2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Decentralized Mobility Models for Data Collection in Wireless Sensor Networks

S. Hanoun, D. Creighton, and S. Nahavandi, *Senior Member, IEEE*

***Abstract*—Controlled mobility in wireless sensor networks provides many benefits towards enhancing the network performance and prolonging its lifetime. Mobile elements, acting as mechanical data carriers, traverse the network collecting data using single-hop communication, instead of the more energy demanding multi-hop routing to the sink. Scaling up from single to multiple mobiles is based more on the mobility models and the coordination methodology rather than increasing the number of mobile elements in the network. This work addresses the problem of designing and coordinating decentralized mobile elements for scheduling data collection in wireless sensor networks, while preserving some performance measures, such as latency and amount of data collected. We propose two mobility models governing the behaviour of the mobile element, where the incoming data collection requests are scheduled to service according to bidding strategies to determine the winner element. Simulations are run to measure the performance of the proposed mobility models subject to the network size and the number of mobile elements.**

I. INTRODUCTION

Wireless sensor networks (WSNs) are generally composed of a large number of sensor nodes, which are densely deployed either inside the phenomenon or very close to it [1]. Sensor nodes capture, encode and transmit relevant information from a designated area to a base station. This creates a comprehensive global-view in which analysis and decision making can be performed. Multi-hop routing is used to deliver the sensed information to the base station, where the sensor's energy is consumed mostly in relaying data, causing nonuniform depletion in the network energy, thus nodes nearby the base station die quickly and base station is disconnected from the network.

Researchers focus on minimizing the energy expenditure and prolonging the lifetime of the network by several means. Designing tiny low-power transceivers, sensing and processing units [1, 2] with efficient hardware power-management strategies [3] addressed the energy problem locally on the sensor level. Deployment strategies addressing network topological lifetime [4], energy aware routing protocols [5] and data fusion using mobile agents [6] are among the topics concerned with minimizing energy globally over the sensor network.

Mobility has been proposed by different patterns as a solution for overcoming the network energy problems related to multi-hop routing. Random Mobility [7], Predictable Mobility [8], and Controlled Mobility [9] are various behaviours in which the mobile element acts towards achieving its collection goal. The concept of data mules has been introduced in [7], where the mules randomly traverse the sensor network and collect data occasionally from sensors when approaching their communication range. Sensors, knowing in advance the trajectory of the mobile element, sleep, saving their energy until the predicted data transfer times come when they become alive and send their data to the mobile element [8]. Mobile element scheduling [9] proposes a schedule for a data collector to visit the sensors collecting their data based on knowing in advance the sensors sampling rates and the rate by which the events in the environment occur. The schedule takes into consideration that visits should be arranged so as no sensor buffer overflow occurs. Two heuristic solutions were proposed. The Earliest Deadline First (EDF) considers the sensor's buffer overflow deadline, while the Minimum Weighted Sum First (MWSF) considers the buffer overflow deadline with the travelling distance between the sensor nodes.

Mobile element scheduling with multiple mobiles [10] extend the work proposed in [9] by considering the heuristics used for the Vehicle Routing Problem (VPR) presented in [11]. The initial schedule is the result of the vehicle routing problem algorithm. Each next scheduled visit is then allocated to the mobile element with the least cost. The number of mobiles used is not fixed or initially predetermined, as the vehicle routing problem algorithm tends to use as many mobiles (vehicles) until all nodes (requests) are scheduled with the best optimum available tour. A centralized scheduling scheme is used for searching for the best mobile or calling for a new one.

In this paper, we are concerned with the decentralized behaviour of multiple mobile elements performing data collection in wireless sensor networks. The mobiles act through a competitive strategy, based on single-item lowest-price sealed-bid auction [12], for wining the incoming collection requests. We reason that the next scheduled visit to the sensor node is not known in advance, as the local estimation and filtration techniques operating on the sensor alter the time taken for the sensor buffer to become full. This depends on the quality of samples sensed, noise in the environment where the sensor is deployed, and configuration

---

Samer Hanoun is a researcher with the Intelligent Systems Research Lab, Deakin University, Geelong, Australia, Victoria, 3217 (phone: +61-3-5227-1352; fax: +61-3-5227-1046; e-mail: sehan@deakin.edu.au).

