

Scheduling Algorithms for Data Collection in Wireless Sensor Networks - A Survey

Rajeev Paulus, Alok Singh, Aditi Agrawal, Manvendr

Abstract - Wireless Sensor Networks (WSNs) face unavoidable challenges in the areas of maintaining connectivity, network lifetime and energy efficiency in its implementation. Recently, several WSN architectures based on mobile elements (MEs) have been proposed which can be leveraged to overcome these problems. Mobility of the mobile elements is controlled by the protocols & algorithms. We surveyed on different aspect data collection algorithms for WSNs such as Partitioning Based Scheduling (PBS) algorithm, Multihop Route to Mobile Element (MRME) algorithm, etc. and found the method scheduler of mobile element in WSNs for data collection. We have highlighted their objective, feature, complexity in this paper.

Keywords - Mobility, Mobile elements, TSP, Wireless Sensor Network.

I. INTRODUCTION

WSN is a collection of sensor node able to sense their environment, collect and process various data communicate among each other. The use of WSN have been proposed for critical applications such as battlefield surveillance, habitat monitoring, traffic monitoring, nuclear, chemical and biological attack detection [1],[2]. In such applications, it is often necessary to collect the data accumulated by each sensor node for processing. Data collection is one of the most important applications of wireless sensor networks [3]. But, there are some numerous challenges faced while designing WSNs, maintaining the connectivity, reducing energy consumption and maximizing the network life time stand out as critical conditions.

The mobile elements are used to overcome these challenges like balance energy consumption in wireless sensor networks. The mobile devices can also be used as an orthogonal method to address the connectivity and lifetime problems in WSNs [4]. With communication devices on mobile platforms,

the connectivity, network lifetime and energy efficiency problems can be addressed as follows:

- *Connectivity*: Mobile devices can be used to carry information between the distributed sensor nodes in the network.
- *Network lifetime*: The energy consumption in the sensor nodes is reduced considerably by transmitting information through the mobile devices which reduces multi hop communication between nodes.
- *Energy efficiency*: Information carried in mobile devices can reduce the energy consumption of sensor nodes by reducing multi hop communication.

Mobile Elements can be deliberately built into the system to improve the lifetime of the network and act as mechanical carriers of data, which move around in the sensing field. The mobile element, whose mobility is controlled, visits the nodes to collect their data before their buffers are full.

Thus the scheduling of mobile elements, i.e., how they traverse through the sensing field and when they collect data from which sensor, is of ultimate importance and has attracted increasing attention from the research community. This scheduling of mobile element is carried by algorithms which have been explained in this paper. The first one is the Earliest Deadline First (EDF) algorithm where the node with the closest deadline is visited first [11],[12]. To improve EDF, the second algorithm, EDF with k-lookahead, is proposed [12]. Instead of visiting a node whose deadline is the earliest, this algorithm considers the k! permutations of the k nodes with smallest deadlines, and chooses the next node which leads to the earliest finish time. Consequently, the EDF with k-look ahead algorithm performs better than pure EDF. The third algorithm is the Minimum Weight Sum First (MWSF) algorithm [12], which accounts for the weights of deadlines as well as distances between nodes in determining the visiting schedule. After then the PBS

algorithm is explained to schedule the ME visits to the sensor node to collect routing messages without buffer overflow[13]. Finally, MRME algorithm is explained that extends PBS to deliver urgent messages (like traffic jams, accidents *etc.*) to MEs within specific delay bounds [14]. The rest of the paper is organized as follows: We have presented mobility in WSNs in Section 2. In Section 3, we have introduced algorithms such as Earliest Deadline First (EDF), EDF with k-lookahead, Minimum Weight Sum First (MWSF), Partitioning Based Scheduling (PBS), and Multihop Route to Mobile Element (MRME) algorithm. Conclusion is presented in Section 4.

II. MOBILITY FORME INWSN

Mobile Elements can be deliberately built into the system to improve the lifetime of the network and act as mechanical carriers of data, which move around in the sensing field. The mobile element, whose mobility is controlled, visits the nodes to collect their data before their buffers are full. Various type of mobility have been considered for data collection in WSNs. These can be categorized with respect to the property of sink mobility as well as the wireless communication methods for data transfer:

- Mobile Base Station-based solution: An MBS is a mobile sink that changes its position during operation time. Data generated by sensors are relayed to MBS without long term buffering [5],[6].
- Mobile Data Collector-based solution: An MDC is a mobile sink that visits sensors. Data are buffered at source sensors until the MDC visits the sensors and downloads the information over a single-hop wireless transmission [7], [8], [9].

