateway Scheduling with Visiting Deadline (MGS-VD).

We address this problem by dividing it into two steps: (1) *clustering* and (2) *path planning*. The ground concept of these two steps is to partition the network into clusters so that the gateway path can be constructed from only one sensor from each cluster. The sensors that will be involved in the gateway tour will work as cluster heads as they are responsible to store other sensors data. When the gateway enters the physical transmission range for every cluster head, the data stored in the cluster head will be transferred into the gateway's memory. These two steps will work recursively to ensure that number of clusters is maximal and the established tour also satisfies the visiting frequency constraint.

The rest of the paper is organised as follows. Section 2, provides a formal definition of the MGS-VD problem. In Section 3, the related work of this research area is presented. Section 4 presents our algorithmic approach. A number of experiments and the corresponding analysis and results are presented In Section 5. The paper is concluded in Section 6.

2. PROBLEM DEFINITION

Assume an undirected graph $G = \langle V, E, v_s \rangle$ to represent a network topology. Here, V is a set of sensor nodes as vertices, E is a set of edges denoting a communication link between two sensor nodes, and v_s is a unique vertex in the network to represent its base-station/sink. We also assumed that all sensors have the same sampling rate with a limited buffer size. Sensor data must be delivered to the sink once every t time steps. The value of t can be determined as a combination of the end-user interest and sensors overflow restriction. The network is equipped with a mobile gateway to gather network data by traveling the network at a constant speed. Every time the gateway reaches a sensor, it downloads the sensor data into its memory. The mobile gateway tour starts and ends at v_s, while the travelling time of the gateway tour must be bounded by t. The MGS-VD problem can now be defined as follows:

- Partitioning the set V into k disjointed sets $= \{S_1, .., S_k\}$, so that $\bigcup_i S_i = V$, $S_i \cap S_j = \emptyset$, and the vertices in S_i are connected.
- Finding the minimum travelling time tour for the mobile gateway that starts and ends at v_s, and also contains exactly one element from each of the groups S_i, the travelling time of this tour must be less than or equal to t.
- The partitioning S is determined so the average sensor forwarding traffic is minimized

3. RELATED WORK

The use of mobile gateway as data carries has recently been explored in the literature. In [4], an investigation that discussed several advantages of using mobile gateways have been presented, whereas this investigation have mainly focused on communication protocols and reliability. In [10, 11], radio-tagged zebras and whales are used as mobile gateways. These animal-based gateways move randomly in the network terrain and exchange messages opportunistically. In [12], the *message ferries* approach are used to route the data in a sparse network. The main concept of this approach is to determine the mobile gateway path that minimizes the average delay. In [5] , the investigation explored the benefit of employing mobile gateways, which travel the network in parallel straight lines. To reduce delay, sensors away from the gateway path must forward there packets to nearby sensors.

The mobile element scheduling (MES) problem [8, 9] has some similarity with the MGSVD problem. This problem deals with determining the gateway path in which there is no data loss due to sensor node buffer overflow. By adopting the assumption that the sensors must be visited before their buffers become full, MES and MGS-VD share the property that the sensors must be visited based on a deterministic frequency. However, MGS-VD addresses the situation where constructing the gateway tour to visit all sensors without violating the visiting frequency constraint is unachievable.

In situations where the gateway tour can be constructing to include all sensors without violating the visiting frequency constraint, the MGS-VD becomes an extend of the well-known Orienteering problem [13]. This problem is defined as determining the minimum tour length for a vehicle to visit *n*-cities before a pre-determined time deadline. In this situations both of these problems share the property that the visiting must be done before a pre-determined time deadline.

4. ALGORITHMIC SOLUTION

In this section we present an algorithmic approach to handle MGS-VD problem. Our goal is to determine the gateway tour that satisfy the visiting frequency constraint and maximally reduces the energy expenditures due to packets relaying. Here, the ground concept is to partition the network into energy-aware clusters before establishing the gateway's tour to visit the constructed clusters.

