Master's Thesis

Informatik-Ingenieurwesen

Joint Analysis of TDMA and Geographic Routing for a Large Scale Wireless Mesh Network

by

Tobias Lübkert

ORCID iD: 0000-0001-5125-5191

July 2015

Supervised by

Florian Meier
Institute of Telematics, Hamburg University of Technology

First Examiner

Prof. Dr. Volker Turau
Institute of Telematics
Hamburg University of Technology

Second Examiner

Prof. Dr.-Ing. Andreas Timm-Giel
Institute of Communication Networks
Hamburg University of Technology

Please cite by using the DOI: 10.15480/882.1664

Institute of Telematics
Hamburg University of Technology

TUHH

Declaration by Candidate

I, TOBIAS LÜBKERT (student of Informatik-Ingenieurwesen at Hamburg University of Technology), hereby declare that this thesis is my own work and effort and that it has not been submitted anywhere for any award. Where other sources of information have been used, they have been acknowledged.

Hamburg, July 1st, 2015

Tobias Lübkert

Abstract

Wireless Mesh Networks are a promising technique for large scale networks with low-power and high reliability requirements such as in industrial plants. The shared transmission medium requires a medium access control mechanism which reduces the amount of parallel transmission, because nearby simultaneous transmitting devices hinder the successful reception at the destination device. The common CSMA-CA mechanism cannot satisfy the reliability requirement of an industrial plant, as many messages get dropped when the channel is busy for too long.

Current standards introduce TDMA schemes for wireless mesh networks to improve the reliability with higher throughput and deterministic latency. In this work these standards are reviewed and one of the specified TDMA protocols, the deterministic synchronous multi-channel extension (DSME), is analyzed in more detail by examining its performance analytically and also by simulation. For this purpose, a simulation model was implemented, which is used to simulate an example network utilizing DSME and a promising geographic routing algorithm for a solar tower power plant.

The introduced analytical model allows to determine the maximum performance for a network using DSME. The simulation results show that DSME achieves a strongly improved packet loss probability compared to CSMA-CA.

Table of Contents

Chapter 1

Introduction

Wireless Mesh Networks are currently at the transition from an academic topic to the use in our every days life and cover a wide range of applications. Smart home devices increase the quality of life by autonomously adapting to changing conditions, for example by controlling the heater or air-conditioning systems in an optimal way. Internet of things connects these devices to the Internet, providing them arbitrary information, which allows cost- and energy-effective control when considering up to date energy prices. Wireless Sensor Networks allow a long-term monitoring of isolated or even hostile areas without the need for recurrent maintenance by a person. Sensor Networks are also an emerging technique in healthcare systems by building up a body area network of several sensors placed on the human body. The monitored data can get collected by special medical device or even by the patients own smart phone. The cost-effective and convenient deployment of such networks make them equally well suited for industrial applications where reliable monitoring and controlling of remote subsystems is essential. By using wireless field-buses instead of conventional wired field-buses, the investment costs can be reduced significantly.

Several standards have been published to assure the robustness of systems even when utilizing devices of different manufacturers, such as the IEEE 802.15.4 standard which covers the physical layer and the medium access control. There are several standards covering upper layers like ZigBee, 6LoWPAN or Z-Wave, which play a role in many projects. However, these standards are still facing issues to assure reliable communications for large scale networks of several thousand devices, which is essential for industrial applications. An applicable example is a solar tower power plant where heliostats redirect the sunlight collected on a large area to a single part of a tower. Electricity gets generated using steam turbines which are powered by the heat collected at the tower. Such power plants may comprise hundreds of thousands of heliostats placed on concentric circles with the tower in the center. Figure 1.1 shows an exemplary illustration of such a plant that would highly benefit of a wireless field bus.

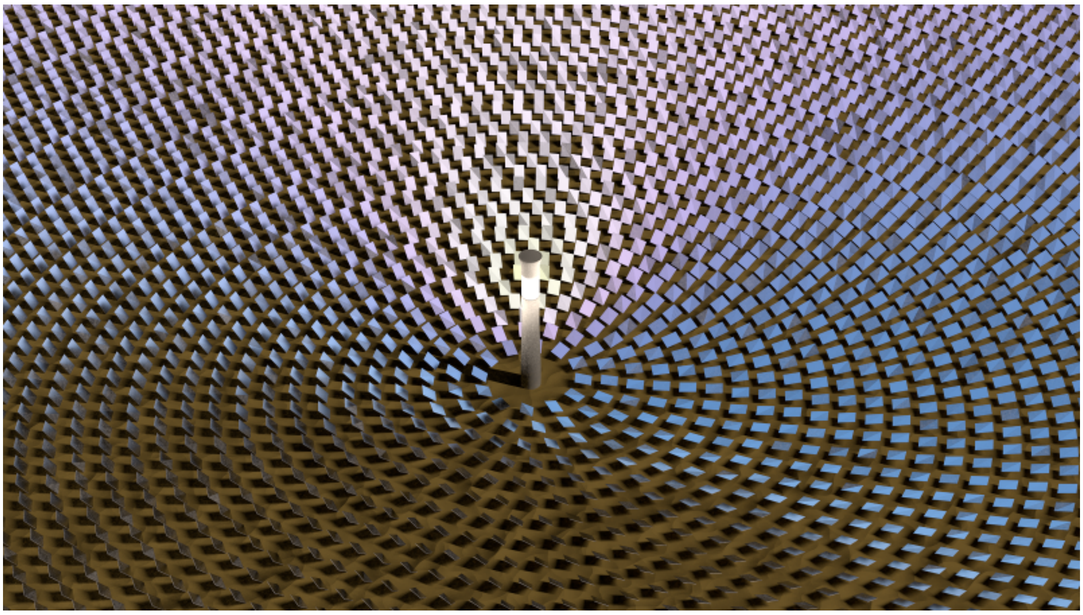


Figure 1.1: Illustration of a solar tower power plant. CC BY, Florian Kauer

The main problem yields from the common CSMA-CA medium access control mechanism, which is easy to implement and often supported by the hardware. CSMA reduces the number of simultaneous transmissions by assessing the channel before transmitting and retransmits messages in case of disturbances. However, in larger dense networks with high traffic demands, such as in solar tower power plants, this technique reaches its limits and thus shows an unpromising scalability. Furthermore, most routing protocols used in wireless mesh networks depend on large routing tables and thus are not suitable for large networks, as well.

The target of this work is to analyze the maximum achievable performance of recent techniques that promise a much better scalability analytically and to determine the influencing factors. Furthermore, an exemplary large scale wireless mesh network for a solar tower power plant is evaluated by simulation to determine a robust network configuration.

TDMA is a promising mechanism to achieve scalability with high reliability. With TDMA collisions are avoided by a priori scheduling the transmission time for each node. Another strong influence on the scalability is given by the routing protocols. A lot of work has been done to analyze the performance of these techniques independently. Whereas this work analyzes the possible performance of DSME, a recently standardized TDMA protocol, jointly with a promising geographic routing within the context of the regarded concentric circle topology.

Routing algorithms determine the next-hop forwarding device to allow communication with far destinations in a large scale network, requiring intermediate devices to forward messages, as wireless transmission is limited in range. These decisions have a strong impact on a TDMA based medium access control mechanism. Thus it is essential to jointly analyze the TDMA and the overlying routing protocol. This work considers a geographic routing protocol to be most promising, because of its scalability through hop-by-hop routing decisions without the need for large routing tables. An analytical model is proposed, which allows to determine the maximum possible performance for a given network setup using DSME and geographic routing. Analytical results are faster attainable and easier to be reproduced compared to simulative results and allow their verification as well. After analytically examining the throughput, latency and packet loss probability of these techniques, the implementation of a simulation model using OMNeT++ and the INET framework is presented. This model is then used to evaluate the performance of an example network and to compare the results to the analytical model.

The results show that utilizing DSME and geographic routing achieves much better packet loss probabilities compared to CSMA-CA. Thereby, this work makes an important contribution to the industrial realization of a wireless control for a heliostat power plant.

Chapter 2 surveys the set of problems and techniques to increase the scalability. Different medium access control mechanisms are described, which determine how devices arrange their communication. Furthermore, current standards for the field of applications and several routing strategies are reviewed. The standards mostly focus on the physical layer and medium access control mechanisms. DSME is discussed in more detail in Chapter 3. Geographic routing and the underlying network topology considered in this work are discussed in Chapter 4, which also proposes a suitable algorithm to efficiently distribute the traffic within the topology.

In Chapter 5 the implementation of the simulation model is presented, which was implemented as part of this work to deeper analyze the performance of the given network using the selected techniques. The performance of this protocol is then further discussed analytically as part of this work in Chapter 6, considering throughput, latency and packet loss. Furthermore, the simulation results of an exemplary network with different traffic loads using the proposed routing algorithm and the implemented simulation model of DSME are analyzed in Chapter 7.

Chapter 2

State of the Art

Devices in a wireless network communicate with neighbors using a shared medium within a limited range. To reach more distant destinations, messages have to be forwarded by neighboring nodes on a routing path. The mesh topology allows nodes to be linked with multiple neighbors, which in general enables higher reliability through several routing paths between two nodes [HL07] and is therefore also well suited for wireless sensor networks [MPSP10]. As radio signals interfere within a limited range, parallel transmission of neighbors may lead to high bit error rates which make a successful reception impossible. In addition to that, the quality of a wireless link is also affected by other objects in the environment or the weather conditions. According to [HL07], the network capacity may be increased by optimizing the medium access and routing. Beside the transmit power, also the channel and appropriate time to transmit have to be assigned to every device within the network.

This chapter briefly describes common medium access control mechanisms, basic routing aspects with the focus on geographic routing as well as joint design strategies of these layers. Furthermore, current standards suitable for the field of application are discussed, which mostly cover the physical layer and medium access control. The last section consolidates fundamentals of queueing models to allow estimating network performances.

2.1 Medium Access Control

Wireless network devices utilize the air as shared transmission medium. Therefore, simultaneous communication of different devices on the same channel may cause interference at a receiving device, which then is unable to receive transmitted data. To avoid such situations, wireless devices need to co-ordinate their communication. Such mechanisms are described in this section.

2.1.1 Carrier Sense Multiple Access - Collision Avoidance (CSMA-CA)

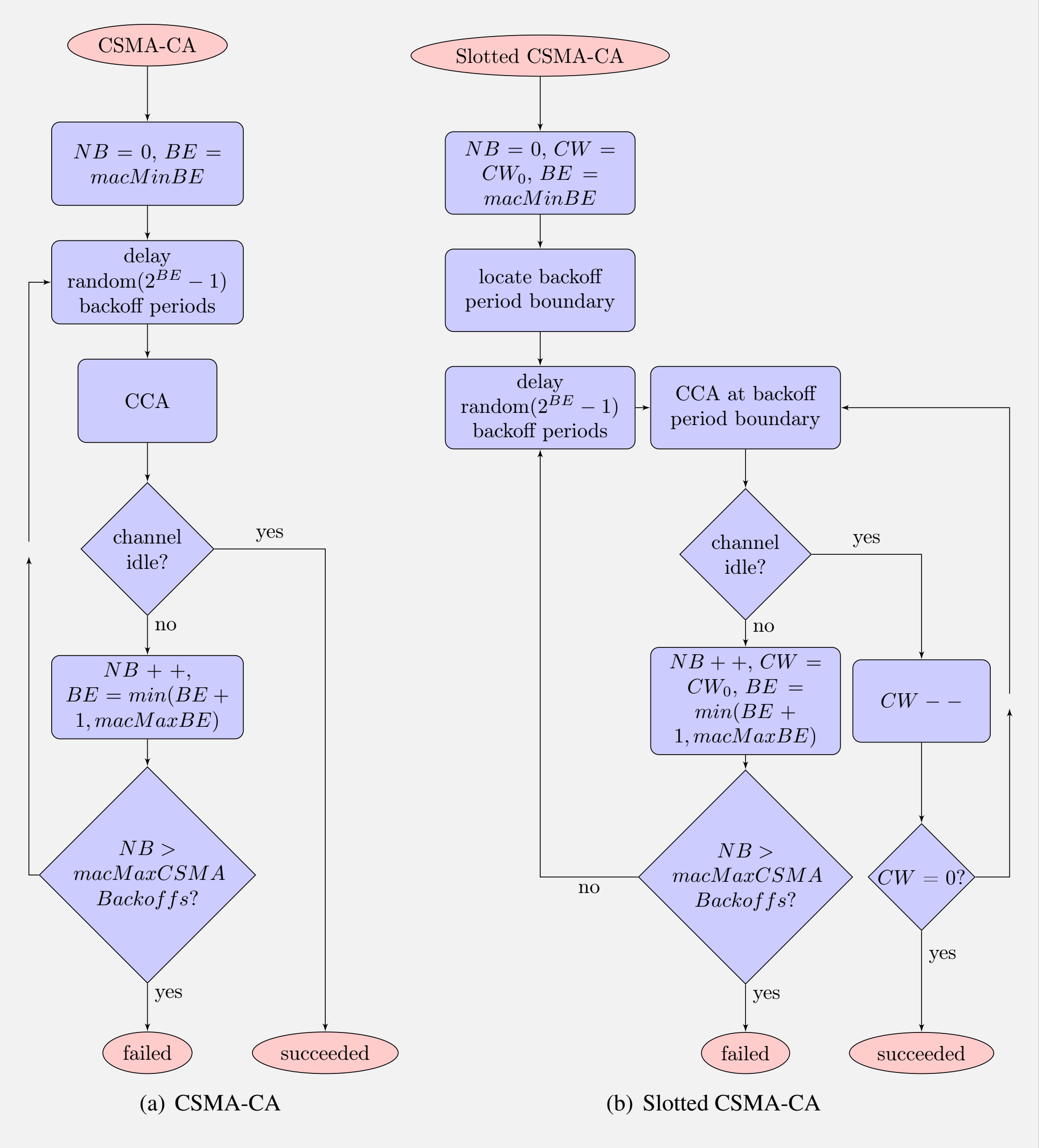
The CSMA-CA mechanism is commonly used in wireless communication networks as it requires no information about other network devices and is thus easy to implement [PD11]. With CSMA-CA, as described in [IEE11], a device senses the medium before it starts transmitting itself. This is done within the clear channel assessment (CCA) part of the flow chart shown in Figure 2.1(a). To reduce concurrent transmission of multiple devices, the CCA is done after a random delay. Depending on the backoff exponent BE, the random delay has a value between 0 and 2^{BE}, where BE has a default minimum value of 3 and a maximum of 5. If the channel was not idle, because some other device was transmitting, the number of backoffs NB and the backoff exponent BE are incremented and the next CCA will be performed after another random delay, which may be twice as long as before. If the channel was busy too often, CSMA will stop further attempt to transmit the given data and inform the next higher layer. The default maximum backoff count $macMaxCSMABackoffs$ is 4, which will stop reattempts after the fifth CCA with a busy channel.

2.1.2 Slotted CSMA

In slotted CSMA [IEE11], devices of the same network need to be time synchronized, as the transmission and the CCA will only be performed at time slot boundaries. In contrast to unslotted CSMA, devices perform multiple CCAs within a contention window CW_0. A device will start transmission when the channel was idle for CW_0 slots in a row, where CW_0 usually has a value of 2. Figure 2.1(b) shows the flow chart of slotted CSMA. An advantage of slotted CSMA is, that it reduces power consumption as the receiver only has to be activated for a short time around the slot boundaries and for receiving actual frames. However, it was shown in [WLZ09] that slotted CSMA is unable to achieve a better performance and it may even show an inferior throughput and collision probability.

2.1.3 Time-Division Multiple Access (TDMA)

TDMA is also a common protocol for wireless networks, as it is promising to achieve better performance results for multi-hop networks [PD11]. With TDMA, devices may exclusively


CSMA-CA
$NB = 0$, $BE = macMinBE$
delay random$(2^{BE}-1)$ backoff periods
CCA
channel idle?
yes
no
$NB++$, $BE = min(BE+1, macMaxBE)$
$NB > macMaxCSMABackoffs?$
yes
failed
succeeded
Slotted CSMA-CA
$NB = 0$, $CW = CW_0$, $BE = macMinBE$
locate backoff period boundary
delay random$(2^{BE}-1)$ backoff periods
CCA at backoff period boundary
channel idle?
yes
no
$NB++$, $CW = CW_0$, $BE = min(BE+1, macMaxBE)$
$CW--$
$NB > macMaxCSMABackoffs?$
no
yes
$CW = 0?$
yes
failed
succeeded

(a) CSMA-CA

(b) Slotted CSMA-CA

Figure 2.1: Flow Charts: CSMA-CA

communicate within guaranteed time slots. Fur this purpose all devices have to be time synchronized and need to know at which times they may receive or transmit data. To avoid disturbance of other communications, the time slots need to be scheduled over the whole network, such that only a single device transmits on the same time and channel within its interfering area. Of course, the same time and channel can be spatially reused as long as the distances are large enough. Such scheduling can be achieved by centralized or distributed algorithms. Centralized algorithms may utilize all necessary information of all devices participating in the network, whereas distributed algorithms may only use information of their local neighborhood or might use any preconfigured scheduling scheme. Although the concept of TDMA is simple to understand, achieving an optimal schedule is very complex and there exist many different strategies [SVV15]. Especially achieving an interference free time slot schedule, while attaining a minimum overall traffic rate is known to be NP-complete [ET90]. Also the problem of finding a schedule which has minimum length is NP-complete as well [GOW07].

2.1.4 Channel Assignment

Multiple non-interfering channels increase the network capacity by allowing parallel communication of different neighboring devices. With channel assignment mechanisms conflicting links are resolved by assigning them different channels. Channels may be spatially reused assuring a distance large enough to eliminate interference. Kyasanur et al. [KSCV06] classify therefore channel assignment protocols as static, dynamic or hybrid. With the simpler static assignment, devices may not change their channel and therefor cannot react to radio disturbances unless a device utilizes multiple radios. Dynamic assignment allows to adapt to disturbances as for example interference or high neighboring traffic. As stated in [PD11], dynamic assignments are promising to improve the system capacity, but increase the design complexity. Hybrid schemes imply multiple radios, where some use static and other dynamic channel assignment.

2.2 Routing

In multi-hop networks, where devices can not communicate directly with every other device, messages have to be forwarded by intermediate devices. The selection of such devices has to be determined for specific paths by a routing protocol. For wireless networks not only the shortest path, but also the link quality should be considered as routing metric [DCACM03]. Therefor different more applicable metrics have been proposed in the literature. A comprehensive summary is given in [PD11]. These metrics additionally consider the probability of successful transmission over a link, the expected duration of a transmission, channel diversity, interference and channel switching costs. Also several routing protocols were classified by [PD11], including the following methods:

- **Route discovery:** Uses flooding messages, which are forwarded by multiple nodes to discover the shortest path. The resulting route may then be stored in a routing table.
- **Opportunistic:** Hop-by-hop routing, where each node tries forwarding messages to far away nodes and only considers closer nodes if the transmission was not successful.
- **Multi-path:** Utilizes multiple routes to a single destination, which allows load balancing.
- **Geographic:** Hop-by-hop routing, which utilizes the device locations to select the node which is closest to the destination.
- **Hierarchical:** The network is partitioned into clusters, which are used for routing.
- **Multi-radio and multi-channel:** Considers interference among paths and selects optimal channel with regard to switching costs.

For wireless mesh networks, proactive hop-by-hop routing schemes are most suitable using efficient routing tables which indicate the next hop [YWK05]. Especially geographic routing is promising for large scale wireless networks, as it enables routing without the need of information about far away nodes [LJDC$^+$00].

2.2.1 Geographic Routing

As summarized by [PD11], geographic routing protocols utilize the position or the relative direction of neighboring devices to select the next hop node which is closer to the destination. According to [MMW09, Chapter 4.3.1], the basic methods to select the next-hop neighbor are

- **Nearest with Forwarding Progress (NFP) [HL86]:** Selects the node, which is nearest to the current node but closer to the destination.
- **Most Forwarding progress within Radius (MFR) [TK84]:** Uses the projected positions on the straight line between source and destination of the neighbors and selects the node which is closest to the destination.
- **Compass Routing [KSU99]:** Choses the neighbor N which creates the smallest angle $\angle NCD$ at any current node C to the destination D.
- **Greedy Forwarding [Fin87]:** Selects the neighbor which has the smallest distance to the destination.

Although the greedy hop-by-hop forwarding is efficient, it may lead to routes which end at a node without another neighbor in direction of the destination and thus a recovery scheme would be necessary for further routing [LBB05]. Such a scheme is included in the Greedy Perimeter Stateless Routing (GPSR) [KK00], which uses the right-hand rule to route around routing holes. As discussed in [FGG06] such a hole is caused by stuck nodes. A stuck node p only has neighbors which are at most as close to the destination as itself and the angle $\angle upv$ spanned by two neighbors u, v is greater than 120°. Thus a node can have maximal three stuck angles. It is shown that, if all such angles are less or equal to 120°, a node is not a stuck node. Thus a network topology has no routing holes if this holds for all nodes.

To learn about the neighbor location information, nodes may periodically broadcast beacon messages as implemented in GPSR. There also exist beacon-less algorithms as discussed in [WT10], which allow message forwarding without the need of previously collected neighbor information. As the minimum distance to the destination without regarding the wireless link qualities is not a sufficient metric to select the next-hop in wireless networks, Lee et al. [LBB05] propose a metric called normalized advance (NADV), which additionally considers an arbitrary cost function. NADV is defined as the factor of the difference in the distance to the destination divided by the link cost $NADV = \frac{D_S - D_n}{cost_n}$. The cost function can be represented for example by the link delay, packet error rate or power consumption.

2.3 Joint Design Strategies

As discussed in [PD11], the different design problems routing, time scheduling and channel assignment strongly depend on each other. Joint designs consider multiple interdependent design problems with different optimization targets. Many of the surveyed strategies focus on two problems like:

- **Power control and scheduling:** With power control, the number of devices affected by interference can be reduced by using less transmit power, but this also reduces the number of reachable neighbors. Such strategies focus on minimizing the power consumption while maximizing the throughput.

- **Routing and scheduling:** Links have to be scheduled to fulfill all the given routes. If any necessary link cannot be scheduled at some point, the traffic may have to be routed using a different path. Therefor many approaches create routes and schedule links in an iterative fashion.

- **Routing and power control:** This approach focuses on routes with minimal power consumption while assuring guaranteed delays.

- **Routing and channel assignment:** Such strategies try to find routes with a wide range of available channels.
- **Scheduling and channel assignment:** Combining the benefits of TDMA and Channel assignment to increase the network capacity.

There also exist some strategies which combine two of these targets, as "routing, scheduling and channel assignment" or "routing, scheduling and power control".

