DynaMO - Dynamically tuning DSME Networks

Harrison Kurunathan, Ricardo Severino, Anis Koubaa, Eduardo Tovar
CISTER/ISEP and Prince Sultan University
Porto, portugal and Saudi Arabia
{hhkur,rarss,emt}@isep.ipp.pt,akoubaa@psu.edu.sa

ABSTRACT

Deterministic Synchronous Multichannel Extension (DSME) is a prominent MAC behavior first introduced in IEEE 802.15.4e supporting deterministic guarantees using its multisuperframe structure. DSME also facilitates techniques like multi-channel and Contention Access Period (CAP) reduction to increase the number of available guaranteed timeslots in a network. However, any tuning of these functionalities in dynamic scenarios is not explored in the standard. In this paper, we present a multisuperframe tuning technique called DynaMO which tunes the CAP reduction and Multisuperframe Order in an effective manner to improve flexibility and scalability, while guaranteeing bounded delay. We also provide simulations to prove that DynaMO with its dynamic tuning feature can offer up to 15-30% reduction in terms of latency in a large DSME network.

KEYWORDS

IEEE 802.15.4e, DSME, Multisuperframe tuning

ACM Reference Format:
Harrison Kurunathan, Ricardo Severino, Anis Koubaa, Eduardo Tovar. . DynaMO - Dynamically tuning DSME Networks. In *Proceedings of ECRTS (RTN'19)*. ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION

IEEE 802.15.4 [2] is one of the legacy protocols that supports low-rate communication with Guaranteed Time Slot (GTS) allocation mechanism that provides guaranteed bandwidth for time-critical data. However, it suffered from limited scalability as the number of GTS provided was restricted to 7. The enhancement of this protocol, the IEEE 802.15.4e [1], [10] rectifies this problem by the provision of multichannel and CAP reduction techniques. DSME is supported by a multisuperframe structure (Fig. 1) which is a stack of several superframes containing a Contention Access Period (CAP) and Contention Free Period (CFP) for communication. This multisuperframe structure is defined by a Multisuperframe Order (MO).

DSME also introduces a new technique called CAP reduction with which the number of GTS resources to accommodate transmissions can be further increased. This is achieved by removing the CAP in a multisuperframe except for the first, hence radically increasing the number of available GTSs. To invoke CAP reduction in the network, the coordinator has to send an Enhanced Beacon (EB) with a CAP reduction primitive.

Traditionally, DSME networks require a careful planning of its several MAC parameters, such as MO and CAP Reduction usage, by an experienced network engineer, to achieve adequate QoS levels. As of now, these values are determined statically at the beginning of the network. In scenarios where traffic or the number of nodes can change, which is increasingly becoming a common place in large-scale IoT networks, static settings inevitably lead to some kind of compromise in terms of delay or throughput that can only be addressed by devising mechanisms that can adapt on-the-fly to new conditions.

The main contribution of this paper are as follows:

- We introduce a dynamic Multisuperframe and CAP reduction tuning technique (DynaMO) that yields better QoS performance in terms of delay.
- We provide a numerical analysis to calculate the overall delay of the network.
- We evaluate DynaMO using the simulation platform "OpenDSME" to validate our analytical model.

In the following section we provide a brief literature survey. In section III we discuss the problem, then in Section IV, we provide the DynaMO algorithm and discuss is functionality. Later in Section V, we provide a numerical analysis for delay. We complement this analysis using simulation in Section VI. We wrap up our work with conclusions and discussions in Section VII.

2 RELATED WORKS

In our previous research [12], we observed reduced delay in DSME network when CAP reduction was utilized. But this analysis was only made for a static network. There have also been several research works like [3] and [5] in which the performance of DSME was analysed. However in these simulative studies, features like the CAP reduction and superframe structure were kept static. We believe this static configuration can be an impediment to the overall Quality of Service of the network.

In classic IEEE 802.15.4, researchers in [6] and [14] have used algorithms to adjust Superframe Order (SO) at the coordinator by considering parameters of end devices such as queue size, queuing delay, energy consumption per bit and data rate. This helped in improving the overall network life time. In one of our earlier works [8], in contrast to the traditional explicit allocation of GTS in IEEE 802.15.4, we used implicit allocation as the number of GTSs is limited. We were able to produce betterment in QoS in terms of bandwidth utilization.

