

Improving Network Formation in IEEE 802.15.4e DSME

Carlo Vallati, Simone Brienza, Maurizio Palmieri, Giuseppe Anastasi

*Dip. Ingegneria dell'Informazione, University of Pisa, L.go L.Lazzarino 1, I-56122 Pisa,
Italy*

Abstract

Wireless Sensor and Actuator Networks are becoming attractive also for industrial applications, since recent standardization efforts have introduced significant improvement to reliability and deterministic communication delays. In this context, IEEE 802.15.4e is widely considered the major improvement, introducing many enhancements to the original IEEE 802.15.4 standard aimed at supporting critical applications. Among the new defined MAC protocols, *Deterministic and Synchronous Multi-channel Extension* (DSME) represents the most suitable option for applications with time-varying requirements. In this paper, an analysis of the IEEE 802.15.4 DSME MAC protocol during network formation is presented. The goal is to study the protocol performance and propose solutions to reduce the network formation time, improving energy and resource efficiency. To carry out the performance evaluation, DSME has been fully implemented in Contiki OS, an actual operating system for sensor nodes. The study has highlighted issues and inefficiencies in the network formation process, allowing to consequently propose effective solutions. In particular, it is proposed a set of guidelines for DSME configuration to the original MAC protocol that are proved to increase significantly the network formation efficiency.

Keywords: Wireless Sensor Networks, IEEE 802.15.4e, Deterministic and Synchronous Multi-channel Extension (DSME), Network Formation

1. Introduction

Wireless Sensor and Actuators Networks (WSANs) will be a major enabling technology for the future Internet of Things (IoT) [1], as they allow rapid and low-cost deployment of devices for remote control and monitoring. However, a large number of future applications will be characterized by requirements that go beyond the simple best-effort data forwarding [2]. Applications for smart healthcare, home and building automation and smart cities will demand for transmission reliability and timeliness [3].

Recent standardization efforts within IEEE and IETF have focused on the definition of solutions to guarantee reliable and timed information delivery in WSANs, still providing flexibility and rapid installation. In this context, the IEEE 802.15.4e standard [4] has been published in 2012 as revision of the original IEEE 802.15.4 [5] to introduce enhancements that can guarantee deterministic latency and reliability in message delivery. In particular, in replacement to the 802.15.4 CSMA-CA MAC protocol – which is unsuitable for applications with strict timing requirements [6][7] – several new MAC behavior modes are defined [8]. Among them, the *Deterministic and Synchronous Multi-Channel Extension* (DSME) has been conceived to guarantee *deterministic delay* and *high reliability*, still preserving the main advantages of CSMA-CA, i.e. *flexibility* and *adaptability* to time-varying traffic and operating conditions. Deterministic communication latency, *resilience to external interference* and rapid adaptability to traffic conditions make DSME the best choice in a large number of scenarios, characterized by delay-sensitive traffic with varying load. DSME exploits *time-slotted* channel access to guarantee timely data delivery, and adopts *multi-channel communication* and frequency hopping to mitigate the effects of external interferences. DSME alternates *Contention Access Periods* (CAPs) and *Guaranteed Time Slots* (GTSs), to enable both best effort communication and guaranteed transmission opportunities, respectively. GTSs are allocated/deallocated dynamically on-demand by nodes in a *point-to-point* fashion whenever it is necessary, through a *distributed coordination procedure*. This two-

fold MAC structure makes DSME a very versatile protocol, particularly suitable for those applications in which traffic changes over time. DSME can react to new conditions allowing nodes to reserve additional GTSs during peak of traffic or free unused GTSs.

35 In this paper, an analysis of the IEEE 802.15.4e DSME performance focusing on the initial phase of the network formation is presented. The goal is to assess DSME performance and derive a set of guidelines on how parameters should be set to avoid long network startup times. In particular, the analysis focuses on how to maximize the efficiency of CAP operations to guarantee low DSME-
40 GTS allocation delays and minimize, at the same time, the CAP size to reduce energy consumption and waste of unused resources. It is important to highlight that, although optimizing CAP operations has the major impact on the network initialization phase, it can improve the performance for the whole network lifetime, since GTS allocation/deallocation procedures are executed also after
45 network setup, as traffic demand varies over time.

Our results highlight that, besides tuning CSMA-CA parameters, a minor modification to the behavior of nodes can improve CAP efficiency. To this aim, an improvement to the original MAC protocol called *Active Backoff* is proposed to further reduce DSME-GTS allocation delays, without additional energy cost.
50 Performance evaluation shows that proper tuning of CSMA-CA parameters with Active Backoff can dramatically reduce DSME-GTS allocation delays by 60%. The analysis is performed exploiting a real DSME implementation realized for the *Contiki* operating system, an operating system for sensor motes, and run on *Cooja*, a WSN emulator for the Contiki OS. To the best of our knowledge,
55 this is the first work exploiting a real DSME implementation running on an actual operating system, rather than a simplified model for simulation [9][10], or mathematical analysis [11].

The rest of the paper is structured as follow. Section 2 provides a description of the 802.15.4e DSME MAC behavior mode. Section 3 analyzes the state of the art. Section 4 overviews the reference scenario and its settings. In Section 5
60 the effects of CAP size are assessed with respect to the opportunity of enabling

the CAP Reduction strategy. In Section 6 the DSME-GTS allocation procedure is analyzed, while in Section 7 a tuning strategy along with the Active Backoff mechanism are proposed to increase its efficiency. In Section 8 the proposed solutions are evaluated considering a lossy channel. Finally, in Section 9 conclusions are drawn.

2. Deterministic and Synchronous Multi-channel Extension (DSME)

In a DSME network, some nodes – referred as coordinators – periodically transmit an *Enhanced Beacon (EB)* frame, used to keep all the nodes synchronized and allow new nodes to join the network. The time between two subsequent EBs sent by the same coordinator is called *Beacon Interval (BI)*. The latter is composed of several *superframes* that follow one another seamlessly. Within a Beacon Interval, it is possible to define cycles of repeated superframes, called *multi-superframes*, as shown in Fig. 1. Each superframe is divided in 16 equally spaced slots and consists of three parts, namely an *Enhanced Beacon slot*, a *Contention Access Period (CAP)* and a *Collision Free Period (CFP)*. Slot 0 is used to transmit EBs. The CAP, which starts after the EB slot and ends before slot 9, is typically used to transmit control messages and aperiodic data. During the CAP, nodes use the *slotted CSMA-CA (Carrier Sense Multiple Access with Collision Avoidance)* algorithm for medium access (see Sec. II.A). Both the EB and all the frames sent during the CAP are transmitted using a single channel. During the CAP, a transmission can be received from any other node in its transmission range. Hence, a node must remain active for all its duration. The remaining slots compose the CFP. Here each slot represents a *DSME Guaranteed Time Slot (DSME-GTS)* and it is exclusively dedicated to communication between two nodes. Since DSME-GTSs are scheduled to be collision free, they are used to transmit periodic traffic and delay sensitive data. Multiple transmissions can be accommodated during the same DSME-GTS by using different channels, thus significantly increasing the network capacity. Since during the CAP all nodes are in reception mode, DSME

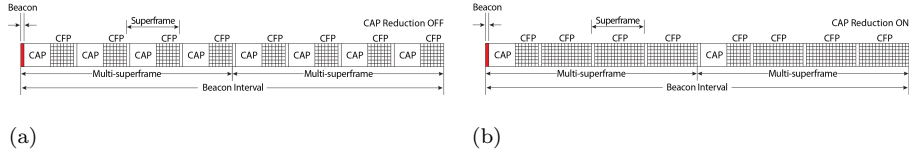


Figure 1: DSME multi-superframe structure with CAP Reduction disabled (a) and enabled (b)

provides a *CAP Reduction* mechanism to save energy. By enabling this option, only the first CAP inside the multi-superframe is maintained, while the others are replaced with wider CFPs of 15 DSME-GTSs, as shown in Fig. 1b.

