Deadline Perception with Enhanced Dynamic Multilevel Priority Scheduling Scheme for Wireless Sensor Networks

Amulya P[#1], Subba Raju KV[#2]

[#1]*Computer Science and Engineering, MVGR College of Engineering, India*

[#2]*Computer Science and Engineering, MVGR College of Engineering, India*

[1] amulya9120@gmail.com

[2]srkakarlapudi@gmail.com

Abstract**—The swift towards wireless sensor networks has motivated extensive research endeavours that target to reduce energy consumptions and end-to-end transmission delays. One such thread that aims to achieve is the packet scheduling algorithm. The multilevel priority packet scheduling schemes that target to reduce transmission delays, schedule data packets in a multilevel queue based on type of packet and in turn packets are scheduled within each level of queue based on shortest job first (SJF) and First Come First Serve (FCFS) to break the tie. The real difficulty with SJF algorithm lies in determining the processing time of next task. And FCFS increases transmission delays.**

To overcome the problems in the internal scheduling in multilevel queues in multilevel priority packet scheduling, we propose Deadline Perception with Enhanced Dynamic Multilevel Priority (DP-EDMP) packet scheduling. In the proposed scheme, data packets are first scheduled using type of packet among multiple levels of queue and then within each level of queue they are scheduled using Earliest Deadline First (EDF), thereby reducing average packet waiting time and end-to-end delay. Secondly, we remove the packets with expired deadlines from the medium which reduces data processing overhead and also saves energy consumptions of the sensor nodes. This facet is achieved by programming the sensor nodes to be intelligent enough to sense packets with expired deadlines.

Keywords**—energy consumptions, end-to-end delays, packet scheduling, SJF, FCFS,**

I. Introduction

Wireless sensor network (WSN) is an assortment of sensor nodes which possess sensing, processing and radio communication abilities, are strewn over a geographical region and collectively record the physical parameters in the region of interest, track certain events and route the sensed data to the base station (BS) if needed. In the course, the BS might also broadcast messages to every node, and sensor nodes would also communicate with every other sensor nodes too. During this exchange of data packets among the sensor nodes packet scheduling (interchangeably used as task scheduling) plays a crucial role as it reduces data transmission delay to a very large extent and sensor energy consumption to some extent. The scheduling schemes are extremely essential as they ensure delivery of various categories of packets based on priority and tolerance with minimum delay. Wide-ranging research has been carried on for scheduling the sleep-wake times of sensor nodes [1]-[18], very few works exist on packet scheduling in WSN [19]-[22]. Existing scheduling mechanisms like pre-emptive, non-pre-emptive priority algorithms incur high processing overhead, cause starvation of real-time and non-real time packets as well. In fact, most existing WSN environments use First Come First Serve (FCFS) [23] scheduling algorithm; which route data packets in the order of their arrival times which leads to long waiting times and long end-to-end delays. Dynamic Multilevel Priority (DMP) packet scheduling scheme [24] deals with the above mentioned problems, by maintaining multilevel queue. In this scheme sensor nodes are nearly prearranged into a tree like structure. Nodes that are located at equal hop count from the BS are assumed to be at the same hierarchical level. Each node maintains three levels of priority queues, this is because data is classified as (i) real-time priority (priority 1), (ii) non-real-time remote data received from other nodes (priority 2), and (iii) non-real-time data that is sensed at the node itself (priority 3).except the leaf nodes; which have only two levels because they do not get data from other nodes. DMP schedules non-real-time based on Shortest Job First (SJF) [34] and FCFS to break the tie. Real-time packets are scheduled using FCFS. And when packets of equal priority arrive at same time at the ready queue, the one which has come from lower level is given higher priority.

Two problems arise due to use of SJF and FCFS. One, the difficulty in determining the processing time of next task as a result of the former and the other is the increased end-to-end data transmission delay and high processing overhead due to the latter. Hence delivery of packets to BS is not sufficient, but in reality the packets must reach prior to the expiration of deadline. And even, real-time emergency data should be routed to the BS with the shortest possible delay. In order to solve the so discussed issues, in this paper we propose Deadline Perception with Enhanced Dynamic Multilevel Priority (DP-EDMP) packet scheduling scheme. The proposed scheme uses multilevel queue as in DMP [24], data packets are scheduled among the levels of queue using type of packet and then within each level using Earliest Deadline First (EDF) so as to lessen end-to-end transmission delay and also to make sure that packets reach the BS before the expiry of their deadline. The packet's whose deadline has been expired before reaching the BS, we consider to remove such by programming the sensor nodes to sense the medium for those packets. Such packets are dropped at the node where they are sensed, in contrast other scheduling schemes drop the packets only after they are received by the BS. By dropping packets with expired deadlines at the intermediate nodes, cuts network traffic and data processing overhead and also saves energy consumptions of the sensor nodes.

