Extension of the IEEE 802.11ah ns-3 Simulation Module

Le Tian
University of Antwerp – imec
Belgium
le.tian@uantwerpen.be

Amina Šljivo
Ghent University – imec
Belgium
amina.sljivo@ugent.be

Serena Santi
University of Antwerp – imec
Belgium
serena.santi@uantwerpen.be

Eli De Poorter
Ghent University – imec
Belgium
eli.depoorter@ugent.be

Jeroen Hoebeke
Ghent University – imec
Belgium
jeroen.hoebeke@ugent.be

Jeroen Famaey
University of Antwerp – imec
Belgium
jeroen.famaey@uantwerpen.be

ABSTRACT

IEEE 802.11ah, marketed as Wi-Fi HaLow, is a new Wi-Fi standard for sub-1Ghz communications, aiming to address the major challenges of the Internet of Things (IoT), namely connectivity among a large number of densely deployed power-constrained stations. The standard was only published in May 2017 and hardware supporting Wi-Fi HaLow is not available on the market yet. As such, research on 802.11ah has been mostly based on mathematical and simulation models. Mathematical models generally introduce several simplifications and assumptions, which do not faithfully reflect real network conditions. As a solution, we previously developed an IEEE 802.11ah module for ns-3, publicly released in 2016. This initial release consisted of physical layer models for sub-1GHz communications and an implementation of the fast association and Restricted Access Window (RAW) channel access method. In this paper, we present an extension to our IEEE 802.11ah simulator. It contains several new features: an online RAW configuration interface, an energy state model, adaptive Modulation and Coding Scheme (MCS), and Traffic Indication Map (TIM) segmentation. This paper presents the details of our implementation, along with experimental results to validate each new feature. The extended Wi-Fi HaLow module can now support different scenarios with both uplink and downlink heterogeneous traffic, together with real-time RAW optimization, sleep management for energy conservation and adaptive MCS.

CCS CONCEPTS

• **Networks** → Network performance evaluation; Network algorithms; Network management; • **Sensor networks**;

KEYWORDS

IEEE 802.11ah, Restricted Access Window (RAW), Traffic Indication Map (TIM) segmentation, ns-3, Internet of Things (IoT)

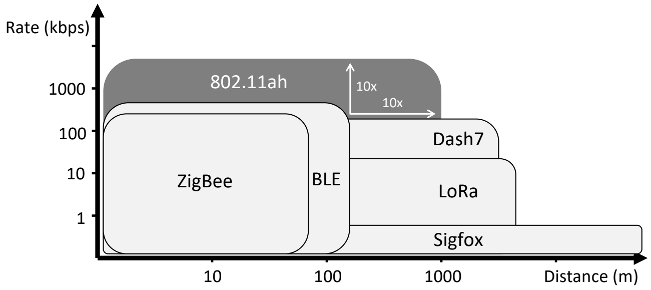


Figure 1: Position of IEEE 802.11ah compared to existing WPAN and LPWAN technologies

ACM Reference Format:
Le Tian, Amina Šljivo, Serena Santi, Eli De Poorter, Jeroen Hoebeke, and Jeroen Famaey. 2018. Extension of the IEEE 802.11ah ns-3 Simulation Module. In *Proceedings of the 2018 Workshop on ns-3 (WNS3 2018), June 2018, Surathkal, India.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3199902.3199906

1 INTRODUCTION

Many companies predict that the Internet of Things (IoT) will consists of tens of billions of devices connected together over the Internet. Currently, IoT offers plenty of solutions to connect devices with each other. There are two categories of low-power IoT communication technologies. First, wireless personal-area network (WPAN) provide connectivity up to tens of meters at data rates of kilobits per second. Second, low-power wide-area network (LPWAN) connect devices over an extended range up to several kilometers, but at a much lower throughput of bits per second.

Recently, the IEEE 802.11ah standard, marketed as Wi-Fi HaLow, was released as an alternative to existing WPAN and LPWAN technologies. It combines the advantages of Wi-Fi and low-power communication technologies, operating in the unlicensed sub-1GHz frequency bands (e.g., 863–868 MHz in Europe and 902–928 MHz in North America). Its main characteristic is to be a technology that provides a configurable trade-off between range, throughput and energy efficiency (cf. Figure 1). On the MAC layer it offers new mechanisms in order to support power limited stations in dense networks, such as hierarchical organization, short MAC header, fast association, Restricted Access Window (RAW), Traffic Indication Map (TIM) segmentation, and Target Wake Time (TWT).

WNS3 2018, June 2018, Surathkal, India

ACM ISBN 978-1-4503-6413-3/18/06…$15.00
https://doi.org/10.1145/3199902.3199906

Although the IEEE 802.11ah standard has only been officially released in May 2017, researchers have been investigating it already for several years, evaluating the new key features. Current research on the MAC layer of 802.11ah mainly focuses on performance of the RAW [5, 10–12] and TIM mechanisms [3, 7, 8]. Originally, much of this research was based on mathematical modeling of the saturated network state, which does not accurately capture real network behavior and is hard to adapt to non-saturated network conditions. Additionally, hardware for this standard is not on the market yet, so that the many features of Wi-Fi HaLow, such as fast association, RAW and TIM cannot be evaluated in a realistic manner. In order to fill this gap, we have previously implemented part of the IEEE 802.11ah PHY and MAC standard in the ns-3 network simulator [9]. This original version of the simulator consisted of sub-1GHz channel models, the different Modulation and Coding Scheme (MCS) supported by 802.11ah, the fast association, and RAW.

