An Efficient Collision Free Dynamic Multilevel Priority Packet Scheduling In WSN

Megha R[1]
Student, M.tech
Department of electronics and communication[1]
Adichunchanagiri Institute of Technology
Chikkamagaluru,India
rmeghaa1234@gmail.com

Dr.Jagadesh Chandra A.P[2]
Professor
Department of electronics and communication[2]
Adichunchanagiri Institute of Technology
Chikkamagaluru,India
apjchandra@gmail.com

***Abstract*— A wireless sensor network (WSN) is a computer network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions such as temperature ,sound, vibration , pressure , motion or pollutants, at different locations. The development of wireless sensor networks was originally motivated by military applications i.e. battlefield surveillance. However, wireless sensor networks are now used in many civilian application areas, including environment and habitat monitoring, healthcare applications, home automation, and traffic control. Scheduling different types of packets, such as real-time and non-real-time data packets, at sensor nodes with resource constraints in Wireless Sensor Networks (WSN) is of vital importance to reduce sensors' energy consumptions and end-to-end data transmission delays. Most of the existing packet-scheduling mechanisms of WSN use First Come First Serve (FCFS), non-preemptive priority and preemptive priority scheduling algorithms. These algorithms incur a high processing overhead and long end-to-end data transmission delay due to the FCFS concept, starvation of high priority real-time data packets due to the transmission of a large data packet in non preemptive priority scheduling, starvation of non-real-time data packets due to the probable continuous arrival of real-time data in preemptive priority scheduling, and improper allocation of data packets to queues in multilevel queue scheduling algorithms. Moreover, these algorithms are not dynamic to the changing requirements of WSN applications since their scheduling policies are predetermined. Real-time packets are placed into the highest-priority queue and can pre-empt data packets in other queues. Non-real-time packets are placed into two other queues based on a certain threshold of their estimated processing time. Leaf nodes have two queues for real-time and non-real-time data packets since they do not receive data from other nodes and thus, reduce end-to-end delay. We evaluate the performance of the proposed DMP packet scheduling scheme through simulations for real-time and non-real-time data.**

Keywords— Wireless sensor network, packet scheduling, pre-emptive priority scheduling, non-preemptive priority scheduling, real-time, non-real-time, data waiting time, FCFS.

I INTRODUCTION

A wireless sensor network (WSN) is a computer network which consists of spatially distributed autonomous devices using sensors to look after physical or environmental conditions like temperature, vibration, sound, pressure, motion or pollutants in various locations. Military applications gave motivation for the development of wireless sensor networks i.e. in battlefield. Now a days wireless sensor networks are used in many civilian applications, healthcare applications, home automation and in traffic control. Scheduling is the most widely used concept in WSNs because it determines the order of transmission of number of data packets based on their data priority and transmission deadline. For instance, real time data packets are given the highest priority when compared to that of non-real time data packets. Some of the available or existing scheduling mechanisms in wireless sensor networks are First Come First Serve, Premptive Priority and Non prepmtive Priority algorithms. The major drawbacks of using these algorithms are that the end-to-end transmission delay will be more and processing overhead will be high. Dynamic refers to the system which is active and undergoes progress frequently. Multilevel priority indicates that instead of single queue, multiple queues are used to assign different priorities to the incoming packet. Packet scheduling is the process used to select which packet to be serviced or which to be dropped based on the priority such as real time packet and non-real time packet. Packet scheduling can guarantee quality of service and improve transmission rate in wireless sensor networks The proposed scheme Dynamic Multilevel Priority (DMP) packet scheduling is for the processes where the inputs changes dynamically. In this scheme, zone based topology is used where the nodes are organized in virtual hierarchy. All the nodes except the last level has three different levels of priority queues. Real time data packets are placed into highest priority queue and can preempt the data packets in the other queues The leaf node has only two queues. One is for real time data packets and other is for non-real time data packets because it will not receive any data from lower level nodes. Hence this scheme reduces average waiting time and end-to-end transmission delay.

The remainder of the paper is organized as follows. In Section II, we discuss several existing WSN packet or task scheduling algorithms. Section III presents literature survey. Section IV presents the assumptions and protocol used.

Section V provides working principle of the proposed DMP packet-scheduling algorithm. Section VI provides methodology and system architecture of DMPPS .Section VII evaluates the performance of the DMP packet scheduling scheme through simulations and compares it against that of the existing FCFS and Multilevel Queue Scheduler algorithms. Finally, Section VIII concludes the paper defining some future research directions.

II RELATED WORK

A. Factor: Deadline

First Come First Serve: This may be the simplest way for a scheduler to schedule the packets. In fact, FCFS does not consider the QoS parameters of each packets, it just sends the packets according to the order of their arrival time. Thus, the QoS guarantee provided by FCFS is in general weak and highly depends on the traffic characteristic of flows. For example, if there are some flows which have very bursty traffic, under the discipline of FCFS, a packet will very likely be blocked for a long time by packets burst which arrives before it. In the worst case, the unfairness between different flows cannot be bounded, and the QoS cannot be no longer guaranteed. However, since FCFS has the advantage of simple to implement, it is still adopted in many communication networks, especially the networks providing best effort services. If some level of QoS is required, then more sophisticated scheduling algorithm is needed.

