



# SMART COLLECT - INTELLIGENT DYNAMIC TIME AND EVENT SWITCHED ENERGY EFFICIENT DATA GATHERING PROTOCOL

A. Gnana Soundari<sup>1</sup> and V. L. Jyothi<sup>2</sup>

<sup>1</sup>Department of Computer Science and Engineering, Sathyabama University, Chennai, Tamil Nadu, India

<sup>2</sup>Department of Computer Science and Engineering, Jeppiaar Engineering College, Sathyabama University, Chennai, Tamil Nadu, India  
E-Mail: [gnanasoundari1@gmail.com](mailto:gnanasoundari1@gmail.com)

## ABSTRACT

With rapid growth in modern science and technology, performance centric automated systems play a vital role in wireless communication. Wireless Sensor Networks are randomly deployed in large scale to fetch data, which is used to establish a control system of critical and non-critical type like Disaster Alert Management, Fire and Safety Systems and Habitat Monitoring etc. Data Gathering process becomes a vital for such systems and needs continuous research to find out better solution. To maintain a successful data gathering, further WSN should able to periodically self-organize and reconfigure itself where manual interventions nor configuration is not feasible. Introducing intelligence to sensors and having self organized network is one of such attempt to have data gathering effectively. Our work contribution is towards having a Smart Network Management and Reconfigurable Sensor Nodes to achieve successful Data gathering process for mission critical applications. A novel "Proactive Event and Time driven (Pro\_ET)" data gathering protocol is proposed in this article with energy efficient weighted fairness queuing (WFQ) mechanism for supporting on-demand reconfigurable sensors referred to as "Smart Collectors (SCs)". "Smart Collectors" will function as data aggregator and gatherer with self-organizing ability towards critical and non-critical events detected on time function. SCs ensure right scheduling and fairness index in packet processing maintaining Quality of Service. Simulation results of Pro\_ET shows promising gains on reduced latency and high packet delivery rate compared to other existing methods.

**Keywords:** wireless sensor network, energy efficient, smart collector, data gathering, packet scheduling.

## INTRODUCTION

Advances in Wireless Sensor Networks (WSN) have led to rapid transformation of manually operated systems to performance centric automated systems. Most applications involve placement of sensor devices at different geographical locations for monitoring environmental parameters and are controlled centrally at a remote site. With its fundamental objective being functional for a long period of time, it becomes impossible to hand place sensor nodes where battery replacement is usually cumbersome. Especially for time critical situations, challenging deployment environments with severe resource constraint sensors pose intricacies in reliable and efficient data communication [3] to the sink. The merit and aptness for autonomous and unattended monitoring in critical and non-critical applications, has captivated to immense growth in WSNs. Researchers believe that by increasing the number of static sink nodes one can distribute the traffic load all over the network through clustering [1] [16] [21] techniques and consequently balance energy consumption around the sink. Low latency [2] with guaranteed data delivery from source node to the sink is the main concerns in this research area.

Several research works have been employed in these areas to prove the efficacy of the network [7]. Among the large amount of data sent from sensors to the sink, only fractional part of such data is considered to serve the purpose (considered as useful data) for processing and analysis. For instance, data sensed for real time applications have higher priority. In such scenarios, serving data with high importance can be prioritized

delaying or dropping the ones with lower importance. Among many key network design issue such as routing protocols [9] [17] [20], data gathering and aggregations [8-10] [12-13] [18] [22-23], critical event monitoring, task or packet scheduling [15] at sensor nodes is considered to be highly important to ensure delivering data based on priority and fairness with minimum latency for guaranteed network performance. As the primary objective of WSN is to improve the network performance, scheduling packets at sensors plays a vital role to reduce sensor's energy consumption and end-to-end data transmission delays. Most existing WSN operating system uses First Come First Served (FCFS) [19] scheduling algorithms for packet processing at sensors. Using FCFS schedulers data packets (both real time and non-real time data) are processed in the order of their arrival time. FCFS schedulers with single queue causes long delay for real time data packets when delivered to sink. To eliminate problem caused in FCFS, Multilevel Queue (MLQ) [11] schedulers were supported. MLQ uses different number of queues based on the location of sensor nodes. These schedulers uses simple priority-based (data enter the queue according to priority) and multi-FIFO queue-based (two or more queues with priority set to high, mid, or low for data packet handling. According to packet's priority it is sent to the relevant queue for processing) mechanism. Drawback of MLQ was in its improper allocation of data packets (real and non-real time data) to queues causing high processing overhead and starvation rate at peak traffic rate. Moreover, these algorithms are not dynamic to the changing requirements



of WSN applications since their scheduling policies are predetermined.

Challenge for real time applications is in delivering emergency data on time with shortest possible delay to sink. Clearly, network in which data collection rate dominates data forwarding rate (in a traffic-intense application, e.g. video-based target tracking), typically congestion builds up at the bottleneck nodes. Consequently, packet dropping incident and/or retransmission becomes more frequent, leading to increasingly degraded network performance.