In this paper, we present several extensions and improvements to the original simulator. The new features include proper radio state transitions and energy consumption modeling, adaptive MCS, TIM segmentation, and new RAW configuration interface that allows real-time optimization of RAW parameters. A new version of the IEEE 802.11ah ns-3 module with these extended features is publicly available as open source[1].

The contributions of this paper are twofold. First, we outline the implementation of the extensions to the simulator. Second, we have conducted an in-depth simulation study to validate our implementation. The remainder of this paper is structured as follows. Section 2 provides a brief overview of the original implementation of the 802.11ah module. Subsequently, Section 3 describes the implementation of the newly added features. In Section 4, we present the validation results. Finally, conclusions are discussed in Section 5.

2 ORIGINAL 802.11AH MODULE

The original IEEE 802.11ah ns-3 simulator was developed and published in 2016, and based on the existing IEEE 802.11n model in ns-3, version 3.23. As shown in Figure 2, the 802.11ah ns-3 module consists of 4 main components:

- Wi-Fi Channel: An approximation of the physical medium over which data is transmitted, including propagation loss model and delay model. The propagation loss model characterizes the signal strength loss during transmission through the air, the propagation delay model describes the transmission delay between two nodes.
- Wi-Fi Phy layer: The PHY part of the IEEE 802.11ah module, where the format of PLCP Protocol Data Unit (PPDU) frame is defined and frames are sent and received through the Wi-Fi Channel. It consist of the *WifiPhy/YansWifiPhy, InterferenceHelper* and *ErrorRateModel* classes.
- Mac Low layer: It includes the *MacLow, DcaTxop/EdcaTxopN, MacRxMiddel* and *WifiStationManager* classes, implements RTS/CTS/DATA/ACK transactions, Distributed Coordination Function (DCF), Enhanced Distributed Channel Access (EDCA), part of RAW, packet queues, fragmentation, retransmission and rate control.

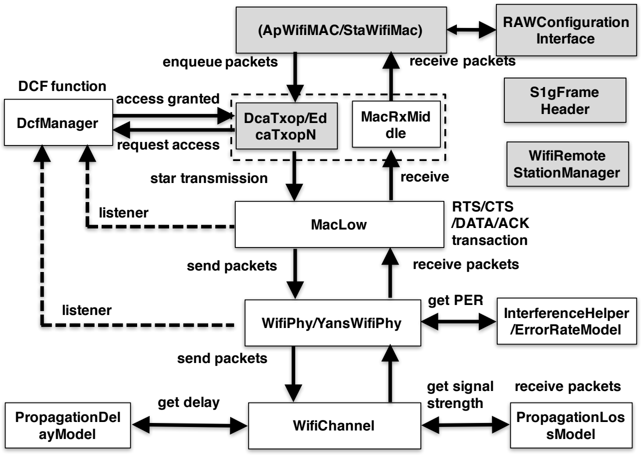


Figure 2: ns-3 802.11ah Wi-Fi models; grey background denotes changed/new components in our implementation

- Mac High layer: It implements management functions such as beacon generation, probing, association, fast association and part of RAW. It consists of the *ApWifiMac* (Access Point (AP)) and *StaWifiMac* (non-AP station) classes, which share a common parent class *RegularWifiMac*.

2.1 Physical Layer Features

For the physical layer, the simulator supports all the defined modulation and coding schemes for channel bandwidths 1, 2, 4, 8 and 16 MHz, except for MCS10 of channel width 1 MHz which requires coding rate of 1/2 with 2x repetition. The three different PPDU formats of IEEE 802.11ah are all supported. S1G_1M is for channel width 1 MHz, S1G_SHORT is for SU transmissions using channel width 2 MHz, 4 MHz, 8 MHz and 16 MHz, and S1G_LONG is used for MU and SU beamformed transmissions using 2 MHz, 4 MHz, 8 MHz and 16 MHz. The *ErrorRateModel* including the *NistErrorRate* and *YansErrorRate* models, is modified to be capable of calculating the packet error rate of 802.11ah with support of the QAM256 modulation scheme. As IEEE 802.11ah targets scenarios different from legacy 802.11 technologies, the existing propagation loss model cannot be applied. Therefore, we implemented the indoor and outdoor propagation loss models for 802.11ah developed by Hazmi et al. [6]. They proposed two different outdoor models, one for macro deployments and one for pico or hotzone deployments. Moreover, we evaluated several path loss models for IEEE 802.11ah based on a large scale sub-urban measurement campaign, including macro line-of-sight (LoS), pico LoS, and pico non-LoS with different as well as equal antenna height deployments. Based on the results of measurement, we proposed more realistic propagation loss models and updated the propagation loss model in the IEEE 802.11ah ns-3 simulator [4].

2.2 Medium Access Layer Features

IEEE 802.11ah introduces several novel features at the MAC layer, such as fast association, RAW, TIM segmentation and TWT, aiming to address the requirements of dense IoT networks. Among

[1] https://github.com/imec-idlab/IEEE-802.11ah-ns-3

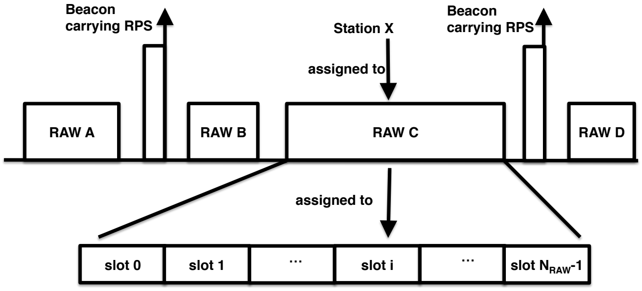


Figure 3: Schematic representation of the RAW mechanism

these features, the fast association and RAW mechanisms are implemented by modifying the Mac high (i.e., *StaWifiMac*, *ApWifiMac*) and Mac low layers (i.e., *DcaManager*, *DcaTxop*, *EdcaTxopN*).

