Seamless roaming and guaranteed communication using a synchronized single-hop multi-gateway 802.15.4e TSCH network

Jetmir Haxhibeqiri[a,1], Abdulkadir Karaagac[a], Ingrid Moerman[a], Jeroen Hoebeke[a]

Technologiepark Zwijnaarde 15, 9000 Ghent, Belgium

[a]*IDLab, Department of Information Technology at Ghent University - imec*

Abstract

Industrial wireless sensor and actuator networks (WSANs) are being used to improve the efficiency, productivity and safety of industrial processes. One open standard that is commonly used in such cases is IEEE 802.15.4e. Its Time-Slotted Channel Hopping (TSCH) mode employs a time synchronized based medium access control (MAC) scheme together with channel hopping to alleviate the impact of channel fading and interference. Until now, most of the industrial WSANs have been designed to only support static nodes and are not able to deal with mobility. In this paper, we show how a single-hop, multi-gateway IEEE 802.15.4e TSCH network architecture can tackle the mobility problem. We introduce the Virtual Grand Master (VGM) concept that moves the synchronization point from separated Backbone Border Routers (BBRs) towards the backbone network. With time synchronization of all BBRs, mobile nodes can roam from one BBR to another without time desynchronization. In addition to time synchronization, we introduce a mechanism to synchronize the schedules between BBRs to support fast handover of mobile nodes. We show that with the proposed network architecture handovers happen instantly without any packet losses, while the handover time can be up to tens of seconds without any time synchronization between BBRs. The solution is evaluated in

---

[1]Corresponding author: jetmir.haxhibeqiri@ugent.be

a testbed setting as well as in a real industrial environment.

1. Introduction

Under the umbrella of the Internet of Things (IoT) vision, a large number of objects are being connected to the Internet, sharing a huge amount of data that is consumed by applications. Apart from IoT applications in less criti-
5 cal domains such as home automation, IoT technologies are increasingly being utilized in industrial settings. This Industrial IoT will help to improve the productivity, quality and robustness of the industrial processes. At the same time, the IIoT will increase the connectivity of devices and people in the factory floor, improving safety for workers, process efficiency as well as productivity. In many
10 industrial applications such as process control, factory automation, warehousing operations, IIoT is considered as a key enabling technology.

Industrial wireless sensor and actuator networks (WSANs) are used to report measurement information to a central controlling point. After processing the collected information, it may inform actuator points to trigger actions. Such
15 communication might be critical, requiring high reliability up to 99% in a harsh industrial environment. In some cases, determinism in terms of latency as well as low power usage are required too. In order to meet these requirements, Time Synchronized Channel Hopping (TSCH) became the main medium access control (MAC) technique used in industrial applications since its adoption by
20 the WirelessHART [1] standard in 2007. The new amendment of the IEEE 802.15.4-2011 [2] standard for low-power networks, 802.15.4e [3], amends the MAC protocol with Time-Slotted Channel Hopping (TSCH) mode. Going one step further in the convergence of IP-based networks and industrial networks, the IETF working group of 6TiSCH is investigating end-to-end IPv6 connectiv-
25 ity over the TSCH mode of the IEEE 802.15.4e protocol [4].

So, WirelessHART and IEEE 802.15.4e have interesting properties to offer determinism in terms of latencies. However, they have been mainly designed

2

for static networks without any node mobility. Consequently, they have difficulties to cope with network dynamics. For IEEE 802.15.4e, at first hand, this comes as a problem of time synchronization between nodes as well as time slot scheduling. Once a node is desynchronized from its master due to mobility, it needs to resynchronize with another node, a process that is both power and time consuming. Moreover, with the selection of a new master node, the mobile node needs to update its schedule accordingly. In case one wants to combine the advantages of IEEE 802.15.4e with mobility support, one needs to revisit the architectural design of the protocol. This is exactly the motivation for this work, which considers a use case that requires such mobility support.

Our use case under consideration, is a "shuttle" system, consisting of robots that move in 2 dimensions within the storing racks of a warehouse in order to transport and store good. These shuttles report about their behavior (sensing) and are controlled remotely (actuation). This sensing and actuation requires reliable and real-time connectivity. In [5] we already showed the environmental difficulties through measurements in a real warehouse in terms of link reliability and latency. In addition, we launched some initial ideas for the proposed network solutions too, leveraging on the deterministic capabilities of 802.15.4e.

In this paper, one of the proposed architectures, namely a single-hop multi-gateway 802.15.4e TSCH network is designed and evaluated. This novel architecture combines the advantages of TSCH with support for mobility. More concretely, the following contributions are made. We present a single-hop 802.15.4e network architecture with multiple 802.15.4e gateways and mobile nodes. We introduce the network grand master concept in order to assure smooth and real-time handovers from one connecting point to another of the IEEE 802.15.4e mobile node. This concept enables the network to remain fully time synchronized from a grand master in the network, thus making possible for the node to freely roam in the network area without time desynchronization. In addition, three different scheduling schemes to cope with different scalability needs are presented. Finally, the architecture is validated, first using a real-life testbed and then in a real industrial environment.

The rest of the paper is organized as follows: in Section 2 the problem statement is discussed. Section 3 gives an introduction to the IEEE 802.15.4e TSCH mode. Related work is presented in Section 4, while Section 5 details the system design dealing with the network architecture, node synchronization and traffic management. Section 6 gives a mathematical model to evaluate the worst case RTT latency for one of the proposed scheduling schemes. Section 7 presents the results obtained for both the deployment in a testbed and in a real industrial environment. Finally, Section 8 concludes the paper.

2. Problem Statement

Industrial environments are harsh environments characterized by metallic structures that obstruct the wireless communication. Different propagation effects are present in these environments such as shielding, scattering, diffraction or absorption effects. They result in difficulties for wireless communication towards achieving required reliability.

Within such a challenging environment, we consider an automated warehouse system based on mobile intelligent self-contained transport vehicles, 'shuttles'. These shuttles can move in 2D within the storing racks of a warehouse and are able to store goods in a very compact way. In addition to the harsh environment, the 2D shuttles can drive at reasonably high speed up to 3 m/s under the pallets of stored goods at certain heights.

Considering the possibility of dense storage of goods, a multi access point (AP) networking system needs to be considered for covering the whole racking system. However, due to the relatively high speed of the 2D shuttles, the use of a multi-AP 802.11 network might result in frequent handovers that will increase the overall communication latency, or even worse, break the communication link for some time. In addition, 802.11's CMSA/CA mechanism is unable to deliver guarantees with respect to the deterministic delivery of data.

Due to their mobility, shuttles need to be battery powered, only being charged during inactive periods of time. Finally, shuttles always have to be

4

connected to a central server via a reliable and real-time wireless communication infrastructure in order to provide timely status updates and to receive order assignments.

Considering all of the aforementioned challenges, the optimal network solution for such a case needs to provide a handover latency that is lower than 1s [6], capacity requirements up to several bps, and the possibility to serve up to 100 shuttles at a time. The data packets will be 8 to 128 bytes with a maximal communication frequency of 1Hz in uplink. In addition to frequent uplink monitoring data, infrequent transmission of picking orders (downlink data) can happen. The reliability is targeted to be as high as 99% in a controlled industrial environment. All of these requirements have been provided by a company that provides logistic solutions for warehouses, as part of the HYCOWARE research project.

3. IEEE 802.15.4e TSCH

The IEEE 802.15.4e standard [3] has been published in 2012 as an amendment of the IEEE 802.15.4-2011 MAC protocol [2]. Three different MAC protocols are proposed: Low Latency Deterministic Networks (LLDN), Time Slotted Channel Hopping (TSCH) and Deterministic and Synchronous Multi-channel Extension (DSME). Compared to IEEE 802.15.4-2011, the new standard IEEE 802.15.4e overcomes the drawbacks such as high power consumption, fading and interference. This is achieved by adopting a time synchronization based MAC scheme for low power usage and channel hoping to alleviate the impact of channel fading and interference.

In TSCH mode, time is sliced in time slots and time slots are organized in slot frames. The standard does not impose the length of the time slot or the number of time slots in a slot frame. However, the time slot needs to be long enough to be able to transmit a packet and to receive an acknowledgement (ACK). The slot frame length is determined based on power usage and throughput trade-off. When slot frames are long then the node's power consumption will decrease at
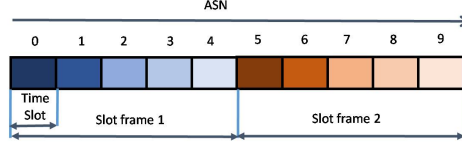
5

Figure 1: Organization of slot frames.

the expense of lower throughput and vice versa. The duration of a slot frame can be from 10s up to 100s of time slots and it repeats periodically. TSCH mode also defines the Absolute Time Slot Number (ASN) that initializes with the network formation and increments with each time slot. The organization of time slots and frames is shown in Figure 1.