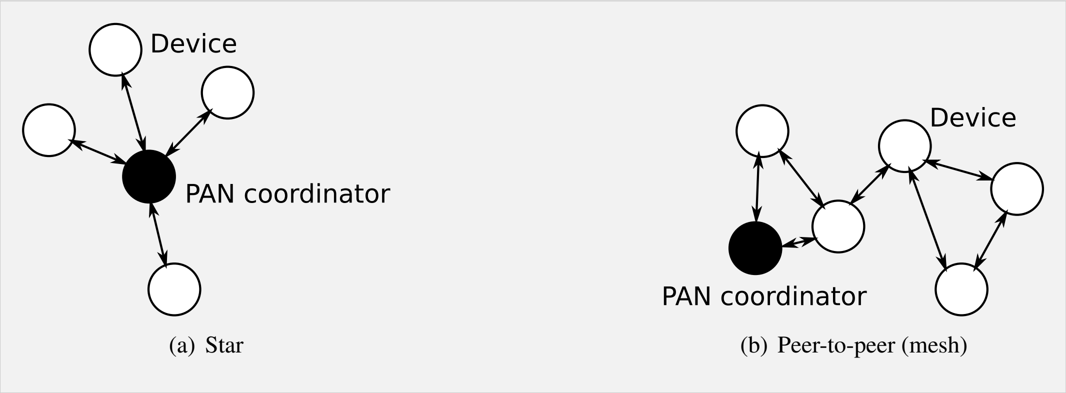
2.4 Current Standards

The designation "wireless network" is often related to wireless local area networks (WLAN) which are covered by the IEEE 802.11 standards. Such networks provide internet access with high bandwidth and originally are not operated in a mesh topology. Many work has been done to realize wireless mesh network using the 802.11 standard. Whereas this work focuses on wireless sensor networks utilizing a mesh topology. These networks contrastingly feature lower bandwidth and throughput, with smaller range and may have more critical energy constraints. Such applications are covered by the IEEE 802.15 standards.

The IEEE 802.15.4 Standard for Low-Rate Wireless Personal Area Networks (LR-WPANs) [IEE11] defines the physical layer and the medium access control sublayer as part of the data-link layer. It also addresses network topologies to interconnect the devices in such a network. Its scope of application are small, cost-effective, power-efficient devices with low data-rate to build up WPANs with small distances between the network nodes. First published in 2006, IEEE 802.15.4 compatible transceivers are utilized in many devices available on the market today i.e. included in a system on chip with a microcontroller. Furthermore there are several standards based on IEEE 802.15.4 which define the higher OSI layers [Zim80] up to the application layer. Well-known examples are the ZigBee standard, 6LoWPAN, ISA100.11a [ISA09] or the WirelessHART [Fou14b] standard which is common in industrial applications.

2.4.1 IEEE 802.15.4

An IEEE 802.15.4 LR-WPAN consists of at least two devices and must include one personal area network (PAN) coordinator, which manages the device addresses. The network devices communicate either within a star or peer-to-peer (mesh) topology. As shown in Figure 2.2, in a star topology all devices communicate directly with the PAN coordinator and no other device. However, devices in a mesh network may communicate with any device which is in its range and may have no direct connection to the coordinator. The coordinator may manage all communication e.g. by initiating, terminating or routing, but this has to be defined by upper



(a) Star

(b) Peer-to-peer (mesh)

Figure 2.2: IEEE 802.15.4 LR-WPAN Topologies

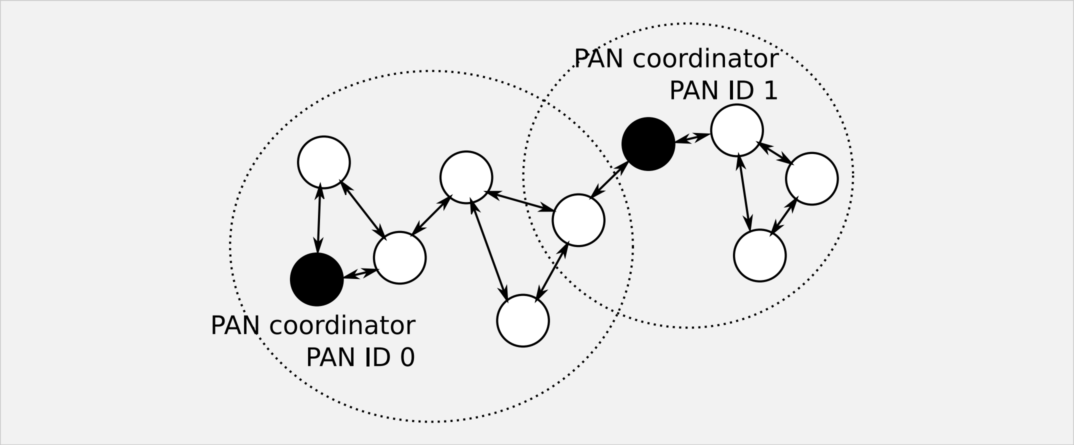


Figure 2.3: Network of adjacent networks

layers. The star topology is more suitable for smaller networks, whereas peer-to-peer are promising to construct larger mesh networks.

Devices have a unique 64-bit extended or 16-bit short address (allocated by the PAN coordinator) to dedicate the communication target. Short addresses are used in combination with a 16-bit PAN identifier (PAN ID), which enable devices to distinguish between independent adjacent networks. It also allows to build up a network from multiple mesh newtworks with different PAN IDs where some devices are related to more than one network as illustrated in Figure 2.3.

Physical Layer

The physical layer (PHY) provides wireless communication on one out of 16 non-overlapping channels at the same time when using the 2450 MHz band. This allows independent communication of multiple devices in parallel as well as frequency hopping to overcome temporary

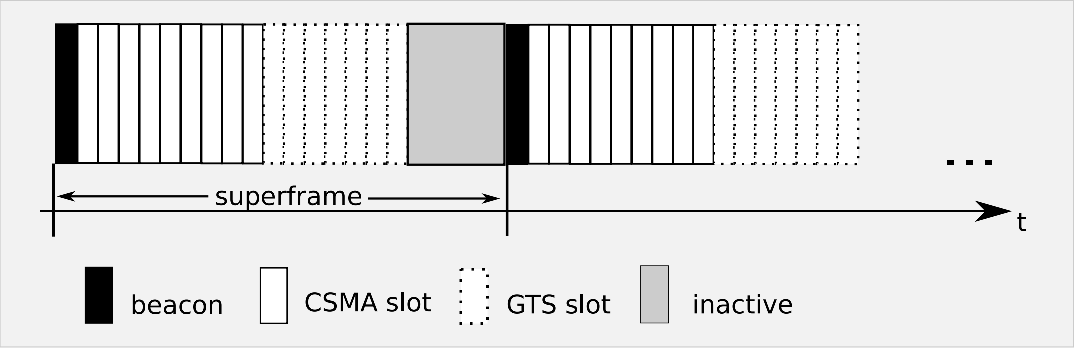


Figure 2.4: IEEE 802.15.4 MAC Superframe

radio disturbance. Wireless communication includes transmitting and receiving of PHY protocol data units (PPDUs) which include the data received by the upper layer. This payload contains up to 127 bytes.

Medium Access Control

The medium may be accessed either purely contention based using the CSMA-CA mechanism described in Section 2.1.1 or divided into time slots by using a superframe structure defined in the standard, which utilizes slotted CSMA and guaranteed time slots. However, the superframe structure is only considered for single-hop star networks.

As illustrated by Figure 2.4, a superframe consists of 16 time slots of the same length, whereas the first slot is reserved for a network beacon. The beacon frame is sent by a coordinator. It includes the PAN ID as well as the structure of the superframe and a timestamp which is used for synchronization by other devices. However, synchronization of devices which are out of the coordinator's range in a mesh network is not covered by the standard. Access within a time slot is contention based in general, utilizing slotted-CSMA. But the coordinator may allocate up to seven guaranteed time slots (GTS), which ensure exclusive communication for specific devices. The PAN coordinator's superframe may include some additional time between the last slot and the next beacon where. Within this time the channel stays inactive allowing other coordinators to send beacons for their superframe.

To improve reliability of the communication, acknowledgments may be used, which are sent by the receiving device to confirm successful reception of a MAC frame. Furthermore, bit errors may be detected by using a cyclic redundancy check (CRC). If an error occurred, the receiver will not acknowledge the frame. Both methods combined allow the transmitting device to retransmit the frame until its correct reception is assured.

PAN Maintenance

When a PAN coordinator wants to start a new PAN, it may first perform an energy detection (ED) scan to select the channel with minimum energy to operate on. Devices which want to associate with a PAN first need to actively or passively scan for existing PANs. In an active scan, the device sends a beacon request command and gathers every unique received beacon frame informations. Using the passive scan, a device waits for periodically sent beacon frames. After selecting a PAN to associate with, the device sends an associate request command to the coordinator and expects an association response command including the new address on success or indicating that the association has failed. Such a response may be send using the indirect transmission, where the coordinator indicates within the beacon frame that it is holding pending data for a device. The device then sends a data request command to the coordinator and awaits its response. Indirect transmission allows devices to reduce power consumption by deactivating the receiver when no message is expected.

Interframe Spacing (IFS)

When using time slotted communication, transmitting devices directly start the transmission of the frame at the slot boundary. If acknowledgements are used, the receiving device will transmit the acknowledgement frame after a short delay t_{ACK}. To allow the receiving device to further process the received frame, following frames may only be transmitted after a short delay of either $SIFS$ for short or $LIFS$ for large frames. As illustrated in Figure 2.5, the transmitting device has to assure that this delay fits within the current time slot. The time values of the delays depend on the settings of the PHY and therefore are given in symbols. With a center frequency of 2450 MHz, the duration of a symbol is 16 µs, which leads to IFS durations of $SIFS = 12 \hat{=} 192\,\mu s$, $LIFS = 40 \hat{=} 640\,\mu s$ and $t_{ACK} = SIFS$.

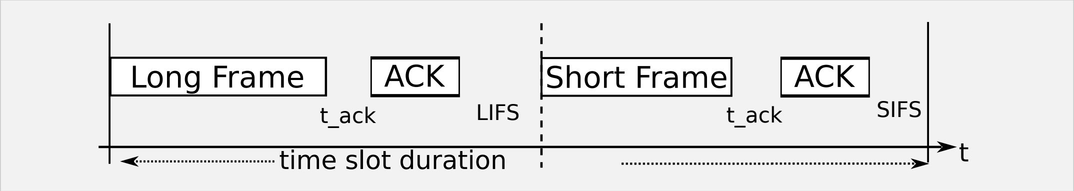


Figure 2.5: Interframe spacing for large and small frames

2.4.2 IEEE 802.15.4e

The target of the IEEE 802.15.4e amendment [IEE12] is to achieve a more robust and deterministic communication with deterministic latency for industrial applications. This is done by extending the behavior of the medium access control. The standard introduces several MAC behavior modes for different applications which are summarized in this section.

Low Latency Deterministic Networks (LLDN)

In an LLDN, the sensor data shall be transmitted within a minimum latency of 10 ms, which is equivalent to the duration of the superframe. LLDNs only utilize the star topology and according to the standard such a network may not have much more than 100 devices. By running multiple networks on distinct channels the amount of devices may be increased to a few hundred. For these reasons LLDNs are not further considered in this work, as it focuses on large scale networks with high traffic demands.

Deterministic and Synchronous Multi-Channel Extension (DSME)

The DSME enhances the original standard by increasing the amount of guaranteed time slots (GTS) in a superframe. This is achieved by using multiple channels in parallel within the contention free period. Furthermore GTS in DSME support multihop communication for scalable mesh networks using distributed beacon scheduling and slot allocation. Details on the DSME superframe structure and GTS management protocol are further described in Section 3.1.

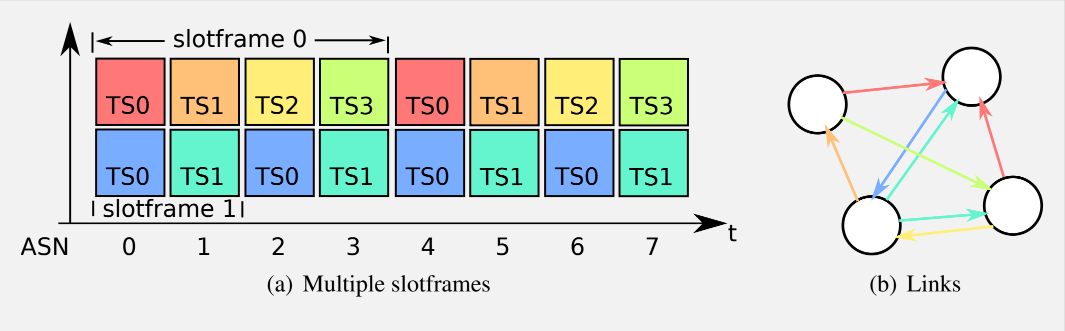
Timeslotted channel hopping (TSCH)

With TSCH collisions and radio interference shall be reduced through time synchronized communication and channel hopping to build reliable, scalable and flexible networks. Instead of a superframe structure where network beacons are used for synchronization, TSCH uses a slotframe concept without predefined fixed beacon slots. The devices determine the slot boundaries by taking advantage of a shared system time. Communication in a time slot may be contention based when there are more than one link using the same channel or it is guaranteed when there is only one link. A link represents an allocated directed communication between two devices. Multiple slotframes with different amounts of time slots may be used in parallel as shown in the example link schedule in Figure 2.6(a). Figure 2.6(b) shows the corresponding device linking.

The channel CH to be used for communication in a time slot is determined given a hopping sequence list (hopChList), the absolute slot number (ASN) and a channel offset (ChOff)

$$CH = hopChList[(ASN + chOff)\ \%\ \mathrm{size}(hopChList)].$$

The ASN gets incremented with each time slot and has to be known by every device.



(a) Multiple slotframes

(b) Links

Figure 2.6: IEEE 802.15.4e TSCH multiple slotframes example

To build up a TSCH PAN, the PAN coordinator initially advertises its network by sending enhanced beacons which contain the information for time synchronizing, channel hopping, the time slot durations and the initial slotframes with included links. Other coordinators may later also advertise the network periodically by sending enhanced beacons. New devices join the network by either scanning for such a beacon or sending a beacon request. After receiving the information, the device synchronizes with the network and may send an association request to retrieve a short address.

Synchronization is further maintained for each device by using one or more of its neighbors as time source. A device may synchronize with a neighbor when exchanging any frame by utilizing the frame arrival time or the time correction information included in each acknowledgement. To ensure synchronization every device has to communicate with their time sources periodically. For this purpose keep alive messages have to be send to each time source when no communication occurred within a specific time. An acknowledgement may be either positive (ACK), indicating that the reception of the frame was successful and it is being further processed by the device or negative (NACK), in case that the device has received the frame correctly but is not able to process it. When the frame could not be received without errors no acknowledgement will be send. In those two cases the transmitting device will retransmit the frame in the next available link to the receiver until a maximum amount of retries is reached, then the upper layer will be informed.

The timing of a transmission and the corresponding reception in a time slot is shown in Figure 2.7. The transmitting device may perform a clear channel assesment (CCA) before transmitting the frame in case of a shared link. Offsets before the expected reception of a frame or an acknowledgement allow time deviation between the devices. The receiving device determines the time correction by calculating the difference between the actual and the expected time of reception. A device then adapts its own time by computing the average drift to all its time sources or sends back the time correction included in the acknowledgement.

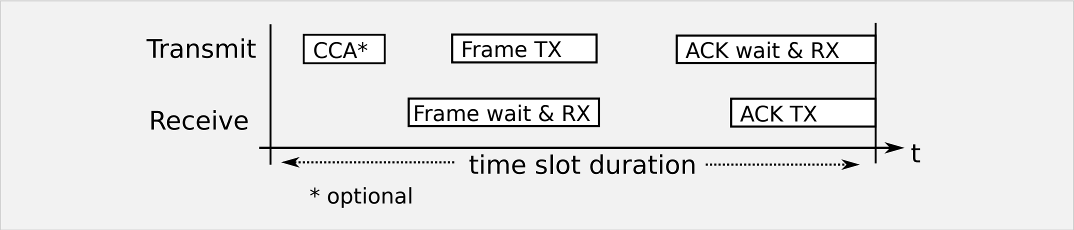


Figure 2.7: IEEE 802.15.4e TSCH time slot

2.4.3 IEEE 802.15.4k

The IEEE 802.15.4k amendment [IEE13] focuses on low energy critical infrastructure monitoring (LECIM) networks and specifies an improved physical layer with corresponding adaptations in the MAC layer. Critical infrastructure systems are such, which affect the public health and safety as well as general or economic security of a nation in case of system failure. An electrical power plant fits this definition quite well. LEICM networks are characterized as commissioned minimal infrastructure networks which utilize the star topology with devices specifically preconfigured for their assembly. Receivers with higher sensitivity and interference robustness allow communication over long range. Devices usually have a limited power supply and less performance than the PAN coordinator. Also, the data flow from devices to the coordinator is predominant to the opposite direction.

The standard adds two different improvements to the original physical layer. One is a direct sequence spread spectrum (DSSS) which achieves a higher processing gain by using a decreased data rate and also allows parallel communication on the same physical channel using different gold codes. The other one is using a frequency shift keying (FSK) method which also results in a higher receiver sensitivity and provides more channels with lower data rate. Thereby, the possible receiver sensitivity allows path losses of 120 dB, which is a great improvement compared with the original standard allowing about 85 dB path loss. Using the channel model from [IEE06, p. 265], the amendment allows communication within distances up to approximately 580 m, which is more then 10 times higher than the original (about 50 m). Both methods increase channel diversity and thus allow more parallel communications and reduce interference with neighboring networks. As the data rate is very low using this methods, fragmentation of packets is introduced to reduce the medium access duration and the overhead of potential retransmissions. Using priority channel access allocation within the CSMA period allows devices to compete less when they need to transmit a frame with higher priority.

For further range extension the time-slot relaying based link extension (TRLE) may be utilized. TRLE relay devices act as coordinators and forward frames in a star topology from the PAN coordinator to one or more devices associated to the TRLE relay or another TRLE relay and from other devices to the PAN coordinator. The forwarding is included within the

MAC sublayer by performing a frame filtering to decide if a frame has to be relayed. Therefore such link extensions may be deployed within networks without further configuration of other devices which are not TRLE-enabled. Frames get relayed after a given delay which is a multiple of the superframe duration or within an previously allocated time slot.

2.4.4 WirelessHART

The WirelessHART standard [fS14] brings wireless communication to the common HART protocol [Fou14a] which is widespread in the industry for process control. With its first release in 1988, the protocol was based on a pure analog data transfer and has been adapted to digital communication later and latest also to wireless communication. Backwards compatibility has been achieved in each revision, which creates the need for a wireless adapter for interoperability with older devices. This section gives a short overview of a WirelessHART network, more comprehensive summaries can be found in [CMN10, KHPD08] and in [SHM$^+$08] which also provides some aspects on a basic implementation.

A WirelessHART network utilizes an IEEE 802.15.4 WPAN with a mesh topology. As illustrated in Figure 2.8, the networking devices are managed centrally by a network manager, which is either connected to or included within a gateway together with the security manager. The gateway connects the application host network with the wireless network. The network manager handles association of all new devices, provides routing information, assigns time synchronization sources and schedules the necessary links. The security manager maintains joining keys to authenticate new devices and provides them unique session keys for each end-to-end connection to achieve confidentiality and data integrity. Data is routed via all wireless devices. Routing devices are only used for data forwarding, field devices are either directly equipped with a wireless transceiver or are connected to an adapter. Handhelds are mobile monitoring devices which allow direct access to the network in the field.

The standard defines all layers above the physical layer. The data link layer consists of the MAC and the logical link control (LLC) sublayers, whereas the MAC layer uses a TDMA scheme homologous with TSCH using time slots with a duration of 10 ms. To improve the reliability of TSCH, channels with bad characteristics may be blacklisted for the channel hopping pattern. The LLC handles device addressing, flow control using priorities and performs message authentication as well as error detection. Preferring messages with high priority allow the network manager to maintain the network even at high traffic rates. End-to-end sessions are handled within the transport layer given the source and destination address and the identifier of the route to be used. Redundant routes are created based on the sets of neighboring devices which each device provides to the network manager. Routes may be optimized using battery status and current traffic of devices. The HART protocol resides in

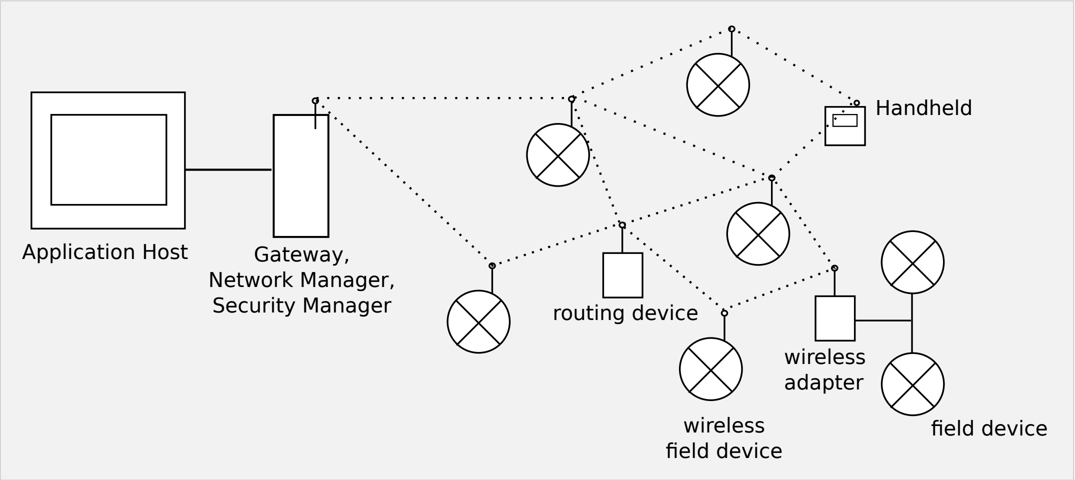


Figure 2.8: WirelessHART network

the application layer. This allows the application to execute HART commands and respond or send a status report.

WirelessHART networks usually consist of less than thousand devices and because of the centralized management, less complex routing schemes are more suitable [KHPD08]. Implementing WirelessHART is a complex task, especially when considering a fully functional network manager. Even though it is an open standard, many details are left open or may be covered by the specifications, which are not open. Especially the fundamental implementation of the network manager and how it configures the network is unspecified. However, there exist some university projects on implementing WirelessHART [Ska12, Kon10, Sí11].

2.5 Related Work

As the IEEE 802.15.4e standard is a recent amendment to the original (published in 2006), there are only a few considerations to be found in the literature yet. Some analyze the performance considering specific scenarios and other propose further enhancements.

2.5.1 DSME

In [WJ12] the performances of a 1-hop star and a 5x5 multi-hop network were evaluated. The throughput and energy consumption of DSME with enabled and disabled CAP reduction are compared to slotted CSMA. Their results show that DSME outperforms slotted CSMA in the multi-hop network with a 12 times higher throughput. Within a star topology the throughput is comparable for a small node count, but higher for larger networks with more than 40 nodes using DSME. The energy consumption increases exponentially with the node count using slotted CSMA, whereas it remains constant with DSME. The same authors analyzed the performance of DSME under the influence of WLAN interference in [LJ12]. It is shown that DSME, compared to slotted CSMA, experiences lower frame error rates and may achieve higher overall throughput with increasing WLAN traffic as well as signal strength.

Other work has been done to further improve the DSME standard. An analysis and enhancement of the beacon scheduling is discussed in [HN14] and the fast association mechanism was enhanced in [XXS+13]. Hwang and Nam [HN14] show that their beacon allocation pattern using the proposed enhanced protocol reaches a success ratio of almost 100 % in a dense topology. The enhanced fast association mode proposed in [XXS+13] drastically reduces the number of retransmission by a factor of 500 compared to the original for a scenario where hundreds of nodes associate with a single PAN coordinator.

2.5.2 TSCH

Accetura and Piro [AP14] contribute to the recent Internet Engineering Task Force (IETF) working ground named "IPv6 over the TSCH mode of IEEE 802.15.4e" (6TiSCH), which standardizes the use of IPv6 with TSCH and specifies the missing scheduling techniques of the TSCH standard. In their paper they review the backgrounds of 6TiSCH and highlight their contribution of an optimal and secure scheduling protocol.

