Design model of an IEEE 802.11ad infrastructure for TSN-based industrial applications

Rosario Giuseppe Garroppo[a,*], Maria Grazia Scutellà[b]

[a] *Dipartimento di Ingegneria dell'Informazione, University of Pisa, Pisa, Italy*
[b] *Dipartimento di Informatica, University of Pisa, Pisa, Italy*

Abstract

This paper addresses two main aspects related to the application of the IEEE 802.11ad technology in an industrial environment, i.e., to guarantee a low latency and reliable data delivery while reducing the number of the deployed Access Points (APs). First, we discuss the parameter settings of the IEEE802.11ad MAC which guarantee a low delay by exploiting the synchronous service periods (SPs) mechanism. Then, we propose a binary linear programming model to enhance the reliability of the system under the Frame Replication and Elimination for Reliability (FRER). The model minimizes the number of deployed APs by taking into account the constraints on the available synchronous SPs. We also propose two heuristics, based on the described mathematical model, in order to efficiently solve the design problem, then compare the computational performance of the proposed approaches. The reported computational results highlight the complementary features of the compared methods and provide useful guidelines about the usage of the proposed approaches.

Keywords: IEEE 802.11ad, Directional MultiGigabit Station (DMG STA), Industrial Automation, Time Sensitive Networking (TSN), Frame Replication and Elimination for Reliability (FRER), Service Period (SP).

[*] Corresponding author
Email address: rosario.garroppo@unipi.it (Rosario Giuseppe Garroppo)

1. Introduction

By considering a communications perspective, the revolution of Industry 4.0 is characterized by the sharing of network and computing resources between digital Information Technology (IT) and physical Operational Technology (OT) domains. In particular, the IT infrastructure needs to be enhanced in order to support the strictly time-synchronized and deterministic low latency communications required by some OT functions [1].

To this end, Time Sensitive Networking (TSN) [2][3] and its evolution to the wireless communication systems, i.e., Wireless TSN (WTSN) [4], have been introduced. WTSN brings many benefits given by the wireless networks, such as flexibility, easy reconfigurability, mobility, and lower maintenance and life-cycle costs. With respect to TSN, however, WTSN has to cope with different research challenges to guarantee time-sensitive communications with low latency and extremely high reliability [5].

Although some new wireless technologies have been specifically designed for the industrial environment [6], a simple integration between OT and IT may be obtained by referring to the IEEE 802 LAN technology. Indeed, TSN link layer capabilities can be mapped seamlessly from Ethernet to Wi-Fi, without architecture changes or protocol translation gateways. However, the IEEE 802 LAN technology needs to be enhanced in order to solve the issues related to device synchronization, low latency and high reliability of packet delivery. These issues have been analysed by referring to both the below 6 GHz and the 60 GHz systems. Time synchronization is a feature of TSN accomplished by IEEE 802.1AS, which is a profile of the IEEE 1588 precision time protocol (PTP) standard. The TSN requirement is to deliver time synchronization information over the IEEE 802.3 (Ethernet) or the IEEE 802.11 (Wi-Fi) links to achieve sub-microsecond clock error between devices [7]. Recent experimental results [8][9] show that a synchronization accuracy of 1 μs can be achieved by using integrated IEEE 802.11ac WiFi cards, whereas higher performance can be obtained by using a novel method of precise synchronization based on IEEE 802.11 networks with FPGAs [10]. The synchronization is necessary to implement time-aware traffic scheduling, defined in 802.1Qbv [11]. The schedule is based on time division multiple access (TDMA). In TSN, an extremely high reliability is obtained by referring to the sub-standard IEEE 802.1CB, which details the Frame Replication and Elimination for Reliability (FRER) strategy. FRER consists of sending duplicate copies of each frame over disjoint paths in order to provide proactive seamless redundancy. Therefore, the approach doubles the network resource consumption. To minimize the network congestion, FRER can be applied only to determined traffic classes (e.g., critical or time-sensitive traffic) and paths. The usage of IEEE 802.1CB into the IEEE 802.11 ecosystem requires, however, the definition of new mechanisms.

The IEEE 802.11be considers two alternative complementary enhancements [12][13]: 1) Multi-link operation (i.e., frequency diversity) based on the usage of separated bands to transmit the same frame between multi-link devices [14]; 2) Multi-AP operation, where the reception probability in the DownLink is improved by using a joint transmission (JTX) from different Access Points (APs). JTX provides a better performance gain compared to other multi-AP transmission schemes, at the expenses of an increased synchronization complexity [15]. Spatial diversity can then be exploited to increase the frame reception probability [16].

In addition to the systems operating in the spectrum below 6 GHz, IEEE defined the next generation WiFi standards, known as Wireless Gigabit (WiGig). The WiGig systems, i.e., IEEE 802.11ad [17] and 802.11ay [18], represent interesting solutions for having high performance wireless networks working in unlicensed spectrum in the mmWave band at 60 GHz. These standards make use of a large channel bandwidth (2.1 GHz), Multiple Input Multiple Output (MIMO) and channel bonding techniques to provide data rate of tens of Gbps. However, the few millimetre wavelength of the 60 GHz signal implies that these systems have to face radio propagation problems, due to the additional attenuation added by oxygen absorption, and to the difficulties in going through most of the physical obstructions. The coverage range of these systems can range from tens to about one hundred of metres, depending on the environments [19]. Due to this, the Directional MultiGigabit (DMG) MAC is designed by taking into account the usage of beamforming that enables the directional communications necessary to mitigate the propagation issues. Furthermore, to enhance the support of low latency communications, IEEE 802.11ad/ay introduces a contention-free channel access called Service Period (SP), which is used to reserve exclusive channel resources for the communication between a pair of nodes. For periodic traffic, such as the one generated in industrial automation applications, SP represents an important tool for guaranteeing low latency.

1.1. Paper contribution

In this work, we investigate the application of the 60 GHz IEEE 802.11ad technology in an industrial environment.

The paper contribution can be summarized in three points:

- We show how the delay constraints that are typical of the industrial applications can be satisfied by exploiting the synchronous SPs.
- We define a design model for minimizing the number of the APs. The minimization takes into account the maximum number of STAtions (STAs) that can have access to the synchronous SPs available in an AP. These constraints guarantee a low latency in data delivery. Simultaneously, the model takes into account the implementation of the FRER by using the multi-AP operation technique. Specifically, a pair of APs is assigned to each STA to provide independent radio channels for increasing the probability of a correct data delivery.
- We propose two heuristics to solve the optimization problem.

The paper is organized as follows. Section 2 presents the related work, while Section 3 summarizes the main elements of the IEEE 802.11ad technology which are necessary to understand the presented results. Section 4 introduces the radio channel models used in this study, by providing link-level simulation results aimed to obtain, for each Modulation and Coding Scheme (MCS), the link data rate and the minimum Signal to Noise Ratio (SNR) which guarantee a target Frame Error Rate (FER). The problem definition and the related assumptions are provided in Section 5, whereas Section 6 presents the procedure for determining the available SPs in each AP of the network. Section 7 presents the mathematical model we propose, while the heuristics for solving the addressed problem are described in Section 8. Section 9 presents the SP Slot (SPS) assignment problem, introducing a heuristic for the simple case where the SPS assignment can be performed without additional constraints. Section 10 reports the results of a wide computational experience, with an analysis of the main features of the presented methods and a brief discussion about their limitations. Some ideas on how to enhance/extend the proposed approaches are also provided. Finally, Section 11 concludes the paper.

2. Related Work

Recent studies show the capability of the IEEE 802.11ad technology [17] to wirelessly support high-performance applications, such as a backhaul mesh network of 5G [20] or a wireless solution for high-fidelity Virtual Reality (VR) headsets [21]. Our paper extends these works by studying the issues related to the application of IEEE 802.11ad to support TSN-based industrial applications.

Works related to IEEE 802.11ad usually focus on the performance evaluation of the channel access method, with a little consideration for the latency of data delivery [22][23]. A recent study [24] describes a method to configure the WTSN scheduling to meet the time budget of time-sensitive collaborative robotic applications, and validates it in a realistic industrial scenario. The paper describes two critical TSN features that extend IEEE 802.11, specifically, time synchronization (IEEE 802.1AS) and time-aware shaping (IEEE 802.1Qbv). Furthermore, the authors illustrate a typical hybrid TSN network architecture where the TSN capabilities in the wired segment are extended to the Wi-Fi segment of the network. The network is centrally managed as in most industrial IoT deployments. By assuming static STAs, our work analyses both the timing-related and the reliability aspects for supporting TSN-based applications using IEEE 802.11ad.