The remainder of the paper is structured as follows. In Section II, we talk over numerous existing WSN packet scheduling algorithms. Section III discusses overall assumptions. Section IV presents the mode of operation and pseudo-code of the proposed DP-EDMP packet-scheduling scheme. Section V evaluates the performance of the DP-EDMP packet scheduling scheme and compares it against that of FCFS and DMP scheduling algorithms [24]. As a final point, Section VI concludes the paper with some suggestions for forthcoming research.

II. RELATED WORK

There exist many packet or task scheduling algorithms, which are categorized based on numerous aspects as shown in Figure 1.

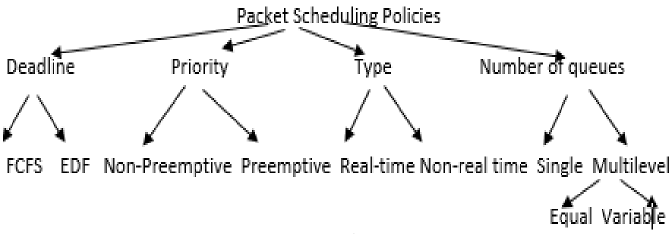

Fig. 1 Categorization of scheduling policies

A. Deadline

Packet scheduling schemes can be classified based on the arrival of data packets to the base station (BS). The following are such schemes.

First Come First Serve (FCFS): Many of the existing WSN applications use First Come First Serve (FCFS) schedulers which route data in the order of their arrival times at the ready queue. In FCFS scheme data packets that come from distant nodes experience long waiting times.

Earliest Deadline First (EDF): In this scheme, the data packet which has earliest deadline is routed first and this system is well-thought-out to be proficient in terms of average packet waiting time and end-to-end delay.

The scheduling scheme proposed by Lu C. et al. [25] is a real-time communication architecture applicable for large-scale sensor networks; uses a priority-based scheduler. The data packets at ready queue are prioritized based on distance from BS and EDF. If at all the deadline of a specific task expires, the related data packet is dropped at an intermediate node. Doing so reduces network traffic and also data processing overhead but is not resourceful as it devours memory and computational resources and even increases the processing delay. Mizanian et al. [26] proposed RACE, which is a scheduling scheme and as well as routing algorithm which discovers paths with least possible traffic load and delay between sensor nodes and BS using a loop-free Bellman-Ford algorithm. At the ready queue tasks are prioritized based on EDF.

B. Priority

Packet scheduling policies can be classified based on the priority of data packets which are sensed at the sensor nodes in a WSN environment.

Non-Pre-emptive: In non-pre-emptive priority packet scheduling, when a packet p_1 starts execution, it is carried on even if a higher priority packet p_2 than the packet that is currently under execution; here p_1 arrives at the ready queue. Thus, p_2 must wait till p_1's execution is completed.

Pre-emptive: In pre-emptive priority packet scheduling, higher priority packets are processed first and they can pre-empt lower priority packets if at all they are under execution.

Min Y.U. et al. [27] presented a packet scheduling policy that can be used in TinyOS [28], [29]. TinyOS is the widely used operative system in WSN. The proposed packet scheduling scheme is classified as either cooperative or pre-emptive. Cooperative scheduling mechanisms are more likely based on dynamic priority scheduling policy, such as EDF and Adaptive Double Ring Scheduling (ADRS) [30]. The scheduling policy practices two queues of different priorities, scheduler shifts dynamically between queues taking into interpretation the deadline of packets that have newly arrived. The decision of which packet to be placed into a queue is based on the deadline of the packets that are waiting at the ready queue. The one with shorter deadline is placed in the higher priority queue and the longer in the lower priority queue. On the other side, pre-emptive scheduling policy is based on the Emergency Task First Rate Monotonic (EF-RM) mechanism. Rate Monotonic (RM) is a static priority scheduling where in the shortest deadline job is agreed to be the higher priority one. Then comes the EF-RM, an extension of RM, in which tasks are alienated into Period Tasks (PT) their priorities are decided by RM strategy and higher priority non-period tasks (NPT) than PTs can interfere a running PT when necessitated.

C. Packet Type

Packet scheduling policies can be categorized based on the type of data packet, which are defined as follows.

Real-time packet scheduling: Packets at the sensor nodes can be handled based on their types and even their priorities as well. Real-time packets are given the highest priority amongst all the packets residing in the ready queue. So these are processed first and are transported to the BS with least possible end-to-end delay.

