Journal of Advances in Computer Research
Quarterly ISSN: 2008-6148
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 4, No. 4, November 2013), Pages: 61-71
www.jacr.iausari.ac.ir

Time Slot Allocation Algorithm Based On TDMA Protocol In Wireless Sensor Networks

Somayeh Rahmani✉[1], **Ahmad Khademzadeh**[2], **Amir Masoud Eftekhari Moghadam**[1]
1) Department of Electronic, Computer and IT, Qazvin Branch, Islamic Azad University, Qazvin, Iran
2) Head of Education & National Scientific and International Scientific Cooperation Department at CyberSpace Research Institute, Iran, Tehran
somaieh.rahmani@gmail.com; zadeh@itrc.ac.ir; eftekhari@qiau.ac.ir

Received: 2013/06/11; Accepted: 2013/08/15

Abstract

In recent years, many researchers have worked in the area of wireless sensor networks that has led to the variety of algorithms and methods for the collection and transmission of environment information; but still many challenges faced by wireless sensor networks using methodologies are applied in various fields. Such as Media Access Control that due to the media being shared on these networks, there is a possibility of collision and data loss. Since the retransmission increases network latency and consequently reduces network energy, therefore various solutions have been proposed for media access control that their main purpose is media access control and prevention of collisions. Hence media access control protocols requirement, more and more felt. In this study, a free collision algorithm with the acceptable ability to media access channel is presented in which we have tried to reduce the number of time slots and increase the network throughput as much as possible.

Keywords: *Sensor Networks, throughput, time slots, Media Access Control, collision*

1. Introduction

Sensor networks consist of many small and cheap sensor nodes with limited energy and computation power. The nodes communicate with each other and collect and transfer information from their environment toward the base station by using hop by hop mechanism.

In recent years, there was a great leap in wireless sensor network for various applications, such as military, medical, environmental monitoring, industrial buildings and etc. each node in a sensor network typically has sensor, communication, processing and power unit. They can also include motion sensors or GPS [1].

The establishment of a multi-hop sensor network infrastructure for data transmission between sensor nodes, needs to establish a communication link. This communication link is a shared media (air) that for its management should be applied a media access control protocol to avoid the collision [2].

Several strategies have been proposed to solve the problem of access to shared media. These strategies can be divided into three main groups: fixed allocation, allocation based on demand and hybrid allocation [3].

One of the fundamental problems in the design of the media access control protocols is shared channel between sensor nodes so that avoids or minimizes collision [4], [5]. On the other hand, the proposed protocols that clam eliminate interferences, have typically lower throughput and this is because these protocols assign longer period to each node. This means that each node must wait more for retransmission. In applications such as real-time applications that require data to be sent immediately, it may not be acceptable. In other words, in these protocols, number of nodes that can simultaneously send their data is less. Therefore, when selecting a media access control protocol that is optimal in terms of interference, it should also consider the factors such as throughput and latency, not only confined to the low number of collisions. Time slot allocation mechanisms, are known to node coloring problem. In node coloring problem, each time slot is corresponds to a color. In slot allocation mechanisms, the sensor nodes that have the same color (time slot) can transmit data simultaneously without collision occurring [6]. Henceforth we use the word 'color' instead of time slot. Therefore if the number of colors used to coloring be less, then the number of nodes that can transmit data simultaneously will be increased and algorithm will be better in terms of throughput. For this reason, in this study, an algorithm is proposed which uses the routing information in wireless sensor networks that besides the nodes coloring, it reduces the number of using colors and consequently the number of nodes that can transmit data simultaneously and throughput also will be increased.

In this study, previous work is briefly described in Section 2. The proposed algorithm is described in detail in Section 3. The simulation results and analysis are presented in section 4 and the conclusion is described in Section 5.

2. Related works

Much research has been done on time-slot allocation in wireless sensor networks. DRAND randomized distributed algorithm is one of them. In this algorithm there are four states for a node: IDLE, REQUEST, GRANT, and RELEASE. In first, all nodes are in IDLE state, then with a probability that is dependent on the degree of each node, will negotiate with the neighbors through the exchange of messages to access the channel. DRAND algorithm has very time and message overhead and consumes a large amount of energy due to sending many messages that is not suitable for sensor networks. This algorithm will assign time slot to nodes and because of data sending direction in nodes is not clear, it could not allocate same time slot to nodes which are in less than three-hop away from each other [7].