Apart from time slotting, TSCH includes channel hopping too. There are 16 channels available that are identified by *channelOfsset*, an integer value in the range [0,15]. Each link in TSCH mode is represented by its time slot offset inside the frame, n, and *channelOffset*. The actual frequency to be used for transmission or reception is calculated as follow:

$$F = (ASN + channelOffset)\%N \qquad (1)$$

where N is the total number of available channels and $\%$ note modulo operation.

The node joining process, required for the formation of the network, is based on active beaconing. When a node wants to join the network it waits for Enhanced Beacon (EB) frames, frames that are transmitted by other nodes to advertise the network. EBs are transmitted periodically and contain information about the time slot length, frame length, ASN and the time slot during which the beaconing node will listen. In order to minimize the average of joining time, different approaches to schedule EB transmissions are proposed. In [7], a random based algorithm is proposed where each node will transmit EBs with a probability p_{EB} on a pre-assigned link. The probability is calculated as such to minimize the collision probability. In [8], they propose two algorithms: Random Vertical filling (RV) and Random Horizontal filling (RH). In both algorithms the network coordinator will transmit an EB in the first advertisement

6

slot using channel offset 0. In the RV algorithm other nodes will send EBs in the same slot using a random channel, whereas in the RH algorithm they will use channel offset 0 with a random slot offset.

A packet is transmitted exactly at *TxOffset* time after the start of the time slot, while the receiving node will start listening *GuardTime* before. This allows a slight desynchronization between nodes, which cannot become larger than *GuardTime*. Due to these strict timings for transmission and reception, nodes need to be accurately synchronized with each other. Two types of synchronization between the node and its clock master are foreseen in the 802.15.4e TSCH standard: the packet-based and acknowledgement-based synchronization.

In the packet-based synchronization approach, every time the slave node receives a packet from its time master, it checks the received time. If the packet was received after *TxOffset* time, the node will decrease the next time slot for the (*RxTime - TxOffset*) difference. If the packet was received before *TxOffset* time, then the slave will increase the next time slot for the (*TxOffset - RxTime*) difference. This way, the slave node makes sure its timing is aligned with its master.

In many cases there is only uplink traffic, making packet-based synchronization impossible due to the absence of downlink packets from the master to its slaves. In this case, ACK-based synchronization will be used. In the ACK-based synchronization approach, the time difference between the time instant the packet was received and the *TxOffset* is measured at the master side. This information is communicated to the slave node via the ACK packet.

In TSCH mode, nodes communicate with each other based on a schedule of time slots and channel hopping. The schedule is either predefined, centralized or can be negotiated between nodes at the time when nodes join the network. By assigning a single time slot for one-way communication between a couple of nodes, collision-free transmission is achieved. The power usage is related with the number of active time slots per node, enabling nodes to go off during the non-active time slots. On the other hand, channel hopping protects communication against the frequency fading that is common in industrial environments. Thus

TSCH based MAC protocols offer determinism based on the structure of the schedule on each node, low-power usage and robustness against channel fading. These area all features that are needed for an industrial network application.

In [9] and [10] two literature reviews are presented. The first one [9] gives an introduction to IEEE 802.15.4e networks and performs a literature review on different aspects of TSCH mode ranging from node synchronization to traffic scheduling problems. The authors of [10] group different industrial applications based on their requirements and discuss different WSN standards that can satisfy those requirements.

4. Related Works

There are different studies with respect to different aspects of IEEE 802.15.4e TSCH networks such as network synchronization, network formation, node mobility support, traffic scheduling and power consumption. Until now, most of the studies on IEEE 802.15.4e and the application of WSNs in industrial environments are related to static node deployment cases. Here, we will only present works related to node mobility support and industrial applications.

In [11] a novel solution for routing in TSCH networks in the presence of mobile nodes is presented. The network is composed of a number of static nodes, called anchor nodes, and mobile nodes. Routing between anchor nodes is done using RPL [12] and the anchors' positions are known to the mobile nodes. Every mobile node estimates its distance to each of the anchor nodes and selects as its parent the anchor that minimizes the number of expected transmissions (ETX) towards the sink. Work in [13] extends the work done in [11] by using Kalman filtering to reduce the impact of channel dynamics on the position estimation of the mobile node.

In [14], a mobility-aware TSCH framework is proposed. The MTSCH framework accelerates the node association process and minimizes the latency caused by disassociation. In order to accelerate the node association process, MTSCH uses ACK messages as passive beacons. At the end of each slot frame, a node

8

will send a group ACK to acknowledge all received packets from all of its neighbours together with information when the node will listen for any mobile nodes. Moreover, the ACKs are sent using a specific channel offset. MTSCH improves the joining time in range from 3% to 50%.

In [15] a network architecture that supports mobility for wireless sensor networks (WSN) is presented. The slot frame is divided in two parts: a CSMA/CA part and a TDMA part. For communication between mobile nodes and static ones, a CSMA/CA based scheme is used, while the transmission between static nodes is done in fixed time slots. As its anchor towards the network, the mobile node will select the static node that has the highest RSSI value, as retrieved from its beacon transmissions. In [16] a similar approach was used, also dividing the slot frame in two parts. However, instead of CSMA/CA, mobile nodes will select random access mini-slots for sending their join requests transmissions. The random access mini-slots part allows to avoid collisions between data transmissions and join requests, offering faster network convergence.

In [17] authors present a TDMA MAC approach for low power networks based on low power listening (LPL) and distributed queuing (DQ). The network is a single-hop network with a single coordinator and multiple mobile nodes. The communication process is divided into two phases: a network synchronization phase and a data communication phase. During the network synchronization phase, a coordinator transmits wake-up packets with a certain frequency and using a specific channel. Mobile nodes by default are on low power listening mode and will wake up to listen for wake-up packets. Once the mobile node receives such a wake-up packet, it synchronizes to the network. The data communication period uses a time-fixed frame structure for communication.

In [18] handover reliability in an industrial WSN is improved by offering a more accurate handover triggering mechanism. This handover triggering mechanism uses following metrics: moving state, channel conditions and packet delivery ratio. They achieve a handover reliability of up to 98%, but do not provide further results on the handover timing [18]. Albeit an improved handover triggering mechanism will have an impact on decreasing the handover

9

timing, additional techniques that also incorporate the update of the schedules of the mobile nodes have to be considered. This is a key contribution of our work, where mobile nodes can continue to use the same schedule even when it changes its point of attachment to the network.

In [6] a performance estimation of TSCH networks as a whole is given. Different metrics such as latency, power consumption, throughput, reliability are considered for the performance tests. Also requirements for different use cases are given.

Other solutions in literature for low power mobility consider the use of Bluetooth Low Energy (BLE) advertisements [19], which are broadcast packets. These solutions currently focus on health-care applications for residential activity monitoring systems [20]. Compared to such a BLE beacon advertisement based solution offered in [20] our solution, which also leverages on broadcast packets for uplink traffic, generates these broadcasts in specific time slots avoiding collision possibility. In [20] advertisements from different end nodes can collide between each other as they might be sent at the same time.

Compared to [14] we overcome the disassociation problem by time synchronizing all the Backbone Border Routers (BBRs) from a network based grand master. This implies that the mobile node only associates once with the network, namely at network formation time. In [11] they do not show how the scheduling scheme of the mobile node will be updated once the mobile node will move towards another anchor. In our design, we show how the scheduling scheme will be updated in all BBRs, once the mobile node joins the network. Both mechanisms, time and schedule synchronization between BBRs, will improve the handover timing for mobile nodes. In comparison with [16], where the handover latency was bigger than 0.5s, we achieve better latencies that are lower than 100ms. In [16] they improve joining time of the node to the cluster in order to improve the handover time, while presented approach in this paper makes possible for the device to be connected all the time using the same schedule when the network access point is changed. In addition to that, the solution offer also determinism even for mobile nodes what can not be achieved with the

solution from [15].

5. System Design

As introduced before, for our use case, we target a single-hop multi-gateway IEEE 802.15.4e TSCH network architecture. In order for such an architecture to work and to enable seamless mobility and guaranteed deterministic communication, two major improvements are needed. First, mobile nodes need to be able to roam from one 802.15.4e gateway (or access point) to another without desynchronizing from the network. Second, the mobile node should be able to maintain its schedule when moving to the new access point, without any need to change. This new architecture and the study of both aspects is the main focus of this paper and will be discussed in detail in the following subsections.

5.1. Network architecture

The key components of the network architecture are shown in Figure 2. It involves the mobile nodes, the Backbone Border Routers (BBRs) and the Network Server (NS). All BBRs are connected via a wired backbone network with the Network Server (NS). The mobile nodes only use single hop wireless communication towards the backbone, with the BBRs serving as access points for the mobile nodes. As such, the network behaves as a single-hop Low power Lossy Network (LLN), but with multiple BBRs. The Network Server, which is the brain of the network, acts as the virtual root of this LLN, enabling time synchronization between the BBRs and performing traffic management. It ensures that all BBRs are listening at the same time to the same channel, that the schedules of all mobile nodes are known by all BBRs and that both uplink and downlink traffic are handled properly. More details are provided in the following subsections.