Partitioning the network aims to construct clusters that have approximately the same number of nodes. Such construction balances the sensors energy consumption since they will have approximately the same forwarding load. Once the clusters are constructed, the gateway tour will be established to involve one sensor from each cluster, whereas these selected sensors will work as clusters heads, and they will store the other sensors packets.

The partitioning and the tour constructing steps will work recursively to find the maximum possible number of clusters that satisfy the visiting frequency constraint. At the beginning, n number of clusters will be constructed, n is equal to the number of nodes in the networks. Then, in each round, if the tour that connects the cluster heads doesnot satisfy the visiting constraints, the clustering process will be re-triggered and the number of clusters will be divided by two. This process will stop and the gateway path will be obtained when the maximum number of clusters that satisfy the visiting constraint is found.

In the partitioning step, the sensors will be grouped into clusters, where minimizing the distance between nodes belong to the same cluster is the construction criteria. In this context, distance is defined as the number of hops in the shortest path to connect two nodes. Figure 1 outlines the process of this step. This process starts by selecting N_c random nodes as the initial clusters. Once these nodes are identified, each node will be assigned to its nearest cluster. After all nodes are assigned to clusters, the centre node (C_n) for each cluster is determined. In a cluster, the centre node is the node that has the minimum distance to all nodes in this cluster. In situations where these identified nodes do not match the previous nodes, the process will be repeated. This process is terminated to obtain clusters when the identified centre nodes are similar to the nodes identified in the previous iteration.

```
Input: $G$, $N_c$
Clusters ← $\{C_1, .., C_{N_c}\}$
$C_n$ [N$_c$] ← select $N_c$ random nodes from $G$
O__node [N$_c$] ← 0
hops[$N_c$][$n$]
for $i$ ← 1 to $N_c$
    do add C_CENTER [$i$] to $C_i$
Stable ← false
While not stable
    do for $i$ ← 1 to $N_c$
        do for j ← 1 to $n$
            do hops[$C_n$ [i]][j] ← number of hops in the shortest path between i and j
    for each $I \in G$
        do add $I$ to $C_j$ if $C_n$ [j] is the closet $C_n$ to $i$
    O__node ← $C_n$
    for each $C_i$
        do $C_n$ [i]← findC$_n$s($C_i$)
    If O__node = $C_n$ then Stable ← true
```

Figure 1: the clustering process

Once the clusters are identified, the path planning step will be triggered to construct the gateway tour. The path planning step aims to determine the minimum tour length that visits exactly one sensor from each cluster, where this tour start and end at the sink. This description results in considering this problem as a variant of the One-of-a-Set TSP [14], which is also been referred to as the Errand scheduling problem[15]. The One-of-a-Set TSP deals with determining the minimum tour length that visits at least one node from each set, where the Errand scheduling problem deals with determining the best order of performing specific errands, each in which can be performed at different nodes in the graph. The only difference in definition between these two problems and the problem of planning the gateway path is in the number of nodes that the constructed tour can visit from each set. In One-of-a-Set TSP and Errand scheduling problem, the lower bound of this number is one, where there is no restriction about the upper bound. In the problem of planning the gateway path, the number of nodes the tour must include from each set is exactly one. This consideration emphasis the inherited relation between the TSP[16] and the path planning problem investigated in this work. TSP deals with determining the minimum tour length for a salesman to visit n-cities. Therefore, it is reasonable to construct the gateway tour based on the TSP tour.

The process of constructing the tour is divided into two steps, *nodes-identification* and *tour-constructing*. In *nodes-identification*, the identity of the nodes that will participate in the tour will be identified. In *tour-constructing*, the TSP tour for these identified nodes will be constructed. To construct the TSP tour, we employ Christofides algorithm [16], which is well-citied practical algorithm and it has been used as a benchmark whenever a new algorithm is proposed for the TSP. There are many other sophisticated algorithms that outperform Christofides but since we are using a heuristic approach, starting from a simple and very robust algorithm has the advantage of simplifying the implementation. The *nodes-dentification* step aims to select the nodes to form the gateway path. In each round, this step tags the closest node to the partial-constructed tour as a selected node. This process will stop when a node from each cluster is selected. Once these nodes are identified, the *tour-constructing* will construct the TSP tour for the selected nodes using Christofides algorithm.