In this paper, a novel "Proactive Event and Time driven (Pro\_ET)" data gathering protocol is designed to support on-demand reconfigurable sensors referred as "Smart Collectors" whose behaviour or functionality dynamically changes based on events detected. Pro\_ET protocol is a:

- a) Dynamic priority based packet scheduling mechanism designed to suit real time applications.
- b) Re-organizable and re-configurable algorithm designed as per application requirement
- c) Simple, robust and energy efficient weighted fairness queuing (WFQ) mechanism designed to serve multiple objective (maximize the QoS) of the system.

Simulation was conducted to prove the efficacy of the proposed Pro\_ET technique. Model was developed using self written script in MATLAB to evaluate the performance of the proposed scheme. The results shows that more than 99% of high priority packets were successfully delivered on time to the sink improving packet delivery ratio with reduced latency of 40% when compared to existing schemes. In Addition, Pro\_ET scheme was proved to be extremely suitable for wireless sensor networks where the major task is for the sink to

collect readings from all the sensor nodes simultaneously. The remainder of this paper is organized as follows: Section 2 elaborates the proposed methodology involved in Pro\_ET protocol; Section 3 details the results and discussion of Pro\_ET. Finally, Section 3 concludes the paper.

## SYSTEM DESCRIPTION

In Pro\_ET, we consider the topology of WSN with large number of randomly deployed static sensor nodes ( $N$ ) and one sink node ( $S_{node}$ ). At time  $t = 0$ , we assume that all sensor nodes are deployed. The transmission of sensor nodes varies (increase or decrease) dynamically depending on its location, residual energy and transmission capability.

The proposed Pro\_ET system architecture includes the following main components:

**Sensor nodes:** The sensor nodes ( $S_{node}$ ) senses the environment, collects sensory information and communicates to the Smart Collector.

**Smart collector:** The role of sensor node has been extended to act as smart collectors. The node is configured for detecting events (the node can sleep until an event occurs, providing significant power savings when compared to polling) and handles such events whenever detected. The sensor nodes are customized to realize additional service bringing intelligence that optimizes the node's behaviour specific to application and network requirement.

**Sink:** It is the back-end centralized control system that collects information from sensors and continuously synchronizes data received from sensor nodes over time to the server. The collected information represents a vital source of big data for the statistical and research activity.

The System model of Pro\_ET scheme is depicted in Figure-1.

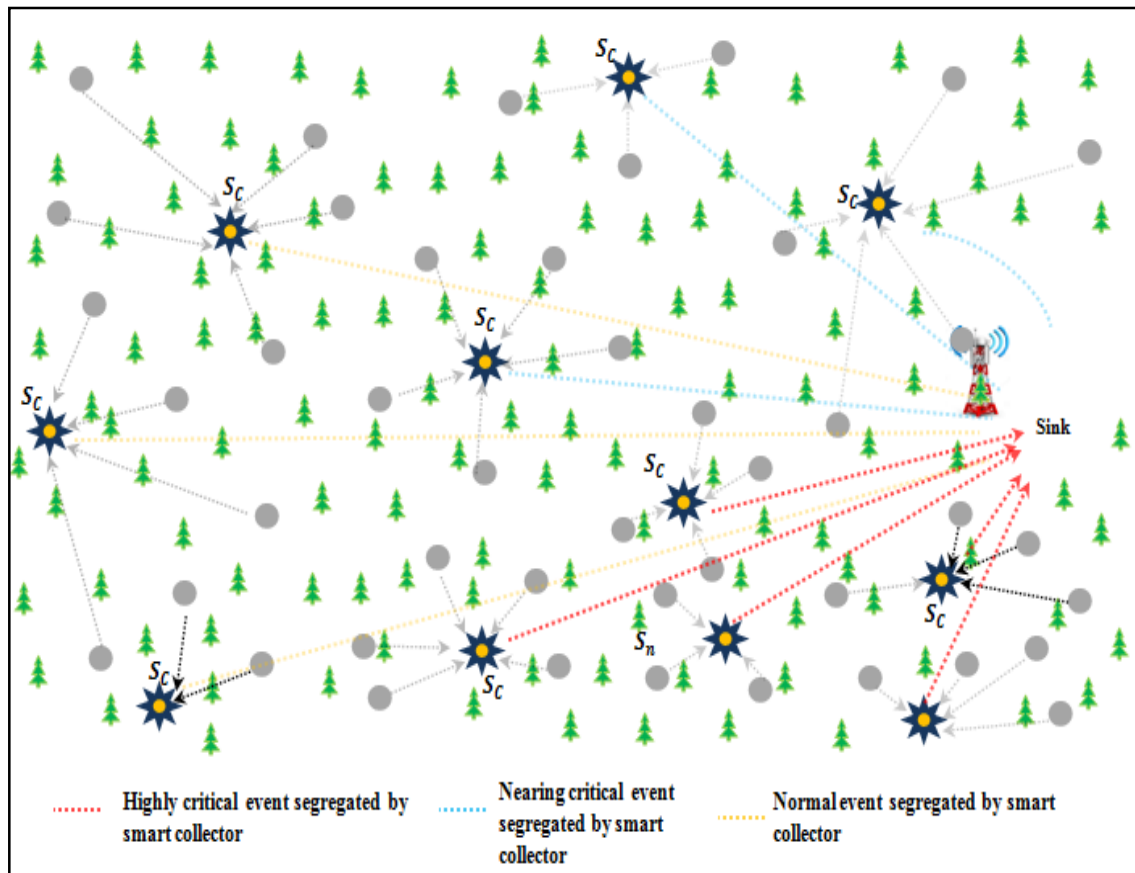


Figure-1. Pro\_ET system model.