5.1.1. Backbone Border Routers (BBRs)

The BBRs are at the edge of the backbone network and serve as access points for the IEEE 802.15.4e mobile nodes. They serve as DODAG roots for
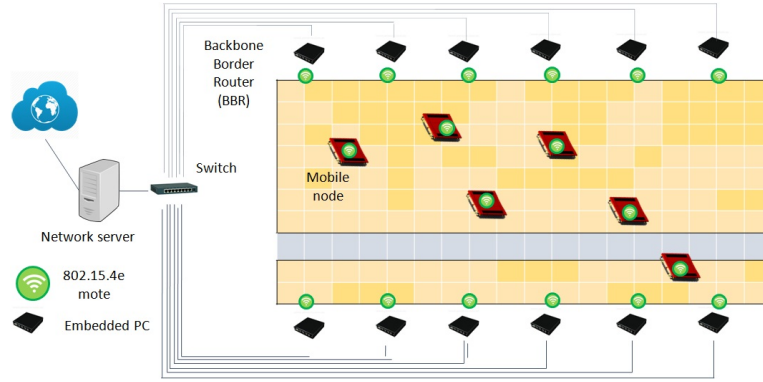
11

Figure 2: Single hop network architecture for 2D shuttle systems.

the mobile nodes that are connected to them and that are within their coverage zone. At the same time, they broadcast the IPv6 prefix to nodes that want to join the network.

For the nodes to be able to use the same IPv6 address throughout the network, all BBRs have to advertise the same subnet prefix. As such, the mobile nodes will use the same subnet prefix any time during their operation, resulting in a fixed public and local IPv6 extracted from their subnet prefix and their MAC. From the mobile node perspective, all BBRs will be seen as a single access point towards the backbone network regardless of their position in the network. This is the first key enabler to achieve seamless mobility.

A backbone border router will perform IPv6 header compression for packets routed towards the mobile nodes, and decompression for the packets routed from the LLN network towards the backbone network. For the packets that are going from the backbone network to a mobile node, the IPv6 header is mapped to a 6LowPAN header and the physical address of the next hop is determined. As there is only one hop between a BBR and a mobile node, the physical address of the next hop can be directly derived from the IPv6 destination address.

Apart from using a single IPv6 address for the mobile nodes, BBRs need to be time synchronized in order to offer smooth handovers for the mobile nodes. This time synchronization between BBRs is done via the backbone network.

12

Another enabler for seamless mobility is the traffic scheduling inside the LLN. To decrease the amount of signaling traffic for routing (such as Neighbor Discovery traffic) inside the LLN network, mobile nodes will send layer three unicast packets encapsulated in layer two broadcast packets. BBRs within range will receive the broadcast packets and will route them to the Network Server. Downlink traffic will be scheduled from the Network Server as unicast packets, by selecting the 'best' BBR for the downlink stream.

5.1.2. Network Server (NS)

The Network Server is the heart of the backbone network. It acts as a Virtual Grand Master (VGM) for the whole network. Based on its time, all other nodes, the BBRs and mobile nodes, are being synchronized. In addition to network time synchronization, the Network Server performs deduplication of upstream packets from multiple BBRs and selects the best BBR for the downstream communication. This selection can be based on the RSSI or SNR values of the last couple of upstream packets. This information is collected by BBRs and is sent to the Network Server. The Network server keeps track of the last BBRs via which the mobile node was reachable.

The last function performed by the Network Server is the management and update of the BBRs' TSCH schedule. Once a mobile node has negotiated a schedule with its root BBR, the Network Server will ask other BBRs to also install that schedule for that node. The root BBR represents the BBR via which the schedule negotiation is being done.

5.2. Synchronization of the backbone and TSCH network

One of the key enablers to achieve smooth and fast handovers of mobile nodes in our IEEE 802.15.4e TSCH network is an improved time synchronization process. The synchronization process is time and power consuming for each node in an IEEE 802.14.4e TSCH network. During unsynchronized periods, a node needs to keep its radio on for resynchronization purposes, which increases its power consumption. On the other hand, if radio duty cycling is used during the

13

synchronization process to decrease the power consumption, the time it takes
to complete the synchronization process will be increased.

Due to mobility, our mobile nodes would desynchronize every time they move
from the coverage zone of one BBR to the coverage zone of another BBR. If
they had to resynchronize every time they change BBR, this would result in long
periods without connectivity. As such, the main challenge for handling mobility
in real-time is to achieve continuous time synchronization of the mobile nodes
with all BBRs.

To improve mobility handling and to reduce the synchronization time, we
therefore propose BBRs to be time synchronized between each other by means of
the NS. Once all BBRs are time synchronized, the mobile node does not need to
resynchronize with a new BBR once it moves to its coverage zone. This requires
a mapping between the time mechanism used in the 802.15.4e TSCH network
and the one in the backbone network, as will be explained in the following
subsections.

5.2.1. Synchronization in 802.15.4e TSCH Networks

In the current implementation of the 802.15.4e TSCH network in OpenWSN
[21], each DODAG part of the LLN will be synchronized based on its DODAG
root time. The DODAG root is the time master of the network as it does not
synchronize with any other node in the network. Different DODAGs will have
different time masters, consequently being unsynchronized between each other.
All other slave nodes have to align the boundary of their time slots with their
time masters.

As we discussed in the technology introduction, in Section 3, there are two
types of synchronization between a slave node and its master, namely packet-
based and acknowledgement-based synchronization. Regardless the adopted
synchronization strategy, data packets are always used for synchronization. In
cases when data exchanges are rare, then 'keep alive' messages are exchanged
to keep the synchronization.

In IEEE 802.15.4e there is no concept of absolute timing, but the time is

14

measured in the elapsed number of $'clock\ ticks'$ since the start time of the last

370 time slot. So the time unit is $'clock\ ticks'$ and not seconds. The time slots are aligned between a master and its slave node using the ASN number. So except time synchronization, the slave node also needs to synchronize the ASN number with its master.

As the backbone network time is expressed in seconds, there should be a

375 mechanism to translate seconds in $'clock\ ticks'$ and ASN number to make it understandable for the IEEE 802.15.4e node of the BBR.

5.2.2. Virtual Grand Master Synchronization

At the backbone network side, the BBR's network initialization time is taken as a time reference to calculate the ASN. This reference time is communicated

380 to the BBRs by the Network Server. This makes it possible to add new BBRs to the network without resetting the whole network and loose synchronization. The network keeps calculating the elapsed ASN number and the $'clock\ ticks'$ (as seen by the 802.15.4e node) based on this time reference. This information is communicated periodically to the BBRs by the Network Server. By moving

385 the clock master of the mobile nodes from separate BBRs towards the backbone network, all nodes are now synchronized with a single time source. This is referred to as Virtual Grand Master (VGM) synchronization.

5.3. Handling of upstream and downstream data traffic

The Network Server is responsible for managing the data traffic coming from

390 and going to the LLN network.

Upstream traffic from a mobile node can be received by multiple BBRs simultaneously, as all BBRs are synchronized and will listen to the same channel at the same time (see the next subsection). This does not mean that the fading alleviation flexibility reached by channel hopping is lost. Still, the channel will

395 change over time as the actual channel in TSCH network is determined by the channel offset and ASN. BBRs will forward the traffic to the Network Server, which has to perform deduplication in order not to send the packet multiple

15

times to the final destination. To this end, the Network Server maintains a deduplication table. If a packet is received for the first time, a hash value of the packet is calculated. This hash value is stored in the table and the packet is forwarded to its final destination. If a duplicate packet arrives, the calculated hash value will already be present in the table. In that case, the packet is discarded. Each entry in the table has a timer and will be removed when this timer expires.

Next to this, the Network Server also has a BBR table that keeps track via which BBRs each of the mobile nodes in the network can be reached. This table is updated upon the reception of upstream traffic from BBRs. It also keeps track of the RSSI values of the received upstream packets. This table is then used for routing downstream traffic. In case multiple BBRs are available for a single mobile node, the best BBR can be selected. Next, the packet is unicasted by the NS to the selected BBR. Each entry of the BBR table has a timer that indicates how long it is valid. Once the timer elapses the entry is cleared. In case downstream traffic arrives at the Network Server and no entry is yet present in the table, the NS will broadcast the packet to all BBRs. Upon the discovery phase each BBR that discovers the mobile node will inform the NS for the RSSI value in uplink from the end node. At this end, the BBR table in the NS will be updated. Then only the BBR that has already discovered the mobile node and has the highest RSSI value in the uplink from the mobile node, can relay the data packet. This will ensure that the data packet is not collided in downlink.

5.4. Schedule Organization

Next to this, the NS also plays a role in the scheduling process, resulting in the exchange of signaling traffic with the BBRs.