A recent work of Vogli et. al. [VRGB15] proposes a fast join and synchronization schema to overcome the time and energy consuming joining phase of TSCH. Besides the implementation and experimental validation of the mechanisms, also the average joining time depending on the node density, communication reliability, and beacon transmission frequency was modeled analytically.

2.6 Queueing Models

Communication systems usually use queues to temporarily buffer messages before they are being transmitted or forwarded to the next higher layer after reception on a lower layer. They are necessary because the layers may process the messages with a different speed or are blocked for some time. In this section some queueing fundamentals and two selected models which are of interest for this work are consolidated. A detailed explanation of the fundamentals of several queueing models can be found in [NS08]. In general, queueing systems are modeled with an input, a queue and one or more servers in parallel or series, which are processing the messages and forwarding them to an output. The input has an arrival rate λ, the server a service rate μ and the output a departure rate γ. The utilization of the system is then given by $\rho = \frac{\lambda}{\mu}$. Different models are classified by their arrival and service pattern, number of servers and the queue capacity. This is reflected in the Kendall notation A/B/X/Y/Z, where A is the arrival, B the service pattern and X the amount of servers. Y is the queue limit and if not present defaults to infinity and the serving discipline Z defaults to first-come-first-served.

2.6.1 M/M/1/K

The M/M/1/K model presumes that the inter-arrival time between two messages is distributed exponentially. The number of arriving messages within a time interval is then described by a Poisson process, which holds the property of being memoryless (also called Markovian). In this model only one server is considered, which has a service rate with the same distribution as the arrival. To describe realistic systems with limited resources, the queue holds a maximum of K messages.

The probability that such a system is saturated and further messages are blocked by the queue is called saturation or blocking probability $P_K = P_b$. As shown in [NS08, p. 120] this probability can be determined by

$$P_b = \frac{(1-\rho)\,\rho^K}{1-\rho^{K+1}}.$$

P_b has a singularity at $\lambda = \mu$, which is a removable discontinuity given by the limit

$$\lim_{\lambda \to \mu} P_b = \frac{1}{K+1}.$$

Because some messages might not be stored in the queue, an effective arrival rate $\lambda' = \lambda\,(1 - P_K)$ is used to analyze the system performance. If this is the only point where loss occures, the departure rate of the output is $\gamma = \lambda'$. Such a system is illustrated in Figure 2.9.

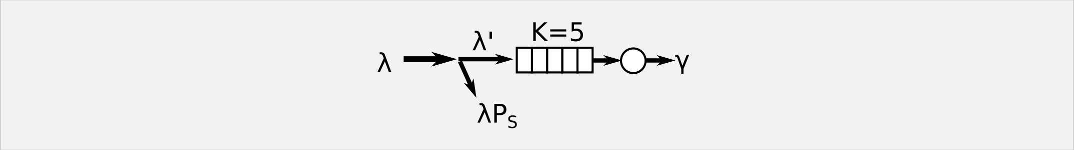


Figure 2.9: M/M/1/S Queue model

An important influence on the system performance is the average time a messages stays in the system

$$\begin{aligned} T_S &= \frac{1}{\lambda - \mu} - \frac{S\rho^{S+1}}{\lambda - \mu\rho^{S+1}} \\ \lim_{\lambda \to \mu} T_S &= \frac{S+1}{2\mu} \end{aligned}$$

and the average time spent in the queue

$$\begin{aligned} T_Q &= \frac{\rho}{\lambda - \mu} - \frac{S\rho^{S+1}}{\lambda - \mu\rho^{S+1}} \\ \lim_{\lambda \to \mu} T_Q &= \frac{S-1}{2\mu} \end{aligned}$$

These equations are also undefined for $\lambda = \mu$ and thus their limits have to be considered.

Figure 2.10 exemplarily shows the influence of ρ and K on the average time T_S a frame stays in the system and the probability P_b that an arriving frame gets blocked by a full queue. Using a larger queue, lower blocking probabilities are achieved, but on the contrary the frames reside longer in the queue. In contrast to that, increasing the service rate μ improves both, the time delay and the saturation probability decrease.

2.6.2 M/G/1/K and M/D/1/K

The M/D/1/K queue model is a special case of the M/G/1/K model, which considers general service time distribution. Such a system is called semi-Markovian, as in contrast to the M/M/1/K model the service time depends on the past and thus the process is not memoryless. An M/D/1/K system uses a deterministic service time and therefore is commonly considered to analyze communication systems utilizing TDMA according to [NS08]. Unfortunately only M/G/1 and also shortly M/D/1 systems with unlimited capacity are captured in that book. The analysis of such limited systems are more complex and therefore were often inspected using simulations or computational models. However, there exist some analytical solutions for M/G/1/K and M/D/1/K queues. One less complex approximation for the blocking probability

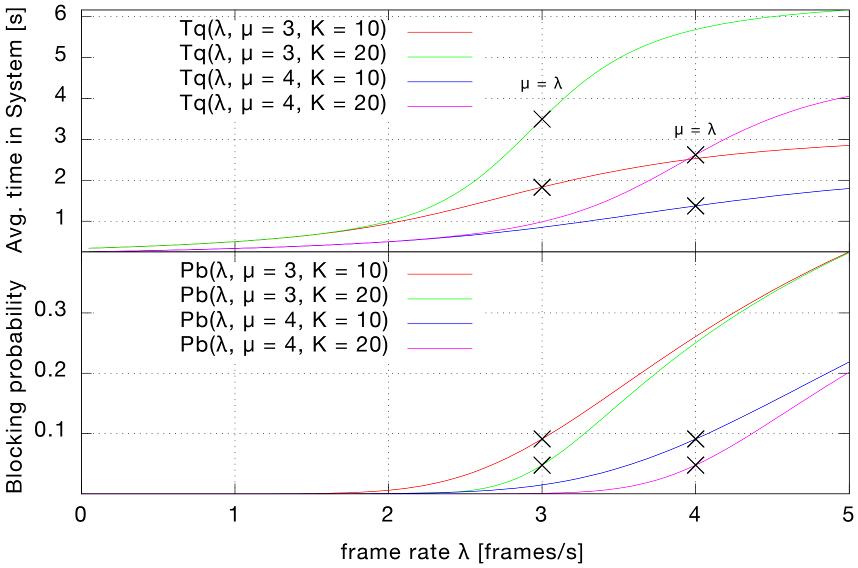


Figure 2.10: M/M/1/K Queue delay and blocking probability

P_K with general service time distribution was determined in [Mac11]

$$P_K = \frac{\rho^{\left(\frac{\Delta+2K}{2+\Delta}\right)}(\rho-1)}{\rho^{2\frac{\Delta+K+1}{2+\Delta}}-1}$$
$$\Delta = \sqrt{\rho e^{-s^2}}s^2 - \sqrt{\rho e^{-s^2}},$$

where s^2 is the squared coefficient of the service process variation.

A solution for the special case M/D/1/K queue to determine the blocking probability P_B is given in [Seo14]

$$P_B = \frac{(1-\rho)E_K}{1-\rho E_K}$$
$$E_K = 1-(1-\rho)\sum_{j=0}^{K}\frac{(-1)^j\rho^j(K-j)^j e^{\rho(K-j)}}{j!}.$$

Both of these examples are still more complex compared to the M/M/1/K model. But it is shown in [Mac11] that for bigger queues ($K > 4$) the average fill rates of both models are close to each other. Thus the M/M/1/K model also can be considered to estimate the performance of an TDMA system. The comparison of the blocking probabilities of the M/M/1/K and M/D/1/K model in Figure 2.11 shows however, that the probabilities differ significantly for the important setting, where $0.5\lambda \leq \mu \leq \lambda$. The Markovian service rate evinces a higher blocking probability compared to the model using a deterministic rate. The difference is bigger

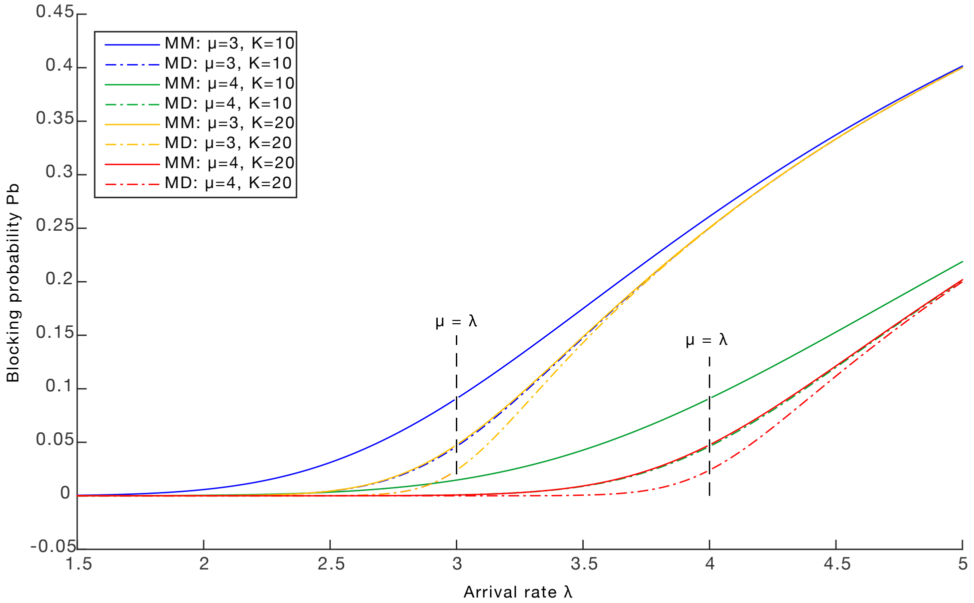


Figure 2.11: Blocking probabilities M/M/1/K and M/D/1/K

for a smaller queue limit. Also it is notable, that the probability curves of the M/D/1/K model seems to be very close to curves of the M/M/1/K model with a queue limit twice as big.

The average time spent in an M/D/1/K system is also derived from the M/G/1/K model by [Seo14] with

$$T_S = \frac{1}{\lambda}\left(\frac{1}{1-E_K}\right)\left(\sum_{j=1}^{K} E_j\right) + \frac{K}{\lambda}\left(1 - \frac{1}{1-E_K}\right)\rho.$$

Figure 2.12 shows that the resulting timings are comparable to the M/M/1/K model shown in Figure 2.10 for the given parameters. However, using the same utilization and queue size, the time spent in an M/D/1/K system are a little higher for $\rho > 1$, while the blocking probability is smaller.

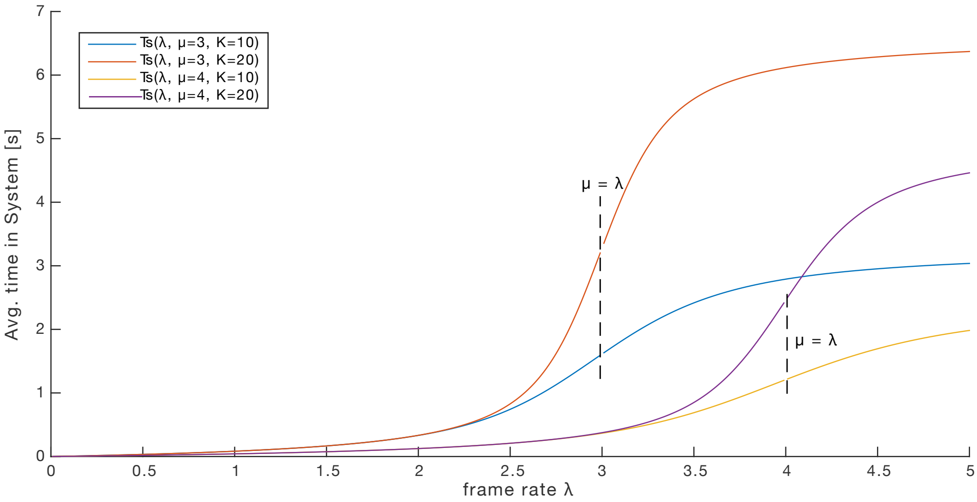


Figure 2.12: M/D/1/K Queue Delay

Chapter 3

Deterministic Synchronous Multi-Channel Extension (DSME)

DSME is a promising TDMA scheme for large scale networks, because it manages channel assignment and time slot scheduling in a distributed fashion using local utilization information. Furthermore, DSME was designed for robust and scalable industrial applications, which require a high reliability and deterministic latency in an efficient manner while sustaining flexibility for failure scenarios. DSME provides a downwards compatible superframe structure as well as a complete protocol for the decentralized management of the time slot scheduling and channel assignment. Therefore, DSME is chosen to be further analyzed for the use in the network considered within this work. This chapter includes a more comprehensive survey of the DSME standard, which is followed by a summary of related work considering performance analysis and enhancements of DSME. The last section discusses some open issues related to the protocol.

3.1 DSME Definition by the IEEE 802.15.4e Standard

The IEEE 802.15.4e amendment introduces a multi-channel contention free period as well as multi-superframe structure. These structures and the management of beacons and guaranteed time slots (GTS) provided by the standard are described in this section.

3.1.1 Superframe Structure

The structure of the DSME superframe is derived from the original structure of the IEEE 802.15.4 standard discussed in Section 2.4.1. As shown in Figure 3.1, a DSME superframe starts with a network beacon from a coordinator followed by a contention access period (CAP) allowing all devices to communicate on a common channel using slotted-CSMA. After that, devices may communicate on different channels within guaranteed time slots (GTS).

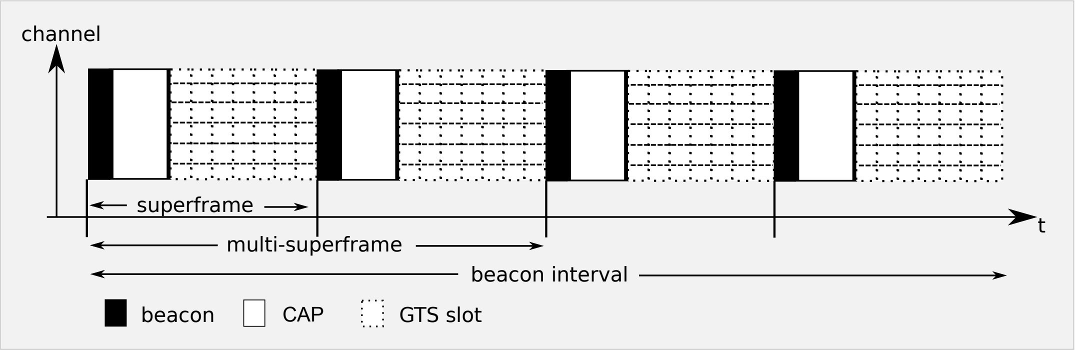


Figure 3.1: IEEE 802.15.4e DSME superframe

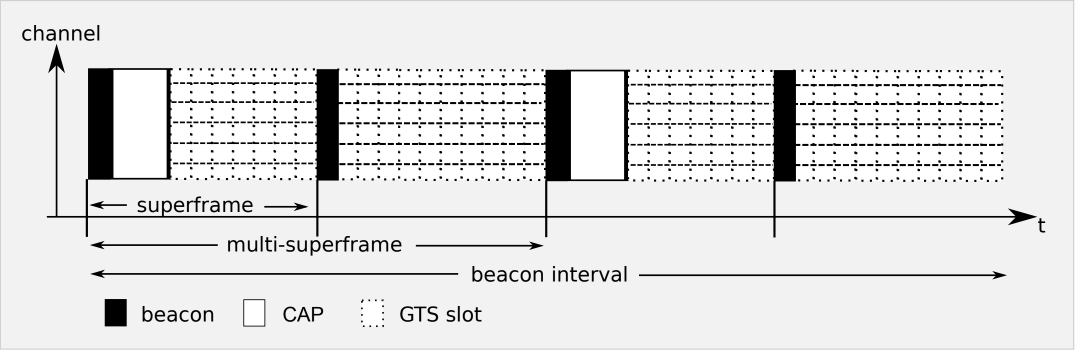


Figure 3.2: IEEE 802.15.4e DSME CAP reduction

Furthermore, a multi-superframe structure may be used where multiple coordinators send a beacon for different superframes. With the CAP reduction mode, it is possible to have only one CAP within such a structure, which takes place in the first superframe. The remaining superframes then provide 8 additional GTS in that period. This mechanism is illustrated in Figure 3.2.

The duration of a superframe SD, multi-superframe MD and the beacon interval BI is specified in orders of two and also depends on the symbol duration of the PHY, which is 16 µs when using a center frequency of 2450 MHz. The orders of superframe duration SO, multi-superframe MO and beacon interval BO are related as follows:

$$0 \leq SO \leq MO \leq BO \leq 14.$$

The number of superframes per multi-superframe N_S and the number of multi-superframes per beacon interval N_M directly result from the differences of the given orders. These values also imply the number of unique beacon slots N_B and GTS per channel N_{GTS}. With a fixed superframe duration base $baseSD$, the durations are represented by the following equations:

$$
\begin{aligned}
SD &= baseSD \cdot 2^{SO} \ \ [symbols] \\
MD &= baseSD \cdot 2^{MO} \ \ [symbols] \\
BI &= baseSD \cdot 2^{BO} \ \ [symbols] \\
baseSD &= slotDuration \cdot numSlots = 60 \cdot 16 \ \ [symbols] \\
N_S &= 2^{MO-SO} \\
N_M &= 2^{BO-MO} \\
N_B &= 2^{BO-SO} \\
N_{GTS} &= N_S \cdot 7
\end{aligned}
$$

3.1.2 Time Slot Management Protocol

A GTS has to be allocated through a GTS Request before devices may use it for transmitting. Given that the slot is available and a device was the first requesting it, the device may communicate within that GTS until it deallocates it. Figure 3.3 shows an exemplary GTS allocation handshake, where device B requests a time slot to communicate to device A. To allocate a slot, the device sends a GTS Request to its counterpart including the communication direction, the preferred superframe and slot identifier. When the slot is free, it replies with a broadcast message with the status "success" and containing a sub block of the slot allocation bitmap (SAB), which includes the newly allocated slots. On reception of this reply, a broadcast is send to notify all neighboring devices about the new allocated slots. Every device receiving one of the broadcasts will then check if the slot allocation is conflicting with its current SAB. If there is no conflict, devices will update their SAB. Otherwise they send a GTS duplicate allocation notification, whereupon the other device will deallocate the corresponding slot. When an allocated slot is not needed any longer it gets deallocated in a similar way, using the request, reply and notify command messages.

3.1.3 Network Coordinators, Device Association and Beacon Management

New devices scan the network for enhanced beacons with a period of the maximum beacon interval on multiple channels. In addition to the superframe information, these beacons include the timestamp and the beacon schedule of the transmitting device. This schedule indicates which beacon slots are allocated by devices in its neighborhood. If a device detects a free slot, it notifies its neighbors about allocating it within the CAP and later sends its own beacon using that slot. When more than one device tries to allocate the same slot, an intermediate

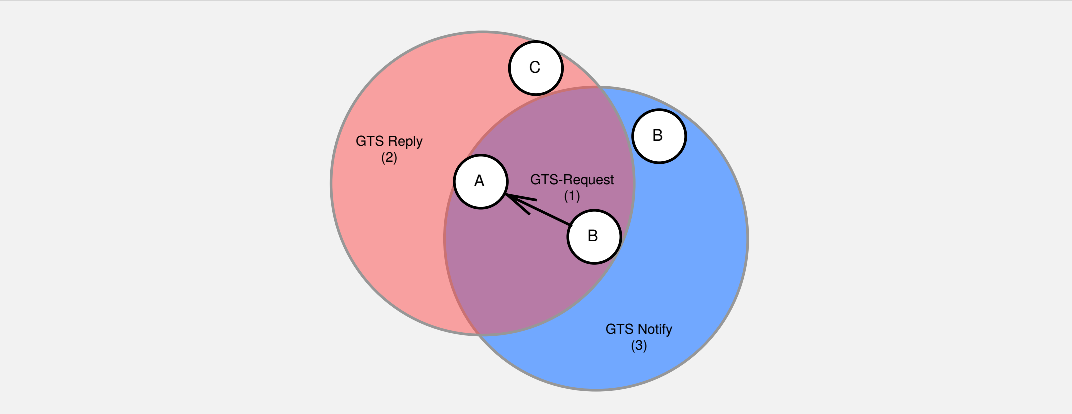
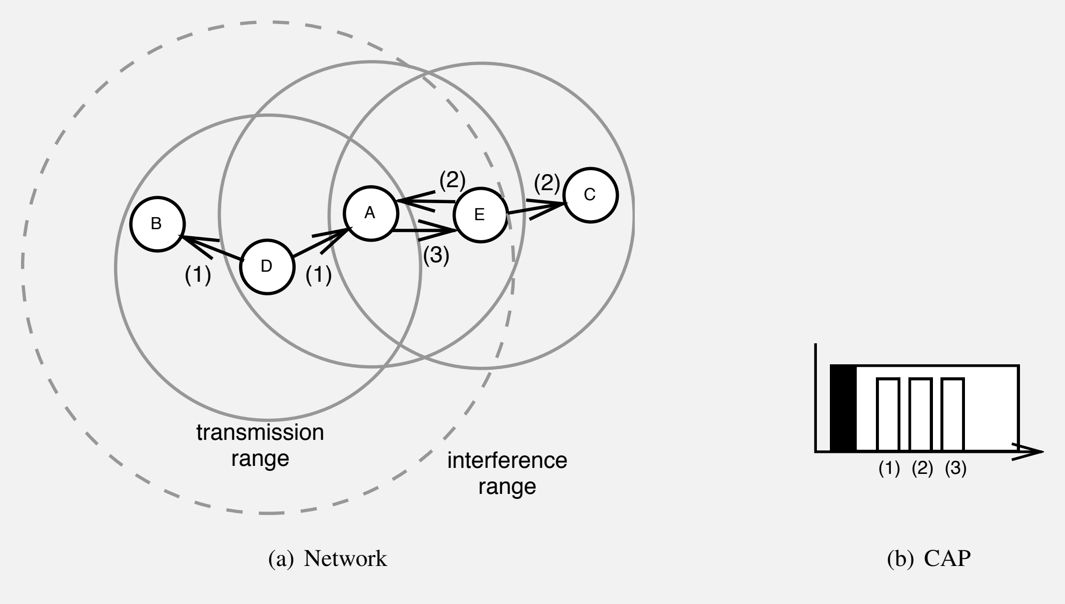


Figure 3.3: IEEE 802.15.4e DSME GTS Allocation handshake

device shall solve the collision by replying to the device which has notified later. Figure 3.4 illustrates an example where node A is the intermediate device, which first receives a "beacon allocation notification command" from node D and also later from device E for the same slot index. Node A then sends a beacon collision notification command to device E.



(a) Network

(b) CAP

Figure 3.4: IEEE 802.15.4e DSME Beacon collision avoidance Example

After associating with a coordinator, a device uses the timestamp information only of that device's beacon frame to synchronize timing with the network. Given the start time of the superframe and the beacon offset of its coordinator, the device updates its timestamp by calculating the difference to the actual receiving time.

3.2 Protocol Definition Issues

Throughout this work some issues related to the protocol were noticed, which could not be clarified regarding the standard documents. This section describes these open issues.

3.2.1 Contention Access Period

The standard does not explicitly define if the contention access period shall use slotted or unslotted CSMA-CA. The superframe Figure is misleading, as the CAP is illustrated as a single unslotted period. The IEEE 802.15.4e amendment explicitly mentions the use of slotted CSMA for LLDN and TSCH in Section "4.5.4.1 CSMA-CA mechanism", but includes no statement to DSME. However, the original standard states in the same section, that beacon-enabled PANs use slotted CSMA. Thus this should also hold for DSME.

