DOI. 10.1109/WISP.2015.7139151

General Slot Stealing TDMA Scheme to Improve the Low Channel Utilization Factor

Valentin Stangaciu, Mihai V. Micea, Vladimir I. Cretu
Computer and Software Engineering Department
Politehnica University of Timisoara
2, Vasile Parvan Blvd., 300223, Timisoara, Romania
valys@dsplabs.cs.upt.ro, mihai.micea@cs.upt.ro, vladimir.cretu@cs.upt.ro

Voicu Groza
School of Electrical Engineering and Computer Science
University of Ottawa
800, King Edward, Ottawa, KIN 6N5, Canada
groza@site.uottawa.cs

***Abstract*—Time division multiple access (TDMA) schemes are intensively being the main solution to provide predictability and timeliness in data communication systems requiring real-time operation. In this paper we investigate the problem of low efficiency and channel utilization, common to the TDMA algorithms. As a solution, we propose a general slot stealing version of TDMA, able to significantly increase the channel utilization factor, while preserving the required level of predictability, both in wired and in wireless real-time networks.**

Keywords— TDMA; slot-stealing; real-time; wired sensor networks, wireless sensor networks

I. Introduction

Medium access protocols have been studied lately due to the high demand in nowadays wired or wireless communication of embedded devices. These protocols can be divided into two major categories: *contention based* (CSMA) where nodes randomly try to access the communication medium in a competing mode and *scheduling based* (TDMA, FDMA) where the medium is usually divided in such a way that contention is strongly reduced or even eliminated [1].

The latter type of protocols are the most suitable for real-time applications mainly because they are highly predictable thus this characteristic is mandatory for a real-time system [2].

TDMA based medium access protocols are highly predictable and are suitable for real-time applications but they also have a number of disadvantages. One of the most important disadvantages is that the nodes of a wireless or wired network using this access scheme have to be perfectly synchronized in order to detect the correct timeslot. This can be easily overcome by either using appropriate time synchronization protocols [3] or by using coordinated beacon-based network topologies [4, 5].

Another important disadvantage is that in many situations some nodes of a network have a large amount of data to transmit while other nodes are silent in terms of communication. In this case a TDMA algorithm not only provides low channel utilization, but also limits the high transmission demand of some network nodes.

A TDMA protocol can be used as a medium access protocol in both wired and wireless networks where the network nodes share the same communication medium. The medium is divided into timeslots which are assigned to the network nodes for a limited or unlimited period of time. The length of a timeslot is usually taken for the maximum data packet to fit [6]. In a classic TDMA scheme each node is restricted to its assigned timeslot. The number of timeslots is usually equal to the number of nodes present in the network.

In this paper we provide a solution to the low channel utilization problem by proposing a general slot stealing TDMA based access scheme that also offers the chance to transmit data to nodes requiring high channel utilization. Furthermore our approach may be used in both wired and wireless networks.

II. Related work

Many of the studies regarding TDMA based access protocols have been conducted in order to reduce communication delay when dealing with networks with numerous nodes or to reduce idle channel times when some nodes are silent while other nodes have high data transmission demands. Also many of the studies have been concentrated on medium access protocols for wireless sensor networks.

To overcome some of the TDMA disadvantages while maintaining its predictability, a hybrid solution was proposed combining both TDMA and FDMA techniques in order to reduce the TDMA frame size while using multiple frequencies for communication [7]. Such a solution can offer good real-time results but is limited due to the need of frequency hopping, which in a wireless sensor network with numerous nodes can be costly in terms of battery life and pricing.

Another idea was presented in [8] where a combination between CSMA and TDMA is proposed. The authors assign a time slot for each node but the access to the time slot is contention based with the restriction that the timeslot owner has the highest access priority. However, when contention is involved, predictability is highly affected.

One of the most implemented protocol stack is ZigBee [9] which has the IEEE 802.15.4 standard as its medium access protocol. Many of current applications are based on this stack disregarding the real-time issues and unreliability of the access protocol [10] even with its later improvement for a minimal real-time supported via GTS. Giving the fact that this stack is

This work was partially supported by the strategic grant POSDRU/159/1.5/S/137070 (2014) of the Ministry of National Education, Romania, co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resources Development 2009-2013

highly used some researchers offer improvement solutions integrating TDMA based extensions [11].

During the last period a new concept has been proposed by many researchers, focusing to improve the slot assignment to nodes in order to offer the chance for higher channel access for nodes with high data transmissions demand. The concept is presented in the literature as *slot-stealing*. However, this concept is highly restrictive in the available work. For example, a slot-stealing access scheme is presented in [12] and later improved in [13] where authors improve the capacity of data transmission using the TDMA scheme where the nodes are also allowed to transmit in a non-owned timeslot in the situation where the owner of the timeslot is not transmitting. The authors also impose high restrictions: their algorithm only applies to static, beacon based wireless sensor networks with cluster tree organization. Furthermore the nodes allowed for slot-stealing have to be on the same tree level and must have the same parent.