By considering 802.11 MAC/PHY, different strategies aimed at satisfying the low latency requirements of TSN have been studied [25], [26]. In particular, the 802.11ad/ay standards have introduced a contention-free channel access called Service Period (SP). SP provides a dedicated channel access which is exclusively reserved for communication between a pair of nodes. As shown in [27], admission control algorithms and SP can guarantee the performance requirements of isochronous traffic. An analytical model for access during SP is presented in [28]. The authors determine both the worst case and the average delay faced by SP packets, in different traffic conditions, for a variable length of the beacon interval and a random allocation of the asynchronous SPs to the nodes. The results show how a network designer can perform an optimal allocation of the SP and set the CBAP duration in order to achieve a trade-off between SP delay and CBAP throughput. The scheduling of the periodic traffic streams by using the SP allocation has been considered jointly with the admission control problem in [29], whereas in [30] the authors propose reinforcement learning (RL) based scheduling of the SP allocation which finds the optimal duration of each SP. Our paper assumes the use of isochronous or periodic SPs for low-latency communications. The proposed design model enables deterministic communication by scheduling isochronous or periodic SPs to each STA, by taking into account the maximum number of these special SPs available in each AP. Additionally, the design model selects an APs pair for each STA to facilitate the implementation of the FRER mechanism, which improves the reliability of data delivery on the radio interface. These objectives are achieved by the design model by minimizing the number of the APs and determining the optimal location of the deployed APs.

3. The DMG MAC of IEEE802.11ad

The standard IEEE802.11ad introduces new MAC functions for a new class of nodes that are called DMG STAs. A key aspect of the DMG STAs is the use of directional antennas, which enable mmWave connections while complying with legal power emission regulations and decreasing energy consumption. In this approach, all the possible directions from an STA are divided into predefined sectors. The beamforming training allows the STA to determine an optimal sector, i.e., the one having the best quality of the radio channel. The training is based on a sequential bidirectional transmission of beamforming training frames for each sector. This procedure is typically done during the initial setup and the STA association, with optional refinements during the data transfer to account for changes in the environment or the device movement [17].

The channel access by a DMG STA occurs during Beacon Intervals (BIs), and it is coordinated using a schedule. As shown in Figure 1, the BI structure is divided into two parts: the Beacon Header Interval (BHI), used for traffic control, and the Data Transmission Interval (DTI), used for transmitting user data.

The BHI is organized into subperiods as follows:

- BTI (Beacon Transmission Interval) is the beginning phase of the BHI. During this period the AP may transmit Beacon Frames (BFs), informing any STA about its existence, its capabilities, and the specific structure of the remainder of the BI. BFs are transmitted at the lowest Modulation and Coding Scheme (MCS), lengthening the transmission.

- A-BFT (Association BeamForming Training) represents the phase where the STAs may be associated with the AP, and where they exchange

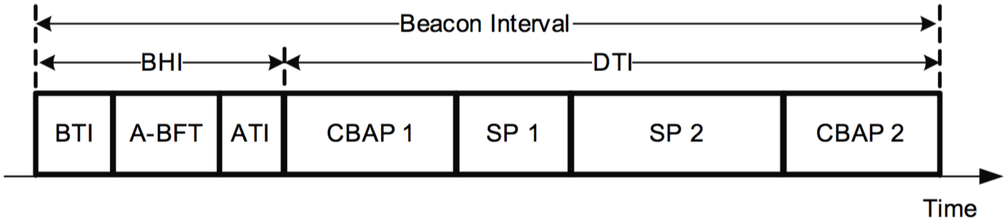


Figure 1: Beacon Interval (BI).

frames with the AP in the beamforming process, in which the optimal sector is selected. The A-BFT phase is divided into several slots, and each STA picks one of them randomly in a contention-based approach.

- ATI (Announcement Transmission Interval) is the period used by the AP to exchange management information with the already associated STAs through a unicast, higher-MCS request-response mechanism. During this period, a management access mechanism based on request-response is used by the STA for resource allocation and security association.

The DTI comprises contention-based access periods (CBAPs) and scheduled SPs. During the CBAP, the channel access is quite similar to the EDCA, with some improvements that take into account the high available data rate and the utilization of directional antennas. In other cases, the AP schedules SPs for each STA of the network. The organization of the DTI is communicated by using the optional Extended Schedule element in the BFs transmitted during the BTI. Other than having two fields giving general information on the element (such as type and length), this element is composed of a set of 15 octets Allocation fields [17]. These fields provide the starting time of each individual allocation of an SP (or CBAP) by means of this relation:

$$A_{start} + (i-1) * A_{period}, \tag{1}$$

where

- A_{start} is the value of the Allocation Start field for the SP or CBAP;
- A_{period} is the value of the Allocation Block Period field for the SP or CBAP; the time unit is μs and the range is $[1, 32767]$ for the SP;

- i is an integer greater than 0 and less than or equal to the Number of Blocks field for the SP or CBAP.

The end of the i-th individual SP or CBAP allocation is computed by adding the Allocation Block Duration field to the start time of the i-th individual allocation.

A DMG STA can indicate two types of allocation scheduling: isochronous and asynchronous. It should establish an isochronous allocation if it needs periodic access to the channel, and the amount of time frequently allocated is expected not to change. Alternatively, it should create an asynchronous allocation if it wants to reserve a specific amount of channel time to fulfil future requests when they happen.

The Allocation Block Period is set to an integer multiple or to a submultiple of BI to keep the allocation position consistent with the beginning of a beacon interval. The first setting allows one SP allocation with a period equal to BI or to a multiple of it, while the second one permits multiple SP allocations per BI, which are repeated every BI.

An STA accesses the channel during BI coordinated by a scheduling algorithm in the AP. As shown in Figure 1, the DTI is divided into SPs and CBAPs. Based on the QoS requirements of the packets, the SP or CBAP resources can be used. Packets with stringent QoS (e.g., gaming, augmented reality, isochronous services, etc.) are transmitted using SPs, while CBAPs are used by packets that do not have stringent QoS requirements. During the ATI, the AP polls all the STAs and, in response to the poll, the node sends a service period request (SPR). The AP can either accept or reject the SPR. If the request is accepted, in order to announce the reservation, the AP uses the Extended Schedule element in the BF of the BI in which the SP is allocated.

4. Radio Links Model

At 60 GHz, the wireless channel is unstable and some events, such as moving humans and obstacles, can lead to dramatic results such as the blockage of the communication. However, some experimental works [31][32][33] and standardization activities [34] have defined suitable channel models.

Despite these important past studies, the understanding of the mmWave channel properties in factory automation environments still remains very limited. Differently from other indoor environments like offices, hotels or restaurants, the factory layouts incorporate a wide range of metallic equipment, robots and painted materials that strongly affect radio signal propagation.

Recent works have shown some peculiarities of the mmWave radio links in the industrial environment scenario. In [35], the authors study the important mmWave channel properties by considering two different kinds of factories: light industry and heavy industry [36]. These represent the extreme cases of a factory classification based on the level of technology, the density and the size of the equipment, and the goods produced. Indeed, the light industry is typically characterized by consumer-oriented factories, whose products (e.g., clothes, shoes, furniture) are directed to end-users. The scenario of these factories is composed of tables, chairs and shelves, which may have a relatively small impact on radio propagation. On the other hand, heavy industry includes factories that deploy larger and heavier equipment (e.g., automotive, aerospace, chemical), including large machine tools, automated robots, and intricate shapes of buildings, which aid in executing complex processes. This scenario thus complicates radio propagation.

In the two factory automation scenarios discussed before, the deployment of mmWave radio systems may be challenging due to shorter wavelengths [37]. To overcome this issue, in [35] the authors investigate the relevant channel properties of two mmWave frequency bands, i.e., 28 and 60 GHz, which contain important spectrum resources for licensed and unlicensed-band communications, respectively. Their findings suggest that the deployment of the mmWave systems in indoor industrial environments should not rely blindly on past propagation studies available in the literature, but they should take into account a more accurate and reliable evaluation of the environment. In heavy factories, they observe that the mmWave channel properties are more difficult to predict due to more reflections caused by the higher density of metallic machines/robots. In this scenario, accurate propagation models can be obtained only through mmWave simulations in a 3D ray-based modelling tool. The results of their study are the models of the simulated industrial scenarios in terms of i) Line-of-Sight (LoS) and Non-LoS (NLoS) path loss and (ii) LoS probability. The starting point is the following general path loss formula (scaled in dB):

$$PL(dB) = PL(d_0) + 10\beta \log_{10}\left(\frac{d}{d_0}\right) + \chi_\sigma \tag{2}$$

where $PL(d_0)$ represents the path loss at the reference distance d_0 (often named the free-space path loss), β is the path loss exponent, d is the distance (in metres) between the Tx and the Rx, and χ_σ is a lognormal random variable with 0 dB mean and standard deviation σ. In Table 1, we report the parameters of the path loss formula (2) related to the 60 GHz band in a factory automation deployment [35].

Table 1: Parameters of the path loss formula in the considered factory automation deployments - 60 GHz [35].