Earliest Deadline First (EDF): For networks providing real-time services such as multimedia applications, earliest deadline first (EDF) is one of the most well-known scheduling algorithms.The concept behind EDF is straightforward. It essentially schedules the packets in a greedy manner which always picks the packets with the closest deadline. Compare with strict priority discipline, we can regard EDF as a scheduling algorithm which provides time-dependent priority to each eligible packet

B Factor: Priority

Packet scheduling schemes can be classified based on the priority of data packets that are sensed at different sensor nodes.

Non-preemptive: In non-preemptive priority packet scheduling, when a packet t1 starts execution, task t1 carries on even if a higher priority packet t2 than the currently running packet t1 arrives at the ready queue. Thus t2 has to wait in the ready queue until the execution of t1 is complete.

Preemptive: In preemptive priority packet scheduling, higher priority packets are processed first and can preempt lower priority packets by saving the context of lower priority packets if they are already running.

B.Factor: Packet Type

Packet scheduling schemes can be classified based on the types of data packets, which are as follows. Real-time packet scheduling: Packets at sensor nodes should be scheduled based on their types and priorities. Real-time data packets are considered as the highest priority packets among all data packets in the ready queue. Hence, they are processed with the highest priority and delivered to the BS with a minimum possible end-to-end delay.

Non-real-time packet scheduling: Non-real time packets have lower priority than real-time tasks. They are hence delivered to BS either using first come first serve or shortest job first basis when no real-time packet exists at the ready queue of a sensor node. These packets can be intuitively preempted by real-time packets.

D. Factor: Number of Queue

Packet scheduling schemes can also be classified based on the number of levels in the ready queue of a sensor node. These are as follows. Single Queue: Each sensor node has a single ready queue. All types of data packets enter the ready queue and are scheduled based on different criteria: type, priority, size, etc. Single queue scheduling has a high starvation rate. Multi-level Queue: Each node has two or more queues. Data packets are placed into the different queues according to their priorities and types. Thus, scheduling has two phases: (i) allocating tasks among different queues, (ii) scheduling packets in each queue. The number of queues at a node depends on the level of the node in the network. For instance, a node at the lowest level or a leaf node has a minimum number of queues whilst a node at the upper levels has more queues to reduce end-to-end data transmission delay and balance network energy consumptions.

III LITERATURE SURVEY

A Dynamic Sleep Scheduling Protocol for Prolonging the Lifetime of Wireless Sensor Networks [1] consists of small and inexpensive sensor nodes. that have limited memory, limited computing power, and that operate using batteries. Since most of the time the batteries of sensor nodes are unchangeable and unrechargeable, the available energy in the batteries determines the lifetime of the sensor network. Therefore the battery energy of sensor nodes has to be very carefully and cleverly utilized. Additionally, it is also very important to balance the energy consumption of the nodes so that the network stay connected and functional for a long time. DSSP (Dynamic Sleep Scheduling Protocol), a centralized scheme for extending the lifetime of densely deployed wireless sensor networks by keeping only a necessary set of sensor nodes active. We present an algorithm for finding out which nodes should be put into sleep mode, and the algorithm preserves coverage and connectivity while trying to put as much nodes as possible into sleep mode. The algorithm is executed at the base station periodically. In this way, the network is reconfigured periodically, which also helps to a more even distribution of energy consumption load to sensor nodes. The major disadvantage in this approach is it may be hard to preserve connectivity and optimization can be more difficult. So, in our proposed system we can overcome this by applying the proper scheduling using the shortest job first(SJB) scheduler scheme with an advantage where broadcasting is done without collision and we can avoid sleeping nodes due to which efficiency increases. Another approach is ,Sleep scheduling for critical event monitoring in wireless sensor networks, [2] where we consider setting up sleep scheduling in sensor networks. We formulate the problem as an instance of the fractional dogmatic partition problem and obtain a distributed approximation algorithm by applying linear programming approximation techniques. Our algorithm is an application of the Garg-K¨ onemann (GK) scheme that requires solving an instance of the minimum weight dominating set (MWDS) problem as a subroutine. Our two main contributions are a distributed implementation of the GK scheme for the sleep-scheduling problem and a novel asynchronous distributed algorithm for approximating MWDS based on a primal-dual analysis of Chv´ atal's set-cover algorithm. The sleep-scheduling problem can be modeled using a pairwise redundancy relationship between sensor nodes. The major disadvantages of this scheme are it can't handle the heavy loads and the load balancing of energy sensors is poor. It is overcomed in proposed project due to dynamic routing where load can be handled. Advantages of proposed system are it can handle the heavy loads and the load balancing of energy sensors is more. Another existing approach is distributed wakeup scheduling scheme for supporting periodic traffic in WSN's [3]. In many applications in wireless sensor network source nodes generate and send periodic traffic to the sink through a number of forwarder nodes. In such a multihop network forwarders have forwarding duties but should, on the other hand, to spend as much time as possible in an energy-saving deep-sleep mode. In this work we explore decentralized approach to organize the wake-up process of the forwarders with the aim to let them wakeup at "just the right time" to catch an incoming packet, forward it and go quickly back to sleep mode. A key assumption for this work is that the forwarders do not know the traffic period beforehand, but they have to estimate the period and maintain their estimate over time. A key difficulty is that the period estimation and the scheduling of wakeup times will have to deal with jitter in the packet inter-arrival times. If a packet arrives before the forwarder wakes up, it is lost. Some of the drawbacks of this scheme are end to end delay is high and overhead is high. This can be overcome in our proposed scheme using this shortest job first (SJF) scheduler scheme where end-to-end delay is reduced. The major advantages of proposed system are end to end delay is less and overhead is very less.