3.2.2 Deallocation upon Duplicate Allocation Notification

When a device receives a duplicate allocation notification it has to deallocate the specified slots. For this purpose it will send a deallocation request to the corresponding device and both devices will send a broadcast message to inform their neighborhood about the freed slot. However, the other device which was previously claiming that slot and sent the duplicate notification, still uses it for communicating with a different device. Because these devices most probably have intersecting neighborhoods, some of the neighboring devices later may request exactly this slot, as they marked it as free through the deallocation process. To overcome this situation, the device still occupying the slot could send another notify broadcast after the deallocation. However, this is not considered in the standard.

3.2.3 Non Preferred Slot Allocation

The DSME standard states that the preferred requested slot shall be considered for allocation on the destination device if possible. Otherwise the next free slot may be selected. Since also multiple slots may be requested within a single request command, it may also occur that only a subset of the slots can be allocated as preferred. The slot allocation bitmap included within the request only marks the occupied slots of the source's neighborhood, thus it may happen that the destination allocates a slot which is free in both neighborhoods, but the source device already uses it on a different channel. As the allocation decision is not part of the standard, it is also not defined how to react when receiving such a reply message. The source device could either disapprove the allocation or approve a part of it (in case of multiple slots). If all slots get disapproved because some could not get allocated, it will take more time until a later request will successfully allocate multiple slots. It also may happen that many slots remain unallocated. Whereas when approving a part of the request, the destination device needs to deallocate the unapproved slots and inform its neighborhood. Figure 3.5 illustrates a more complex situation, where the source device A requests two slots from destination C and already occupies slot b with device B. Node C then selects two slots *a* and *b* distinct from the preferred, without knowing that *b* is already occupied. This may occur if A received the reply of B after sending the request to C or if device C selected a free slot of a superframe which was not included in the received SAB. When node B receives the reply broadcast it will directly send a duplicate allocation notification to C, which then immediately deallocates slot *b* and sends a corresponding request to node A. If device A would not validate the address related to the GTS, this would even cause the loss of the link between A and B.

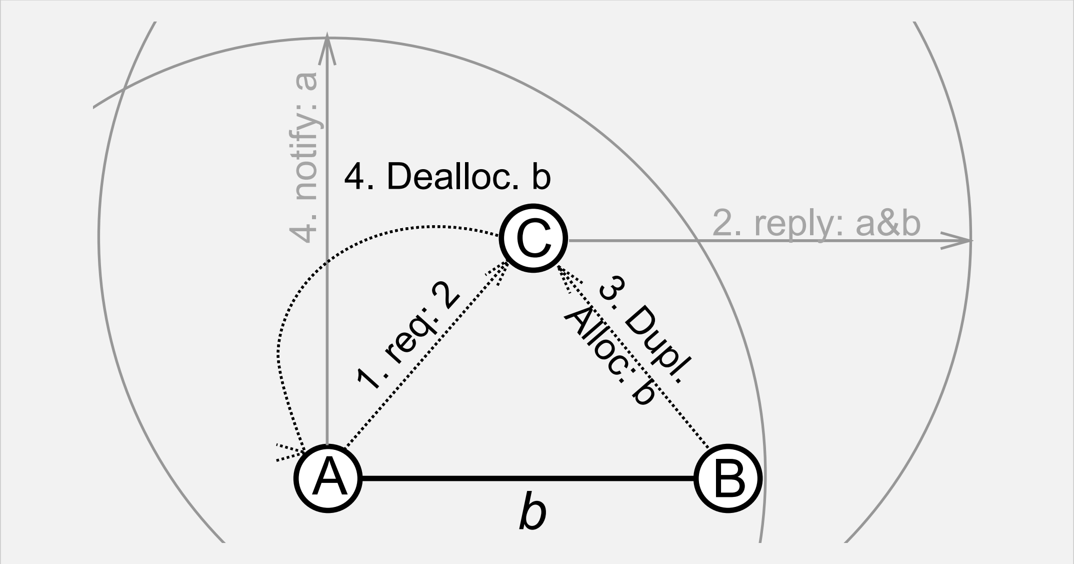


Figure 3.5: DSME: Duplicate allocation on same device

Chapter 4

Geographic Routing in a Concentric Circle Topology

Geographic routing protocols allow forwarding of messages without utilizing a route discovery mechanism and stored routing tables. The use of conventional routing protocols may result in large routing tables, unless each node only communicates with a small set of destinations, as for example in a pure data collection. This makes geographic routing very promising for large scale networks. Another advantage of geographic routing is that it is well suited for a joint design of routing and scheduling, because its hop-by-hop decisions allow fast adaptation on changing links. In this chapter the utilized network topology and two different geographic routing schemes as well as an estimation of the traffic for each node are discussed.

4.1 Topology

In this work a solar tower power plant was chosen as example network topology, where each mirror within the plant represents a wireless network node. The resulting topology is a partially connected mesh consisting of multiple equidistant concentric circles. Each ring contains as many equidistantly distributed nodes as possible. Figure 4.1 shows an example network with 2042 nodes distributed on 25 rings. The nodes have approximately the same distance to the next ring as to the next node on the same ring. The network was created using a MATLAB script, which determines how many nodes fit on each circle with the given distance and then distributes them with equal space between the nodes. Using a sufficient transmission range, which allows communication with each direct neighbor, this dense topology shows no routing holes.

However, such a network may look different in practice as the mirror density usually is higher near to the center. Also there may be maintenance roads within the power plant which create a large gap in the topology. This could lead to different routing decisions.

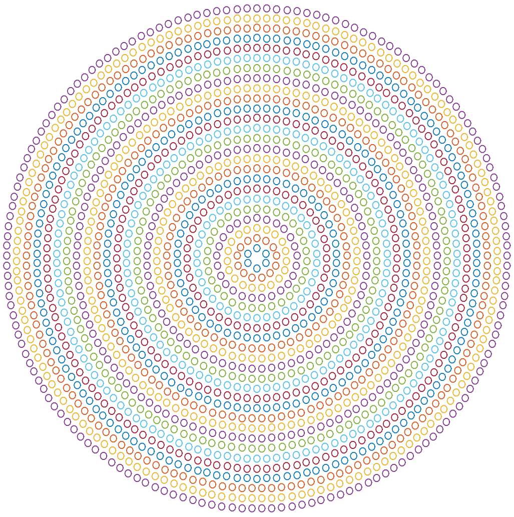


Figure 4.1: Example network with 2042 nodes

4.2 Nearest Neighbor

As discussed in Section 2.2.1, a common approach to determine the next hop device is the "greedy forwarding" method, which selects the node out of the neighboring devices which is closest to the destination.For this purpose, a device computes all the squared euclidean distances $||x_i - x_d||_2^2$ of the positions of its neighbors x_i and the destination x_d. As the device only needs to select the node having the minimum distance and the actual range is not of interest, the squared distance is used to avoid computing the square root on the device.

An example bidirectional route is shown in Figure 4.2. Routes are represented by black lines, where the lower route is directed from the left node to the right and the upper route goes back. The routes are computed by a MATLAB script given the node positions and an communication range which allows connections to every direct neighbor. These possible connections are marked with dashed red lines between the nodes. The straight blue line shows the direct path between the source and destination node. Both routes have the same amount of 23 intermediate nodes, but they go along different positions.

A fully routed network is shown in Figure 4.3(a). It shows the result of determining every first routing step of each node's route to the center. The traffic of nodes is highly depending on how many child nodes it has in its involved routes. Therefore the nodes with the highest traffic are the nodes on the inner ring. As may be seen in the figure, the traffic at the inner nodes is not equally distributed, some nodes even have no children at all. Figure 4.3(b) shows the distinct areas of the children set for each inner node, which illustrates the unequally distribution of traffic. One of 6 inner nodes has to route half of the network traffic and 2 nodes have no child

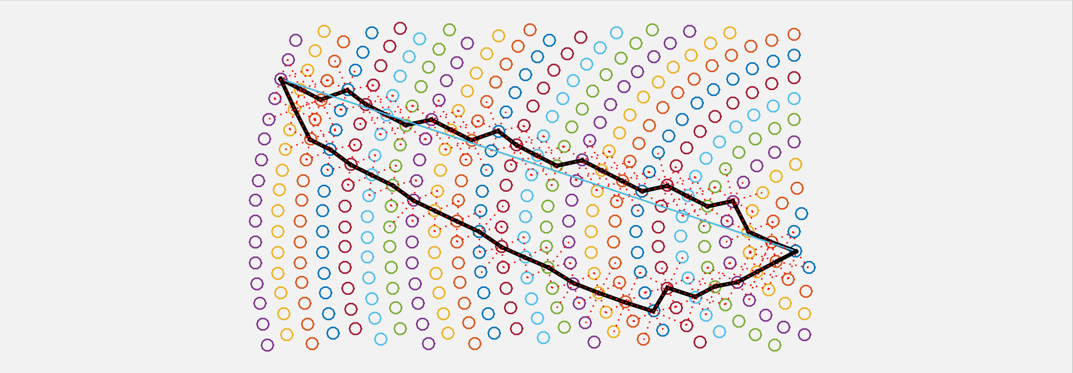
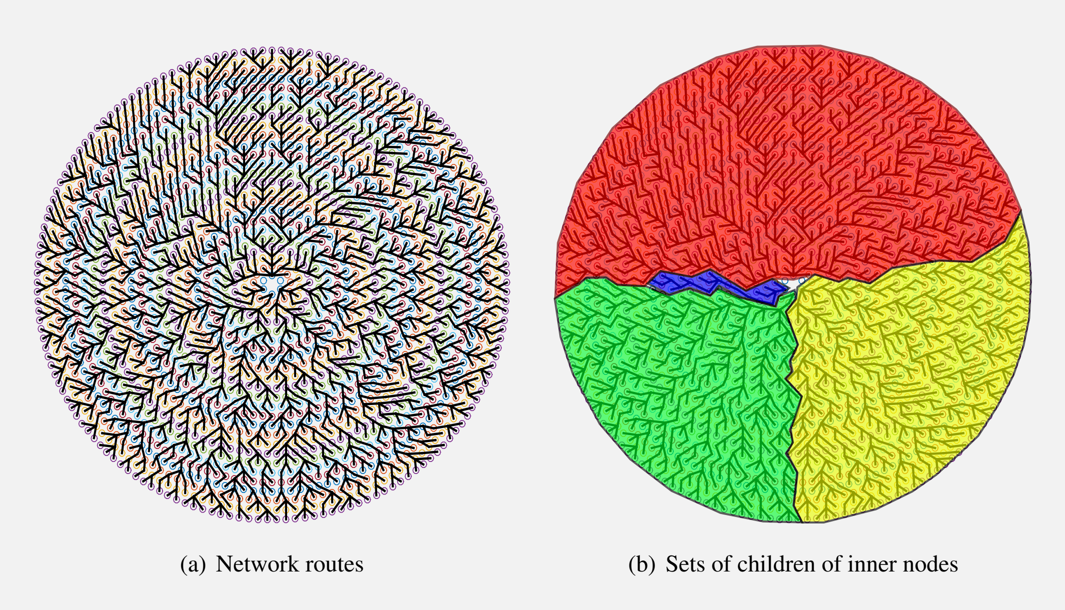


Figure 4.2: Example bidirectional route using nearest neighbor



(a) Network routes

(b) Sets of children of inner nodes

Figure 4.3: Fully routed network with 2042 nodes using nearest neighbor

at all. The overall distribution of children is given by the plot in Figure 4.4, where the node index corresponds with the nodes position. The inner nodes have the indices 1 to 6. The children count is given for every node. This plots verifies the unequal traffic distribution using this routing method for such a network structure.

4.3 Straightest Neighbor

To overcome the unevenly traffic distribution of the nearest neighbor method and thus improve scalability of the geographic routing, this section introduces a modified approach called straightest neighbor. This method is similar to "compass routing" which selects the node with

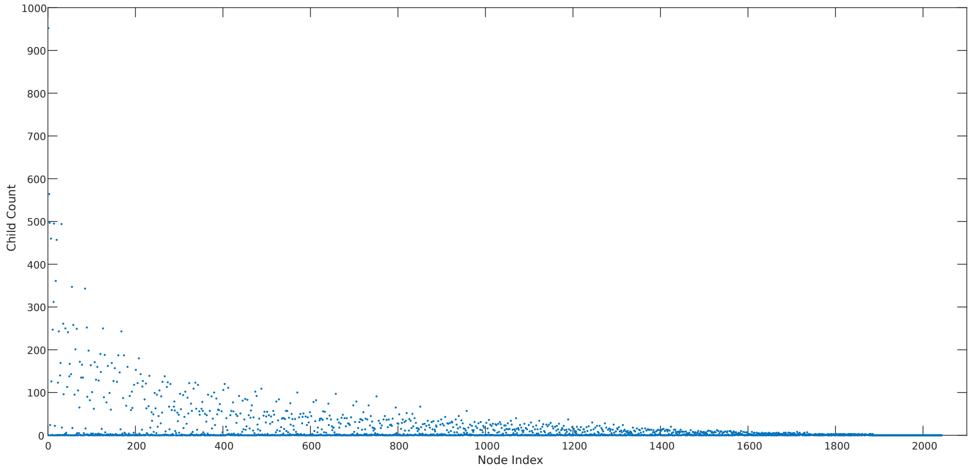


Figure 4.4: Nearest neighbor: Traffic distribution

the smallest slope from the current position to the destination as described in Section 2.2.1. In contrast to that, straightest neighbor always considers the source location at every intermediate forwarding node. Thus it constructs the straightest route from source to destination by selecting the node in direction of the target which has minimal distance to the line between source and destination. For this purpose every intermediate node computes the distance of any node, which is closer to the destination than itself, to the line $\vec{g}$ from source a to destination b. The distance d of a node at point $\vec{P}$ is given by the length of the orthogonal distance vector from $\vec{P}$ to the point on $\vec{g}$. This point is determined as illustrated in Figure 4.5 by the intersection of the orthogonal straight line through $\vec{P}$ given by $\vec{g}_p$, which yields s.

$$\begin{aligned}
\vec{g}\,(s) &= \vec{a} + s \cdot \vec{c} \\
\vec{g}_p\,(\vec{x}) &= \vec{c} \cdot \vec{x} \\
\text{with} \quad \vec{c} &= \vec{b} - \vec{a} \\
\vec{g}_p\,(\vec{g}\,(s)) &= \vec{g}_p\left(\vec{P}\right) \\
\Rightarrow s_p &= \frac{\vec{c} \cdot \left(\vec{P} - \vec{a}\right)}{||\vec{c}||_2^2} \\
d^2 &= \left|\left|\vec{P} - \vec{g}\,(s_p)\right|\right|_2^2
\end{aligned}$$

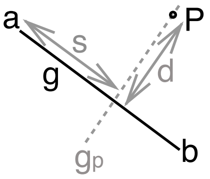


Figure 4.5: Distance from point to straight line

Figure 4.6 shows an example route between the same nodes as in Figure 4.2 using straightest neighbor. The generated bidirectional route now uses the same 23 intermediate nodes in both directions. However, in general this method may create longer routes using more intermediate routing devices than nearest neighbor, as the next-hop does not assure the greatest advance.

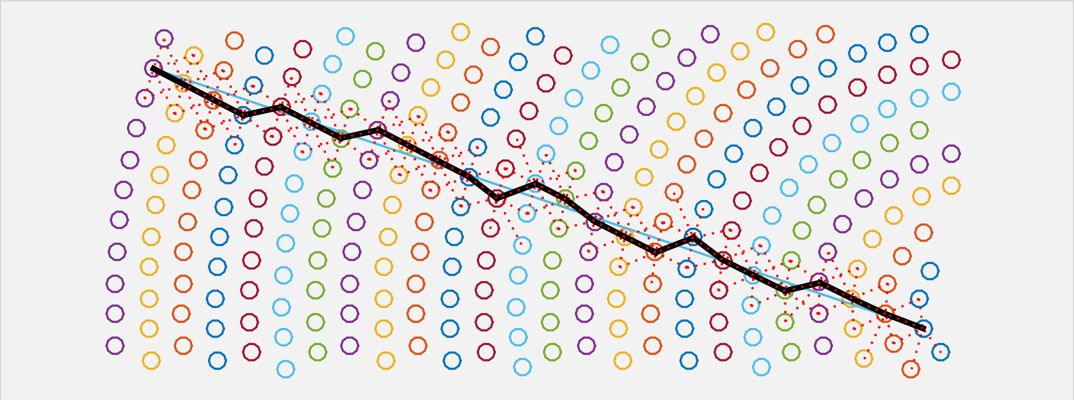


Figure 4.6: Example bidirectional route using straightest neighbor

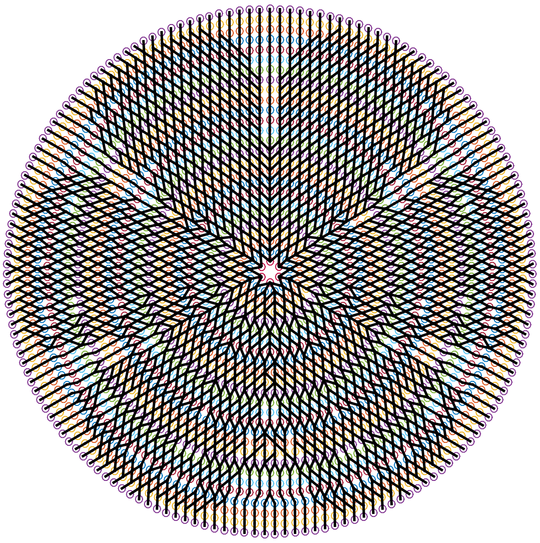


Figure 4.7: Fully routed network with 2042 nodes using straightest neighbor

The fully routed network shown in Figure 4.7, including all routes from any node to the center, evinces a more symmetric traffic distribution compared to the result using nearest neighbor in Figure 4.3(a). The children count is given in the plot in Figure 4.8. The children are now more evenly distributed on each ring compared to the results of the nearest neighbor method (represented by blue data set). Using straightest neighbor, the children count (orange data set) on the inner ring is between 338 and 341. This comparison also holds for a more dense neighborhood, as shown in Figure 4.9 with an example network using a range of 3 nodes. Although the straightest neighbor distribution shows a stronger variation than before, it still achieves a more equal distribution compared to nearest neighbor, which continues to leave some nodes routing most of the traffic and many nodes without children.

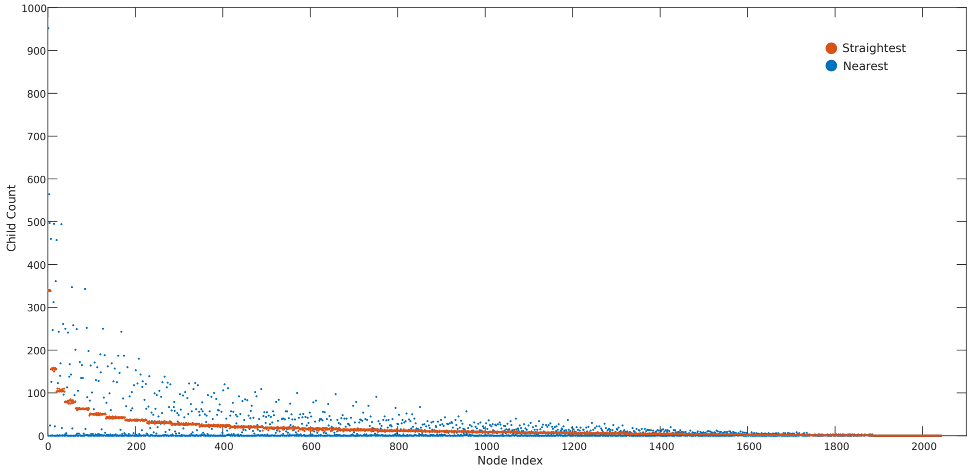


Figure 4.8: Straightest Neighbor: Traffic distribution

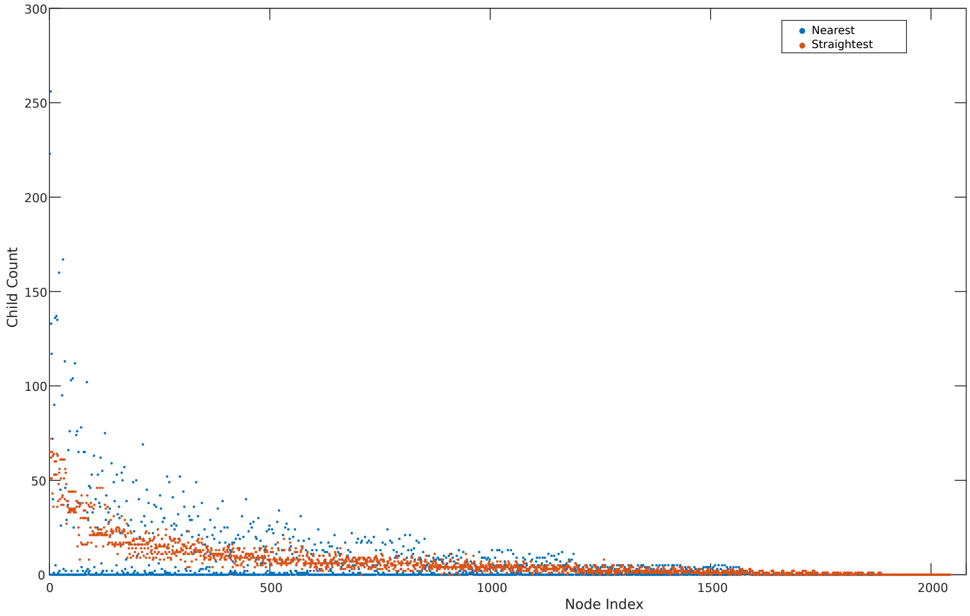


Figure 4.9: 3-hop Neighborhood Traffic distribution

These results promise a strong improvement on the scalability of the routing algorithm using the same information and a feasible increase of computational costs. In addition to the more equally distributed traffic, most nodes now forward to more than one device and thus are less susceptible to a failure of a single node. Also the property of the bidirectional route can be exploited for time slot allocation if useful for the application.

4.4 Traffic Estimation

Assuming that the routed traffic is equally distributed on all nodes located on the same ring, the number of children per node may be estimated by the number of nodes on outer rings divided by the node count on the considered ring. If the nodes shall have about the same distance d_N on a ring, the number of nodes on a ring n with radius r_n is given by

$$N_n = round\left(\frac{2\pi r_n}{d_N}\right).$$

The amount of nodes located on rings behind a ring m is the difference of all nodes and the nodes up to that ring, where the total number of nodes up to a ring may be estimated as the sum of nodes of each ring

$$N_{m+} = round\left(\sum_{k=1}^{n}\frac{2\pi r_k}{d_N} - \sum_{i=1}^{m}\frac{2\pi r_i}{d_N}\right).$$

If the distance between the rings is about equal to the distance between nodes on a ring, the estimation for N_{m+} can be simplified using $r_n = n \cdot d_N$, which yields

$$N_{m+} = round\left(\left(n\left(n+1\right) - m\left(m+1\right)\right)\pi\right).$$

With the initial assumption of equal traffic distribution, the count of children per node on a ring m may then be approximated by

$$N_c \approx \frac{N_{m+}}{round\left(2\pi m\right)}.$$

Figure 4.10 shows the resulting traffic distribution of Section 4.3 compared to the estimation for a network with 2042 nodes on 25 rings. The detailed view of the child count of the most inner ring evinces that the given estimation is close to the mean value of the nodes child count.