5. EXPERIMENT EVALUATION

In this section, we have conducted extensive set of experiments to evaluate the performance of the presented approach, namely MG. This validation is performed using J-Sim simulator [17]. We observe the network lifetime and gateway tour length as the evaluation metrics. In the lifetime evaluation, MG performance will be benchmarked against two other schemes; mobile-sink and static-sink. In mobile-sink scheme, the sink will visit each sensor to download its data, where in static-sink scheme; the sensors have to forward their data to reach the sink in a multi-hop fashion. In this evaluation, only the cost of transmitting and receiving the actual data will be counted as the energy expenditure. This consideration aims to emphasis the actual influence of MG on network lifetime, since counting other sources of energy expenditure will trivially shows the benefits of MG on network lifetime. To evaluate the quality of the tour length obtained by MG, we compare its solution against the optimal result obtained by CPLEX .

For the purpose of this simulation, we adopt the two-ray propagation model. With transmission power set to 21 mW, and receiving power set to 15 mW. The data packet size is set to 50 byte and the data rate to 115 Kbps. Sensors are assumed to sample their reading once every second, and they have 5 K-byte storage capability. Unless mentioned otherwise, each simulation is run on a network with 200 node randomly deployed across $200\times200\ m^2$. The gateway is considered to move in the network at 1 *m/s* speed. Each experiment is repeated 10 times and the average is obtained.

5.1 Network Lifetime

Now we evaluate MG performance in term of network lifetime. In this evaluation, we consider the *x percent network lifetime* metric, which is defined as the time until *x* percent of nodes run out of energy. The values of *x* used in this evaluation are 10% and 50%. To simplify the analysis, the value of MG tour length bound is mapped to facilitate controlling the number of obtained clusters. Figures 2 (a) and (b) show the results for this evaluation. From these figures we can see that when the number of clusters is greater than one, MG is able to significantly increase the network lifetime. These figures also clearly show that increasing the number of clusters substantially increases the gap between MG and static-sink scheme. As expected, this behaviour is due to the forwarding cost, which becomes evident in this situation.