The literature in varying the structure of MAC to improve QoS is not limited to DSME. Mashood Anwar [4] studied the variations in superframe of LLDN an other key MAC behavior of IEEE 802.15.4e and was able to provide an insight on the tuning of superframe to yield better network performance. Several parameters like sensors refresh rate, number of devices accommodated in network, data payload exchanged between the devices and even different levels of security were analyzed in this work.

RTN'19, July 2019, Stuttgart, Germany

We believe that dynamic tuning of the multisuperframe parameters such as MO and CAP reduction primitives has a possibility to yield better network performance. Hence we investigated several scenarios of DSME networks and propose a dynamically tunable multisuperframe scheme that yields better performance in terms of delay. In what follows, we present the scenarios of the problem that DynaMO helps to overcome.

3 BACKGROUND TO THE PROBLEM

The DSME network provides deterministic communication using its beacon enabled mode in which the entire time frame is separated into multisuperframes accommodating several superframes as shown in Figure 1. The superframe is defined by *BO*, the *Beacon Order* which is the transmission interval of a beacon in a superframe, *MO* the *Multi superframe Order* that represents the enhanced beacon interval of a multi-superframe and *SO* the *Superframe Order* that represents the beacon interval of a superframe within a Multisuperframe duration. The number of superframes in a multisuperframe can be given by $2^{(MO-SO)}$ and the number of superframes that a multisuperframe should accommodate is set by the PAN coordinator and is conveyed to the nodes via an Enhanced Beacon (EB) at the beginning of each Multisuperframe.

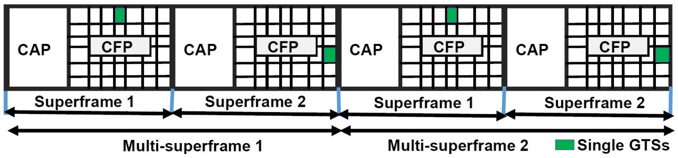


Figure 1: Superframe structure with BO=3, MO=3, SO=2

Under CAP reduction, all the superframes in a multisuperframe can be converted into complete CFPs except for the first. In accordance to the standard, both CAP reduction and MO are determined statically at the start of a multisuperframe by the Personal Area Network Coordinator (PAN-C). The network that is statically defined at the beginning will have limited capabilities to cope with constantly evolving network with joining and leaving of the nodes. Some of the adverse results can be *"an improper bandwidth allocation either due to not enough GTS slots"* or *"wasted bandwidth increasing the contribution to the delay."*

Having a routing layer such as RPL (Routing Protocol for Lossy Networks) over DSME is a fundamental mechanism to solve this problem. In our approach, an updated routing tree of the varying network topology is provided to the PAN-C by the RPL. As the number of nodes changes (via association/disassociation), RPL updates this information and the PAN-C generates a schedule spread into the available GTSs resources. A detailed report on implementing RPL over DSME can be found in [11].

In this contribution, we design an algorithm that is able to set the most adequate value of MO and toggle CAP reduction considering the needed resources. In doing so, we are able to minimize latency. The necessary changes to the values of the MO or the CAP reduction primitive are sent in the beacon payload of an EB at the beginning of every multisuperframe. Hence, with a dynamic evolution of a wireless sensor network with addition/removal of nodes new values for MO and CAP reduction primitives can be dynamically set, eventually improving the overall QoS of the network.

4 DYNAMO ALGORITHM

In this section, we introduce an efficient multisuperframe tuning algorithm called DynaMO. The general idea of this algorithm is *adaptively increasing and decreasing the multisuperframe structure based on the evolution of GTS allocation requirements over time.*

Algorithm 1 presents the DynaMO adaptive network algorithm and Table 1 presents the notation used for the description of the algorithm.

Notations	Description
N	total number of nodes
a_i	node a_i where $i\epsilon(1, N)$
$N_{Channels}$	number of channels = 16
T_i	index of the timeslot in the multisuperframe
N_{CFP}	total number of GTSs in the CFP of a multisuperframe
$NCAP$	number of GTS added when CAP reduction is activated

Table 1: Notations for DynaMO

As the network grows/diminishes dynamically, the routing layer will update the topology and forward the respective schedules that contain the list of pair-wise GTSs transmissions. This is provided as an **input** (Algorithm line 1). Let us consider pairs of neighbor nodes (a_i, a_N) to transmit between each other. This transmission list will be provided as a bitmap to the link layers using the RPL backbone for every beacon interval.