The multi-superframe structure described above is regulated by some parameters – namely *macSuperframeOrder* (SO), *macMulti-superframeOrder* (MO) and *macBeaconOrder* (BO) – with $0 \leq SO \leq MO \leq BO \leq 14$ – that determine superframe duration, multi-superframe duration and beacon interval, respectively, as follows:

$$T_{SD} = aBaseSuperframeDuration \times 2^{SO} \text{ symbols}$$

$$T_{MD} = aBaseSuperframeDuration \times 2^{MO} \text{ symbols}$$

$$T_{BI} = aBaseSuperframeDuration \times 2^{BO} \text{ symbols}$$

where *aBaseSuperframeDuration* is a constant equal to 960. Such values are expressed in symbols, i.e., the minimal transmission unit defined in the IEEE 805.15.4 standard. The values of SO, MO and BO are set according to the application requirements and are included in all the EBs.

2.1. CSMA-CA Algorithm

In the slotted CSMA-CA algorithm, used during the CAP, all nodes use a single channel, time is divided into slots of equal duration (*backoff slots*) and all the operations are aligned with them. Upon receiving a data packet to transmit, each node executes a *backoff stage*, i.e., it waits for a *random* number of backoff slots (*backoff time*) and, then, performs two consecutive *Clear Channel Assessments* (CCAs) to check the channel state. Specifically, the number of

Table 1: IEEE 802.15.4E CSMA-CA PARAMETERS AND VALUES [5]

Parameter	Min-Max values	Default value
macMinBE	0-7	3
macMaxBE	3-8	5
macMaxCSMABackoffs	0-5	4
macMaxFrameRetries	0-7	3

110 backoff slots to wait is randomly chosen in the interval $[0; 2^{BE} - 1]$, where BE represents the *backoff exponent*. It is initialized to *macMinBE* and incremented each time the channel is found busy during the CCAs, until the maximum value *macMaxBE* is reached. If the channel is found idle on both CCAs, the node transmits the packet; otherwise, it performs a new backoff stage. If the number of stages exceeds the maximum value *macMaxCSMABackoffs*, the packet is dropped. After transmission, the sender waits for an Acknowledgment. If it is not received within a timeout, a retransmission is issued, until the maximum number of retransmissions, specified by *macMaxFrameRetries*, is reached. When this occurs the packet is dropped. The CSMA/CA parameters are summarized in Table 1 with their default and min-max values. With this respect, in a previous work Anastasi et al. [12] shown that CSMA/CA performance is significantly influenced by them, i.e. delivery ratio, latency and energy consumption increase with the values of these parameters.

2.2. DSME-GTS management

125 The DSME-GTS functionality allows a pair of neighboring nodes to reserve a portion of the CFP in the superframe to issue communications, being sure that no other node of the network will interfere. In case of a multi-hop flow, DSME-GTSs should be allocated sequentially hop-by-hop. Each DSME-GTS can be deallocated at any time by one of the two communicating endpoints, when (i) it is no longer used for packet transmission; or (ii) no data are received in several consecutive multi-superframes. In order to manage DSME-GTSs, each node stores two data structures, namely *DSME Allocation Counter Table (ACT)*

and *DSME Slot Allocation Bitmap (SAB)*. The former is a table containing information on the DSME-GTSs assigned to the node. The latter, instead, is
135 a bitmap indicating, for all the possible pairs (*channel, timeslot*) which are allocated to the node or to its neighbors. The scheduling of DSME-GTSs is performed in a fully distributed way. Specifically, the allocation of a DSME-GTS consists in a three-step message exchange, performed during the CAP. As example, consider the situation illustrated in Fig. 2, where node A needs to
140 allocate a DSME-GTS to communicate with node B. The following steps are executed:

1. REQUEST. Initially, node A transmits a *DSME-GTS Request* command frame to node B. Node A specifies the number of requested DSME-GTSs and the preferred slots and channels for allocation. In addition, a subset
145 of the Slot Allocation Bitmap is sent to inform node B about the slots that can be used. After transmission, node A waits for a reply command frame from node B. If the reply frame is not received within a predefined time interval, the allocation is assumed to be failed.
2. REPLY. Upon receiving a request, node B selects the GTSs to allocate,
150 considering the preferences expressed by node A. If the preferred slots are not available, the next ones are considered. Specifically, node B compares the received bitmap with its own, in order to find a number of common free slots equal to the requested number of DSME-GTSs. If they are found, node B updates its two data structures. Then, it creates a *Reply*
155 command frame including the indication of the newly allocated slots. The Reply frame is broadcast and its payload contains the address of node A.
3. NOTIFY. When node A receives the *Reply* command frame, it adds the newly allocated DSME-GTSs to its data structures. Then it broadcasts a *Notify*
160 command frame, containing the address of node B and indicating the slots that have been allocated.

Reply or *Notify* frames are broadcast to allow neighboring nodes to check potential conflicts. To this aim, a node not involved in the allocation (e.g., C and

D in Fig. 2) must check if the newly allocated slots – specified in the received command frame – conflict with slots allocated to itself. If a conflict is found, the node sends a *Duplicated Allocation Notification* frame to the source of the received command to inform that the performed allocation is not valid and must be canceled. If no conflicts are detected, the node updates its bitmap to take note of the allocated DSME-GTSs.

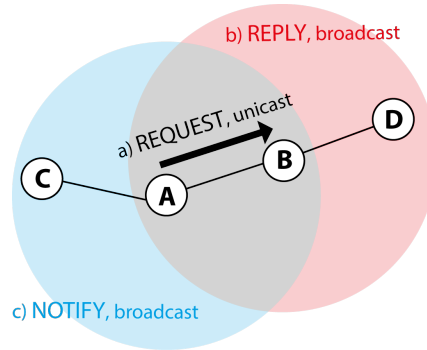


Figure 2: Example of a handshake for DSME-GTS allocation

3. Related work

The majority of the related work on DSME focuses on analyzing its performance [25] [26] or proposing modifications to improve its efficiency (e.g. energy efficiency) at the steady state [27]. Only a few works study its performance during network formation. Specifically, they analyze the initial association phase in which nodes discover the DSME network through the reception of a beacon message and perform the association handshake. In this work, instead, we focus on the allocation of DSME-GTS, which is performed after the association. Liu et al. [21] study the association procedure defined in the IEEE 802.15.4e standard, showing that it might require many beacon intervals to complete. In order to overcome the issue they propose an Enhanced Fast Association mechanism to reduce the contention among nodes due to the association handshake. Lee et al. [23], instead, study the distributed beaconing scheduling mechanism

that regulates the emission of beacon messages. They show that beacon collisions might occur frequently, thus causing long waiting time for the network construction. To overcome this problem, the authors present a new beacon scheduling algorithm, the Distributed Fast Beacon Algorithm. The problem of
185 beacon scheduling is addressed also by Hwang et al. [22]. In order to introduce a set of improvements to avoid beacon collision, the authors propose a modified version of the protocol, named DSME-E. A modified version of the DSME protocol is also proposed by Sahoo et al. [24]. Specifically they present a set
190 of modifications to DSME that aims at reducing the association time and the overall energy consumption, i.e. a new contention channel access scheme and a new beacon scheduling algorithm.