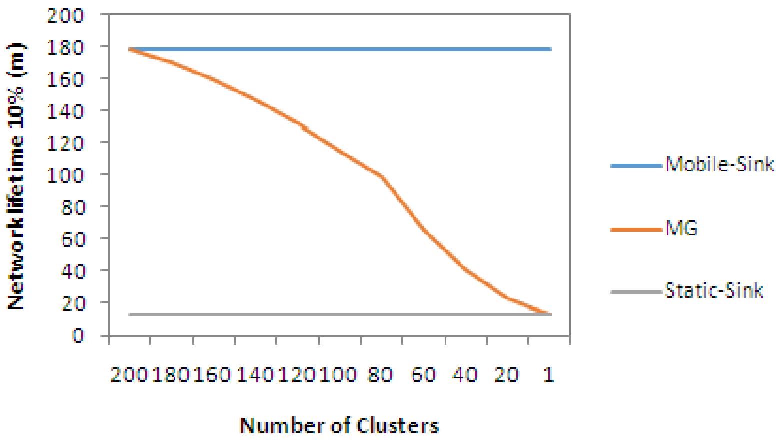

Figure 2(a): Number of clusters against 10% network lifetime

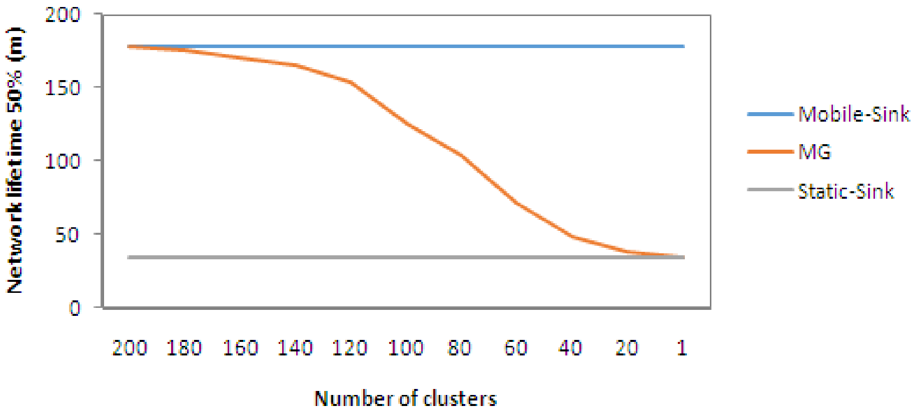

Figure 2(b): Number of clusters against 50 % network lifetime

5.2 Gateway Tour Length

To evaluate MG tour, we compared its quality against the optimal solution, which is obtained using CPLEX. To obtain the optimal solution we modified TSP formulation to incorporate the tour length bound. We also modified the formulation input parameters to have the clusters obtained by MG as an input, where one node from each cluster must be involved in the tour. Due to the NP-hardness of TSP, we limit the maximum number clusters used in this evaluation to 14 clusters and the total number of nodes to 40. Figure 3 depcits the impact of varying the number of clusters on the tour’s travelling time. The result shows that reducing the number of clusters reduces the gap between MGs performance and the optimal solution. This is due to the fact that reducing the number of clusters reduces the valid tours permutation, and therefore increases the probability that MG obtains a near-optimal solution. Also, we can see that MG is within 3/2 factor of the optimal result.

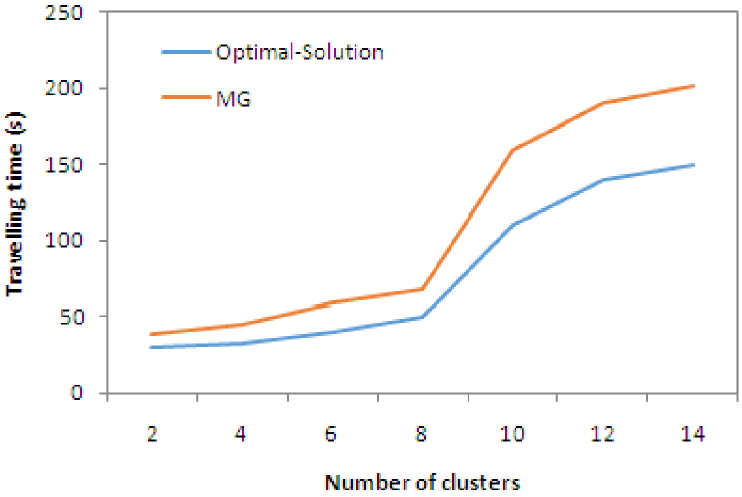

Figure 3: Number of clusters against travelling time

To evalute the impact of the clusters size on the tour travelling time, we ran experiment for a 10 clusters network. Figure 4 depicts the result. The result shows that increasing the number of sensors increases the gap between MG and the optimal solution. This is due to the fact that increasing the number of nodes increases the space of the valid tours.