The Pro\_ET architecture is categorized into the following phases:

- Reconfigurable and reorganizable smart collectors (SC).
- Dynamic priority based packet scheduling mechanism using Pro\_ET.

#### Reconfigurable and reorganizable smart collectors

In our work, the role of sensor node has been extended to act as smart collectors. i.e., the sensor nodes are customized to realize more flexible dynamic reconfiguration and high performance processing. It provides additional service bringing intelligence through interpretation techniques that optimizes the node's behaviour specific to application and network requirement. Dynamic reorganizing capability of the sensors provides greater advantage. As resource constraint of sensors makes it undesirable to implement expensive algorithms, we design Pro\_ET very carefully such that its operation in sensors makes it energy efficient (with minimal load and energy consumption while it executes). The sensor node is pre-equipped with multiple reprogramming capability [5] (each of which corresponds to particular event), using which its sampling and transmission rate decreases to conserve energy. Not all sensors in the field execute Pro\_ET, while only those nodes that are elected as smart collector (head node) are permitted to do so. Initially,

Pro\_ET selects a node randomly to initiate the smart collector(SC) selection process. Based on number of neighbors in a sub region, groups are formed in non-uniform distribution. Most recent information about the current status of neighbor nodes (various dynamic constraints like Node's ID ( $N_{id}$ ), Node's location ( $N_{location}$ ), Node's distance ( $N_{distance}$ ) and Node's residual energy ( $NE_{res}$ )) plays a vital role in selection of SCs. Score is calculated and recorded using the constraints. Each node calculates its score independently and notifies its neighbors about it. One of the neighbor with the highest score is selected to be the SC. SC is the leader of a group. It performs Pro\_ET activity based on varied time and events detected. Additionally it is capable of collecting data from its neighbors, aggregates and relay it to the sink. Steps involved in selection of smart collectors are:

**Step 1:** Each node initiates the process by broadcasting a "hello" message at low power level. Objective of this broadcast is to send the message only to neighbors who are in close vicinity. Sensor node's, Node ID ( $N_{id}$ ), Node's location ( $N_{location}$ ), Node's current transmission power ( $N_{tp}$ ), Node's distance from sink ( $N_{distance}$ ) and Node's residual energy ( $NE_{res}$ ) forms part of the message transmitted.

**Step 2:** Nearby neighbor nodes receives the broadcasted "hello" message and records the sender's information in its local aware table (LAT). It then sends an



acknowledgement “ack” message back to the sender. Sender node receives the “ack” message and records the information in its LAT about its neighbors.

**Step 3:** Each sensor node calculates its (as well as its neighbors) score or weight using the information recorded in LAT. Node with high transmission power ( $N_{tp}$ ), greater residual energy ( $NE_{res}$ ) and minimal distance ( $N_{distance}$ ) is given higher score. Node using the calculated score compares its score with other neighbors and node with highest score is considered for smart collector selection candidate.

**Step 4:** Let node ‘ $n_{sc}$ ’ is considered for SC selection candidate, the node ‘ $n_{sc}$ ’ propagates “Group Formation” message to its neighbors declaring itself as SC. Neighbor nodes which listens to this message becomes part of the group by sending a “Confirmation Message” to node ‘ $n_{sc}$ ’ confirming its selection as smart collector node of the group. Any further message for group formation are discarded by those nodes.

**Step 5:** SC selection process is triggered after every ‘ $t_i$ ’ time slot (‘ $t_i$ ’ depends on application requirement), if node ‘ $n_{sc}$ ’ is found to be inappropriate node to act as smart collector (comparing the current updated scores of nodes). Now, the node ‘ $n_{sc}$ ’ sends “Smart Collector Message” to the node ‘ $n_i$ ’ with the highest score among the group to become the SC. After notifying the node ‘ $n_i$ ’, node ‘ $n_{sc}$ ’ waits for ‘ $\delta$ ’ time duration for the best node ‘ $n_i$ ’ to respond. Within time slot ‘ $\delta$ ’, if the node ‘ $n_{sc}$ ’ fails to receive a response on “Group Formation” message from node ‘ $n_i$ ’ or any other node in the group, then the node ‘ $n_{sc}$ ’ itself continues to remain as SC of the group.

Algorithm 1 depicts the steps involved in SC node selection is referred below:

**Algorithm 1:** Smart Collector selection process

```

 $N_X$  broadcasts “hello” msg;
If ( $N_Y$  receives “hello” msg of  $N_X$ ) then
LATlist_Y = fetch_msginfo( $N_{id}$ ,  $N_{location}$ ,  $N_{tp}$ ,  $N_{distance}$ ,  $NE_{res}$ );
 $N_Y$  broadcasts “ack” msg;
If ( $N_X$  receives “ack” msg of  $N_Y$ ) then
LATlist_X = fetch_msginfo ( $N_{id}$ ,  $N_{location}$ ,  $N_{tp}$ ,  $N_{distance}$ ,  $NE_{res}$ );
end
end
scoredata = find_score(LATlist_X);
scorehigh = compare_score(scoredata);
SCnode = get_sc_node(scorehigh); // Node with
high score is selected as Smart Collector node
SCnode sends “Group_Formation” msg to Nodeneighbor
neighbors;
Nodeneighbor receives “Group_Formation” msg and becomes
part of group;
Nodeneighbour sends “Confirmation” msg to SCnode and
discards re-current “Group_Formation” msg .

