Superframe Formation Algorithms in 802.15.3 Networks

Attila Török [1], Lóránt Vajda [1]
Dept. of Telecommunications and Media Informatics
Budapest University of Technology and Economics
Budapest, Hungary
{torok,vajda}@ttt-atm.tmit.bme.hu

Kyu-Jung Youn, Sun-Do June
Wireless PAN Technology Project Office
Korea Electronics Technology Institute
Seoul, Korea
{kjyoun,jsd}@keti.re.kr

Abstract— **The IEEE 802.15.3 wireless standard aims to achieve high performance and scalability by using a hybrid, time-slotted MAC protocol with variable length superframes. The design of a proper superframe formation algorithm is not trivial. This can be attributed to some limitations in the possibilities of superframe resizing. Based on our findings, we propose a novel superframe formation algorithm, which avoids the pitfalls of current algorithms. A comprehensive performance analysis with various performance metrics and different types of traffic is presented in the paper. The proposed algorithm is a low complexity solution that supports power-saving and proper treatment of CBR traffic, while it provides higher channel utilization, higher error tolerance and stability.**

Keywords: superframe formation, 802.15.3, Medium Access Control, Multimedia QoS Support

I. INTRODUCTION

In the last decade, wireless networks gained an important role in spreading new networking services among users with special demands. The wide range of emerging applications triggered the appearance of specialized wireless protocols. Bluetooth represents such a specialized solution, with a primary target of building cheap and simple Personal Area Network (PAN) [10]. It uses a frequency hopping technique, and its MAC protocol is based on a centralized (master-slave hierarchy) and connection-oriented ad-hoc networking architecture. Targeting applications with different demands, such as broadband and Quality of Service (QoS) requirements, the 802.15.3 opts to use a different channel access technology [6]. Similar to some wireless ATM protocols [2], it uses a hybrid MAC protocol that employs a time-slotted superframe structure. The networking architecture in 802.15.3 is also based on a master-slave type ad-hoc network, but unlike in Bluetooth, the master (called PNC) has only the responsibility of network and resource management, without the packet forwarding functionality.

The time-slotted superframe structure can be separated to a *channel request* part and a *data transmission* part. For channel requests 802.15.3 uses a contention-based protocol, while for data transmission it uses time division multiple access (TDMA) based time slot allocations. These kinds of hybrid protocols have three phases: channel request, scheduling, and data transmission. The channel request phase is used by the nodes to send their requirements to the master. Using the gathered information, the master can schedule the time slots for the next superframe. Finally, in the data transmission part the nodes transmit their data to other nodes in time slots specified by the master. In order to optimize channel usage, some protocols adopt variable length superframes [2]. The 802.15.3 system uses variable length not only for superframes, but also for the different components of the superframe (data slots, channel request part, etc.). Unfortunately, the variable length superframes also introduce some difficulties in assigning capacity for nodes with strictly timed applications [1].

In this paper we investigate the possibilities allowed by the 802.15.3 standard for superframe formation algorithms. As we will show, the design of a proper superframe formation algorithm is not trivial. This can be attributed to some limitations in the possibilities of superframe resizing and the urge to support special features, such as power-saving [5] and proper treatment of strictly timed applications. In previous papers dealing with variable length superframes [2][3] wireless ATM networks were considered; this assumes different design choices than in 802.15.3. As far as we know, the only work that deals with 802.15.3 networks is presented in [9]. Here the authors present a performance analysis of two superframe formation (*static* and *dynamic*) algorithms. They show, through simulation results, that the dynamic algorithm exhibits better performance than the static one. The dynamic algorithm considered in this work is a hypothetical solution, which cannot be implemented in a real 802.15.3 system. In order to have a more realistic scenario, we designed and implemented a superframe sizing algorithm, called *gradual superframe sizing algorithm*. This algorithm was designed to suit the specifications of the 802.15.3 standard.

During the performance analysis of the above mentioned (static and gradual) algorithms we found some drawbacks that lead to network resource wastage. Based on our findings, we designed a new superframe sizing algorithm, called *hierarchical superframe formation*, which avoids the pitfalls of the previous solutions. We demonstrate through simulations that our system outperforms the other algorithms in terms of channel utilization, error tolerance and stability, while it satisfies special demands as power saving as well.