Paper [8] has reduced overhead of DRAND algorithm by using clustering method. Priority is given to clusters which are decreased parallel and in the border cluster has used coordinator to avoid collision. Authors of this paper believe that when network density is high, coordinators and cluster heads are backbone of the network and therefore time slots are assigned only to these nodes and other nods can sleep.

In [9], LEACH method is used for clustering, but size of time slots are not considered fixed and will change as compromise and dynamically. At the beginning of the transmission cycle all nodes have to be awake to receive the scheduler that is transmitted by cluster head. The conflict between clusters is controlled by using simple CSMA method.

SRSA protocol clusters sensor network then each cluster head assigns time slot to the nodes within its cluster independently. There is a problem here that is the nodes which are located in the border area of cluster may interfere with neighboring cluster nodes and consequently lose their transmitted data. To solve this problem each node at the start of its own time slot uses the CSMA method [10].

In LEMMA algorithm each node maintains a time slot table. At first all the cells are empty. This protocol is assumed that routing tree is already determined and the time slot allocation starts from the sink [11]. This algorithm is performed as centralized and execution time and numbers of transmitted messages are very high.

DCA algorithm performs channel allocation as dynamically. The DCA is entirely message driven, i.e. action taken by a node is a function of the type of incoming packet. This algorithm uses a one-hop clustering structure. Each cluster and node are assigned a weight based on the number of connections and ID. Each cluster head knows weights of cluster head of nodes which are in 2-hop away from its own within nodes. A CSMA/CA mechanism has been used throughout the implementation of algorithm and is assumed that there is no collision when implementing the algorithm. A cluster head begin coloring its cluster only when color allocation by the all cluster head which have more weight is completed. Although by using clustering and coloring operations within the same cluster, it has been tried to achieve energy efficiency, but slot allocation to nodes which is in two-hop or less from each other is not possible yet. Due to using color table for all nodes in same cluster, Memory usage also will be increased [12].

Other methods like using tokens for coloring or slot allocation is presented. In [13], a token is created in DFS traversal and when receiving token, each node chooses the smallest allowable color and after sending this information to all neighbors, it puts this color on the token and sends it to his parent. BFS method has been also used for coloring [14].

3. The proposed channel assignment algorithm

In coloring algorithms that already exist, it is argued that if the nodes which are in two-hop or less from each other and have same color, it is possible that collision occur. However, in many cases, not only in tow- hop distance collision does not occur, even it may don't occur in one-hop away.

As shown in Figure. 1 although nodes 1 and 3 are located in the collision area, but because they are transmitter and not receiver, collision does not occur. We have used this point and proposed an algorithm that tried to use minimum number of time-slot and maximum throughput. Unlike similar algorithms, this algorithm has less overhead and is suitable for sensor networks.

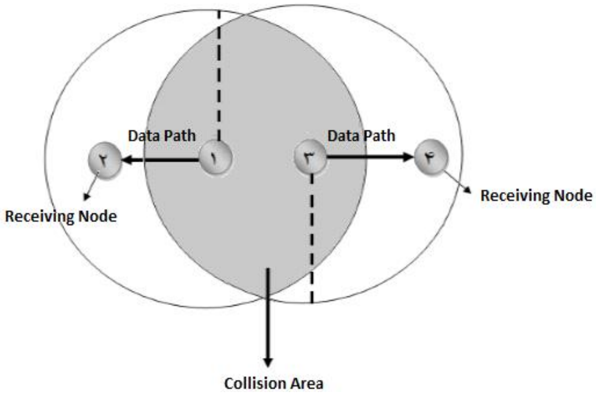

Figure. 1: No collisions occur in less than two steps

3.1. Algorithm Assumptions

This algorithm is an algorithm based on the message. During the coloring process, it is assumed that each sensor node is aware of its routing information; this means that each node knows its next hop which data must deliver to it. Also it is assumed that during the coloring process, sensor network nodes are not damaged. Also no new node is added to the network and nodes are fixed and do not move. Data is transmitted in one direction towards the sink node. The sink node can be considered as central data collection and it is connected to base station which data must deliver to it. The algorithm acts as superficial and coloring process will start at level 1 and until the end of slot allocated its progress.