IV PRELIMINARIES

In this section, we present general assumptions and deﬁne some terminologies that are used in designing the Dynamic Multilevel Priority (DMP) packet scheduling scheme.

A. Assumptions

We make the following assumptions to design and implement DMP packet scheduling scheme.

• Data trafﬁc comprises only real-time and non-real-time data, e.g., real-time health data sensed by body sensors and non-real-time temperature data.

• All data packets (real-time and non-real-time) are of same size.

• Sensors are time synchronized.

• No data aggregation is performed at intermediate nodes for real-time data.

• Nodes are considered located at different levels based on the number of hop counts from BS.

• Timeslots are allocated to nodes at different levels using TDMA scheme, e.g., nodes at the lowest level, lk are assigned timeslot 1. Details of timeslot allocation are explained in the "Terminologies" subsection.

• The ready queue at each node has maximum three levels or sections for real-time data (pr1) non-real-time remote data (pr2) and non-real-time local data (pr3).

• The length of data queues is variable. For instance, the length of real-time data queue (pr1) is assumed to be smaller than that of non-real-time data queues (pr2 and pr3). However, the length of the non-real-time pr2 and pr3 queues are same.

• DMP scheduling scheme uses a multichannel MAC protocol to send multiple packets simultaneously.

B. Protocol Used

The Zone Routing Protocol (ZRP) aims to address the problems by combining the best properties of both approaches. ZRP can be classed as a hybrid reactive/proactive routing protocol. ZRP reduces the proactive scope to a zone centered on each node. In a limited zone, the maintenance of routing information is easier. Further, the amount of routing information that is never used is minimized. Still, nodes farther away can be reached with reactive routing.Despite the use of zones, ZRP has a flat view over the network. In this way, the organizational overhead related to hierarchical protocols can be avoided. First hybrid routing protocol with both a proactive and a reactive routing component.IARP periodically computes the route to all intrazone nodes (nodes that are within the routing zone of a node) and maintains this information in a data structure called IARP routing table. In order to know about a node's direct neighbours and possible link failures, IARP depends on a neighbour Discovery Protocol (NDP) provided by the MAC layer. IERP is a family of reactive routing protocols like DSR or AODV that offer enhanced route discovery and route maintenance services based on local connectivity monitored by IARP.For route discovery by IERP, the notion bordercasting is introduced.

The bordercast packet delivery service is provided by the Bordercast Resolution Protocol (BRP)

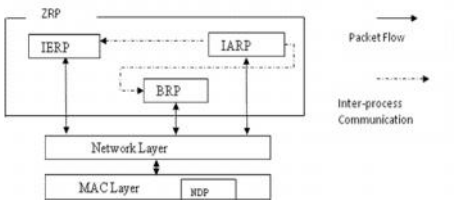

Fig 1 ZRP Architecture

V PROPOSED DYNAMIC MULTILEVEL PRIORITY PACKET SCHEDULING SCHEME

Among many network design issues, such as routing protocols and data aggregation, that reduce sensor energy consumption and data transmission delay, packet scheduling (interchangeably use as task scheduling) at sensor nodes is highly important since it ensures delivery of different types of data packets based on their priority and fairness with a minimum latency. For instance, data sensed for real-time we propose a Dynamic Multilevel Priority (DMP) packet scheduling scheme for WSNs in which sensor nodes are virtually organized into a hierarchical structure as shown in figure 3.1. Nodes that have the same hop distance from the BS are considered to be located at the same hierarchical level. We consider a network then divide it into zones. Each zone has a zone head which is used for routing. Zone head are used for routing data to the destination i.e. to the base station. Data are transmitted with the help of zone head. Other member nodes are not used for routing. They only transmit data to their zone head within the zone. Within a zone data are sending through intra-zone routing and outside the zone data are sending through inter-zone routing.

A .Dividing network into different zones

When the network is created next step is dividing it into different zones. We are considering varying size of zone. Its size can be increases or decreases as per requirement as when the routing path is larger zone size is increased and when routing path is smaller zone size is decrease. When zone size is small there will be more number of zones in a network and it will take more time to reach data to the destination(Base station) ,which may also cause path breakage. We increase zone size when routing path is larger so that data reach to the destination without causing any path break.