The PAN Coordinator has access to all information needed to establish a multi-channel GTS allocation, including, the number of channels ($N_{Channels}$), the number of the GTSs time slots (N_{TS}) and the total available GTS resources ($N_{CFP} = N_{Channels} * N_{TS}$). The number of time slots can sometimes vary if the CAP reduction primitive is activated. In such a case, the number of time slots will be $7 + N_{CAP}$, where N_{CAP} is the number of time slots added via CAP reduction. The PAN-C initially randomly determines the values of BO, MO, and SO and the CAP reduction primitive. Any change in the network is reported to the PAN-C or the routing parent nodes for every multisuperframe interval. The delay taken to accommodate a new network will depend on the size of the multisuperframe.

In our algorithm, we first determine the number of resources that need to be allocated in the network. This is achieved through a near optimal scheduling algorithm such as simulated annealing [15] or Symphony [11]. In fact, an optimal schedule must use the minimum number of time slots and channels so that minimal latency can be achieved. The nodes must also be placed in such a way that there is no overlapping transmissions amongst them.

5 DELAY ANALYSIS UNDER CAP REDUCTION

For our numerical analysis first we derive the value of $N_{CFP(n)}$, which is dependent on the values of the MO, BO and SO. This value is calculated to know the overall GTSs resources available under CAP reduction, then we calculate its respective delay. D_{Max}

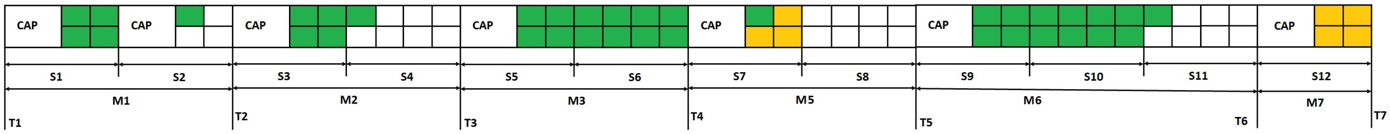
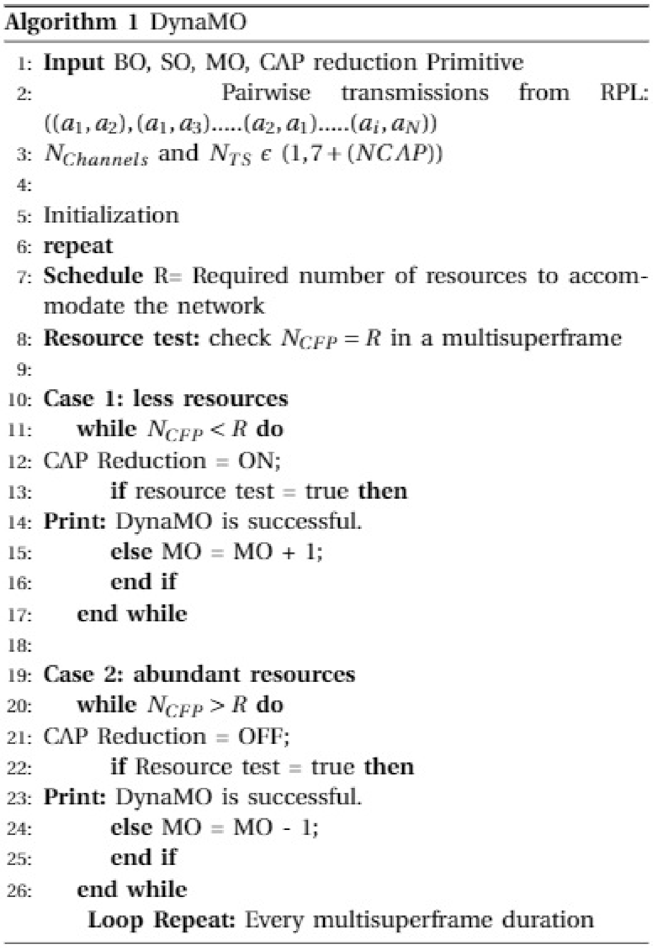