At the end of the algorithm a color (time slot) will be assigned to each node and they can only send data at time slot of themselves. Each node as a parent node to its children and as a child for its parent has to be considered. Nodes receive data from their children and send it to their parent and this procedure continues until reaching the sink node.

3.2. Message Types

Proposed algorithm uses three type of messages to time slot allocation:

- STARTING
- REPLY
- INFORM

3.2.1. STATRTING Message

This message is generated by the parent node and sends to the children. It consists of an array that contains colors which are diagnosed prohibited by father for child.

3.2.2. REPLY Message

This message is sent from a child node to the parent node. When a child node receives a START message from his father, it sends a REPLY message for his father. This message consists of an array that contains colors, and is not allowed for a child node to choose it. Children fill this array by considering prohibiting own colors which are received from its parent and its 1-hop neighbor and sends it to father through REPLY message. It is the responsibility of parents to decide to assign what color for each child.

3.2.3. INFORM Message

This message is sent by parent node to all 1-hop neighbor nodes. This message is sent when that parent has been received REPLY message from all its children. INFORM Message consists of an array that contains color of itself (parent) and color of children. Recipients of this message are in 2 groups:

- Children. They receive this message and aware of their colors. Besides they aware of their father colors to prohibit it to their children.
- 1-hop neighbor. They receive this message and aware of colors which are prohibited for their children and for themselves.

3.3. sending STARTING message

The algorithm starts by sending a STARTING message from sink node to all children (Figure. 2).

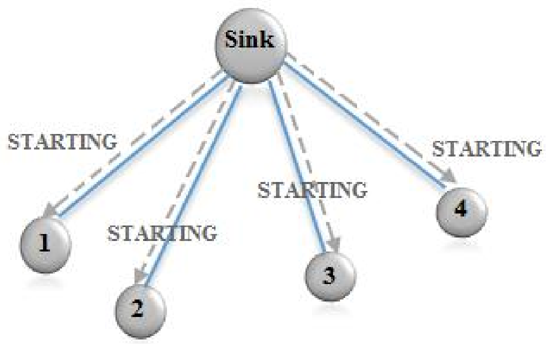

Figure. 2: STARTING message

As noted earlier, this message contains colors that the parent node has prohibited them for its children. We noted that parent node prohibit color of his father (which is the grandfather of the children) to its children. Otherwise, if the color of its child and its parent be the same, the collision occurs at itself.

Since at first the sink node starts the algorithm, no color is prohibited from its neighbors and as a result, no color is prohibited for children and STARTING message which is sent to children is empty.

The array which carries prohibited color for children is called *forbiden_for_child* that its size is dynamic and depends on the degree of the network. In first, all cells of this array filled by '-1'. If a cell of array be filled with '-1' then the index of this cell is color number that parent node has not prohibited it to his children and children are allowed to choose this color (index). On the contrary if a cell of array be filled with the number other than '-1' then the index of this cell is color number that parent node has prohibited it to his children and Children are not allowed to choose this color (index).

For example, suppose that *forbidden - for - child* array filled as Figure. 3. Filling cells 1, 2, 4, 5, 8 and 9 with '-1' means that the colors with number 1, 2, 4, 5, 8 and 9 are permitted by the parents to children to choose them. On the contrary filling cells 3, 6 and 7 with the number other than '-1' imply that that the colors with number 3, 6 and 7 are prohibited by the parents to children.

Receiving this array, children compare it with *Update-color-neighbor* array which placed in INFORM message that are received from one-hop neighbors. The *Update-color-neighbor* contains prohibited color for children. Then children learn all prohibited color for themselves. For example if we consider *forbidden-for-child* array as Figure. 3 which has been received from parent and *Update-color-neighbor* as Figure. 4 which has been received from one-hop neighbors, then all of colors that have been prohibited for children are 1, 3, 5 ,6 ,7 and 9.

