www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 3 March-2014 Page No. 5118-5122

ENHANCEMENT OF QUALITY OF SERVICE IN WIRELESS SENSOR NETWORKS THROUGH ENERGY EFFICIENT QUERY SCHEDULING SCHEME

***G.Srimathy*[1], *I. Aarthi* [2] ,*V.Geetha*[3],*V.Priyadharsini*[4] and *T.Kalaivani*[5]**
[1] Assistant Professor, Sri Manakula Vinayagar Engineering College, India.
[2,3,4,5] Student, Sri Manakula Vinayagar Engineering College, India.
[1]sri.mathy73@gmail.com, rt617doss@gmail.com[2],
geethavenkat2592@gmail.com[3],priyasagittarius92@gmail.com[4],vanivijay2007@gmail.com[5]

***Abstract* — A Wireless Sensor Network (WSN) consists of sensor nodes capable of collecting information from the environment, process the collected information and direct it to the desired destination (sink). Due to the recent advances in wireless communications, it is important to address the quality of service (Qos) issues. Energy consumption, delay and throughput are some of the Qos issues. The WSN with Static Sink (SS) which uses multihop communication between sensors and SS have certain drawbacks which degrade the Qos of the system. Due to the use of multihop paradigm, the sensors around SS deplete their energy very fast due to high forwarding data traffic. Thus the higher energy consumption by these sensor nodes cannot be avoided. Also, workload of sensor nodes is very high due to the SS. Thus the energy efficient query scheduling scheme is proposed which uses mobile sink to save sensor energy spent for multihop communication. This aims at extending the network lifetime by optimizing the energy consumption, throughput and delay thereby improving the Qos of the system**

Index Terms —Wireless Sensor Networks (WSN), Quality of service (Qos), energy consumption, multi hop communication, mobile sink.

I. INTRODUCTION

In WSN, scheduling plays a dominant role in reducing sensors energy consumption. Most of the existing scheduling schemes, does not optimize energy consumption. Higher the energy consumed by the sensors, higher will be the depletion of its battery. This drastically reduces the network lifetime of WSN. Thus energy constraint is a major factor affecting the Qos of the system. Most of the energy is spent during transfer of data packets from sensor to destination. In order to reduce the energy consumption mobile sinks have been adopted.

The possible ways of initiating data transfer from sensors to the sink can be categorized into four classes: event-driven, time-driven, query-based, and hybrid[3]. In event-driven data collection approach, data needs to be collected whenever a specific event is detected and then forwarded to a sink (base station).

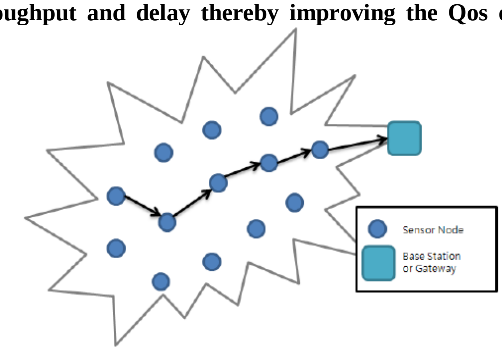

FIGURE 1: WSN Model

. Similarly, query-based data collection is triggered upon receiving a user's query, and response should be routed to a sink. On the other hand, in a time-driven approach, data collected by sensors is transferred to a sink periodically. One common way to decrease the energy consumption in the communications between sensors and static sink (SS) is using multihop forwarding instead of direct connection which is adopted in the existing scheduling scheme [1].

Multihop communication has the advantage of using low power in transmitting data to a nearby sensor. However, one of the main problems of applying Multihop paradigm in WSN is that the sensors around SS deplete their energy very fast due to high forwarding data traffic. When that occurs, the sink becomes unreachable and WSN is nonfunctional thereby making the existing scheduling scheme [1] less energy efficient due to the indispensable energy constraints involved in multihop communication.

As an alternative to Multihop communication, the concept of mobilizing sinks to collect data from the sensors is proposed [2]. These mobile sinks (MSs) are capable of moving in the monitored field and contacting sensors either by one hop (direct) or limited number of hops. As a result, the required energy for transferring data from sensors to sink is reduced considerably and the life time of WSN is extended significantly [3]. This approach is highly suitable for delay tolerant applications [4].