Non-real-time packet scheduling: Non-real-time packets are assigned lower priority than real-time packets. When no real-time packet resides in the ready queue, then the real-time packets are delivered to the BS based on either FCFS or shortest job first. And even these packets can be pre-empted by real-time packets. Zaho Y. [31] has proposed an improved priority-based soft real-time packet scheduling algorithm to overcome the drawbacks existing in packet scheduling mechanism of TinyOS. The latter schemes cannot be applicable to all applications because of long executions of certain packets and sometimes real-time packets suffer starvation. Most importantly the data queues are filled up very rapidly if the frequency level of local data packets is very high, which in turn leads to removal of real-time packets from other nodes. The improvised priority-based scheduling mechanism by Zaho Y. go over the waiting queue to decide on the smallest packet ID as the first/highest priority to execute.

D. Number of Queues

Packet scheduling schemes can even be classified based on the number of levels maintained in the ready queue at a sensor node. The following are such policies.

Single queue: Each sensor node maintains a single ready queue, all types of data are placed in the queue and are scheduled based on various available conditions: priority, type, size, arrival times, etc. This scheme suffers from starvation.

Multi-level Queue: Each sensor node maintains two or more levels in the ready queue. The data packets that arrive at the ready queue are systematized into various levels of the ready queue based on the type of the packet and their priorities. In multi-level queue mechanism, scheduling is generally divided into two phases: (i) distributing data packets among different levels of a ready queue and, (ii) finally scheduling the data packets at each level. The number of levels at a node varies depending on the level at which the node is located in the network. For case in point, node at upper levels has more levels in the ready queue while leaf nodes i.e. nodes at lower level divide ready queue into minimum number of levels. Following this, reduces end-to-end transmission delays and also optimize energy consumptions.

Lee et al. [32] proposed a multi-level queue scheduler that uses variable number of queues based on the location of the nodes in the network. Simple priority-based and multi-FIFO queue-based are the two kinds of scheduling used by the above scheduling policy. Karimi E. and Akbari B. [33] have also proposed a priority queue scheduling algorithm in which the queue is alienated into four queues to hold three different types of video and one regular data frame. Nidal N, Lutful K and Tarik T [24] proposed Dynamic Multilevel Priority Packet Scheduling Scheme which serves the requirements of the WSN in dynamic and efficient way and determines the scheduling criteria on fly. This scheme organizes the network into a hierarchical structure where in each node maintains three levels in the ready queue except that at the last level. Real-time data are put into the highest priority queue and they can pre-empt the data packets in other two levels. Non-real-time data are placed into other two levels. The last level nodes maintain two levels, one for real-time data and the other for non-real-time data since these nodes do not receive data from other nodes, thus reducing end-to-end delay.

III. Assumptions

This section discusses some general assumptions taken into account while working on our proposed Deadline Perception with Enhanced Dynamic Multilevel Priority (DP-EDMP) packet scheduling scheme.

The assumptions made during the design and execution phases of DPDMP are as follows:

- Firstly traffic includes of only real-time and non-real-time data.
- All data packets are of same size.
- The network is virtually organized into various levels of hierarchy.
- Nodes which measure same hop counts from the base station (BS) are assumed to be at same hierarchical level.
- The ready queue has a maximum of three levels. First level for real-time data (pr_1), second for non-real-time remote data (pr_2) and third for non-real-time local data (pr_3) and the priorities are in the order of $pr_1>pr_2>pr_3$.
- The intermediate nodes are programmed intelligently so as to perceive packets whose deadline has been expired as in contrast to normal sensor nodes.

IV. Proposed DP-EDMP Scheme

There exist mainly two problems with Dynamic Multilevel Priority Packet Scheduling Scheme [24], one being the effects of using Shortest Job First (SJF) i.e. difficulty in determining the processing time of next task and second regarding the optimal energy consumption. To solve the former issue, we propose to use Earliest Deadline First (EDF) scheduling policy to schedule the data packets in each level of the ready queue instead of SJF. It is proved to be efficient based on two parameters-average packet waiting time and end-to-end delay [23]; which are the major concerns in WSNs. The latter issue – optimal energy consumption, this can be achieved in numerous number of ways. The technique by which we try to optimize is, by removing the task with expired deadlines from the medium. This can be done by programming the sensor nodes to be intelligent enough to perceive the tasks whose deadlines have been expired. Generally in a WSN environment, intermediate sensor nodes possess capabilities only to sense, route and aggregate data and only the base station (BS) possess some extra capabilities for example to drop packets with expired deadlines. In such case even though the packet's deadline has been expired at an intermediate node, even then they are sent to the BS and there at the BS packet's header is checked for and dropped. Doing so network traffic increases due to wandering of dead packets in the medium, energy of the sensor nodes gets wasted due to the sensing of expired packets which is unnecessary and even data processing overhead gets increased. We can overcome all the above stated disadvantages by just inducing some extra capabilities into the intermediate sensor nodes i.e. programming them to be intelligent enough to sense the data packets with expired deadlines and drop them there itself without delivering such packets to the BS. So, we propose a

Deadline Perception with Enhanced Dynamic Multilevel Priority (DP-EDMP) packet scheduling policy.