B .Creating zone head

Every zone has a zone head which is used for routing data to the destination. One node is considered as zone head and all other is simple node. These nodes sends data to their zone head and then zone head send the data to the base station. For to creating zone head we are considering mobility factor. Every node sends its mobility factor to another node within the zone and node which have highest mobility factor are zone head of the zone.

C . Routing in the network

After creating the zone head routing take place. Data are sending from source to the destination. All nodes within the zone send its data to its zone head. After receiving the data from the different nodes zone head communicate with base station directly or via other zone head by choosing the best path to the destination

D .Working Principle

The proposed scheduling scheme assumes that nodes are virtually organized following a hierarchical structure. Nodes that are at the same hop distance from the base station (BS) are considered to be located at the same level as shown in figure 5.1.

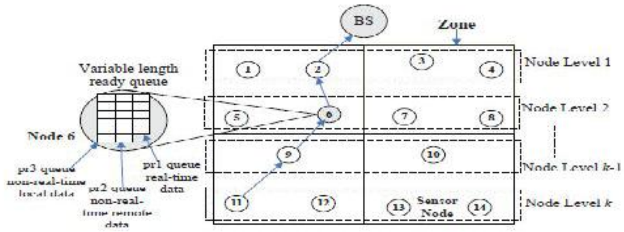

Fig 5.1 Dynamic Multilevel priority scheduling scheme

Data packets of nodes at different levels are processed using the Time-Division Multiplexing Access (TDMA) scheme. For instance, nodes that are located at the lowest level and the second lowest level can be allocated timeslots 1 and 2, respectively. We consider three-level of queues, that is, the maximum number of levels in the ready queue of a node is three: priority 1 (pr1), priority 2 (pr2), and priority 3 (pr3) queues. Real-time data packets go to pr1, the highest priority queue, and are processed using FCFS. Non-real-time data packets that arrive from sensor nodes at lower levels go to pr2, the second highest priority queue. Finally, non-real time data packets that are sensed at a local node go to pr3, the lowest priority queue as shown in below figure 3.2. The possible reasons for choosing maximum three queues are to process (i) real-time pr1 tasks with the highest priority to achieve the overall goal of WSNs, (ii) non real-time pr2 tasks to achieve the minimum average task waiting time and also to balance the end-to-end delay by giving higher priority to remote data packets, (iii) non-real-time pr3 tasks with lower priority to achieve fairness by preempting pr2 tasks if pr3 tasks wait a number of consecutive timeslots.

In the proposed scheme, queue sizes differ based on the application requirements. Since preemptive priority scheduling incurs overhead due to the context storage and switching in re- source constraint sensor networks, the size of the ready queue for preemptive priority schedulers is expected to be smaller than that of the preemptable priority schedulers. The idea behind this is that the highest-priority real-time/emergency tasks rarely occur. They are thus placed in the pre-emptive priority task queue (pr1 queue) and can pre-empt the currently running tasks. Since these processes are small in number, the number of pre-emptions will be a few. On the other hand, non- real-time packets that arrive from the sensor nodes at lower level are placed in the pre-emptable priority queue (pr2 queue). The processing of these data packets can be pre-empted by the highest priority real-time tasks and also after a certain time period if tasks at the lower priority pr3 queue do not get processed due to the continuous arrival of higher priority data packets. Real-time packets are usually processed in FCFS fashion. Each packet has an ID, which consists of two parts, namely level ID and node ID. When two equal priority packets arrive at the ready queue at the same time, the data packet which is generated at the lower level will have higher priority. This phenomenon reduces the end-to-end delay of the lower level tasks to reach the BS. For two tasks of the same level, the smaller task (i.e., in terms of data size) will have higher priority.Moreover, it is expected that when a node x senses and receives data from lower-level nodes, it is able to process and forward most data within its allocated timeslot; hence, the probability that the ready queue at a node becomes full and drops packets is low. However, if any data remains in the ready queue of node x during its allocated timeslot, that data will be transmitted in the next allocated timeslot. Timeslots at each level are not fixed. They are rather calculated based on the data sensing period, data transmission rate, and CPU speed. They are increased as the levels progress through BS. However, if there is any real-time or emergency response data at a particular level, the time required to transmit that data will be short and will not increase at the upper levels since there is no data aggregation. The remaining time of a timeslot of nodes at a particular level will be used to process data packets at other queues. Since the probability of having real-time emergency data is low, it is expected that this scenario would not degrade the system performance. Instead, it may improve the perceived Quality of Service (QoS) by delivering real-time data fast. Moreover, if any node x at a particular level completes its task before the expiration of its allocated timeslot, node x goes to sleep by turning its radio off for the sake of energy efficiency.

VI METHODOLGY OF THE PROPOSED SYSTEM

Methodology of the proposed system is as shown in the figure 6.1

A. Network Initialization

In the network design the nodes 'n' are deployed randomly in the networks. The node which is in the centre of the network is chosen as the base station. And the communication range of the node is set to 250m.Area where the nodes are deployed is divided into number of zones and for each zone ,zone head is choosen.

