International Journal of Advanced Research in
Computer Science & Technology (IJARCST 2014)
Vol. 2 Issue Special 1 Jan-March 2014
ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

Optimized Energy-Delay Scheduling During Critical Event Monitoring in Wireless Sensor Networks

[I]Ravisankar. A, [II]Swetha.G
[I]PG Scholar, Dept. of CSE, Adhiyamaan College of Engg., HOSUR, India
[II]Assistant Professor, Dept. of CSE, Adhiyamaan College of Engg., HOSUR, India

Abstract

This paper proposed to monitor a critical event in wireless sensor networks. Whenever a critical event occurs, the critical event is detected by the nearby sensor nodes. Immediately these sensor nodes should broadcast an alarm message to the entire network. To increase lifetime of the network, sleeping methods are always implemented in WSNs, it increases the delay of alarm message broadcasting. In this paper, we propose a novel sleep scheduling method to reduce the delay of alarm broadcasting from any sensor node in WSNs. scheduling methods are always employed in WSNs, we tend to style two determined traffic ways for the transmission of alarm message ,and level-by-level offset based wake-up pattern according to the paths, respectively. Once a crucial event happens, An alarm is quickly transmitted on one among the traffic ways to a middle node, and so it's instantly broadcast by the middle node on another path without collision.

Keywords

Wireless Sensor Network (WSN),critical event monitoring, sleep scheduling, broadcasting delay, multichannels

I. Introduction

We design a novel sleep scheduling method based on the level-by-level offset schedule to achieve low broadcasting delay in wireless sensor networks (WSNs). Two phases are set for the alarm broadcasting. Firstly, when a node detects a critical event, it originates an alarm message and quickly transmits it to a center node along a predetermined path with a level-by-level offset way. Then, the center node broadcasts the alarm message to the other nodes along another path also with a level-by-level offset way. Through designing a special wake-up pattern, the two possible traffics could be both carried by a node. To eliminate the collision in broadcasting, a colored connected dominant set (CCDS) in the WSN via the IMC algorithm.

However, it is still a challenge for us to apply the level-by level offset to alarm broadcasting in the critical event monitoring. First, the order of nodes' wake-up should conform to the traffic direction. If the traffic flow is in the reverse direction(as show in Fig.2), the delay in each hop will be as large as the length of the whole duty cycle. Second, the level-by-level offset employed by the packet broadcasting could cause a serious collision. Finally, the transmission failure due to some unreliable wireless links may cause the retransmission during the next duty cycle, which also results in large delay equaling the whole duty cycle.

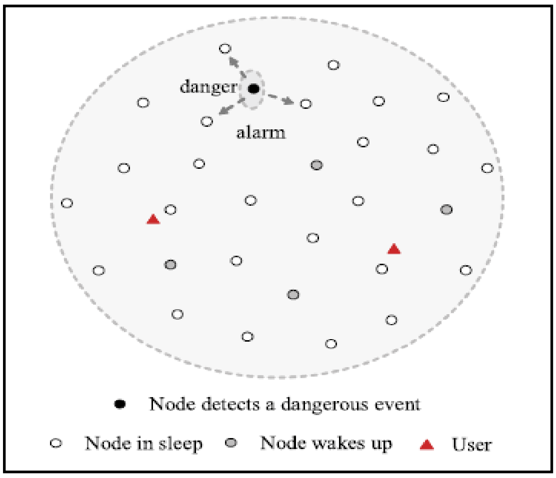


Fig. 1: Critical Event Monitoring With a WSN

First, when a node detects a critical event, it originates an alarm message and quickly transmits it to a center node along a predetermined path with a level-by-level offset way. Then, the center node broadcasts the alarm message to the other nodes along another path also with a level-by-level offset way. Through designing a special wake-up pattern, the two possible traffics could be both carried by a node, and the node just needs to be awake for no more than T time in each duty cycle, where T is the minimum time needed by a node to transmit an alarm packet. To eliminate the collision in broadcasting, a colored connected dominant set (CCDS) in the WSN via the IMC algorithm is established. Each node transmits or receives packets in a specific channel according to the color assigned.