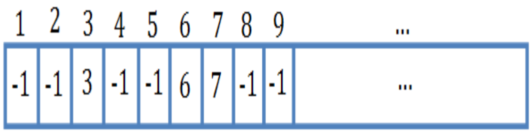

Figure. 3 :forbiden_for_child array

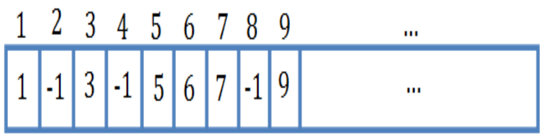

Figure. 4: Update_color_neighbor array

Considering that children may receive too many INFORM messages, they store prohibited colors in an array that called *forbidden-for-self* and It will be constantly updated. Receiving STARTING message, children send Last updated version of this array as INFORM message format to their parents.

3.4. sending REPLY message

As mentioned earlier, REPLY message sent by children consists of all prohibited color for them. The parent node before doing anything, waits to receive REPLY message from all its children. After receiving all message and comparing prohibited colors announced from children, the parent node selects a unique color (among children) for all children.

Color selection is the responsibility of the parent because if it is given to the children, due to there may be no communication link between them, two or more children may have assigned same color in each case, when sending data to the parent collision occur. For example as illustrated in Figure. 5, if node 2 and 3 have same color, when they send data to parent, collision occurs in parent node. Shaded area is collision range.

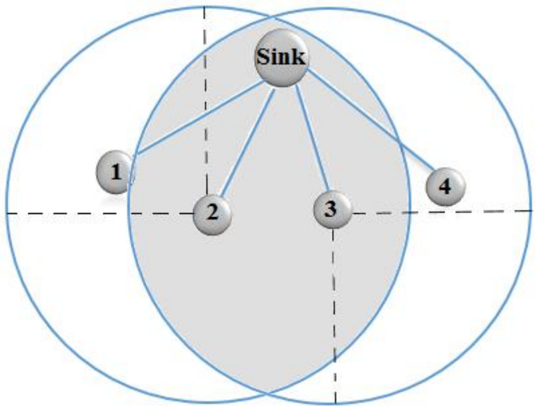

Figure. 5 :Same color in children

3.5. sending INFORM message

After selecting the color of the children, parent node sends an INFORM message to all neighbors (include children and neighbors nodes) which consist of its own color and its children color. This information placed inside *Update color* array.

Children and neighbors nodes after receiving this message performs tasks that are discussed in the below.

Receiving INFORM message, a child node performs two tasks:

- Extracting its assigned color in order to prohibit it for its children. Also broadcast its color to one-hop neighbors to prohibit it for its children.
- Extracting its parent color in order to prohibit it for itself.

It should be noted that each node has two local arrays, *forbiden_for_self* and *forbiden_for_child*. These two arrays keep prohibited color for itself and for children respectively.

Receiving INFORM message, a neighbor node performs two tasks:

- Extracting the color of node that has been sent message in order to prohibit it for its children.
- Extracting the color of children of node that has been sent message in order to prohibit it for itself.

At the end of algorithm, all nodes in the network are assigned a color and each node can sends data at its time-slot in a free collision situation.

4. Simulation Results

In this study, a new algorithm for time-slot allocation to the wireless sensor nodes is proposed. Many of the proposed algorithms, in order to avoid collisions allocate different time-slot to nodes which are placed in two-hop or less from each other.

Although, this time-slot allocation type prevent from collision, but the number of time-slots will be increased and throughput will be decreased. So the network is forced to compensate the throughput rate of the network, increasing its bandwidth of nodes which makes more energy consumption. The algorithms such as DCA although have tried to use clustering to optimize energy consumption within the same cluster, but slot allocation to nodes which are placed in two-hop or less from each other is not possible. Here we evaluate and compare the performance of proposed algorithm with DCA. The simulation is done by the software OMNeT++. This software provides change and configuration capabilities of the network parameters. The nodes are considered in a square area with side of length 750 meters and are distributed randomly. Each node has a radius sending data of 40 meters and links bandwidth are 2Mbps. Factors that have been compared are: the number of assigned color, the throughput network and network delay.