Industry	$PL(d_0)$[dB]	β	σ
LOS			
Light	62.7	2.05	4.53
Heavy	64.7	1.91	3.99
NLOS			
Light	64.0	2.35	6.29
Heavy	26.8	5.67	10.07

Regarding the LoS probability, the results of [35] suggest that, for the environments of interest, the classical statistical models may be inaccurate. The curve fitting the results obtained by the 3D-ray tracing simulation indicates that the LoS probability can be expressed by the following analytical expressions:

$$P_{LoS}^{Light}(d) = \begin{cases} 1, & \text{if}\, d \leq 8m \\ 111\exp(-0.01829\, d)+ & \\ +(-0.0002933\exp(0.3443\, d)), & \text{if}\, 8m < d < 35m \\ 0, & \text{if}\, d \geq 35m \end{cases} \tag{3}$$

$$P_{LoS}^{Heavy}(d) = \begin{cases} 1, & \text{if}\, d \leq 8m \\ 153.1\exp(-0.1141\, d)+ & \\ +483.2\exp(-0.3254\, d), & \text{if}\, d > 8m \end{cases} \tag{4}$$

In our numerical analysis, in order to calculate the path loss we first evaluated the distance d between the Tx and the Rx. Then, depending on the industry scenario, we applied the relation (3) or (4) to calculate the probability to have an LoS link between Tx and Rx. Finally, depending on the extracted value of P_{LoS}, we calculated the path loss by means of formula (2), by setting the parameters according to Table 1. All the simulations refer to the 60 GHz band and to the heavy industry scenario reported in [35].

4.1. Estimation of Link Data Rate from the Path Loss P_L

To estimate the link data rate from P_L, first, we carried out an end-to-end simulation to determine the frame error rate (FER) for the 802.11ad Single Carrier (SC) link, with an AWGN channel at a selection of SNR values. The simulation has been carried out using MATLAB 2020a. For each considered modulation and coding scheme (MCS), at each SNR value, multiple frames are transmitted through the AWGN channel. The received frames are compared with those transmitted to determine the frame error rate (FER). The power of the AWGN is set in such a way as to obtain the considered SNR.

The frame size is 200 bytes (i.e., 34 of MAC overhead and 166 bytes of data at the MAC layer). Such a small packet size is suggested by the scenario denoted as Factory of the Future in [38].

For each MCS, we selected the SNR range to simulate by taking into account the transition from "all frames being decoded with errors" to "all frames being successfully decoded". In this range, we analysed the values at steps of 0.5 dB. For each SNR and MCS point, the FER estimate has been obtained by transmitting 10^7 frames. In the simulation, the clock accuracy considered to derive the carrier frequency offset (CFO) is set to 20ppm, whereas a delay of 500 samples is added to the transmitted signal. The results of the simulation are shown in Figure 2.

By assuming that each station is able to perform the SNR estimation without errors, then the choice of the MCS according to the curves in the figure gives the maximum link data rate of the IEEE 802.11ad device that guarantees a desired FER on the link. In our study, by considering a FER lower than 10^{-5}, for each MCS we can thus obtain the SNR range and the available link data rate as reported in Table 2.

Also assuming that the mmWave beams employed for the communications are narrow enough to make negligible the risk of beam collisions, we can neglect the interference and calculate the SNR as:

$$SNR = \frac{G_t \cdot G_r \cdot P_t \cdot PL^{-1}}{N_0}, \tag{5}$$

where G_t and G_r are the antenna gain of the transmitter and of the receiver respectively, P_t is the transmission power, N_0 is the noise floor at the receiver,

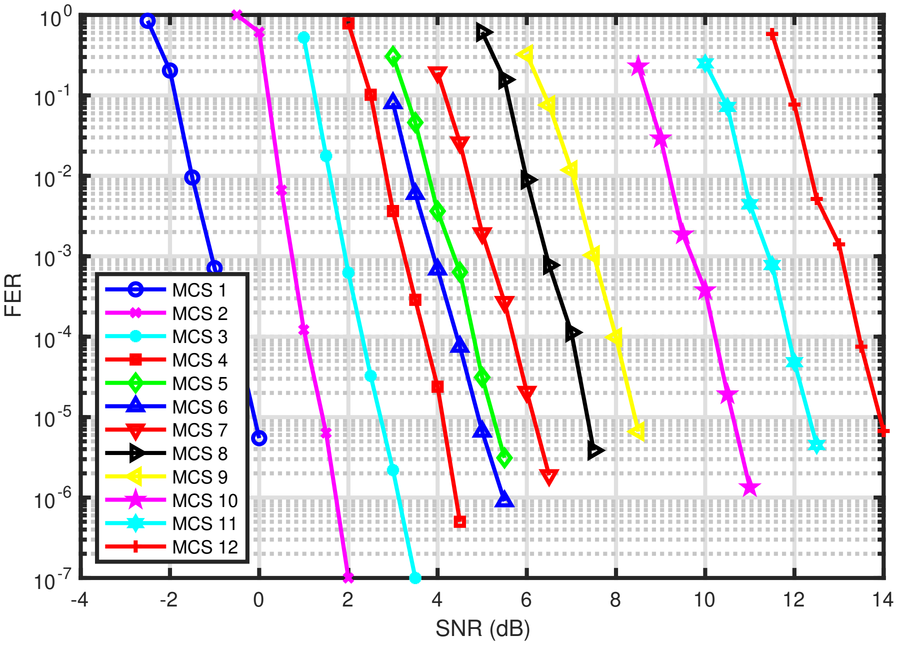


Figure 2: FER vs SNR for $MCS = 1, \cdots, 12$ - Frame length 200 bytes

and PL is the path loss of the wireless channel between the transmitter and the receiver. As reported in [39], N_0 at the receiver can be calculated according to:

$$N_0(dB) = -174 + NF + 10\log_{10} B, \tag{6}$$

where NF is the noise figure of 1.5 dB, and B is the channel bandwidth of the IEEE 802.11ad channel (i.e., 2160 MHz).

5. Problem definition and assumptions

Referring to the IEEE 802 technologies, the coexistence of different classes of traffic is guaranteed by two TSN sub-standards: IEEE 802.1Qbu implements a frame pre-emption to interrupt any ongoing operation in case a time-sensitive frame is selected for transmission, whereas IEEE 802.1Qbv creates exclusive time slots for time-sensitive frames managed by a time-aware shaper. On the other hand, the IEEE 802.11ad offers the SPs, which can guarantee the data transmission without contention when coordinated

Table 2: Link data rate (DR in Mbps) and minimum SNR (in dB) which guarantee a FER $\leq 10^{-5}$, for each MCS in Figure 2.

MCS	Min. SNR	DR	MCS	Min. SNR	DR
1	0.0	385	7	6.5	1925
2	1.5	770	8	7.5	2310
3	3.0	962.5	9	8.5	2502.5
4	4.5	1155	10	9.5	3080
5	5.5	1251.25	11	11.0	3850
6	5.0	1540	12	12.5	4620

by a central scheduler. The directional antenna of the IEEE 802.11ad devices and the SPs allow the implementation, in a coordinated manner, of a TDMA-like access mechanism for TSN traffic. An admission control mechanism allows the protection of time-sensitive traffic from SPs resource overloading. Only the traffic associated with the CBAP part of the DMG MAC may collide. Hence, a key aspect is to define the available resources (i.e., SPs for each AP) for the TSN traffic. The DMG MAC of IEEE 802.11ad allows the transmission of time-sensitive and non-time-sensitive traffic on a periodic basis with isolation between the two classes. Section 6 presents a procedure for evaluating the available SPs in an 802.11ad AP, pointing out the related assumptions.

In TSN, a reliable packet delivery is considered by the sub-standard IEEE 802.1CB, which basically sends duplicate copies of each frame over disjoint paths to provide proactive seamless redundancy. To obtain disjoint paths in an IEEE 802.11ad infrastructure, one must plan the wired network connecting the APs to the rest of the network, and to coordinate the APs which are responsible for the TSN data exchange over the wireless link.

The planning of the wired infrastructure is out of the scope of this paper. On the other hand, aim of the model presented in Section 7 and of the heuristics described in Section 8 is to determine the number and the location of the APs, and to provide each STA with two independent wireless links while considering the maximum number of SPs available in each AP. The two links are used for implementing the FRER by using the multi-AP operation. Indeed, the FRER mechanism can be applied without the need of transmitting the same frame from different APs at the same time. However, in the case it would be necessary to satisfy this additional constraint, in Section 9 we present a simple heuristic to determine the SP Slots (SPS) to use in the APs pair, in a coordinated manner, during their simultaneous communication with an STA. This coordination mechanism is based on the assumption of having a common clock for APs and STAs. The TSN sub-standard IEEE 802.1AS includes a version of the Precision Time Protocol (PTP), which enables the distribution of a single reference clock across the network devices on a master/slave basis. Having a common clock for APs and STAs allows to use the allocated SPSs for transmitting either uplink or downlink TSN traffic, depending on the application needs. In the presented model, however, we do not take into account the direction of the traffic (i.e., uplink or downlink), because this depends on the specific application addressed.

The proposed model is also based on the assumption that the stations are stationary with known traffic patterns. This assumption can be accepted when the TSN communication is used for monitoring and management of stationary industrial things, as well as for direct control of machines for factory automation representing the main components of the shop floor [40]. The IEEE 802.11ad infrastructure can support the wireless TSN communications which are necessary for exchanging data between Programmable Logic Controllers (PLCs), Human-Machine Interface (HMIs) and IIoT devices (i.e., sensors and actuators). Note that the lifetime of the shop floor configuration tends to be in the order of years. Therefore, unlike in the commercial networking application context, the IEEE 802.11ad network infrastructure, when correctly designed, can support the TSN application for years, without the need of modifications.