Dr. Doug Creighton is a research academic with the Intelligent Systems Research Lab, Deakin University, Geelong, Australia, Victoria, 3217 (e-mail: dougc@deakin.edu.au).

Prof. Saeid Nahavandi is the director of the Intelligent Systems Research Lab, Deakin University, Geelong, Australia, Victoria, 3217 (e-mail: nahavand@deakin.edu.au).

978-1-4244-1647-9/08/$25.00 ©2008 IEEE.

of the threshold parameters. Accordingly, in passive sensor networks where the rate of change of the phenomenon monitored is slow, the sensor can sleep after sending a collection request waiting for a mobile collector to arrive for servicing the request, thus we achieve no buffer overflow and the collection requests arrive ordered in time for every sensor whose buffer is full. Two mobility models are proposed to govern the action of each mobile element. The first mobility model *Nearest-to-Last* acts greedy where the second one *Nearest-Neighbour-Next* acts based on a tour optimization heuristic. Performance metrics, such as latency, amount of data collected and overall distance travelled, are used to compare the models. Simulation results demonstrate the behaviour of each of the proposed models according to the number of mobile elements.

This paper is organized as follows. Related work is presented in Section 2, which describes some heuristic used by one of the proposed models. Section 3 provides the system model and a formal definition for the problem. The proposed algorithms are presented in section 4. Assumptions, simulation methodology, and results are discussed in section 5. Tuning performance measures are shown in section 6. Finally, we conclude in section 7, outlining some directions for extending this work further.

II. THE CLARKE-WRIGHT HEURISTIC

One of the best known heuristics for route construction, used in solving the vehicle routing problem, is the Clarke-Wright method [13]. This method, which is also commonly referred to as the savings method, initially starts by assuming that a single vehicle services a single customer. Therefore, for a set of N customers, the savings method assumes that N vehicles are required. The procedure then calculates the savings S_{ij}, namely saving in the distance, that can be obtained by merging customer i and j and servicing them with one single vehicle where the distance between customers i and j is indicated by d_{ij} and d_{io}, d_{oj} represent the distance between customers i and j to the depot respectively. The savings S_{ij} is calculated using the formula:

$$S_{ij} = d_{i0} + d_{0j} - d_{ij} \tag{1}$$

According to Equation 1 the savings is obtained by reducing the number of vehicles required to service customers i and j by one at the expense of increasing the distance to be travelled by the vehicles servicing customers i and j. The calculated savings S_{ij} is sorted in decreasing order and customers i and j that have the highest savings are merged together as long as the capacity restrictions are not violated. The method merges all customers without violating the capacity constraint and stops when no more merging can be done. The Clarke-Wright method requires only the distance metric between the customers to construct the routes that are constructed in a sequential manner.

This heuristic tends to solve the problem in a centralized way. That is, an algorithm is used to assign customers to a single vehicle, until the vehicle cannot accept any more customers due to the capacity or distance restrictions. At this stage, a second vehicle is called and customers from the available pool are assigned to it until no more customers can be added to it. This continues until all customers are assigned to the vehicles. Once all customers have been assigned to the vehicles optimization methods can be used to improve the solution.

Rather than solving the VRP, we use this heuristic to construct the route that the mobile element will follow to traverse the sensor network and collect data as it approaches the intended sensors. The heuristic is used to minimize the overall travelling distance for the mobile element and to construct the optimum tour. As our models work in a distributed manner, there is no need for a centralized algorithm to assign sensors to mobile elements. This is achieved through competition among mobile elements in which the bid value determines the most feasible tour for each sensor. As the collection requests arrive in order by the sensors requesting the collection, the route is constructed dynamically and in a distributed manner.

III. SYSTEM MODEL AND DEFINITION

This section describes the system model and all assumptions made. A formal definition for the problem is presented to give an insight into the problem objectives.