In this paper, a unique way of using MS for real time application like query-based data gathering with deadlines, by trading off delay in response with energy consumption is used. In this class of applications, location-based queries are submitted to WSN. Responses should be collected before the specified deadline expires. For this reason, a query scheduling algorithm to exploit MS deterministic mobility for saving energy in communication and to exploit speed of Multihop communication for minimizing delay is designed. This algorithm balances the system throughput and energy consumption by optimizing the number of hops and duration of response time.

The summary of the paper is organized as follows. In section II, the existing scheduling scheme is discussed. Section III focuses on the quality of service Support (QoS) in WSN. Section IV presents the various terminologies of the proposed scheduling scheme. Section V provides a performance analysis of the proposed Deadline aware energy efficient query scheduling algorithm. Section VI provides conclusion.

II. RELATED WORKS

A. First Come First Served (FCFS)

The processing of data is done in the order of their arrival times at the ready queue, in the case of First Come First Served (FCFS) schedulers [1]. Here, the data from nearby neighboring nodes take less time to be processed at the intermediate nodes when compared to that of the data from the distant leaf nodes. Also, many data packets arrive late and so they experience higher delay. The number of packet failures results from the increased delay, which affects the Qos of the system.

B. Single Queue

Each sensor has single ready queue. Single queue scheduling has a high starvation rate [1]. Due to high starvation rate the deadline of the packet may get expired, this results in packet loss. So the reliability of the system diminishes which in turn degrades the Qos of the system.

C. Multi-level Queue

Each node has two or more queues. Based on the priorities and types of data packets are placed into the different queues. Thus, scheduling involves two phases: (i) allocating tasks among different queues, (ii) scheduling packets in each queue. The number of queues at a node determines the level of the node in the network [1]. For instance, a node at the lowest level or a leaf node has a minimum number of queues whereas a node at the upper levels has more queues.When a packet is received by a node, the priority of the packet is decided by the node according to the hop count of the packet and accordingly sends it to the relevant queue.

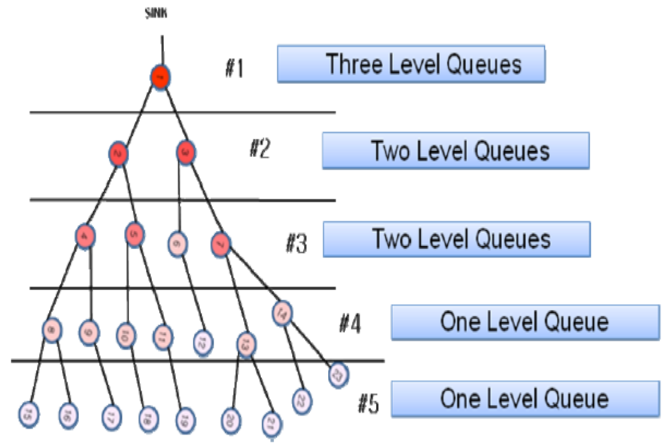

FIGURE2: Multilevel Queue

The other scheduling schemes such as pre emptive and non pre emptive schemes also experience more starvation rate which may lead to the packet loss. This decreases the throughput of the system which in turn degrades the Qos of the system.

D. Dynamic Multilevel Priority Scheduling

Nodes that are at the same hop distance from the base station (BS) are considered to be located at the same level. Time-Division Multiplexing Access (TDMA) scheme is used for processing the data packets of nodes at different levels. For instance, the leaf nodes (the last level of nodes) are assigned the first timeslot and the second lowest level (the level of nodes preceding the leaf nodes) of nodes are assigned to the next timeslot.

Three-level of queues are considered, that is, the maximum number of levels in the ready queue of a node is three: priority 1(*pr*1), priority 2 (*pr*2), and priority 3 (*pr*3) queues[1]. The highest priority queue contains the real time data packets, which are processed using FCFS. In case of non-real-time data, those packets that arrive from sensor nodes at lower levels go to *pr*2, the second highest priority queue and the non-real time data packets that are sensed at a local node go to *pr*3. In DMP, if the resources are held by real time tasks for a longer period of time, other tasks need to wait for an undefined period time, which causes the occurrence of a deadlock.

This degrades the performance of the system. Moreover, this scheme could be less energy efficient because it takes more processing cycles for categorizing and placing the tasks in the different queues and also for context saving and switching(preemption). The scheduling schemes, which are discussed above have their own drawbacks which degrades the Qos of the system.

III. QUALITY OF SERVICE (QOS) SUPPORT IN WSN

A. Node Communications

The process of providing the routes have a great impact on energy considerations. Comparing to direct communication of nodes, multi-hop routing will consume less energy [10]. Since the transmission power is proportional to the square of the distance or even higher order, than direct communication.