In every TSCH network, scheduling plays an important role. The schedule determines the trade-off between the power usage and achievable throughput of a certain node. When more time slots are scheduled for communication with a node, the throughput of that node will be increased, at the expense of an
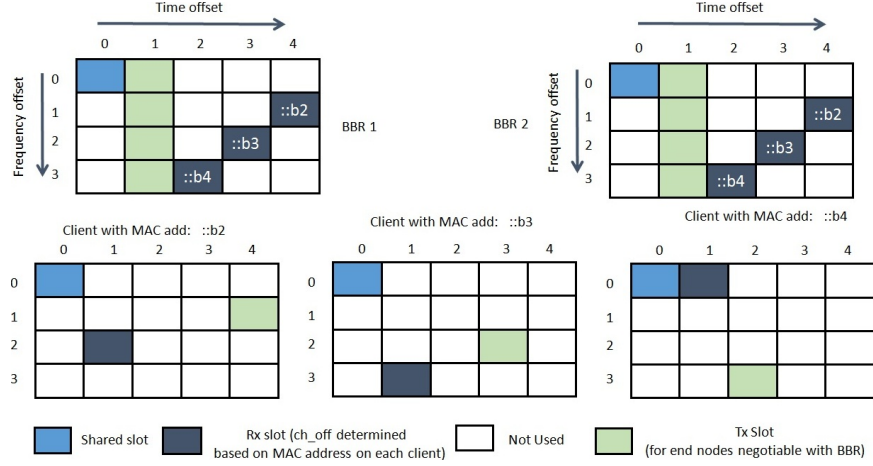
Figure 3: Example of a schedule using shared downlink slot and negotiable uplink slot.

increased power consumption. The schedule also affects scalability. With an increasing number of nodes in the network, the number of scheduled time slots must increase, expanding the slot frame length and increasing the overall latency in the network.

Apart from that, we also need to address following additional questions. How can the mobile node know when to transmit/receive once it has changed its BBR? If scheduling is decentralized and negotiated between the mobile node and the BBR, how can this schedule be communicated to other BBRs? If the schedule is centralized, how is it communicated to the mobile nodes? These questions need to be solved in order to achieve proper scheduling as well as schedule synchronization in order to enable seamless mobility.

Downstream scheme: For downstream traffic, we allocate for every mobile node a dedicated time slot in the slot frame. This time slot, used for downlink traffic to a mobile node, is determined by the Network Server and is communicated first to the BBRs and then to the mobile node itself by a BBR once it is synchronized with the network. The time slot for downlink traffic of different mobile nodes can only be shared in the time domain, as different channel offsets will be used for different mobile nodes. The actual channel offset used for down-

17

link needs to be specific for each mobile node and to be determined dynamically. If there are more mobile nodes in the network than available channel offsets, additional downlink time slots can be added. For downstream traffic, spatial reuse is possible, as different BBRs can transmit at the same time to different end nodes by using different channels.

For upstream traffic we cannot use the same logic, i.e. using the same time slot and different channel offsets, as the communication is one to many. This is different from the downstream case where one BBR communicates with one mobile node at a time. Hence, for the upstream traffic we propose three different schemes. Due to usage of unicast layer three packets encapsulated in broadcast layer two packets in the upstream direction, a mobile node is able to communicate with multiple BBRs at the same time. Packets in upstream will have as layer three destination address the IPv6 unicast address of the peer to which they are trying to communicate. Thus, it is necessary that all BBRs share the same subnet with NS and the mobile nodes, but it is not necessary to have the same IPv6 address. As multiple of them will receive layer two broadcast packets from mobile nodes there is no need to send IPv6 multicast packets from the mobile nodes. At this end, BBRs will forward to the NS all the traffic that comes from the mobile nodes.

Upstream scheme 1: The first scheme uses a shared transmit slot for upstream traffic for all mobile nodes. In this case the slot frame length will be short and repeats often, decreasing the latency. However, the probability of collisions will increase too. Regarding the power usage of mobile nodes, nodes can go to sleep during this shared transmit slot if they do not have anything to send.

Upstream scheme 2: The second scheme proposes a single transmit slot for each mobile node. This slot is negotiable with one of the BBRs, and once it is installed in the root BBR it will be communicated to all others via the Network Server. The negotiation of the schedule is done as in [22]. Hence, the mobile node can transmit its layer 2 broadcast upstream packets at a collision free time slot. The packet will be received by multiple BBRs as they are time

18

and schedule synchronized. The broadcast nature of upstream traffic will imply the absence of MAC layer ACKs. However, the collision free transmission and duplication of packets (reception of the same packet by multiple BBRs) will boost reliability. An example of this scheme is shown in Figure 3. For large scale deployments, this solution will increase the length of the slot frame, keeping the collision probability in upstream practically zero.

Upstream scheme 3: The third scheme is a trade off between the first two. We do not share the same slot between all nodes, rather we share them between a subset of nodes. If we share the same time slot between 3 mobile nodes then the latency is improved by a factor of 3 compared to the second scheme, however, the probability of collision is increased by a factor of 3 as well.

As the upstream traffic uses layer two broadcast packets there will be no layer two ACKs transmitted by BBRs. This employs that there will be no ACK-based synchronization between mobile nodes and the BBRs. Thus, the only synchronization possibility is via enhanced beacons (EBs) transmitted by the BBRs.

5.5. Implementation

Each BBR comprises an IEEE 802.15.4e mote that is connected via serial to an embedded PC. The IEEE 802.15.4e mote runs the OpenWSN [21] implementation of 802.15.4e TSCH, while on the embedded PC a Click [23] router chain is running to handle traffic from/to the Network Server.

The 6LowPAN layer is implemented in a Click router element. For the packets coming from the LLN network, Click creates the IPv6 header and forwards them towards the Network Server, which further routes them to their final destination. For the downstream packets entering the LLN, Click creates the 6LowPAN header and determines the layer 2 next hop address based on the IPv6 address of the mobile node. The resulting 6LowPAN packet is sent via the serial interface to the IEEE 802.15.4e mote together with the next hop layer 2 address. The IEEE 802.15.4e mote itself creates the 802.15.4 header, and sends the packet at the scheduled time to the mobile node.

19

The Network Server is a PC that also runs a Click router chain that manages the traffic. This chain creates two tables, a table for mapping mobile nodes to BBRs (based on the RSSI of the received packets) and a table for the delivered upstream packets towards the Internet (to remove duplicates). In our use case we set a timeout of 1s for the entries in both tables. Due to the relatively high speeds in our use case (3 m/s) this short timeout value will keep the tables up to date. This will prevent the selection of a wrong BBR by the Network Server due to obsolete information.

In addition to performing traffic management, the Network Server also runs a Precise Time Protocol (PTP) server for time synchronization of the backbone network. We used a PTP daemon [24] in a 1.2GHz clock PC. The second part of the synchronization consists of the synchronization of the time of the BBR's IEEE 802.15.4e mote with the PC time to which it is connected to. PTP offers sub-microseconds accuracy, which is enough considering the IEEE 802.15.4e time accuracy requirements.

The IEEE 802.15.4e mote of the BBR communicates with the PC via the serial port using the High-Level Data Link Control (HDLC) protocol. As the communication via serial port is triggered from the IEEE 802.15.4e mote, this case is similar with the ACK-based synchronization.

The reference time is communicated to all BBRs by the Click instance in the Network Server. We also implemented a Click element that translates the elapsed time since initialization to the ASN number and the number of $'clock$ $ticks'$ (as seen by the 802.15.4e node) from the start of the last time slot. Each time the IEEE 802.15.4e mote triggers the PC-to-node serial communication, the Click element calculates the current ASN and the number of $'clock\ ticks'$, and appends it to the data packet's metadata header. This information is then used by the IEEE 802.15.4e mote to align the time slot and ASN number with the actual values calculated by the PC. Serial communication delay in both ways is taken into account as part of these calculations.

Because the timings of the embedded PCs are synchronized via PTP in the network, thus using the same time reference, all BBRs in the network are time

and ASN synchronized. As such, we move the clock master of the mobile nodes from the separate BBRs towards the Network Sever that can be seen as the Virtual Grand Master (VGM).

The current Click element can communicate with these platforms: Open-Mote, Zolertia Remote (rev A and rev B1) and Zolertia. In order to support other platforms, it is necessary to change the mapping between $'clock\ ticks'$ and milliseconds, which is already parametrized in the code of Click element.

In our schedule solution, we propose the downlink slot to be shared in the time domain, while different channel offsets are being used for different mobile nodes. The channel offset to be used by BBRs for transmission is determined as $mod16(A)$, where A represents the last two bytes of the physical destination address and $modulo16()$ is used as there are 16 channel offsets. The mobile node will know which channel offset to use for listening by applying the modulo 16 operation on the last 2 bytes of its physical address. This will increase scalability as only a single time slot will be used for the downstream traffic. Considering the required scalability of the presented use case, namely 100 nodes, there will be at most 6 nodes scheduled at the same channel offset in case all 16 channels are used. In that case, additional downlink slots can be added, and nodes that produce the same channel offset can be scheduled in different time slots. This would lead to 5 additional downlink time slots. Another solution is to select different BBRs for downstream traffic in case different destinations map to the same modulo output. If this is not possible, transmission is done sequentially by the BBR.