The only work that studies the GTS allocation procedure in beacon-enabled IEEE 802.15.4 networks is Abdeddaim et al. [20]. The work proposes a method-
195 ology to adapt the CSMA-CA parameters dynamically. The approach, however, is specifically tailored for single-hop topologies, e.g. star networks. In this paper, instead, we consider multi-hop topologies.

4. Experimental Scenario

In order to assess the DSME performance in a realistic environment, a DSME
200 implementation for the *Contiki OS* has been realized.

The performance evaluation is carried out exploiting *Cooja*, an emulator for WSANs. Using an emulator instead of real hardware is a common practice when evaluating a network protocol for WSANs (e.g., Ancillotti et al. [13]). *Cooja*, in fact, allows to evaluate DSME in a realistic environment, that can be
205 easily controlled and replicated. In particular, it can reproduce the behavior of real hardware, running the same binaries compiled for real devices, while wireless communications are simulated through well-known channel models. This methodology has been adopted in order to run an exhaustive range of experiments covering a wide range of scenarios and settings. Experiments based on
210 real devices are left for future work.

Table 2: Emulation Parameters

Parameter	Value
Bit Rate	250 Kbps
Symbol duration	4 bits
Data Packet (Payload) Size	140 (127) bytes
ACK Size	11 bytes
Beacon Order (BO)	9
Multisuperframe Order (MO)	9
Superframe Order (SO)	5
Power consumption in RX state P_{RX}	56.4 mW
Power consumption in TX state P_{TX}	52.2 mW
Power consumption in Idle state P_{idle}	1.28 mW

The considered network topology is a regular grid, as shown in Fig. 3, with a number of nodes that varies from 4 to 49. In the presented experiments, a mesh communication scenario is considered. At the beginning of the experiment, all the nodes are activated simultaneously, and each node generates a data flow towards a random receiver, chosen among all the other nodes in the network. Such scenario is characterized by a high traffic demand. It has been considered in our experiments to study the performance of the system in a worst-case scenario. At every multi-superframe, each node sends a data packet, with the maximum payload size allowed by the standard (127 byte), addressed to the chosen destination. In order to enable multi-hop routing of traffic, a static routing based on shortest path evaluation is configured. By using a static routing, nodes have immediate knowledge on the network topology and can immediately derive the next hop for each data packet. Even though the contribution of the routing protocol in the network formation process is well-known to be significant, in this work a static routing is considered to focus on evaluating the dynamics of DSME, hence avoiding biases introduced by the routing protocol. The assessment of routing protocols, such as RPL, on top of DSME is left for future work.

DSME-GTSs for end-to-end traffic delivery of each flow are activated on demand node-by-node. When a data packet is received from a neighbor, or generated locally, the next hop is determined according to the static routing table. If a DSME-GTS towards the next hop exists, the packet is transmitted in the next opportunity; otherwise, the allocation procedure is triggered first. Considering that DSME-GTSs are allocated per next hop, multiple flows passing through the same nodes can share the same DSME-GTS for data forwarding.

The allocation of DSME-GTS for data transmission is driven by the scheduling policy, which is not specified in the IEEE 802.15.4e standard. Considering that the definition of a scheduling policy is out of the scope of this work, we adopted a static allocation that guarantees the transmission of data avoiding delays due to data buffering. To this aim, the Superframe Order (SO) has been set to a value that allows the transmission of more than one packet in a single DSME-GTS, specifically four packets of maximum size, as four flows passing through the same node is the worst case in our scenario. This configuration eliminates potential delays due to buffered data, allowing the evaluation of the pure network setup time.

The Multisuperframe Order (MO), instead, has been chosen to allocate a proper number of DSME-GTSs (about 100) with respect to a medium size network, as the one studied in this analysis. Based on the chosen values of SO and MO, a multi-superframe consists of 16 superframes. CSMA-CA parameters are set following the default values defined in the standard (see Table 1). Power consumption refers to the Chipcon CC2420 radio transceiver, commonly used in sensor nodes. They have been derived from the CC2420 datasheet[14], considering a voltage equal to 3 V and a transmission power of 0 dBm. The transmission range for all nodes is set to 25m, unless specified otherwise. Specifically, nodes are deployed in the network so that they can communicate only towards their one-hop neighbors, as shown in Fig. 3. Initially, an ideal channel and, thus, the disk model is considered, i.e. transmissions inside the disk are not subjected to loss due to the channel. This allows to better assess the impact of CAP reduction on the performance. Then, to make the analysis more real-

istic, experiments with the Gilbert-Elliot channel model considering different
 260 values of Packet Error Rate (PER) have been conducted and are presented in
 Section 8. The settings of the Contiki OS and the Cooja emulator are summa-
 rized in Table 2, which also shows the values of the multisuperframe parameters.
 CSMA-CA parameters are set to the default values (see Table 1).

For each experiment, ten independent replications have been run, according
 265 to the method of independent replications. The following metrics are estimated
 along with 95% confidence intervals.

1. *Network setup time*, defined as the time interval between the beginning
 of the experiment and the completion of the last DSME-GTS allocation.
 Expressed in number of multi-superframes, this metric measures the time
 270 required to allocate all the DSME-GTSs necessary to deliver all the data
 flows to destination. It provides the time required before the network
 becomes fully operational.
2. *Energy per node*, defined as the amount of energy consumed, on average,
 by a node during the network setup time. The energy, expressed in micro
 275 Joule, is measured using the *powertrace* tool available for the Contiki OS
 [15].

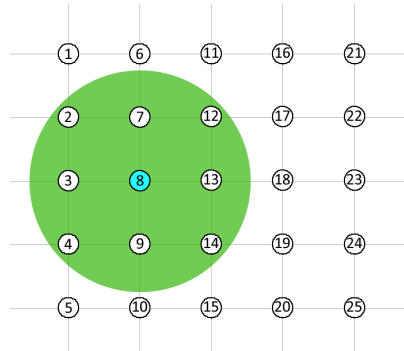


Figure 3: Mesh topology considered in emulations

5. Impact of CAP Reduction

CAP operations are very energy consuming, as nodes must set their radio in reception mode for all the CAP duration. The CAP Reduction option has been defined in order to significantly reduce energy overhead, suppressing the CAPs after the first one in the multi-superframe structure. For instance, given the chosen configuration parameter values (see Table 2), in the considered scenario the ratio between the number of CAPs with the option enabled and disabled is equal to 1:16. CAP Reduction allows not only to reduce energy consumption, but also to increase the number of available DSME-GTSs, as suppressed CAPs are replaced with 8 additional DSME-GTSs that can be allocated to nodes for data communication. CAP suppression, however, has significant repercussions on protocol's operations and performance. Management packets, such as control packets for DSME-GTS allocation/deallocation, can be exchanged only during CAPs. Hence, by enabling CAP Reduction, the opportunities to transmit control packets are drastically reduced, thus increasing the latency in allocating/deallocating DSME-GTSs.

In this section, the opportunity to enable this option is investigated by assessing its effects, during and after the network formation phase, in terms of network setup time and energy consumption.