In summarization, characteristics of the proposed sleep scheduling scheme are

1. The boundary of the broadcasting delay is 3D+2L, wherever D is that the most hop of nodes to the middle node, and L is that the length of duty cycle, the unit is that the size of your time slot. because the delay is merely a linear combination of hops and duty cycle, it may be terribly tiny even in giant scale WSNs.
2. The broadcasting delay is freelance of the length of the duty cycle, however it will increase linearly with the quantity of the hops.
3. The broadcasting delay is freelance of the density of nodes.
4. The energy consumption is extremely low as nodes rouse for under one squeeze the duty cycle throughout the observance.

II. Problem Description

We assume that a certain node, called as center node, in the network has obtained the network topology in the initialization (e.g., sink n node).The center node computes the sleep scheduling according to the proposed scheduling scheme and broadcasts the scheduling to all the other nodes.

We define $f(n_i)$ as the slot assignment function. If $f(n_i)=s, s\{0,\ldots,L-1\}$, it means that node n_i wakes up only at slot s to receive packets. Meanwhile, we define $F(n_i)$ as the channel assignment function which assigns a frequency channel to node n_i.

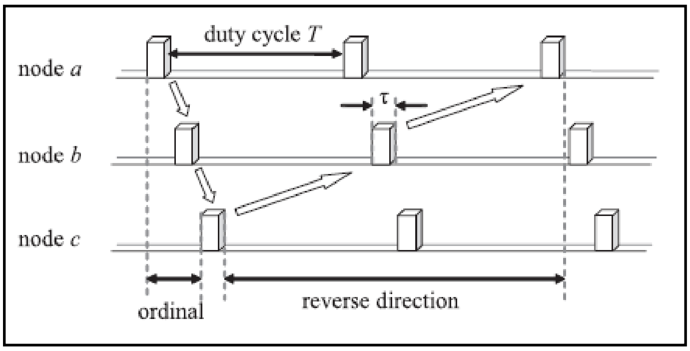
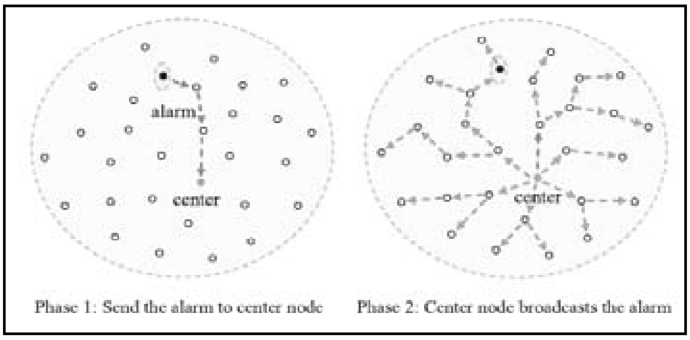


Fig. 2: The level-by-level offset schedule

III. The Proposed Scheduling Method

A. Basic Idea

The proposed scheduling method includes two phases: 1) any node which detects a critical event sends an alarm packet to the center node along a predetermined path according to level-by-level offset schedule; 2) the center node broadcasts the alarm packet to the entire network also according to level-by-level offset way. Fig. 3 illustrates these two phases of the processing. We define the traffic paths from nodes to the center node as uplink and define the traffic path from the center node to other nodes as downlink, respectively. Each node needs to wake up properly for both of the two traffics. Therefore, the proposed scheduling scheme should contain two parts: 1) establish the two traffic paths in the WSN; 2) calculate the wake-up parameters (e.g., time slot and channel) for all nodes to handle all possible traffics.



Phase 1: Send the alarm to center node
Phase 2: Center node broadcasts the alarm

Fig. 3: Two phases of the Alarm Broadcasting

To minimize the broadcast delay, we establish a breadth first search (BFS) tree for the uplink traffic and a colored connected dominant set for the downlink traffic, respectively.