Based on a similar idea, the sTDMA protocol is presented in [14], where a node may use timeslots of other nodes, but only by request. This improves channel utilization but introduces some overhead messages for the requests.

Another important approach is a slot-stealing protocol [15] that can be influenced by higher protocols in the communication stack, like the routing protocols. Based on active routes timeslots can be taken from silent nodes and assigned to high communication demand nodes.

III. Greedy TDMA Access Scheme

In order to present our proposal we shall consider a network with *n* nodes communicating in a classic TDMA scheme each having its own timeslot for an unlimited period of time. The basic idea of the *slot-stealing* concept is that, considering two nodes of the network, designated as node *i* and *j*. with node *i* being scheduled before node *j* in the TDMA frame and also considering that node *i* has no data to transfer in its current timeslot, then node *j* is allowed to send its data in the unused timeslot of node *i* if its data has already been prepared. This situation is also presented in Fig. 1.

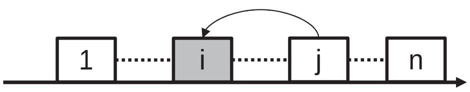

Fig. 1. Slot stealing concept

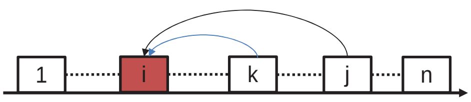

Fig. 2. Slot stealing concept failure

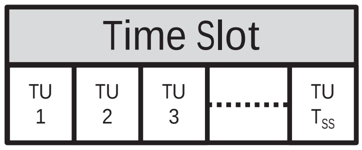

Fig. 3. Discrete timeslot

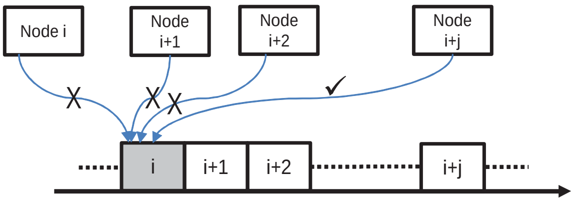

Fig. 4. Slot-stealing rule for timeslot i and node i+j

The situation presented in Fig. 1 can easily lead to a conflict situation, presented in Fig. 2. For example if another node, designated as *k*, with *i*<*k*<*j*, has data ready to be transmitted and wants to obtain the unused timeslot of node *i* at the same time with node *j*, then both nodes will transmit in the same time, resulting in a collision.

Before presenting our solution we will consider a discrete version of a timeslot which is being divided into TS_S time units (TU) as depicted in Fig. 3. We will also define the capacity of a time unit (CAP_{TU}) which represents the number of bytes a node can transfer within a time unit. A timeslot must have enough time units with a suitable capacity in order to allow a node to transmit a maximum sized packet of data. Furthermore we consider that each node has a unique address that corresponds to the index of the timeslot within the TDMA frame.

In general, our solution states that a node can steal the timeslot of another node only if it does not detect a transmission, not only from the node it wants to steal the timeslot from, but also from the nodes between them.

In Fig. 4 we show that if a node designated as *i* has no need of the channel in the current timeslot and node (*i+j*) has data ready for transmission then this node has to check whether all the nodes from node *i* to node *(i+j)* will be silent during the current timeslot, thus they won't be stealing the timeslot for themselves. Using this idea the conflicting situation will be eliminated. The disadvantage is that the detection of silent nodes consumes time from the current timeslot and if there are too many nodes between *i* and *(i+j)* then a large amount of time will be consumed by node *(i+j)* for the detection of the silent nodes, thus remaining less time for actual data transmission. This implies the restriction that the node *(i+j)* can transmit only if the data packet can fit in the remaining part of the timeslot and only after it has detected that no other node has already stolen the timeslot.

A much simplified situation is presented in Fig. 5. We shall consider the timeslot is 8 TU wide and a 5 node network with each node having its own timeslot allocated by a consecutive address. We also consider that the current timeslot in

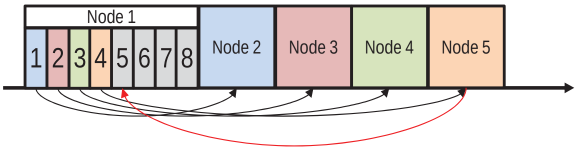

Fig. 5. Greedy TDMA concept example

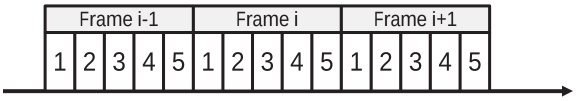

Fig. 6. Greedy TDMA frame iteration

communication is the timeslot of node 1 which currently has no data ready to transmit.