6. System configuration for low delay traffic

The time which is necessary to transmit a frame can be calculated by considering the configuration parameters of the PHY layer, the frame length and the MCS. Specifically, the timing related parameters for calculating the latency in the case of the IEEE 802.11ad SC systems are summarised in Table 3 [17].

As shown in Figure 3, T_{STF} and T_{CE} are parts of the Physical Protocol Data Unit (PPDU) preamble.

The duration of the preamble is then $T_P = T_{STF} + T_{CE} = 3328 \cdot T_c$. The PHY Header has a fixed duration of $T_H = 1024 \cdot T_c$, while T_{DATA} is the time necessary to transmit the Physical Service Data Unit (PSDU), i.e., the MAC

Table 3: Timing related parameters of IEEE 802.11ad SC systems, from Table 21-4 in [17].

Parameter	**Value**
T_c: Chip Time	0.57 ns
T_{seq}	$128 \cdot T_c = 72.7$ ns
T_{STF}: Detection sequence duration	$17 \cdot T_{seq} = 1236$ ns
T_{CE}: Channel Estimation sequence duration	$9 \cdot T_{seq} = 655$ ns
T_H: Header duration for SC systems	$1024 \cdot T_c = 0.582$ ns
T_{DATA}: Time to transmit the PSDU	$(N_{BLKS} \cdot 512 + 64) \cdot T_c$

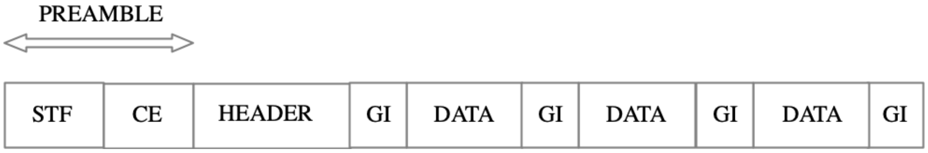


Figure 3: SC PHY PDU

frame. Differently from T_P and T_H, T_{DATA} depends on the used MCS in addition to the PSDU length, i.e., L. As reported in Table 3, T_{DATA} is a function of the number of symbol blocks (N_{BLKS}) used to transmit the PSDU bits. Every block is composed of 64 symbols used as a guard interval (GI) and of 448 symbols used for transmitting the PSDU (represented as `DATA` in the figure). To calculate N_{BLKS}, it is necessary to consider the LDPC encoding process, as detailed in [17]. We summarize below the definitions of the elements used to calculate N_{BLKS}:

- L is the length of PSDU;
- L_{CW} is the length of the LDPC codewords, which is fixed to 672 bits;
- N_{CW} indicates the number of the LDPC codewords;
- N_{CBPB} is the number of the coded bits per block composed of 448 symbols;
- N_{DP} represents the number of the padding bits.

The first step consists of calculating N_{CW} by means of the following formula:

$$N_{CW} = \left\lceil \frac{L}{R \cdot \frac{L_{CW}}{\rho}} \right\rceil, \tag{7}$$

where R is the code rate of the MCS, while ρ is the repetition factor. Only the MCS 1 has $\rho = 2$, whereas the others have $\rho = 1$. In (7), the denominator considers the overhead added by the coding rate and the replication mechanism. This overhead reduces the amount of PSDU bits that can be transmitted in the 672 bits of the codeword.

Given the ceiling operation, N_{DP} padding bits need to be added to L:

$$N_{DP} = N_{CW} \cdot L_{CW} \cdot R - L. \tag{8}$$

Since each block is composed of 448 symbols, N_{BLKS} can be obtained by dividing the number of the bits which are coded after the LDPC encoding by the number of the bits transported by the 448 symbols of the single block:

$$N_{BLKS} = \left\lceil \frac{N_{CW} \cdot L_{CW}}{N_{CBPB}} \right\rceil \tag{9}$$

In (9), N_{CBPB} depends on the modulation rate R_m. Specifically, R_m is 1, 2 or 4 depending on the symbol mapping, i.e., $\pi/2 - BPSK$, $\pi/2 - QPSK$ or $\pi/2 - 16QAM$, respectively. In particular, N_{CBPB} is equal to 448 times R_m.

The transmission latency T_{PPDU} can be calculated by summing up T_P, T_H and T_{DATA}, i.e.,

$$\begin{aligned} T_{PPDU} &= (T_P + T_H + (N_{BLKS} \cdot 512 + 64)) \cdot T_c = \\ &= (3328 + 1024 + N_{BLKS} \cdot 512 + 64) \cdot T_c = \\ &= (4416 + 512 \cdot N_{BLKS}) \cdot T_c \end{aligned} \tag{10}$$

The impact of the selected MCS on (10) is twofold:

- in calculating N_{CW} using (7), it is $\rho = 1$ for all MCSs except MCS 1, and therefore R has a value in the set $\{1/2, 5/8, 3/4, 13/16\}$ depending on the MCS;
- in (9), N_{CBPB} depends on MCS through R_m.

By assuming that one SP is used to transmit $N_{BLKS} = 10$, we can estimate the minimum time duration of an SP and the maximum PSDU that can be transmitted. In fact, setting $N_{BLKS} = 10$ in relation (10), we obtain $T_{PPDU} = 9536 \cdot T_c = 5.43$ μs. In the remainder of the paper, we shall assume that the duration of an SP is equal to 10000 T_c, which is slightly greater than T_{PPDU} because we consider a time guard interval. As a consequence of this assumption, we have $T_{SP} = 5.7$ μs. By just taking into account the MCS with $\rho = 1$, the worst case is represented by MCS 2. Therefore, by considering the previous formulas and the MCS parameters of the SC, reported in Table 21.18 of [17], this MCS allows the PSDU transmission of utmost 2016 PSDU bits in the T_{PPDU}. In other words, in the considered conditions, a MAC PDU of 252 bytes can be transmitted within the SP of 5.7 μs in the worst case.

The DMG MAC of IEEE 802.11ad has different configuration parameters that impact the QoS. In our study, we assume that the IEEE 802.11ad system is configured with the duration of the BI equal to $T_{BI} = 1$ ms. However, this setting implies some assumptions about the BHI. Indeed, the default BHI configuration for an 8-sector AP takes 1.664 ms (see [21]). To reduce the BHI length, first, we can exploit the fact that, according to the standard, the BTI information, transmitted at the lowest MCS, is not mandatory in every BHI. In fact, the standard only requires that BTI be present once every 15 BIs. This configuration has the important effect that the STAs will no longer receive a BF every BI. As a result, the STA does not know which allocations were assigned within the DTI for these BIs without a BF. To overcome this problem, the allocations can be marked as Pseudo-Static (PS). These allocations are assumed to be repeated, each time at the same offset from the beginning of the BI. Also, CBAP can be considered PS, by using the `CBAP-only` flag which informs the STAs about a CBAP allocation spanning over different BIs. Therefore, the reduced number of BFs does not affect the STAs' ability to participate in data transfer during the PS allocations, as long as no BFs are lost.

Secondly, also the A-BFT is required only once per 15 BIs, and it can have as few as one slot. This configuration affects the performance when regular beamforming is necessary due to changes in the STA's location or in the environment dynamics.

The two improvements above allow one to reduce the worst-case BHI duration of an 8-sector AP to $T_{BHI} = 249\mu$s [21]. Then, if we apply an admission control algorithm, such as the one in [27], and use the PS allocations mechanism for the SP, each node will have a periodic SP with a period equal to $T_{BI} = 1$ ms.

The next step is to calculate the maximum number of SPs available at each AP. For this computation, we exploit the information that, according to the standard, the minimum amount of the DTI time dedicated to the CBAP traffic is $T_{CBAP} = 500 \mu s$. The maximum number of SPs available for each BI in each AP, i.e. N_{SP}, can then be calculated as:

$$N_{SP} = \left\lfloor \frac{T_{BI} - T_{BHI} - T_{CBAP}}{T_{SP}} \right\rfloor = 44 \tag{11}$$

In the remainder of the paper, we shall denote by N_S the maximum number of SPs statically allocated to each AP for isochronous services or for periodic transmission of data. Moreover, we shall assume $N_S = N_{SP}$. In case $N_S < N_{SP}$, the additional SPs can be used for delay-sensitive asynchronous traffic. The allocation of these SPs requires the request-response procedure in the ATI sub-interval of the BHI. In this case, depending on when the packet arrives at the transmission queue of the node, the delay added by the channel access can be higher than T_{BI}, as shown by the results in [28].

7. The mathematical model

The optimization problem we address consists of minimizing the number of the APs while satisfying the constraints on the maximum number of SPs available in each AP and assigning two different APs to each STA in order to have independent wireless links, with the aim of increasing the frame delivery probability. Preliminary studies may be carried out in order to establish which pairs of APs can provide independent radio links while satisfying the constraints on the channel data rate. As an example, the independence of the radio links can be related to the property that the angle between the two links is higher than a given threshold, e.g., $\pi/2$. The rationale is that, if a temporary obstacle on one link worsens the radio link quality, then the angular distance should reduce the probability that this obstacle may have an impact also on the alternative link.

The main symbols used to describe the problem, the corresponding mathematical model, and the heuristics we propose to solve it, are summarized in Table 4.