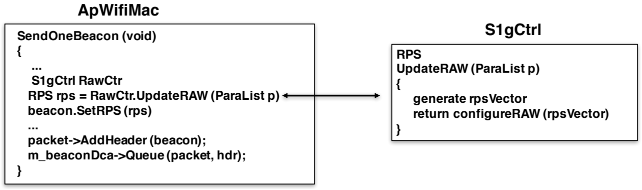
The fast association feature reduces the association time when a large number of stations are simultaneously trying to associate with the AP at the same time. An authentication control element contains a threshold that is broadcast to all stations in the beacon frame. When initialized, a station generates a random value from the interval [0, 1022], and sends an association request only if the random value is smaller than the threshold obtained from the received beacon. Otherwise it postpones association until the next beacon. The AP dynamically adjusts the threshold to allow all stations to associate in due time.

The goal of the RAW mechanism is to reduce collisions and improve throughput in dense IoT networks where hundreds or thousands of associated stations need channel access. As Figure 3 shows, RAW splits the stations into groups. Each group has one or more slots over which the stations belonging to that group are evenly split. During a RAW slot, only the stations that belong to that slot are allowed to access the channel. IEEE 802.11ah uses the RAW Parameter Set (RPS) information element to broadcast information about RAW groups at the start of each beacon interval. It specifies the stations that belong to each RAW group, the number of slots, slot format and slot duration count. The latter two jointly determine the RAW slot duration. The stations belonging to a RAW group have sequential Association ID (AID) and are assigned to RAW slots in a round-robin fashion.

RAW-related parameters can be pre-configured through attributes of the *ApWifiMac* class by experimenters. Before the simulation starts running, this user-defines RAW configuration is used to configure RAW-related attributes of *ApWifiMac*. During a simulation, the AP generates an RPS element with the pre-defined configuration and attaches it to each beacon frame. The RPS can only be statically configured, and cannot be changed once the simulation starts running, even if the network conditions change and the pre-configured RAW parameters become suboptimal.

3 OVERVIEW OF NEW FEATURES

We extended the original IEEE 802.11ah simulator with several new features, including the energy consumption model, RAW configuration interface, adaptive MCS and TIM segmentation.



```
SendOneBeacon (void)
{
    ...
    S1gCtrl RawCtr
   RPS rps = RawCtr.UpdateRAW (ParaList p)
   beacon.SetRPS (rps)
   ...
   packet->AddHeader (beacon);
   m_beaconDca->Queue (packet, hdr);
}
```

```
RPS
UpdateRAW (ParaList p)
{
    generate rpsVector
    return configureRAW (rpsVector)
}
```

Figure 4: RAW Configuration Interface

Figure 2 illustrates the framework of the IEEE 802.11ah module in ns-3. Grey components denote the extended 802.11ah module, e.g. *ApWifiMac/StaWifiMac, DcaTxop/EdcaTxopN, WifiRemoteStationManager, S1gFrameHeader*, and *RAWConfigurationInterface*. Among these components, only the *RAWConfigurationInterface* component is newly added, whereas the others already existed, but are modified to support new features.

The newly added *RAWConfigurationInterface* component interacts with *ApWifiMac* and any user-defined RAW optimization algorithms. It receives the information from the AP, delivers it to a RAW optimization algorithm and passes resulting RAW parameters obtained from the algorithm to the AP. As the energy consumption model works together with RAW and TIM, *StaWifiMac and DcaTxop/EdcaTxopN* are modified to allow stations to switch between sleep and awake modes. The adaptive MCS feature is implemented in *WifiStaWifiManager*, the parent class of all rate control algorithms. TIM segmentation indicates downlink traffic existence on several levels and its configuration is carried in two information elements, namely the Page Slice (PS) and the TIM element. TIM element is broadcasted with every beacon frame, whereas the PS element is broadcasted every n^{th} beacon frame, where n is defined in TIM element. Using TIM segmentation, stations can avoid receiving some of the beacon frames so as to sleep longer. This feature is implemented in the *ApWifiMac/StaWifiMac*. Additionally, page slice and TIM classes are created and added to the *S1gFrameHeader* component. The remainder of this section provides detailed description about the implementation of the above features.

3.1 RAW Configuration Interface

In the original simulator, the RAW parameters can only be configured through attributes of *ApWifiMac* before the start of the simulation. In this extended version, a dynamic RAW configuration interface (implemented in the *S1gCtrl* Class) is added to allow any user defined RAW optimization algorithm (e.g., Traffic-Aware RAW Optimization Algorithm (TAROA) [10], Enhanced Traffic-Aware RAW Optimization Algorithm (E-TAROA) [11]) to change the RAW parameters in real-time. The dynamic RAW configuration algorithm is executed when the AP generates the beacon frame. As Figure 4 shows, the AP calls the *S1gCtrl::UpdateRAW (ParaList p)* function, then the AP attaches the returned RPS element to the beacon frame and broadcasts it to all the stations. The list of parameters {*ParaList*} represents information about the network that is needed to determine the new RAW parameters. In the *S1gCtrl* class, the *S1gCtrl::UpdateRAW (ParaList p)* function determines the number of RAW groups and parameters of each group as represented by *rpsVector*. The parameters that can be configured are listed in Table 1.

With these parameters, the *S1gCtrl::configureRAW (rpsVector rpsV)* function creates the RPS elements. In order to use this interface, the user only needs to define their own RAW optimization algorithm and generate the RAW parameters in the format of *rpsVector*.