The MDC based proposal can be broadly classified as random, predictable or controlled:

Random Mobility: MDCs move in random patterns as proposed in [7].

Predictable Mobility: An MDC's movement pattern is known, as presented in [8].

Controlled Mobility: An MDC's movement is actively controlled in real time, as proposed in [9].

- Rendezvous-based solution: Rendezvous based solutions are hybrid solutions where sensor data is sent to rendezvous points close to the path of

mobile devices. Data are buffered at rendezvous points until they are downloaded by mobile devices, [10]. Rendezvous based solution adapts both the properties of MBS and MDC based solutions.

III. SCHEDULING ALGORITHMS

In this section, we have presented a survey of the existing scheduling algorithms for the Mobile Element (ME) in Wireless Sensor Networks (WSN). The Earliest Deadline First (EDF) algorithm exhibits the characteristics of scheduling the ME starting with the least deadline nodes [11],[12]. EDF with K-lookahead algorithm schedules with reference to the k value [6]. The schedules are formulated with reference to the look-up of k nodes. The Minimum Weighted Sum First (MWSF) algorithm schedules are done based upon a weight calculated on the deadline of the sensor nodes [12]. The partitioning Based Scheduling (PBS) algorithm that computes periodic paths of an ME to avoid sensor data loss at low ME speeds [13]. The Multihop Route to Mobile Element (MRME) algorithm that extends PBS to deliver urgent messages to MEs within specific delay bounds [14]. In the scheduling process, the ME's visit is scheduled only for these boundary-near nodes as the aggregated data is available in this boundary-near nodes. Rendezvous Points finds the optimal RPs when MEs move along the data routing tree in order to perform data collection.

3.1 Earliest Deadline First (EDF)

The EDF algorithm is the simplest of all ME Scheduling algorithms. In this algorithm, the node which has the least deadline value is visited first, irrespective of all other constraints. The advantage of this is that, it is very easy to be implemented in a sensor network and less complexity is involved. But one obvious shortcoming of this algorithm is that it does not take into account the cost values and relies only on deadlines. We are given the following notation for explain the EDF algorithm:

- A fully connected graph of n nodes: node[1..n]
- A matrix cost[1..n][1..n] that denotes the time taken to go from one node to another
- A vector that contains buffer overflow times, overflow_time[1..n].

- A starting $node_0$.

The Earliest Deadline First (EDF) algorithm can be summarized below.

EDF Algorithm

1. Input:
 $cost[1..n][1..n], overflow_time[1..n], start_node$
2. Initialize:
 $current_time = 0, current_node = start_node,$
 $deadline[1..n] = overflow_time[1..n]$
3. Main: Repeat the following
 - i. Choose the node i $current_node$ whose deadline is closest
 - ii. If $deadline[i] < current_time + cost[current_node][i]$
-Declare failure and stop
 - iii. Else
- $current_time + = cost[current_node][i]$
- $current_node = i$
- $deadline[i] = current_time + overflow_time[i]$ END

The drawback of EDF algorithm is that when involved the higher cost, this leads to several failed nodes which could have been avoided.

3.2 EDF with k-lookahead

The EDF-k look ahead algorithm is an extension of EDF algorithm. In this, the ME does not visit the node where the deadline expires immediately, instead the deadlines of the next k nodes is analyzed, to know if all the nodes satisfy their deadlines. Only if all the deadlines are satisfied, then the ME goes to the first node in the selected path. The EDF with k-lookahead algorithm is presented below-

EDF with k-look ahead

1. Input:
 $k, cost[1..n][1..n], overflow_time[1..n], start_node$
2. Initialize:
 $Current_time, current_node, deadline[1..n]$ as EDF.
3. Main: Repeat the following
 - i. Sort $deadline[1..n]$ in increasing order.
 - ii. Using the first k entries:
 - Find an ordering of these k entries so that
 - (a) None of the k nodes miss their deadlines in the next k steps,

- (b) Arrival time at the node at (+) entry is minimum, and
- (c) The first node in the resulting permutation is not the current node.

-If none exists, declare failure and stop.

- iii. Let the first node in the ordering found be i .

- $current_time + = cost[current_node][i]$
- $current_node = i$
- $deadline[i] = current_time + overflow_time[i]$

3.3 Minimum Weighted Sum First (MWSF)

The Minimum Weighted Sum First algorithm is different from the normal EDF algorithm because in addition to considering the deadlines, it also takes into account the cost of each edge in the network.