Table 1 shows the characteristics of the network used for the simulation.

Table 1: the characteristics of the network

Software	OMNET++
area	750m×750m
radius data	40m
Links speed	2Mbps

Figure. 6 compares the number of colors used in the proposed algorithm and DCA algorithm in terms of degree of network .Number of colors that are assigned to the nodes in proposed algorithm is less than DCA and is improved significantly .This difference is the fact that in some cases may the nodes which are placed in two-hop or less from each other, have same color without any collision and the proposed algorithm takes advantage of this situation.

If the number of used colors is less, then each node sends data in shorter time period. For example if a network is used 20 colors for coloring, assuming that each data transmission lasts 10ms and 1ms take too long to be delayed until the next transmission then period of time of the whole network and each node to transmit data will be 220ms (20*(10+1)). Now if the number of used colors be 10 colors, then period of time to transmit data will be 110ms (10*(10+1)) and this makes increasing in the throughput of network (In this example, when the data is continuous flooding has doubled approximately) and consequently the network latency is reduced.

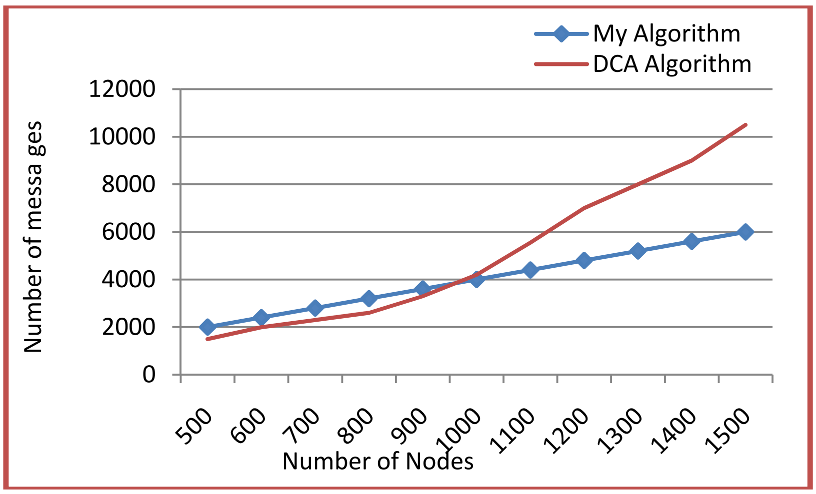

Figure. 6: Comparison between the numbers of colors allocated in the proposed algorithm and DCA

The number of messages for the implementation of the algorithm is another important factor in the network coloring .With more number of messages ,the algorithm will incur additional overhead and moreover the energy consumption of network is more.

During the proposed algorithm, each node sends four messages. These messages include STARTING message that will be sent to the children, REPLY message that will be sent to the parent, INFORM message which is sent from the parent node to its neighbors and INFORM message that is sent from the children node to its neighbors.

In Figure. 7 Total numbers of messages exchanged by the two algorithms are compared. As shown in the figure, when the number of nodes in the network is less than 1000, the number of messages in the proposed algorithm is more, but in networks with more than 1000 nodes, the proposed algorithm uses less messages and this difference will be exponentially more.

Because DCA algorithm uses the clustering, messages must be exchanged between cluster heads that include the color of the cluster members and furthermore, each node inside cluster must also exchange many messages with its cluster head and neighbor. With the increasing number of nodes, more messages are needed for this coordination. The results shown in Figure. 7 are quite clear.