A. System Model

We model our system based on the following:

- Sensor nodes remain stationary and are deployed uniformly at random in the environment where the phenomenon of interest is to be monitored.
- Each sensor node has a deployment location (X_s, Y_s), buffer size of K_s bytes, radio communication range R_s in meters and sensing interval T_s in seconds.
- All sensors nodes have the same radio communication range R_s and different sensing intervals to demonstrate different sampling frequencies.
- The time required for the sensor's buffer to be full is $\geq K_s * T_s$, where the next time the buffer becomes full is unknown and differs from time to time. This shows that some sensors may need to be visited more frequently than others. The quality estimation threshold over the sensed samples is modelled by Bernoulli process to estimate whether the sensed sample of a quality to be stored or otherwise discarded.
- Once the buffer is full, the sensor node sends a collection request and sleeps waiting for the collector arrival.
- While sleeping the sensor does not sense any new samples but it relay collection requests send by others.
- The sensor network is modelled as a graph $G(N,E)$, where N is the set of all nodes in the environment and E is the set of all links (i,j) where j is in the radio communication range of i. Node j is considered a neighbour to node i if node j is in node i communication range and vice versa.
- Each mobile element has a current position (X_m, Y_m), radio communication range R_m in meters and can move by a

speed of S_m in m/sec.

- All mobile elements are assumed to start at a central position in the environment (X_c, Y_c), have reasonably high amount of energy that can last beyond the network lifetime and their buffer size can accommodate all data generated during the network operational time.

When any sensor node sends a collection request it uses the network resources to convey this request to reach one of the mobile collectors. The first mobile collector receiving the request acts as the auctioneer. Communicating the request to other mobiles for placing their bids is done through the mobile elements' own network with no extra overhead on the sensor network resources.

B. *Formal Definition*

We present the problem formulation to show formally the main objectives and constraints. Let n be the number of sensor nodes where $n \in Z^+$, and m be the number of mobile elements where $m \in Z^+$.

Parameters:

- S_i: Count of requests generated by sensor i, where $i \in \{1..n\}$
- D_{ij}: Count of data collected requests by mobile j from sensor i, where $i \in \{1..n\}$ and $j \in \{1..m\}$
- C_{ij}: Cost of servicing sensor i by mobile j, where $i \in \{1..n\}$ and $j \in \{1..m\}$

Variables:

- $x_{ij} = 1$ if sensor i is serviced on mobile j tour; 0 otherwise, where $i \in \{1..n\}, j \in \{1..m\}$.

Objectives:

$$\max \sum_{i=1}^{n} \sum_{j=1}^{m} D_{ij} * x_{ij} \tag{2}$$

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} * x_{ij} \tag{3}$$

While Equation 2 requires maximizing the amount of data collected, Equation 3 on the other hand tries to minimize the overall distance travelled by each mobile element. Both of these combined produce the ultimate case where more amount of data can be collected while travelling as minimum as possible to overcome high energy expenditure for the mobile elements.

Constraints:

The count of data items collected by all mobile elements should not exceed the original count of requests generated by node i. Therefore,

$$\sum_{j=1}^{m} D_{ij} \le S_i \quad \forall i = 1..n \tag{4}$$

The maximum number of sensor nodes that can be serviced by all mobile elements is n. So,

$$\sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} \le n \tag{5}$$

Finding the existence of a feasible set of x_{ij}s will result in a tour; each mobile element can follow to maximize the amount of data collected and minimize the overall distance travelled. As the requests arrive dynamically and not known ahead, it is not possible to find a static schedule to follow and the solution should be generated online based on some heuristic, trying to satisfy Equations 2 and 3.

IV. MOBILITY MODELS

The main goal of each mobile element is to construct a route to visit the sensor nodes requiring data collection. The heuristic used by each mobile element determines its bid where the element with minimum bid is the winner. This guarantees that the next request to arrive is serviced by the best mobile element according to the current positions of all mobiles. The heuristic tries to meet the objectives presented previously in Equations 2 and 3.