Symbol	Meaning
N_S	Maximum number of SPs statically allocated to each AP for each BI.
N_{SP}	Maximum number of SPs available for each BI.
$\mathcal{I}$	Set of the STAs.
$\mathcal{J}$	Set of the candidate AP locations.
$\mathcal{C}$	Set of all possible pairs of APs, $\mathcal{C} = \{(h,k) : \forall h,k \in \mathcal{J}, h \neq k\}$.
Ψ	Matrix with elements $\psi_{i,c}$ for each STA i and each pair $c = (h,k) \in \mathcal{C}$, indicating if the pair c can be used for STA i (1) or not (0)
y_j	binary variable, $\forall\ j \in \mathcal{J}$, indicating if AP j is powered on (1) or not (0).
$x_{i,c}$	binary variable, $\forall\ i \in \mathcal{I},\ c \in \mathcal{C}$, indicating if STA i is associated with the pair of APs c (1) or not (0).
$\mathbf{x}_k$	Solution of the dual problem at step k for variable x of the primal problem.
$\mathbf{y}_k^*$	Solution of the dual problem at step k for variable y of the primal problem.
$\mathbf{y}_k$	Feasible solution of primal problem computed starting from $\mathbf{x}_k$ considering the constraints (14) .
$W(\pi^k)$	Dual objective with dual multipliers π^k.
$\overline{W}$	Upper bound of the dual objective.
$\nabla W(\pi^k)$	Subgradient of the dual problem with dual multipliers π^k.
π^k	Dual multipliers at step k.
t_k	Weight of $\nabla W(\pi^k)$ for updating π^k.
G_c	Bipartite graph defined for the flow-based approximation problem, $G_c = (N_1, N_2, A)$.
A	Set of arcs in G_c; each arc (j,h) corresponds to a pair of APs which is candidate for some STAs.
A_i	Subset of A corresponding to the candidate pairs for STA i.
f_{uv}	Nonnegative and integer flow variable for each arc $(u,v) \in A$.
$BS(u)$	Backward star of node u.

$FS(u)$	Forward star of node u.
$\mathcal{C}^*$	Subset of candidate pairs in the flow-based approximation heuristic, $\mathcal{C}^* = \{(j,h) \in \mathcal{C} : f^*_{jh} \geq 1\}$.

Table 4: Summary of the meaning of the used symbols

Let $\mathcal{I}$ denote the set of the STAs, and $\mathcal{J}$ the set of the candidate AP locations. By *powered on AP* or *activated AP* we mean a candidate location which is selected to have a working AP. Moreover, a *candidate pair* for the STA i is a pair of APs providing links that satisfy the constraints on the offered data rate as well as the independence property with respect to i. For each wireless link, the available data rate is obtained by using Table 2 with SNR given by formula (5) in Section 4. Moreover, regarding the independence property, we assume that it is satisfied if the angle between the two links is higher than $\pi/2$.

As an additional constraint, the total number of STAs requesting synchronous SPs, i.e., assigned to each powered on APs, must be less than or equal to N_S, which in turn must be lower than or equal to N_{SP}, calculated as reported in Section 6.

Let $\mathcal{C} = \{(h,k) : \forall h,k \in \mathcal{J}, h \neq k\}$ be the set of all the possible pairs of APs. Moreover, let Ψ be a matrix, having one row for each STA and one column for each pair of APs, which indicates the candidate pairs of the STAs. Precisely, for each STA i and each pair $c = (h,k) \in \mathcal{C}$:

$\psi_{i,c}$ is 1, if the data rate of both links (i,h) and (i,k) is higher than a given threshold T_d and the angle between the two links is higher than the threshold T_a, and it is 0 otherwise.

Note that, besides the method described in Section 4 and the link independence assumptions, Ψ can alternatively be calculated through site survey measurements or 3D-ray tracing (as in [35]) applied to the actual scenario.

Given the input data above, we define the following sets of binary variables:

- y_j: set to 1 if AP j is powered on (i.e., if the candidate AP location j is selected), and to 0 otherwise, $\forall\ j \in \mathcal{J}$;

- $x_{i,c}$: set to 1 if STA i is associated with the pair of APs c, and to 0 otherwise, $\forall\ i \in \mathcal{I}$, $c \in \mathcal{C}$.

The considered design problem can then be formulated by means of the following Binary Linear Programming (BLP) model:

$$z = \min \sum_{j \in \mathcal{J}} y_j \tag{12}$$

$$\sum_{c \in \mathcal{C}} \psi_{i,c} x_{ic} = 1 \qquad i \in \mathcal{I} \tag{13}$$

$$\sum_{i \in \mathcal{I}} \sum_{c \in C_j} \psi_{ic} x_{ic} \leq N_S y_j \qquad j \in \mathcal{J} \tag{14}$$

where $C_j = \{c \in \mathcal{C} : j \in c\}$ is the subset of $\mathcal{C}$ composed of pairs of APs comprising j.

Constraints (13) assign one candidate pair c to each STA, while (14) are both capacity and linking constraints. In fact, for each AP j, they state that if $y_j = 0$, i.e., the AP location j is not activated, then no candidate pair comprising j can be assigned. Otherwise, i.e., $y_j = 1$, then (14) guarantee that the number of STAs assigned to j does not exceed N_S.

Observe that the model has $|\mathcal{J}| + O(|\mathcal{I}||\mathcal{J}|^2)$ variables and $(|\mathcal{I}| + |\mathcal{J}|)$ constraints. In particular, the number of used variables may be huge.

We conclude this section with a proof of the time complexity of the problem addressed.

Theorem 7.1. *The problem is NP-Hard.*

Proof: The proof is by reduction from the Set Covering Location Problem (SCLP), which is NP-complete in its decisional form [41]. Consider an instance of the decisional form of SCLP, say P_1. Using the notation introduced in [42], given a set I of demand nodes, a set J of candidate facility locations, a set N_i containing those candidate facility locations that can "cover" the demand point i, i.e., can satisfy the request of i, $\forall i \in I$, and given an integer number K, P_1 asks to verify whether it is possible to cover all the demand nodes by locating at most K facilities.

Given P_1, define the following instance of the decisional version of the problem addressed in this paper, say P_2. In P_2 there is one STA for each demand node in P_1, i.e., $\mathcal{I} = I$. Moreover, there is one AP for each candidate facility location plus a special one, denoted as l. That is, $\mathcal{J} = J \cup \{l\}$. Regarding the set of the pairs of APs which are candidates for the STA i, i.e., the pairs of APs that can be assigned to i, we define it as $\{(j, l) : j \in N_i\}$, $\forall i \in \mathcal{I}$. Finally, let the capacity of the APs, i.e., N_S, be very high in P_2 so as to result negligible. For example, $N_S = |\mathcal{I}|$. Given the parameter setting introduced above, P_2 asks to verify whether it is possible to assign a candidate pair to each STA by activating at most K APs in the set J, in addition to the special AP l. Notice in fact that, by construction, the special AP l must be necessarily activated in order to get a feasible solution.

By construction, it is possible to cover all the demand nodes by locating at most K facilities, i.e., the instance P_1 is feasible, if and only if it is possible to assign a candidate pair to each STA by activating at most K APs in the set J, in addition to the special AP l, which is mandatory. Equivalently, if and only if P_2 is feasible. The thesis follows. □

Since the capacity constraints come not into play in the presented proof, as a corollary we get that also the uncapacitated version of the problem addressed is NP-Hard.

8. The heuristics

In order to solve the design problem, we propose two different heuristics: one based on a Lagrangian relaxation, and the other one exploiting a flow-based version of the mathematical model.

8.1. Lagrangian Relaxation Based Algorithm

Consider the mathematical model in Section 7, which represents an exact formulation of the addressed design problem, and dualize with respect to constraints (14). By simple algebra, we get the following expression of the Lagrangian function:

$$L(\mathbf{x}, \mathbf{y}, \pi) = \sum_{j \in \mathcal{J}} (1 - \pi_j N_S) y_j + \sum_{j \in \mathcal{J}} \pi_j \sum_{i \in \mathcal{I}} \sum_{c \in C_j} \psi_{ic} x_{ic} \tag{15}$$

where π is the vector of the Lagrangian multipliers corresponding to the constraints (14). Given π, the corresponding Lagrangian problem is given by

$$W(\pi) = \min_{\mathbf{y},\mathbf{x}} L(\mathbf{x}, \mathbf{y}; \pi) \tag{16}$$

$$\sum_{c \in C} \psi_{i,c} x_{ic} = 1 \qquad i \in \mathcal{I} \tag{17}$$

$$\pi_j \geq 0 \qquad j \in \mathcal{J} \tag{18}$$

The solution to this problem returns values to the primal binary variables $\mathbf{x}$ and $\mathbf{y}$, which are dependent on the fixed values of π. The corresponding Lagrangian Dual problem is given by

$$\max_{\pi \geq 0} W(\pi) \tag{19}$$

According to [43], for each given $\pi \geq 0$, $W(\pi)$ provides a lower bound on the optimal value of (12)-(14). Consequently, the Lagrangian Dual problem provides the best possible Lagrangian lower bound, i.e., the maximum one. The solution of the Lagrangian Dual problem can be efficiently found in an iterative manner by solving (16) for a given π, so obtaining values for the binary variables $\mathbf{x}$ and $\mathbf{y}$, and then changing π in order to find the solution to the problem (19) by using a subgradient technique.