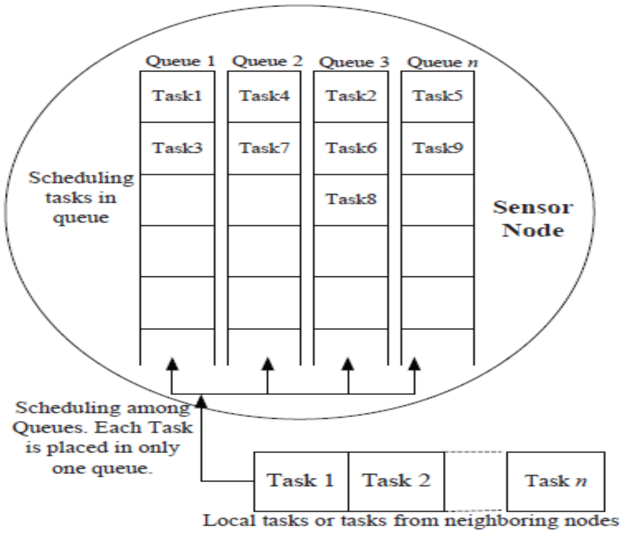

Fig. 2 Scheduling tasks among multiple queues

A. *Mode of Operation:*

Each sensor node maintains a ready queue and in multilevel scheduling scheme, the ready is divided into two or more levels. The number of levels depend on various factors taken into consideration such as position of the node in the network, type of policy implemented and etc. When data packets are sensed by a node and arrive at the ready queue they undergo two phases. First phase being allocation of data packets among multiple levels and second being, scheduling the packets in each level. These two phases are illustrated in Figure 2. For case in point, as in Figure 2 data packets say $Data_1$ to $Data_n$ undergo through the first phase i.e. they are allocated to particular level in the ready queue based on some criteria – here we take the criteria – priority of the data packet. Figure 2 demonstrates $Data_1$, $Data_5$ are placed in $Level_1$, $Data_2$ and $Data_6$ are placed in $Level_2$ and so on, which completes the first phase. Here is the second phase in which $Data_1$, $Data_5$ are scheduled to be delivered based on different conditions – here EDF. And also when data packet is sensed by a sensor nodes, it reads the packet header and checks whether the packet's deadline is expired, if so it is dropped there itself without transmitting it further.

We assume that the network is nearly prearranged into levels of hierarchy as in Dynamic Multilevel Priority (DMP) [24]. The network structure is shown in Figure 3. The distance from the BS is calculated based on level at which the node is located. Sensor nodes which are same number of hops from the BS come under same level. Each node maintain a ready queue and in this we divide the ready queue into three levels to accommodate different types of data in the network. Each level is fixed with certain priority. As discussed in assumptions, traffic comprises of real-time and non-real time data. The real-time traffic is given the highest priority and is placed in the highest prioritized level in the ready queue say $Level_1$. The other variant non-real-time remote data is given

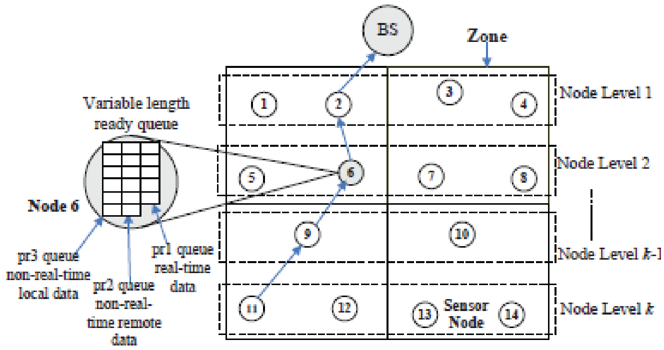

Fig. 3 Network Structure

the next highest priority and is placed in $Level_2$ of the ready queue and non-real-time local data is placed in least prioritized queue. For instance say real-time data is placed in the priority pr1 level in the ready queue, non-real-time remote data – pr_2 queue and non-real-time local data – pr_3 queue, their priority levels are arranged as follows : $pr_1 > pr_2 > pr_3$.