B. Traffic Paths

First of all, we choose a sensor node as the center node c. Then, we construct the bfs tree which divides all nodes into layers H_1, H_2, H_3, ... , H_D, where H_i is the node set with minimum hop i to c in the WSN. With the BFS tree, the uplink paths for nodes can be easily obtained.

To establish the second traffic path, we establish the CCDS in G with three steps: 1) construct a maximum independent set (MIS) in G; 2) select connector nodes to form a connected dominated set (CDS), and partition connector nodes and independent nodes in each layer into four disjoint sets with IMC algorithm proposed in [12]; 3) color the CDS to be CCDS with no more than 12 channels. The details are described as follows, and the variables therein are defined in Table 1.First, we construct a MIS I. As all nodes have been divided into H_1 , H_2 , H_3 , ... , H_D with the BFS tree, the MIS can be established layer by layer (i.e., hop by hop) in the BFS as follows: Start from the 0th hop, we pick up a maximum independent set, then, move on to the first hop, pickup another maximum independent set. Note that, independent nodes of the first hop also need to be independent of those in the previous hop. Repeat this process until all hops of nodes have been worked on.

Second, we construct the CDS by selecting connector nodes C from V/I to interconnect independent nodes as follows: Obviously, for any two 2-hop neighboring independent nodes ,at least one node in G is adjacent to both of them. Hence, the node is possible to be selected as a connector nodes .We use the idea of the IMC algorithm to select the connector nodes, which partitions independent nodes $I \cap H_i$ in each layer into four disjoint subsets $U_{i,j}$ ($0 \le j \ge 3$), and selects four disjoint subsets $W_{i-1,j}$ ($0 \le j \ge 3$) among ($H_{i-1} \cap H_{i-2}$) ∩ I as connector nodes to cover I ∩ H_i. When nodes in $W_{i-1,j}$ broadcast simultaneously, they will not cause any collision among nodes in $U_{i,j}$. By this way, the CDS is established. We further color the CDS to be CCDS as follows: We divide all nodes in CDS into several sets according to their minimum hops to c in CDS. As CDS is based on $G^2(I)$, the number of hops from independent nodes to c in the CDS is even, and the number of hops from connector nodes to c in the CDS is odd. Therefore, we obtain I_0, I_2, I_4, ... and C_1,C_3, C_5, ... In addition, dominated node B could be divided into B0, B2, B4, ...They are dominated by I_0, I_2, I_4,..., respectively. Since any two independent nodes cannot be adjacent, the distribution of independent nodes actually sparse. It has been proved that each independent node has less than 12 neighbors in I within 2-hop distance. Therefore, G| could be colored with ch_1, ... , ch_{12}. Hence, when independent nodes in each layer broadcast simultaneously, they will not cause any collision at connector nodes. We define sending channel as $ch_s(n_k)$ and receiving channel as $ch_r(n_k)$ for each node n_k, corresponding to channels in which nk sends packets and receives packets, respectively. Each node nk in I_i gets its chs (n_k) according to its color, and each node nt in C_i obtains its chr(nt) according to the color of one of its parents in I_{i-1}. In addition, we color the subsets $U_{i,j}$ and $W_{i-1,j}$ with cl_j($0 \le j \ge 3$)in each layer. Hence, when connector nodes in each layer (i.e., $W_{i-1,j}$, $0 \le j \ge 3$) broadcast simultaneously, they will not cause any collision at independent nodes in the next layer (i.e., $U_{i,j}$, $0 \le j \ge 3$). Each node nk in I_i gets its $ch_r(n_k)$ according to the color of Ui,j that it belongs to, and each node nt in Ci obtains its $ch_s(n_t)$ according to the color of $W_{i,j}$ that it belongs to. While, each node n_s in B_i obtains it's chr (n_s) according to the sending channel of an independent node in Ii which dominates n_s.