The advantage of the considered Lagrangian problem is that it can be decoupled in the following two binary subproblems:

$$\min_{\mathbf{y}} L_y(\pi) = \min_{\mathbf{y}} \left[\sum_{j \in \mathcal{J}} (1 - \pi_j N_S) y_j \right] \tag{20}$$

and

$$\min_{\mathbf{x}} L_x(\pi) = \min_{\mathbf{x}} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \pi_j \sum_{c \in C_j} \psi_{ic} x_{ic} \tag{21}$$

$$\sum_{c \in \mathcal{C}} \psi_{i,c} x_{ic} = 1 \qquad i \in \mathcal{I} \tag{22}$$

For a given nonnegative π^*, the subproblem (20) can be easily solved. Indeed, the minimum is obtained by assigning the following values to $\mathbf{y}$:

$$y_j = \begin{cases} 1 & \text{if } \pi_j^* > 1/N_S \\ 0 & \text{otherwise} \end{cases} \qquad j \in \mathcal{J} \tag{23}$$

In more detail, this setting implies that we do not add the positive terms $(1-\pi_j^* N_S)$ to $L_y(\pi^*)$ by setting the corresponding y_j variables to 0, and that on the other hand we add the negative terms by setting the corresponding y_j variables to 1.

For the solution of subproblem (21)-(22), for each STA i, the constraints (22) indicate that we must select only one candidate pair from $\mathcal{C}$. Thus, an optimal solution can be found by defining the subset $S_i = \{(h,k) \in \mathcal{C} : \psi_{i,(h,k)} = 1\}$ for each STA i, and then assigning the STA i to the pair $\overline{c} = (\overline{h}, \overline{k})$ having the minimum value of $\pi^*_{\overline{h}} + \pi^*_{\overline{k}}$ in S_i.

For a given π^*, the presented strategies permit the computation of the optimal solutions x^* and y^* of subproblems (20) and (21)-(22). Using these results, we can easily estimate a subgradient corresponding to them taking into account that in general $\nabla W(\pi) = (\partial W(\pi)/\partial \pi_1, \partial W(\pi)/\partial \pi_2, ... \partial W(\pi)/\partial \pi_{|\mathcal{J}|})$, where

$$\partial W(\pi)/\partial \pi_j = \sum_{i \in \mathcal{I}} \sum_{c \in C_j} \psi_{ic} x^*_{ic} - N_S y^*_j \qquad \forall j \in \mathcal{J} \tag{24}$$

The subgradient $\nabla W(\pi)$ is then used to select a new π for computing the solution of (19) [43].

8.1.1. Iterative Solution of the Lagrangian Dual Problem

The proposed Lagrangian heuristic is based on an iterative algorithm aimed at finding the solution to the Lagrangian Dual problem. As shown in Algorithm 1, the algorithm begins with the setting of initial conditions of some variables and some parameters (lines 1-2). The loop is based on a sequence of 4 steps. In the first step, indicating the Lagrangian multipliers π^* found at step k by π^k, we compute the $\mathbf{y}^*_k$ by using relation (23). Moreover, we compute $\mathbf{x}_k$ as indicated before. That is, for each i, we look for the pair $\overline{c} = (\overline{h}, \overline{k}) \in S_i$ giving the minimum value $\pi^*_{\overline{h}} + \pi^*_{\overline{k}}$ (line 5). In general, ($\mathbf{y}^*$,$\mathbf{x}_k$) can be infeasible to the primal problem, due to the relaxation of constraints (14). Thus, a primal feasible solution has to be computed based on the solution $\mathbf{x}_k$ at iteration k, in order to obtain a vector $\mathbf{y}_k$ satisfying (14) (line 6).As the iterations progress, we store the primal feasible solution with the minimal primal cost, until the end of the loop (lines 8-10). Moreover, since ($\mathbf{y}^*$,$\mathbf{x}_k$) is not necessarily an optimal solution to the Lagrangian Dual problem, for solving the dual problem we compute the subgradient $\nabla W(\pi^k)$ using the relations (24) and the obtained $\mathbf{y}_k^*$ and $\mathbf{x}_k$ (line 14). In the lines 15-17 we update the dual objective value using (15), the step size t_k and the dual variables π^{k+1} by means of the following relations:

$$t_k = \frac{\rho\left(\overline{W} - W(\pi^k)\right)}{\|\nabla W(\pi^k)\|^2} \tag{25}$$

$$\pi^{k+1} = \max\{\pi^k + t_k \nabla W(\pi^k), 0\} \tag{26}$$

In relation (25), ρ is commonly set in the range $[0, 2]$, while $\overline{W}$ denotes an upper bound of the optimal dual objective value. $\overline{W}$ can be easily found because of the Lagrangian Duality theory, which states that any feasible solution to the primal problem provides such an upper bound. In relation (26), the max operator assures the nonnegativity of the dual multipliers.

8.2. A flow-based approximation heuristic

This heuristic is composed of two parts. The first one corresponds to finding the exact solution of a flow-based approximation of the exact model. The output of this problem is used by an iterative algorithm to find a solution to the design problem. The goal of the flow-based approximation model (FBAM) is to reduce the number of the variables of the mathematical model presented in Section 7. For the formulation of FBAM, we define a bipartite graph $G_c = (N_1, N_2, A)$, where N_1 and N_2 have one node for each AP, while each arc $(j, h) \in A$ corresponds to a pair of APs which is a candidate for some STAs. Specifically, j represents the first AP in the candidate pair (j, h), while h is the second AP. To avoid repetitions, the considered pairs, and the corresponding arcs, are such that in each pair (j, h) it is $j < h$. As shown in Figure 4, it is $|N_1| = |N_2|$. However, not all the arcs from N_1 to N_2 are present in G_c, since they must correspond to candidate pairs for some STAs, and due to the assumption $j < h$ for each arc (j, h) going from N_1 to N_2. For example, the arc $(n1, n4)$ is not present in Figure 4 since we are assuming, in the corresponding example, that the APs pair $(1, 4)$ is not a candidate for any STA.

Algorithm 1 Lagrangian Heuristic

```
Setting $\pi^0 \sim U[0,1]$, $k_{max}$, $k = 0$, $F_b = \infty$;
Setting $\rho = 2$, $\rho_{min} = 0.005$, $\rho_{iter} = 0$, $\rho_{maxiter} = 5$;
while $k < k_{max}$ do
    $k = k + 1$ and $\rho_{iter} = \rho_{iter} + 1$;
    Given $\pi^k$ find the solutions $\mathbf{x}_k$ and $\mathbf{y}_k^*$ of the subproblems $\min_{\mathbf{x}} L_x(\pi^k)$ and $\min_{\mathbf{y}} L_y(\pi^k)$;
    Given $\mathbf{x}_k$, compute $\mathbf{y}_k$ that satisfies the constraints (14)
    Using $\mathbf{x}_k$ and $\mathbf{y}_k$ compute the primal objective $F$
    if $F < F_b$ then
        $F_b = F$, $x^{bt} = x^k$, $y^{bt} = y^k$ and $\bar{W} = F_b$
    end if
    if $\rho_{iter} > \rho_{maxiter}$ then
        $\rho = \max\{\rho/2, \rho_{min}\}$ and $\rho_{iter} = 0$
    end if
    Compute $\nabla W(\pi)$ by relations (24) using $\mathbf{x}_k$ and $\mathbf{y}_k^*$
    Compute $W(\pi)$ by relation (16) using $\mathbf{x}_k$ and $\mathbf{y}_k^*$
    Compute $t_k$ using (25)
    Compute $\pi^{k+1}$ using (26)
end while
```

Now, extend G_c by adding a super-source s, which is connected to each node j in N_1 via an arc (s, j), and a super-destination t, which is linked to each node h in N_2 via an arc (h, t). This extension, too, is shown in Figure 4. Let N denote the extended set of nodes while, with a little abuse of the notation, hereafter A will denote the extended set of arcs. Moreover, for each i, let A_i denote the subset of A corresponding to candidate pairs for the STA i.

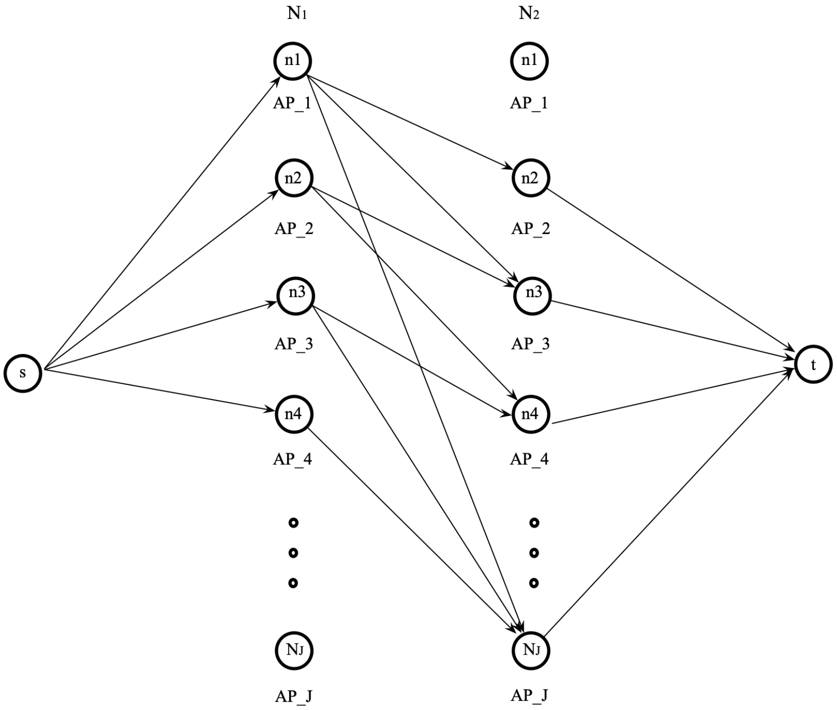


Figure 4: FBAM: the bipartite graph $G_c = (N_1, N_2, A)$.