The mode of operation of our proposed scheduling scheme goes as: when a sensor node sense data, they are scheduled among the levels of ready queue based on the priority of the data packet. Within each level the data packets are scheduled for transmission based on Earliest Deadline First (EDF) so that packets with lower value of deadline might reach the BS before their expiry, which increases the packet delivery ratio. Each time the packet is sensed its header is checked for deadline value and if it is expired it is dropped at that intermediate node itself. This lowers the data processing overhead at the successive sensor nodes and also reduces the network traffic because the expired data packets do not wander in the medium.

The real-time data are processed first and then data at other levels are processed later. The running task is pre-empted if the currently sensed packet has a lower value of deadline when compared to the current running task. The task pre-empted can be a higher prioritized task when compared to the currently sensed one. This is a situation where non-real-time data pre-empts real-time data. At time say T1 if two packets of equal priority with same deadline arrive at the ready queue, the packet that has come from lower levels of hierarchy in the network structure is set to higher priority. When data packets that arrive from same level, the one with least deadline value will be assigned higher priority. Assigning higher priority here means, that task is processed first.

B. *Algorithm:*

A step by step procedure that is followed in our proposed scheme is presented in this section.

---

while $packet_{l,n}$ is sensed/ received by $sensor\_node_n$ at $level_l$ **do**
  if traffic_Type($packet_{l,n}$) = real-time **then**
    allocate $packet_{l,n}$ to pr_1 queue
  else if $sensor\_node_n$ is not a leaf node in the hierarchy **then**
    if $packet_{l,n}$ is remote data **then**
      allocate $packet_{l,n}$ to pr_2 queue
    else
      allocate $packet_{l,n}$ to pr_3 queue


```
        end if
    else
        allocate packet_{l,n} to pr_2 queue
    end if
All packets are processed using Earliest Deadline First (EDF)
    if deadline of packet_{l,n} is expired then
        drop packet_{l,n} at sensor_node_n
    end if
end while
while pr_i's are not empty do
    if deadline (current processing packet > sensed packet) then
        preempt current processing packet
        execute the sensed packet
    end if
    if time_of_arrival (packet_{i,j} = packet_{m,n}) && deadline(packet
        _{i,j} = packet_{m,n}) then
        if i<m then
            process packet_{i,j} //packet that has originated from lower level
            in the hierarchy
        else
            process packet_{m,n}
        end if
    end if
    if exits(non-real-time) then
        if deadline(packet_{i,j}) at pr_2 || deadline(packet_{m,n}) at pr_3 <
        deadline(packet_{p,q}) at pr_1 then
            preempt packet_{p,q}
                if deadline(packet_{i,j} < packet_{m,n}) then
                    process packet_{i,j}
                else
                    process packet_{m,n}
                end if
        end if
    end if
end while
```

V. Performance Evaluation

This section presents the experimental results obtained by simulating the WSN environment using NS2. The log files are analysed using awk programming, the values obtained after analysis are given as an input to X-graph to generate graphs. The simulation model is used to evaluate the performance of Deadline Perception with Enhanced Dynamic Multilevel Priority (DP-EDMP) packet scheduling scheme - which is our proposed scheme, Enhanced Dynamic Multilevel Priority (EDMP) packet scheduling scheme – which uses Earliest Deadline First (EDF) for internal scheduling within the levels of ready queue, Dynamic Multilevel Priority (DMP) packet scheduling scheme and First Come First Serve (FCFS) policy. The comparison is made pertaining to end-to-end delay, average waiting time and energy consumption.

Figure 4 and 5 illustrates the end-to-end delay of real-time tasks and non-real-time tasks respectively. The Y-axis represents the end-to-end delay in micro seconds and X-axis denotes the time at which the different scheduling policies are executed. We observe from both graphs that our proposed DP-EDMP scheduling scheme outperforms the stated existing policies. This so because the data packets with expired deadlines are removed from the medium at intermediate nodes itself, which reduces network traffic and hence forth reduces end-to-end delay of all packets. Moreover, the end-to-end delay is much lower for non-real-time data packets, as our scheme pre-empts real-time packets if deadlines of the former packets are very low.