3.2 Energy Consumption Model

In the original IEEE 802.11ah implementation, radio power states were not implemented, and each station was assumed to be awake at all times. However, RAW and TIM allow stations to sleep outside of their assigned slots, thus saving power. In order to model this and allow realistic energy consumption measurements, some modifications were applied to allow devices to go into a sleep state outside their transmission or reception periods. Based on the state diagram in Figure 5, we implemented the radio state transitions of the device in the 802.11ah ns-3 module. Depending on the current state, the radio will go into the IDLE, RX, TX or SLEEP state. Sleep is initiated by calling *MacLow::GetPhy()->SetSleepMode()* in the *MacLow* class.

When the station receives the beacon, it calculates when its RAW slot will start and then it goes into sleep mode until its assigned slot or a shared slot (e.g., outside of the RAW period) starts. At this time, the station wakes up and checks if it has packets in its transmission queue. If so, it starts contending for the channel with the other devices active in the slot, otherwise it goes back to sleep mode, in order to save energy. The station will also wake up if a packet arrives in its transmission queue in the middle of a slot it can access. If the station is still awake the end of the assigned RAW or shared slot, it will also go back to sleep. Finally, the station will always wake up to listen to the beacon. Moreover, we calculate how much time each device spends in the TX, RX, IDLE and SLEEP states, in order to calculate the energy consumption. This is done by tracing the PHY state and storing the time intervals spent in each state.

3.3 Adaptive MCS

In order to achieve a high performance under varying conditions, IEEE 802.11 introduces rate control algorithms to adapt the MCS dynamically. In ns-3, multiple rate control algorithms are available, such as for example *ArfWifiManager*, *ConstantRateWifiManager* and *MinstrelWifiManager*. In the original simulator, the IEEE 802.11ah device could only use these algorithms to adapt the MCS within the same channel bandwidth. As IEEE 802.11ah supports multiple channel bandwidths (i.e., 1,2,4,8, and 16 MHz), we modified the *WifiRemoteStationManager* class (the parent class of all the rate control algorithms) to allow automatic switching among different channel bandwidths.

Concretely, a *channelwidth* attribute is added to the *StaWifiMac* and *ApWifiMac* classes, and the *S1gCapabilities* element is created to carry the channel width information. The user sets the channel width of the station or AP through the *channelwidth* attribute before the simulation starts. When the station is initialized, it generates the *S1gCapabilities* element with the configured channel width, attaches it to the association request packet and sends it to the AP. When the AP receives the association request, it extracts the *S1gCapabilities* element and calls the function *AddStationS1gCapabilities* of *WifiRemoteStationManager* to store the features of the station including the channel width. The stations obtain the capabilities of the AP with a similar procedure but through the association response packet from AP. With the obtained channel width, the above rate control algorithm is able to adapt the MCS across various channel widths.

Table 1: Configurable parameters of a RAW group

Parameter	Description
Aidstart	Lowest station AID in the RAW group
AidEnd	Highest station AID in the RAW group
SlotFormat	Slot format
SlotDurationCount	Slot duration count
NRawSlotNum	Number of slots in the RAW group
SlotCrossBoundary	Cross slot boundary (true or false)

According to the IEEE 802.11ah specification, the *channelWidth* subfield of the *S1gCapabilities* element is represented by 2 bits. Therefore, the *channelwidth* attribute is defined from 0 to 3, denoting 2, 4, 8 and 16 MHz respectively. The initial channel width configuration of a device determines its maximum channel width. As such, a device configured with higher initial channel width supports MCS with lower channel widths, while the inverse is not true.

3.4 TIM Segmentation

IEEE 802.11ah introduces TIM segmentation, which distributes the stations in a hierarchy, enabling effective management of a large number of stations as well as energy conservation and contention reduction in a structured manner. Each station is assigned a unique 13-bits Association ID (AID) in the range 1–8191 (AID 0 is reserved for group addressed traffic). The AID represents the station in a hierarchical structure, as Figure 6 illustrates.

The TIM element in a beacon frame indicates downlink buffered data for all stations in the hierarchy. In a network with a large number of associated stations corresponding to a single TIM and page, a single TIM element indicating the downlink buffered data for all stations in the page can be quite large [1]. To reduce the size of the element needed to communicate buffered data status to power saving stations, the AP may send one portion of a page called Page Slice (i.e. a subset of blocks from a single page) in a beacon frame. Two different types of TIM are distinguished: TIM and Delivery Traffic Indication Map (DTIM). DTIM is broadcast every $T_{DTIM} = n \cdot T_{TIM}$ where n is a positive integer and carries a Page Slice element. A station wakes up to receive the DTIM beacon that contains the Page Slice element for its associated page slice and checks if its Page Slice element indicates buffered data for its block in the Page period. If it indicates there is buffered data, the station wakes up to receive the TIM beacon carrying information on buffered data in its page slice in a TIM element. If not, the station can sleep until the next DTIM beacon. A station can schedule its wake-up time by computing it, using the information received in the Page Slice element and the T_{TIM} broadcast in the association phase. When a station receives its TIM beacon carrying its TIM element, it reads the block bitmap to check if there is traffic indicated for it. If there is data pending for it, it reads the RPS element from the beacon frame to schedule the time to wake up and receive its data. Otherwise, it sleeps until its next DTIM beacon, assuming it has no