One of the proposed schedules uses shared uplink slots for all mobile nodes. If the number of shared uplink slots is large enough (10-16) then we can serve a higher number of mobile nodes while keeping the slot frame length short. On the other hand, this will increase the collision probability. Considering the configuration of the environment in the use case shown in Section 2, the probability of more than 5 shuttles being in the same BBR coverage area is low as shown in [5]. This factor will help to decrease the impact of collisions.

The other proposed schedule scheme uses a single uplink time slot for each

21

mobile node. In such a case, 100 mobile nodes will result in a slot frame length
of 102 time slots. This slot length implies a latency of $\sim$1s for communication
from each mobile node. Such a latency is still acceptable for the use case we are
considering here.

6. Mathematical Evaluation

In order to show the impact of the number of gateways and the number of
mobile nodes on the latency, we provide the following mathematical model to
predict the upper bound RTT latency for upstream scheme 2. Let N_{MN} be
the number of mobile nodes in the network, N_G the number of gateways in the
network and N_{TS} the number of time slots in a slot frame. In case of upstream
scheme 2, the number of time slots in a slot frame will be:

$$N_{TS} = N_{MN} + ceil(N_{MN} \div 16) \tag{2}$$

where $ceil()$ rounds the number to the next integer and 16 is the number of
channel offsets used. The second term in equation 2 avoids the possibility of
scheduling two different mobile nodes on the same downlink time slot and chan-
nel offset. This ensures collision free transmission in downlink. Here we do not
consider the serial communication bottleneck between the BBRs and the back-
bone network, as we will show in Section 7.3. This bottleneck can be avoided
by a better hardware design with a faster interface between the 802.15.4 radio
and the Linux Click router. In this mathematical evaluation, we assume that
the communication between the BBRs and the network happens instantly and
does not interrupt the communication in the LLN part. As such, it does not
require the assignment of time slots for serial communication.

We assume that all downlink time slots are scheduled at the beginning of the
slot frame, before all other uplink time slots. For a node K that is scheduled in
downlink slot offset l, where $l < ceil(N_{MN} \div 16)$, and uplink slot offset k, where
$ceil(N_{MN} \div 16) < k < N_{TS}$, the upper bound RTT is expressed as:

$$RTT_K = (k - l) \times TS_{length} \tag{3}$$

where TS_{length} is time slot length in ms.

If there is more than one mobile node to be served in downlink by the same BBR and at the same downlink time slot, then equation 3 for node K will take the form:

$$RTT = (k - l) \times TS_{length} + (N - 1) \times N_{TS} \times TS_{length} \tag{4}$$

where N is the total number of mobile nodes that need to be served in the same downlink time slot by the same BBR. The second term in equation 4 describes the waiting time (upper bound) for the next slot frame until all other mobile nodes are served in downlink.

The condition that there is no downlink traffic for other mobile nodes scheduled at the same downlink time slot by the same BBR should hold for a packet to be transmitted instantly by a BBR in downlink. This condition does not hold only in case there is more than one mobile node using the same downlink slot offset, these nodes are served by the same BBR and need downlink communication at the same slot frame. The probability of this to occur is related with the number of nodes, the number of gateways installed and the downlink frequency of nodes.

From [5] it was shown that in a typical 2D shuttle warehouse system the coverage can go up to 80 m, if the maximal transmit power of 20 dBm for 2.4 GHz is used. The warehouse under consideration in [5] had dimensions of 13x14x100 m with 4 levels. For a density of one shuttle per 1000 m^3, 18 shuttles are needed. For this number of shuttles, at least two downlink time slots need to be used. On the other hand, for such a case you need at least two gateways per level to cover the whole zone. Moreover, for our use case, the downlink traffic has a lower frequency than the uplink traffic. As such, the probability that two different mobile nodes using the same downlink slot offset can only be served by the same BBR is low. For such number of end nodes in the network,
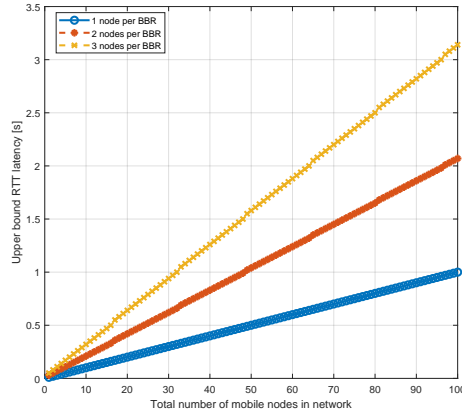
23

Figure 4: The upper bound packet RTT for the case of downstream scheme 2, considering different number of mobile nodes that can be served by single BBR at the same downlink time slot. $TS_{length} = 10ms$

the upper bound RTT latency will not exceed 1s, as it is shown in Figure 4.

In the extreme case where all nodes have the same traffic frequency in uplink and downlink, the RTT latency can be calculated by equation 4. The upper bound RTT latency from equation 4 is derived when k is the maximal index of time slot in slot frame and l is the minimal index of time slot in slot frame. In Figure 4 the upper bound RTT latency is shown for each case. Each curve shows the upper bound RTT latency as a function of the number of mobile nodes for a different number of mobile nodes served by the same BBR. The length of one time slot is 10 ms. In cases where the zone is fully covered by BBRs and mobile nodes have different downlink traffic frequencies the probability that two or more mobile nodes need to be served by single BBR at the same slot frame is low. Even in such a case, the upper bound RTT latency is not more than 2.1 sec for 100 mobile nodes in the network.

7. Results

In order to validate the proposed solution, we performed tests in the w-iLab.2 [25] testbed, which is a generic wireless testbed in a pseudo-shielded environment. The first set of tests validates the synchronization between the

24

Virtual Grand Master, the BBRs and the mobile nodes that are connected to these BBRs. The second set of tests validates the low latency handover of a mobile node from one BBR to another. The last set of tests assesses the impact of the different schedules we described in Section 5.4 on latency and packet losses. For all measurements in the next subsections, Zolertia Remotes rev B1 [26] have been used as IEEE 802.15.4e nodes. The OpenWSN [21] implementation of the IEEE 802.15.4e TSCH mode has been used as a starting point for the development of the system.

7.1. Synchronization Accuracy

As the IEEE 802.15.4e TSCH timing is based on ASN and the number of $'clock\ ticks'$ since the start of the last time slot, our Click router implementation needs to translate the timing from seconds to ASN and $'clock\ ticks'$. Due to fixed point precision, this translation is associated with a calculation error, but it is crucial to keep this error below a certain threshold, namely the $GuardTime$ Interval, which has been set to 1ms. If the time (in terms of ASN and $'clock\ ticks'$ from the last time slot) calculated by the Click element running on the PC differs less than 1 ms with the time measured at the 802.15.4e node of the BBR then the BBR is synchronized with the network.

We send time beacons every 10 seconds from the PC to the IEEE 802.15.4e mote via the serial link. The communication is triggered by a request packet from the IEEE 802.15.4e mote that is sent at the start of the time slot. The PC responds with its time beacon. The time beacon carries the current ASN and the number of $'clock\ ticks'$ since the start of the last time slot in the network. The $'clock\ ticks'$ number also takes into account the transmission time of the time beacon packet. The mote compares the time calculated in the network with the time measured by itself and adjust its time accordingly. The time calculated in the network refers to time translation from seconds to ASN and clock ticks plus the transmission time of the time beacon via serial. Once the adjustment has been done the node informs the PC about the adjustment. This step is only for measurement purposes and is optional during normal use.
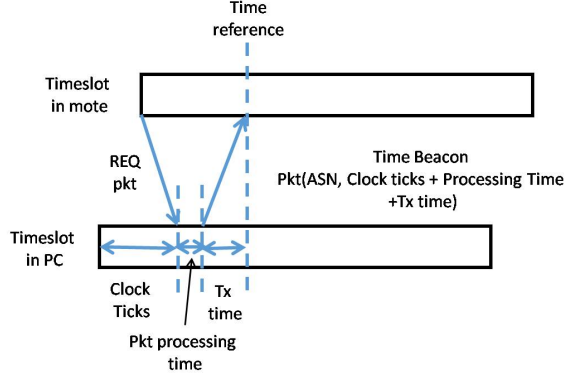
25

Figure 5: Exchanging packets during time synchronization process in BBRs.

The request packet coming from the mote is 6 bytes long (including HDLC headers), while the time beacon packet is 19 bytes long (containing ASN number, number of $'clock\ ticks'$ since the last time slot and HDLC headers). Using a serial connection with a baud rate of 115200 it takes 1.7 ms for both packets to get transmitted.

For evaluating the synchronization accuracy, the calculated time at which the time beacon arrives at the mote and the measured time at which the mote received the time beacon are taken as reference time values. The order of packet exchanges is shown in Figure 5 together with the time reference.