Figure 2: multisuperframes in DSME network



Algorithm 1 DynaMO

```
Input BO, SO, MO, CAP reduction Primitive
Pairwise transmissions from RPL: $((a_1,a_2),(a_1,a_3).....(a_2,a_1).....(a_i,a_N))$
$N_{Channels}$ and $N_{TS}\ \epsilon\ (1,7+(NCAP))$

Initialization
repeat
Schedule R= Required number of resources to accommodate the network
Resource test: check $N_{CFP} = R$ in a multisuperframe

Case 1: less resources
    while $N_{CFP} < R$ do
CAP Reduction = ON;
        if resource test = true then
Print: DynaMO is successful.
        else MO = MO + 1;
        end if
    end while

Case 2: abundant resources
    while $N_{CFP} > R$ do
CAP Reduction = OFF;
        if Resource test = true then
Print: DynaMO is successful.
        else MO = MO - 1;
        end if
    end while
      Loop Repeat: Every multisuperframe duration
```

represents the maximum delay a transmission has to undergo for a successful GTS allocation in a multisuperframe.

In accordance to the standard, there will be an Inter Frame Spacing (IFS) period between every successful transmission. Depending on their size if less than *aMaxSIFSFrameSize*, it is called Short Inter Frame Spacing (SIFS), else it is called Long Inter Frame Spacing (LIFS). Under LIFS, the size extends for a minimum period of *minLIFSPeriod* symbols. This IFS contributes to the delay along with other parameters such as L_{frame}, the frame length, R_s, the symbol rate and R_b the bit rate. In accordance to research work [13] done towards calculating delay in a superframe intervals, the maximum delay can be given as:

$$D_{max} = \begin{cases} D_{SIFS} = \dfrac{(L_{frame} \times R_s)}{R_b} + minSIFSPeriod, \\ D_{LIFS} = \dfrac{(L_{frame} \times R_s)}{R_b} + minLIFSPeriod \end{cases} \tag{1}$$

The duration of the multisuperframe slot will depend on the multisuperframe order (MO) issued by the PAN coordinator. This varies with respect to topology obtained through RPL. Let T_{MS} be the duration of the multisuperframe slot, N_{MD} be the total number of symbols forming the multisuperframe, N_{MD_i} be the total number of symbols forming the multisuperframe since the value of $SO = 0$,

$$T_{MS} = \frac{N_{MD}}{T_{CAP} + T_{CFP}} = N_{MD_i} \times 2^{MO-4} \tag{2}$$

Equation 2 stands true for a scenario with CAP reduction for a single multisuperframe period encompassing all the GTSs in the CFP time period. It also considers a CAP region of duration T_{CAP}.

A single GTS can span across several superframe slots, and so we should provide a constraint on it. GTS must be greater than the total forward delay D_{max}. Let us consider N_{min} to be the minimum number of superframe slots a single GTS can extend over. The total forward delay D_{max} can be given by:

$$D_{max} = T_{MS} \times N_{min} \tag{3}$$

As we consider a critical data oriented network, we neglect the delay that occurs in the CAP region of the traditional IEEE 802.15.4. Under CAP reduction the absolute number of GTSs is not certain, however it can be expressed as $m \times N_{CFP}$, where m is the number of channels and N_{CFP} is the timeslots in CFP. From these, the maximum number of GTSs that can be allocated to devices can be given by:

$$N_{CFP(n)} = min\left(\left\lceil \frac{(T_{CAP} + T_{CFP})(1 - \dfrac{T_{CAP}}{T_{MS}})}{N_{min}} \right\rceil, m \times N_{CFP}\right) \tag{4}$$

As given in Figure 2, for the need of simplicity, we consider a CFP with just 2 timeslots and 2 channels (4 available GTSs resources), this can be generalized for a larger number of channels. In this Figure we present several scenarios across the different time intervals. A delay analysis was performed for all these scenarios.

The scenarios (Figure 2) taken for the numerical analysis are listed as follows:

(i) From T1 to T2: This is a multisuperframe in which normal DSME without CAP reduction is employed. The multisuperframe in this scenario is expected to support 5 GTS transmissions. It should be

noted that without CAP reduction, the superframe has to wait for a "duration of CAP" before it is able to transmit.

(ii) From T2 to T3: This is a multisuperframe with CAP reduction employed in it. Unlike the previous discussed case, the final transmission need not wait for a CAP.