Table 1: Definitions of Some Variables

c	The center node in the networks
H_i	The nodes with minimal hop i to c in G
H'_i	The nodes with minimal hop i to c in CDS
I_i	The *independent nodes* with minimal hop i to c in CDS
C_i	The *connector nodes* with minimal hop i to c in CDS
B_i	The *dominated nodes* dominated by I_i

C. Wake-Up Patterns

After all nodes get the traffic paths, sending channels and receiving channels with the BFS and CCDS, the proposed wake-up pattern is needed for sensor nodes to wake-up and receive alarm packet to achieve the minimum delay for both of the two traffic paths.

As described above, there are two traffic paths for the alarm dissemination, and sensor nodes take two level-by level offset schedules for the traffic paths. Fig. 4 shows the two level-by level offset schedules, 1) sensor nodes on paths in the BFS wake up level-by-level according to their hop distances to the center node; 2) after the center node wakes up, the nodes in the CCDS will go on to wake up level-by-level according to their hop distances in the CCDS. Hence, when an alarm packet is originated, it could be quickly forwarded to the center node along a path in the BFS, then, the center node immediately broadcasts it along the paths in the CCDS. Since it is hard to predict when the alarm occurs, the two level-by-level offset schedule are taken periodically as shown in Fig. 4. Moreover, it is needed to effectively arrange time slots for sensor nodes at different positions in the topology, so that the two level-by-level offset schedule can periodically work without interfering with each other. The assignment of time slots is summarized in Table 2, which can be briefly described as follows: 1) all nodes in H obtain slots for uplink traffic according to their hops in H and the sequence number of duty cycles; 2) nodes in H obtain slots for downlink traffic according to their hops in H and the sequence number of duty cycle; 3) nodes in B_i obtain the same slot as C_{i+1} for downlink traffic. For example, a sensor node nj in H1 obtains slot L - 1 in odd duty cycles for uplink traffic. On the other hand, n_j may also be in H_1, and it obtains slot 2 in even duty cycles for downlink traffic. In addition, it is obvious that, whenever a sensor node detects a critical event, it waits for no more than two duty cycles before its time slot for uplink traffic comes.

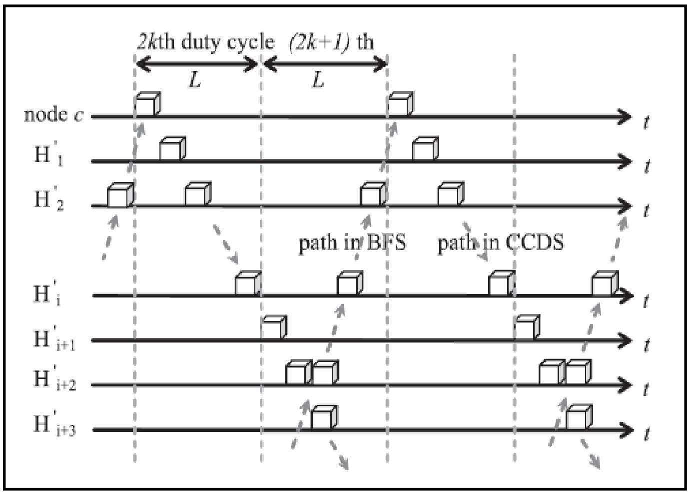


Fig. 4: Two Periodic Level-by-Level Offset Schedules

Furthermore, for nodes which are both in H2mL+s and H2nL+t, when s+t=L, nodes will be assigned the same slot for uplink traffic and downlink traffic, i.e., nodes need to wake up for only one time slot every two duty cycles and it can receive the possible alarm transmitted both in uplink and downlink. Therefore, their receiving channels need to be modified. Suppose n_j is a node with the same slot for uplink traffic and downlink traffic. It should wake up in its chw channel, and its child in the BFS also should send the possible alarm to nj in nj's chw channel instead of ch1.