This paper is organized as follows: Section II provides a brief overview of the 802.15.3 system and the superframe formation algorithms. Section III describes the proposed algorithm, while Section IV provides the performance analysis. Finally, Section V concludes the paper.

[1] This work was carried out while working at Korea Electronics Technology Institute, Wireless PAN Technology Project Office, 2003.

II. SUPERFRAME FORMATION IN 802.15.3 NETWORKS

In this section we present the possibilities allowed by the 802.15.3 standard for superframe formation. We also describe the *gradual superframe formation algorithm*, designed to take advantage of these possibilities. Some drawbacks of this algorithm are presented as well.

An 802.15.3 piconet is composed from a master node (PNC) and several slave nodes, which are in the radio range of the PNC. The slave nodes can communicate directly with each other; only the connection establishment involves the PNC. In the region of the *parent piconet, child* and *neighbor piconets* can be created on the same frequency. These special piconets will get pseudo-static channel time allocations (CTA) from the superframe of the parent PNC. This CTA is used as the superframe of the neighbor piconet, which is managed by an elected PNC.

The MAC layer of the protocol employs a time-slotted superframe structure, constituted from the *beacon*, the optional *Contention Access Period* (CAP), and the *Contention Free Period* (CFP). The beacon is used to carry control and channel allocation information to the entire piconet; it is built and broadcasted by the PNC. The CAP is used for association request/response, channel time request/response, and to exchange asynchronous data. The CFP is composed of *Guaranteed Time Slots* (GTS) and *Management Time Slots* (MTS) in a TDMA frame structure. The GTS slots are used by slaves for isochronous and asynchronous data communication. These slots can be dynamic or pseudo-static, depending on whether their relative position within the superframe can vary from one superframe to the next, or not.

A. Superframe formation algorithms

Superframe formation in 802.15.3 can be *static* or *gradually dynamic*. In the case of *static superframe formation* the PNC uses a constant superframe size; thus, the beacons are broadcasted periodically.

The 802.15.3 allows changing the size of the superframes [6]. If the PNC decides to change the length of the superframe, then, for a certain amount of time, the desired superframe length must be advertised. After this interval it can switch to the new length. The main reason for the size of the superframes to be changed only after the advertisement period is the presence of pseudo-static channel time allocations (CTA).

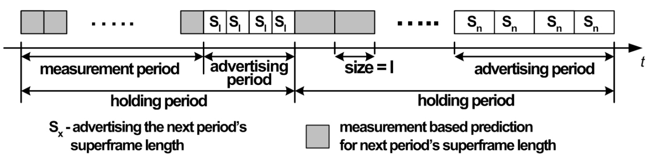

Figure 1. Gradual superframe sizing

We designed the *gradual superframe formation algorithm* so as to take advantage of this feature. The main idea was to propose an algorithm that dynamically adapts to the network load, trying to optimize the channel utilization. To optimize the superframe size the algorithm needs some information about the upcoming network load. Therefore, piggybacking techniques are used to inform the PNC about the nodes internal status [1][3][4][9] (this technique is used during all the analysis). In our case the internal status is the number of packets in the queue of the flow at sender nodes. By using this information, the PNC can schedule the flows in an optimal manner.

In the gradual algorithm, the *measurement period* (see Fig. 1) is used to gather information about the network load, and to estimate the impending traffic. At the end of this period the superframe size for the next cycle is predicted. This size will be advertised in the beacons of the *advertising period*. After the advertising period ended, the PCN can switch to the new superframe size.