MWSF Algorithm

1. Input:
Weight $\alpha \in [0, 1],$
 $cost[1..n][1..n], overflow_time[1..n], Start_node$
2. Initialize
 $current_time, current_node, deadline[1..n]$ as EDF algorithm.
3. Main: Repeat the following
 - a. $\forall i$, calculate, $weighted_sum[i] = \alpha * (deadline[i] - Current_time) + (1 - \alpha) * cost[current_node][i]$
 - b. Choose the node i whose $weighted_sum[i]$ is minimum
 - if $deadline[i] < current_time + cost[current_node][i]$
*declare failure and stop
 - Else
* $current_time + = cost[current_node][i]$
* $current_node = i$
* $deadline[i] = current_time + overflow_time[i]$ END

Table 2. Effect of α values on Minimum Weighted Sum First heuristic

$\alpha \in [0, 1]$	Result
1	Weight to deadlines only. Same as EDF.
0	Weight to cost only. Results in back and forth motion between closest-distance nodes. All other nodes miss their deadlines.
small	Higher priority to closer to nodes
big	Higher priority to closer deadlines

Table 2 shows the effect of different values. When $\alpha < 0.5$, the scheduling is performed predominantly based on the cost. When $\alpha > 0.5$, it relies more on the deadlines. With a single ME, Minimum Weighted Sum First is found to perform better than EDF with $\alpha \leq 0.1$. The MWSF algorithm performs the best among the EDF and EDF with k-lookahead proposed algorithms.

3.4 Partitioning Based Scheduling (PBS)

Partitioning-Based Scheduling (PBS) algorithm is designed to solve the Mobile Element Scheduling problem, which aims to schedule the visits of the mobile element to each sensor to avoid data loss due to sensor buffer overflow. PBS tackles the MES problem in two phases:

- Partitioning phase
- Scheduling phase

In the partitioning phase nodes are grouped in such a way that nodes in a group have similar buffer overflow times and are closely located.

In the scheduling phase the trajectories inside groups are computed, and then concatenated to form a complete trajectory.

Commonly used symbols and notation for PBS & MRME algorithms are follows:

n_i Sensor node i

o_i Overflow time of

$W_{i,j}$ Geographic distance between node and

B_i Bin i of partitioning by overflow time

$B_{i,j}$ j^{th} sub-bin of Bin i

$N_{i,d}$ The set of distance- d neighbors of

The maximum delay that can be tolerated for urgent messages

D_i The worst case delay for urgent messages generated on node

The PBS algorithm can be presented as below:

PBS Algorithm

1. Partition the nodes $\{ \}$ into M bins according to the buffer overflow times $\{ \}$.
2. Geographically partition each bin into 2 sub bin using partitioning algorithm.
3. Calculated a TSP path for each sub bin.
4. Concatenate all TSP paths to build the overall schedule.

The partitioning of nodes into groups is accomplished with a two-step process. In the first step, all nodes are grouped into bins $B_i, 1 \leq i \leq M$, such that the minimum overflow time o_{i+1} in B_{i+1} is twice the minimum overflow time o_i in B_i . Although every bin B_i has nodes with different overflow times, we assign the minimum overflow time to all nodes in that bin. Consequently, we can treat all o_i nodes in the same bin equally without violating the overflow conditions. In the second step, nodes in a bin B_i are partitioned into 2^{i-1} sub-bins $B_i^j, 1 \leq j \leq 2^{i-1}, 1 \leq i \leq 2$, using a 2d tree algorithm such that nodes in the same sub-bin are located close to each other.

The generation of visiting schedules is also accomplished in a two-step process. In the first step paths that traverse each node in a sub-bin once are calculated for each sub-bin individually. The sub-bin paths are calculated using a solution to the Traveling Salesman Problem (TSP). In the second step these partial paths are concatenated in such a way that the nodes are visited at appropriate frequencies. Note that a bin B_{i+1} is partitioned into twice as many sub-bins as B_i .

However, a sub-bin B_{i+1} needs to be visited only half as frequently as a sub-bin of B_i .