(iii) From T3 to T4: This is a multisuperframe with CAP reduction employed and the number of transmissions it has to accommodate is 13. But the MO in this scenario is static, the final transmission of this use case also has to wait for an entire CAP period before its transmission.

(iv) From T4 to T5: This is a multisuperframe with CAP reduction employed with a static MO, but it should be noted that it just needs to accommodate 3 GTSs. As a result of this 8 GTSs remain unoccupied contributing to the wasted bandwidth eventually affecting the overall throughput of the network.

(v) From T5 to T6: This holds the same condition as scenario *iii*, but with DynaMO, PAN-C counts the number of transmissions to be accommodated by the CFP. As value is above the number of timeslots available, it increases the MO by 1 adding a superframe to the multisuperframe. In this use case, the MO is 2, thus joining 3 superframes within a multisuperframe, eventually reducing the overall delay.

(vi) From T6 to T7: In this case the number of GTSs to be accommodated is 4. PAN-C deploys CAP reduction in this scenario eventually providing a single superframe to accommodate the 4 transmissions. This method will reduce the wastage of bandwidth thus increasing the throughput.

We calculated the delay of the network for all the use cases as mentioned above using Equation 2. We considered a network that dynamically grows and thus demanding more GTSs resources. For CAP reduction scenarios, we take the value of MO to be 1. For this numerical analysis we consider idle time to be 0 and a constant bit rate of 1kbps.

From Figure 3, it can be noted that under traditional DSME, the transmission delay of the GTS frames starts to increase at a point where the multisuperframe cannot allocate more GTSs. As the MO is constant, delay inevitably starts to increase when enough resources are not available, imposing a transmission deference to the next superframe. However, if CAP reduction is triggered, delay is much smaller when compared to the normal DSME, as more GTSs resources are available. With DynaMO, the MO is increased when more resources are needed, hence, it provides better results than networks with solely CAP reduction enabled (by 15%) and DSME networks with constant, non-dynamic settings (by 35%).

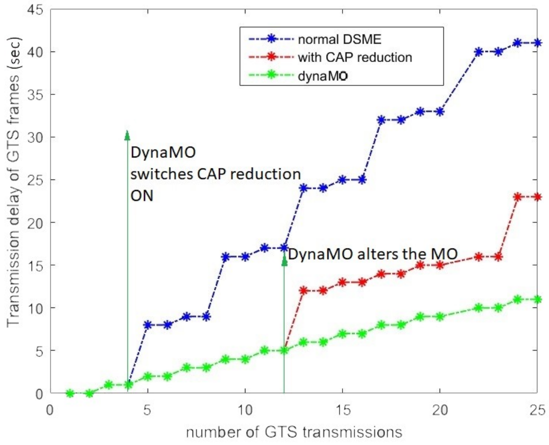


Figure 3: Comparison in terms of delay

Application type	BO	SO	MO	CAP reduction
Delay sensitive	6	0	1	Enabled
Reliability sensitive	8	3		Disabled
Energy Critical	14	1	14	Enabled
High throughput	10	5	6	Disabled
Large scale	10	1	8	Enabled

Table 2: Application scenarios for BO,MO,SO variation

6 SIMULATION ANALYSIS

For evaluating DynaMO, we use the OpenDSME simulation platform [7]. OpenDSME is a OMNET++/C++ simulation based environment that is dedicated for the simulation of the IEEE 802.15.4e DSME protocol. OpenDSME also provides the possibility of implementing a viable network layer on top of it. The DSME sublayer of OpenDSME employs a typical slot based reservation system for a schedule that is provided by the top layer.

In our model, we provide BO, MO, SO and the CAP reduction primitives as a direct input. Other network simulation parameters such as traffic rate, the burst size, the interference and the mobility models are also be given directly. Furthermore, there is also a possibility to input the schedule in accordance with a static schedule. We have also incorporated delay and throughput parameters [9] in the network definition files to obtain the appropriate output for the network simulated. The simulations were carried out on a mesh network and the overall network delay was observed.

IEEE 802.15.4e standard provides certain suggestive values for BO, SO and MO for application specific scenarios (Table II). These values when kept static provide us a multisuperframe format with a specific number of superframes. For the delay sensitive settings BO, SO and MO is 6,0,1, hence number of superframes within a multisuperframe will be 2. In such a case, a transmission need not wait for a long time for the eventual transmission. However, when kept static, it may result in increased latency.