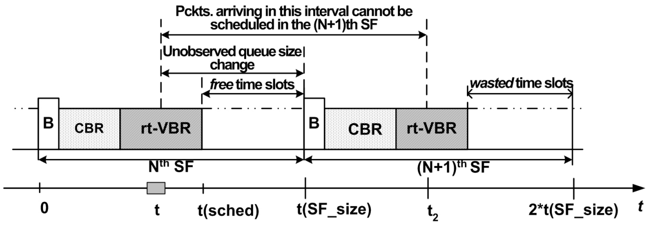

Figure 2. Problems with static length superframes

B. Problems with the superframe formation algorithms

The constant superframe size used by the static algorithm has some drawbacks. In Fig. 2 two consecutive static superframes are presented, with their beacon (B), CBR and real-time Variable Bit Rate (rt-VBR) traffic portions. Let us assume that node A has allocated time slots for its rt-VBR traffic at time interval *t*. By the time when A sends the packets in the allocated interval, the PNC can get information about the node's queue status. This information is used by the PNC to optimize the scheduling for the next, $(N+1)^{th}$ superframe. Unfortunately, node A stops sending packets after the elapse of its allocated time slots. Therefore, the PNC will not be able to get any information about the queue status of node A until the next beacon. Packets generated/arrived during the (t, t_2) interval cannot be scheduled for the $(N+1)^{th}$ superframe. This can lead to suboptimal scheduling, which means that a portion of the $(N+1)^{th}$ superframe (called *wasted time*) remains unutilized. Moreover, the PNC has a different view about the queue status of the node, which affects its decision-making and scheduling performance for the next superframe interval. We call the difference between the real length of the node's queue and the length known by the PNC *queue size inconsistency;* in our simulations this information is used as a performance metric.

The gradual algorithm tries to overcome the appearance of wasted and free time slots by modifying the superframe size. Unfortunately, the size of the superframe can be changed only after the advertising period, which can be relatively long in some cases (e.g., presence of neighbor piconets). Thus, the reaction of the system to network load fluctuation is slower than for an instantaneously reacting superframe sizing algorithm. On the other hand, if the size of the superframes is changed too frequently, accurate timing and position for strictly-timed applications cannot be guaranteed. As we will show, the CBR flows suffer from using the gradual algorithm.

III. THE HIERARCHICAL SUPERFRAME FORMATION ALGORITHM

A. Concept and algorithm description

In order to avoid the drawbacks of the aforementioned algorithms, we propose a new solution, called the *hierarchical superframe formation algorithm*.

To keep the proposed algorithm simple, we used fixed size periods for superframes (called superframe-periods). On the other hand, the resizing of the superframes is advantageous for more effective bandwidth utilization. Therefore, we also adopted the idea of dynamic superframe sizing as follows. We use two kinds of superframes: the *normal superframe* with its *normal beacon*, and the *mini superframe* with its *mini beacon*, both having an arbitrary, dynamically changing size. In each superframe-period we have one normal superframe and one or more mini superframes. If the normal superframe does not extend the chosen superframe-period, then we can introduce the mini superframes (Fig. 3).

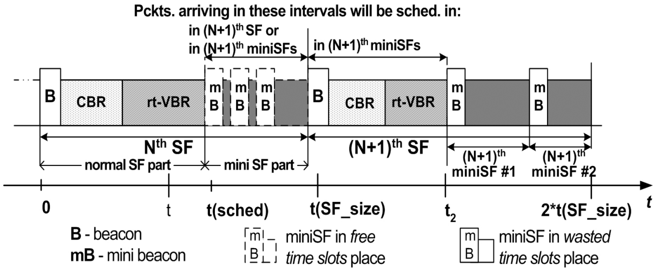

Figure 3. Hierarchical superframe structure

As for the static algorithm presented in Fig. 2, we consider two superframe-periods: the Nth with free time slots (when neither of the nodes has data to send), and the (N+1)th with wasted time slots (when, due to inaccurate information, the PNC did not schedule anyone to transmit). By introducing the hierarchical superframe concept the packets in the *(t, t_2)* interval can be scheduled in the (N+1)th superframe-period. This is possible because we get information about the respective flows before t_2; therefore, the packets can be scheduled in the (t_2, 2* *t(SF_size)*) interval. In this interval we can allocate mini superframes until we reach the end of the (N+1)th superframe-period. Thus, we obtain better channel utilization and more accurate information for the scheduling of the next superframe. During the free time slots, mini beacons can also be generated in order to give a chance for the mobile nodes to send their status information.