Introduce a nonnegative and integer flow variable f_{uv} for each arc (u, v) of the extended network, and consider the y_j variables introduced in Section 7.

Then define the following formulation, called FBAM, on G_c, where $BS(u) = \{(v, u) \in A\}$ denotes the backward star of node u, i.e., the subset of the arcs of G_c entering u, while $FS(u) = \{(u, v) \in A\}$ denotes the forward star of node u, i.e., the subset of the arcs of G_c leaving u:

$$z = \min \sum_{j \in \mathcal{J}} y_j \tag{27}$$

$$\sum_{(v,u) \in BS(u)} f_{vu} - \sum_{(u,v) \in FS(u)} f_{uv} = \tag{28}$$

$$= \begin{cases} -|\mathcal{I}| & \text{if } u = s \\ +|\mathcal{I}| & \text{if } u = t \\ 0 & \text{otherwise} \end{cases} \qquad u \in N \tag{29}$$

$$\sum_{(j,h) \in A_i} f_{jh} \geq 1 \qquad i \in \mathcal{I} \tag{30}$$

$$f_{sj} + f_{jt} \leq N_S y_j \qquad j \in \mathcal{J} \tag{31}$$

Within the model, f_{jh} gives the number of times the candidate pair of APs (j, h) is selected. Constraints (29) impose the selection of candidate pairs of APs for a total of $|\mathcal{I}|$ selections (a candidate pair can be selected for more STAs), while constraints (30) impose the selection of at least one candidate pair in each subset A_i, for each STA i. On the other hand, constraints (31) indicate that, if a candidate pair involving an AP j is selected, then j must be powered on. In addition, they guarantee that the number of candidate pairs involving j, and so the number of STAs assigned to j, does not exceed N_S.

Notice that model FBAM has $3|\mathcal{J}|+O(|\mathcal{J}|^2)$ variables and $(|\mathcal{I}|+3|\mathcal{J}|+2)$ constraints, and therefore it has fewer variables and constraints than the exact model in Section 7. However, here the assignment of the STAs to the candidate pairs is not explicit. Moreover, although $|\mathcal{I}|$ selections of candidate pairs of APs are provided, there is no guarantee to have one candidate pair for each STA in the selected pool, due to the relaxed constraints (30). This is why FBAM only approximates the addressed design problem. In particular, its optimal solution value provides a lower bound for the exact model in Section 7.

8.2.1. Algorithm for the Assignment STA-APs Pair (ASAP)

The solution of model FBAM returns the subset of APs to power on, and the number of times each candidate pair of APs (j, h) is selected, i.e., f_{jh}. Starting from this output, we propose a heuristic algorithm to assign a candidate pair to each STA. The algorithm is organized in four steps, which are described in Algorithm 2 (steps 1 and 2), 4 (step 3) and 6 (step 4). Table 5 lists the main symbols used in the description.

Symbol	**Definition**
$\mathcal{C}^*$	Set of APs pairs with $f^* \geq 1$
$\mathcal{J}^*_{ON}$	Set of APs powered on
$\mathcal{J}^*_{OFF}$	Set of APs powered off
$\mathcal{C}_{Ci}$	Set of candidate APs pairs for STA i
$\mathcal{I}_{Nap}$	Set of STAs without an APs pair
Nfr_j	Number of free SPs in AP j
J^*_{ONwr}	Subset of $\mathcal{J}^*_{ON}$ with free SPs
c_i^1	Unique APs pair for STA i
AAP	Associated APs pair vector: the element i of AAP indicates the APs pair associated with STA i
J_{TT}	Set of APs to test for powering on a new AP

Table 5: List of symbols used in the description of ASAP.

The first two steps are illustrated in Algorithm 2. In particular, in Step 1 (lines 3-11), the algorithm considers each STA i for which a unique candidate pair (j, h), also referred to as c_i^1, exists in $\mathcal{C}^*$ (lines 5-6), and assigns it to i using the function `Associate_flow` described in Algorithm 3, after controlling if that is possible according to the solution of FBAM (line 7). The function performs the assignment (line 2) and then updates $f^*_{c_i^1}$ and the sets $\mathcal{I}_{Nap}$ and $\mathcal{C}^*$. The update of $\mathcal{C}^*$ is necessary only if $f^*_{c_i^1}$ is zero after its decrement (lines 5-7).

In Step 2 (lines 12-24), the algorithm cyclically performs the following actions until $|\mathcal{I}|$ loops are performed or $\mathcal{C}^* == \emptyset$. In each cycle, we consider an APs pair, say c, of $\mathcal{C}^*$. Then we evaluate the set of the STAs, without an assigned APs pair, which could be assigned to c (line 15). Specifically, we assign c to the STAs in $\mathcal{I}_c$ until we have enough flow resources, i.e., f^*_c is positive (loop of lines 17-20). More in detail, within the inner `WHILE` loop there is the assignment of the APs pair to the STA using the function `Associate_flow` (line 18). At the end of Step 2, some STAs may still be without an assigned APs pair, and some f^*_{jh} can still be positive since no STA can be assigned to these pairs.

The goal of Step 3, presented in Algorithm 4, is to generate new APs pairs

Algorithm 2 ASAP Algorithm - Steps 1 and 2

```
Input $\mathcal{C}^* = \{(j,h) \in \mathcal{C} : f^*_{jh} \geq 1\}$, $\mathcal{I}$, $\mathcal{J}$, $\mathcal{J}^*_{on} = \{j \in \mathcal{J} : y_j == 1\}$
$\mathcal{I}_{Nap} = \mathcal{I}$
for $i \in \mathcal{I}_{Nap}$ do
    $\mathcal{C}_{Ci} = \{(j,h) \in \mathcal{C}^* : \psi(i,(j,h)) \neq 0\}$
    if $|\mathcal{C}_{Ci} \cap \mathcal{C}^*| == 1$ then
        find $c_i^1 \in \mathcal{C}^*$ {Find the unique APs pair for STA $i$}
        if $f^*_{c_i^1} > 0$ then
            Associate_flow $(i, c_i^1, f^*_{c_i^1}, AAP, \mathcal{C}^*, I_{Nap})$
        end if
    end if
end for
$N_c = 1$
while $N_c \leq |\mathcal{I}| \wedge \mathcal{C}^* \neq \emptyset$ do
    for $c \in C^*$ do
        $I_c = \{i \in \mathcal{I}_{Nap}; \psi(i,c) \neq 0\}$
        if $|I_c| \geq 1$ then
            while $|I_c| \leq f^*_c \wedge |I_c| > 0$ do
                Associate_flow $(i, c, f^*_c, AAP, \mathcal{C}^*, I_{Nap})$
                $I_c = I_c \setminus \{i\}$
            end while
        end if
    end for
    $N_c + +$
end while
```

Algorithm 3 Function `Associate_flow` used in Algorithm 2

```
Function Associate_flow $(i, c, f^*_c, AAP, \mathcal{C}^*, I_{Nap})$
$AAP(i) = c$
$f^*_c - -$
$\mathcal{I}_{Nap} = \mathcal{I}_{Nap} \setminus \{i\}$
if $f^*_c == 0$ then
    $\mathcal{C}^* = \mathcal{C}^* \setminus \{c\}$
end if
return  $f^*_c, AAP, \mathcal{C}^*, I_{Nap}$
```

that can be used by the STAs belonging to the set $\mathcal{I}_{Nap}$, by exploiting $\mathcal{J}^*_{ON}$. In particular, we define the set $\mathcal{J}^*_{ONwr}$, which is composed of the powered on APs having some free SPs (line 1). Then, the `FOR` loop (lines 2-13) tries to generate new APs pairs to be used by the STAs in $\mathcal{I}_{Nap}$. The approach consists of analysing a single element of $\mathcal{J}^*_{ONwr}$ at a time, in order to devise the set of the APs that may generate new APs pairs, i.e., J_{TT} (line 3). For each element of J_{TT}, the outer `WHILE` loop (lines 4-12) defines the set of the STAs that may use the new APs pairs (line 7). Finally, the inner `WHILE` loop (lines 8-11) assigns the new APs pair to an element of I_{Ass}, after verifying that both APs of the considered pair have free SPs. Within this loop, the function `Associate_AP`$_{rsc}$ shown in Algorithm 5 performs the assignment and various control and update actions, similar to those performed by `Associate_flow`. Notice that no additional APs are powered on until this step.