B. Mobility

For allowing communication between the connected components of a network, mobility is indispensable. Higher the mobility lesser is the energy consumed by the sensors which in turn increases the life time of the nodes. Sensor mobility also allows better coverage [9].

IV. TERMINOLOGIES

- Received Query: Number of queries whose response that will arrive at the MS on time.
- Generated Query: Number of queries that are created by remote central during the simulation.
- Submitted query: Number of queries which are submitted to WSN by MS.
- Missed Deadline: Number of queries whose response could not get to MS before the deadline exceeds. Any sensor node which is one hop away from the route can directly communicate with passing by MS. These sensors are called Gateway Sensors (GSs)
- Network Life Time: Duration between the time that simulation begins and the time that any sensor's battery power gets lower than a specified level. The levels are sliced as 10%of the starting battery capacity.
- Average Energy Consumption Per Submitted Query: Average amount of energy consumed for forwarding query and response messages in WSN during simulation time.

V. PROPOSED DES ALGORITHM

The main aim of Deadline-Aware Energy-Efficient Query Scheduling Algorithm (DES) is to minimize the energy spent during the multihop communication. In order to minimize the energy consumption, the number of transmissions from the sensor nodes has to be minimized. Thus by adopting mobile sink, responses are provided before the given deadlines and minimum amount of sensor energy is consumed during multihop communication. The following illustrates the DES algorithm.

A. Problem Definition

The main requisites of WSN are sensor nodes, mobile sink, and Multihop routing protocol. The messages are routed from source to destination through a multihop routing protocol. The mobile sink moves with a fixed velocity on a predefined route. In reality, mobile sink could be embedded onto a regularly moving object.

The sensors which are at one hop distance from the mobile sink can directly communicate with the mobile sink are referred to as Gateway Sensors (GSs). The mobile sink forwards the query to a target sensor as specified by the remote central. The queries which are specified by the remote central contains the location of the Target Sensor (TS) and the query deadline. Query deadline is the time before which the query should be executed and the response should be delivered to the remote central.

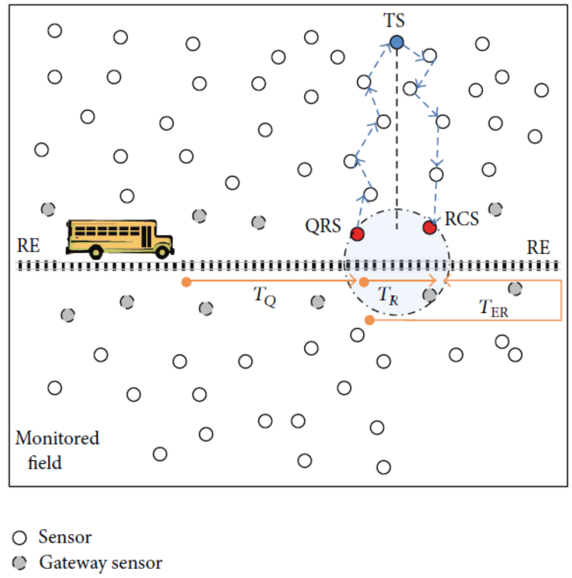

FIGURE 3: Assumed WSN model

QRS	RCS	TS	Deadline	Query

RCS	Deadline	Response

FIGURE 4: Query and response message content.

The queries with different deadlines arrive dynamically as MS keeps moving on the path. The mobile sink attempts to create a schedule such that the energy consumption for forwarding messages would be minimum and response would be uploaded to the remote central before the given deadline. A schedule comprises of QRS (a gateway sensor which injects a query into the WSN) and RCS (a gateway sensor which collects the response from the target sensor). The trip from current MS location to QRS location is named as query release trip. TS processes the query, prepares the response message, and forwards it to RCS.

As MS is traverses its path, it attempts to collect the response message from RCS. The trip from QRS to RCS is called Response Collect Trip. If the response message has arrived already at RCS, MS collects and uploads it to the remote central. If the response message, on the other hand has not arrived at RCS yet, MS takes an Extended Response Collect Trip by moving to the route end (RE) and comes back to RCS once more. Thus, MS tries to collect the response message from RCS for the second time. To meet the deadline, the MS must reach the RCS before deadline. The response message must also reach the RCS before the MS reaches the RCS. To have minimum energy consumption in transferring query and response packages, the QRS and RCS from GS should be selected such that they have minimum number of hops from the target sensor. However, these are not the sufficient conditions. Thus, the critical problem in scheduling is to find QRS and RCS such that MS gets its response before the deadline with minimum energy consumption. The following elucidates the DES.

