Efficient Packet Scheduling in Real Time WSN

Abhimanyu Dalal
M.tech Scholar, ECE Department
Vaish College of Engineering

Pawan Kumar
Asstt. Proff. In ECE Department
Vaish College of Engineering

***Abstract*— The research modifies the existing real time packet scheduling. In the existing real time packet scheduling, the intermediate node is selected on the basis of the slack time and the remaining distance from destination. The source node will transmit the data to the selected node then this selected node to other intermediate node. This process goes on until destination reached. But the entire data packets between same source and destination follow the same path in the existing scheme; this creates congestion in the network. The proposed technique handles this kind of congestion by the acknowledge time. If the transmitting node doesn't receive the acknowledgement within the threshold period of time then the data is retransmitted through another path. The proposed work is implemented using the NS2 and the results are analyzed by varying the nodes and the area. The simulation result confirms the better performance of the proposed work.**

Keywords*— Wireless Sensor Network, Real Time Packet Scheduling, Effective Real-Time Scheduler.

I. INTRODUCTION

A wireless sensor network consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions such as pressure, temperature, sound, vibration motion or pollutants. WSN is used to locate not only the objects whose area of location is known but also the objects whose location is anticipated to be around a certain domain. Each node in a sensor network is typically equipped with a radio receiver, a small micro controller, energy source usually a battery. Sensor networks can be used for target tracking, system control and chemical and biological detection. In military application's sensor, networks can enable soldiers to see around corners and to detect chemical and biological weapons long before they get close enough to cause harm them. Civilian uses include; environmental monitoring, traffic control and providing health care monitoring for the elderly while allowing them more freedom to move around. Sensor networks are typically characterized by restricted power supplies, low bandwidth, small memory size and limited energy. This leads to a very demanding environment to provide security. Radio frequency communication is used in sensor networks for communication between sensor nodes. So security of this broadcast communication is of paramount importance and one of the difficult issues to resolve. In a broadcast medium it is easy to intercept, eavesdrop, inject and change transmitted data .Sensor network installations may be done on an insecure setting; enemies can steal nodes, hack cryptographic material and pose as the authorized nodes. Sensor networks can also be pushed to resource consumption attack. This means enemies would send data to drain a node battery and reduce network bandwidth [1].

WSN differs dramatically from the traditional RT systems due to its wireless nature, limited resources (power, processing and memory), low node reliability and dynamic network topology [2]. Thus, developing real-time applications over WSN should consider not only resource constraints, but also the node and communication reliability and the globally time varying network performance. Very little prior work can be applied directly.

Real-time (RT) wireless sensor systems have many applications especially in intruder tracking, fire monitoring, medical care and structural health diagnosis [3, 4]. In intruder tracking, surveillance may require the position of an intruder be reported to a command center within 15sec so that pursuing actions can be initiated in time [5]. Data in the same system may have different deadlines due to different requirements. For example, locations of tanks have shorter update deadlines than those of pedestrians [5]. On the fire monitoring side, applications of sensor networks are numerous.

II. EFFECTIVE REAL-TIME WIRELESS SENSOR NETWORKS (ERT-WSNS)

Wireless Sensor Networks [WSNs] can be considered as distributed computing communication infrastructure with many stern constraints in terms of memory, power, size, bandwidth and processing capabilities [6] . WSN having its intrinsic unique promising features such as small size, low power consumption, autonomous, mobility, dense in volume; self-healing and self-organizing ; finds an immense range of applications that includes real-time applications. Real-Time Wireless Sensor Networks (RT-WSNs) applications are performance critical that require delimited latency service. Applications such as radiation monitoring, intrusion detection, fire detection to name but a few, where the sensed information has to reach the destination within the stipulated time, otherwise it may lead to catastrophic effect . The focal task of achieving real time data communication in case of WSN is to minimize the End-to-End delay. This End-to-End delay is the aggregation of all the possible delays incurred in between source and destination namely: communication delays, processing delay and queuing delay. In case of fewer traffic conditions, communication and processing delays dominate the queuing delay, and hence it shows a less impact on End-to-End delay. Whereas, in heavy traffic conditions, packets at

intermediate nodes experiences substantial amount of queuing delay which is not ignorable? Therefore, an effective packet scheduling policy is obligatory to minimize the queuing delay at the intermediate nodes and thus plays an imperative role in meeting the packet deadlines.
In existing work, proposed an Effective Real-Time packet Scheduling policy [ERTS] and presented a performance analysis in terms of Average delay. Packet Drop Ratio [PDR], Packet Miss Ratio [PMR].