In Table 3, we provide the parameters that we have used for all the scenarios we put under extensive simulations.

6.1 Comparison against static CAP reduction

For this comparison we calculate the values of the overall delay of the network with respect to the number of GTSs transmissions. For this simulation we analyze the delay of 50 nodes under different traffic rates ranging from 5-75 Kbps for CAP reduction and without CAP reduction scenarios in Fig 4. This result complements our theoretical analysis shown in Figure 2, clearly showing DynaMO in action.

With a limited number of GTSs transmissions, the delay performance does not have a significant decrease with the scenarios without CAP reduction (5,10,15 transmissions). Delay performance

Parameters	6.1 against CAP reduction	against different traffic rates	6.2 against high throughput settings
Packet Length	75B	75, 100B	75, 100B
Packet Traffic Interval	50, 30, 15ms	50, 30, 15ms	50, 30, 15ms
Destination	sink	sink	sink
MAC Queue Length	30	30	30
MAC Frame Retries	7	7	7
BO	6	10	6, 10
SO	3	5	3, 5
MO	DynaMO	6, DynaMO	4, 6, DynaMO
Number of Nodes	5 to 50	5 to 50	5 to 50
Traffic Rate	15, 25, 75k Kps	15, 25, 50, 75 Kbps	25, 50, 75, 100 Kbps
CAP Reduction	DynaMO	OFF	ON/OFF/DynaMO

Table 3: Simulation Parameters

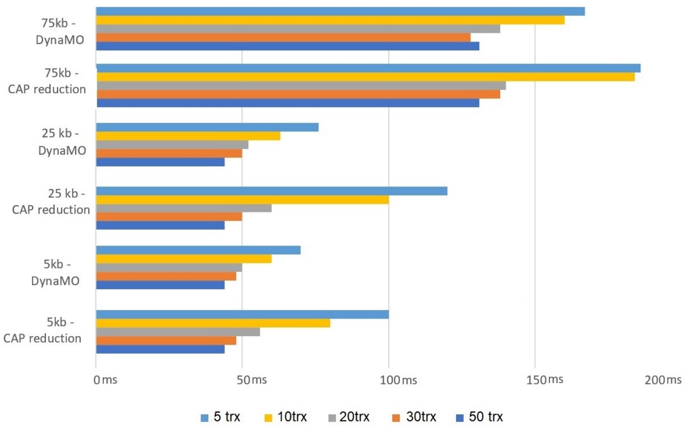


Figure 4: Delay analysis against static CAP reduction settings

is in-fact sometimes better without CAP reduction when the number of nodes is less than 10, due to less wasted bandwidth. However, as the number of transmissions increases, with CAP reduction, delay is minimized. This is due to the fact that nodes need not wait till another superframe duration to accommodate the transmissions that did not occur during the initial superframe interval.

DynaMO switches the CAP reduction parameters according to the resource requirements and hence doesn't compromise on the delay for those scenarios in which CAP reduction is still not needed, offering a clear advantage over static settings.

For clear understanding, the example of DynaMO is demonstrated along with the 75Kbps and the 5Kbps case in Figure 5. The dotted lines represent the scenario with static CAP reduction. Initially, the CAP reduction is OFF providing minimal delay (similar to the scenario without CAP reduction), whereas at T0, due to the scarcity of the resources, the CAP reduction is turned ON dynamically and we can witness a reduction in delay by almost 30%. Above 20 scheduled transmissions, an increase in MO under DynaMO further maintains a lower delay in comparison to static settings including the CAP reduction enabled setting.