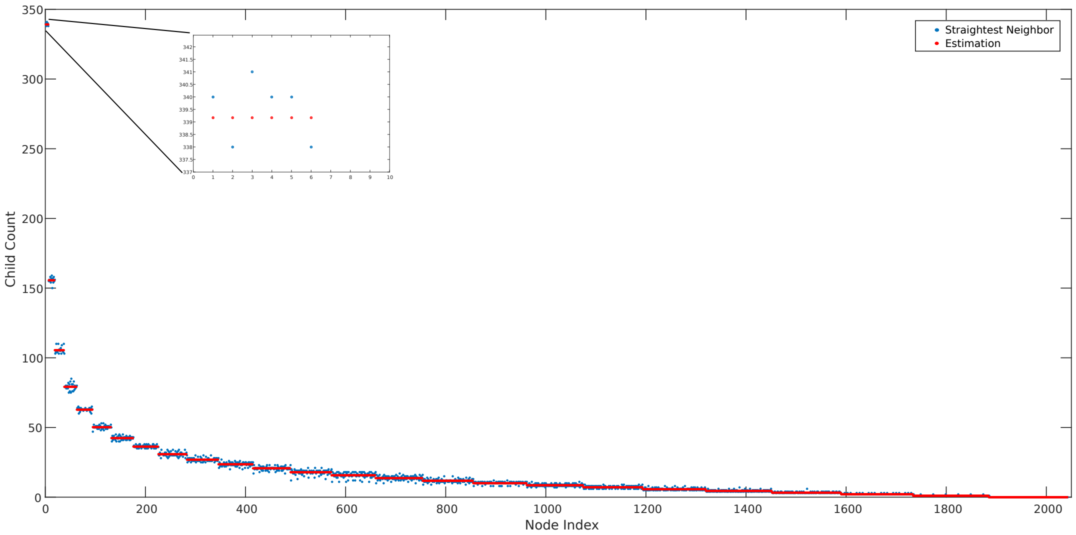


Figure 4.10: Straightest Neighbor: Traffic distribution estimation

Chapter 5

OMNeT++ Implementation

To allow simulative evaluating the performance of example networks, a simulation model of DSME and the straightest neighbor routing mechanism have been implemented as part of this work using the INET framework [ine] for the OMNeT++ [VH08] simulator. The basic structure and behavior of the implemented software is described in this chapter.

5.1 Class Description

The geographic routing is implemented within the network layer using a neighbor table which gets created by the network configurator at the beginning of the simulation, thus the nodes do not have to discover their neighboring devices. This table maps their coordinates to the MAC address. Messages received from the upper layer are encapsulated into a geographic datagram which includes the source and destination coordinates in its header. This allows to forward messages to the next hop node using the straightest neighbor method from Section 4.3. The geographic datagram is then handed over to the link layer, where DSME is implemented. Figure 5.1 shows the class diagrams of the network layer implementation. The `GeographicNetworkProtocol` extends the `GenericNetworkProtocol` of INET, by adding the coordinate information to the datagram and then selecting the next hop node using the neighbor table.

The `DSME` class is implemented as extension of the `CSMA` implementation of INET, since slotted CSMA-CA is used within the contention access period. To convert the implemented unslotted CSMA-CA algorithm to a slotted version, the self-messages, which are used to trigger events like the end of the backoff time, are intercepted and rescheduled by `DSME::handleSelfMessage`. The class diagrams shown in Figure 5.2 shows the most important methods of the `DSME` and `CSMA` classes as well as the utilized message and data types.

To send a message to the next hop node via the physical layer, DSME first has to establish a link to it by allocating at least one GTS. Until the slot is successfully allocated, the message

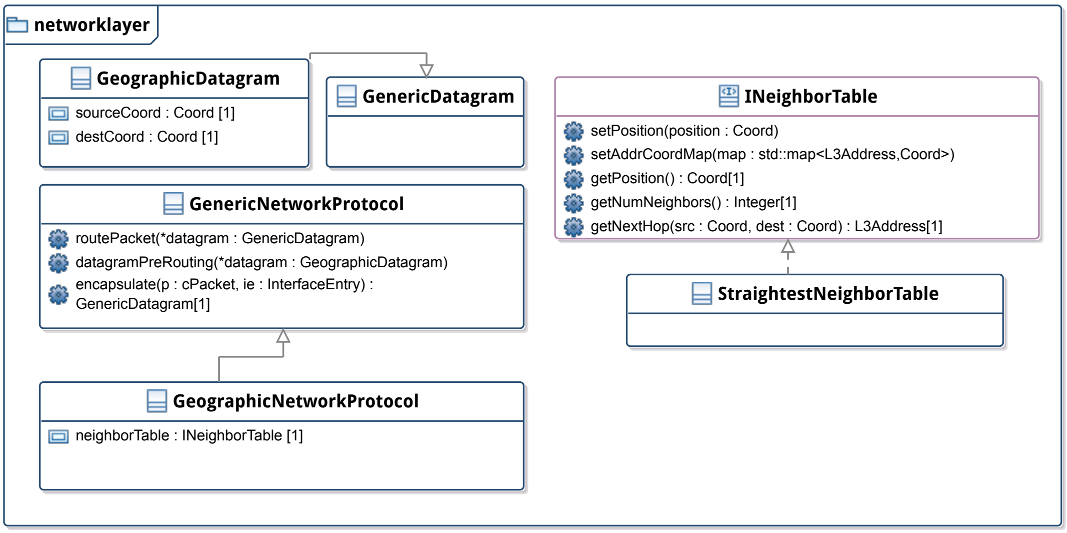

networklayer
GeographicDatagram
sourceCoord : Coord [1]
destCoord : Coord [1]
GenericDatagram
INeighborTable
setPosition(position : Coord)
setAddrCoordMap(map : std::map<L3Address,Coord>)
getPosition() : Coord[1]
getNumNeighbors() : Integer[1]
getNextHop(src : Coord, dest : Coord) : L3Address[1]
GenericNetworkProtocol
routePacket(*datagram : GenericDatagram)
datagramPreRouting(*datagram : GeographicDatagram)
encapsulate(p : cPacket, ie : InterfaceEntry) : GenericDatagram[1]
StraightestNeighborTable
GeographicNetworkProtocol
neighborTable : INeighborTable [1]

Figure 5.1: DSME Class diagrams: Network layer

is stored in the transmit queue `gts_queue` for the given destination address, thereby each destination has its own queue of the same size. Before a device may allocate slots it has to be associated and synchronized to a coordinator of its PAN. For simplification, nodes only use their unique extended address and assume that they have successfully associated with a coordinator once they received a beacon frame from it. Further details on the beacon management are described in Section 5.2. All of the DSME notification and command messages are transmitted using slotted-CSMA by calling the `sendCSMA` method, which uses `handleUpperPacket` of `CSMA` to store the message in its queue `macQueue` and initiates the algorithm.

The application traffic received from the upper layer is only sent within allocated GTS. For this purpose, `DSME::handleUpperPacket` will store the message into the transmit queue call `checkAndHandleGTSAllocation` to allocate new slots if appropriate. This mechanism as well as the transmission and reception using GTS are described in more detail in Section 5.3.

5.2 Beacon Management

The first coordinator which initially sends a beacon is the PAN coordinator, other coordinators start to allocate their beacon slot once they received a beacon. When the `handleLowerPacket` method detects that a received broadcast message is an `EnhancedBeacon`, it will call `handleEnhancedBeacon`. Figure 5.3 shows the flow chart for the behavior of a device when receiving a beacon of a coordinator. First, it synchronizes its system

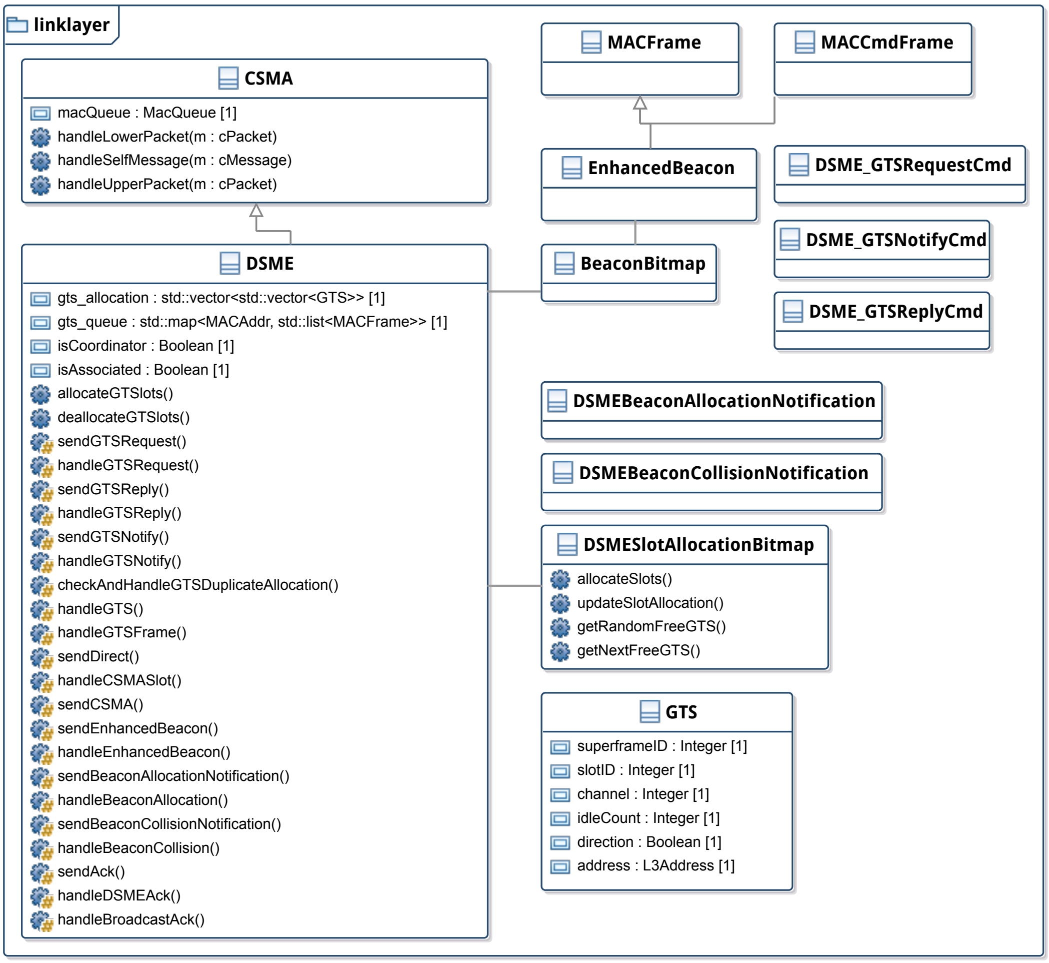


Figure 5.2: DSME Class diagrams: Link layer

time by rescheduling the next-slot timer using the timestamp value included in the beacon frame. It also updates the bitmap of heard beacons by setting the corresponding bit for the given slot ID. The beacon frame also includes a bitmap indicating at which beacon slots the coordinator hears other beacons. This information is stored in the bitmap for beacons heard by neighbors. If at a later time a different coordinator wants to allocate one of these slots, the device will detect the conflict within the `handleBeaconAllocation` method and sends a beacon collision notification using `sendBeaconCollisionNotification`. After these updates non-coordinator devices now behave as associated to the PAN. Coordinator devices which do not have a successfully allocated beacon slot then select a random free slot considering the heard beacons and send a beacon allocation notification broadcast message using the `sendBeaconAllocationNotification` method. They will send their own beacon within that slot using a timer until they receive any collision notification. In that case,

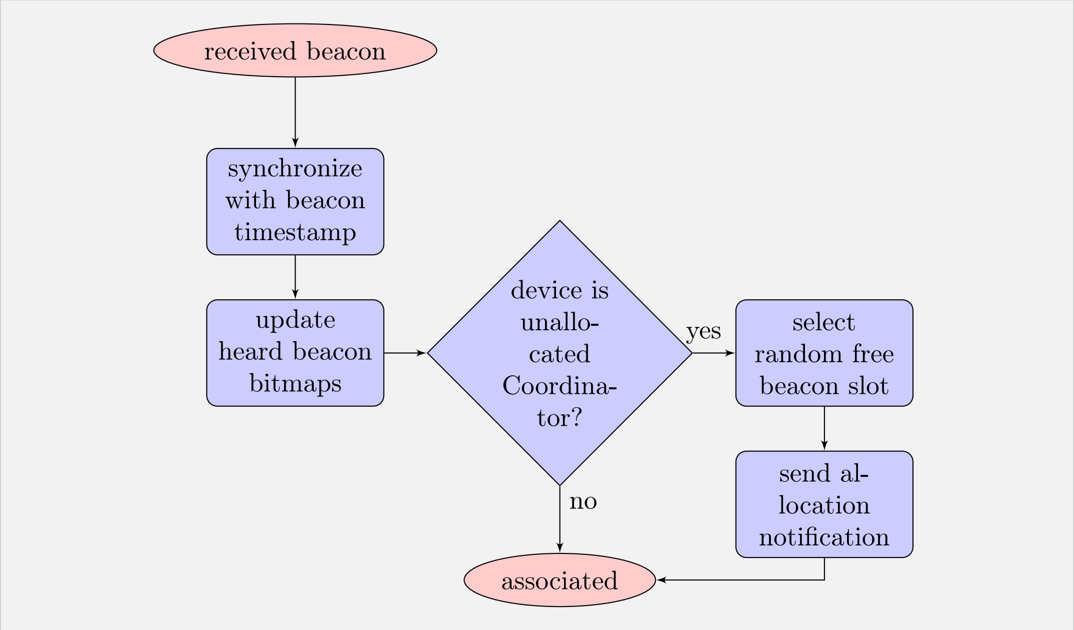

received beacon
synchronize with beacon timestamp
update heard beacon bitmaps
device is unallocated Coordinator?
yes
select random free beacon slot
send allocation notification
no
associated

Figure 5.3: Flow chart: Reception of beacon

the `handleBeaconCollision` method will stop the beacon timer and reset the status, to later trigger a new allocation.

5.3 GTS Management

Two timers are used to handle the communication over guaranteed time slots. The first timer event triggers a short time before the next slot to prepare the physical layer for either receiving or transmitting on the specified channel if the next slot is allocated. The second timer then allows the transmitter to start the transmission directly at the slot boundary while the first timer assured that the receiver is already listening on the correct channel. This next-slot timer updates the value of `currentSlot` and calls `handleGTS` if it is an GTS. The `handleGTS` method will then select GTS information from `gts_allocation` and transmit the next frame waiting in the queue to the corresponding destination using `sendDirect`. If the transmitting device has no message in the queue for the destination for the current slot, it increments the idle counter. A receiving device also increments this counter for slots where no frame was received. The idle count gets reset when a frame was received or transmitted.

As illustrated in the flow chart shown in Figure 5.4, the status of the GTS allocation is checked recurrently. If a maximum idle count is set, the device looks for GTS which exceeds this limit and initiates the deallocation process for one destination if any where found. As a next step the transmission queues are being checked if there exist enough allocated GTS to each destination to handle the traffic. The number of allocated GTS is assumed to be sufficient if either the predefined maximum is reached or the number of messages in the queue does not exceed the amount of GTS. If for any destination the allocation status is insufficient, one or more random free GTS will be selected as preferred and requested from the destination device. A random allocation pattern is promising to achieve a small probability of duplicate allocations, as devices start allocating simultaneously. However, this scheme may lead to an imperfect slot utilization as discussed in Section 7.4

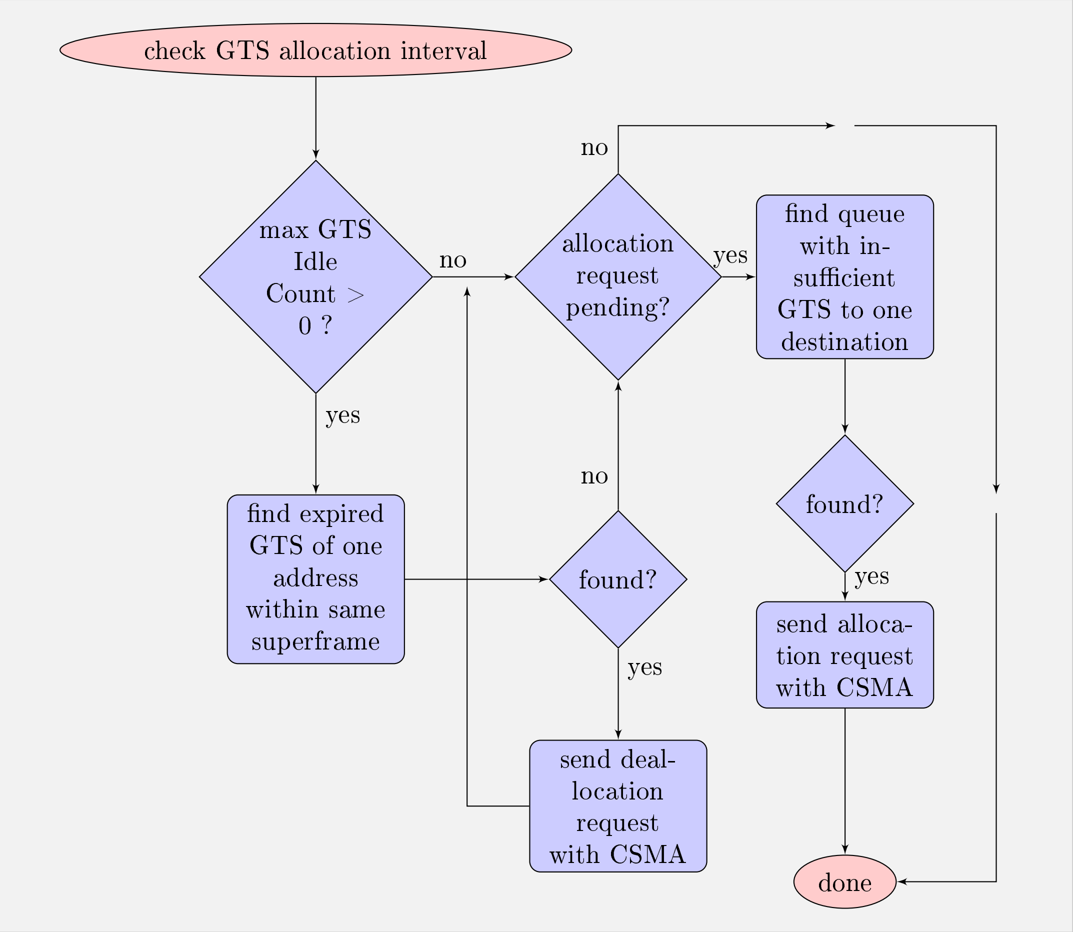

check GTS allocation interval
max GTS Idle Count > 0 ?
no
yes
allocation request pending?
no
yes
find queue with insufficient GTS to one destination
find expired GTS of one address within same superframe
no
found?
yes
found?
yes
send allocation request with CSMA
send deallocation request with CSMA
done

Figure 5.4: Flow chart: GTS De-/Allocation management

The method `handleGTSRequest` allocates the preferred slot if possible or a follow-up slot of the same superframe and sends the reply using `sendGTSReply` via `CSMA` as a broadcast message including an encapsulated destination address within the payload. `DSME` expects an acknowledgement from this destination, therefore the frame resides in the `macQueue` of `CSMA` and gets retransmitted in case no acknowledgement was received. Neighboring devices which receive this broadcast message will check the allocation for duplicates within `handleGTSReply` as well as in `handleGTSNotify` when receiving the notify com-

Parameter	Description
beaconOrder	Influences number of unique beacon slots
superframeOrder	The Superframe and slot duration
multiSuperframeOrder	Number of Superframes per Multi Superframe and therefor the amount of GTS per channel (maximum allocatable per device) see Section 2.4.2 for details
isPANCoordinator	Set for the center device which initially sends beacons
isCoordinator	Set for every node which shall act as coordinator
coordinatorEveryNthNode	Sets isCoordinator automatically for every n'th device, depending on its host id
maxNumberGTSAllocPerDevice	The maximum number of GTS a device is allowed to allocate per destination
maxNumberGTSAllocPerRequest	The maximum number of GTS a device is allowed to allocate in a single request
maxGTSIdleCount	Maximum number of slots a GTS may be unused in a row until it shall get deallocated
queueLength	The length of a queue for a single destination

Table 5.1: OMNeT++/INET - DSME Parameter

mand. The destination device of this reply message will directly send an acknowledgement and the notify using `sendGTSNotify`. As the `CSMA` implementation does not expect acknowledgements upon a broadcast message, it was modified to expect it and to call the `handleBroadcastAck` method upon its reception, which is implemented by the `DSME` class.

5.4 Parameter

The behavior of the implemented DSME MAC can be varied by the parameters described in Table 5.1. Besides the configuration of the multi-superframe structure, it defines if a device acts as a coordinator, how many messages may be stored in the queue per destination and how many slots a device is allowed to allocate per destination and request. The slotted CSMA uses the same parameters as the CSMA and the additional contention window CW_0 parameter as discussed in Section 2.1.2.

Chapter 6

Analytical Performance Evaluation

In this chapter the theoretical performance of DSME is discussed in more detail, also the factors which influence the throughput, latency and packet loss are highlighted. The last section shows the results for an exemplary network using the topology described in Section 4.1 and the forwarding mechanism of Section 4.3.

6.1 Throughput

The throughput in the considered concentric circle topology, using an application where the center node collects data from all outer nodes, highly limits the overall throughput by the inner nodes. The absolute limit is given by the center node, as it participates in the overall application traffic from outer nodes. The maximum achievable throughput for a single DSME device is determined by the number of time slots N_{GTS}, the number of frames N_F per slot, the frame length and the bitrate. These values depend on the DSME network configuration parameter SO as shown in Table 6.1. The maximum frame length depends on the slot, ACK message and interframe spacing durations as well as the bitrate. The timing details are shown in Figure 2.5 in Section 2.4.1. It is also limited by the physical layer which supports a maximum payload of 127 bytes and a bitrate of 250 kbps. The PHY also adds a preamble sequence of 8 symbols and a one byte header to the frame. The maximum frame length (including the header) can then be determined by

$$\begin{aligned} maxFrameLength = min\big(127, \Big(&baseSlotDuration \cdot 2^{SO} - t_{PRE} - t_{ACK} - t_{IFS}\Big) \\ &\cdot secondsPerSymbol \cdot bitrate \cdot 8 \\ &- PHYHeaderLength - ACKLength\big). \end{aligned}$$

The maximum frame rate f_{max} is then given by the number of frames per multi-superframe duration t_{MD}. The throughput is the product of the frame rate and the frame length. Consid-

SO	t_{SD}	N_F	$maxFrameLength$	T_{max}
1	1.92 ms	1	20 byte	45.57 kbps
2	3.84 ms	1	80 byte	77.47 kbps
3	7.68 ms	1	127 byte	57.88 kbps
4	15.36 ms	2	127 byte	57.88 kbps
5	30.72 ms	5	127 byte	72.35 kbps
6	61.44 ms	11	127 byte	79.58 kbps
7	122.88 ms	22	127 byte	83.2 kbps
8	245.76 ms	45	127 byte	85 kbps

Table 6.1: Maximal DSME throughput without CAP reduction

ering the maximum throughput of the center node, the maximal frame rate or minimal send interval for outer nodes can be determined. Assuming that every node except the center node sends a new frame to the center with a send interval t_{SI} and frames are not getting dropped on their path, the maximum frame rate has to be divided by the number of nodes. This gives the maximal frame rate per node f_n and the minimal send interval t_{SI}.

$$\begin{aligned}
\text{Maximum frame rate } f_{max} &= \frac{N_F \cdot N_{GTS}}{t_{MD}} \\
\text{Node frame rate } f_n &= \frac{f_{max}}{N_{Nodes}} \\
\text{Send interval } t_{SI} &= \frac{1}{f_n} \\
\text{Throughput } T_{max} &= f_{max} \cdot maxFrameLength \\
N_{GTS} &= 7 \cdot 2^{MO-SO}
\end{aligned}$$

When cap reduction is used, the number of available GTS increases depending on the number of superframes per multi-superframe $N'_{GTS} = N_{GTS} + 8 \cdot (2^{MO-SO} - 1)$. Eight additional GTS are then available within every superframe except the first one.