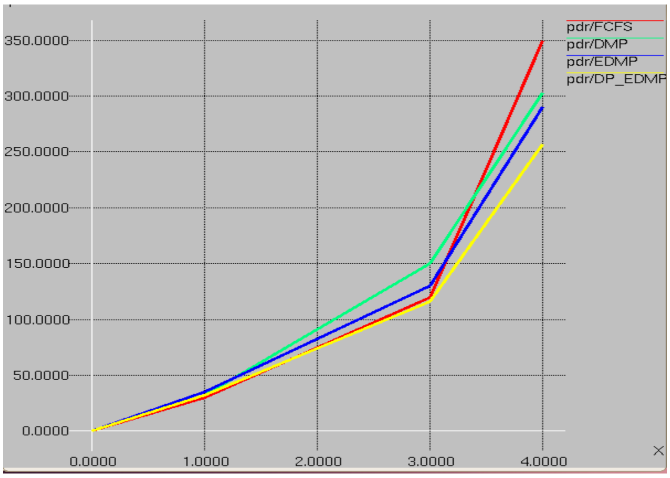

Fig. 4 End-to-end delay of real-time packets

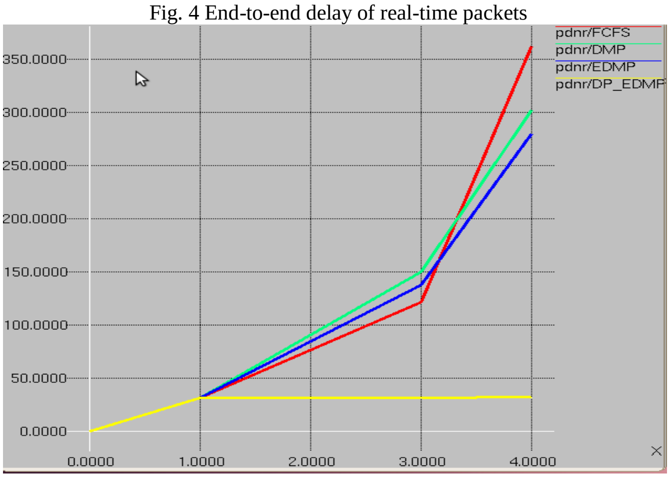

Fig. 5 End-to-end delay of non-real-time packtes

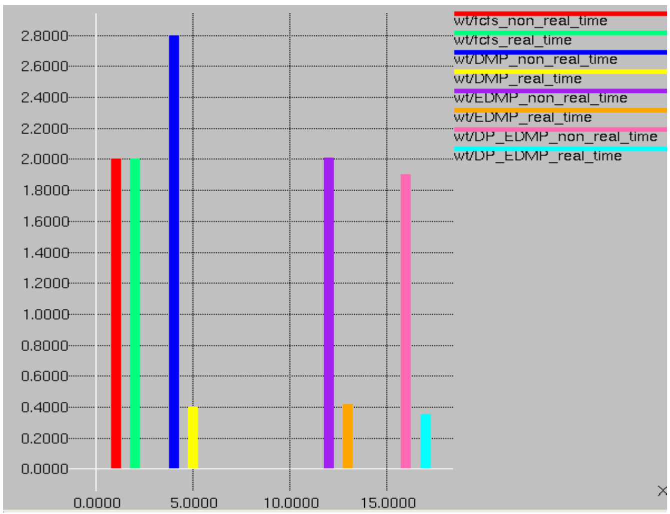

Fig. 6 Average waiting times of real-time and non-real-time packets

Figure 6 demonstrates the average waiting time of real-time and non-real-time tasks. It proves that DP-EDMP has better performance pertaining to average waiting time when compared with stated policies. The probable reasons which are explained related to differences in performance for end-to-end delay holds well even in this case.