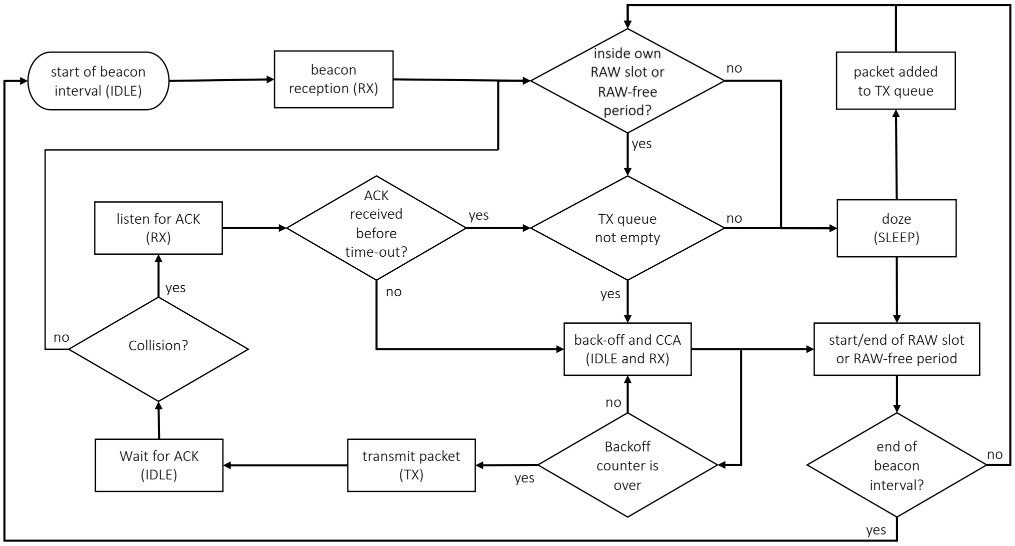


Figure 5: Radio state diagram for IEEE 802.11ah with RAW

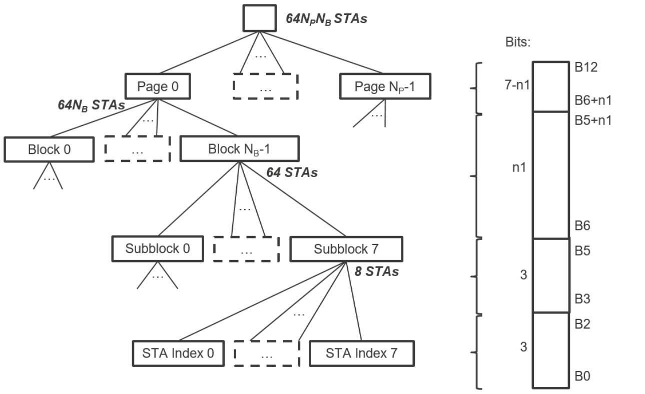


Figure 6: Each station is assigned a unique AID that represents the station in a hierarchical structure [8]

Table 2: Configurable parameters of a Page Slice

Parameter	Description
Page Period	Number of BIs between two successive DTIMs
Page Slice Length	Number of blocks in each TIM (except last)
Page Slice Count	Number of TIMs in one Page Period
Block Offset	Offset of block in the first PS from the first block in the page assigned within the Page Period
TIM Offset	Number of BIs from the DTIM to the beacon that carries the first PS of the page indicated in DTIM

BI - Beacon Interval; PS - Page Slice

data to transmit, as shown in Figure 7. Page slicing is defined by configuration parameters listed in Table 2.

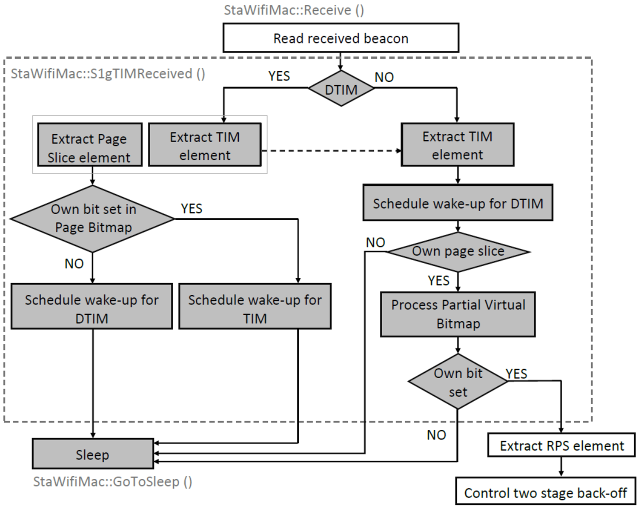


Figure 7: New functionality in the *StaWifiMac* class (dark background): handling TIM segmentation; grey background denotes changed/new components in our implementation

4 VALIDATION AND EVALUATION

This section presents results of the extended features of our simulation framework. This includes RAW optimization using the RAW Configuration Interface, the energy consumption model, adaptive MCS, and bi-directional traffic using TIM segmentation.

4.1 Simulation Setup

We evaluate IEEE 802.11ah in four scenarios: (a) an IoT monitoring scenario where a large number of battery-powered sensors with

Table 3: Default PHY and MAC layer parameters used in our experiments

Parameter	Value
Frequency	900 MHz
Transmission power	0 dBm
Transmission gain	0 dB
Reception gain	3 dB
Noise Figure	3 dB
Coding method	BCC
Propagation loss model	Outdoor, macro [4]
Error Rate Model	YansErrorRate
CWmin	15
CWmax	1023
Traffic access categories	AC_BE
Payload size	64 bytes
Cross Slot Boundary (CSB)	enabled
MAC header type	legacy header
Beacon Interval	0.1 s
Station distribution	random
Rate control algorithm	constant