B. Packet Classification

Packets are classified based on the priority levels. Packet scheduling at sensor nodes is highly important since it ensures delivery of different types of data packets based on their priority and fairness with a minimum latency. For instance, data sensed for real-time applications have higher priority than data sensed for non-real-time applications. The processing of data packets available at a sensor node and also reduces energy consumptions

.

C. Task Scheduling

Allocation is done for task schedulers. Based on th priority queue packets are scheduled. Real time packets which are given the higher priority are scheduled using FCFS. Non real time packets with other two lower priorities are processed using SJF

D. Performance Evaluation

Evaluating the performance of this system gives positive results on minimum average waiting time and reduction in end-to-end delay while transmission. We compare results obtained from DMPPS with that of FCFS and obtain simulation results.

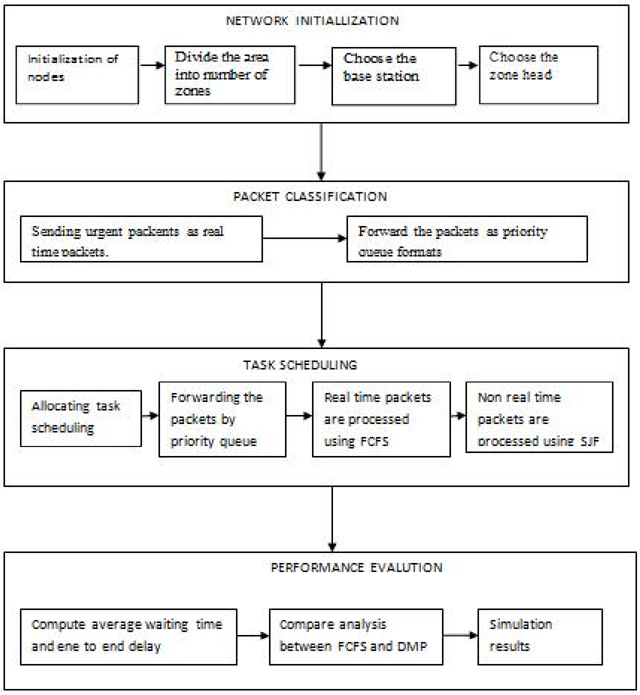

Fig 6.1 System Architecture of the DMPPS

VII. PERFORMANCE EVALUATION

The simulation model is implemented using the C programming language. It is used to evaluate the performance of the proposed DMP packet scheduling scheme, comparing it against the FCFS, and Multilevel Queue scheduling schemes. The comparison is made in terms of average packet waiting time, and end-to-end data transmission delay. We use randomly connected Unit Disk Graphs (UDGs) on a surface of 100 meter × 100 meter as a basis of our simulations. The number of simulated zones varies from 4 to 12 zones. Nodes are distributed uniformly over the zones. The ready queue of each node can hold a maximum of 50 tasks. Each task has a

Type ID that identifies its type. For instance, type 0 is considered to be a real-time task. Data packets are placed into the ready queue based on the processing time of the task. Moreover, each packet has a hop count number that is assigned randomly, and the packet with the highest hop count number is placed into the highest-priority queue. We run the simulation both for a specific number of zones, and levels in the network until data from a node in each zone or level reach BS. Simulation results are presented for both real-time data and all types of data traffic.

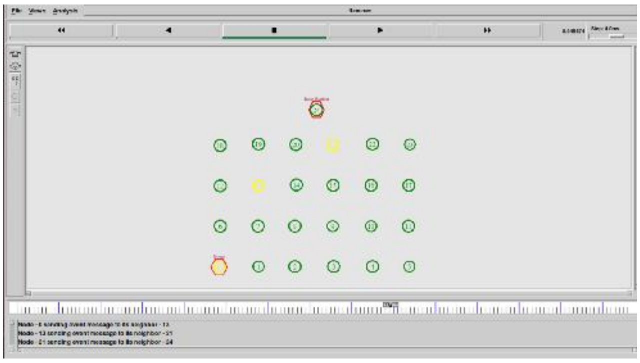

Fig 7.1 Final output obtained after scheduling packets

Figures 7.2 and 7.3 illustrate the end-to-end data transmission delay of real-time tasks over a number of zones and levels, respectively. In both case, we observe that the proposed DMP scheduling scheme outperforms the existing FCFS, and Multi-level Queue scheduler. This is because the proposed scheduling scheme gives the highest priority to real-time tasks and also allows real-time data packets to preempt the processing of non-real time data packets. Thus, real-time data packets have lower data transmission delays.