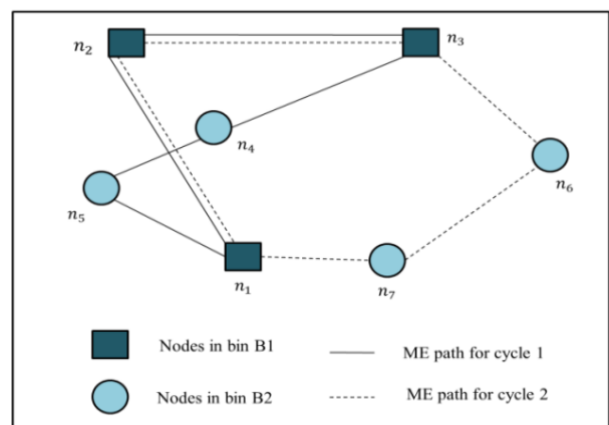


Fig. 1 A sample network deployment and the ME path generated by the PBS algorithm. The overflow times for the nodes are as follows: $o_1 = o_2 = o_3 = 4$ time units; $o_5 = o_6 = o_7 = 8$ time units.

Figure 1 demonstrates a simple example of the PBS algorithm. Partitioning with respect to buffer overflow times results in bins $B_1 = \{n_1, n_2, n_3\}$ and $B_2 = \{n_4, n_5, n_6, n_7\}$ for the sample network. Then B_i is partitioned into two with respect to sensor locations resulting $B_2^1 = \{n_4, n_5\}$ and $B_2^2 = \{n_6, n_7\}$. Nodes in B_1 are visited every cycle, and nodes in B_2 are visited every other cycle. Let us assume B_{21} is visited every even cycle, and B_2^2 every odd cycle. In the example of Fig. 3.1, two cycles of ME visits are formed by the PBS algorithm. The first cycle consists of the nodes in B_1^1 and B_2^1 with visit order of n_1, n_2, n_3, n_4, n_5 and back to n_1 . The second cycle consists of the nodes in B_1^1 and B_2^2 with the order n_1, n_2, n_3, n_6, n_7 and again back to n_1 .

3.5 Multihop Route to Mobile Element (MRME)

Since, the primary objective of the PBS algorithm is to ensure that the data is collected before sensor buffer overflows occur. It is also possible that sensors generate urgent messages that must be delivered to MEs within given delay bounds in addition to regularly generated data. As an example, sensors in the traffic observation scenario may also detect accidents in addition to regular data collected. The accident information is an Urgent Message (UM) that must be delivered to an ME in amount of time. For solving this problem, MRME algorithm is used. Therefore, MRME aims to minimize the ME speed while finally in the last phase, the PBS algorithm is applied using the reduced overflow times. MRME reducing overflow times to meet the specified UM delay. MRME algorithm consists of three phases.

Overflow time reduction phase

- Initial covering phase
- PBS algorithm phase

In the overflow time reduction phase a node is selected and its overflow time is reduced based on a gain criterion. Then, nodes covered thanks to overflow time reduction are determined. This phase is repeated until all nodes are covered.

In the initial covering phase, nodes n_i that satisfy D_i are determined. We refer to the nodes that can meet the UM deadline requirement as covered nodes

guarantees that the generated ME schedule solves the DMD problem.

The MRME algorithm can be presented below:

MRME Algorithm

1. Build n_i, d for all $n_i \in N$ and $d = 0, 1, \dots, d_{\max}$ by applying Floyd's all-pairs shortest path algorithm.
2. Mark all the nodes as uncovered
3. Determine nodes n_i that satisfy D_i and mark them as covered
4. **while** There are uncovered nodes **do**
5. Find the node with maximum gain and reduce its overflow time
6. Determine nodes n_i that satisfy D_i and mark them as covered
7. **end while**
8. Run PBS algorithm with modified overflow times.

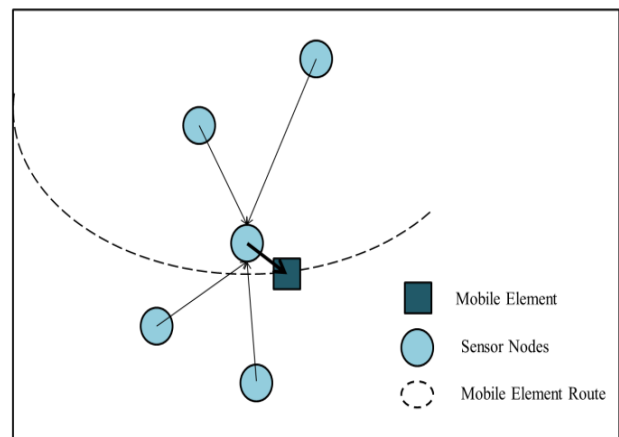


Fig.2. Multi-hop communication to reduce UM delay in MRME algorithm

In Figure 2, UMs can be relayed to nearby neighbors that are visited more frequently by the ME as shown is the example of the MRME algorithm. Therefore, UMs do not have to wait until the next ME visit to be picked up as in the case of a PBS solution. Therefore the Multihop Route to Mobile Element (MRME) algorithm that extends PBS to deliver urgent messages to MEs within specific delay bounds.