In Figure 6 the time reference values histogram distribution calculated at the network side and measured by the mote itself are shown. Time beacons are sent every 10 seconds and no data packets are used during the measuring time. The mean values are 2.4 ms and 2.5 ms, for the time calculated in the network and by the mote itself, respectively. Due to the packet processing in the mote and inside the Click chain at the network side, the mean value is slightly larger than the transmission time of the packets via serial (2.4 ms > 1.7 ms).

The source clock of the mote itself is a crystal clock of 32 KHz with a clock drift of 10 ppm. Due to the clock drift the mote can desynchronize $\pm$ 10 microseconds every 1 second with respect to the network. So the mote needs 100s to pass the *GuardTime* interval of 1 ms. Considering the delays in serial
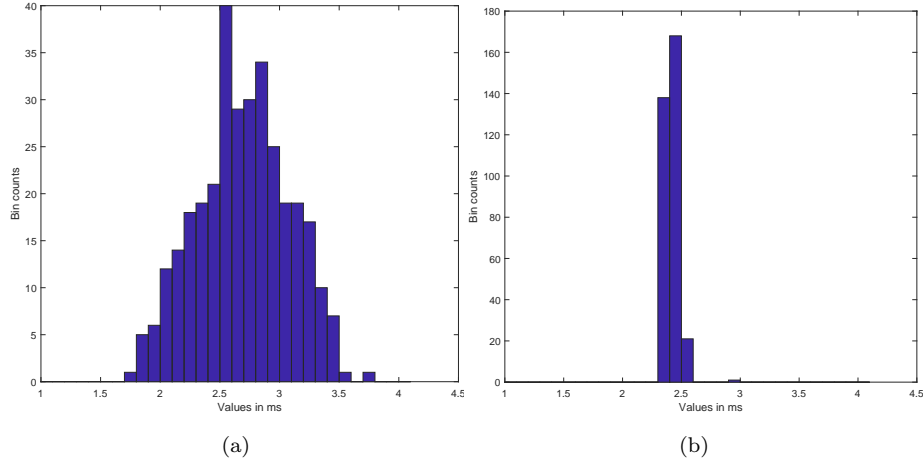
26

Figure 6: Histogram of reference time values a) measured in mote and b) calculated in network. The mean values are 2.4 and 2.5 ms, respectively. The bin intervals are 0.1 ms.
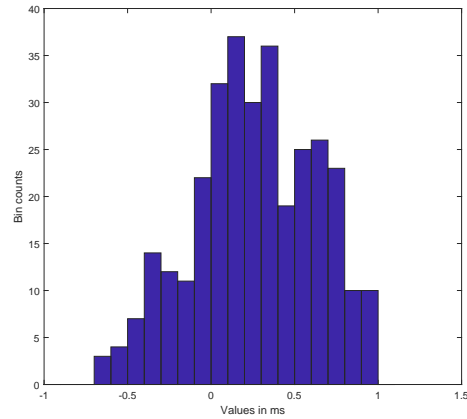

Figure 7: Histogram of time difference between mote and network with mean of 0.28ms. Synchronization beacons are sent every 10 sec. The bin intervals are 0.1 ms.

communication and the packet processing time in Click, we take a security factor of 10 and send time beacons every 10 seconds. As such, every 10 seconds the node will resynchronize with the network.

In Figure 7 the time difference between the reference time values measured by the mote itself and the values calculated in the network are shown. The mean value is 0.28ms which is lower than the 1ms *GuardTime* interval. We also
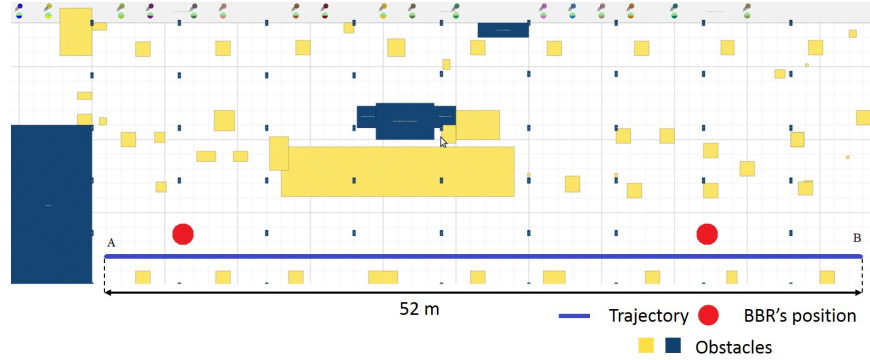
27

Figure 8: Map of test bed with indication of mobile node trajectory and the position of BBRs.

performed measurements for the cases where the time beacons were sent every 6 second and 3 seconds, respectively. In these cases the mean time difference was even lower, being 0.17 and 0.067 ms, respectively.

The considered industrial environments (racked warehouse systems) have constant temperatures in the range of 16^0-20^0 C [27]. This is to save products' quality and there is not any case when temperature will be higher. Contrary, in refrigerated warehouses (mainly for food products or temperature sensitive products) temperature might be even lower than 0. Both mobile nodes and BBRs are attached to devices that do not produce or release any heat. The first ones are attached to shuttles while the later ones are attached to racking system structure. Thus, there will be no temperature impact on clock drifts of the mobile nodes. In cases when the temperature is variable and can be high, the only impact that can be expect is on the clock accuracy of mobile nodes. They have lower accuracy clocks [26] and need to be synchronized wireless. The BBRs employ an embedded PC that has much better clock accuracy [28] (1 GHz clock), and moreover, it is synchronized via back-end using PTP synchronization. In cases when mobile nodes operate in higher temperature than normal one, enhanced beacons need to be send more often to account for higher clock drifts at certain temperatures.
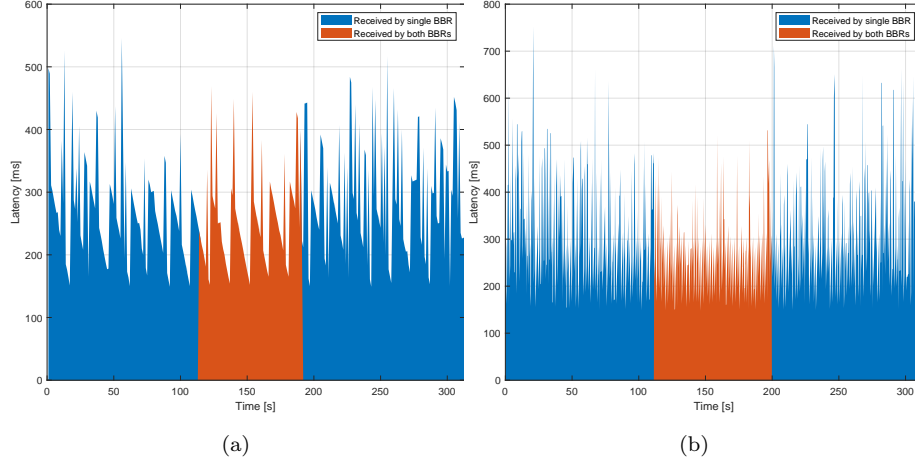
Figure 9: The packet latency during handover for traffic frequency of a) 3.33 Hz and b) 1 Hz. The red part shows packets that are received by both BBRs.

7.2. Handover Latency

In a network where all BBRs are fully time synchronized between each other the handover of mobile nodes should exhibit a low latency. To show that this is indeed the case with our architecture, we conducted a number of experiments with different traffic frequencies. For this we used the w-iLab.2 [25] testbed, a wireless test bed equipped with 58 static nodes and 18 mobile nodes.

We used two static nodes with Zolertia Remotes rev B1 as BBRs and a mobile node which was roaming around. In Figure 8, the position of the BBRs is indicated with red circles while the trajectory of the mobile node has as starting point A and ending point B. The distance between the BBRs was 35m while the length of the trajectory was 52m. Both BBRs are connected to the backbone network where the Network Server runs the PTP server for network time synchronization. The transmit powers of the BBRs and the mobile node were 3dBm. In order to decrease the intermediate coverage zones we added a fixed attenuation of 20 dB to antenna ports of the BBRs. This resulted in a coverage zone of each of the BBRs of ∼25m.

We generated an 80 byte packet every 300ms and 1sec, respectively, from

29

the Network Server towards the mobile node. The RTT latency was measured as well as packet losses. At the same time the BBR table at the Network Server was monitored every second to check at what time the handover took place.

In Figure 9, the latency over time is shown. The packet frequency was 3.33Hz and 1Hz, respectively. As it can be seen, there is no packet loss when the handover happens. However, there were some losses (0.5% - 1%) during communication, but those were mostly due to serial communication between the BBR's mote and the PC. The red part of the latency graph shows the latency of packets that were received by both BBRs in the intermediate coverage zone. The lowest latency is shown as the second packet is discarded by the network server.