```

Re-selection of SC is scheduled to trigger in fixed time interval enabling dynamic reorganization and reconfiguring of network specific to events detected. Using Pro\_ET protocol, SC appropriately interprets events

to serve high priority real time data packets of vital importance to the sink on time. Classification of packets using Pro\_ET approach is discussed in detail in next section.

Dynamic Priority Based Packet Scheduling Mechanism using Pro\_ET:

Pro\_ET’s vital importance is to reduce sensors energy consumption and end-to-end data transmission delay through effective and dynamic packet scheduling at SC’s for guaranteed network performance. Energy efficient weighted fairness queuing (WFQ) mechanism supported by Pro\_ET serve challenging deployment environments for reliable and efficient data communication to the sink. Let us consider the following assumptions to implement Pro\_ET scheme. Sensor nodes are randomly located within the geographical area considered. All sensors are time synchronized. The time of all nodes is synchronized with respect to the sink. Localization of nodes is done through simple static configuration, assuming that the nodes’ locations are known priori. Data traffic comprises of two types of data: real-time emergency data and real-time normal data. Size of the data packets (both real-time emergency and real-time normal) remains the same. Only Smart collector performs data aggregation and dynamic packet scheduling mechanism. Aggregation is done for normal events where the leader or the Smart collector of the group waits to receive minimum number of data (or reports about an event) before transmitting it to the sink. Each SC node comprises of two queues (high priority queue and low priority queue) for data packet handling. High and low priority queue is used for handling real-time emergency and real-time normal data packets. The length of real time emergency data queue is considered to be smaller than real-time normal data queue. Each data packet has a “Priority ID” and “Type ID” to identify its priority and type. Role of smart collector in performing dynamic priority based packet scheduling enables it in doing the following activities:

**Step 1:** Smart collector collects data packets from its group. It preprocesses the data packet by analyzing its data value and sets the Type ID and Priority ID for each packet.

**Packet type:** Each data packet has a *Type* identified by its *Type ID* ( $T_{ID}$ ). Type ID is set as normal (1), extreme (0) and abnormal (-1) based on data value ( $D_{value}$ ). Data packets from sensor nodes are forwarded to Smart Collector. SC verifies the data value in each packet and sets the Type ID appropriately.

```

if ( $D_{value} \leq N_{limit}$ ) then
 $T_{ID} = 1$ ;
//normal type
else if (( $D_{value} \geq E_{min\_limit}$ ) && ( $D_{value} \leq E_{max\_limit}$ )) then
 $T_{ID} = 0$ ; // extreme
type
else if (( $D_{value} \geq E_{max\_limit}$ ) || ( $D_{value} \geq AN_{limit}$ )) then
 $T_{ID} = -1$ ; //
abnormal type

```



```
else
```

```
    TID = NAN;  
    //Not a number (discard packet)
```

```
end
```

Where  $N_{limit}$  indicates the normal limit,  $E_{min\_limit}$  indicates the minimum value of extreme limit,  $E_{max\_limit}$  indicates the maximum value of extreme limit,  $AN_{limit}$  indicates the abnormal limit.

**Step 2:** SC analyzes  $P_{ID}$  of each packet and allocates them in most suitable queue for processing. Priority ID is set to “1” if the data’s  $T_{ID} = 1$ , it is considered as a normal event. Data of such packets can be aggregated removing duplicates and processed through low priority queue. Similarly, Priority ID is set to “0” if the data’s  $T_{ID} = 0$ , it is considered as an extreme event that has the high capability to cause an emergency incident. Data of such packets need to be placed in high priority queues for faster processing. While, Priority ID is set to “-1” if the data’s  $T_{ID} = -1$ , it is considered as a high alert emergency event that has triggered an incident. Data of such packets are placed in high priority queues pre-empting all other packets that exists in the queue as these data needs to be notified to the sink immediately.

**Packet Priority:** Each data packet has a “Priority ID ( $P_{ID}$ )”. Priority of data packets is set as per its type. I.e.,

```
if (TID = 1) then  
    PID = 1; //  
Priority is normal  
else if (TID = 0) then  
    PID = 0; //  
Priority is high  
else if (TID = -1) then  
    PID = -1; //  
Priority is emergency  
end
```

**Step 3:** Based on events, SC segregates data packets by placing them in most suitable queues as per the case scenarios referred below.

**Case 1:** When a packet ‘ $p_1$ ’ is identified by SC to have a  $P_{ID} = 1$ , it considers it as a normal packet of low priority and initiates data aggregation process. During the aggregation process, SC accumulates and packages such low priority packets (preventing duplicates) for a time duration of ‘ $t_1$ ’secs. The time duration varies as per the application requirement. Aggregated data packets are then placed in low priority queue ( $LP_{queue}$ ), processed and delivered to the sink.