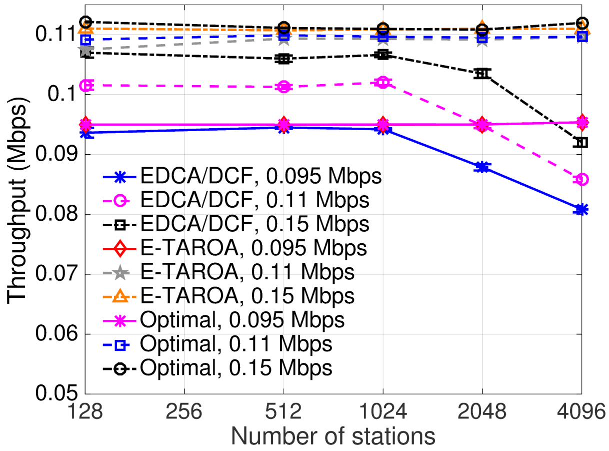


Figure 8: Performance comparison between E-TAROA, “optimal” and EDCA/DCF for different traffic loads and number of stations

heterogeneous traffic requirements send measurements to a backend server, (b) a single mobile sensor station continuously sending UDP packets to the AP, (c) a video streaming scenario that needs consistent high throughput and (d) an IoT closed-loop control scenario where a large number of battery-powered remotely-deployed sensors and actuators employ bi-directional communication with their corresponding controllers. All scenarios consider a single AP. The default PHY and MAC layer parameters are presented in Table 3. Note that some MAC parameters take different values in different scenarios, in which case they are additionally specified for each individual scenario.

We evaluate throughput of E-TAROA [11] in scenario (a). Three different total traffic loads $T = \{0.095, 0.11, 0.15\}$ Mbps, and 5 different number of stations $S = \{128, 512, 1024, 2048, 4096\}$ are simulated. All the stations use MCS1 with channel width 1 MHz and have different transmission intervals, following a uniform distribution and the ratio between any two sensors' transmission interval in any experiment is lower than 20. The stations are randomly placed around the AP in a circle of 450 m with the presence of hidden nodes. Throughput is calculated as the average number of successfully received payload bytes by the AP per second. To obtain the energy consumption, We measured the total transmit-time, receive-time and sleep-time for each simulated station. In the SLEEP state, the radio is turned off at the PHY layer and its consumed energy is negligible (i.e., assumed to be 0). The power consumption for the other states is based on measurements reported by Ba *et al.* [2] for their IEEE 802.11ah transceiver hardware. Specifically, we use a TX power of 7.2 mW and an RX/IDLE power of 4.4 mW.

Scenario (b) is used to evaluate the adaptive MCS where one mobile station continuously transmits 64-byte UDP packets to the AP, while moving from -500 m to 500 m relative to the centrally placed AP at the speed of 1 m/s by using the *ConstantVelocityMobilityModel*. Besides the mandatory 1 and 2 MHz channel widths, both station and AP also support the 4 MHz channel.

We evaluated reliability, jitter, round-trip time (RTT) and throughput with bi-directional traffic in scenarios (c) and (d). In the closed-loop control scenario, we consider randomly distributed pairs of sensor/actuator nodes and controller nodes modeled as Constrained Application Protocol (CoAP) clients and servers positioned up to 500 m away from the AP, where all control-loops have uniform dynamics. We disable cross-slot boundary (CSB) and evaluate sensitivity to TIM segmentation, not employing RAW. In the video-streaming scenario, stations represent IP motion cameras that stream on motion detection. To ensure streaming, we set the motion probability to 1. If motion is detected cameras stream for 10 s at a specified datarate. On the MAC layer, a single RAW group per beacon interval is configured, with the RAW duration maximized within a beacon interval. We use non-CSB, vary MCS, the number of slots, and the slot duration. We also vary the TIM segmentation configuration parameters, i.e. page slice length and the number of page slices that match the DTIM period. Block offset and TIM offset are set to zero and a DTIM interval corresponds to a single page.

4.2 RAW Optimization Algorithms

In realistic scenarios, the AP normally does not have prior knowledge about the traffic of each station. In order to group stations efficiently, the AP needs to estimate the traffic patterns based on the packet transmission information. Therefore, we proposed E-TAROA, which interacts with the AP through the RAW configuration interface. It estimates the network conditions every beacon interval based on network information obtained from the AP, and updates the RAW parameters (cf., Table 1). More details about E-TAROA can be found in our prior work [10, 11].

We compare the performance of E-TAROA, “optimal” grouping, and the legacy channel access method EDCA/DCF. The “optimal”

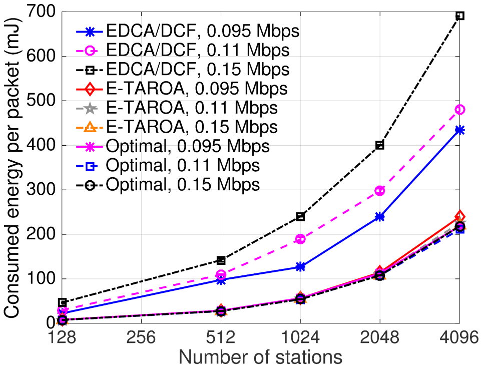


Figure 9: Energy consumption comparison between E-TAROA, "optimal" and EDCA/DCF for different traffic loads and number of stations