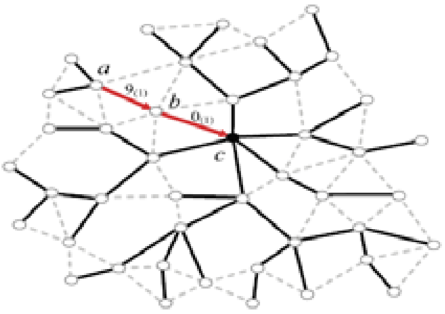
Table 2: Wake-Up Patterns

Node	*Time Slot for wake-up*	
$(0 \leq s \leq L-1)$	in 2kth duty cycle	in $(2k+1)$th duty cycle
$n_j \in H_{(2m+1)L+s}$	$f(n_j) = L - s$	
$n_j \in H_{2mL+s}$		$f(n_j) = L - s$
$n_j \in H'_{2mL+s}$	$f(n_j) = s$	
$n_j \in H'_{(2m+1)L+s}$		$f(n_j) = s$
$n_j \in B_i$	$f(n_j) = f(n_t)$, *where* n_t *is any node in* C_{i+1}	

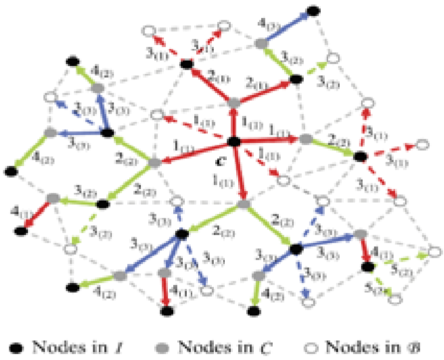
D. An Example

In order to show the assignment more clearly, we give an example shown in Fig. 5, where the numbers in brackets denote the frequency channels, and the numbers in front of brackets denote the time slots in a duty cycle. The length of duty cycle is set 10. Consider two nodes a and b (shown in Fig. 5a), which are in H2 and H1, respectively, in the BFS. Suppose node detects a critical event. It will originate an alarm packet and sends it to node b at time slot 9 in the earliest odd duty cycle in channel ch1. When node b wakes up at time slot 9 in channel ch1 and receives the alarm, it sends the alarm to the center node c which wakes up at time slot 0 in each even duty cycle in channel ch1. After receiving the alarm, node c begins to broadcast the alarm packet among the CCDS, as shown in Fig. 5b. The solid lines are the paths in the CCDS. In the broadcasting phase (i.e., in even duty cycle for nodes a and b), node a and node b are in H3 and H1 , respectively, in the CCDS. Therefore, they wake up at time slots 3 and 1, respectively, in each even duty cycle in their receiving channels (channel 3 and channel 1, respectively).

When receiving the alarm packet, node a broadcasts it in its sending channel (channel 2), while node b does not broadcast the packet as it is a dominated node. From Fig. 5b, all the transmissions at the same time slot do not cause any collision, and the broadcast is executed level-by-level without waiting. Furthermore, since the alarm can be quickly relayed to center node in an uplink path and center node could immediately begin to broadcast it, the



(a) Uplink traffic in BFS



(b) Downlink traffic in CCDS

Fig. 5. An example of the alarm broadcast with the proposed scheduling method.

broadcasting delay is much lower. In addition, the energy consumption of nodes is also very low, since most nodes stay awake for only one time slot in each duty cycle. Moreover, the center node and nodes with the same wakeup slots for uplink traffic and downlink traffic stay awake for one time slot every two duty cycles. Obviously, Ii, Ci, and Bi are used only for downlink traffic to solve the collision.

IV. Analysis and Simulation

A. Performance Analysis

Lemma 1: The maximum hop of the shortest path in the CCDS from any node to the center node is no more than 2D. Proof: Consider any independent node nj, there must be a parent in C connecting another independent node which is closer to the center node than nj. If the parent is in the same layer with nj in the BFS, then, it increases the hops of nj to c in the CCDS. Otherwise, the number of hops does not increase. Consider the worst case for each hop with one increment on the shortest path from a node in layer HD to c, the maximum length of the shortest path in the CCDS is consequently 2D. tu Lemma 2. The upper bound of alarm broadcasting delay in WSN is no more than 3D + 2L.