**Case 2:** When a packet ‘ $p_1$ ’ is identified by SC to have a  $P_{ID} = 0$ , it considers it as an extreme packet of high priority and places it directly into the high priority queue ( $HP_{queue}$ ). The packet ‘ $p_1$ ’ starts execution priority packet and carries on even if similar higher priority packet ‘ $p_2$ ’ arrives at the  $HP_{queue}$ . I.e., packet ‘ $p_2$ ’ has to wait in the  $HP_{queue}$  until the execution of packet ‘ $p_1$ ’ is complete.

**Case 3:** When a packet ‘ $p_1$ ’ is identified by SC to have a  $P_{ID} = -1$ , it considers it as an emergency high alert packet of highest priority. SC then pre-empties all other packets intuitively from  $HP_{queue}$  and  $LP_{queue}$ , ensuring

packet ‘ $p_1$ ’ to be processed with highest priority and delivered to the sink with a minimum possible end-to-end delay.

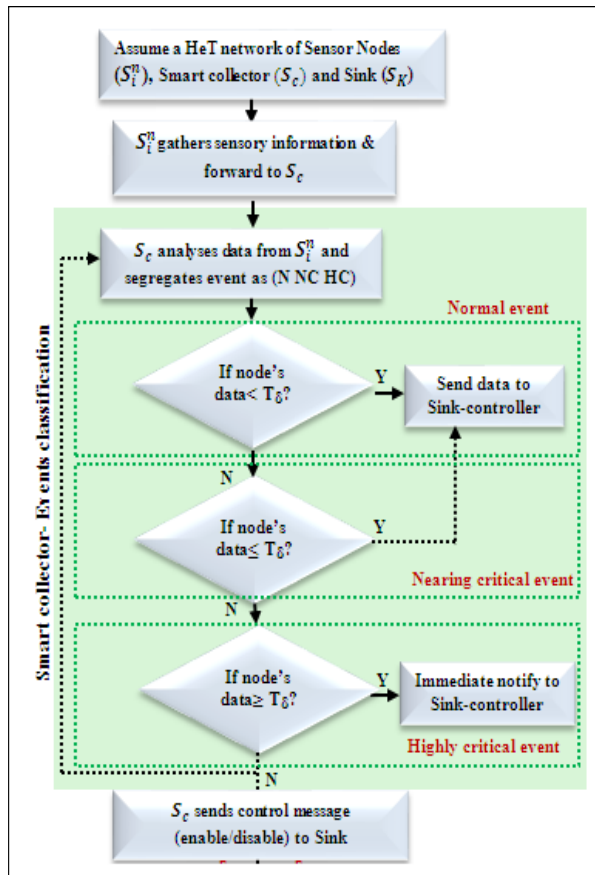
Algorithmic step that depicts the packet processing mechanism in Pro\_ET is referred in Algorithm 2:

#### Algorithm 2:

```
Collect data from sensors;  
Set TID and PID for each data packets;  
DPcount = getDataPacketCount();  
i=0;  
While (i<=DPcount)  
if (PID = 1) then  
    Datalist = getDataPacket(i);  
    aggregateDataPacket = ignoreDuplicateData(Datalist);  
    Send aggregateDataPacket to LPqueue;  
else if (PID = 0) then  
    dataPacket = getDataPacket(i);  
    Send dataPacket to HPqueue;  
  
else if (PID = -1) then  
    dataPacket = getDataPacket(i);  
    preemptive LPqueue and HPqueue Queue;  
    Send dataPacket to both HPqueue and LPqueue Queue for  
    processing;  
end  
end
```

Resource constraints on sensor nodes often make it undesirable to implement expensive algorithm. Pro-ET is designed very carefully so that it is efficient for resource-constrained sensor nodes to operate energy effectively. Although each sensor node is pre-equipped with Pro\_ET capabilities, each of which corresponds to a particular event for handling data packets as shown in Figure-2. Only node which is elected as SC is permitted by sink to execute it. All other sensors act as normal nodes to sense and transmit data packets to its SC node. For instance, the SC node’s behaviour is customized to detect a critical emergency event [6] when the temperature data value rises above critical level, under such occasions, the SC node responds by notifying the sink regarding its criticality. Using the Pro-ET capability the SC decreases its sampling (and transmission) rate to conserve the remaining battery life. By default, an un-programmed node transmits every sample acquired back to the sink at a fixed sample rate, so the battery life of a node is directly linked to its sample rate. Working principle of Pro\_ET mechanism uses dynamic priority based packet scheduling scheme that ensures a trade-off between fairness and priority. Fairness ensures that packets of different priorities get carried out with a minimum waiting time at the WFQ based on its priority.