A. *Nearest-to-Last*

This mobility model uses a time oriented nearest-neighbor greedy heuristic. The incoming request is placed at the end of the requests list where the new tour cost is calculated based on that insertion point. The bid is evaluated as the difference in the cost between the current route and the new route after inserting the incoming request. The winner is the one with the minimum bid.

```
ALGORITHM: Nearest to Last
Input: Collection Request $Q$ with fields $\{ID, X, Y\}$
Output: Bid $B$
Define: m: mobile element current position;
$D_{u \to v} = \sqrt{(X_u - X_v)^2 + (Y_u - Y_v)^2}$
Body:
1. If Requests_List is Empty then $B = D_{m \to Q}$
2. ELSE
3.     $Q_T$ = Requests_List_Tail
4.     $B = D_{Q \to Q_T}$
END
```

The distance from the last existing request to the current incoming request determines the cost added to the current tour which is considered as the bid. It is obvious that the algorithm has one insertion point which is the tail of the requests list. As the heuristic acts greedy, the back-and-forth movements between far away sensor nodes are not avoided, which leads to high distance travelled regarding the amount of data collected.

B. *Nearest-Neighbour-Next*

As the main objective is to optimize the mobile element tour, this model uses the heuristic described in Section 2 to find an insertion point for the incoming request that minimizes the extra travelled distance. The route is constructed by adding the new request to the nearest existing request. The bid is evaluated as the difference in the cost between the current route and the new route after inserting the incoming request in the best feasible insertion point.

```
ALGORITHM: Nearest-Neighbour Next
Input: Collection Request $Q$ with fields $\{ID, X, Y\}$
Output: Bid $B$
Initialize: n = (Requests_List_ Size + 1), Cost [1 ... n]
            Insertion_Position = 0
Define: m: mobile element current position;
$D_{u\to v} = \sqrt{(X_u - X_v)^2 + (Y_u - Y_v)^2}$
Body:
If Requests_List is Empty then $B = D_{m\to Q}$
ELSE
    $Q_H$ = Requests_List_Head
    $Cost[1] = D_{m\to Q} + D_{Q\to Q_H} - D_{m\to Q_H}$
    Repeat the following, $\forall i$ in $\{1, 2, 3 \ldots n-1\}$
        $Q_i$ = Request at Position ($i$)
        $Q_{i+1}$ = Request at Position ($i+1$)
        $Cost[i+1] = D_{Q_i\to Q} + D_{Q\to Q_{i+1}} - D_{Q_i\to Q_{i+1}}$
    $Q_T$ = Requests_List_Tail, $Cost[n] = D_{Q_T\to Q}$
    $B = \min_{i=1..n} \{Cost[i]\}$, Insertion_Position = $i$
END
```

The algorithm tries to fit in the incoming request at the most feasible point within the current tour. This helps in minimizing the overall travelling distance while maximizing the amount of data collected along. Globally, as the winner is the one with the minimum bid, the incoming request is placed on the most feasible tour, at the most feasible position in that tour.

V. SIMULATION METHODOLOGY AND RESULTS

This section presents the evaluation of the algorithms presented in the previous section. The various parameters used are for real world systems, although these were evaluated in simulation. The PowerBot [14] can move with a maximum speed of 2.1 m/s, equipped with a wireless radio modem, and a GPS for localization. The mica2 motes [15] are commonly used wireless sensor nodes. We consider sampling a phenomenon occurring in a square area of 200mx200m. The sensor nodes sampling interval varies from 1 second to 10 seconds. The sensor node buffer size is 1kB. We do not know ahead when the sensor's buffer will be full.

A. Simulation Methodology

We consider the following parameters:

- *Sensor Network Deployment*. We consider 100 sensor nodes deployed uniformly at random in a square area of 200m x 200m.
- *Sensor Network Topology*. A radio communication range of 25m is used to achieve full connectivity and no sensor nodes are out of range and/or isolated.
- *Mobile Element Parameters*. Initially the mobile element is localized at the centre of the 200m x 200m area. Each mobile element moves by a fixed speed of 1m/s, and has a communication range of 50m. It is assumed that the mobile element achieves communication with the sensor node when it is in the sensor's node range.
- *Simulation Time*. The models are tested and simulated for 20,000 seconds.
- All results are averaged over 20 different topologies with different sampling rates for each sensor.