B. Solution

Using the shortest path between QRS-TS and TS-RCS, DES first attempts to construct least Energy Consuming Schedule(LECS). If it is not possible due to deadline or message routing, it can attempt to create either Optimum Energy Consuming Schedule (OECS) keeping TS-RCS path shortest but extending QRS-TS path or Maximum Energy Consuming Schedule (MECS) modifying both TS-RCS and QRS-TS paths to gain more time. Considering MS current location, movement direction, and speed along with route ends location, target sensor location, query size, response size, data transfer rate, query processing time, expected numbers of hops from QRS to destination, expected number of hops from destination to RCS (Hop RCS) and query deadline, DES creates plausible schedules.

Initially, LECS checks if MS has enough time to reach RCS location before the deadline, after calculating all these parameters. If deadline allows, MS should ensure the time needed for query forwarding and processing, and response forwarding would be less than trip time to RCS. There are certain cases wherein the response message may arrive late at RCS. And at that instant, MS is allowed to move up to RE and return back to RCS. Due to late arrival of response message to RCS, LECS algorithm fails to create the shortest routing path. So, it calls OECS to select an alternative QRS (AQRS) such that MS can release query earlier. The LECS algorithm fails because of deadline expiration before MS finishes collect trip. MECS algorithm is called for choosing AQRS and alternative RCS (ARCS) such that MS and the response message would meet at ARCS before the deadline.

(1) Select QRS and RCS
(2) Calculate the $T_{Q,}$,T_R,T_{ER},T_{EQF} and T_{ERF}
(3) **if** $T_Q + T_R >$ Deadline_Q **then**
(4) **failed**
(5) run MECS algorithm
(6) **else**
(7) **if** $T_{EQF} + T_{ERF} <= T_R$ **then**
(8) **successful**→ Schedule QRS and RCS
(9) **else**
(10) **if**($T_{EQF} + T_{ERF} <= T_R$)and($T_Q + T_{ER} <= \text{Deadline}_Q$) **then** Select QRS and RCS
(11) **Successful**→ schedule QRS and RCS
(12) **else**
(13) **failed** →response is late
(14) run OECS algorithm
(15) **end if**
(16) **end if**
(17) **end if**

ALGORITHM1: Calculate Least Energy Consuming Schedule

C. Constructing OECS

When the response message arrives late at RCS before MS passes by, OECS algorithm searches an alternative routing path. OECS algorithm relocates QRS such that query message forwarding would begin earlier and reply message can arrive to RCS on time. Since the response data size is expected to be larger than query, to save more energy in routing messages, response forwarding path should be kept minimum. A minimum energy consuming path for forwarding response messages is constructed by creating a shortest path between TS and RCS. The optimum route is constructed only if we can build a routing path such that query forwarding would take higher number of hops, but response forwarding takes only least number of hops. Thus the response is kept ready for MS.

The OECS algorithm calculates alternative QRS (AQRS) location such that the total time required would be less or equal to response collect trip time. AQRS is selected among gateway sensors one hop away from above location, if such a location is feasible. Instead, the next nearest sensor from AQRS is selected and tested such that the extended response collect trip can be run before deadline. Before MS reaches RCS, the response would be ready at RCS. If both LECS and MECS fail , MECS is called to create an alternative schedule.

Require: RCS
Ensure: LECS found: $T_Q + T_R <= \text{Deadline}_Q$ and $T_{EQF} + T_{ERF} > T_R$
(1) select AQRS s.t. updated $T_{EQF} + T_{ERF} < T_R$ holds
(2) **if** AQRS is feasible **then**
(3) **successful**→ schedule AQRS and RCS
(4) **else**
(5) select AQRS s.t. updated $T_{EQF} + T_{ERF} < T_{ER}$ and $T_Q + T_{ER} <= \text{Deadline}_Q$ holds
(6) **if** AQRS is feasible **then**
(7) **successful**→schedule AQRS and RCS
(8) **else**
(9) **failed**→ Response is late
(10) run MECS algorithm
(11) **end if**
(12) **end if**