A. Effective Real-Time Packet Scheduling Policy

This section introduced an Effective Real-Time Scheduling policy based upon the work of JiTS with a shortest path routing protocol. The shortest path routing algorithm used here selects the shortest path among all paths. The existing real time scheduler aids in success of real time data communication between the source and destination by minimizing the queuing delay at each intermediate node. The existing scheduler can be sandwiched between routing and MAC layer or within the routing layer of the protocol stack. Since it is designed as a layer independent, there is no need for any support or changes to be done in lower layers. The overview of the proposed architecture is as follows:

i) Scheduler Framework

The existing Effective Real-Time Scheduler (ERTS) for Real-time data dissemination .judiciously schedules the all incoming packets at each intermediate node based on the remaining slack time and remaining distance to the destination. It takes a routing information and the required deadline for a packet and decides how long to keep a packet in a queue, while leaving enough time to meet the deadline. In actual sense, the existing scheduling scheme gives output as time to stay in queue. i.e. the packets having high remaining slack time can allow to stay in queue more time compared to the packet with less slack time; that allows the packets approaching the deadline will be send with less queuing delay. Scheduling packets in this way can reduce the delay due to the network aggregation and congestion. A single priority queue where priority will be assigned based upon the remaining slack time is used instead of using multiple priority queues as of VMS. The governing equation for the Target Delay of scheduler is given as follows:

$$\text{Target Delay} = \frac{(\text{NLRST} - \text{EETD})}{\text{Distance (X,sink)}} . \alpha \quad (1)$$

$$\text{NLRST} = 2^{-\text{ET}} . \text{RST} \quad (2)$$

NLRST is Non Linear Remaining Slack Time.
ET= Elapsed Time.
RST = Remaining Slack Time.
α = Safety Margin.
EETD= Estimated End-to-End Delay, which is an estimation of lower layer transmission delay.

$$\text{EETD} = \beta . \frac{\text{Distance (source .sink)}}{\text{one hop distnace}} \quad (3)$$

Where β is one-hop estimated transmission delay, which can be measured by periodically measuring the time between the packet transmission and acknowledgement. Normally in heavy traffic conditions, queuing delay dominates the lower layer transmission delay, so a precise estimation of EETD in not needed. From the above equation (1), researchers can clearly say that the Target Delay of a packet in a queue is directly proportional to the non-linear slack time. Scheduling the packets in this way helps in meeting their deadline by minimizing the queuing delay.

III. PROPOSED WORK

In the existing real time packet scheduling, the intermediate node is selected on the basis of the slack time and the remaining distance from destination. The source node will transmit the data to the selected node then this selected node to other intermediate node. This process goes on until destination reached. But the entire data packets between same source and destination follow the same path in the existing scheme; this creates congestion in the network. The proposed technique handles this kind of congestion by the acknowledge time. If the transmitting node doesn't receive the acknowledgement within the threshold period of time then the data is retransmitted through another path. The process is explained in the following algorithm:

A. Proposed Algorithm

1. The source node say S will broadcast data.
2. G= The group of nodes at one hop from source S will receive request
3. Select Node with minimum slack time and the minimum distance from destination.
4. Send request to selected node
5. if ack not received within threshold time
6. Then discard the node
7. Go to step 3
8 else
9. if selected node is not destination
10. Broadcast from selected node
11. Go to step 2
12. else
13. exit
14. end if
15. end if

This algorithm can be implemented using NS2.

IV. RESULTS

The research implements the proposed protocol by using NS2.33 which is installed. The simulation is analyzed over different scenarios having nodes 10, 50, 100 respectively.

V. PARAMETER ANALYSED

A. Packet Delivery Ratio (PDR)

The number of delivered data packet ratio to the destination and this also illustrates the level of delivered data to the destination.

∑ Number of packet receive / ∑ Number of packet send

B. *Routing Overhead*

The total number of routing packets transmitted during the simulation i.e. the sum of all transmissions of routing packets sent during the simulation.

Calculated as Routing Overhead= Σ Transmission of routing packets

C. *Loss Ratio*

The total number of packet drop to the total number of packet generated.

Loss ratio= Packet Drop/Packet generated.

TABLE I. Result Analysis Of Existing Method At 10 Nodes

Area(mxm)	Generated Packet	Received Packet	PDR	Loss Ratio	Routing Overhead
500X500	7390	7301	98.79	0.0120	1.012
1000X1000	164	148	90.24	0.0975	1.108

TABLE II. Result Analysis of Proposed Method At 10 Nodes.

Area(mxm)	Generated Packet	Received Packet	PDR	Loss Ratio	Routing Overhead
500X500	7519	7432	98.84	0.0115	1.011
1000X1000	166	150	90.36	0.0963	1.106

TABLE III. Result Analysis of Existing Method At 50 Nodes.