5.1. Network Formation

Fig. 4 shows the results obtained comparing the performance of the network with *CAP Reduction enabled* (*CR_E*) and *disabled* (*CR_D*), when networks of different size are considered. As expected, enabling CAP Reduction increases the network setup time (Fig. 4a) as the number of CAPs where control packets can be exchanged is reduced, thus delaying all the DSME-GTS allocation operations. The difference in performance between the two options increases as the number of nodes grows up, because a larger network requires the allocation of more DSME-GTSs, hence increasing the competition during the CAP. On the other hand, the situation is different when energy consumption during network

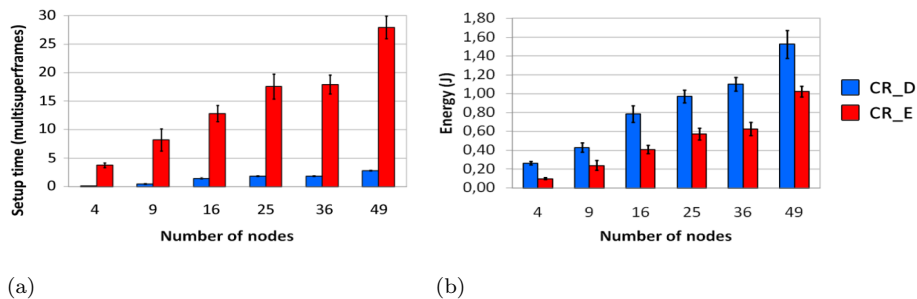


Figure 4: Setup time (a) and Energy consumption (b) with CAP Reduction disabled and enabled, ideal channel conditions

startup is considered. Fig. 4b shows that CAP Reduction helps in reducing the energy consumed by each node, even though it requires a longer network formation time. This can be explained considering that during the CAP nodes must remain active in reception, while during a DSME-GTS a node can switch its radio on the idle (or sleep) state (if the GTS is not allocated to the node itself), thus reducing dramatically its energy consumption.

5.2. Long-term Energy consumption

Once all the DSME-GTS allocations have been performed, CAPs still remain allocated, although they are not used, or they are used less frequently since DSME-GTS allocation/deallocation are performed less frequently than during formation. For this reason, it is important to assess the impact of CAP Reduction also during the operational phase, i.e., after the network formation phase. The energy consumption during the operational phase, with and without CAP Reduction, can be derived analytically. The energy consumption spent in a multi-superframe by a single node can be calculated as follows:

$$E = P_{idle}T_{idle} + P_{TX}T_{TX} + P_{RX}T_{RX} \quad (1)$$

where P_{RX}, P_{TX}, P_{idle} are the radio power consumption while receiving, transmitting or being idle, respectively, whereas T_{RX}, T_{TX}, T_{idle} represent the amount of time that the radio spends in each state.

Following (1), and assuming that α packets of maximum size are always

exchanged in each allocated DSME-GTS, the worst-case energy consumption of a node can be derived. E_E and E_D denote the energy consumption with CAP reduction enabled and disabled, respectively. Also, indicate by T_{CAP} the CAP size, T_{MD} the multi-superframe duration (both expressed in seconds) and n_{TX}/n_{RX} the number of DSME-GTSs allocated for transmission/reception. E_E and E_D can be expressed as follows:

$$\begin{aligned}
E_E &= P_{idle}T_{MD} \\
&\quad + T_{CAP}(P_{RX} - P_{idle}) \\
&\quad + n_{RX}(T_{GTS}(P_{RX} - P_{idle}) - \alpha T_{ACK}(P_{RX} - P_{TX})) \\
&\quad + \alpha n_{TX}(T_{data}(P_{TX} - P_{idle}) + T_{ACK}(P_{RX} - P_{idle})) \\
E_D &= P_{idle}T_{MD} \\
&\quad + 2^{MO-SO}T_{CAP}(P_{RX} - P_{idle}) \\
&\quad + n_{RX}(T_{GTS}(P_{RX} - P_{idle}) - \alpha T_{ACK}(P_{RX} - P_{TX})) \\
&\quad + \alpha n_{TX}(T_{data}(P_{TX} - P_{idle}) + T_{ACK}(P_{RX} - P_{idle}))
\end{aligned}$$

325 where $T_{data}(T_{ACK})$ is the time for the transmission of a packet of maximum size (ACK). Obviously, $n_{TX} + n_{RX}$ must be less than, or equal to, the overall number of DSME-GTSs inside the multi-superframe.

For ease of comparison, the *CAP Reduction energy ratio* during the operational phase is considered, defined as the ratio $\frac{E_E}{E_D}$. Fig. 5 shows the CAP
330 Reduction Energy Ratio versus n_{TX} and n_{RX} , where α is assumed equal to 4 according to the experimental settings presented in Section 4 and the other parameter values that are shown in Table 2. As can be seen, during the operational phase, i.e. after network setup, CAP Reduction can help in reducing the energy consumption of at least 56%. However, considering realistic values for n_{TX} and
335 n_{RX} (e.g., between 1 and 20, as in the considered experiments) it is possible to observe a higher reduction, up to 90%. In conclusion, CAP Reduction is a key aspect in DSME configuration, considering that in a typical scenario nodes are battery powered. The results presented so far show that CAP Reduction is highly desirable as it can help in reducing the energy consumed by each node,

340 both in the network formation and during the operational phase. Its activation, however, results in a longer network setup time, which may be unbearable for some application scenarios. In the following section, the reason why high setup times are experienced will be investigated. This will allow to better understand how to reduce it.

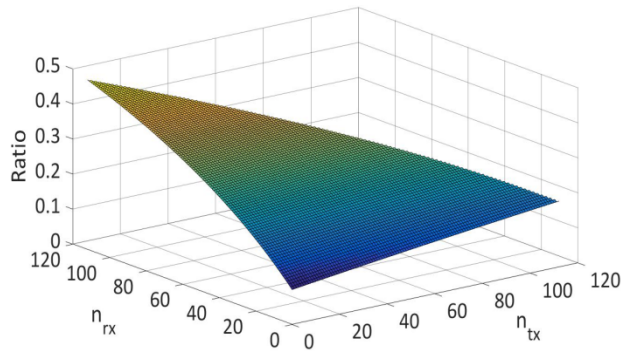


Figure 5: Steady state CAP Reduction energy ratio

345 6. DSME-GTS Allocation Analysis

In this section, the network setup phase is analyzed in detail to identify the reason of high setup times when CAP reduction is enabled. Since the time required to have the whole network operational depends on the time needed to establish each DSME-GTS, the allocation procedure is analyzed to gather an insight on the factors that influence its latency. Then, some possible solutions for reducing the latency introduced by DSME-GTS allocation are presented, so as to shorten the overall setup time and make CAP reduction acceptable also in scenarios where a fast network setup is desirable.

355 As explained in Section 2.2, DSME-GTSs are allocated through a three-way handshake during CAP. Nodes compete to gain access to the wireless medium through the CSMA-CA protocol (Sec. II.A), which, however, does not guarantee timed or collision-free transmission. Therefore, DSME control messages can be delayed, or even dropped, if the channel is found busy or collisions occur, delaying the completion of the allocation procedure or causing its failure.