For the purpose of evaluation, we consider the following performance metrics to compare the proposed models:

- *Latency*. We define latency as the time taken between the send of the collection request and the time of arrival of the mobile collector, indicated by the sleep time. This is averaged over all requests per sensor node, across all nodes.
- Another metric is the *collection ratio*, which is the amount of *data collected* to the amount of *requests generated*. This gives an indication of how fast the requests are been serviced to the rate of generation in the network.
- The *distance travelled* is used to show how much effort the mobile team exert. This can be used as a measure for energy expenditure. Minimizing this measure while maximizing the data collected shows to what extent the used model is efficient.

B. Discussion

The performance results for the proposed models are presented and discussed in this section. For the ease of proper evaluation and comparison we fixed the number of mobiles (*mobiles*). We use four combinations of (*mobiles*, *mobility model*) in simulations: (5, Nearest-to-Last), (5, Nearest-Neighbour-Next), (10, Nearest-to-Last) and (10, Nearest-Neighbour-Next). We study the effect of the team size on the performance metrics described earlier for both proposed models.

Fig. 1 shows the impact of the mobility models on the latency in the network for both cases of 5, and 10 mobile elements. It appears that the Nearest-Neighbour-Next mobility model achieves better latency than the Nearest-to-Last model even when the number of mobile elements has increased. The performance impact due to the Nearest-Neighbour-Next model is better and the network activity is maximized.

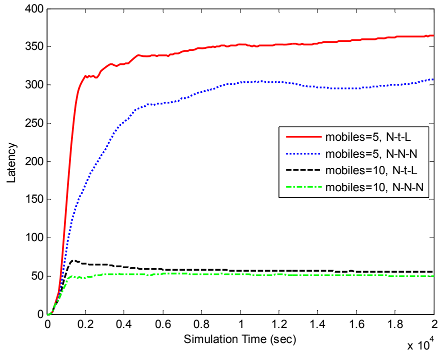

Fig 1. Latency in the network

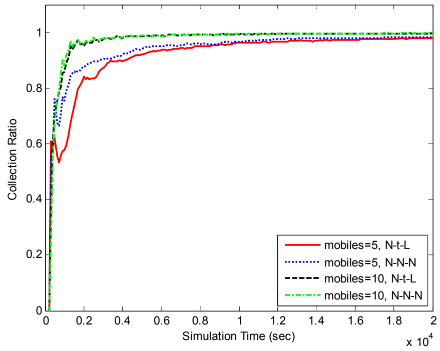

Fig 2. Collection Ratio

Fig. 2 presents the collection ratio rate. While both models produce high collection ratio, the increase in the number of mobile elements, achieves a fast service for the requests generated in the network. The collection ratio tending to 1.0 illustrates that all generated requests are considered for collection as fast as possible, thus decreasing the latency in the network.

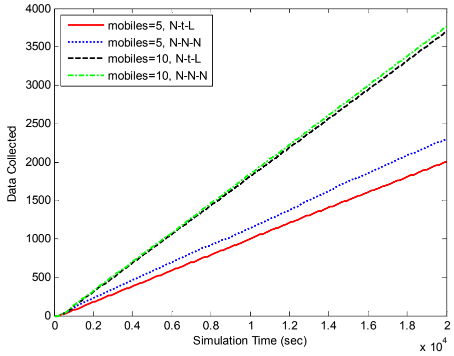

Fig 3. Data Collected

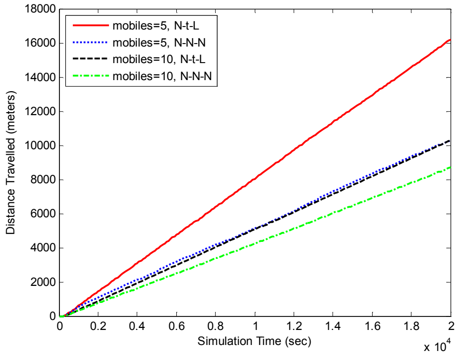

Fig 4. Distance Travelled

The amount of data collected relative to the distance travelled for the Nearest-to-Last model is lower than that for Nearest-Neighbour-Next model as shown by Fig 3 and Fig. 4. As the number of mobile elements increase to 10, it is shown that the amount of data collected is equal while the distance travelled for Nearest-Neighbour-Next model is better. While the main concern is the amount of data collected, the latency stands as a critical issue for the operating the network.