Area(mxm)	Generated Packet	Received Packet	PDR	Loss Ratio	Routing Overhead
500X500	7066	6929	98.06	0.0193	1.019
1000X1000	3527	3373	95.63	0.0436	1.045

TABLE IV. Result Analysis of Proposed Method At 50 Nodes.

Area(mxm)	Generated Packet	Received Packet	PDR	Loss Ratio	Routing Overhead
500X500	6721	6628	98.61	0.0138	1.014
1000X1000	3444	3315	96.25	0.0374	1.038

TABLE V. Result Analysis of Existing Method At 100 Nodes.

Area(mxm)	Generated Packet	Received Packet	PDR	Loss Ratio	Routing Overhead
500X500	7458	7366	98.76	0.0123	1.012
1000X1000	5851	5729	97.91	0.0208	1.041

TABLE VI. Result Analysis of Proposed Method At 100 Nodes.

Area(mxm)	Generated Packet	Received Packet	PDR	Loss Ratio	Routing Overhead
500X500	8075	8008	99.17	0.0082	1.008
1000X1000	4003	3845	98.05	0.0394	1.021

Fig. 1 and Fig. 2 both shows the graph of PDR over 500×500 and 1000×1000 area in the existing system(AODV) and in the proposed system(RMAODV) . Fig. 3 and Fig. 4 both shows the graph of Loss Ratio over 500×500 and 1000×1000 area in the existing system (AODV) and in the proposed system(RMAODV) . Fig. 5 and Fig.6 both shows the graph of Routing Overhead over 500×500 and 1000×1000 area in the existing system (AODV) and in the proposed system (RMAODV) . The graphs are drawn for the various numbers of nodes.