B. Normal and mini beacons

The introduction of the *mini beacons (mini superframe)* raises some questions regarding their influence on the 802.15.3 system. Firstly, how do the mini beacons affect the performance of the system in terms of induced complexity and traffic overhead? Secondly, what kind of additional services can be offered for nodes, and what is the trade-off between the offered services and the introduced overhead?

The proposed algorithm allows for nodes with power saving demands, strictly timed applications (e.g., CBR flows), or very low capabilities to use only the normal superframe part, while nodes with higher demands can profit from the presence of mini superframes. In the latter case, some specific control fields can be omitted from the mini beacons. These control fields can be related to piconet management and power saving, for example. The management information related to child and neighbor piconets can also be omitted, because these require pseudo-static time slots that can be guaranteed only in the normal part of the superframes. As long as the mini superframes do not contain the CAP, the fields related to CAP control can be omitted from the *Information Element* and *Piconet Synchronization Parameters* fields (see in [6]). This can reduce the complexity and the length of the mini beacons. By reducing the length, the traffic overhead introduced by the mini beacons is also decreased. The advertisement of the mini beacons can be solved by using one of the *Reserved* bits from the *Piconet Mode* field of the beacon [6].

IV. PERFORMANCE ANALYSIS

This section describes the simulation parameters (performance metrics, applied traffic, simulation scenario) and the achieved results. Firstly, the performance of our algorithm is compared with the other two superframe formation algorithms of Section II. Secondly, because the loss of a beacon message could lead to severe performance degradation, the system's tolerance to beacon loss was also investigated.

Simulation results were obtained using a discrete-event simulator, the VINT project Network Simulator *(ns2)* [7] with the 802.15.3 module presented in [9]. We implemented the gradual and the hierarchical algorithms, and enhanced the simulator in order to support our analysis demands. The simulation topology consists of three nodes (one PNC and two slave nodes) located in the same coverage area so that they can communicate directly with each other. This topology is the simplest topology for testing the performance of the algorithms. All data packets were originated by the first slave node and received by the second one. The PNC node took only care of admission control and scheduling tasks.

All simulations ran for 200s, with 40 random seeds and different superframe sizes (4, 8 and 10ms). For the gradual algorithm we used a minimum superframe size of 4 ms, a measurement period of 5 superframes, and an advertising period of 5 superframes. The rt-VBR traffic consisted of MPEG-4 video streams, generated with the mpeg4 traffic generator [8]. The main parameters used in the simulations are listed in the table below:

TABLE I. PARAMETERS FOR SIMULATION

Simulation Parameters	Value
Channel bit rate	100Mbps
Minimum GTS for rt-VBR	28μs
Mean offered load by rt-VBR	8Mbps
GTS for CBR	1076μs
CBR bandwidth	2Mbps
Packet deadline	33ms
Used superframe size	4ms, 8ms,10ms
Used superframe formation alg.	Static, Gradual, Hierarchical
Total number of devices	3 (1 PNC, 2 slaves)
Beacon loss rate applied	0%, 3%, 5%
Simulation time	200sec

In order to analyze the performance of our algorithm, we defined the following performance metrics. The *Job Failure Ratio (JFR)* is a drop ratio metric, counts all the packets dropped because of exhausted deadline. *Response Time (RT)* is the time between passing a packet from the upper layer to the MAC layer, sending it, and receiving back a MAC layer acknowledgement. *Queue Size Inconsistency (QSI)* is the summation of all the queue size differences, given by the subtraction of the queue size known by the PNC from the real queue size at the sender node.

A. Queue Size Inconsistency

Fig. 4 presents the QSI in the case of the first and last flows as a function of the network load. The labels on the graph also show the QSI as the percentage of all packets for the respective flows. The high value of this metric induces an inaccuracy in scheduling at the PNC, which leads to bandwidth wastage. The *first flow* in Fig. 4 always occupies the first part of the superframe. Under low and medium network load the static and gradual algorithms have a QSI which represents almost all the traffic of the first flow (94%). The QSI decreases gradually to 71%. This is due to the fact that in the case of high network load the flows get smaller parts from the superframe; thus, the allocated time slots are distributed among more superframes. Therefore, the chance of the node to inform the PNC increases. Meanwhile, the hierarchical algorithm starts from a low QSI value (37%), with an increasing tendency until medium network load; then, it decreases as the static algorithm.