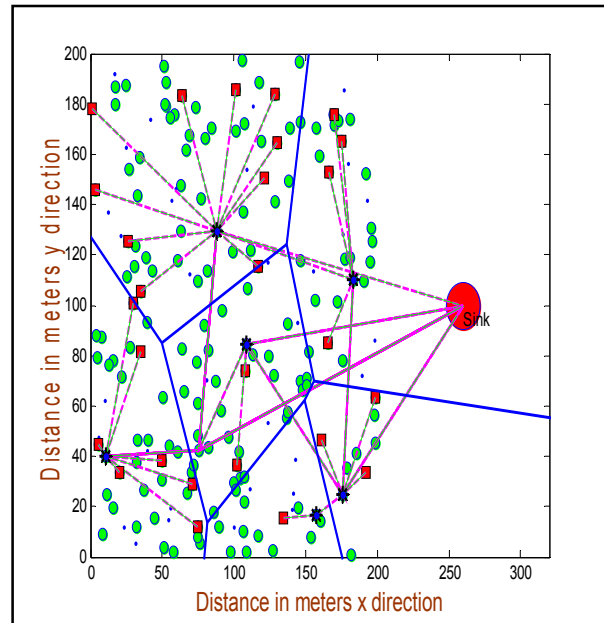




**Figure-2.** Flow process of event classification using Pro\_ET in Smart Collector.

## RESULTS AND DISCUSSIONS

This section evaluates the performance of Pro\_ET scheme using simulation model developed using MATLAB. Simulation setup considers a Wireless Sensor Network 10 X 10 km area, where sensor nodes varying from 100 to 500 are randomly deployed. Smart Collector is selected among group of sensors within a particular sub-region. Sink is introduced away from the deployed location field as displayed in Figure-3.



**Figure-3.** Simulation model of Pro\_ET Scheme.

Simulation was triggered considering randomly chosen multiple sensor nodes to act as source that tends to generate data packets and let it send it to its Smart collector node on varied time slot. Simulation results were analyzed for all types of data traffic (real-time emergency data and normal data packets). The performance of the proposed Pro\_ET scheme is compared against existing Multilevel Queue (MLQ) [11] scheduling and First Come First Served (FCFS) [19] schemes. Simulation parameters and their values used during simulation are presented in Table-1.

**Table-1.** Simulation parameters and their values.

Parameter	Value
Network Size	10 km x 10 km
Number of Nodes	100 - 500
Initial Energy of Node ( $E_0$ )	2 Joules
Energy consumed for Transmission	50 nJoule/bit
Energy consumed in air or free space	0.01 nJoule/bit/m <sup>2</sup>
Transmission Speed	100Kbps

**Delivery rate:** For analyzing the packet delivery rate, data traffic with diversified data packets (consisting all three types of data packets such as normal, extreme and emergency packet) were generated from different source node and transmitted to the smart collector at varied time slot. Smart collector using Pro\_ET scheme was able to dynamically fill packets to appropriate queues either the low or the high priority queue for data processing. Results captured during simulation is depicted in figure 4 indicates that Pro\_ET can reach desired delivery rate (99%) earlier



than other existing FCFS and MLQ schemes. Dynamic packet handling mechanism using Pro\_ET was able to ensure balanced trade-off between fairness and priority in packet processing (with low latency). This ensures higher chances for a packet to get delivered to the sink successfully. Additionally, even when the network density reaches beyond a certain upper limit, Pro\_ET can guarantee the desired deliverable rate of 99%. This is because even with increase in network density, multiple (low and high priority) queues for handling low, high and emergency packets assures 99% successful delivery on emergency packets, though the delivery rate could be low for low priority packets on time critical events.

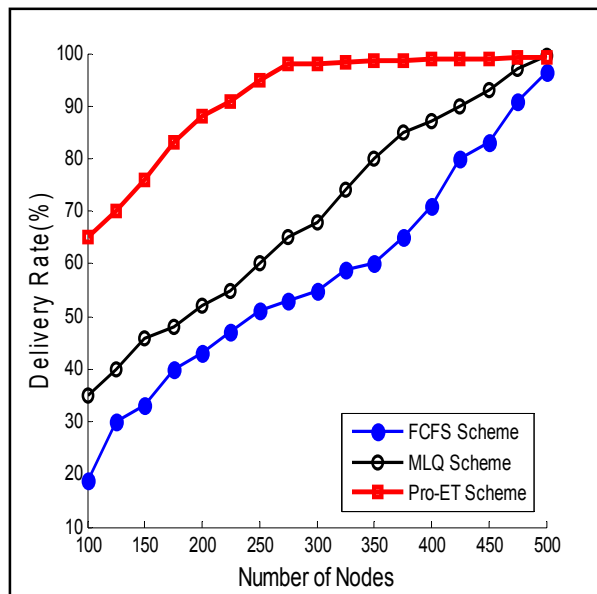


Figure-4. Average delivery rate.

This makes the Pro\_ET scheme more suitable for time-critical application guaranteeing emergency packet delivery on time.

**Communication latency:** Latency is the time required from sending a request (from sensor node) to confirming the response (smart collector) between two peers. This metric is vital for time-critical applications such as forest fire detection [4] etc. To estimate communication latency, the time which is spent from sensor node to the sink ( $CLatency_{snode\_sink}$ ) is calculated.