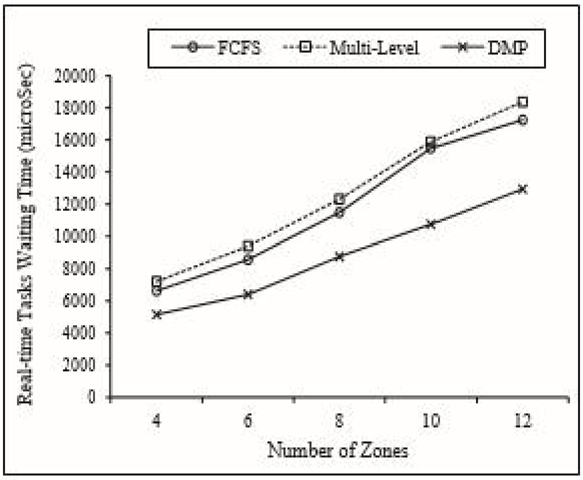

Fig. 7.2 End-to-end delay of real-time data over a number of zones

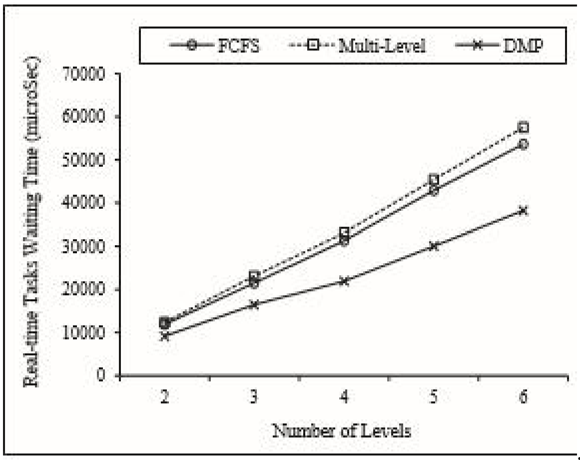

Fig. 7.3 End-to-end delay of real-time data over a number of levels

Similarly, Figures 7.4 and 7.5 demonstrate the end-to-end delay of all types of data traffic over a number of zones and levels, respectively. From these results,we find that the DMP task scheduling scheme outperforms FCFS, and Multilevel Queue scheduler in terms of end-to-end data transmission delay. This is because in the proposed scheme, the tasks that arrive from the lower level nodes are given higher priority than the tasks at the current node. Thus, the average data transmission delay is shortened

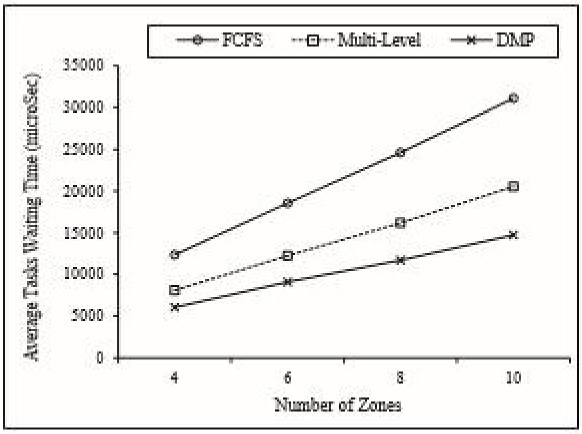

Fig. 7.4 End-to-end delay of all data packets over a number of zones

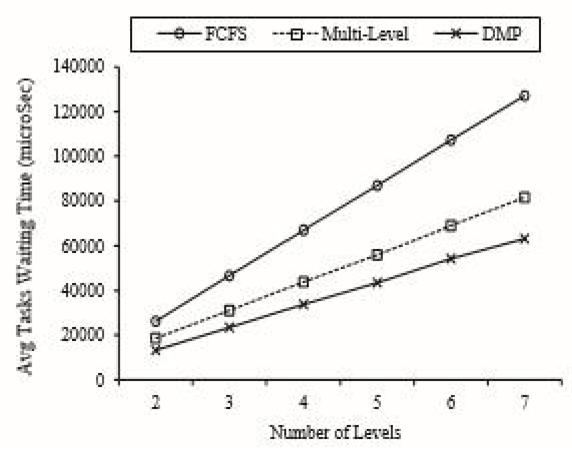

Fig. 7.5 End-to-end delay of all data packets over a number of levels

Figures 7.6 – 7.10 demonstrate that the DMP task scheduler has better performance than the FCFS, and Multilevel Queue scheduler in terms of average task waiting time, both for real-time tasks, and all types of tasks. We have already explained the possible reasons for this performance differences. We also perform student's t-test at a 95% confidence level and find the p-value to be less than 0.05 in most cases. This test validates our claim about the performance of the proposed DMP scheduling scheme. We also measure, and compare the fairness of executing non-real-time task in terms of the total waiting time of non- real-time tasks over total waiting time of all tasks. Figure 7.11 illustrates that the fairness index of DMP scheduling scheme is higher or better than that of the other two approaches. The number of levels in the network topology increases as the number of zones multiplies, which increases the average waiting time for non-real-time tasks over real-time tasks. Thus, the fairness index slightly decreases or remains almost same as the number of zones increases. Using the concept of three-level priority queues at each node, the proposed DMP task scheduling scheme allows different types of data packets to be processed based on their priorities. Since real-time, and emergency data should be processed with the minimum end-to-end delay, they are processed with the highest priority, and can preempt tasks with lower priorities located in the two other queues. On the other hand, in existing multilevel queue schedulers, a task with the highest hop count is given the highest priority. Hence, real-time tasks are prioritized over other task types only if their hop counts are higher than those of non-real-time tasks. Moreover, in FCFS and multilevel queue schedulers, the estimated processing time of a task is not considered when deciding the priority of a task. Thus, FCFS and Multilevel Queue schedulers exhibit longer task waiting times and end-to- end delays, in comparison to the DMP task scheduling scheme. Furthermore, the average waiting time of a task contributes largely to the experienced end-to-end data transmission delay.