Table 6.1 lists the maximal throughput which a single node can achieve in a DSME network depending on the parameter SO. In this case only large frames (more than 18 bytes) with the use of acknowledgements are considered and CAP reduction is not used. Thus the number of superframes has no influence on the throughput and $N_{GTS} = 7$ and $t_{MD} = 16 * t_{SD}$. Furthermore, $t_{PRE} = 8$, $t_{ACK} = SIFS = 12$, $t_{IFS} = LIFS = 40$, $secondsPerSymbol = 16\,\mu\text{s}$, $bitrate = 250\,\text{kbps}$, $PHYHeaderLength = 1$ byte and $ACKLength = MACHeaderLength = MACHeaderLength_{min} = 9$ bytes.

6.2 Latency

The average latency can be determined using the queueing model discussed in Section 2.6.2, when assuming a long term constant slot allocation and an application, which generates Poisson distributed traffic. In this section additionally a simplified model is considered to highlight the influencing factors on the latency of TDMA in general and DSME in particular.

6.2.1 Simplified Model

In an optimal network without any delay caused by a queue or unusable intermediate time slots, which allows every node in a path to forward a received frame directly in the next slot, the latency only depends on the number of hops N_{hops} and the slot duration t_{SD}. If every t_{SD} a valid time slot is available, the latency is given by the product $L = N_{hops} \cdot t_{SD}$. Using DSME there are only 7 guaranteed time slots available every 16 slots, thus a delay of 9 slots occurs after 7 slots when starting at the first slot. The minimal possible latency within a DSME network for N_{hops} hops is given by

$$L_{min}\left(N_{Hops}\right) = \left(N_{hops} + 9 \cdot \left\lfloor \frac{N_{hops}}{7} \right\rfloor\right) \cdot t_{SD}.$$

If CAP reduction is used, this delay only occurs in the first Superframe of a Multi Superframe, the following $M-1$ Superframes have just one non-guaranteed time slot. The latency then reduces to

$$L_{min,CR}\left(N_{Hops}\right) = \left(N_{hops} + 8 \cdot \left\lfloor \frac{N_{hops}}{(M-1) \cdot 15} \right\rfloor + 1 \cdot \left\lfloor \frac{N_{hops}}{7} \right\rfloor\right) \cdot t_{SD},$$
$$\text{with } M = 2^{MO-SO}.$$

Assuming that the time slot schedule is not optimal, TDMA causes additional delays at every forwarding node in a path. Figure 6.1(a) illustrates a suboptimal time schedule between three forwarding nodes. Each node has allocated one of five slots for transmitting the next node and has to wait for a few slots after receiving from the prior device. With the assumption that the $N_{GtsAlloc}$ allocated out of N_{GTS} time slots of a single node are evenly distributed over time, the delay increases at maximum by $\frac{N_{GTS}}{N_{GtsAlloc}}$ at every hop. By simplifying that every node has the same amount of slots and that this delay occurs on every reception, the resulting overall latency in a path depending on the number of hops and the allocated slots is given as

$$L_g\left(N_{hops}, N_{GtsAllocn}\right) = L_{min}\left(N_{Hops} \cdot \frac{N_{GTS}}{N_{GtsAlloc}}\right).$$

As multiple frames may be received before any frame can be transmitted at the next slot, frames are usually stored in queues before transmitting them. Otherwise frames would be dropped in such a case. If a queue is not empty, the incoming frame will be transmitted with a delay as illustrated in Figure 6.1(b). If frames have equal priority and are transmitted in a first-in first-out (FIFO) pattern, the additional delay at every hop is equal to the number of frames N_Q waiting in the queue. Under the assumption that every node has the same amount of frames in the queue and may always transmit a frame at the next time slot, the latency in the path is given with $L = N_{hops} \cdot (N_Q + 1)$ or for a DSME network by

$$L_q\left(N_{Hops}, N_Q\right) = L_{min}\left(N_{hops} \cdot (N_Q + 1)\right).$$

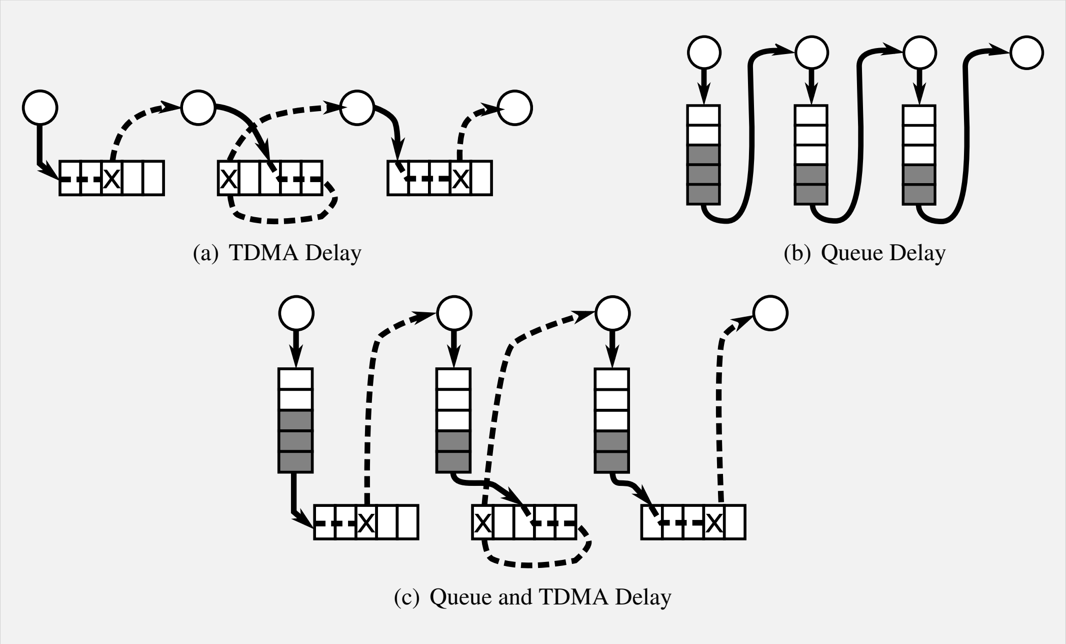
The combination of these two influences is shown in Figure 6.1(c), where transmit queues are used and a suboptimal equally distributed time slot schedule is assumed. In this case the delay caused by the time slot schedule occurs for each frame taken out of the queue. Thus the resulting delay at a single node is the product of the number of frames in the queue and the amount of slots to wait until transmission $\frac{N_{GTS}}{N_{GtsAlloc}} \cdot (N_Q + 1)$. Using the same simplification as before, that this is always the case at every hop, the path latency is given by $L = N_{hops} \cdot \frac{N_{GTS}}{N_{GtsAlloc}} \cdot (N_Q + 1)$ or when considering the DSME Superframe structure by

$$L_{gq}\left(N_{hops}, N_{GtsAlloc}, N_Q\right) = L_{min}\left(N_{hops} \cdot \frac{N_{GTS}}{N_{GtsAlloc}} \cdot (N_Q + 1)\right).$$

The equations introduced in this section are simplified and only hold under the conditions, that each node holds the same amount of evenly distributed time slots and the queue has a constant fill rate. These assumptions are improbable in a real communication network. However, they may be considered as a rough latency estimation for suboptimal configured networks e.g. by using mean values of the slot allocation and queue fill rate. Despite that, the observations made here show that the alignment and the number of allocated slots as well as the average queue occupancy have a big influence on the overall latency when forwarding frames on a path.

6.2.2 Queuing Models

Using the queuing model M/M/1/K or M/D/1/K discussed in Section 2.6.1 and 2.6.2, the average overall path latency can be determined by summing up the average time spent at each node GTS period T_S and the additional time used for the CAP and beacon. As T_S only considers the time spent within the GTS period, the actual latency is more than twice as long, because this period is just a part of $\frac{N_{GtsSF}}{16}$ of the superframe. Where $N_{GtsSF} \leq 7$ unless CAP reduction is used. The service rate μ is given by the number of allocated transmission


(a) TDMA Delay

(b) Queue Delay

(c) Queue and TDMA Delay

Figure 6.1: Latency influence of queues and TDMA

slots $N_{GtsAlloc}$ within a Multi Superframe: $\mu = \frac{N_{GtsAlloc}}{t_{MD}}$ in frames per second. The arrival rate λ depends on the utilized send interval t_{SI} and the number of children N_C whose traffic has to be forwarded. With $\lambda = \frac{N_C+1}{t_{SI}}$, the arrival rate for an outer node without children is $\lambda = \frac{1}{t_{SI}}$ [frames/s]. Given the service and arrival rates and the queue size K, the average overall latency can then be determined by

$$\hat{L}_{gq}\left(N_{hops}, K\right) = \frac{16}{N_{GtsSF}} \cdot \sum_{i=0}^{N_{Hops}} T_{S,i}.$$

Assuming all nodes in the path have the same service and arrival rate, the latency may be estimated by

$$\tilde{L}_{gq}\left(N_{hops}, \mu, \lambda, K\right) = \frac{16}{N_{GtsSF}} \cdot N_{hops} \cdot T_S.$$

6.3 Packet Loss

In contrary to CSMA, where packet collisions are common, using DSME, packets only get dropped by the MAC sublayer when the transmit queue is saturated. As discussed in Section 2.6.1, the saturation probability of the queue highly depends on the utilization (ratio of arrival and service rate) $\rho = \frac{\lambda}{\mu}$ and also on the queue size K. To minimize the packet loss, the service rate should be greater than the arrival rate and thus a node should allocate more

λ	μ	$K_{MM} \geq$	$K_{MD} \geq$
1	4	9	5
2	4	16	9
3	4	36	19

Table 6.2: Exemplary minimum queue sizes

GTS for transmission than it holds for reception. In addition to that, the queue size has to be big enough. Given an arrival and service rate as well as the minimum packet loss probability P_L, the queue size can be easily determined. Table 6.2 shows the necessary queue size K to achieve a maximum packet loss of one in a million $P_L = 10^{-6} > P_K$ using the M/M/1/K and M/D/1/K queue models.

With a utilization of 0.5, the queue should already be capable of storing 16 frames given an M/M/1/K and only 9 for an M/D/1/K system. Given a utilization of 0.75, the queue size needs to be more than twice as big. When considering a multi-hop path, the requirement increases. As the probability holds at each node independently, the total packet loss probability of the path is given by the sum of probabilities of each node that the packet was not blocked by a previous but at the regarded node

$$P\left(L\right) = P\left(L_0\right) + \sum_{i=1}^{N_{hops}} \left(P\left(L_i\right) \prod_{j=0}^{i-1} \left(1 - P(L_j)\right) \right).$$

For example, a path with 3 nodes, all having a packet loss probability of P_K would result in a path loss probability of

$$\begin{aligned} P\left(L\right) &= P_K + P_K\left(1 - P_K\right) + P_K\left(1 - P_K\right)^2 \\ &= 3P_K - 3P_K^2 + P_K^3 \approx 3P_K. \end{aligned}$$

Assuming equal conditions for all nodes in a path, the requirement $N_{hops} * P_K < P_L$ can be used for small P_K to determine the size of the queue.

6.4 Exemplary Network Performance

As discussed in Section 6.1, the maximal total network throughput does not depend on the number of nodes in the network structure considered in this work, because of the bottleneck at the center. The center throughput is given by the slot length and the amount of GTS per multi-superframe. Using a slot order of $SO = 3$ and without CAP reduction, the maximum center throughput is 57.88 kbps. This limits the throughput of each node transmitting to the center and thus the send interval increases with increasing node count. The impact of the

Rings	Nodes	Send Interval		Latency	
		CAP	CAP red.	L_{min}	L_{gq}
4	62	1.09 s	0.54 s	4 Slots 0.03 s	1,664 Slots 12.8 s
25	2042	35.8 s	17.9 s	52 Slots 0.4 s	10,400 Slots 79.9 s
100	31,730	557 s	279 s	226 Slots 1.74 s	41,600 Slots 319 s
180	102,353	1798 s	898 s	405 Slots 3.11 s	74,880 Slots 575 s

Table 6.3: Example DSME send interval and latency, with $t_{SD} = 7.68\,\text{ms}$, $N_{GTS} = 56$, $N_F = 1$, $t_{MD} = 0.983\,04\,\text{s}$, $N_{GtsAlloc} = 4$, $N_Q = 12$

network size is regarded in Table 6.3 starting with a small network of 62 nodes up to a very large network with more then 100,000 nodes. It shows the minimal possible send interval per node and the minimum achievable latency L_{min} for nodes on the outer ring as well as the latency estimation for a suboptimal time scheduling and a partially filled queue L_{gq} as introduced in Section 6.2. The considered networks are assumed to have a maximum hop count equal to the number of rings and thus only nodes of the next ring are in range. The latencies are given in number of slots and in seconds using an exemplary slot duration of 7.68 ms. With 8 Superframes this gives a Multi Superframe duration of about 0.98 s and a total of 56 GTS allocatable per device, which doubles when using CAP reduction. For the latency estimation with L_{gq}, it is assumed that every node has a mean of 4 GTS allocated and their queue holds 12 packets.

For the small network, outer ring nodes have a maximum hop count of 4, thus a minimum latency of $L_{min}(4) = 30.72\,\text{ms}$ may be achieved with an optimal DSME time slot schedule. But when considering the assumed suboptimal network configuration, the latency increases distinctly to almost 13 s, which is 400 times higher than the optimum. With about 100,000 nodes an optimal latency of 3.11 s might be achieved in theory and a suboptimal configured network may lead to a latency which is about 185 times higher with roughly estimated 9 minutes. With CAP reduction, the number of GTS doubles with the given setting. Thus the send interval is reduced by half and the frame rate is doubled. The optimal latency also reduces with about a half: $L_{min,CR}(180) = 212$ Slots. The suboptimal latency only reduces comparably if the number of GTS per device also is increased, otherwise it even increases because of the factor $\frac{N_{GTS}}{N_{GtsAlloc}}$.

Using the M/D/1/K queueing model, the average latency can be determined exploiting the traffic estimation or the routing evaluation. Figure 7.1(b) on page 58 shows the values for an example network with 4 rings. The path with the maximum child count shows a series of $N_{C,i} = (11, 4, 2, 0)$. Assuming a queue size of 25, a send interval of 1.5 s and $N_{GtsAlloc} = N_C + 1$, gives $\lambda_i \approx (8, 3.33, 2, 0.67)$ and $\rho_i \approx 0.65$. The latency then can be easily determined

$$\hat{L}_{gq} \approx \frac{16}{7} \left(0.078\,\mathrm{s} + 0.19\,\mathrm{s} + 0.31\,\mathrm{s} + 0.93\,\mathrm{s}\right) = 3.45\,\mathrm{s}.$$

The example shows that the outer nodes with less children cause higher latencies than the inner nodes. This can be avoided by increasing the service rate μ through allocating more slots in relation to N_C for outer nodes.

Chapter 7

Simulative Performance Evaluation

The target of this work is to increase the network reliability by minimizing the overall packet loss caused by the link layer. TDMA is very promising to achieve this, while maintaining or even increasing the throughput compared to the use of CSMA. With CSMA, the packet loss increases with high traffic demand in a dense network, because the channel is busy and packets get retransmitted a finite time until they get dropped. TDMA in general does not depend on retransmissions, as communication between nodes should be guaranteed to be undisturbed by other devices participating within the network.

In this chapter the performance of DSME with geographic routing is further inspected by simulation using the OMNeT++ simulator. First, the configuration of the simulated network is described, followed by an evaluation of the resulting setup. The performance results are analyzed within the subsequent sections. The last sections discuss an issue caused by the utilized slot allocation pattern as well as approaches for improvement within future work.

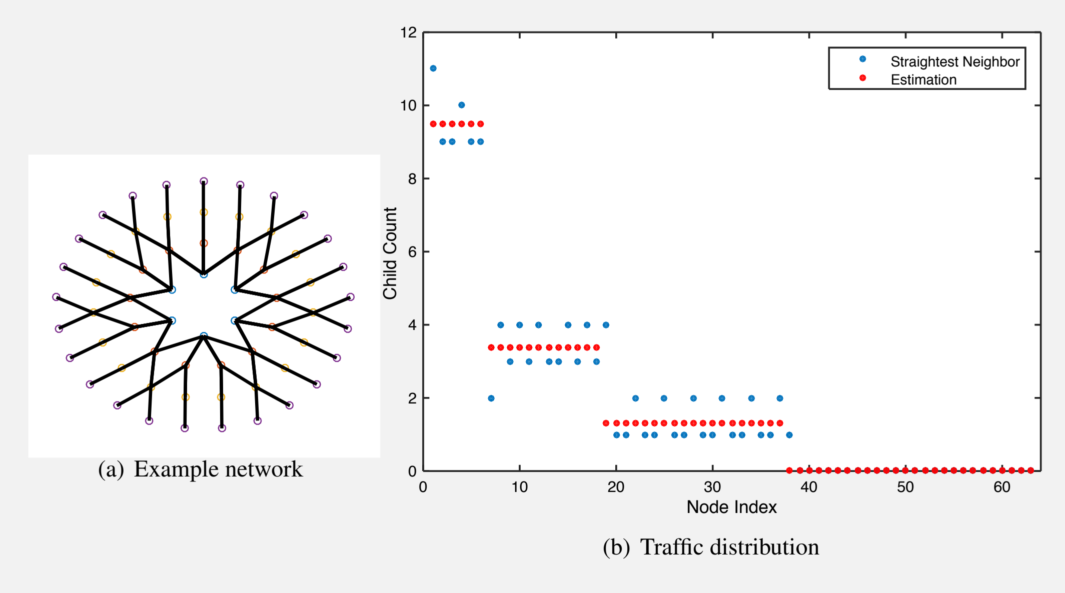
7.1 Simulation Setup

As the nodes allocate new GTS depending on how many messages reside in the transmit queue, the queues may overflow quickly at the setup phase depending on the send interval. Once the GTS allocation is saturated, the nodes should be able to forward all incoming and their own messages without packet loss caused by a full queue. To analyze the setup phase separately from the saturation phase, two separate traffic generators are used. The first generator only creates messages within the setup phase and the queues of all nodes are flushed after that. The second generator can then be configured differently and allows to inspect the performance of an established setup. The target is to determine the best achievable send interval where no packets get dropped by the DSME MAC layer.

7.1.1 Network Configuration

For performance reasons only a small network of 4 rings is simulated in this work. The application traffic is created by each node with a preconfigured send interval and is always sent to the center node. Figure 7.1(a) shows the 62 node network including the straightest neighbor routing paths and the corresponding child count in Figure 7.1(b). To increase the traffic through packet forwarding, nodes can only reach the next ring, which leads to a maximum path length of 4 hops. The center node has only 6 neighbors transmitting to it, thus at least 6 unique GTS are necessary. Considering that the inner nodes have different child counts, more slots are necessary to achieve finer distribution of the slot allocation and hence the traffic. Thus, the DSME Multi Superframe structure is configured with $BO = 7,\ SO = 3,\ MO = 6$, which allows up to 56 GTS per device and 16 beacons within a 2-hop neighborhood. The slot duration of 7.68 s allows a maximal frame length of 127 byte. To reach all nodes within the low density network, every second node acts as Coordinator.

The setup traffic generator is starting at second 5 and generates new messages after a given send interval, which is exponentially distributed. After 90 s have passed, the queues get flushed and the second traffic generator creates up to 100 messages using the same send interval. Considering the maximal center throughput (as discussed in Section 6.1) with 56.97 frames per second, the optimal send interval can not be shorter than $t_{SI} = 1.088$ s.



(a) Example network

(b) Traffic distribution

Figure 7.1: Example network with 62 Nodes

For the DSME configuration two queue sizes are simulated, once with a maximum of 25 and the other with 56 frames. Slots are getting deallocated after being idle for 25 superframes. The compared CSMA configuration uses a single queue with a length of 50 frames, as each node has at most two different destination devices to forward to. The default values $macMinBE = 3$, $macMaxBE = 5$ and $macMaxCSMABackoffs = 4$ are used for the CSMA as well as the slotted CSMA included in the DSME configuration. CSMA frames are getting dropped after 3 retry attempts. All setups are simulated using send intervals between 1.3 s and 2.5 s with a step size of 0.2 s. Each configuration is run 3 times with different random seeds to improve reliance on the results.

7.1.2 Evaluation of the Setup Phase

At the beginning of the setup phase the queues of some outer nodes will already start to fill without the possibility to transmit any messages, because the beacon spread has not reached every node after 5 s. As shown in the sample setup timing diagrams in Figure 7.2, some nodes hear the first beacon after about 8 s. The first beacon was heard at about 2 s, thus the spreading took up to approximately 6 s until every node in a 4-hop distance heard a beacon. The chart in Figure 7.3 shows a sample of how many beacons each node hears directly and how many are heard by its neighborhood. Most nodes have more than 3 coordinators in their one-hop neighborhood, some even here up to 9 or 10 beacons. However, there are a few outer nodes which only receive beacons from a single coordinator. The number of heard beacons in the two-hop neighborhood shows a maximum of 13, thus the number of beacon slots is sufficient. Although the simplified distribution of coordinators is just sufficient, a higher coordinator density could be considered for the outer ring.