IV. CONCLUSION

In this survey paper, we have presented the scheduling algorithms one by one for data collection in wireless sensor network. Mobility with scheduling

algorithm for data collection in WSNs can maximize network life time and increasing the connectivity of sensor nodes and clusters. Also, concluded that the performance of MRME algorithm is better in terms of decreasing urgent message loss rate while minimizing the mobile element speed to avoid data loss.

Our further work includes investigation of methods to utilize more than one mobile element for data collection to solving the MES and urgent message problems.

REFERENCES

- [1] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of ACM Wireless Sensor Networks and Applications Workshop (WSNA 2002)*, 2002.
- [2] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: Research challenges," *Ad Hoc Networks Journal (Elsevier)*, vol. 2, pp. 351–367, October 2004.
- [3] R. Sugihara and R. Gupta, "Optimal speed control of mobile node for data collection in sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 1, pp. 127–139, 2010.
- [4] A. Boukerche and X. Fei, "Adaptive data-gathering protocols with mobile collectors for vehicular ad hoc and sensor networks," in *Proc. IEEE WIMOB'8*, 2008.
- [5] S. R. Gandham et al., "Energy Efficient Schemes for Wireless Sensor Networks With Multiple Mobile Base Stations," *Proc. IEEE GLOBECOM*, 2003.
- [6] Z. M. Wang et al., "Exploiting Sink Mobility for Maximizing Sensor Networks Lifetime," *Proc. 38th Hawaii Int'l. Conf. Sys. Sci.*, 2005.
- [7] R. Shah et al., "Data mules: Modeling a Three-Tier Architecture for Sparse Sensor Networks," *Proc. IEEE Wksp. Sensor Network Protocols and Apps.*, 2003.
- [8] A. Chakrabarti, A. Sabharwal, and B. Aazhang, "Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks," *Proc. 2nd Int'l. Wksp. Info. Proc. 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 Int'l. Real-Time Sys. Symp.*, 2004.
- [10] D. Jea, A. A. Somasundara, and M. B. Srivastava, "Multiple Controlled Mobile Elements (Data Mules) for Data Collection In Sensor Networks," *Proc. IEEE/ACM Int'l. Conf. Distrib. Comp, in Sensor Sys.*, 2005
- [11] C. Liu and J. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *Journal of the Association for Computing Machinery*, 20(1), Jan 1973.
- [12] A. A. Somasundara, A. Ramamoorthy, and M. B. Srivastava, "Mobile element scheduling for efficient data collection in wireless sensor networks with dynamic deadlines," in *25th IEEE International Real-Time Systems Symposium (RTSS'04)*, pp. 296–305, 2004.
- [13] Y. Guet *al.*, "Partitioning-Based Mobile Element Scheduling in Wireless Sensor Networks," *Proc. IEEE ComSoc Conf. Sensor and Ad Hoc Commun. and Net.*, 2005.
- [14] Y. Gu, D. Bozdog, and E. Ekici, "Mobile Element Based Differentiated Message Delivery in Wireless Sensor Networks," *Proc. IEEE Int'l. Symp. World of Wireless, Mobile and Multimedia Networks*, 2006

AUTHOR'S PROFILE



Rajeev Paulus

is working as an Assistant Professor in the Department of Electronics & Communication Engineering in SHIATS, Allahabad. He received the degree of M.Tech from MNNIT, Allahabad. He is pursuing Ph.D. He has presented & published various research papers in national and international journals and conferences. His research interests are wireless Sensor Network and Computer Networks.



Alok Singh

is pursuing M.Tech in Advanced Communication System from SHIATS, Allahabad. He has 14 months experience in National Informatics Center (NIC) as a network Engineer in Gonda (UP). His research interests are Wireless Sensor Network and Computer Networking.



Aditi Agrawal

is working as a Teaching Assistant in department of Electronics Communication Engineering in SHIATS, Allahabad. She pursuing M.Tech. in Wireless Communication from SHIATS, Allahabad. Her research interests are Digital Communication and Wireless Sensor Network.



Manvendra

is working as a Teaching Assistant in department of Electronics & Communication Engineering in SHIATS, Allahabad. He is pursuing M.Tech in Advanced Communication System from SHIATS, Allahabad. His research interests are Communication Networks.