B. Simulations in Unreliable Environment

We use ns-2 simulator to evaluate the performances of the proposed scheduling method in unsteady WSNs. In Fig. 6, 225 sensor nodes are randomly deployed in an area of 150*150m2.The successful communication probability p to characterize the wireless link between any two nodes is employed. Considering the interference caused by non neighboring nodes, we define the worse link quality than that in practice with assumption p = 1-(d/20)2, where d is the distance between two nodes and d < 20. The links with p ≥50 % are chosen to form the topology of network for the proposed scheme, as shown in Fig. 6. The dashed lines are the links with p < 50%. The duty cycle is 1 s.

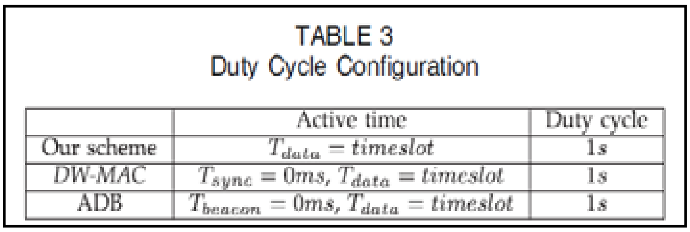


TABLE 3
Duty Cycle Configuration

	Active time	Duty cycle
Our scheme	$T_{data} = timeslot$	$1s$
DW-MAC	$T_{sync} = 0ms, T_{data} = timeslot$	$1s$
ADB	$T_{beacon} = 0ms, T_{data} = timeslot$	$1s$

1. Different Sizes of Time Slot

We first set the size of the time slot to be the minimum time for sensor nodes to transmit an alarm packet, e.g., 2 ms .When an alarm transmission fails between two adjacent nodes with the proposed scheme, the sender node has to retransmit the alarm after 2 duty cycles. While, for the ADB and the improved DW-MAC schemes, the sender node retransmits the alarm after 1 duty cycle. Obviously, the proposed scheme does not exhibit good performance in the case of minimum time slot. To improve it, we set the size of the time slot to be 10 ms. Hence, the transmission delay could be largely reduced. It can be seen, the broadcasting delay with the proposed scheme becomes much lower when the size of time slot is 10 ms . which affects the results in the experiments. For example, in experiments 1 and 8, packets usually cannot be successfully transmitted within a time slot, and have to be retransmitted after 2 s in next duty cycle. Therefore, the delay becomes large. Compared with the proposed scheme, the delay with the ADB and the improved DW-MAC schemes is even larger in most experiments. As sensor nodes in ADB wake up asynchronously, the average transmission delay in each hop is at least about half a duty cycle even if all transmissions were successful. While, for the improved DW-MAC, because the SCH (i.e., alarm) is forwarded within synchronous time slots, the number of hop counts of SCH transmission in each duty cycle is restricted by the size of time slot Tdata. In ideal case, the number is Tdata/t. However, due to unsteady links, the number is dynamic. Hence, the number of duty cycles needed for the broadcasting in the network is random, resulting in highly dynamic results in the experiments. We further enlarge the time slot to be 20 ms for the three schemes. It can be seen, the proposed scheme achieves a distinct predominance to the other two schemes. Moreover, the broadcasting delay with the proposed scheme and the ADB scheme becomes much steadier in 10 experiments, as almost each packet can be successfully transmitted within 20 ms. While, The delay with the improved DW-MAC is still dynamic, because the number of hop counts of alarm transmission in each duty cycle is still uncertain due to unsteady links. It is unnecessary to further enlarge the size of time slot, because the performance of the proposed scheme could not be further promoted. On the other hand, further enlargement of time slot increases energy consumption of sensor nodes, especially for the improved DW-MAC as nodes have to keep awake during the whole of the synchronous time slot. We conduct more experiments with the schemes in several networks. All the networks are generated randomly with 225 sensor nodes. In each network, we made 20 experiments and the average broadcasting delay with the standard deviation is shown in Tables 4 and 5. For example, the average broadcasting delay in network 1 with the proposed scheme is 5.4 s and the standard deviation of the delay is 4.8 s, which is denoted as 5:4/4:8 in Table 4. From Tables 4 and 5, the average broadcasting delay of the proposed scheme is always much lower than that of the other two methods.