360 To evaluate the performance of the DSME-GTS allocation mechanism and highlight the causes of high setup times, the outcomes that can occur when a node starts a DSME-GTS allocation request have been investigated. Four different outcomes are possible, depending on the result of the packet transmission through CSMA-CA:

- 365 • **Successful.** The procedure terminates successfully. All the allocation messages are successfully delivered. It is worth to highlight that a successful message exchange might result in a conflicting allocation, i.e. an allocation that conflicts with another existing GTS. In this case, a duplicated allocation frame is sent and another procedure is issued to negotiate
370 a different GTS.
- **Channel Busy.** The allocation procedure fails as the Request message is dropped for exceeding the maximum number of backoff stages. This happens when the node finds the wireless medium busy for a number of consecutive attempts (equal to the value of the *macMaxCSMABackoffs* CSMA-CA parameter).
375
- **No Ack.** The allocation procedure fails as the Request message is dropped for exceeding the maximum number of retransmissions, set to *macMaxFrameRetries*. This can be due to collisions with concurrent transmissions from other nodes, or to a external interference.
- 380 • **Timeout.** The allocation procedure fails after the successful transmission of a Request message as the Response message is not received within a maximum waiting time. The Response message might be dropped for exceeding the maximum number of transmissions or backoff stages, or it may have collided or been corrupted.

385 It is important to highlight that in the considered DSME implementation, whenever a DSME-GTS allocation procedure fails or a conflict allocation is detected, a new allocation procedure is triggered in the successive CAP, until a DSME-GTS is finally established.

Fig. 6 shows the fraction of the four outcomes in the experiments performed
390 in Section 5.1 in terms of the percentage with respect to the total number of
allocation requests. As can be observed, only a small fraction of allocation re-
quests end successfully, which justifies the long time required for allocating all
the DSME-GTSs. The high number of unsuccessful allocation requests shows
that the wireless channel is often found busy and collisions can occur very fre-
395 quently. This can be explained considering that, in the considered setup, each
node has one packet to transmit at network startup. Hence, all the nodes try to
access the channel during the same initial CAP. This results in a large number
of allocation requests that fail and are re-issued subsequently, requiring more
CAPs to be completed.

400 These results allow to draw some conclusions on the factors that cause longer
delays in DSME-GTS allocations.

1. *CSMA-CA default parameter values.* The default parameter values are
very low. Therefore, backoff windows are short and few attempts are
performed in sending a packet. A proper tuning of these parameters would
405 be advisable to reduce the number of collisions and to give packets more
chances to be transmitted.
2. *Radio state during backoff.* According to the CSMA-CA algorithm, before
sending a packet, a node waits for a random backoff time. During this time,
most CSMA-CA implementations set the radio to the idle (or sleep) state,
410 to minimize energy consumption. However, since in this state, incoming
transmissions are ignored, control packets sent from neighbors are not
received, thus delaying DSME-GTS allocation.
3. *Loss of Reply packets.* Since Reply messages are sent in broadcast, they
are not acknowledged. When a Reply message is lost (e.g. due to a
415 collision), inconsistency between nodes occurs. The node that sent the
Reply assumes a successful allocation, while the other node assumes a
failed procedure. As consequence, after a timeout, the initiator starts a
new allocation procedure, thus increasing latency and wasting resources.

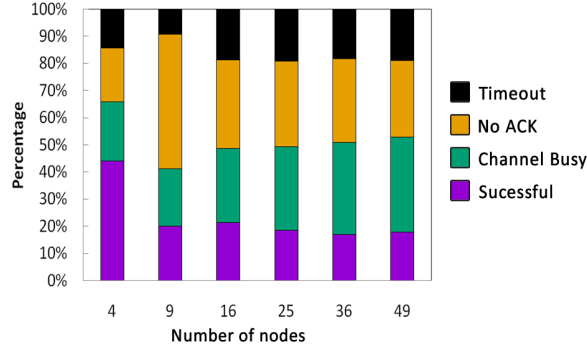


Figure 6: Outcomes for DSME-GTS Allocation Requests

7. DSME-GTS Allocation Improvements

420 In order to improve the efficiency of the allocation procedure, possible improvements to mitigate the effects of the highlighted issues are analyzed, focusing on the first two, as they do not require modifications to the DSME standard. In particular, the tuning of CSMA-CA parameters is investigated, and a minor modification of the node behavior, called *Active Backoff*, is proposed to increase
 425 the probability in successfully receiving control messages during the allocation process.

7.1. CSMA-CA Parameter Tuning

Tuning the CSMA-CA parameters to satisfy communication requirements – in terms of reliability, latency, energy efficiency, etc. – is a well-known practice
 430 in the literature [16][17]. In the considered scenario, the goal is minimize the number of CAPs necessary to complete the allocation procedure. However, tuning the CSMA-CA parameter values is not a trivial task and results in a trade-off. On one side, large backoff windows and numerous backoff/transmission attempts would be desirable to efficiently manage contention. On the other, too
 435 long backoff periods or too many backoff/transmission attempts may cause the transmission of allocation packets to extend over successive CAPs, thus further increasing the overall allocation time.

The latency of a DSME-GTS allocation/deallocation procedure is strictly dependent on the time needed for exchanging Request and Reply messages. In fact, nodes can start using the allocated DSME-GTS for data transmission right after the transmission/reception of the Reply message. Although the handshake is concluded by the Notify message, this last message is only a notification message that confirms the successful allocation of the DSME-GTS to the requesting node. Based on this premise, the goal of CSMA-CA parameter tuning is to *fit the exchange of Request and Reply messages within a single CAP*. To this end, it is worth noting that, according to the standard, a Reply message must be received within a timeout period equal to $macMaxFrameTotalWaitTime$ symbols from the Request transmission. In particular, $macMaxFrameTotalWaitTime$ can be expressed as follows [5]:

$$\begin{aligned}
 macMaxFrameTotalWaitTime = & \\
 & \left[\left(\sum_{k=0}^{m-1} 2^{macMinBE} \right) + (2^{macMinBE}) \times (macMaxCSMABackoffs - m) \right] \\
 & \times aUnitBackoffPeriod + phyMaxFrameDuration
 \end{aligned}$$

where $aUnitBackoffPeriod$ is the number of symbols forming a backoff slot, and $m = \min(macMaxBE - macMinBE, macMaxCSMABackoffs)$. Starting from this consideration, it is reasonable to assume that also Request packets should be sent within $macMaxFrameTotalWaitTime$ symbols from the instant of their generation. Hence, to complete the allocation in just one CAP, the CAP size must be

$$CAP = 2 \times macMaxFrameTotalWaitTime \quad (2)$$

By solving equation (2), the CSMA-CA parameter values that allow to complete the allocation within one single CAP can be obtained. Such set of parameters is referred hereafter as *Analytical Parameter Set (APS)*. A possible approach to determine the APS, fixed the *Superframe Order (SO)*, is presented in Algorithm 1. Specifically, the $macMaxFrameTotalWaitTime$ is calculated for each possible

ALGORITHM 1. *Analytical Parameter Set Derivation*

```
1  $T_{CAP} \leftarrow aBaseSuperframeDuration \times 2^{SO} \times \frac{9}{16}$ 
2 for macMinBE =  $MinBE_{min}$  to  $MinBE_{max}$ 
3   for macMaxBE =  $MaxBE_{min}$  to  $MaxBE_{max}$ 
4     for macMaxCSMABackoffs =  $MaxCSMABackoffs_{min}$  to
        $MaxCSMABackoffs_{max}$ 
5       FrameTotalWaitTime = Calculate_time(macMinBE, macMaxBE,
       macMaxCSMABackoffs);
6     if  $|T_{CAP} - 2 \times FrameTotalWaitTime| < |T_{CAP} - 2 \times APS\_FTWT|$ 
7       APS_FTWT = FrameTotalWaitTime
8       APS[macMinBE] = macMinBE
9       APS[macMaxBE] = macMaxBE
10      APS[macMaxCSMABackoffs] = macMaxCSMABackoffs
```

Table 3: Derived Analytical Parameter Set (APS)

Parameter	Value
macMinBE	6
macMaxBE	8
macMaxCSMABackoffs	4
macMaxFrameRetries	3

combination of parameter values (*lines 2-5*). The APS is chosen as the set of values closer to fulfilling equation (2) (*lines 6-10*). To avoid a combinatory search, the mechanism can be easily implemented through a lookup table. The minimum and maximum values for the CSMA-CA parameters are indicated in Table 1. The APS for the considered emulation scenario is summarized in Table 3.