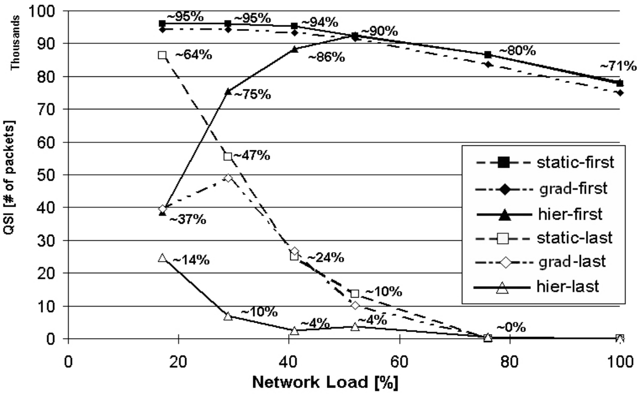

Figure 4. Queue Size Inconsistency

The increasing QSI is attributed to the fact that, as the network load becomes higher, the size of the superframe tends to become larger; thus, the distance between the time slots allocated for the first flow and the end of the superframe increases. This means that the probability for the queue size to be changed meanwhile is higher. If the network load increases further, the used superframe size tends to reach the maximum (used by the static algorithm) and the hierarchical algorithm starts to behave as the static algorithm does.

The inconsistency for the *last flows* shows different characteristics (Fig. 4). All the algorithms have a decreasing tendency as the network load increases. As the superframe tends to become overloaded in the static and the gradual cases, the time slots of the last flow get closer to the end of the superframe; therefore, QSI decreases. The QSI under low network load is smaller in the gradual case than in the static one. This is because the gradual algorithm mainly uses shorter superframe sizes (min. 4 ms); therefore, the last flow gets closer to the end of the superframe, which results in lower QSI.

If the hierarchical algorithm is analyzed, the time slots of the last flow are placed always at the end of the superframe (both normal and mini). However, we see that the hierarchical algorithm also presents a decreasing QSI under low network load. The shorter the superframe size, the higher the probability for new packets to arrive after the header of the last packet (with queue size information) is sent. Therefore, QSI decreases from 14% to 4% for networks with medium load.

B. Job Failure Ratio and Response Time

Results related to the JFR and RT are presented in Fig. 5 and Fig. 6. As the network load increases, the JFR has an increasing tendency for all algorithms. Relevant performance difference between the algorithms can be seen under medium network load. In this case the static and the gradual algorithms present higher JFR than the hierarchical one, though the JFR of the gradual algorithm is slightly lower than in the static case. For example, at 41% network load the hierarchical algorithm presents about 1% JFR, while the static has about 3% JFR. The difference between these two values means a 75% performance increasing from the JFR point of view. In the case of high network load all algorithms have the same JFR.

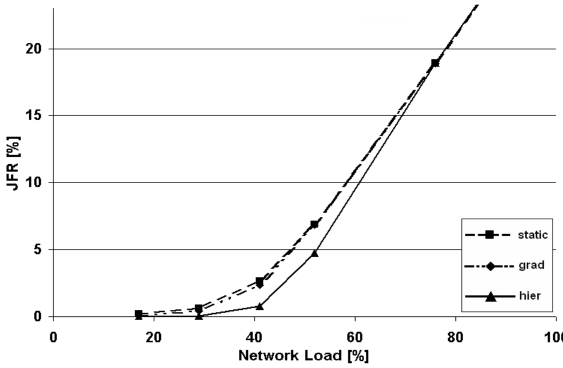

Figure 5. Job Failure Ratio

The performance difference between the algorithms is more obvious if the RT is analyzed (Fig. 6). Under low and medium network load, the hierarchical and gradual algorithms present lower RT values than the static one. This can be assigned to the shorter superframe size. The shorter the used superframe size, the lower the obtained RT.