ALGORITHM2: Calculate Optimum Energy Consuming Schedule

D. Constructing MECS

LECS uses minimum energy by creating shortest routing paths for query and response message delivery, whereas OECS consumes less energy for response messages but more energy for delivering query to gain time. When these two algorithms fail to create a plausible solution, MECS algorithm is made active. The routing path constructed by MECS algorithm costs more energy to gain time by attempting to select an alternative QRS (AQRS) as well as an alternative RCS (ARCS) in the hope that response message can be reachable by MS before the deadline. MECS first decides the possible nearest point on the route to TS where MS can get before deadline finishes. The sensors in the vicinity of are candidates for alternative RCS (ARCS). Then alternative QRS (AQRS) location is calculated such that response message arrives to ARCS before MS. Considering deadline if such AQRS location is not available, MECS algorithm recalculates AQRS and ARCS locations such that the distance between these two locations would be the largest. Before deadline, it is expected that MS moves from AQRS to ARCS, the query and reply message would reach to ARCS.

(1) Select ARCS s.t. Dis(ARCS,TS)is minimum
and updated $T_Q + T_R <$ Deadline$_Q$
(2) Select AQRS s.t. updated $T_{EQF}+T_{ERF}<T_R$
(3) **if**(ARCS and AQRS) is feasible **then**
(4) **successful→** Schedule AQRS and ARCS
(5) **else**
(6) Select ARCS and AQRS s.t. Dis(ARCS, AQRS) is maximum,
updated $T_Q+T_R<$Deadline$_Q$ and updated $T_{EQF}+T_{ERF}<T_R$
(7) **if** (ARCS and AQRS) is feasible **then**
(8) **successful→** Schedule AQRS and ARCS
(9) **else**
(10) **reject query→** Deadline is short
(11) **end if**
(12) **end if**

ALGORITHM3: Calculate Maximum Energy Consuming Schedule.

V. CONCLUSION

A scheduling algorithm for the location-based queries in WSN with a mobile sink which follows a deterministic mobility pattern is introduced. In WSN, for the dissemination of the queries and the responses, Multi hop communication pattern is used. The queries are associated with deadlines.

The proposed scheduling algorithm aims at not only maximizing the number of successful queries but also reducing the sensors' energy expenditure due to Multihop communication which is done by exploiting deterministic sink mobility.

Due to this, prior to submitting queries, the scheduling algorithm selects the release and collect sensors such that two important performance requirements can be met: the energy required to forward data packets should be minimum and the response arrival time should not exceed the specified deadline.

REFERENCES

[1] Nidal Nasser, Lutful Karim, and Tarik Taleb, "Dynamic Multilevel Priority Packet Scheduling Scheme for Wireless Sensor Network", IEEE transactions on wireless communications, April 2013.

[2] M. I. Khan, W. N. Gansterer, and G. Haring, "Static vs. Mobile sink: the influence of basic parameters on energy efficiency in wireless sensor networks," Computer Communications, vol. 36, no. 9, pp. 965–978, 2013.

[3] M. Di Francesco, S. K. Das, and G. Anastasi, "Data collection in wireless sensor networks with mobile elements: a survey," ACM Transactions on Sensor Networks, vol. 8, no. 1, pp. 7:1–7:31, 2011.

[4] Y. Yun and Y. Xia,"Maximizing the lifetime of wireless sensor networks with mobile sink in delay-tolerant applications," IEEE Transactions on Mobile Computing, vol. 9, pp. 1308-1318, 2010.

[5] S. Muthukarpagam, V. Niveditta and S. Neduncheliyan, "Design issues, Topology issues, Quality of Service Support for Wireless Sensor Networks: Survey and Research Challenges", International Journal of Computer Applications (0975-8887), vol. 1-No. 6, 2010.

[6] G. Anastasi, M. Conti, and M. Di Francesco, "Extending the lifetime of Wireless sensor networks through adaptive sleep," IEEE Trans. Industrial Informatics, vol. 5, no. 3, pp. 351–365, 2009.

[7] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: a survey" Ad Hoc Networks, vol. 7, no. 3, pp. 537–568, 2009.

[8] A. Somasundaran, A. Ramamoorthy, and M. B. Srivastava, Mobile element scheduling with dynamic deadlines," IEEE Transactions on Mobile Computing, vol. 6, no. 4, pp. 395–410, 2007.

[9] Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. S. Sukhatme, "Robomote: Enabling mobility in sensor networks, " in IEEE/ACM Fourth International Conference on Information Processing in Sensor Networks (IPSN/SPOTS), Apr. 2005.

[10] K. Akkaya and M. Younis, "An Energy-Aware QoS Routing Protocol for Wireless Sensor Networks," Proc. of the IEEE Workshop on Mobile and Wireless Networks (MWN 2003), Providence, RI, May 2003.