In order to evaluate the allocation performance with this set of parameter values, the same experiments presented in Section 5.1 have been carried out, considering the CAP Reduction option enabled. As terms of comparison, the *Default Parameter Set (DPS)*, see Table 1) and the *Shortest Setup Parameter Set (SSPS)* are used. The latter is the set of parameter values that provides the best performance for the considered scenario, in terms of setup time. It has been determined through experiments, by exhaustively testing all the possible combinations of values. Obviously, it has been used only for comparison purpose, since it *cannot* be determined by nodes at runtime.

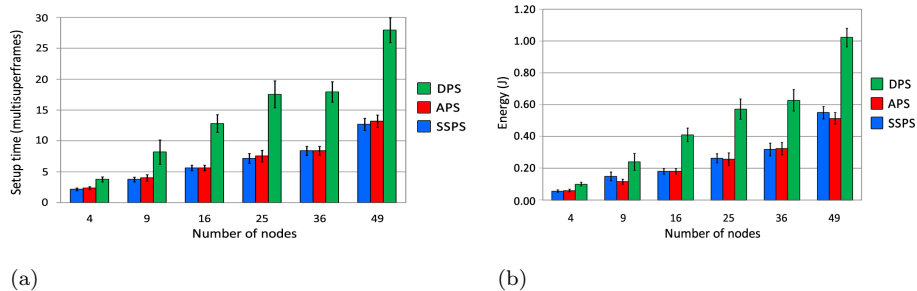


Figure 7: Network setup time (a) and Energy consumption (b) with different Parameter Sets

The results of comparison are reported in Fig. 7. As can be observed, the APS setting succeeds in reducing significantly the network formation time, mitigating CAP congestion. In addition, APS performance is very close to the one of the optimal setting, SSPS, which shows only a negligible improvement. Furthermore, the parameter set that minimizes the setup time does not necessarily minimize also the energy consumption. In fact, APS exhibits an energy consumption similar to SSPS, or even lower (e.g., 49 nodes). It is important to highlight that the gain in performance obtained with parameter tuning increases with the network size. This is because more nodes require the allocation of a larger number of DSME-GTSs, resulting in a higher degree of contention. Hence, a proper tuning of CSMA-CA parameters is more beneficial.

In order to evaluate the performance of the APS setting when the number of nodes in contention varies, an additional set of experiments has been performed. To this aim, different network densities are considered varying the transmission/reception range of the nodes. Specifically we considered the 7x7 grid scenario in which three different transmission ranges are considered, i.e. 25m (the value considered in the previous experiments), 32m and 38m. As shown in Fig. 8, increasing the transmission range results in larger neighborhoods in which an increasing number of nodes compete for channel access, i.e. 8, 12 and 20 nodes in the worst case. Although an increasing transmission/reception range would allow to configure a static routing with shorter routes, the same static routes from the previous experiments are configured in the network. This

guarantees a fair comparison as the same number of DSME-GTS are allocated in all the scenarios.

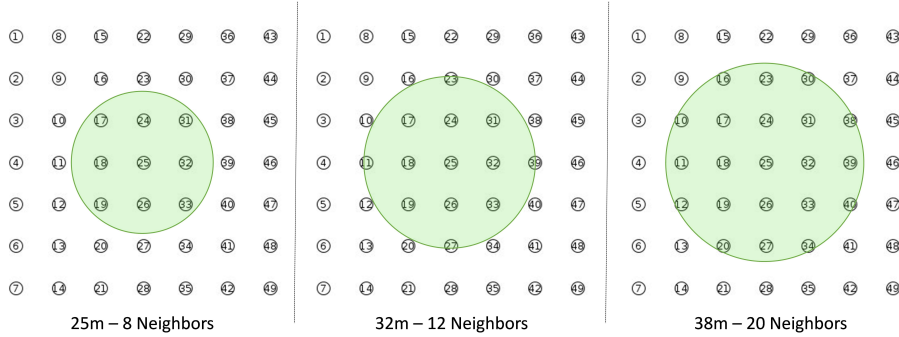


Figure 8: Simulation scenarios with variable transmission ranges.

500 Fig. 9 reports the setup time and the energy consumption. As can be observed, the APS setting is confirmed to succeed in reducing dramatically the network formation time and the energy consumption in comparison with DPS. Compared with the optimal setting SSPS, APS performance is still close both in terms of energy consumption and network formation time. The only noticeable
 505 difference is displayed in the scenario with 20 nodes as neighborhood size, in which a limited increase in the gap between APS and SSPS network formation time is shown. Such difference, however, is limited considering the high network density and the very high degree of contention, which exacerbates the advantage in adopting the optimal set of parameters.

510 *7.2. Active Backoff mechanism*

In order to reduce the number of packets dropped during CAP, a modification to the behavior of nodes during backoff is proposed. In particular, an *Active Backoff* approach is introduced to allow nodes receiving packets also during a backoff time. In other words, when a node starts the backoff procedure to send
 515 a packet, it sets its radio to the reception mode, rather than idle or sleep. If the node starts receiving a packet during the backoff time, it interrupts the backoff count and waits until the packet is completely received and, then, stores the

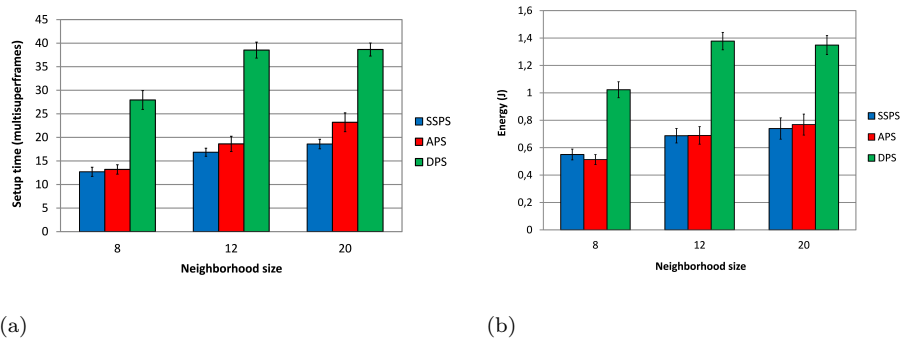


Figure 9: Network setup time (a) and Energy consumption (b) with different neighborhood sizes

packet in a temporary buffer. Afterwards, the node resumes the backoff time, subtracting the time spent for receiving the packet. As soon as the node has performed its transmission, it processes the packet stored in the buffer.