VI. TUNING THE PERFORMANCE

This section describes the procedure we followed for tuning the performance of the Nearest-Neighbour-Next model. The previous simulation results show the case of an open requests list where each mobile element can compete on as many requests as possible without any limitation on the number of requests to be serviced in the future or along its tour.

Fig. 5 shows an example for demonstrating the effect of the size of the requests list of the mobile element on the performance metrics presented in the previous section. Mobile element M1 wins the request issued by sensor S4 based on the heuristic used by the Nearest-Neighbour-Next model. The latency value as M1 arrives at S4 is 120. M2 arrives and services S5 and will keep waiting for new requests to arrive to compete on. In the case that M2 wins S4, the latency value would be 90, thus minimizing it while sacrificing some of the distance travelled. Limiting the requests list size can be used to overcome such situations avoiding the case where one, or few mobiles, are winning all the time and the others are not, which corresponds to long tours for those mobiles, and not utilizing the others in an efficient way.

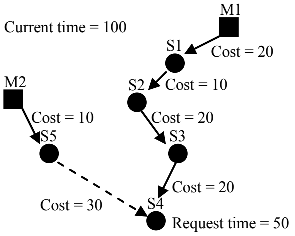

Fig. 5. Nearest-Neighbour-Next example

For the ease of tuning and selecting the best list size that will minimize the overall latency, maximize the amount of data collected and thus increase the number of requests that can be serviced, we run our experiments limiting the requests list size not to grow up over a certain size and compare the performance measures values to determine the best case. Limiting the requests list size may arise of missing some requests as none of the mobile elements can bid on those requests as their list limit prevent that. A *Miss Ratio* measure is introduced to guide in the selection of the best size threshold for the requests list. We calculate it as the ratio of the number of requests missing the service to the total number of requests generated in the network.

Fig. 6 illustrates the latency for different sizes of the requests list normalized to the latency value of the open list size. The latency decreases 42% for a list with a size between 10 and 15, while maintaining a zero miss ratio as shown in Fig. 7. Tuning the list size decreased the latency and increased the amount of data collected by 33% as shown in Fig. 8. However, it increased the overall distance travelled by more than 50% as shown in Fig. 9. We reason that the selection criteria is more related to the network operator objectives, whether to minimize the latency and maximize the data collected or to minimize the distance travelled by the mobile elements while sacrificing the latency and the data collected amount on the other hand.