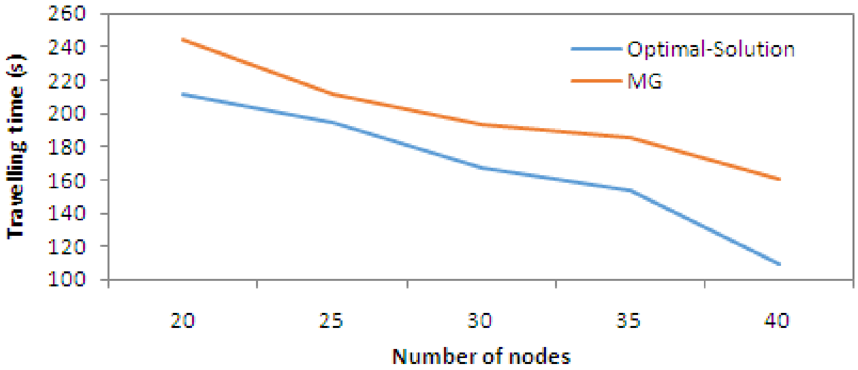

Figure 4: Number of nodes against travelling time

6. CONCLUSION

Using mobile gateway as mechanical data carriers has emerged as a promising approach wherein sensor nodes do not need to form a connected network due to energy restrictions. In this paper, we consider the situations where the travelling time of the gateway tour must be bounded by time constraint to avoid sensors buffer overflow. We presented the problem of scheduling the mobile gateways, so the time bounding is satisfied and sensors lifetime is maximized. To address this problem, we presented an algorithmic approach, which works by partitioning the network into clusters. The gateway tour is then planned to visit one node from each cluster. The experiments showed that in term of network lifetime, the proposed approach significantly increases the network lifetime compared to static-sink scheme. Also, for small-size network, the obtained tour length is within 3/2 of the optimal solution.

REFERENCE

[1] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, "Data Collection, Storage and Retrieval with an Underwater Sensor Network," in ACM Third Int'l Conf. Embedded Networked Sensor Systems (SenSys '05), 2005.

[2] S. Chessa and P. Santi, "Crash Faults Identification in Wireless Sensor Networks," Computer Comm, vol. 25, pp. 1273-1282, 2002.

[3] T. Small and Z. Haas, "The Shared Wireless Infostation Model—A New Ad Hoc Networking Paradigm (or Where There Is a Whale, There Is a Way)," in ACM MobiHoc, 2003.

[4] A. Kansal, A. Somasundara, D. Jea, M. Srivastava, and D. Estrin, "Intelligent Fluid Infrastructure for Embedded Networks," in Proceedings of the 2nd international conference on Mobile systems, applications, and services (MobiSys), 2004.

[5] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling a three-tier architecture for sparse sensor networks," in IEEE Workshop on Sensor Network Protocols and Applications (SNPA), 2003.

[6] S. Jain, R. C. Shah, G. Borriello, W. Brunette, and S. Roy, "Exploiting Mobility for Energy Efficient Data Collection in Sensor Networks," in Modeling and Optimization in Mobile, AdHoc and Wireless Networks (WiOpt), 2004.

[7] A. Chakrabarti, A. Sabharwal, and B. Aazhang, " Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks," in 2nd Int. Workshop on Information Processing in Sensor Networks, ISPN, Berkeley, California, USA, 2003.

[8] A. A. omasundara, A. Ramamoorthy, and M. 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, 2004, pp. 296 - 305

[9] Y. Gu, D. Bozdag, E. Ekici, F. Ozguner, and C.-G. 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.

[10] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D.Rubenstein, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with Zebranet,," in Proc. 10th Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS),, 2002.

[11] T. Small and Z. Haas, "The Shared Wireless Infostation Model—A New Ad Hoc Networking Paradigm (or Where There Is a Whale,There Is a Way)," in Proc. ACM MobiHoc, 2003.

[12] W. Zhao and M. Ammar, "Message ferrying: Proactive routing in highly-partitioned wireless ad hoc networks," in The Ninth IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS), 2003.

[13] B. Golden, L. Levy, and R. Vohra., "The orienteering problem," Naval Research Logistics, pp. 34:307–318, 1987.

[14] J. S. B. Mitchell, Geometric shortest paths and network optimization.

[15] P. Slavík, "The Errand Scheduling Problem," SUNY Buffalo 1997.

[16] N. CHRISTOFIDES, "Worst-case analysis of a new heuristic for the traveling salesman problem," in Symposium on New Directions and Recent Results in Algorithms and Complexity, 1976, pp. Traub, ed. Academic Press, Orlando, Fla., p. 441.

[17] A. Sobeih, W.-P. Chen, J. C. Hou, K. L-C., N. Li, H. Lim, H.-Y. Tyan, and H. Zhang, "J-Sim: A Simulation and Emulation Environment for Wireless Sensor Networks," in IEEE Wireless Communications Magazine, 2005.