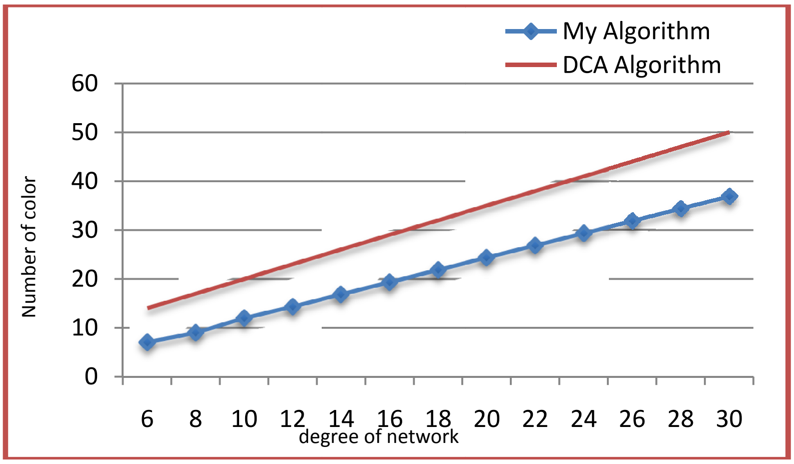

Figure. 7 :Comparison between proposed algorithm and DCA in terms of number of message

Figure. 8 compares the throughput of the network between the proposed algorithm and DCA .Throughput is defined as the amount of information transmitted per second in the entire network.

It is obvious that whatever the number of colors allocated to the nodes is less, the number of nodes that can send data at a time is more. For example when a network is colored with seven colors, this implies that all nodes in the network after a period of seven were able to send their information and if network is colored with ten colors, this means that all nodes in the network after a period of ten were able to send their information. Therefore, the network throughput is inversely related to the number of colors allocated to the network. Because number of colors used in the proposed algorithm is always less than the DCA algorithm consequently the proposed algorithm has a higher throughput.

In the simulation, the number of nodes is 1000 and the throughput has been calculated at the time of 1000ms.

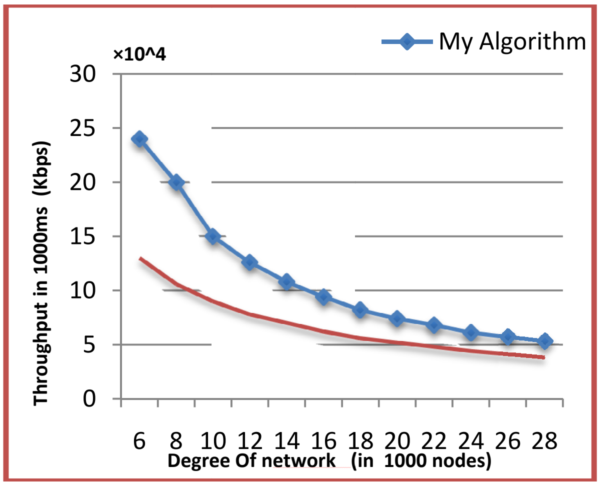

Figure. 8: the comparison of throughput between the two algorithms

Figure. 9 shows the comparison of delay between two algorithms. These factors, as well as Throughput factor are related to the number of colors allocated but it is a direct relation. This means that if number of colors allocated be less than the network latency will be lower and vice versa. As another example, consider a network that be colored with 7 colors. In this case, each node can send data again after a period of seven. If we assume as before that each data transmission lasts 10ms and 1ms take too long to be delayed until the next transmission, once sending, each node should wait 77ms (7*(10+1)). Now if the number of colors used is 10 colors then each node should wait 110ms (10*(10+1)). Because the proposed algorithm uses the smaller number of colors, therefore delays will be less.