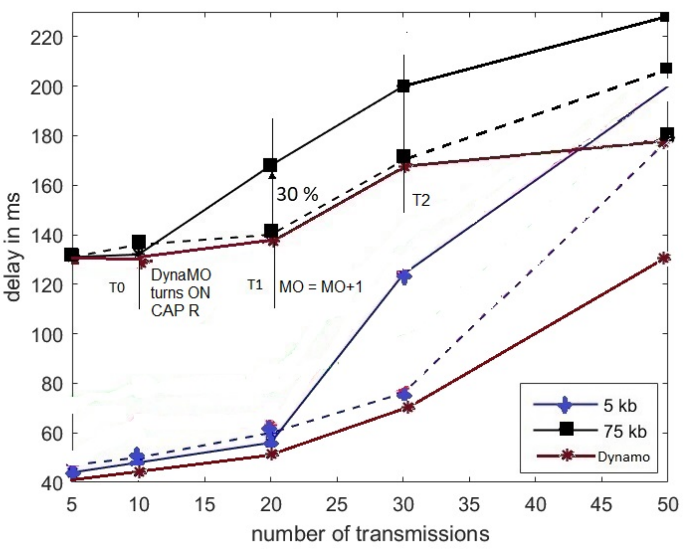


Figure 5: Delay analysis for 75 and 5 Kbps traffic rate

6.2 Comparison against Delay sensitive and high throughput settings

In this experiment, we compare the static high throughput settings and the static delay sensitive settings (dotted lines) with DynaMO. In Figure 6, we demonstrate this comparison over 100Kbps. The other traffic rates also have a similar behavior. OpenDSME does not allow the value of SO to be set to '0' by default. So we took another delay sensitive setting of BO, SO and MO to be 6,3,4 such that the number of superframes within a multisuperframe will be 2 and every beacon interval will have 4 multisuperframes.

The delay is always higher in the high throughput setting, and this gap increases with traffic rate. The higher MO in the high throughput settings causes a wastage of bandwidth which results in additional delay, contrary to the time-sensitive settings in which the superframes are closely packed. We observe almost 20-25% reduction of delay under delay sensitive settings when the number of transmissions is maximized. However, relying on static settings which provide shorter MO is often not an adequate solution, as it

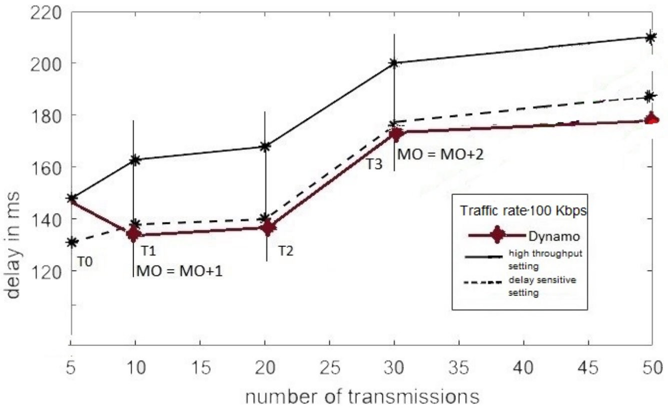


Figure 6: Delay Analysis against delay-sensitive settings

can compromise the QoS if the network needs to accommodate an increase in traffic.

In Figure 6, at T0, we start DynaMO with a high throughput setting, consisting of one superframe in a multisuperframe. However, as the timeframe moves on to T1 and the number of transmissions increases, DynaMO automatically adapts its MO based on the number of resources. In this case, by increasing MO, DynaMO packs more superframes within the beacon interval, providing more GTS bandwidth and eventually obtaining lesser delay.

We can observe a significant reduction in delay, even when compred against the static delay-sensitive settings. Notice, that the delay-sensitive setting does not outperform DynaMO in terms of delay when the the number of transmissions is lesser. Although this could somewhat appear counter-intuitive, as the number of transmissions increases, the short MO is not able to accommodate the transmissions causing a deference of transmissions to the subsequent superframes. This increases delay and its effect is particularly visible above 35 scheduled transmissions. With DynaMO employed, we are able to witness 15-30% reduction in delay when compared to the standard presets.

7 FUTURE WORK

In this paper we introduced an efficient multisuperframe tuning technique that can switch CAP reduction and tune the MO on demand, on a dynamic DSME network. From our simulations and numerical analysis, we learn that static settings are an impediment when it comes to large scale DSME network. With our tuning technique, we were able to obtain 15-35% of reduction in the overall delay of the network.

The network analysis in this paper was focused on delay over a mesh network. DynaMO also impacts other QoS parameters such as throughput and bandwidth utilization, and these will be objective of further work, while applying our technique into other different topologies and scenarios. We hope this algorithm will be part of a package aiming at dynamically improving the QoS of DSME, as we believe this is necessary for this protocol to achieve its full potential.

Though DSME has all the factors to become a de-facto protocol for critical IoT, not much research work has been done on implementing it in real platforms, nor over real time operating systems. We intend to implement DynaMO and DSME over a Commercial off The Shelf Technologies (CoTS) to better assess its capabilities over real hardware.