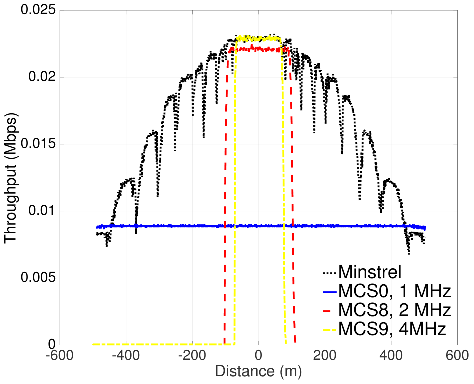


Figure 10: Throughput for *MinstrelWifiManager* and *ConstantRateWifiManager* using different MCS and channel widths, as a function of distance from the AP

grouping method assumes the AP knows the traffic of each station, allowing the AP to group the stations in an optimal way. The results are shown in Figure 8. As the "optimal" method assumes the AP knows the transmission interval of each node in advance, it achieves the highest performance. While E-TAROA is more realistic, the achieved throughput is quite close to the "optimal" and scales much better than EDCA/DCF in dense networks. For traffic load 0.095 Mbps, E-TAROA has similar throughput as EDCA/DCF for 128 up to 1024 stations, while throughput of EDCA/DCF drops to 0.08 Mbps with 4096 stations. For traffic load 0.11 Mbps and 0.015 Mbps, the throughput of E-TAROA remains almost the same for all different number of stations. In contrast, EDCA/DCF achieves around 6% and 4% less throughput for 128 stations, and 22% and 17% less for 4096 stations. These results show the advantages of real-time RAW optimization, as well as the ability of the simulator to support dynamic RAW configuration changes during simulation.

4.3 Energy Consumption

In this section, we evaluate the energy consumption of E-TAROA and EDCA/DCF using the implemented energy state transitions and consumption model. Specifically, four states are considered: TX, RX, IDLE, and SLEEP. The energy consumed by an IEEE 802.11ah station is obtained by multiplying the time the device spends in each certain state by the corresponding power consumption of that state. The X-axis of Figure 9 displays the number of stations, while the Y-axis depicts consumed energy (in mJ). The energy consumption of EDCA/DCF is clearly higher than that of E-TAROA and optimal RAW grouping. This is due to the grouping of stations, which allows channel access to be better managed and the station is able to enter sleep mode when it is not in its slot or has no packets to transmit. When using EDCA/DCF, stations spend a lot of time in the idle state during back-off, especially in denser and more traffic-heavy environments.

4.4 Adaptive MCS

In this section, we select the *MinstrelWifiManager* to evaluate the adaptive MCS capabilities. We compare it to constant MCS by using the *ConstantWifiManager*. As illustrated in Figure 10, the *MinstrelWifiManager* can adapt the MCS and channel width when the distance between the station and AP is changing, offering a dynamic trade-off between distance and throughput. In contrast, the *ConstantWifiManger* is only suitable for a certain distance. For MCS9, 4 MHz, it can achieve almost the same throughput (i.e., 0.22 Mbps) as *MinstrelWifiManager* for distances up to 70 m. For larger distances, the station can not even associate with the AP. For MCS8 with 2 MHz channel width, the station can communicate with AP up to 100 m away, at the cost of slightly lower throughput. For MCS0 with 1 MHz channel width, the station as the same coverage as when using the *MinstrelWifiManager*, but the achieved throughput is on average much lower. These results clearly show the ability of the station to dynamically adapt its MCS based on its current signal strength.

4.5 Bi-Directional Traffic

In the video-streaming scenario, we demonstrated feasibility of reliable and high-throughput applications such as IP camera streaming and over-the-air firmware updates using legacy Internet traffic (TCP/IP) for up to 20 stations over 200 m with 160 – 256 kbps of throughput on average per station. We evaluate the trade-off between the maximum attainable data rate per station and the total number of cameras in the network [8]. As Figure 11 illustrates, already for 10 stations adequate TIM and RAW configuration can achieve a 10 % higher data rate than EDCA/DCF. For 10–80 cameras, the improvement in maximum attainable data rate amounts to 10–30 % and further grows for denser networks.

In the closed-loop control scenario we focus on optimal RAW and TIM configurations that can guarantee $RTT < T_s$, where T_s represents the sampling period of a system response. Due to 802.11ah's energy saving management, RTT can be vastly increased given that a round trip in a closed-loop normally takes 4 hops for a single packet, where only one hop per DTIM period can be performed according to the standard. Moreover, increasing granularity in TIM segmentation increases the minimum attainable RTT and disables the use of control-loops with fast dynamics. In a control-loop, inter-packet delay at the sensor/actuator must be approximately constant and smaller than T_s in order to not destabilize the loop. Our simulations show how inter-packet delay remains approximately constant up to the saturation point that depends on the density and the loop's dynamics, but increasing granularity in TIM segmentation beyond that point results in inter-packet delay divergence as shown in Figure 12.

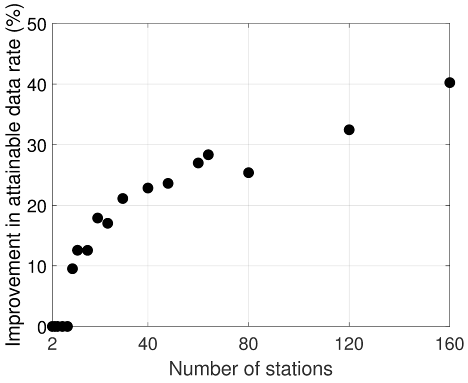


Figure 11: Proper TIM and RAW configuration improves IP cameras' data rate relative to EDCA/DCF, especially in dense networks

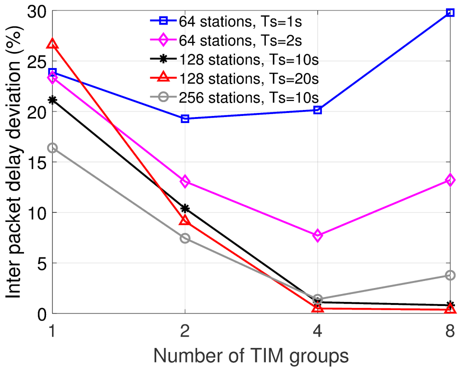


Figure 12: TIM segmentation improves the inter-packet delay deviation up to a point that depends on network density and the dynamics of the control loops