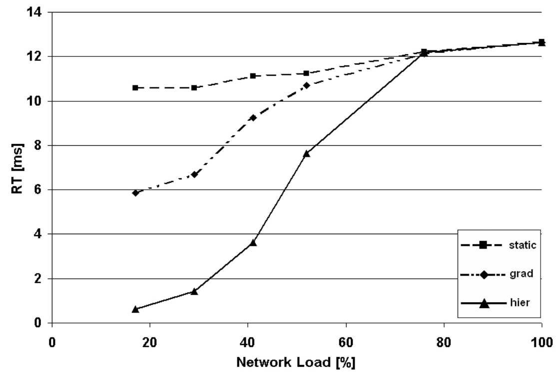

Figure 6. Response Time

C. Stability analysis

In order to have an ideal condition for scheduling, before the superframe formation phase the queue size information is passed from slave nodes to PNC through the simulator. With this technique, the information inaccuracy introduced by the QSI can be eliminated, obtaining an ideal case. Therefore, we can investigate how the performance of the currently used algorithms remains behind this ideal case. The difference between the corresponding performance metrics under normal and ideal conditions represents the JFR *alteration* value (Fig. 7). Higher *stability* is reached if the JFR alteration is lower.

In Fig. 7 the static and the gradual algorithms exhibit a different JFR alteration tendency than the hierarchical one. Taking a closer look, at medium network load for the static algorithm about 2% JFR alteration is present. This is the JFR difference between the normal (about 7%) and the ideal (about 5%) case. Therefore, this JFR alteration means a performance increase of about 25%. Meanwhile, our algorithm presents a minimal JFR alteration of 0.5%. Under ideal conditions the JFR was 4.5%, while in the normal case about 5%. The performance increase resulted with queue size passing was about 8%.

In the case of the static and the gradual algorithms, the JFR is significantly improved until medium network load is reached, because the QSI is high. This means that the static and the gradual superframes contain a high number of wasted time slots, which leads to bandwidth wastage. If network load increases further, the system becomes overwhelmed by the traffic excess and the JFR alteration drops sharply. In this case, the amount of wasted time slots is similar for both algorithms. This means that the throughput cannot be improved remarkably. Therefore, the system, even with optimal queue size information, will not be able to cope with the traffic excess, and the JFR alteration will drop to a much moderate value (0.8%).

The hierarchical algorithm has a different JFR curve than the static and the gradual algorithms. From low to medium network load there is no substantial JFR improvement, mainly because in this interval the algorithm presents only minor QSI and negligible amount of wasted time slots. If the network load increases, the increasing tendency in JFR alteration is due to the system being already overloaded; therefore, the hierarchical algorithm starts to act as the static algorithm.

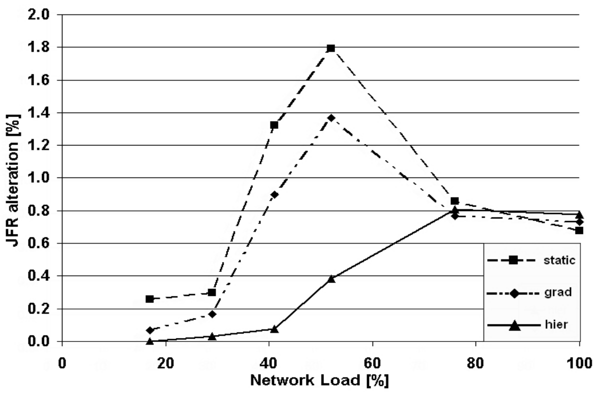

Figure 7. Algorithm stability – JFR alteration

Concluding these results, we can affirm that the performance of the hierarchical algorithm cannot be increased as much as the performance of the static and the gradual algorithms. Therefore, our algorithm has better *stability* than the static and the gradual algorithms.