ACKNOWLEDGEMENTS

This work was partially supported by National Funds through FCT/MEC (Portuguese Foundation for Science and Technology) and co-financed by ERDF (European Regional Development Fund) under the PT2020 Partnership, within the CISTER Research Unit (CEC/04234); also by FCT/MEC and the EU ECSEL JU under the H2020 Framework Programme, within project ECSEL/0002/2015, JU grant nr. 692529-2 (SAFECOP). This work is also support by the Research and Translation Center (RTC) at Prince Sultan University.

REFERENCES

[1] 2011. IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs). *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)* (Sept 2011), 1–314. https://doi.org/10.1109/IEEESTD.2011.6012487

[2] 2016. IEEE Standard for Low-Rate Wireless Networks. *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)* (April 2016), 1–709. https://doi.org/10.1109/IEEESTD.2016.7460875

[3] Giuliana Alderisi, Gaetano Patti, Orazio Mirabella, and Lucia Lo Bello. 2015. Simulative assessments of the ieee 802.15. 4e dsme and tsch in realistic process automation scenarios. In *Industrial Informatics (INDIN), 2015 IEEE 13th International Conference on.* IEEE, 948–955.

[4] M. Anwar and Y. Xia. 2014. IEEE 802.15.4e LLDN: Superframe configuration for networked control systems. In *Proceedings of the 33rd Chinese Control Conference.* 5568–5573. https://doi.org/10.1109/ChiCC.2014.6895891

[5] S. Capone, R. Brama, F. Ricciato, G. Boggia, and A. Malvasi. 2014. Modeling and simulation of energy efficient enhancements for IEEE 802.15.4e DSME. In *2014 Wireless Telecommunications Symposium.* 1–6. https://doi.org/10.1109/WTS.2014.6835017

[6] Joseph Jeon, Jong Wook Lee, Jae Yeol Ha, and Wook Hyun Kwon. 2007. DCA: Duty-cycle adaptation algorithm for IEEE 802.15. 4 beacon-enabled networks. In *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th.* IEEE, 110–113.

[7] Florian Kauer, Maximilian Köstler, Tobias Lübkert, and Volker Turau. 2017. OpenDSME-A portable framework for reliable wireless sensor and actuator networks. In *Networked Systems (NetSys), 2017 International Conference on.* IEEE, 1–2.

[8] Anis Koubaa, Mário Alves, and Eduardo Tovar. [n. d.]. i-GAME: an implicit GTS allocation mechanism in IEEE 802.15. 4 for time-sensitive wireless sensor networks. In *Real-Time Systems, 2006. 18th Euromicro Conference on.* IEEE, 10–pp.

[9] Harrison Kurunathan. 2019. OpenDSME support file. https://github.com/harrisonkurunathan/throughputnedfile. (2019).

[10] Harrison Kurunathan, Ricardo Severino, Anis Koubaa, and Eduardo Tovar. 2018. IEEE 802.15. 4e in a Nutshell: Survey and Performance Evaluation. *IEEE Communications Surveys & Tutorials* (2018).

[11] Harrison Kurunathan, Ricardo Severino, Anis Koubâa, and Eduardo Tovar. 2018. *RPL over DSME: A Technical Report.* Technical Report. CISTER-Research Centre in Realtime and Embedded Computing Systems.

[12] Harrison Kurunathan, Ricardo Severino, Anis Koubâa, Eduardo Tovar, et al. 2017. Worst-Case Bound Analysis for the Time-Critical MAC behaviors of IEEE 802.15. 4e. In *13th IEEE International Workshop on Factory Communication Systems Communication in Automation (WFCS 2017). 31, May to 2, Jun, 2017.*

[13] Pangun Park, Carlo Fischione, and Karl Henrik Johansson. 2009. Performance analysis of GTS allocation in beacon enabled IEEE 802.15. 4. In *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks.* IEEE, 1–9.

[14] Ricardo Severino, Nuno Pereira, and Eduardo Tovar. 2014. Dynamic cluster scheduling for cluster-tree WSNs. *SpringerPlus* 3, 1 (2014), 493.

[15] Peter JM Van Laarhoven and Emile HL Aarts. 1987. Simulated annealing. In *Simulated annealing: Theory and applications.* Springer, 7–15.