5 CONCLUSION

This article presents an extension to the IEEE 802.11ah implementation in ns-3, which was originally released in 2016 [9]. We extended the IEEE 802.11ah module in ns-3 with four additional features. First, we implemented an IEEE 802.11ah radio state and energy consumption model to precisely evaluate power consumption of each station. Second, we made changes to support adaptive MCS across channel bandwidths, i.e. the MCS automatically adapts to the distance and signal strength. Third, we implemented the TIM segmentation mechanism that enables not only simulation of upload-only sensor scenarios, but also scenarios with bi-directional traffic. The TIM segmentation mechanism primarily takes care of downlink data delivery, but also greatly contributes to energy conservation. Finally, the new RAW configuration interface enables real-time RAW reconfiguration as opposed to the original simulator, which only allowed static RAW configuration prior to simulation start.

The presented ns-3 simulation module enables realistic evaluation of the most promising IEEE 802.11ah MAC mechanisms. It is more realistic than mathematical models, which generally assume simplistic traffic models, and packet collision behavior.

ACKNOWLEDGMENTS

Part of this research was funded by the Flemish FWO SBO S004017N IDEAL-IoT (Intelligent DEnse And Long range IoT networks) project. Serena Santi is funded by the Fund For Scientific Research (FWO) Flanders under grant number 1S82118N.

REFERENCES

[1] IEEE Std 802.11ah 2016. 2017. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Sub 1 GHz License Exempt Operation.* Institute of Electrical and Electronics Engineers (IEEE).

[2] A. Ba, K. Salimi, P. Mateman, P. Boer, J. van den Heuvel, J. Gloudemans, J. Dijkhuis, M. Ding, Y. H. Liu, C. Bachmann, G. Dolmans, and K. Philips. 2017. A 4mW-RX 7mW-TX IEEE 802.11ah Fully-integrated RF Transceiver. In *2017 IEEE Radio Frequency Integrated Circuits Symposium (RFIC).* Honolulu, HI, USA, 232–235. https://doi.org/10.1109/RFIC.2017.7969060

[3] A. Bel, T. Adame, B. Bellalta, J. Barcelo, J. Gonzalez, and M. Oliver. 2014. CAS-based Channel Access Protocol for IEEE 802.11ah WLANs. In *European Wireless 2014; 20th European Wireless Conference.* Barcelona, Spain, 1–6.

[4] B. Bellekens, L. Tian, P. Boer, M. Weyn, and J. Famaey. 2017. Outdoor IEEE 802.11ah Range Characterization Using Validated Propagation Models. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference.* Singapore, Singapore, 1–6. https://doi.org/10.1109/GLOCOM.2017.8254515

[5] L. Beltramelli, P. Österberg, U. Jennehag, and M. Gidlund. 2017. Hybrid MAC Mechanism for Energy Efficient Communication in IEEE 802.11ah. In *2017 IEEE International Conference on Industrial Technology (ICIT).* Toronto, ON, Canada, 1295–1300. https://doi.org/10.1109/ICIT.2017.7915550

[6] A. Hazmi, J. Rinne, and M. Valkama. 2012. Feasibility Study of IEEE 802.11ah Radio Technology for IoT and M2M Use Cases. In *2012 IEEE Globecom Workshops.* Anaheim, CA, USA, 1687–1692. https://doi.org/10.1109/GLOCOMW.2012.6477839

[7] T. Kim and J. M. Chang. 2017. Enhanced Power Saving Mechanism for Large-Scale 802.11ah Wireless Sensor Networks. *IEEE Transactions on Green Communications and Networking* 1, 4 (Dec 2017), 516–527. https://doi.org/10.1109/TGCN.2017.2727056

[8] A. Sljivo, D. Kerkhove, L. Tian, J. Famaey, A. Munteanu, I. Moerman, J. Hoebeke, and E. De Poorter. 2018. Performance Evaluation of IEEE 802.11ah Networks With High-Throughput Bidirectional Traffic. *Sensors* 18, 2 (2018). https://doi.org/10.3390/s18020325

[9] L. Tian, S. Deronne, S. Latré, and J. Famaey. 2016. Implementation and Validation of an IEEE 802.11ah Module for ns-3. In *Proceedings of the Workshop on ns-3 (WNS3 '16).* ACM, Seattle, WA, USA, 49–56. https://doi.org/10.1145/2915371.2915372

[10] L. Tian, E. Khorov, S. Latré, and J. Famaey. 2017. Real-Time Station Grouping under Dynamic Traffic for IEEE 802.11ah. *Sensors* 17, 7 (2017). https://doi.org/10.3390/s17071559

[11] L. Tian, S. Santi, S. Latré, and J. Famaey. 2017. Accurate Sensor Traffic Estimation for Station Grouping in Highly Dense IEEE 802.11ah Networks. In *Proceedings of the First ACM International Workshop on the Engineering of Reliable, Robust, and Secure Embedded Wireless Sensing Systems (FAILSAFE'17).* ACM, Delft, Netherlands, 1–9. https://doi.org/10.1145/3143337.3149819

[12] Y. Wang, K. K. Chai, Y. Chen, J. Schormans, and J. Loo. 2017. Energy-aware Restricted Access Window Control with Retransmission Scheme for IEEE 802.11ah (Wi-Fi HaLow) Based Networks. In *2017 13th Annual Conference on Wireless On-demand Network Systems and Services (WONS).* Jackson, WY, USA, 69–76. https://doi.org/10.1109/WONS.2017.7888774