In this state, node 1 will not occupy the channel thus leaving its timeslot free to be used by the other nodes. For instance, if node 2 detects no activity from node 1 during the time unit 1, then it could simply steal its timeslot, if it had already prepared data to transmit. The case is slightly different for node 3 which first must detect if no data is transmitted by node 1 in time unit 1 and also if node 2 did not steal the timeslot. Only then, node 3 is free to take ownership of the remaining time units of the current time slot.

In Fig. 5 we consider that in the current timeslot only node 5 has data prepared to transmit, the other nodes being silent. In this situation, applying the rules defined above, node 5 will have the chance to take ownership of the current timeslot only from time unit 5, after detecting the silence of all the other nodes with lower addresses. The only restriction is that the remaining timeslot capacity has to be greater or equal to the size of the packet that node 5 has prepared to transmit over the channel.

In a classical TDMA access scheme this timeslot would have been idle and therefore lost, mainly because the owner of the timeslot was not prepared to transmit. Moreover, timeslots 2, 3 and 4 would have been lost due to the lack of activity from the corresponding nodes. Using our proposed Greedy TDMA mechanism, this timeslot will not totally be lost, giving a chance of transmission to the only node that has data ready to transmit, in our case node 5. If the data processing of this latter node keeps up with the timeslot synchronization, it can still use its initially allocated timeslot as it would in a classical TDMA access protocol.

On a further analysis one can observe that, for example, node 5 has the possibility to steal timeslots from node 1 to 4 and node 4 can steal timeslots from node 1 to 3. In the currently presented scheme, node 1 has no possibility to steal any timeslots, being the first node in the TDMA frame. A possible solution can be found by analyzing Fig. 6 based on the previously presented example. Here, we consider 3 iterations of the TDMA frame. It can be easily observed that node 1 has also the possibility to steal unused slots of nodes 5, 4, 3 and 2, even if the timeslots belong to another frame. The only restriction which has to be applied is that the time slot borders have to be respected. Any transmissions must end at a timeslot border, to avoid corrupting the following timeslot.

IV. Experimental Results

In order to validate the proposed Greedy TDMA algorithm, a custom designed simulator has been developed. The simulator uses a discrete version of the timeslot with the TU as base time unit. The main input parameters of the simulator are: the number of nodes, time units per timeslot (TS_S), the time unit capacity CAP_{TU}, the number of TDMA frames to be simulated, the maximum message size and the transmission probability.

First, the simulator generates the simulation environment, which is composed of a pair of randomly generated packet size along with a randomly generated transmission probability for each node and for each frame as well. Thus, the generated environment is known at all times throughout the simulation. The simulation environment variables are generated using a linear random algorithm and will be applied both to the classical TDMA algorithm and to the proposed Greedy TDMA method. The current limitation of the simulator is that the first node of a timeslot cannot use the slot stealing technique.

In the following paragraphs we present an example simulation which is highly connected to the concept example depicted in Fig. 5.

The simulation input parameters are detailed in Table I. The message size for each node on each timeslot is randomly generated with a maximum size of 64 bytes.

TABLE I. Simulation Parameters

Node count	5
TS_S	16
CAP_{TU}	4
Simulate TDMA Frames	8
Maximum message size	64
Transmission probability	60

The simulation environment that was generated is presented in Table II. The two integers shown in each cell represent the randomly generated packet size, followed by the randomly generated probability of transmission in percentage. For example in timeslot 2, node 1 will have prepared a packet of 57 bytes, with a transmission probability of 45%.

TABLE II. Simulation Environment

	Node 1	Node 2	Node 3	Node 4	Node 5
Frame 1	(57, 88)	(51, 79)	(43, 45)	(19, 52)	(19, 85)
Frame 2	(57, 45)	(25, 88)	(28, 41)	(34, 93)	(33, 95)
Frame 3	(18, 61)	(47, 29)	(2, 50)	(18, 27)	(22, 48)
Frame 4	(25, 27)	(57, 7)	(22, 47)	(46, 76)	(36, 93)
Frame 5	(56, 47)	(58, 73)	(30, 45)	(14, 29)	(26, 52)
Frame 6	(29, 62)	(39, 42)	(12, 27)	(34, 48)	(1, 4)
Frame 7	(61, 25)	(12, 35)	(41, 37)	(36, 30)	(44, 3)
Frame 8	(33, 1)	(19, 22)	(29, 38)	(44, 93)	(25, 8)

The generated environment presented above has been used in simulating both the TDMA access scheme and the Greedy TDMA access scheme in order to evaluate the performances of the newly presented slot stealing method. The simulation results are presented in Table III.