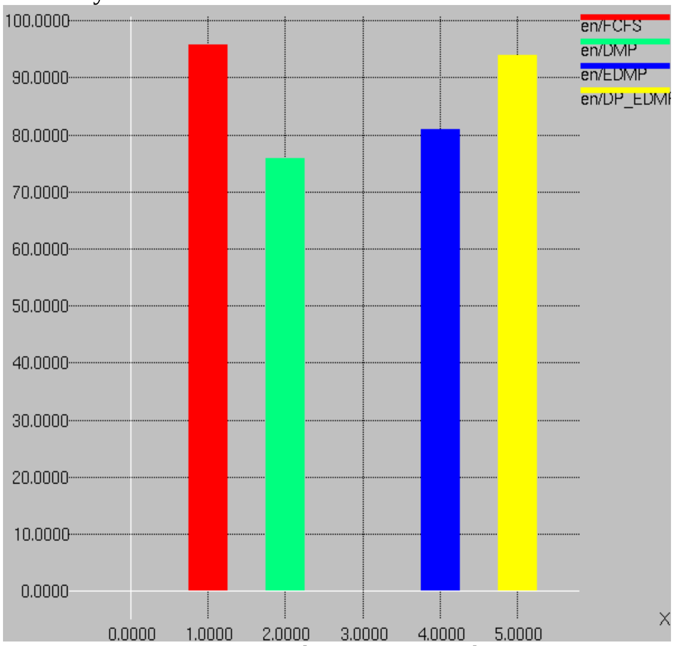

Fig. 7 Energy lasting at sensor nodes

Figure 7 illustrates the average energy lastinF(g at the sensor nodes after all the transmissions are completed. The Y-axis represents the energy values in Joules. Even this performance criteria proves that DP-EDMP is better when compared to DMP. The possible reasons for this is our scheme removes packets with expired deadlines from the medium, without transmitting them to the base station (BS). Due to which the intermediate sensor nodes need not unnecessarily sense the dead packets, by which energy of the sensor can be used optimally. However DMP and DP-EDMP are less energy efficient when compared to FCFS as they require a fewer processing cycles for categorization of data packets among the levels of the ready queue. Even then proposed DP-EDMP is regarded to be highly efficient as it ensures low end-to-end transmission delays, minimum waiting times and optimal energy consumptions.

VI. Conclusion And Future Work

In this paper, we propose Deadline Perception with Enhanced Dynamic Multilevel Priority (DP-EDMP) packet scheduling scheme for Wireless Sensor Networks (WSNs). This uses Earliest Deadline First (EDF) algorithm to schedule tasks within the levels of the ready queue and type of data packet is used to schedule packets among the levels of the ready queue. The nodes are designed to be intelligent enough to perceive data packets with expired deadlines and then they are removed from the medium. The proposed scheme ensures minimum end-to-end transmission delays in case of non-real-time packets as well. It also guarantees low waiting times and even conserves energy of the sensor nodes when compared to DMP.

Deadline perception and removal of expired packets reduce network traffic and data processing overhead, and consumes less power when compared to DMP. But it consumes moderately more energy when compared to FCFS and other multi-level queue scheduling algorithms. As an enhancement we propose to incorporate efficient sleep scheduling algorithm with the existing DP-EDMP scheme to conserve sensor's energy to a better extent.

References

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

[2] G. Bergmann, M. Molnar, L. Gonczy, and B. Cousin, "Optimal period length for the CQS sensor network scheduling algorithm," in *Proc. 2010 International Conf. Netw. Services*, pp. 192–199.

[3] E. Bulut and I. Korpeoglu, "DSSP: a dynamic sleep scheduling protocol for prolonging the lifetime of wireless sensor networks," in *Proc. 2007 International Conf. Advanced Inf. Networking Appl.*, vol. 2, pp. 725–730.

[4] S. Chachra and M. Marefat, "Distributed algorithms for sleep scheduling in wireless sensor networks," in *Proc. 2006 IEEE International Conf. Robot. Autom.*, pp. 3101–3107.

[5] 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.

[6] F. Liu, C. Tsui, and Y. J. Zhang, "Joint routing and sleep scheduling for lifetime maximization of wireless sensor networks," *IEEE Trans.Wireless Commun.*, vol. 9, no. 7, pp. 2258–2267, July 2010.

[7] J. Liu, N. Gu, and S. He, "An energy-aware coverage based nodescheduling scheme for wireless sensor networks," in *Proc. 2008 InternationalConf. Young Comput. Scientists*, pp. 462–468.

[8] 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.

[9] 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.

[10] D. Shuman and M. Liu, "Optimal sleep scheduling for a wireless sensor network node," in *Proc. 2006 Asilomar Conf. Signals, Syst. Comput.*, pp. 1337–1341.

[11] S. Paul, S. Nandi, and I. Singh, "A dynamic balanced-energy sleep scheduling scheme in heterogeneous wireless sensor network," in *Proc.2008 IEEE International Conf. Netw.*, pp. 1–6, 2008.

[12] A. R. Swain, R. C. Hansdah, and V. K. Chouhan, "An energy aware routing protocol with sleep scheduling for wireless sensor networks," in *Proc. 2010 IEEE International Conf. Adv. Inf. Netw. Appl.*, pp. 933–940.

[13] 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 Comput.*, pp. 19–24.

[14] Y. Wang, D. Wang, W. Fu, and D. P. Agrawal, "Hops-based sleep scheduling algorithm for enhancing lifetime of wireless sensor networks," in *Proc. 2006 IEEE International Conf. Mobile Adhoc Sensor Syst.*, pp. 709–714.