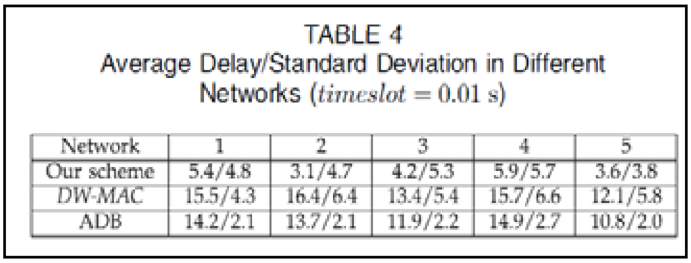


TABLE 4
Average Delay/Standard Deviation in Different
Networks ($timeslot = 0.01$ s)

Network	1	2	3	4	5
Our scheme	5.4/4.8	3.1/4.7	4.2/5.3	5.9/5.7	3.6/3.8
DW-MAC	15.5/4.3	16.4/6.4	13.4/5.4	15.7/6.6	12.1/5.8
ADB	14.2/2.1	13.7/2.1	11.9/2.2	14.9/2.7	10.8/2.0

2. Multiple Alarms

In some cases, the critical event may trigger several alarms in the network, and they may be sent to a parent node when it wakes up. To deal with the collision, we design a mechanism for the proposed scheduling as follows: Suppose the time slot is denoted as k*t. When a sensor node having detected the event is going to send an alarm packet, it keeps transmitting the packet randomly with the probability 1=2 during the time slot. However, if the node detects some others are transmitting alarm packets during the same time slot, it gives up its transmission. Through this way, the nodes sending alarms could be decreased gradually. Note that, the parent node just needs to successfully receive one alarm. The parent node cannot judge whether there is an alarm packet by just detecting the

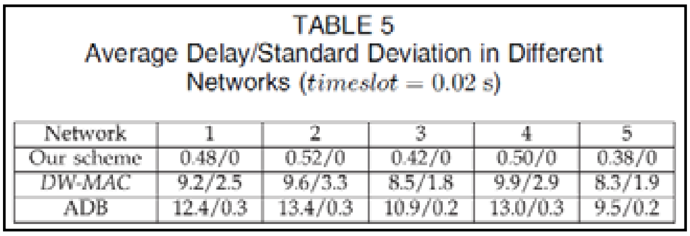


TABLE 5
Average Delay/Standard Deviation in Different Networks ($timeslot = 0.02$ s)

Network	1	2	3	4	5
Our scheme	0.48/0	0.52/0	0.42/0	0.50/0	0.38/0
DW-MAC	9.2/2.5	9.6/3.3	8.5/1.8	9.9/2.9	8.3/1.9
ADB	12.4/0.3	13.4/0.3	10.9/0.2	13.0/0.3	9.5/0.2