$$CLatency_{snode\_sink} = CLatency_{snode\_SC} + CLatency_{SC\_Sink}$$

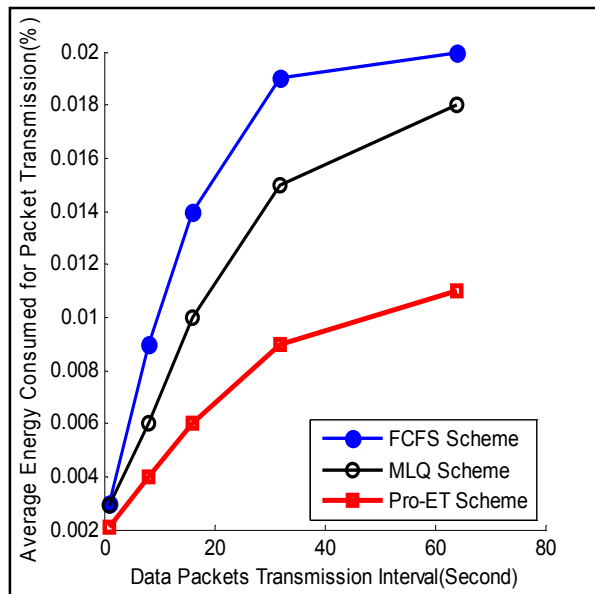
Where,  $CLatency_{snode\_SC}$  indicates the time which is spent from sensor node to smart collector and  $Latency_{SC\_Sink}$  indicates the time which is spent from smart collector to sink. To compute communication latency in simulation model, MATLAB script was employed to track the time taken between each requests and response. From the results observed the proposed Pro\_ET scheme achieves lower latency outperforming other existing schemes. Table-2 displays the communication latency of Pro\_ET, FCFS and MLQ schemes. According to our analysis, Pro\_ET takes up to ~ 6 sec for complete communication, while existing FCFS and MLQ requires upto ~13 sec and ~11 sec for complete communication. As shown in the Table-2, for Pro\_ET, the latency required for communicating between the sensor node and the smart collector was observed to be 1.912 sec and that between smart collector and sink was ~ 4 sec. Whereas for existing FCFS and MLQ schemes it was observed to be 6.009 sec and 3.98 sec for communicating between the sensor node and the intermediate nodes and sec and ~ 7 sec latency time for communicating between intermediate node or gateway and the sink.

Table-2. Performance comparison with proposed ILP-LBDP and MSLBR scheme.

Scheme	$CLatency_{snode\_SC}$	$CLatency_{SC\_Sink}$
Existing FCFS	6.009 sec	~ 7 sec
Existing MLQ	3.98 sec	~ 7 sec
Proposed Pro_ET	1.912 sec	~ 4 sec
Proposed Pro_ET improvements(%)	~50%	~35%

Overall ~ 4 to ~ 6 secs of end-to-end latency is being reduced per packet transmission in Pro\_ET, resulting approximately of about ~40% improvement compared to the existing approaches. This is because, Pro\_ET ensures reorganizing based on priorities, processing and transmitting only relevant data (preventing duplicate data) to the sink. It allows real-time data packets with highest priority (pre-empts the low priority packets and permit highest priority packets processing) to reach the sink with reduced latency during critical events.

**Average energy consumption:** The percentage of average consumed energy for packet transmission towards the sink node through different protocols under the simulation scenario is displayed in the Figure-5. As can be seen from this figure, Pro\_ET reduces the consumed energy for packet transmission up to 30% to 40% compared to the FCFS and MLQ schemes.



**Figur-5.** Average consumed energy for packet transmission towards the sink node by Pro\_ET, FCFS and MLQ.

The reason being, number of attempts taken for data packet to get delivered successfully to the sink is high for existing schemes compared to the proposed scheme. In Pro-ET, dynamic WFQ mechanism at smart collector collects, aggregates and handles packets (based on its type and priority – i.e., high priority packets are delivered faster compared to low priority which has the capability to survive longer duration) appropriately making it possible to deliver them successfully during its first attempt as compared to other existing schemes where the probability of data packet being transmitted to sink in its first attempt either fails or drops due to congestion or delay caused during its traversal (causing retransmission).

## CONCLUSIONS

After thorough research, the key problem of solving the packet drop rates and improving packet deliveries lies on having a superior and versatile packet scheduling mechanism with intelligent packet routing using smart collectors (a dynamic data collection and aggregation node). The proposed mechanism cut across the problem in a 3 dimensional way using a novel protocol - ProET, which uses reconfigurable and self organizing smart collectors. These nodes employ an improved WFQ mechanism for dynamic packet scheduling to switch on priority based on time and event functions. Two main concepts have been postulated in this work, one is the grooming of smart collector and its selection parameters in a randomly deployed WSN and the other is the soft computed protocol Pro\_ET for packet scheduling within the SC. The scheme uses multiple queues to schedule packets based on their type and priority. Efficacy of the model ensures reduced latency with high level of delivery ratio compared to other existing schemes such as FCFS

and MLQ. Under continuous experimental analysis of this protocol, it has shown a better ways of successful data gathering, improvement of network life time in terms of energy dissipation and overall performance optimization in terms of throughput, delay and overheads. The future scope of our work lies in bringing machine learning techniques inspired with biological behaviour algorithms to classify the events and prioritize the services for complete automation.