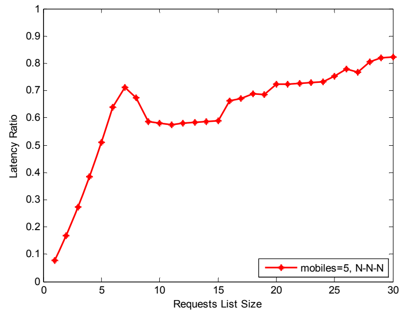

Fig 6. Latency Ratio

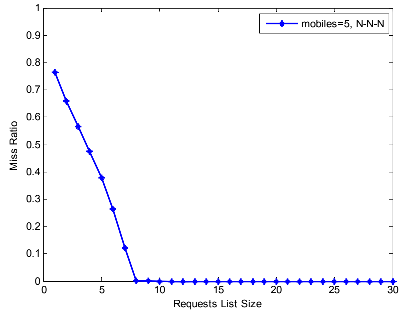

Fig 7. Miss Ratio

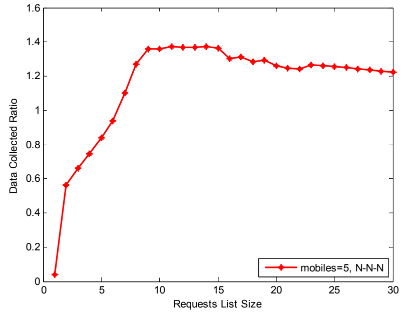

Fig 8. Data Collected Ratio

VII. CONCLUSION AND FUTURE WORK

Using multiple mobile elements for data gathering in sensor networks continues to prove it benefits towards enhancing the network performance and overcoming some problems regarding data collection. We presented multiple mobile elements interacting and competing for visiting the sensor nodes collecting their data buffers. The deadlines were relaxed by making the sensor nodes sleep waiting for the data collector arrival. The impact of the mobility model on the performance measures is shown, in addition to the number of mobiles servicing the network. The presented heuristics are compared and the performance is showed to enhance when tuning some parameters. The mobility models and the heuristics need to be modified to eliminate the need for the sensor node to sleep. The overall energy expenditure is needed to show the advantages of adding the mobile elements to the network.

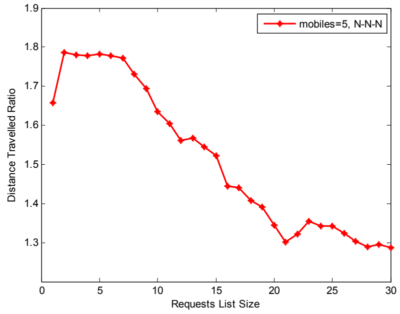

Fig 9. Distance Travelled Ratio

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A Survey on Sensor Networks,” IEEE Communications Magazine, Vol. 40, Issue 8, pp. 102-114, August 2002.

[2] M.A.M. Vieira, C.N. Coelho, D.C., Jr. da Silva, and J.M. da Mata, “Survey on Wireless Sensor Network Devices,” Proc. IEEE Emerging Technologies and Factory Automation Conference (ETFA '03), September 2003.

[3] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava, “Energy-aware Wireless Microsensor Networks,” IEEE Signal Processing Magazine, Vol. 19, Issue 2, pp. 40-50, March 2002.

[4] X. Bai, S. Kumar, Z. Yun, D. Xuan, and T.H. Lai, “Deploying Wireless Sensors to Achieve both Coverage and Connectivity,” Proc. ACM MobiHoc'06, Florence, Italy, May 2006.

[5] J. N. Al-Karaki, and A. E. Kamal, “Routing Techniques in Wireless Sensor Networks: A Survey,” IEEE Wireless Communications, Vol. 11, Issue 6, pp. 6-28, December 2004.

[6] L. Tong, Q. Zhao and S. Adireddy, “Sensor Networks with Mobile Agents,” Proc. IEEE MILCOM'03, Boston, MA, October 2003.

[7] R.C. Shah, S. Roy, S. Jain, and W. Brunette, “Data MULEs: Modeling a Three-Tier Architecture for Sparse Sensor Networks,” Proc. IEEE Workshop Sensor Network Protocols and Applications (SNPA ’03), May 2003.

[8] A. Chakrabarti, A. Sabharwal, and B. Aazhang, “Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks,” Proc. 2nd International Workshop on Information Processing in Sensor Networks, 2003.

[9] A. A. Somasundara, A. Ramamoorthy, and M. B. Srivastava, “Mobile Element Scheduling for Efficient Data Collection in Wireless Sensor Networks with Dynamic Deadlines,” Proc. 25th IEEE International Real-Time System Symposium, 2004.

[10] A.A. Somasundara, A. Ramamoorthy, and M.B. Srivastiva, “Mobile Element Scheduling with Dynamic Deadlines”, IEEE Transactions on Mobile Computing, Vol. 6, Issue 4, pp. 395-410, April 2007.

[11] M. Solomon, “Algorithms for the Vehicle Routing and Scheduling Problem with Time Window Constraints,” Operations Research, Vol. 35, Issue 2, 1987.

[12] E. Wolfstetter, “Auctions: An introduction,” Journal of Economic Surveys, Vol. 10, Issue 4, pp. 367-420, December 1996.

[13] G. Clarke and J. V. Wright, “Scheduling of Vehicles from a Central Depot to a Number of Delivery Points,” Operation Research. Vol. 12, Issue 4, pp. 568-581, 1964.

[14] “PowerBot: The High-Agility High-Payload Robot”, http://robots.mobilerobots.com/, January 2007.

[15] “MICA2: Wireless Measurement System”, Datasheet, http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf, 2007.