For a frequency of 1Hz we see that the latency graph behaves as a saw tooth. This is a direct result of the time slot organization in our slot frames. In this case we used a slot frame containing 11 time slots, 3 of them being time slots for serial communication between the PC and the BBR mote, 1 shared time slot and two others for downlink and uplink, respectively. The remainder of the time slots were not active. For 2-way communication, the maximum RTT latency will occur when one has to wait one slot frame to perform the first transmission. With our configuration, this results in 165ms (11 times 15ms) plus processing and network latency. This value is equal to the deepness of the saw tooth. The same thing happens for the 3.33Hz traffic frequency if we zoom in on the graph.

When the BBRs are not time and schedule synchronized then handover latencies can be as high as tens of seconds. Such a case is shown in Figure 10. When the mobile node moves out of the coverage zone of the first BBR, it needs time to get synchronized with the new BBR. During that time, all communication is lost. The red part of the graph shows the latency of packets received by the second BBR, while the part without any packet shows the time with no connection for the mobile node.
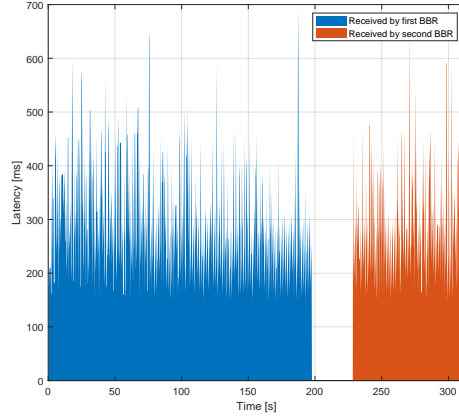
30

Figure 10: The packet latency during handover for traffic frequency of 1 Hz when BBRs are not time synchronized.

7.3. Scheduling Impact on Latency and Losses

As we discussed in Section 5.4, the way how the nodes are scheduled will impact network scalability as well as communication latency. In that section we qualitatively discussed the impact of each of the proposed schemes, while in section 6 we gave the mathematical model to determine the upper bound RTT latency for upstream scheme 2. In this section we will quantify the impact in terms of RTT latency for each node by performing measurements in the test-bed setup.

In this configuration we only used a single BBR, since we only consider the communication latency and packet losses of each node in case when multiple mobile nodes are served by the same BBR. One of the limitations in this scenario is the serial communication between the Zolertia Remote and the PC. In the BBR' schedule, apart from scheduled time slots for communication with mobile nodes, also time slots for serial communication with the PC need to be allocated.

Since we are using only one BBR and the BBR's mote does not have memory to save more than a single 128 byte packet, we can only send one packet at a time from the PC to the BBR mote. This means that we can only send a single packet every slot frame. Consequently, the number of mobile nodes that we can employ in this test depends on the generated traffic frequency and the slot
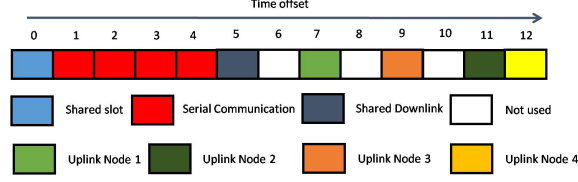
31

Figure 11: Slot frame organization at backbone border router.

frame length.

Considering the scheduling case with a single transmit slot for each mobile node and assuming traffic generation by each node every 1s, we cannot serve more than 4 mobile nodes (i.e. 4 packets per second). In this case, we will need at least 11 time slots in the slot frame (4 for serial communication (PC to mote), 1 shared, 1 for downlink communication, 4 for uplink communication and at least 1 time slot that is off that will be used for serial communication (mote to PC)). Every unused time slot will be used for communication from mote to PC. Such a slot frame is repeated ∼6 times per second. If the number of nodes is increased, slot frame must be increased too, resulting in less repetitions per second which impacts the number of packets that can be sent per second. Having this in mind we decided to stick to 4 mobile nodes and kept the traffic frequency at 1Hz for each of them. When all 4 nodes are served by the same BBR then we expect that the upper bound RTT latency should not exceed $4 \times N_{TS} \times TS_{Length}$ sec, as it was shown in equation 4.

We did tests for 300 seconds sending a packet of 80 bytes every second to each of the mobile nodes. The slot frame contained 13 time slots of 15 ms each. The slot frame organization of the BBR is shown in Figure 11. It should be mentioned that the uplink slots from mobile nodes to the BBR were negotiated between them.

In Table 1, statistics for the RTT latency and packet losses for each mobile node are given. It can be seen that the maximum latency does not exceed the threshold of $4 \times N_{TS} \times TS_{Length}$ time. In this case $N_{TS} = 13$ and $TS_{length} = 15ms$. The best case in terms of latency is when the 2-way communication

Table 1: Statistics for RTT latency for scheduling case with single uplink time slot per mobile node.

	Node 1	Node 2	Node 3	Node 4
Min (ms)	120.12	149.71	342.62	285.34
Max (ms)	780.61	780.51	781.05	782.01
Average (ms)	550.12	398.58	654.45	478.52
Standard deviation	142.71	159.15	122.51	186.67
Losses (%)	3	2	3	3

happens within the same slot frame. For instance, for node 1 the minimal latency is 120ms, which is equal to 8 slot lengths, which is in line with the slot frame organization in Figure 11. The packet is transmitted in time slot 1 from the PC to the BBR mote, after which it is transmitted from the BBR node to the mobile node in time slot 5. The response is transmitted during time slot 7 while at time slot 8 the response is actually received by the PC making thus the RTT latency ~8 time slots length.

Another proposed scheduling approach uses shared slots for the nodes in the uplink. The statistics for such a case, when 4 mobile nodes are used to communicate with the server via the same BBR, are shown in Table 2. The slot frame of the BBR had 9 time slots in total with 4 time slots for serial communication, 1 shared time slot, 1 time slot for downlink and 1 shared time slot for uplink. Two times slots were off. We can see that compared to the previous case the average latency is lower. However, there is no determinism with respect to the highest possible latency that is achieved. Retransmissions due to collisions make the maximum latency to increase, as can be seen from the values in Table 2.

From both Tables (1 and 2) it can be seen that the RTT latency is bounded. The upper bound depends on the schedule organization and the number of end

33

Table 2: Statistics for RTT latency for scheduling case with all mobile nodes sharing uplink time slot.

	Node 1	Node 2	Node 3	Node 4
Min (ms)	222.77	222.7	235.74	223.15
Max (ms)	981.58	1005.9	925.33	810.03
Average (ms)	423	469.87	511.95	361.13
Standard deviation	144.47	134.92	122.43	105.65
Losses (%)	26	35	34	15

nodes that need to be served in downlink by the same BBR at the same time. This brings determinism to the communication latency.

Regarding packet losses, in case of upstream scheme 2 (Table 1) they are mainly due to interference from other networks during measurements and not due to self interference. As our test-bed is not an RF shielded environment this is expected when the same frequency band (2.4 GHz) is also used by other networks in proximity (e.g IEEE 802.11 networks). In such a case all packets are transmitted in self-interference collision free time slot, but the impact of other networks still can not be alleviated. In case when all nodes will share the same uplink time slot then the main source of losses comes from self interference collision. This is shown by higher packet losses in Table 2.

7.4. Performance in Real Industrial Environment

In order to validate our solution in real life, we also performed a performance test in an industrial environment with a dense racking structure. The racking structure was three stories high, with each story having a height of 3 meters. Every story contained a shuttle track, with 4 shuttles per track in total. The length of the track along which the shuttles moved, was 32.5 meters. The measurement track was chosen to be on the second story of the racking system. Two BBRs were used, the first one placed one story above the beginning of

the selected track and the second one another 10 meters away from the end of the track under the selected track. There was no line of sight between BBRs as well as between BBRs and the end node for any of the positions during the measurements. The shuttle track is shown in Figure 12 and apart from its metallic structure it was also partly filled with different food products.

We used a QR code scanner that was connected to the Zolertia end node in order to generate data by scanning QR code tags that were attached to the racking structure. The QR code scanner was connected via serial to an embedded PC that runs a bridge towards the Zolertia remote. The QR codes were attached to the racking structure with a spacing of 2.6 meters between codes and the first code was placed at 3.3 meters from the start of the track as it is shown in Figure 12. The end node was placed on the shuttle and was transmitting information while the shuttle was moving along the track.

The maximal shuttle speed was 4 m/s with an acceleration of 2 m/s^2 [29]. Starting from standstill at the beginning of the track, this maximal speed is achieved after 2 sec and at a distance of 4 m. Running at maximal speed, the scanner will scan a QR code every 0.65 s. In our case, only the first QR code will be read at a slower shuttle speed than the maximal one, more specifically at 3.63 m/s.

In Figure 13 the QR code IDs scanned over time are shown. The red dots show the QR tag IDs that were received by both BBRs in the intermediate zone. It is seen that the handover happens smoothly without missing any QR code tag. While being in this intermediate zone where the handover takes place, packets are temporarily received by both BBRs. During the experiment, only the QR code tag with ID 7 was missed due to a problem with the scanner. The actual time difference between two consecutive transmissions is $\sim 0.67 sec$, being ~ 40 ms larger than the time difference between two consecutive scans. This comes as a result of the time slots used for serial communication. As a conclusion, this measurement campaign reveals that the setup is robust against different speeds of shuttle. In this case the speed was changed between 0-4 m/s and no negative impact on packet losses or latency was observed.