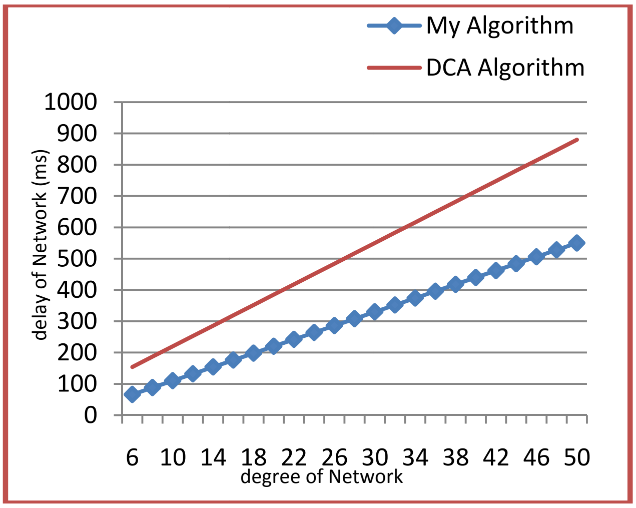

Figure. 9: the comparison of delay between the two algorithms

5. Conclusions

Due to the variety of applications for wireless sensor networks, it should be applied a media access control protocol to avoid the collision. Many efforts have been made towards this goal. When selecting a channel access control protocols that is optimal in terms of collision, the factors such as throughput, latency, and other factors will also be considered and not confined only to the low number of collisions. In some algorithms such as DCA, the minimum distance is considered as 3-hop while in a sensor network, a significant number of nodes that are placed in two-hop or less from each other interference will not occur in their data. By reducing the distance coloring between nodes that are placed in three-hop, network throughput will be improved and in this paper, we use this feature and proposed an algorithm that according to the simulation results its performance rather than DCA algorithm has improved and this improvement is also visible in throughput, number of time-slots and the delay factors.

6. References

[1] J. Chen, Xi. Cao, P. Cheng, Y. Xiao, Y. Sun. 'Distributed collaborative control for industrial automation with wireless sensor actuator networks', IEEE Transactions on Industrial Electronics 57 (12) ,pp.4219– 4230, 2010..

[2] M.O. Daz-Anadń, K.K. Leung, 'A test-based scheduling protocol (TBSP) for periodic data gathering in wireless sensor networks', 3rd International Workshop on Multiple Access Communications (MACOM), in: A. Vinel (Ed.), Lecture Notes in Computer Science, Springer, Barcelona, pp. 25–35, 2010.

[3] Q. Sun, V.O.K. Li, K.-C. Leung,' Adaptive topology-transparent distributed scheduling in wireless networks', in: Proceedings of the IEEE International Conference on Communications (IEEE ICC 2010), South Africa, pp. 23–27, May 2010.

[4] M. Wang, L. Cib, and Y. Xua, 'D-RDT: a Multi-channel MAC Protocol for Dense Multi-hop Wireless Sensor Networks' Advanced in Control Engineering and Information Science Procedia Engineering 15 , pp.2155 – 2159, 2011.

[5] A. Bachir, M. Dohler, T. Watteyne, K. Leung, 'MAC Essentials for Wireless Sensor Networks, IEEE Communications Surveys Tutorials 12' , pp.222 –248, 2010.

[6] H. Ted, T. Sebastien,'A Distributed TDMA Slot Assignment Algorithm for Wireless Sensor Networks ' In Proceedings of the Seventh International Conference on Mobile Computing and Networking, pp.221-235, 2004.

[7] R. Injong, W. Ajit, M. Jeongki and Xu Lisong,'DRAND: Distributed Randomized TDMA' in MobiHoc ACM Press, pp. 190-201, 2006.

[8] L. Shihan, Q. Depei, L. Yi, T. Jie, 'Adaptive Distributed Randomized TDMA Scheduling For Clustered Wireless Sensor Networks', conference wireless network IEEE, pp.2688–2691, 2007.

[9] K. Ryouhei, M. Toshiaki, 'Effective Sensing Function Allocation Using a Distributed Graph Coloring and a Slot Allocation Algorithm in Wireless Sensor Networks', International Conference on Advanced Information Networking and Applications, pp.906–913, 2009.

[10] W. Tao, B. Subir, 'Minimizing inter-cluster interference by self-reorganizing MAC allocation in sensor networks', Springer Wireless Netw pp.691–703, 2007.

[11] N. Mário, G. António and M. Mário, 'Interference-Free TDMA Slot Allocation in Wireles Sensor Networks', IEEE Conference on Local Computer Networks, pp.239-241, 2007.

[12] C. Kaushik, N. Nagesh, C. Pritam and A. Dharma, 'Channel allocation and medium access control for wireless sensor networks' Elsevier Ad Hoc Networks, pp. 307–321, 2009.

[13] C. E. Sinem, V. Pravin, 'TDMA scheduling algorithms for wireless sensor networks', Journal Wireless Networks - WINET Springer, pp.985 - 997, 2009.

[14] G. Shashidhar, Z. Ying, H. Qingfeng, 'Distributed time-optimal scheduling for convergecastin wireless sensor networks' Computer Networks Elsevier, pp. 610-629, 2008.