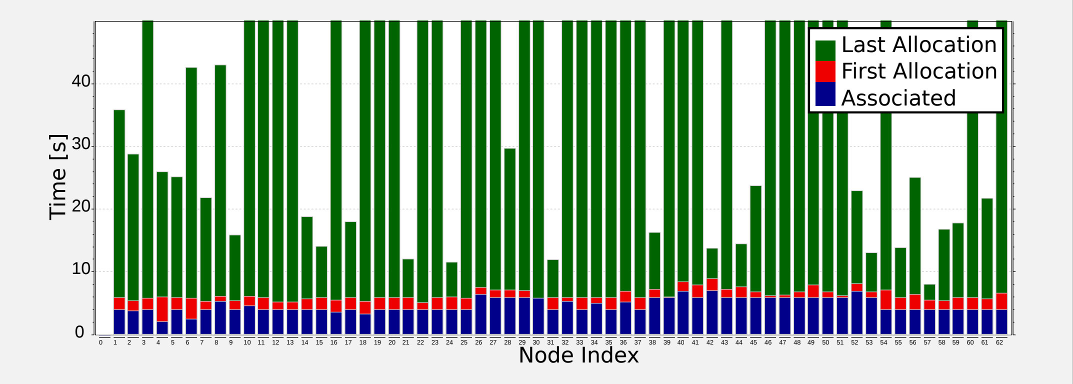


Figure 7.2: DSME Simulation: Setup timing, $K = 25$, $t_{SI} = 2.1\,\mathrm{s}$

Also included in the diagrams of Figure 7.2 is the time of the first successful GTS allocation and the time of the last allocation. As expected, nodes hold their first slot shortly after the setup traffic starts. Some nodes are done with their allocation after 10-20 seconds, whereas other nodes also allocate new slots after the setup phase. This might be because some of their slots got deallocated at some time. As shown in Figure 7.4(a), three nodes are involved in up to three duplicate allocations, whereas in Figure 7.4(b) only a single node receives one duplicate

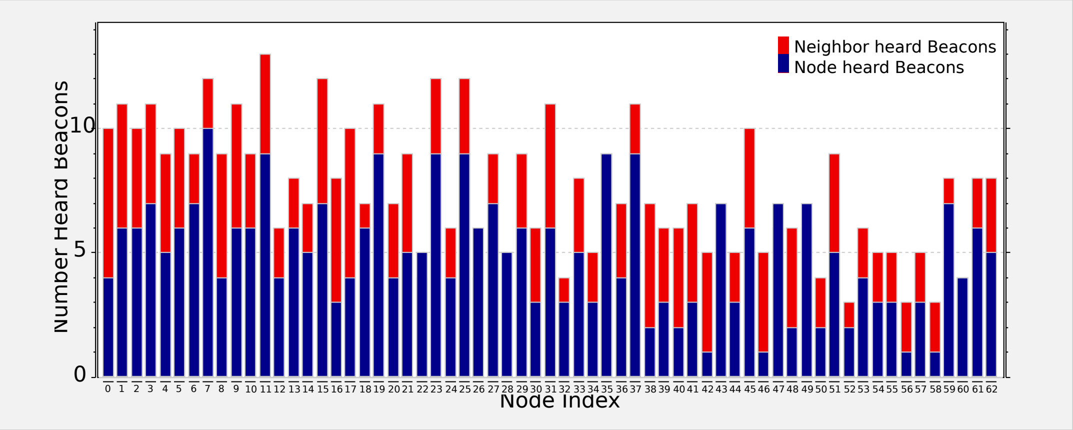
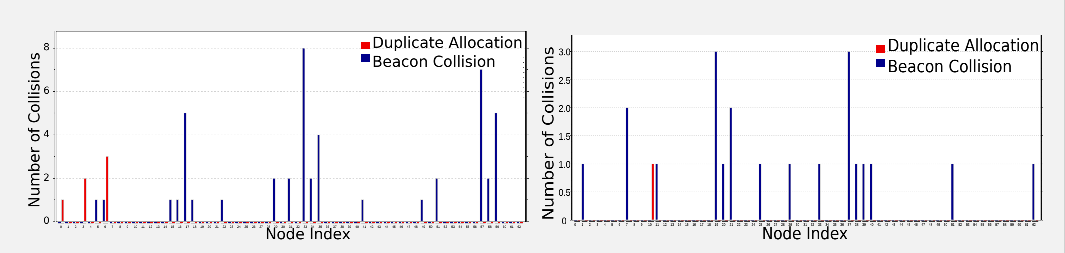


Figure 7.3: DSME Simulation: Number of heard beacons sample



(a) Sample $K = 25$, $t_{SI} = 1.3\,\mathrm{s}$

(b) Sample $K = 25$, $t_{SI} = 2.1\,\mathrm{s}$

Figure 7.4: DSME Simulation: Setup collisions

allocation notification. Within these two samples it seems that the inner nodes experience a more stable allocation using a smaller setup traffic demand. Both figures also show the number of beacon collision notifications, which is close to or equal 0 for most nodes, but some neighboring coordinators experience a few more collisions of up to 8.

Figure 7.5 shows a sample of the resulting GTS allocation of each node after the setup phase. The center node holds 47 of the 56 available guaranteed time slots. This suboptimal slot utilization occurs due to the random slot allocation pattern as discussed in Section 7.4.

The queue fill status is also sampled at the end of the setup phase. Figure 7.6 shows samples of the nodes maximum and mean fill status of their transmit queues for different traffic demands. As shown in Figure 7.6(a), with a send interval of 1.5 s most of the queues have a small fill status and only a few of the inner are close to the capacity. Reducing the systems utilization ρ by using a much lower frame rate with a send interval of 2.1 s in Figure 7.6(b), all of the nodes only hold a few messages in their queues with a saturation of less than 10 %.

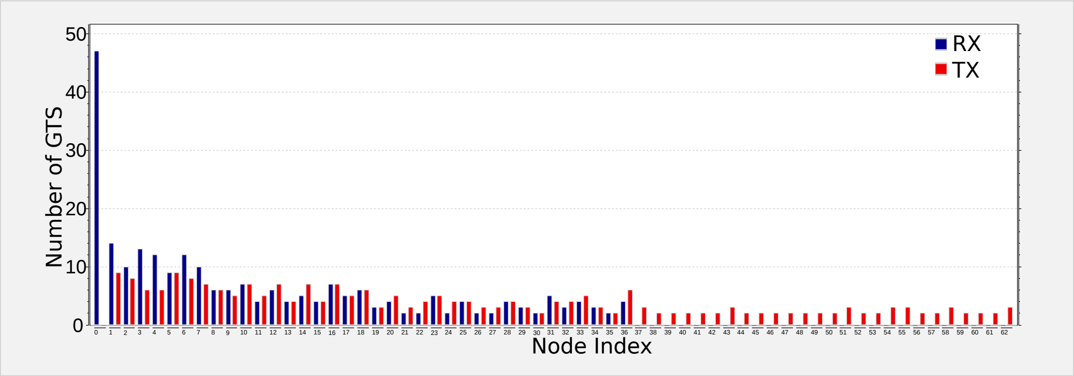
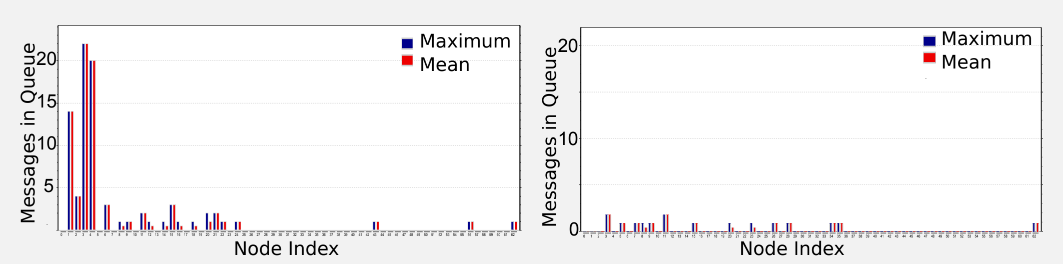


Figure 7.5: DSME Simulation: Setup GTS allocation



(a) Sample $K = 25$, $t_{SI} = 1.5\,\mathrm{s}$ (b) Sample $K = 25$, $t_{SI} = 2.1\,\mathrm{s}$

Figure 7.6: DSME Simulation: Setup queue fill status

7.2 Performance

In this section the performance of the example network using the DSME configurations is compared with a CSMA-CA setup considering throughput, latency and packet loss.

7.2.1 Throughput

The network throughput is measured as the average frames received per second (fps) at the center node within the time interval between the first and the last reception of a frame. The simulation results of the different configurations depending on the send interval are shown in Figure 7.7. It evinces that the throughput mostly depends on the send interval. The DSME setup shows a slightly higher throughput for most of the send intervals using a larger queue. The CSMA configuration shows a little higher throughput for send intervals below 1.9 s and a little less for higher intervals compared to both DSME setups. The achieved throughput is significantly lower than the maximum possible 56.9 fps determined in Table 6.1 on Page 50 using all of the available 56 or also the actual used 47 GTS, which still would allow 47.8 fps. Though it is close to the overall generated network traffic with about 35 of 41.3 fps for a send

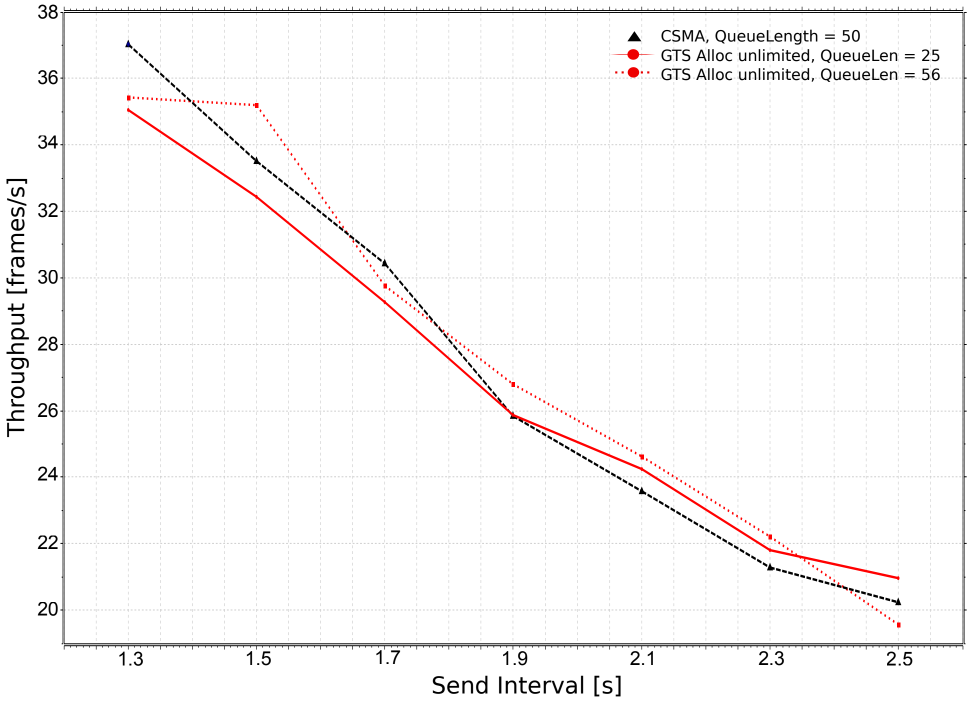


Figure 7.7: DSME Simulation: Average throughput

interval of 1.5 s or about 21 of 24.8 fps at $t_{SI} = 2.5$ s. One reason for this is that at the start of the traffic generation the messages from outer nodes need more time to arrive at the center and at the end the inner nodes stop creating new messages before all of the outer traffic has reached the center.

Overall the throughput using DSME is comparable to the use of pure CSMA with the given setup and slot allocation. Considering only the throughput shows no remarkable advantage of using DSME with this setup rather than CSMA at this point. However, the throughput of DSME might be doubled using CAP reduction and further improved by a higher GTS utilization using a smarter allocation scheme.

7.2.2 Latency

As discussed in Section 6.2, the overall path latency may increase significantly depending on the queue length, the slot allocation alignment and the ratio of arrival and service rate. For this reason the maximum occurring latency is used to compare the different setups. The simulation results are shown in Figure 7.8. It shows that CSMA outperforms DSME with an almost constant maximum latency of about 0.15 s compared to the best achieved using DSME of about 3.6 s for all tested send intervals. The influence of the queue size on a higher latency can be observed, especially for the higher frame rates as then the queues reach closer to their capacity.

Comparing the results to the latency estimation of Section 6.4 shows, that the DSME configuration is far away from the optimal setup, which would allow a latency of about 30 ms. The results are roughly comparable to the suboptimal estimation of about 13 s for a send interval of 1.3 s and a large queue. Although the M/D/1/K queueing model gives no prediction of the maximum latency, the estimation of Section 6.4 for a queue length of 25 and a send interval of 1.5 s shows an average of 3.45 s, which is about half of the maximal latencies of about 7 s from the simulation. As discussed in Section 6.2, the latency can be best improved by providing a better slot alignment or achieving a smaller ρ through allocating more transmit than receive slots. However, this will also reduce the throughput.

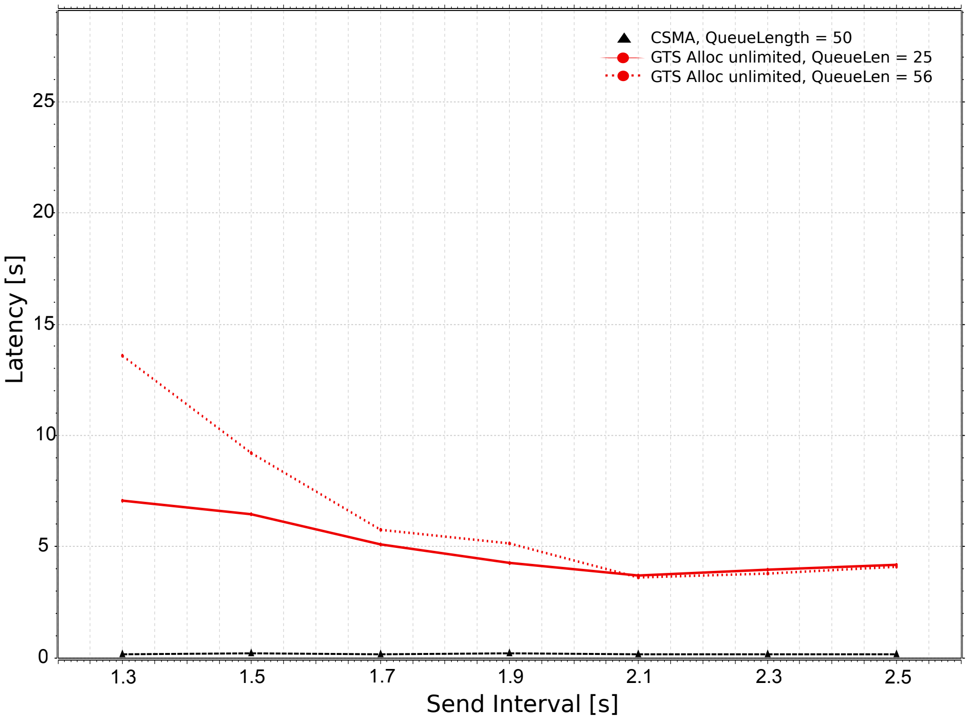


Figure 7.8: DSME Simulation: Maximal latency

7.2.3 Packet Loss

The target of this work is to minimize the packet loss equally for all nodes within the considered network application. Thus the different setups are compared in Figure 7.9 using the maximal occurring packet loss of a single node. CSMA shows an slightly decreasing packet loss of about 4 % for all send intervals. As discussed in Section 6.3, the packet loss highly depends on the ratio of arrival and service rate as packets only get dropped when the queue has reached its capacity. Although the packet loss for some nodes can be dramatically high, given the right configuration DSME outperforms CSMA with a packet loss of 0 for a send interval of 1.7 s or larger. However, a packet loss probability of 0 cannot be guaranteed, as the queueing model M/D/1/K of Section 2.6.2 shows that the blocking probability may be close to but not equal 0.

To increase the frame rate while keeping an optimal packet loss can be achieved by providing a more efficient slot utilization, especially around the center node as discussed in Section 7.5.

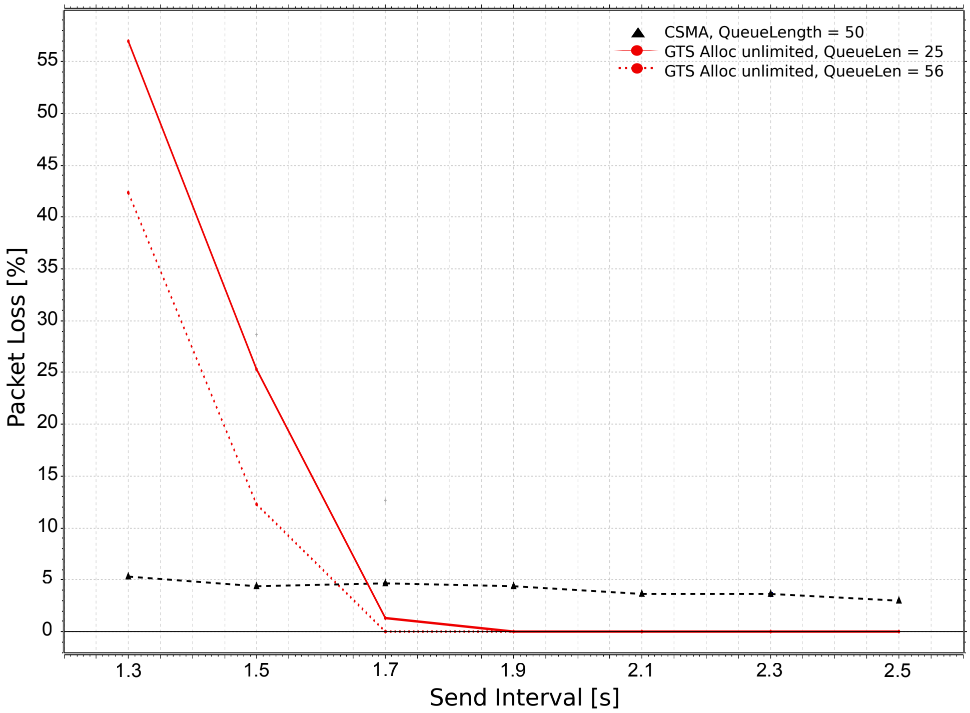


Figure 7.9: DSME Simulation: Maximal packet loss

No.	maxGTS1	maxGTS4	maxGTS2/3/5/6	maxGTS*	t_{SI}	$t_{SI,setup}$	K
1	9	7	6	3	1.3-2.5	0.7 s	25, 56
2	10	8	7	4	1.3-2.5	0.7 s	25, 56
3	10	9	8	5	1.3-2.5	0.7 s	25, 56
4	∞	∞	∞	∞	1.3-2.5	$= t_{SI}$	25, 56

Table 7.1: OMNeT++/INET - DSME Simulation Setups

Fraction	maxGTS1	maxGTS4	maxGTS2/3/5/6	Absolute
N_C	0.193	0.175	0.158	57
$N_{GTS,1}$	0.225	0.175	0.150	40
$N_{GTS,2}$	0.217	0.174	0.137	46
$N_{GTS,3}$	0.196	0.176	0.156	51

Table 7.2: OMNeT++/INET - DSME Setups Traffic Distribution

7.3 Setup Phase Speedup

To reduce the setup phase duration and manually optimize the nodes slot allocation, the setup generator can create messages at higher frequency. To keep fairly distributed slot allocation, the $maxNumberGTSAllocPerDevice$ parameter can be used to limit the number of slots for each node. In this section the performance of different parameter configurations are compared with each other and to a pure CSMA use.

Table 7.1 shows the configurations simulated in this work. The first three setups use a setup traffic with higher frame rates using a send interval of 0.7 s and different GTS allocation limits set for the inner nodes. The chosen values are related to the evaluated traffic distribution shown in Figure 7.1(b) to fairly distribute the available transmit slots to the center node. Node no. 1 needs more slots than no. 4 and both need more than the remaining nodes of the inner ring. Whereas the unlimited Setup 4 is the setup shown in the previous section.

The resulting different GTS distribution provided by the limits of setup 1 – 3 is shown in Table 7.2. The first row shows the fraction of all children for the inner nodes. Node no. 1 and 4 together hold about 36.8 % of the total 57 children. Setup no. 1 and 2 assure the first node a little more resources, whereas node 4 receives about the right amount and the other nodes a little less. Setup 3 provides a distribution which is very close to the child count distribution.

Given the allocation limits $maxGTS_i$, child count $N_{children}$, send interval t_{si} and the queue size K, the blocking probability P_b can be determined with $\lambda = \frac{N_C+1}{t_{si}}$, $\mu = \frac{maxGTS_i}{t_{MD}}$. Utilizing the M/D/1/K model (Section 2.6.2) shows that the first setup can not achieve the constraint of $P_b < 10^{-6}$ for $t_{SI} < 1.7$ s, because node no. 4 has an insufficient utilization of $\rho > 0.84$ for

No.	maxGTS1	maxGTS4	maxGTS2/3/5/6	maxGTS*
1	1.5 s	1.7 s	1.9 s	1.7 s
2	1.5 s	1.7 s	1.7 s	1.3 s
3	1.5 s	1.5 s	1.5 s	1.3 s

Table 7.3: OMNeT++/INET - DSME Setups Minimal Send Interval

$t_{SI} \leq 1.7$ s. Table 7.3 shows the minimum considered send interval each node is capable to achieve with the given setup and a queue size of 25.

7.3.1 Throughput

The simulation results of the different configurations depending on the send interval are shown in Figure 7.10. The results show a greater variation at small intervals with about 32.5 – 37 frames per second. Whereas the throughput with a more than doubled send interval of 2.5 s varies less with approximately 19.5 – 21 frames per second. The first setup is the only setup which shows a significant lower throughput for higher frame rates as it has an insufficient slot utilization. Otherwise the all setups are comparable.

7.3.2 Latency

The influence of the queue size on a higher latency can be observed for all setups, especially for the higher frame rates as the queues reach closer to their capacity. Setup 1 shows the worst results over all intervals when only comparing configurations of same queue size. It has a maximum latency of about 15 s using a message queue of length 25 and 24 s with a doubled queue size. However it gets closer to the other configurations with lower frame rates. Over all send intervals, the best DSME configuration seems to be setup 4 using unlimited slot allocation and the smaller queue size. Though setup 2 and 3 show only slightly higher latencies and at some point achieve an even better result.

7.3.3 Packet Loss

Comparing only the packet loss probability, the best results are achieved using setup 2 or 4. Both setups achieve a constant packet loss of 0 with send intervals greater than at least 1.9 s or 1.7 s when using the larger queue. Considering the configuration overhead of setup 2 makes the unlimited setup the most favorable, although it shows much higher loss rates for lower send intervals. However, a large maximum packet loss is not necessarily an adequate indication for a bad setup. Figure 7.13(b) shows a sample of setup 3 with a maximum loss of about 18 % caused by a single node of the network, which also effects some other outer nodes routing to it.

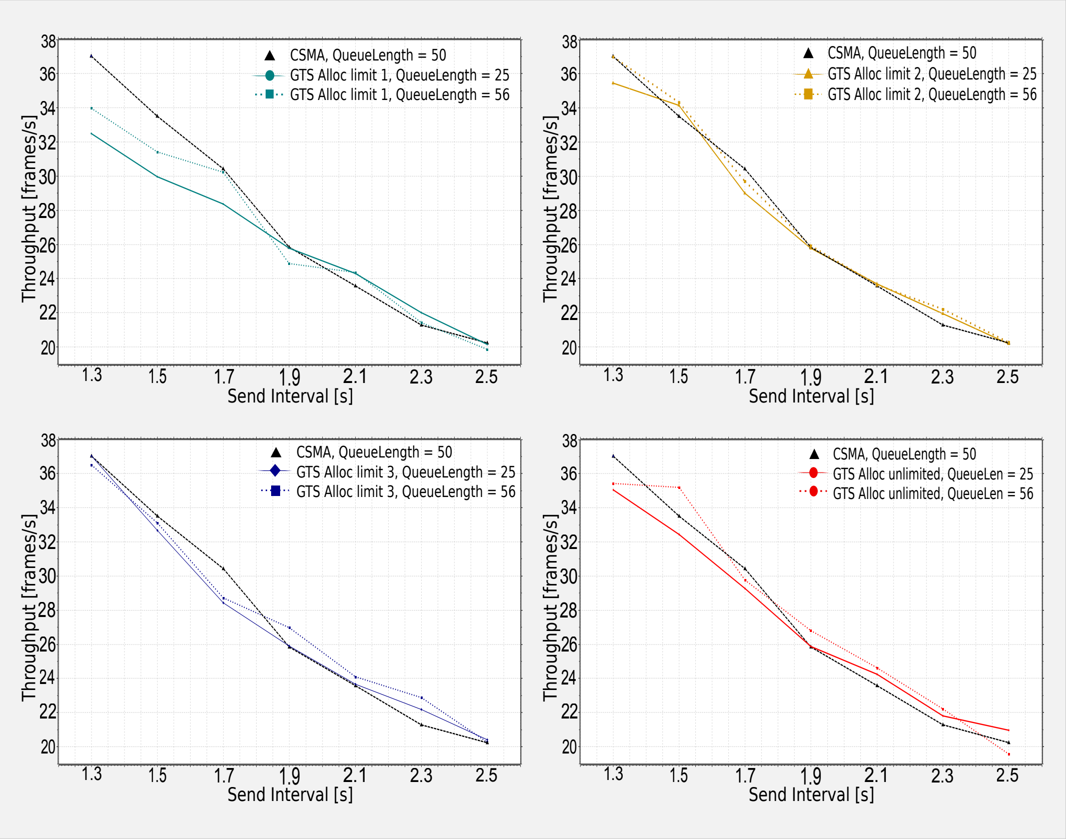


Figure 7.10: DSME Simulation: Average throughput for different GTS allocation limits

In contrast to that, Figure 7.13(a) shows a much worse configuration, where almost every node loses a great amount of their messages and only a few experience small or even no loss at all.