[15] Y. Xiao, H. Chen, K. Wu, B. Sun, Y. Zhang, X. Sun, and C. Liu, "Coverage and detection of a randomized scheduling algorithm in wireless sensor networks," *IEEE Trans. Comput.*, vol. 59, no. 4, pp. 507–521, Apr. 2010.

[16] X. Xu, Y. H. Hu, J. Bi, and W. Liu, "Adaptive nodes scheduling approach for clustered sensor networks," in *Proc. 2009 IEEE Symp. Comput. Commun.*, pp. 34–39.

[17] Y. Zhao, J. Wu, F. Li, and S. Lu, "VBS: maximum lifetime sleep scheduling for wireless sensor networks using virtual backbones," in *Proc. 2010 IEEE INFOCOM*, pp. 1–5.

[18] B. Zeng, Y. Dong, and D. Lu, "Cooperation-based scheduling algorithm

in wireless multimedia sensor networks," in *Proc. 2011 International Conf. Wireless Commun., Netw. Mobile Comput.*, pp. 1–4.

[19] N. Edalat, W. Xiao, C. Tham, E. Keikha, and L. Ong, "A price-based adaptive task allocation for wireless sensor network," in *Proc. 2009 IEEE International Conf. Mobile Adhoc Sensor Syst.*, pp. 888–893.

[20] H. Momeni, M. Sharifi, and S. Sedighian, "A new approach to task allocation in wireless sensor actor networks," in *Proc. 2009 International Conf. Computational Intelligence, Commun. Syst. Netw.*, pp. 73–78.

[21] F. Tirkawi and S. Fischer, "Adaptive tasks balancing in wireless sensor networks," in *Proc. 2008 International Conf. Inf. Commun. Technol.:From Theory Appl.*, pp. 1–6.

[22] X. Yu, X. Xiaosong, and W. Wenyong, "Priority-based low-power task scheduling for wireless sensor network," in *Proc. 2009 International Symp. Autonomous Decentralized Syst.*, pp. 1–5.

[23] W. Stallings, *Operating Systems*, 2nd edition. Prentice Hall, 1995.

[24] N. Nasser, L. Karim, T. Taleb, "Dynamic Multilevel Packet Scheduling Scheme for Wireless Sensor Network," *IEEE Trans. Wireless Communications,* vol. 12, no. 4, pp. 1448–1459, 2013.

[25] C. Lu, B. M. Blum, T. F. Abdelzaher, J. A. Stankovic, and T. He, "RAP: a real-time communication architecture for large-scale wireless sensor networks," in *Proc. 2002 IEEE Real-Time Embedded Technol. Appl.Symp.*, pp. 55–66.

[26] K. Mizanian, R. Hajisheykhi, M. Baharloo, and A. H. Jahangir, "RACE: a real-time scheduling policy and communication architecture for largescale wireless sensor networks," in *Proc. 2009 Commun. Netw. Services Research Conf.*, pp. 458–460.

[27] M. Yu, S. J. Xiahou, and X. Y. Li, "A survey of studying on task scheduling mechanism for TinyOS," in *Proc. 2008 International Conf. Wireless Commun., Netw. Mobile Comput.*, pp. 1–4.

[28] TinyOS. Available: http://webs.cs.berkeley.edu/tos, accessed June 2010.

[29] P. A. Levis, "TinyOS: an open operating system for wireless sensor networks (invited seminar)," in *Proc. 2006 International Conf. Mobile Data Manag.*, p. 63.

[30] K. Lin, H. Zhao, Z. Y. Yin, and Y. G. Bi, "An adaptive double ring scheduling strategy based on tinyos," *J. Northeastern University NaturalSci.*, vol. 28, no. 7, pp. 985–988, 2007.

[31] Y. Zhao, Q. Wang, W. Wang, D. Jiang, and Y. Liu, "Research on the priority-based soft real-time task scheduling in TinyOS," in *Proc. 2009International Conf. Inf. Technol. Comput. Sci.*, vol. 1, pp. 562–565.

[32] E. M. Lee, A. Kashif, D. H. Lee, I. T. Kim, and M. S. Park, "Location based multi-queue scheduler in wireless sensor network," in *Proc. 2010 International Conf. Advanced Commun. Technol.*, vol. 1, pp. 551–555.

[33] E. Karimi and B. Akbari, "Improving video delivery over wireless multimedia sensor networks based on queue priority scheduling," in *Proc. 2011 International Conf. Wireless Commun., Netw. Mobile Comput.*, pp. 1–4.

[34] A. Silberschatz, P.B. Galvin, G. Gagne, *Operating Systems Concepts*, 7th edition. John Wiley& Sons, 2005.