## REFERENCES

- [1] Alnuaimi Mariam, Khaled Shuaib, Klaitheem Alnuaimi and Mohammed Abed-Hafez. 2014. Clustering in Wireless Sensor networks based on Node Ranking. In Wireless Communications and Mobile Computing Conference (IWCMC) IEEE. 488-493.
- [2] Dawei Gong and Yuanyuan Yang. 2014. Low-Latency SINR-Based Data Gathering in Wireless Sensor Networks. IEEE Transactions on Wireless Communications. 13(6): 3207-3221.
- [3] Dezfouli B, Radi M, Razak SA, Whitehouse K, Bakar K A, Hwee-pink T. 2014. Improving broadcast reliability for neighbor discovery, link estimation and collection tree construction in wireless sensor networks. Computer Networks. 62: 101-121.
- [4] Díaz-Ramírez A., L. A. Tafuya, J. A. Atempa and P. Mejía-Alvarez. 2012. Wireless sensor networks and fusion information methods for forest fire detection. Procedia Technology. 3: 69-79.
- [5] Fatma B., Nizar B., R. Boutaba. 2013. Efficient reporting node selection-based MAC protocol for wireless sensor networks. Elsevier J. Wireless Network. 19(3): 373-39.
- [6] Guo,P., T. Jiang, Q. Zhang, and K. Zhang., 2012. Sleep scheduling for critical event monitoring in wireless sensor networks. IEEE Trans. Parallel Distrib. Syst. 23(2): 345-352.
- [7] Gungor V., L. Bin and G. Hancke. 2010. Opportunities and Challenges of Wireless Sensor Networks in Smart Grid. IEEE Transactions on Industrial Electronics. 57(10): 3557-3564.
- [8] HuanKe Song Guo and Toshiaki Miyazaki. 2014. Towards Latency-Aware Data Acquisition in Wireless Sensor Networks. IEEE 8th International Symposium on Embedded Multicore/ManycoreSoCs. pp. 82-87.





- [9] Jin Wang, Xiaoqin Yang, Zhongqi Zhang, Bin Li and Jeong -Uk Kim. 2014. A Survey about Routing Protocols with Mobile Sink for Wireless Sensor Network. *International Journal of Future Generation Communication and Networking*. 7(5): 221-228.
- [10] Kui X., Y. Sheng, H. Du, J. Liang. 2013. Constructing a CDS-Based Network Backbone for Data Collection in Wireless Sensor Networks. *International Journal of Distributed Sensor Networks*, Article ID 258081.
- [11] Lee E. M., A. Kashif, D. H. Lee, I. T. Kim and M. S. Park. 2010. Location based multi-queue scheduler in wireless sensor network. *International Conf. Advanced Commun. Technol.* 1: 551-555.
- [12] Miao Zhao, Ji Li and Yuanyuan Yang. 2014. A Framework of Joint Mobile Energy Replenishment and Data Gathering in Wireless Rechargeable Sensor Networks. *IEEE Transactions on Mobile Computing*. 13(12).
- [13] Mohammadreza Soltani, Michael Hempel, and Hamid Sharif. 2014. Data Fusion Utilization for optimizing Large-Scale Wireless Sensor Networks. *IEEE ICC - Ad-hoc and Sensor Networking Symposium*.
- [14] Nidal Nasser, Lutful Karim, and Tarik Taleb. 2013. Dynamic Multilevel Priority Packet Scheduling Scheme for wireless sensor networks. *IEEE Transactions on wireless communications*. 12(4).
- [15] Nazir B. and H. Hasbullah. 2011. Dynamic sleep scheduling for minimizing delay in wireless sensor network. *Saudi International Electron. Communications Photon. Conf.* pp. 1-5.
- [16] Nuaimi, Mariam Al, Khaled Shuaib and Klaithem Al Nuaimi., 2014. Clustering in WSN Using Node Ranking with Hybrid Nodes Duty-Cycle and Energy Threshold. In *Network Computing and Applications (NCA) IEEE*. pp. 245-252.
- [17] Pantazis N. A., S.A. Nikolidakis, D.D. Vergados. 2013. Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey. *IEEE Communications surveys and Tutorials*. 15(2).
- [18] Shan F., W. Liang, J.L., X. Shen. 2013. Network lifetime maximization for time-sensitive data gathering in wireless sensor networks. *Elsevier Journal, Computer Networks*. 57: 1063-1077.
- [19] Stallings W. 1995. *Operating Systems*, 2nd edition. Prentice Hall.
- [20] Swain A. R., R. C. Hansdah, and V. K. Chouhan., 2010. An energy aware routing protocol with sleep scheduling for wireless sensor networks. *IEEE International Conf. Adv. Inf. Netw. Appl.*, pp. 933–940.
- [21] SaeidMottaghia and Mohammad Reza Zahabi. 2015. Optimizing LEACH clustering algorithm with mobile sink and rendezvous nodes. *Int. J. Electron. Commun. (AEÜ)*, Elsevier. 69(2).
- [22] Shiliang Xiao, Baoqing Li, Xiaobing Yuan., 2015. Maximizing precision for energy-efficient data aggregation in wireless sensor networks with lossy links. *Ad Hoc Networks*, Elsevier. Vol. 26.
- [23] Suraj Sharma and Sanjay Kumar Jena. 2014. Data Dissemination Protocol for Mobile Sink in Wireless Sensor Networks. *Journal of Computational Engineering*. 2014(Article ID 560675).
- [24] Zhao Y., Q. Wang, W. Wang, D. Jiang and Y. Liu. 2009. Research on the priority-based soft real-time task scheduling in TinyOS. *International Conf. Inf. Technol. Computer Science*. 1: 562-565.