7.4 Random Slot Allocation Issue

The evaluation of the setup results in Section 7.1.2 have shown that a the utilized random slot allocation pattern leads to an insufficient slot utilization, leaving some slots unallocated. The reason for this is that the nodes on the inner ring request slots to transmit to the center and at the same time are being requested to receive from outer nodes, which reduces the set of free slots for each node. Figure 7.14 shows a simplified example, where a link between node A and B is already established using the first of three slots. Node C utilizes the second and third slot to receive from an outer node. Therefor C cannot allocate its last remaining free slot to communicate with node B even though it still has two slots available, leaving both with unused slots.

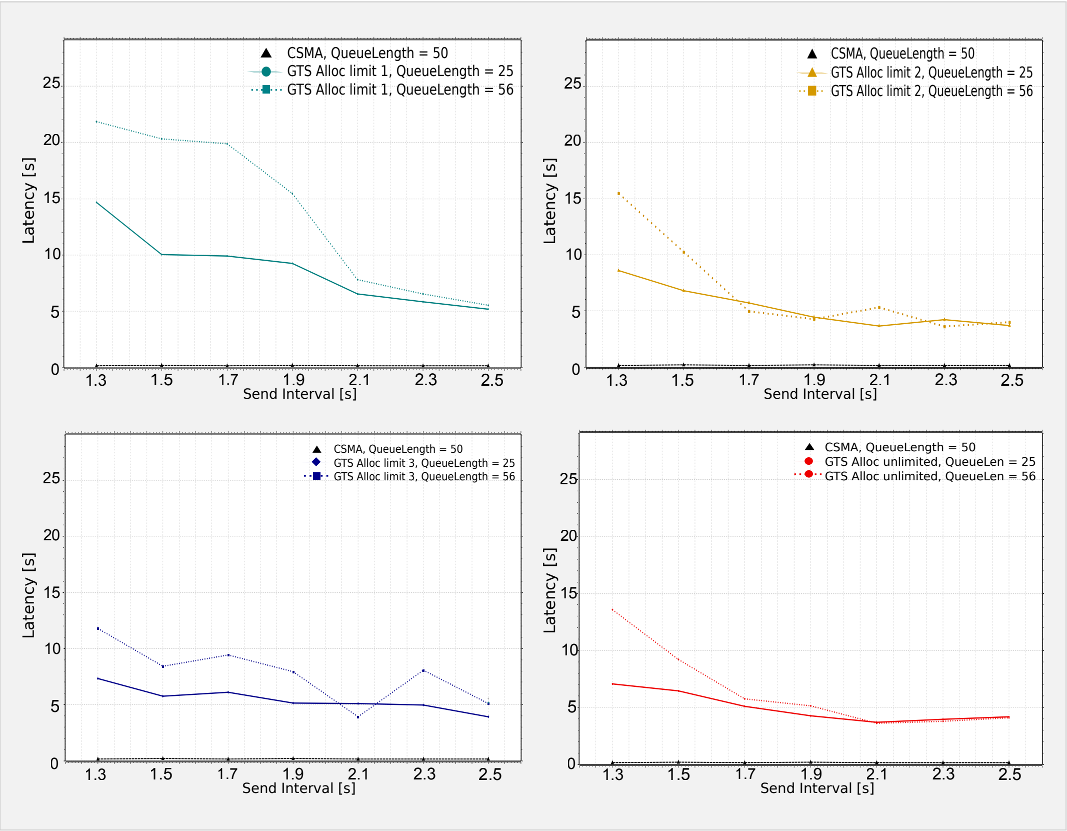


Figure 7.11: DSME Simulation: Maximal latencies for different GTS allocation limits

7.5 Future Work

In this work two considerable simplifications were used to implement a functional DSME MAC sublayer. Devices initially know their neighbors addresses and coordinates and do not send association requests to the PAN coordinator to retrieve a short address. Hence, the 64-bit extended address is used within every MAC header, which requires 4 more bytes which is not intended by the standard for most of the messages and especially within guaranteed time slots. If the association requests were extended with the geographical coordinates, these messages could also be used to passively discover neighboring devices. However, it is unlikely that every association request and response will be heard by all neighbors. Another possibility is to use the beacon slots, either by providing enough superframes such that every device may act as coordinator and broadcasts its beacon or by adding a list of neighboring devices to the payload of a beacon frame. With a superframe order of $SO = 3$ and the maximum beacon order of $BO = 14$, already 2048 beacon slots are available.

As discussed in the previous sections, it is crucial to achieve a complete allocation of the guaranteed time slots at the center node to improve the network performance. A fixed

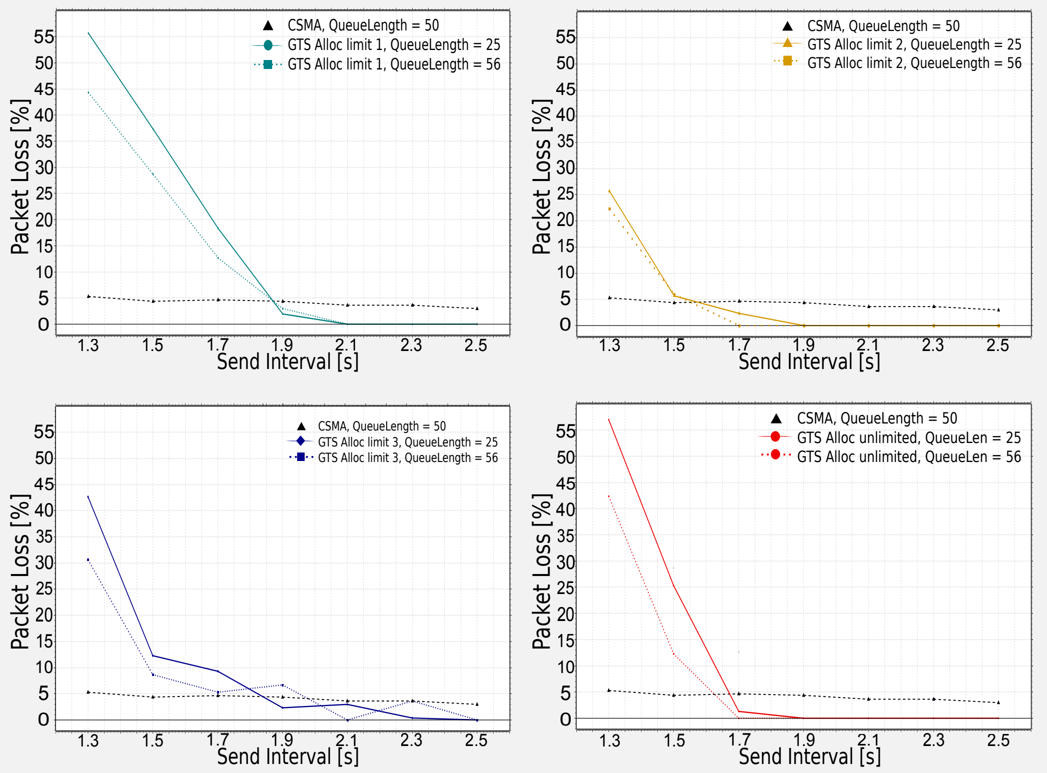
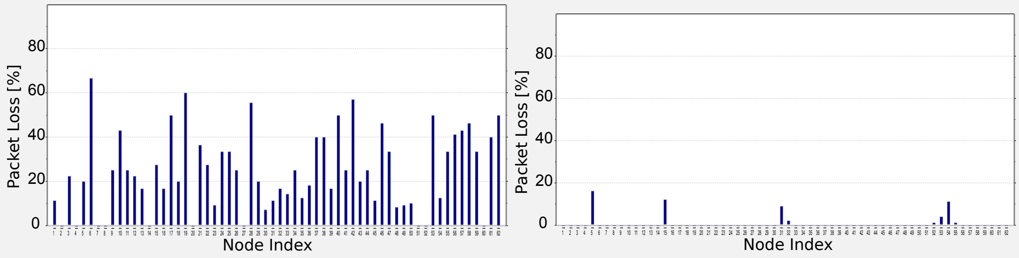


Figure 7.12: DSME Simulation: Maximal packet loss for different GTS allocation limits



(a) Sample Setup 1, $K = 25$, $t_{SI} = 1.5\,\mathrm{s}$

(b) Sample Setup 3, $K = 25$, $t_{SI} = 1.5\,\mathrm{s}$

Figure 7.13: DSME Simulation: Packet loss samples

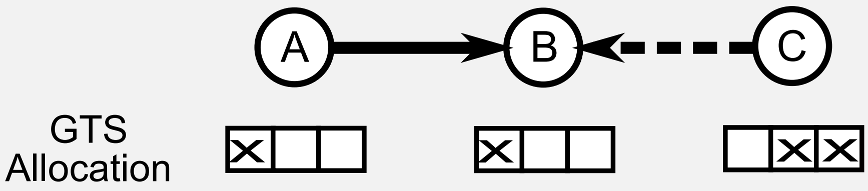


Figure 7.14: Example unallocatable slots

allocation scheme for the six inner nodes, where the requested time slot depends on the node identifier, would avoid a scenario as described in Section 7.4. Requests from outer nodes regarding these slots then would be denied or the next slot has to be selected. However, if the nodes would not independently send messages to the center, but only respond to a request from it, the center node could initially send the next request after receiving the current response. As the nodes on the inner rings did not allocate slots with outer nodes, they are capable to allocate the remaining slots of the center.

To further increase the throughput and reduce the latency, CAP reduction can be used. To maintain the setup time, CAP reduction could be activated after the setup phase. As the number of GTS within a superframe can be more than doubled, it is possible to duplicate the existing allocation into the additional slots, which reduces the amount of requests drastically. The latency might be additionally reduced by using an allocation scheme, which allocates transmit slots directly after a receiving slot, to allow an immediate forwarding of messages.

In Section 3.2 some protocol issues of DSME are discussed. As it is not explicitly stated in the standard, that slotted-CSMA has to be used within the CAP, the use of unslotted-CSMA could be further investigated. As mentioned in 2.1.2, this may reduce the collision probability for the CAP. Furthermore, problems related to the allocation of multiple slots and the deallocation of duplicate allocated slots were discussed, which should be further analyzed to find improvements.

Chapter 8

Conclusion

The target of this work is to achieve a reliable medium access control which enables packet loss probabilities close to zero in a multi-hop large scale wireless mesh network. For this purpose, current standards provide TDMA schemes to achieve a more robust communication, while maintaining or even increasing the throughput compared to the common CSMA-CA method. In a multi-hop network, time slot scheduling strongly depends on the routing protocol which determines the next hop devices. Thus a joint design of both layers is essential. Geographic routing is a promising routing protocol for large scale wireless mesh networks, as it utilizes decentralized hop-by-hop forwarding without requiring a large routing table. This allows to quickly adapt to changes of the wireless neighborhood or traffic demands. Hence, a decentralized time slot scheduling is in favor over a centralized scheduling. The deterministic synchronous multi-channel extension of the IEEE 802.15.4 standard provides such a protocol, which manages time slot allocation and channel assignment. Devices keep track of the allocated slots of their neighbors and request free slots directly from the destination device.

With TDMA, packet loss only occurs if the queue is saturated and further messages are blocked. To prevent the queue from saturating quickly, the service rate needs to be higher than the arrival rate and the queue needs to be large enough to achieve a desired blocking probability. The service rate is determined by the number of allocated transmit slots. These settings also have a strong influence on the latency as messages have to wait in the queue and until the next time slot.

To prevent single nodes from being forwarding hotspots, which would be unable to attain a sufficient amount of time slots, the routing algorithm has to be adapted to the network topology. The considered topology in this work is a mesh build up of nodes distributed on concentric circles around a center node. Using the original approach to select the neighbor which is nearest to the destination leads to an unequal distribution of child nodes and forwarding hotspots. This can be avoided by selecting the neighbor which is closer to the destination and also nearest to the straight line between source and destination.

The performance of DSME was analyzed analytically considering the superframe structure to determine the maximum throughput and latency in a multi-hop network. The M/D/1/K queueing model was described to determine the required queue size to achieve a minimum packet loss and estimate the average latency. Additionally a simplified model was proposed to evaluate the maximal latency.

To further analyze the performance of DSME and verify the ability of achieving a minimal packet loss using DSME and geographic routing, a simulation model has been implemented using the OMNeT++ simulator and the INET framework. The simulation results have shown that a fundamental DSME implementation can achieve a higher reliability compared to CSMA-CA in terms of packet loss if the child nodes and time slots are distributed fairly and the overall traffic demand is limited. The throughput mainly depends on this limitation and is comparable for both mechanisms. However, the throughput is far below the potential of DSME, because the CAP reduction was not used and the utilized random slot allocation pattern is not capable of completely allocating the available slots, especially at the critical center node. This can be improved if the center node initiates all of the traffic or with predefined allocation schemes for critical areas. The latency suffers from the delay caused by the queue and the gaps between time slots, thus DSME can not achieve a comparable small latency as CSMA. However, it can be minimized by achieving a service rate greater than the arrival rate through allocating sufficient transmit slots. Furthermore, the time slots should be aligned to the receiving time of messages to reduce the forwarding delay.

The results of this work have shown that the DSME protocol in conjunction with geographic routing builds up a great basis for a reliable large scale wireless mesh network.

Bibliography

[AP14] N. Accettura and G. Piro. Optimal and secure protocols in the IETF 6TiSCH communication stack. In *Industrial Electronics (ISIE), 2014 IEEE 23rd International Symposium on*, pages 1469–1474, June 2014.

[CMN10] Deji Chen, Aloysius Mok, and Mark Nixon. *WirelessHART: Real-Time Mesh Network for Industrial Automation*. Springer Science+Business Media LLC, Boston, MA, 2010.

[DCACM03] Douglas S. J. De Couto, Daniel Aguayo, Benjamin A. Chambers, and Robert Morris. Performance of Multihop Wireless Networks: Shortest Path is Not Enough. *SIGCOMM Comput. Commun. Rev.*, 33(1):83–88, January 2003.

[ET90] Anthony Ephremides and T.V. Truong. Scheduling broadcasts in multihop radio networks. *Communications, IEEE Transactions on*, 38(4):456–460, Apr 1990.

[FGG06] Qing Fang, Jie Gao, and LeonidasJ. Guibas. Locating and Bypassing Holes in Sensor Networks. *Mobile Networks and Applications*, 11(2):187–200, 2006.

[Fin87] Gregory G Finn. Routing and Addressing Problems in Large Metropolitan-Scale Internetworks. *ISI Research Report*, 1987.

[Fou14a] HART Communication Foundation. HART Communication Protocol. http://en.hartcomm.org/main_article/hart_protocol.html, 04.10.2014. Last visited: 20.11.2014.

[Fou14b] HART Communication Foundation. WirelessHART. http://en.hartcomm.org/main_article/wirelesshart.html, 04.10.2014. Last visited: 20.11.2014.

[fS14] DIN German Institute for Standardization. Industrial communication networks - Wireless communication network and communication profiles - WirelessHART TM (IEC 65C/744/CD:2013), 2014.

[GOW07] Olga Goussevskaia, Yvonne Anne Oswald, and Rogert Wattenhofer. Complexity in Geometric SINR. In *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '07, pages 100–109, Montreal, Quebec, Canada, New York, NY, USA, 2007. ACM.

[HL86] Ting-Chao Hou and V.O.K. Li. Transmission Range Control in Multihop Packet Radio Networks. *Communications, IEEE Transactions on*, 34(1):38–44, Jan 1986.

[HL07] Ekram Hossain and Kin Leung. *Wireless Mesh Networks*. Springer US, Boston, MA, 2007.

[HN14] Kwang-il Hwang and Sung-wook Nam. Analysis and Enhancement of IEEE 802.15.4e DSME Beacon Scheduling Model. *Journal of Applied Mathematics*, 2014(3):1–15, 2014.

[IEE06] IEEE Computer Society. *IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements: Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*. Institute of Electrical and Electronics Engineers, New York, 2006.

[IEE11] IEEE Computer Society. *IEEE standard for local and metropolitan area networks: Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*. Institute of Electrical and Electronics Engineers, New York, 2011.

[IEE12] IEEE Computer Society. *IEEE standard for local and metropolitan area networks: Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs): Amendment 1: MAC sublayer*. Institute of Electrical and Electronics Engineers, New York, 2012.

[IEE13] IEEE Computer Society. *IEEE standard for local and metropolitan area networks: Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs): Amendment 5: Physical Layer Specifications for Low Energy, Critical Infrastructure Monitoring Networks*. Institute of Electrical and Electronics Engineers, New York, 2013.

[ine] INET Framework. https://inet.omnetpp.org/. Last visited: 25.06.2015.

[ISA09] ISA. ISA-100.11a-2009 Standard. Wireless systems for industrial automation: Process control and related applications, 2009.

[KHPD08] Anna N. Kim, Fredrik Hekland, Stig Petersen, and Paula Doyle. When HART goes wireless: Understanding and implementing the WirelessHART standard. In *Factory Automation (ETFA 2008)*, pages 899–907, 2008.

[KK00] Brad Karp and H. T. Kung. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, MobiCom ’00, pages 243–254, Boston, Massachusetts, USA, New York, NY, USA, 2000. ACM.

[Kon10] Kiran Kumar Koneri. Implementation of Collection Tree Protocol over WirelessHART Data-Link. Master’s thesis, Jönköping University, 2010.

[KSCV06] P. Kyasanur, J. So, C. Chereddi, and N.F. Vaidya. Multichannel mesh networks: challenges and protocols. *Wireless Communications, IEEE*, 13(2):30–36, April 2006.

[KSU99] Evangelos Kranakis, Harvinder Singh, and Jorge Urrutia. Compass Routing on Geometric Networks. In *In Proc. 11 th Canadian Conference on Computational Geometry*, pages 51–54, 1999.

[LBB05] Seungjoon Lee, Bobby Bhattacharjee, and Suman Banerjee. Efficient Geographic Routing in Multihop Wireless Networks. In *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc ’05, pages 230–241, Urbana-Champaign, IL, USA, New York, NY, USA, 2005. ACM.

[LJ12] Junhee Lee and Wun-Cheol Jeong. Performance analysis of IEEE 802.15.4e DSME MAC protocol under WLAN interference. *ICT Convergence (ICTC), 2012 International Conference on*, pages 741–746, 2012.

[LJDC$^+$00] Jinyang Li, John Jannotti, Douglas S. J. De Couto, David R. Karger, and Robert Morris. A Scalable Location Service for Geographic Ad Hoc Routing. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, MobiCom ’00, pages 120–130, Boston, Massachusetts, USA, New York, NY, USA, 2000. ACM.

[Mac11] J. MacGregor Smith. Properties and performance modelling of finite buffer M/G/1/K networks. *Computers & Operations Research*, 38(4):740–754, 2011.

[MMW09] Subhas Chandra Misra, Sudip Misra, and Isaac Woungang. *Guide to Wireless Sensor Networks*. Springer-Verlag London, London, 2009. Includes bibliographical references and index.

[MPSP10] Panneer Muthukumaran, Rodolfo de Paz, Rostislav Spinar, and Dirk Pesch. MeshMAC: Enabling Mesh Networking over IEEE 802.15.4 through Distributed Beacon Scheduling. In Jun Zheng, Shiwen Mao, ScottF. Midkiff, and Hua Zhu, editors, *Ad Hoc Networks*, volume 28 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 561–575. Springer Berlin Heidelberg, 2010.

[NS08] Chee-Hock Ng and Boon-Hee Soong. *Queueing Modelling Fundamentals*. John Wiley & Sons, Ltd, Chichester, UK, 2008.

[PD11] Parth H. Pathak and Rudra Dutta. A Survey of Network Design Problems and Joint Design Approaches in Wireless Mesh Networks. *IEEE Communications Surveys & Tutorials*, 13(3):396–428, 2011.

[SÍ1] Juan Héctor Sánchez. WirelessHART Network Manager - Software Design and Architecture. Master's thesis, KTH Technology and Health - The Royal Institute of Technology, 2011.

[Seo14] Dong-Won Seo. Explicit Formulae for Characteristics of Finite-Capacity M/D/1 Queues. *ETRI Journal*, 36(4):609–616, 2014.

[SHM$^+$08] Jianping Song, Song Han, Al Mok, Deji Chen, Mike Lucas, Mark Nixon, and Wally Pratt. WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control. In *2008 IEEE Real-Time and Embedded Technology and Applications Symposium*, pages 377–386. Institute of Electrical and Electronics Engineers, 2008.

[Ska12] Anders Asperheim; Rune V. Sjøen; Kaja F. L. Skaar. Design and Implementation of a Rudimentary WirelessHART Network. Technical report, University of Oslo, Department of Informatics, 2012.

[SVV15] Aggeliki Sgora, Dimitrios J. Vergados, and Dimitrios D. Vergados. A Survey of TDMA Scheduling Schemes in Wireless Multihop Networks. *ACM Comput. Surv.*, 47(3):53:1–53:39, April 2015.

[TK84] H. Takagi and L. Kleinrock. Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals. *Communications, IEEE Transactions on*, 32(3):246–257, Mar 1984.

[VH08] András Varga and Rudolf Hornig. An Overview of the OMNeT++ Simulation Environment. In *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*, Simutools '08, pages 60:1–60:10, Marseille, France, ICST, Brussels, Belgium, Belgium, 2008. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[VRGB15] E. Vogli, G. Ribezzo, L.A. Grieco, and G. Boggia. Fast join and synchronization schema in the IEEE 802.15.4e MAC. In *Wireless Communications and Networking Conference Workshops (WCNCW), 2015 IEEE*, pages 85–90, March 2015.

[WJ12] Wun-Cheol Jeong and Junhee Lee. 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*, pages 7–12, 2012.

[WLZ09] Feng Wang, Dou Li, and Yuping Zhao. Analysis and Compare of Slotted and Unslotted CSMA in IEEE 802.15.4. In *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pages 1–5, Sept 2009.

[WT10] Matthias Witt and Volker Turau. Robust and low-communication geographic routing for wireless ad hoc networks. *Wireless Communications and Mobile Computing*, 10(4):486–510, 2010.

[XXS$^+$13] Xuecheng Liu, Xiaoyun Li, Shijuan Su, Zhenke Fan, and Gang Wang. Enhanced Fast Association for 802.15.4e-2012 DSME MAC Protocol. *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*, 2013.

[YWK05] Yaling Yang, Jun Wang, and Robin Kravets. Designing routing metrics for mesh networks. In *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2005.

[Zim80] H. Zimmermann. OSI Reference Model–The ISO Model of Architecture for Open Systems Interconnection. *Communications, IEEE Transactions on*, 28(4):425–432, Apr 1980.

List of Figures

Appendix A

Content of the CD

The directories found on the attached Compact Disc are described in Table A.1.

Filename	**Description**
Directory	Folder
thesis.pdf	This document
/latex/	Latex sources of this document
/matlab/	Matlab sources used in Chapter 4
/omnet/	OMNeT++ Sources used for network simulation

Table A.1: Disc content