D. Beacon loss

The tolerance of the algorithms to beacon loss was also analyzed (Fig. 8). For this analysis we had to implement the loss model that drops uniformly the beacons with a given probability (in the case of the hierarchical algorithm no difference is made between normal or mini beacons). We have to mention that if a beacon is dropped, then the respective superframe is unusable. The beacon loss tolerance is measured in *JFR alteration,* which represents the JFR difference between the beacon loss and the ideal (no error) case. We applied a beacon loss rate of 3% and 5% respectively. As shown in Figure 8. , the system that uses our algorithm starts to have significant JFR alteration only when medium network load is reached; meanwhile the other two algorithms, especially the static one, have almost 2% alteration, even under low network load.

As the network load increases, the static and the gradual algorithms start to have similar JFR alteration. Meanwhile, our algorithm represents less alteration even if it acts like the other two solutions, because of the previously mentioned reason (the higher the network load, the higher the used superframe size). Under high network load, using the hierarchical algorithm there is a high probability for a mini beacon to be generated after a normal superframe. These mini beacons can also be dropped. Losing the packets in the mini superframe does not modify significantly the JFR, as these are just control packets that do not contain any data, but only queue size information.

Meanwhile, if normal beacons are dropped (in the case of the gradual and the static algorithms) the JFR alteration increases. In this case, the nodes cannot send data, thus, the deadline of the packets often expires. These packets will be dropped at the nodes but still scheduled by the PNC for the next superframe, which leads to bandwidth wastage. In the case of the hierarchical algorithm, if a mini beacon is generated, the queue size information is sent to the PNC. Based on this fresh information, the PNC can schedule the packets more accurately.

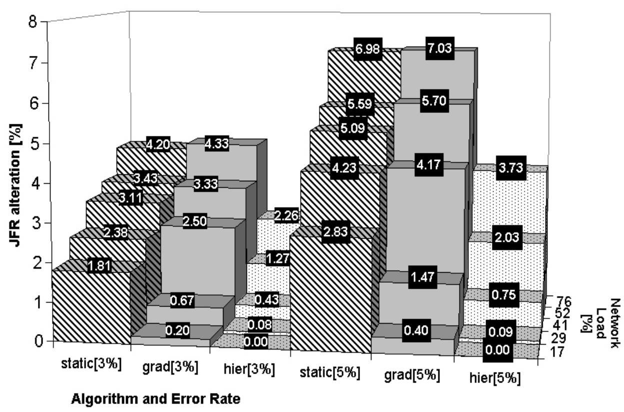

Figure 8. Beacon loss

As a conclusion, the hierarchical algorithm presents higher tolerance against beacon loss than the gradual and the static algorithms do.

E. CBR behavior

In these simulations we applied increasing rt-VBR traffic as background traffic. Besides, the measured CBR flow is applied; it has 2Mbps offered bandwidth (with a packet size of 1000 bytes and a sending rate of 4ms). Fig. 9 examines the *end-to-end delay* (E2EDelay) *variation* of the CBR packets versus the background traffic load.

The static algorithm presents a constant E2EDelay variation. The constant nature of this parameter is strongly related to the periodical behavior of the static superframe, which implies the CBR traffic synchronization to the starting time of the superframe. Meanwhile, under low background traffic load, the hierarchical and the gradual algorithms present a lower E2EDelay variation than the static algorithm does. Furthermore, we can see that by using the hierarchical algorithm lower E2EDelay variation can be reached than in both the gradual and the static cases. Although our algorithm has a periodical behavior, CBR packets can also be sent during the mini superframes.