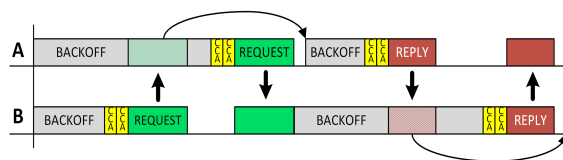


Figure 10: Example of DSME-GTS allocation with Active Backoff

Consider the example in Fig. 10 where node A requires a DSME-GTS to node B and vice-versa. Since both nodes need to transmit a Request packet, they start a backoff stage. In this example, node B chooses a shorter backoff time and starts transmitting while node A is still waiting. In this case, the Active Backoff mechanism allows node A to receive the packet. Once the backoff time ends, node A sends its own Request and processes the received packet. At this point, since both node A and node B have received a Request frame, they start another backoff procedure to transmit the Reply packet. Hence, the Active Backoff mechanism allows to complete the Request-Reply exchange in just one attempt. Conversely, without Active Backoff, the two nodes would have to start over the allocation procedure, as Request messages are not received by

node A and node B. Although the Active Backoff does not require significant modifications to the regular behavior of a node, the allocation of an additional buffer is required. Since sensors are typically constrained devices with limited memory, the allocation of an additional buffer might represent an issue. In the considered implementation, an additional buffer has been introduced to store only one packet. Specifically, only the latest received packet is stored, which overwrites any previous packet. It is important to point out that implementing the Active Backoff mechanism does not imply any modification of the standard. Nodes implementing Active Backoff can interoperate with other nodes that do not use it.

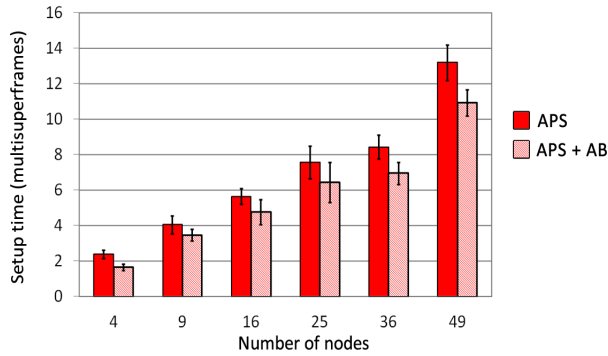


Figure 11: Network setup time with Active Backoff

The gain that can be obtained with Active Backoff is evaluated in Fig. 11, which shows network setup time, with and without Active Backoff, when the APS parameter set is used and CAP Reduction is enabled. As can be seen, the Active Backoff helps in further improving the performance reducing the network setup time up to about 15%. For the sake of brevity, energy consumption is omitted. However, the same trend as the network formation time has been observed for all the considered configurations, with a reduction of about 15%. A comparison of the network setup time obtained with the two proposed mechanisms against the default parameter values (DPS) is presented in Fig. 12. Compared to the original (default) configuration, the two proposed enhancements exhibit a significantly better performance, allowing to reduce the network setup time

of about 40-60%. In particular, the gain increases when the network becomes larger and longer, as parameter tuning and Active Backoff help in mitigating the contention. For the sake of brevity, energy consumption is not reported as it shows a trend alike the network setup time. A reduction of about 40-60% compared with the default settings is reported in all the considered configurations (e.g., with 49 nodes it drops from 1.05 J to 0.45 J). Although such performance is not comparable with the setup time obtained without CAP Reduction, the proposed improvements can make the network formation acceptable in numerous scenarios also with CAP Reduction enabled, allowing to significantly reduce the energy consumption.

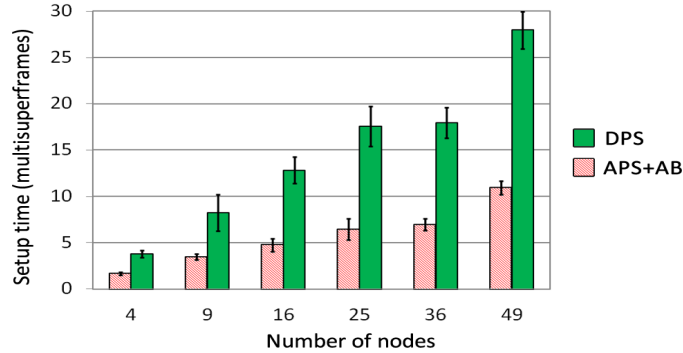


Figure 12: Network setup time: APS with Active Backoff vs. DPS

8. Evaluation with Lossy Channel

In order to assess the performance of the DSME-GTS allocation procedure in a more realistic environment, experiments with lossy channel have been carried out. To this aim, the Gilbert-Elliot (GE) model has been used to simulate packet errors/losses, as it provides a good approximation of fading in industrial environments [18]. In addition, this model has been used in a number of previous performance analysis of industrial wireless systems, e.g. Brienza et al. [16] and De Pellegrini et al. [19]. A continuous-time Markov Chain, consisting of two states, namely bad and good, models the channel for each receiver. When

in good state, *all packets* are decoded correctly, whereas in bad state *packets* are dropped, modeling high interference or drop in signal quality. Sojourn times in the two states follow an exponential distribution and their average value determines the Packet Error Rate (PER) experienced during the entire simulation. Model parameters are derived using the same approach adopted by De Pellegrini et al. [19] and Anastasi et al. [12], in which values are inspired from the real measurements [18]. For instance, if PER is set to 10%, the resulting average sojourn time in the bad and good state is 5.7 and 46.2 ms, respectively. Larger values of PER are obtained by tuning the average sojourn time in the bad state accordingly, while leaving all the other parameters unchanged.

Fig. 13 shows the impact of the lossy channel over the network setup time. With reference to the grid-network of 49 nodes, four different levels of PER have been considered, i.e., 0%, 10%, 20% and 30%. As it can be observed, the network setup time significantly increases as the PER increases, with DPS, APS and the proposed configuration with both APS and Active Backoff. This is due to the fact that the loss of packets is compensated with retransmissions to correctly deliver the messages of the DSME-GTS allocation procedure. This increases the channel congestion and slows down the overall allocation process. Moreover, when a Reply message is dropped, the whole DSME-GTS allocation fails after timeout and is repeated, as Reply packets are sent in broadcast. These results, however, show that the proposed solution helps in significantly reduce the overall setup time in all the considered scenarios. Most of the reduction in the setup time is achieved when using APS instead of default parameters. Active Backoff also helps in further decreasing the setup time.

9. Conclusions

In this paper, the IEEE 802.15.4e DSME MAC behavior mode is analyzed. Initially, the effects of the CAP Reduction option are evaluated, showing how this option can significantly reduce nodes' energy consumption. However, it increases the network setup time. Experimental results highlight that this problem

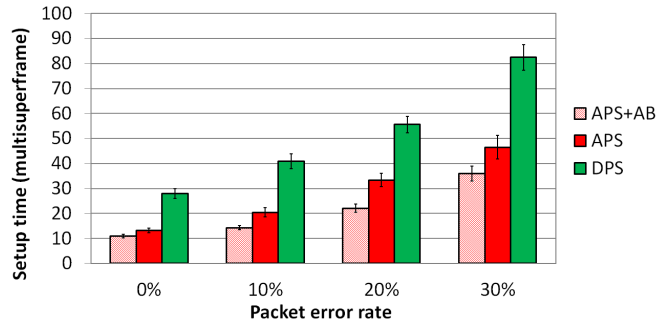


Figure 13: Network Setup time with different PER levels

is caused by the DSME-GTS allocation procedure that requires a three-packet exchange through a CSMA-CA protocol. When nodes compete to allocate DSME-GTSs, a large fraction of these control packets are dropped, since the maximum number of backoff stages and/or the maximum number of retransmissions is exceeded. In order to reduce the network setup time, two solutions are proposed. The first one consists in a simple methodology for proper selection of CSMA-CA parameter values. The second one consists in a minor modification of the behavior of nodes, called Active Backoff, that allows reception of packets even during backoff periods. Both the solutions reduce the collision probability and increase nodes' reception. Experimental results show that the network setup time can be drastically reduced up to 60%.