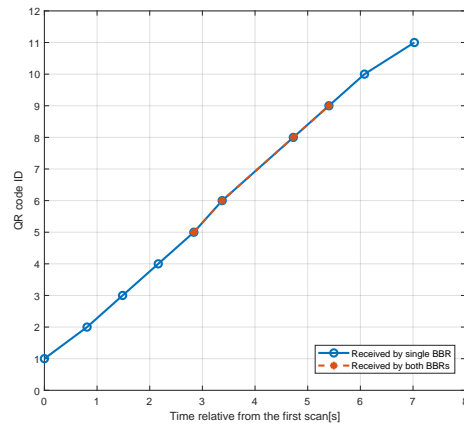Figure 12: Shuttle track inside the racking structure where the measurements were done.

Figure 13: Performance under fast movement of 4 m/s speed. The red QR code points are received by both BBRs.

8. Conclusions

By adopting time synchronized based MAC schemes it has become possible to deploy WSNs that meet strict latency requirements and that can be used in industrial settings. However, their usage has been limited to statically deployed nodes.

This paper has introduced a real-life industrial use case from the logistics domain that requires both the determinism that can be achieved with IEEE

802.15.4e TSCH as well as support for mobility. As a solution, we propose a novel single-hop multi-gateway network architecture where all the gateways, called BBRs, are both time and schedule synchronized by means of a Virtual Grand Master residing at the Network Server. We have addressed the synchronization and scheduling challenges that come with this architecture and have shown that this architecture is indeed able to improve the handover latency and decrease the communication outage due to desynchronization.

In terms of synchronization, we have shown how the synchronization between the IEEE 802.15.4e BBR motes and the network is done via the serial link. We experimentally validated that our approach results in smooth handovers without any communication outage time, compared to unsynchronized solutions, where handovers take up to several seconds.

The Network Server also performs traffic management functions. It performs deduplication of traffic in the uplink direction, while selecting the best BBR for downlink communication. In addition to this, the use of TSCH provides flexibility in scheduling. We have presented different scheduling schemes, one of them using shared downlink time slots while negotiating the uplink time slots between mobile nodes and BBRs. With this scheduling scheme we keep the upper bound latency fixed, while having collision free time slots for uplink and downlink communication. In addition to this we give a mathematical evaluation for the worst case RTT latency for such a scheme.

As such, we may conclude that the proposed architecture is viable and able to deal with mobility while offering communication guarantees by means of the installed schedules. Of course, some capacity must be sacrificed in order to support seamless handovers. However, by further advancement of electronics and possibility of BBRs to listen simultaneously to multiple channels this capacity loss will be avoided. In addition, the proposed architecture deserves further research to assess scalability and to mitigate some of the limitations that come from the serial communication. The proposed concept can be applied in a broader context, as 802.15.4 has some inherent limitations. For instance, it would be worth exploring how the same principles can be extended to higher

bit rate radios such as Wi-Fi or to e.g. large-scale Ultra wide-band (UWB) localization networks.

Acknowledgment

References

[1] Iec 62591, ed. 2.0: Industrial communication networks-wireless communication network and communication profiles-wirelesshart (draft), International Electrotechnical Commission and others.

[2] Ieee standard for local and metropolitan area networkspart 15.4: Low-rate wireless personal area networks (lr-wpans), IEEE Std, IEEE 802 Working Group and others 802 (2011) 4–2011.

[3] Ieee standard 802.15.4e, part 15.4: Low-rate wireless personal area networks (lr-wpans) amendament 1: Mac sublayer, IEEE Std, IEEE 802 Working Group and others.

[4] X. Vilajosana, K. Pister, T. Watteyne, Minimal ipv6 over the tsch mode of ieee 802.15. 4e (6tisch) configuration, Tech. rep. (2017).

[5] A. Karaağaç, J. Haxhibeqiri, W. Joseph, I. Moerman, J. Hoebeke, Wireless industrial communication for connected shuttle systems in warehouses, in: WFCS2017, the 13th IEEE International Workshop on Factory Communication Systems, 2017, pp. 1–4.

[6] T. Watteyne, J. Weiss, L. Doherty, J. Simon, Industrial ieee802.15.4e networks: Performance and trade-offs, in: 2015 IEEE International Conference on Communications (ICC), 2015, pp. 604–609. `doi:10.1109/ICC.2015. 7248388`.

[7] D. D. Guglielmo, A. Seghetti, G. Anastasi, M. Conti, A performance analysis of the network formation process in ieee 802.15.4e tsch wireless sensor/actuator networks, in: 2014 IEEE Symposium on Computers and Communications (ISCC), 2014, pp. 1–6. `doi:10.1109/ISCC.2014.6912607`.

[8] E. Vogli, G. Ribezzo, L. A. Grieco, G. Boggia, Fast join and synchronization schema in the ieee 802.15.4e mac, in: 2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), 2015, pp. 85–90. `doi:10.1109/WCNCW.2015.7122534`.

[9] D. De Guglielmo, S. Brienza, G. Anastasi, Ieee 802.15. 4e: A survey, Computer Communications 88 (2016) 1–24.

[10] K. Ovsthus, L. M. Kristensen, et al., An industrial perspective on wireless sensor networksa survey of requirements, protocols, and challenges, IEEE Communications Surveys & Tutorials 16 (3) (2014) 1391–1412.

[11] M. Barcelo, A. Correa, X. Vilajosana, J. L. Vicario, A. Morell, Novel routing approach for the tsch mode of ieee 802.15.14e in wireless sensor networks with mobile nodes, in: 2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall), 2014, pp. 1–5. `doi:10.1109/VTCFall.2014.6966074`.

[12] P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, R. Alexander, Rpl: Ipv6 routing protocol for low power and lossy networks, RFC 6550.

[13] M. Barcelo, A. Correa, J. L. Vicario, A. Morell, X. Vilajosana, Addressing mobility in rpl with position assisted metrics, IEEE Sensors Journal 16 (7) (2016) 2151–2161. `doi:10.1109/JSEN.2015.2500916`.

[14] Y. Al-Nidawi, A. H. Kemp, Mobility aware framework for timeslotted channel hopping ieee 802.15.4e sensor networks, IEEE Sensors Journal 15 (12) (2015) 7112–7125. `doi:10.1109/JSEN.2015.2472276`.

[15] T. Dang, P. Minet, P. Bellot, C. Mozzati, E. Livolant, Mobility support and service discovery for industrial process monitoring, in: Internet of Things: Novel Advances and Envisioned Applications, Springer, 2017, pp. 57–85.

[16] A. Gonga, O. Landsiedel, M. Johansson, Mobisense: Power-efficient micro-mobility in wireless sensor networks, in: Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on, IEEE, 2011, pp. 1–8.

[17] P. Tuset-Peiro, F. Vazquez-Gallego, J. Alonso-Zarate, L. Alonso, X. Vilajosana, Lpdq: A self-scheduled tdma mac protocol for one-hop dynamic low-power wireless networks, Pervasive and Mobile Computing 20 (2015) 84–99.

[18] J. Ma, D. Yang, H. Zhang, M. Gidlund, A reliable handoff mechanism for mobile industrial wireless sensor networks, Sensors 17 (8) (2017) 1797.

[19] S. Bluetooth, Specification of the bluetooth system-covered core package version: 4.0 (2010).

[20] X. Fafoutis, E. Tsimbalo, E. Mellios, G. Hilton, R. Piechocki, I. Craddock, A residential maintenance-free long-term activity monitoring system for healthcare applications, EURASIP Journal on Wireless Communications and Networking 2016 (1) (2016) 31.

[21] Openwsn implementation (Last visited September 2017).
URL https://openwsn.atlassian.net/wiki/spaces/OW/overview

[22] Q. Wang, X. Vilajosana, T. Watteyne, 6tisch operation sublayer (6top) draft-wang-6tisch-6top-sublayer-01, Tech. rep., Internet-Draft. IETF (2014).

[23] Click router (Last visited September 2017).
URL http://read.cs.ucla.edu/click/click

[24] Ptpd implementation (Last visited January 2018).
URL https://github.com/ptpd/ptpd

[25] Wilab test-bed (Last visited September 2018).
URL http://wilab2.ilabt.iminds.be

[26] Zolertia remote (Last visited September 2017).
URL https://github.com/Zolertia/Resources/wiki/RE-Mote

[27] C. Porras-Amores, F. R. Mazarrón, I. Cañas, Study of the vertical distribution of air temperature in warehouses, Energies 7 (3) (2014) 1193–1206.

[28] Apu embedded pc (Last visited September 2018).
URL https://www.pcengines.ch/apu2.htm

[29] Dematic multishuttle 2 (Last visited May 2018).
URL   http://www.warehouse-logistics.com/Download/Flyer/EN_Flyer_Produkt_Dematic_WMS_004.pdf

41