In the DMP task scheduling approach, the source of a data packet is used to define the priority of data packets other than real-time. The priority of non-real time data packet will be more if it is sensed at remote node rather than the current sending node. Moreover, when no real-time tasks are available, pr3 tasks can preempt pr2 tasks if they are in starvation for a long time. This allows the processing of different types of tasks with fairness. The memory is also dynamically allocated to three queues and the size of the highest-priority queue is usually smaller than the two other queues since pr1 real-time tasks do not occur frequently compared to non- real-time tasks. As the memory capacity of a sensor node is limited, this also balances memory usages. Moreover, tasks are mostly non-real-time and are processed in the pr2 and pr3 queues. Non-real-time tasks that a node x receives from the lower level nodes are known as non-real-time remote tasks and processed with higher priority (pr2) than the non-real- time local tasks that x senses. Thus, non-real-time remote tasks incur less average waiting time. In addition, the average waiting time will not be affected for real-time tasks that are processed using FCFS scheduling, since these real-time tasks occur infrequently with a short processing time. Admittedly, one of the concerns regarding our proposed DMP task scheduling scheme pertains to its energy requirements. Indeed, the DMP task scheduling mechanism could be less energy efficient in comparison to the other two approaches since the DMP scheme requires a few more processing cycles to categorize and place the tasks into three different queues as well as for context saving and switching (for preemption). However, given the increased demand for WSN-based solutions that efficiently support real-time emergency applications and ensure them minimum average task waiting time and end- to-end delay, the proposed DMP task scheduling mechanism can be regarded as highly efficient.

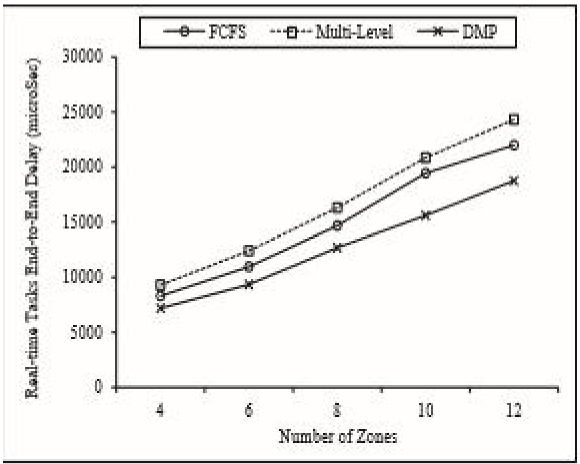

Fig 7.6 Waiting time of real-time data over a number of zones.

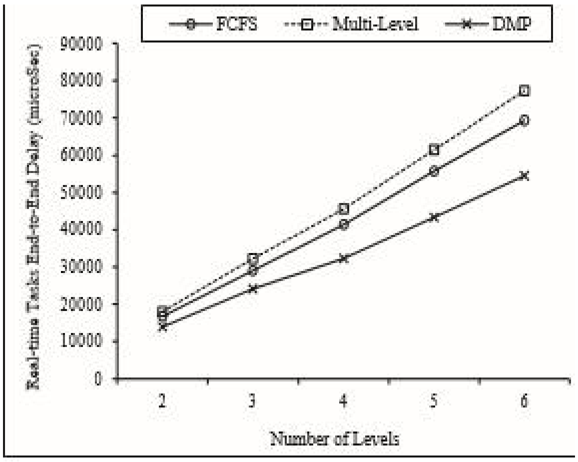

Fig 7.7 Waiting time of real-time data over a number of levels.