References

- [1] Eleonora Borgia, "The Internet of Things vision: Key features, applications and open issues", *Computer Communications*, Volume 54, 1 December 2014, Pages 1-31, ISSN 0140-3664.
- [2] P. Di Marco, G. Athanasiou, P.-V. Mekikis, C. Fischione, "MAC-aware routing metrics for the internet of things", *Computer Communications*, Volume 74, 15 January 2016, Pages 77-86.

- [3] A. Willig, “Recent and Emerging Topics in Wireless Industrial Communi-
620 cations: A Selection,” *IEEE Trans. Ind. Inform.*, vol. 4, no. 2, pp. 102-124,
May 2008.
- [4] IEEE802.15.4e: IEEE standard for local and metropolitan area networks.
Part 15.4: Low-Rate Wireless Personal Area Networks (LRWPANs) Amend-
ment 1: MAC Sublayer, Institute of Electrical and Electronics Engineers
625 Std., Apr. 2012.
- [5] IEEE Computer Society, “IEEE Standard for Information technology, Part
15.4; Wireless Medium Access Control (MAC) and Physical Layer (PHY)
Specifications for Low-Rate Wireless Personal Area Networks (LRWPANs)”,
2011.
- 630 [6] D. C. Hoang, P. Yadav, R. Kumar and S. K. Panda, “Real-Time Imple-
mentation of a Harmony Search Algorithm-Based Clustering Protocol for
Energy-Efficient Wireless Sensor Networks,” *IEEE Trans. Ind. Inform.*, vol.
10, no. 1, pp. 774-783, Feb. 2014.
- [7] J. Niu, L. Cheng, Y. Gu, L. Shu and S. K. Das, “R3E: Reliable Reactive
635 Routing Enhancement for Wireless Sensor Networks,” *IEEE Trans. Ind.
Inform.*, vol. 10, no. 1, pp. 784-794, Feb. 2014.
- [8] D. De Guglielmo, S. Brienza, G. Anastasi, IEEE 802.15.4e: A survey, Com-
puter Communications, Available online 13 May 2016, ISSN 0140-3664.
- [9] S. Capone, R. Brama, F. Ricciato, G. Boggia and A. Malvasi, “Modeling and
640 simulation of energy efficient enhancements for IEEE 802.15.4e DSME,” in
Proc. Wireless Telecommunications Symposium (WTS), 2014, Washington,
DC, pp. 1-6, 2014.
- [10] T. Paso, J. Haapola and J. Iinatti, “Feasibility study of IEEE 802.15.4e
DSME utilizing IR-UWB and S-Aloha,” in *Proc. IEEE 24th International
645 Symposium Personal Indoor and Mobile Radio Communications (PIMRC)*,
London, pp. 1863-1867, 2013.

- [11] I. Juc, O. Alphand, R. Guizzetti, M. Favre, A. Duda, “Energy Consumption and Performance of IEEE 802.15.4e TSCH and DSME,” in *Proc. IEEE Wireless Communications and Networking Conf. (WCNC)*, Doha, Qata, Apr. 2016
- [12] G. Anastasi, M. Conti, and M. Di Francesco, “A Comprehensive Analysis of the MAC Unreliability Problem in IEEE 802.15.4 Wireless Sensor Networks,” *IEEE Trans. Ind. Inform.*, vol. 7, no. 1, pp. 52-65, Feb. 2011.
- [13] E. Ancillotti, R. Bruno and M. Conti, “Reliable Data Delivery With the IETF Routing Protocol for Low-Power and Lossy Networks,” *IEEE Trans. Ind. Inform.*, vol. 10, no. 3, pp. 1864-1877, Aug. 2014.
- [14] Texas Instruments. 2012. CC2420 2.4GHz IEEE 802.15.4/ZigBee ready RF Transceiver. Online: <http://focus.ti.com/lit/ds/symlink/cc2420.pdf>
- [15] A. Dunkels, J. Eriksson, N. Finne, and N. Tsiftes, “Powertrace: Network-level power profiling for low-power wireless networks,” Technical report, 2011.
- [16] S. Brienza, M. Roveri, D. De Guglielmo, and G. Anastasi, “Just-in-Time Adaptive Algorithm for Optimal Parameter Setting in 802.15.4 WSNs,” *ACM Trans. Auton. Adapt. Syst.*, vol. 10, 4, Jan. 2016.
- [17] P. Park, P. Di Marco, C. Fischione and K. H. Johansson, “Modeling and Optimization of the IEEE 802.15.4 Protocol for Reliable and Timely Communications,” *IEEE Trans. Paralle. Distrib. Syst.*, vol. 24, no. 3, pp. 550-564, Mar. 2013.
- [18] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, “Measurements of a wireless link in an industrial environment using an IEEE 802.11-compliant physical layer,” *IEEE Trans. Ind. Electron.*, vol. 49, no. 6, pp. 1265–1282, 2002.
- [19] F. De Pellegrini, D. Miorandi, S. Vitturi, and A. Zanella, “On the Use of Wireless Networks at Low Level of Factory Automation,” *IEEE Trans. Ind. Inform.*, vol. 2, no. 2, May 2006.

- 675 [20] N. Abdeddaim, F. Theoleyre, M. Heusse and A. Duda, “Adaptive IEEE 802.15.4 MAC for Throughput and Energy Optimization” *IEEE International Conference on Distributed Computing in Sensor Systems*, Cambridge, MA, 2013, pp. 223-230.
- [21] X. Liu, X. Li, S. Su, Z. Fan, G. Wang. 2013. “Enhanced Fast Association for 802.15.4e-2012 DSME MAC Protocol” *In Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE)*, Hangzhou, China, 22-23 March 2013.
- 680 [22] K. Hwang and S. Nam, “Analysis and Enhancement of IEEE 802.15.4e DSME Beacon Scheduling Model” *Journal of Applied Mathematics*, vol. 2014, Article ID 934610, 15 pages, 2014.
- 685 [23] W. y. Lee, K. i. Hwang, Y. A. Jeon and S. Choi, “Distributed Fast Beacon Scheduling for Mesh Networks” *IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*, Valencia, 2011, pp. 727-732.
- [24] P.K. Sahoo, R. P. Sudhir , and W. Shih-Lin. “A Novel IEEE 802.15. 4e DSME MAC for Wireless Sensor Networks” *Sensors* (2017): 168.
- 690 [25] W.C. Jeong, and L. Junhee. “Performance evaluation of IEEE 802.15. 4e DSME MAC protocol for wireless sensor networks” *Enabling Technologies for Smartphone and Internet of Things (ETSIoT)*, 2012 First IEEE Workshop on. IEEE, 2012.
- [26] G. Alderisi, G. Patti, O. Mirabella and L. L. Bello, “Simulative assessments of the IEEE 802.15.4e DSME and TSCH in realistic process automation scenarios” *IEEE 13th International Conference on Industrial Informatics (INDIN)*, Cambridge, 2015, pp. 948-955.
- 695 [27] S. Capone, R. Brama, F. Ricciato, G. Boggia and A. Malvasi, “Modeling and simulation of energy efficient enhancements for IEEE 802.15.4e DSME” *Wireless Telecommunications Symposium*, Washington, DC, 2014, pp. 1-6.
- 700