TABLE III. SIMULATION RESULTS

	Total transferred bytes	Total Time units	Total timeslots	Last used TDMA frame	
				TDMA	*Greedy TDMA*
Node 1	232	61	5	8	8
Node 2	174	45	5	8	6
Node 3	207	55	8	8	4
Node 4	121	32	5	7	5
Node 5	118	32	5	8	6

Analyzing the results presented in Table III, one can identify that in the TDMA scheme almost all the nodes finish their data in the last TDMA frame, in our case frame 8. However, when using the proposed scheme, most of the nodes finish all their data transmissions earlier by using the timeslots efficiently.

V. CONCLUSIONS

In this paper we present a general TDMA time slot stealing algorithm that can be successfully applied in both wired and wireless networks. The Greedy TDMA solution was validated using a custom simulator that shows the high advantage over the classic TDMA scheme. As a future work we plan to apply this scheme in order to obtain a real-time medium access protocol for wireless sensors networks.

REFERENCES

[1] D. Dewasurendra and A. Mishra, "Design Challenges in Energy-Efficient Medium Access Control for Wireless Sensor Networks," in Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems, M. Ilyas and I. Mahgoub, Eds., ed Boca Raton, Florida: CRC Press, 2005, ISBN: 0-8493-1968-4, pp. 314-329.

[2] G. Buttazzo, "Research trends in real-time computing for embedded systems," SIGBED Rev., vol. 3, no. 3, pp. 1-10, 2006.

[3] B. Statulat, S. Barzeianu, M. V. Micea, and V. Groza, "Wireless Synchronization Protocols for Collaborative Robotic and Sensor Environments," in 19th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD), Budapest, Hungary, 2010.

[4] IEEE Computer Society, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless personal Area Networks (LR-WPANs)," in 802.15.4, ed: Institute of Electrical and Electronics Engineers, Inc., 2003.

[5] G. Franchino and G. Buttazzo, "WBuST: A real-time energy-aware MAC layer protocol for wireless embedded systems," in Emerging Technologies & Factory Automation (ETFA), 2012 IEEE 17th Conference on, 2012, pp. 1-8.

[6] A. Ephremides and O. A. Mowafi, "Analysis of a Hybrid Access Scheme for Buffered Users-Probabilistic Time Division," Software Engineering, IEEE Transactions on, vol. SE-8, no. 1, pp. 52-61, 1982.

[7] D. Abdeli, S. Zelit, and S. Moussaoui, "RTH-MAC: A real time hybrid MAC protocol for WSN," in Programming and Systems (ISPS), 2013 11th International Symposium on, 2013, pp. 153-162.

[8] R. Injong, A. Warrier, M. Aia, M. Jeongki, and M. L. Sichitiu, "Z-MAC: A Hybrid MAC for Wireless Sensor Networks," Networking, IEEE/ACM Transactions on, vol. 16, no. 3, pp. 511-524, 2008.

[9] ZigBee Alliance, "ZigBee Specifications," version 1.0 r13 Dec 2006.

[10] G. Anastasi, M. Conti, and M. Di Francesco, "A Comprehensive Analysis of the MAC Unreliability Problem in IEEE 802.15.4 Wireless Sensor Networks," Industrial Informatics, IEEE Transactions on, vol. 7, no. 1, pp. 52-65, 2011.

[11] K. Jungsook, L. Jaehan, C. Pelczar, and J. Byungtae, "RRMAC: A sensor network MAC for real time and reliable packet transmission," in Consumer Electronics, 2008. ISCE 2008. IEEE International Symposium on, 2008, pp. 1-4.

[12] S. Gobriel, D. Mosse, and R. Cleric, "TDMA-ASAP: Sensor Network TDMA Scheduling with Adaptive Slot-Stealing and Parallelism," in Distributed Computing Systems, 2009. ICDCS '09. 29th IEEE International Conference on, 2009, pp. 458-465.

[13] J. Yackovich, D. Mosse, A. Rowe, and R. Rajkumar, "Making WSN TDMA Practical: Stealing Slots Up and Down the Tree," in Embedded and Real-Time Computing Systems and Applications (RTCSA), 2011 IEEE 17th International Conference on, 2011, pp. 41-50.

[14] C. Jenhui and J. Shuhua, "Improvement of Slots Utilization with a Stealing-TDMA Protocol for Ad Hoc Network," in Vehicular Technology Conference, 2006. VTC-2006 Fall. 2006 IEEE 64th, 2006, pp. 1-5.

[15] H. Pei, L. Chin-Jung, and X. Li, "TAS-MAC: A traffic-adaptive synchronous MAC protocol for wireless sensor networks," in Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2013 10th Annual IEEE Communications Society Conference on, 2013, pp. 113-121.