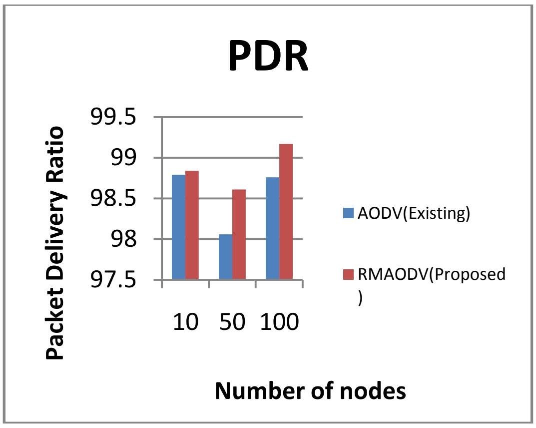

Fig. 1. Comparison of PDR at 500*500 area

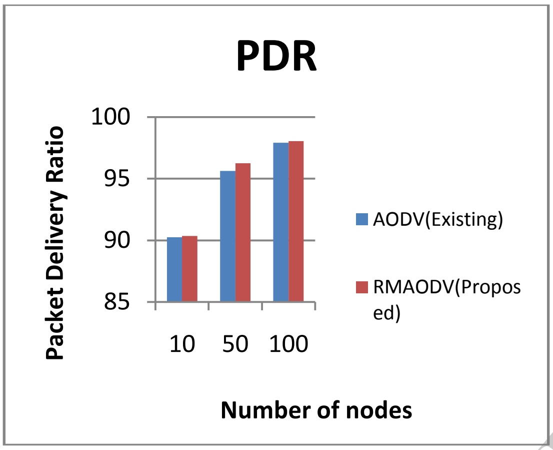

Fig. 2. Comparison of PDR at 1000*1000 area

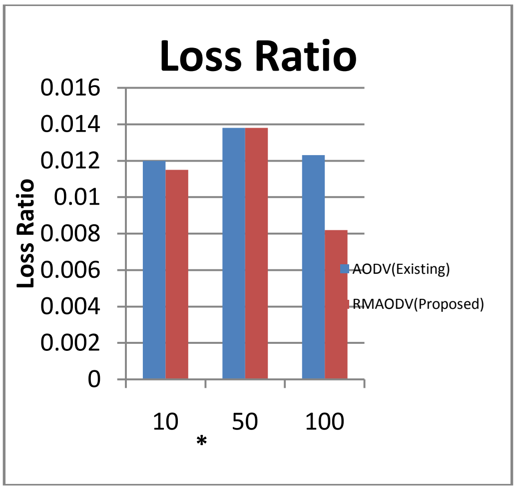

Fig. 3. Comparison of loss ratio at 500*500 area

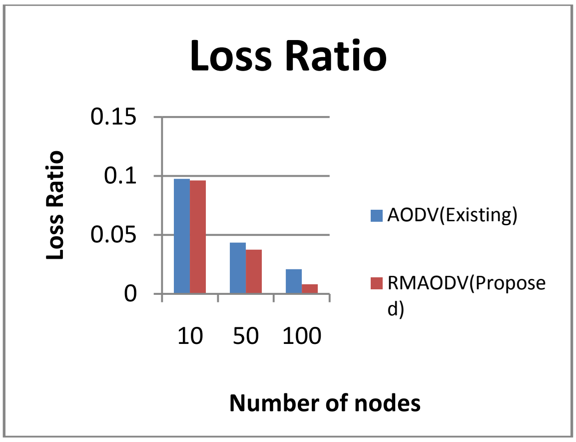

Fig. 4 Comparison of loss ratio at 1000*1000 Area

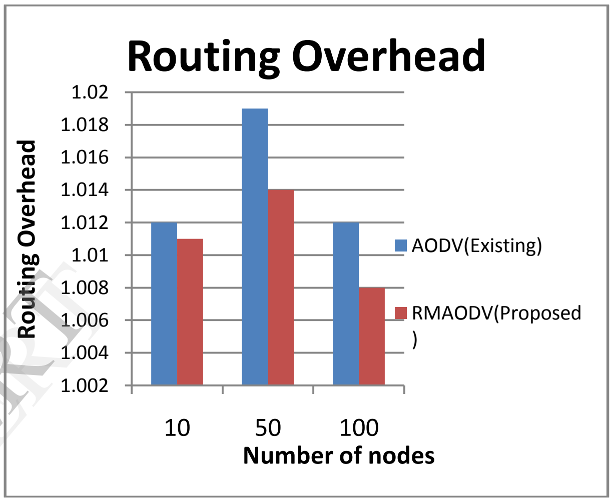

Fig. 5 Comparison of routing overhead at 500*500 area

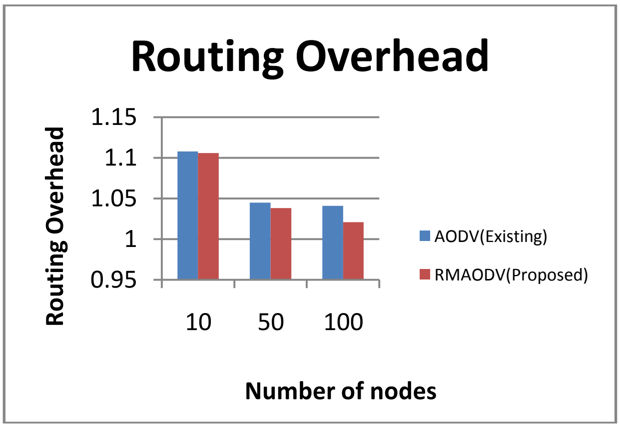

Fig. 6 Comparison of routing overhead at 1000*1000 area

VI. CONCLUSIONS

The research modifies the existing real time packet scheduling. In the existing real time packet scheduling, the intermediate node is selected on the basis of the slack time and the remaining distance from destination. The source node will transmit the data to the selected node then this selected node to other intermediate node. This process goes on until destination reached. But the entire data packets between same source and destination follow the same path in the existing scheme; this creates congestion in the network. The proposed technique handles this kind of congestion by the acknowledge time. If the transmitting node doesn't receive the acknowledgement within the threshold period of time then the data is retransmitted through another path. The proposed work is implemented using the NS2 and the results are analyzed by varying the nodes and the area. The simulation result confirms the better performance of the proposed work. In future following can be done: proposed protocol can be compared with other existing protocols., The proposed protocol can be enhanced in terms of security.

REFERENCES

[1] Manju V.C. ,"A Survey on Wireless Sensor Network Attacks", International Journal of Engineering and Innovative Technology (IJEIT) Volume 2, Issue 2, August 2012.

[2] Stankovic, J., T. Abdelzaher, C. Lu, L. Sha and J. Hou. “Real-time communication and coordination in embedded sensor networks”.Proc. IEEE 91(7), 1002–1022. (2003)

[3] Zhan, An-dong, et al. "A survey on real-time routing protocols for wireless sensor networks." Proceedings of The 2nd China wireless sensor network conference. 2008.

[4] LI Y, CHEN C and SONG Y et al., “Real-time QoS support in wireless sensor networks: a survey,” in the Proceedings of 7th IFAC Int Conf on Fieldbuses & Networks in Industrial & Embedded Systems (FeT’07) [C], Toulouse, France, Nov 2007.

[5] LU C, BLUM B and ABDELZAHER T et al., “RAP: A Real-time Communication Architecture for Large-Scale Wireless Sensor Networks,” in the Proceedings of the Eighth IEEE Real-Time and Embedded Technology and //Applications Symposium (RTAS’ 02) [C], 2002