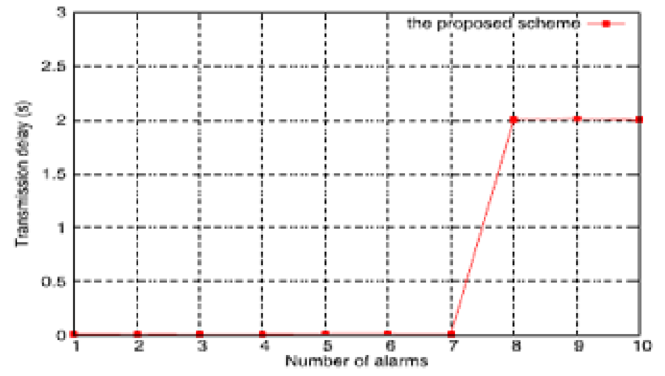


Fig. 6: Transmission Delay for Multiple Alarms

channel, because some configuration packets also need to be transmitted in the network and the alarm packet needs to be exactly received to avoid misinformation. We evaluate the performance of the mechanism with a simple and typical network model. Suppose there are M ($1 \leq M \geq 10$) nodes that need to send packets to a parent node which keeps awake for 20 ms every two duty cycles periodically. The quality of the link between the parent node and each child is 70 percent. Suppose the range of the event region is smaller than that of nodes' radio detection. Fig. 6 shows the time when the parent node successfully receives a packet. For each value of M, we conduct 20 experiments and give the maximum time. It is obviously from Fig.6 that, when M <8, the M children nodes can successfully send one packet to their parent within the 20 ms. When M=9 or 10, it needs two duty cycles to send the packet, resulting in 2s extra delay. However, the total broadcasting delay is still much lower than that of the improved DW-MAC and ADB schemes.

V. Conclusion

in this paper, projected a unique sleeping theme for crucial event watching in wsns. the projected sleeping theme may primarily scale back the delay of alarm broadcasting from any node in wsn. for effective transmission of alarm packet, here notice the shortest path within the already wake-up expressed nodes that shortest path algorithmic program relies on distance between the active nodes and about to transmit that crucial event exploitation that shortest path.

References

[1] *PengGuo, Tao Jiang, Senior Member, IEEE, Qian Zhang, Fellow, IEEE, and Kui Zhang "Sleep Scheduling for Critical Event Monitoring in Wireless Sensor Networks" ieee transactions on parallel and distributed systems, vol. 23, no. 2, february 2012.*

[2] *N. Bouabdallah, M.E. Rivero-Angeles, and B. Sericola, "Continuous Monitoring Using Event-Driven Reporting for Cluster-Based Wireless Sensor Networks," IEEE Trans. Vehicular Technology,vol. 58, no. 7, pp. 3460-3479, Sept. 2009.*

[3] *M.I. Brownfield, K. Mehrjoo, A.S. Fayez, and N.J. Davis IV., "Wireless Sensor Network Energy-Adaptive Mac Protocol," Proc. Third IEEE Consumer Comm. and Networking Conf., pp. 778-782,Jan. 2006.*

[4] *S.C. Ergen and P. Varaiya, "TDMA Scheduling Algorithms for Wireless Sensor Networks," Wireless Networks, vol. 16, no. 4, pp. 985-997, 2010.*

[5] *S.C.-H. Huang, P.-J. Wan, X. Jia, H. Du, and W. Shang, "Minimum-Latency Broadcast Scheduling in Wireless Ad Hoc Networks," Proc. 26th IEEE Int'l Conf. Computer Comm., pp. 733-739, May 2007.*

[6] *Keshavarzian, H. Lee, and L. Venkatraman, "Wakeup Scheduling in Wireless Sensor Networks," Proc. Seventh ACM Int'l Conf. Mobile Ad Hoc Networking and Computing, pp. 322-333, May 2006.*

[7] *G. Lu, B. Krishnamachari, and C. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks," Proc. 18th IEEE Int'l Parallel and Distributed Processing Symp., pp. 224-230, Apr. 2004.*

[8] *G. Lu, N. Sadagopan, B. Krishnamachari, and A. Goel, "Delay Efficient Sleep Scheduling in Wireless Sensor Networks," Proc. 24th IEEE Int'l Conf. Computer Comm., pp. 2470-2481, Mar.2005.*

Ravisankar.A. received the bachelor's degree in Information Technology from Thiagarajar College Of Engineering, Madurai, Tamil Nadu, India. He is currently pursuing M.E degree in the department of computer science from Adhiyamaan College of Engineering, Hosur, Tamil Nadu, India.