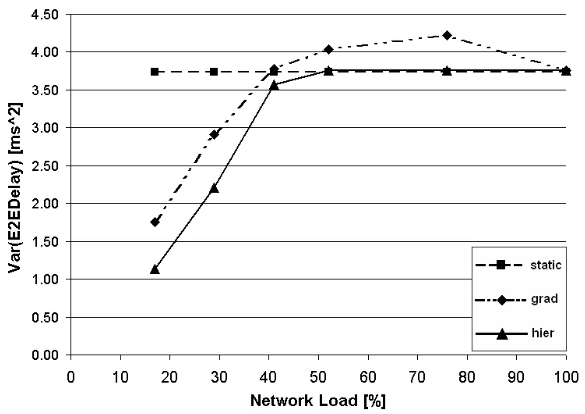

Figure 9. CBR E2E Delay Variation

As the background traffic load increases, the hierarchical and the gradual algorithms use longer superframe sizes, until they reach the size used by the static algorithm. Therefore, the E2EDelay variation under high background traffic load will be similar for both the hierarchical and the static algorithm. Meanwhile, we observe higher E2EDelay variation for the gradual algorithm. The reason is that the gradual superframe has only a pseudo-periodical nature. Even in the case of high network load, when the gradual algorithm most often uses the highest superframe size, there are cases when the superframe size is reduced for a few consecutive superframes. This causes the de-synchronization with the CBR flow; therefore, the E2EDelay of this flow has a higher variance. When maximum network load is applied, the superframe size reduction does not occur. In this case, the packet generation of the CBR flow can be synchronized with the superframe generation time. Therefore, for the gradual algorithm the E2EDelay variation of the CBR flows represents the same value as in the case of the other two algorithms. More the minimum superframe size of the gradual algorithm is reduced, further the E2EDelay variation of the CBR packets increases.

As far as the CBR behavior is concerned, based on the results we can conclude that the proposed algorithm has a proper treatment of the CBR traffic.

V. CONCLUSIONS

This paper investigates some properties of the 802.15.3 protocol, IEEE's emerging wireless standard. The performance of such a system is highly dependent on the used superframe formation algorithm. Therefore, we analyzed the advantages and drawbacks of the previously defined static and gradual algorithms.

Based on our findings, we proposed a new superframe sizing solution, called *hierarchical superframe formation algorithm*. The performance of the proposed algorithm was analyzed and compared to the other two approaches using a discrete-event simulator with rt-VBR and CBR traffic applied. During the analysis, different performance metrics were investigated. We demonstrated that our system outperforms the other algorithms in terms of channel utilization, error tolerance, and stability, and satisfies special demands such as power saving.

Future work includes the design of a new scheduler, optimized to cope with the features of the hierarchical superframe formation algorithm. The performance analysis of TCP traffic is also to be investigated.

REFERENCES

[1] C. G. Kang, C. W. Ahn, K. H. Jang, W. S. Kang, "Contention-Free Distributed Dynamic Reservation MAC Protocol with Deterministic Scheduling (C-FD3R MAC) for Wireless ATM Networks", IEEE Journal on Selected Areas in Communications, Vol. 18 Issue: 9 , Sept. 2000, pp. 1623 -1635

[2] A. C. V. Gummalla, J. O. Limb, "Wireless Medium Access Control Protocols", IEEE Communications Surveys, Second Quarter 2000

[3] N. Passas, L. Merakos, D. Skyrianoglou, F. Bauchot, G. Marmigere and S. Decrauzat, "MAC Protocol and Traffic Scheduling for Wireless ATM Networks" Mobile Networks and Applications, Vol. 3, Issue 3, 1998, pp. 275 – 292, ISSN:1383-469X

[4] R. Cusani, M. Torregiani, F. D. Priscoli, G.Ferrari, "A novel MAC and Scheduling strategy to guarantee QoS for the new-generation WIND-FLEX wireless LAN", IEEE Wireless Communications Magazine, Vol.9 No.3 , June 2002, pp. 46-56

[5] J. C. Chen, K. Sivalingam, P. Agrawal, and S. Kishore, "A Comparison of MAC Protocols for Wireless Local Networks Based on Battery Power Consumption", Proceedings of IEEE INFOCOM '98, April 1998

[6] R. F. Heile, I. C. Gifford, J. D. Allen, P. Kinney,l, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPAN)", Draft Standard for Telecommunications and Information Exchange Between Systems, Draft P802.15.3/D15, October 2002

[7] http://www.isi.edu/nsnam/ns/

[8] http://www.sce.carleton.ca/~amatrawy/mpeg4/

[9] R. Mangharam and M. Demirhan, "Performance and simulation analysis of 802.15.3 QoS", IEEE 802.15-02/297r1, Jul. 2002

[10] www.bluetooth.com