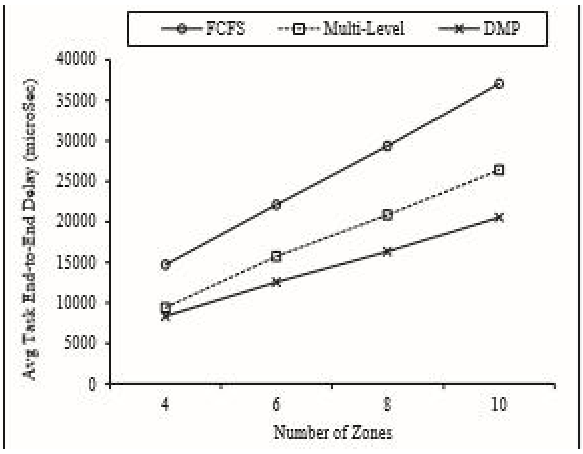

Fig 7.8 Waiting time of all types of data over a number of simulated zones

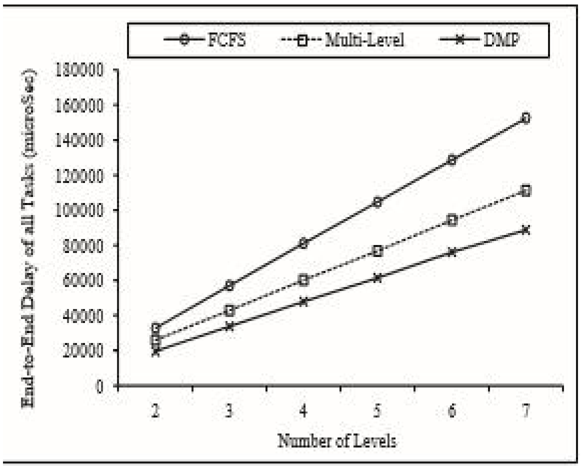

Fig 7.9 Waiting time of all types of data over a number of levels.

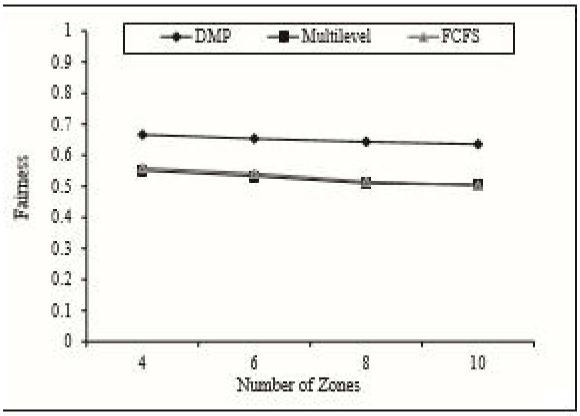

Fig 7.10 Waiting time of all types of data over a number of levels.

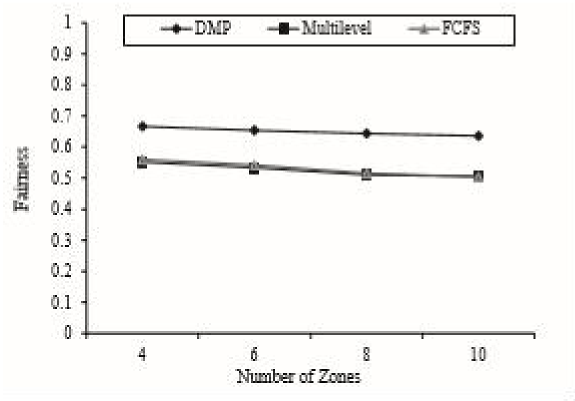

Fig 7.11Fairness in terms of the waiting time of non-real-time data.

VIII CONCLUSION AND FITURE WORK

We have discussed the hybrid Zone Routing Protocol in this paper, which is the combination of reactive and proactive routing protocols and have advantages of both type of protocols. The ZRP protocol is suitable for large networks and is not an independent protocol but rather a routing framework. It is especially well adapted to large networks and diverse mobility patterns and also we propose a Dynamic Multilevel Priority (DMP) packet scheduling scheme for Wireless Sensor Networks (WSNs). The scheme uses three-level of priority queues to schedule data packets based on their types and priorities. It ensures minimum end-to-end data transmission for the highest priority data while exhibiting acceptable fairness towards lowest-priority data. Experimental results show that the proposed DMP packet scheduling scheme has better performance than the existing FCFS and Multilevel Queue Scheduler in terms of the average task waiting time and end- to-end delay.

As enhancements to the proposed DMP scheme, we envision assigning task priority based on task deadline instead of the shortest task processing time. To reduce processing overhead and save bandwidth, we could also consider removing tasks with expired deadlines from the medium. Furthermore, if a real-time task holds the resources for a longer period of time, other tasks need to wait for an undefined period time, causing the occurrence of a deadlock. This deadlock situation degrades the performance of task scheduling schemes in terms of end-to-end delay. Hence, we would deal with the circular wait and preemptive conditions to prevent deadlock from occurring. We would also validate the simulation result using a real test-bed.

REFERENCES

[1] B. Nazir and H. Hasbullah, “Dynamic sleep scheduling for minimizing delay in wireless sensor network,” in Proc. 2011 Saudi International Electron., Communications Photon. Conf., pp. 1–5

[2] P. Guo, T. Jiang, Q. Zhang, and K. Zhang, “Sleep scheduling for critical event monitoring in wireless sensor networks,” IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 2, pp. 345–352, Feb. 2012.

[3] O. Khader, A. Willig, and A. Wolisz, “Distributed wakeup scheduling scheme for supporting periodic traffic in wsns,” in Proc. 2009 European Wireless Conf., pp. 287–292.

[4] Y. H. Wang, Y. L. Wu, and K. F. Huang, “A power saving sleep scheduling based on transmission power control for wireless sensor networks,” in Proc. 2011 International Conf. Ubi-Media Computer, pp. 19–24.