Algorithm 4 ASAP Algorithm - Step 3

```
1: $J^*_{ONwr} = \{j \in J^*_{ON} : Nfr_j > 0\}$
2: for $h \in J^*_{ONwr}$ do
3:     $J_{TT} = J^*_{ONwr} \setminus \cup\{j \in J^*_{ONwr} : j > h\}$
4:     while $Nfr_h > 0 \wedge J_{TT} \neq \emptyset$ do
5:         Get $k \in J_{TT}$
6:         $J_{TT} = J_{TT} \setminus \{k\}$
7:         $I_{Ass} = \{i \in \mathcal{I}_{Nap} : \psi(i,(h,k)) > 0\}$
8:         while $|I_{Ass}| \geq 1 \wedge Nfr_h > 0 \wedge Nfr_k > 0$ do
9:             get $i \in I_{Ass}$
10:            Associate_AP$_{rsc}$ $(i, h, k, APP, \mathcal{I}_{Nap}, I_{Ass}, J^*_{ONwr}, Nfr_h, Nfr_k)$
11:        end while
12:    end while
13: end for
```

Finally Step 4, presented in Algorithm 6, tries to power on additional APs in order to generate other candidate pairs for $\mathcal{I}_{Nap}$. In deciding what APs to power on, the strategy is to give higher priority to those APs which, together with an AP belonging to $\mathcal{J}^*_{ONwr}$, may be used by the largest number of STAs in $\mathcal{I}_{Nap}$. To implement this strategy, first, we calculate I_k, defined as the set of the STAs that may be assigned to the APs pairs composed of the AP k and of any element of $\mathcal{J}^*_{ONwr}$, by considering the imposed form of the APs pair composition, i.e., (h, k) with $h < k$. (lines 3-5). Then, we sort the powered off APs in a descending order with respect to $|I_k|$ (line 7). The

Algorithm 5 Function Associate_AP$_{rsc}$ used in Algorithm 4 and 6

```
Function **Associate_AP**$_{rsc}$ $(i, h, k, APP, \mathcal{I}_{Nap}, I_{Ass}, J^*_{ONwr}, Nfr_h, Nfr_k)$
$AAP(i) = (h, k)$
$\mathcal{I}_{Nap} = \mathcal{I}_{Nap} \setminus \{i\}$
$I_{Ass} = I_{Ass} \setminus \{i\}$
$J^*_{ONwr} = J^*_{ONwr} \cup \{k\}$
$Nfr_h - -$
**if** $Nfr_h == 0$ **then**
    $J^*_{ONwr} = J^*_{ONwr} \setminus \{h\}$
**end if**
$Nfr_k - -$
**if** $Nfr_k == 0$ **then**
    $J^*_{ONwr} = J^*_{ONwr} \setminus \{k\}$
**end if**
**return** $APP, \mathcal{I}_{Nap}, I_{Ass}, J^*_{ONwr}, Nfr_h, Nfr_k$
```

FOR loop (lines 8-20) controls the association of the newly generated APs pairs to the STAs in $\mathcal{I}_{Nap}$, similarly to the outer WHILE loop of Algorithm 4. The control on $\mathcal{I}_{Nap}$ is used to exit from the loop when all the STAs have an assigned APs pair (line 9).

9. SP slot Assignment to STA

The problem of assigning two APs to each STA involves strategic decisions, being a network design problem. On the contrary, the problem of assigning a specific SP Slot (SPS) to each STA, in each of the two associated APs, is more operational and it can be addressed in a subsequent phase, i.e., after the solution of the mathematical model (12)-(14).

To show the feasibility of this approach we consider here, as a reference study, the simpler case where the SPS assignment can be performed without additional constraints, e.g., on the exact position in the BTI or on the relative position of the SPSs allocated to two or more STAs. Algorithm 7 presents a simple approach which, starting from the output of one of the proposed methods, i.e., the mathematical model (12)-(14) or one of the heuristics based on this model, solves the SPS assignment problem. This algorithm assigns the N_{SP} SPSs available in each powered on AP, which are ordered from 1 to N_{SP}, to the STAs. Precisely, the same SPS is assigned to each STA, in both

Algorithm 6 ASAP Algorithm - Step 4

```
$\mathcal{J}_{OFF} = \mathcal{J} \setminus \mathcal{J}^*_{on}$
for $k \in \mathcal{J}_{\mathcal{OFF}}$ do
    for $h \in J^*_{ONwr} \wedge h < k$ do
        $I_k = I_k \cup \{i \in I_{Nap} : \psi(i, (h, k)) > 0\}$
    end for
end for
$\mathcal{J}_{OFFsort}$ =sort$(\mathcal{J}_{OFF}, |I_k|, descend)$
for $k \in J_{OFFsort}$ do
    if $\mathcal{I}_{Nap} \neq \emptyset$ then
        for $h \in J^*_{ONwr} \wedge h < k$ do
            $I_{Ass} = \{i \in \mathcal{I}_{Nap} : \psi(i, (h, k)) > 0\}$
            while $|I_{Ass}| \geq 1 \wedge Nfr_h > 0 \wedge Nfr_k > 0$ do
                get $i \in I_{Ass}$
                Associate_AP$_{rsc}$ $(i, h, k, APP, \mathcal{I}_{Nap}, I_{Ass}, J^*_{ONwr}, Nfr_h, Nfr_k)$
            end while
        end for
    else
        break
    end if
end for
```

the APs composing the pair associated with the STA. Such a common SPS is the maximum free SPS in the two APs.

Algorithm 7 SPS Assignment Algorithm

```
Input $\mathcal{I}$, $\mathcal{J}$, APs pair associated with each STA
for $i \in \mathcal{I}$ do
    Obtain $AP_1$ and $AP_2$, i.e., the two APs assigned to STA i
    if {free SPS in $AP_1$} $\neq \emptyset \wedge$ {free SPS in $AP_2$} $\neq \emptyset$ then
        $S^1_{AP_1}$ is the first free SPS in $AP_1$
        $S^1_{AP_2}$ is the first free SPS in $AP_2$
        $SPS(i) = max\{S^1_{AP_1}, S^1_{AP_2}\}$
    else
        print ‘‘Solution not found!’’
    end if
end for
```

More sophisticated SPS assignment algorithms are out of the scope of the current paper, which is network design oriented, and they will be addressed in future works. Furthermore, it is worth noting that, in case of additional constraints on the SPS assignment, the current resolution approach, consisting in devising an SPS assignment starting from the solution found by the proposed model (12)-(14), might have limited efficacy. In such cases, formulating a “global” problem that incorporates problem (12)-(14) and the SPS assignment problem, by jointly addressing the related decisions, might be a better approach. Also, this type of study is not covered by the current paper, as it depends on the constraints on the SPS assignment which are given by the specific application addressed.

10. Numerical Evaluation

The simulation analysis has been carried out using MATLAB 2020a, with the integration of the solver Gurobi Optimizer ver.9.1.2. All the simulations ran on a MacBook Pro with 2.8 GHz Quad-Core Intel i7 and 16 GB of 2133MHz LPDDR3 RAM.

The simulation scenario is a 50×50 m grid, where the candidate APs are located at points (x, y) with x and y in the set $\{0, 5, 10, \ldots 50\}$. Therefore, in the grid there are $|\mathcal{J}| = 121$ candidate locations. We divided the test field into a set of squares, each square having an AP in the centre and equal side of 5 m. The set of the STAs is split into subsets of equal size, i.e., $\lfloor|\mathcal{I}|/|\mathcal{J}|\rfloor$. Using a bidimensional uniform distribution, the STAs of each subset are then randomly spread over each square area related to an AP. The remaining STAs are randomly spread over the whole test field. This strategy ensures enough uniformity in the placement of the STAs to mimic an industrial environment. The data rate of each link STA-AP has been estimated by using the approach presented in Section 4. The transmission power has been set to 10 dBm, while the antenna gain is $G_t \cdot G_r = 12$ dB. For calculating the elements $\psi_{i,c}$, we set the threshold of the link data rate and of the angle between two links related to the same STA equal to $T_d = 1$ Gbps and $T_a = \pi/2$, respectively.

In our analysis, we considered two values of N_S, i.e., $N_S = N_{SP} = 44$ and $N_S = N_{SP}/2 = 22$, where N_{SP} is given by relation (11) in Section 6. The study considers 5 different configurations, which differ for the number of the STAs, i.e., $|\mathcal{I}| = 100, 200, 300, 400, 500$. For each configuration, ten network scenarios have been generated. They differ regarding the location of the STAs, which is randomly generated for each scenario, whereas the location of the APs is always in the set of the (x, y) points as defined above. Each network scenario has been solved by means of the approaches presented in this paper. It is important to note that the study for different values of parameter N_S considers, for each scenario, the same locations for the APs and the STAs.

The compared algorithms will be referred to by using the following acronyms: OPT is the exact model presented in Section 7, LAG is the Lagrangian heuristic presented in Section 8.1, while FBH is the flow-based heuristic described in Section 8.2.

The considered performance parameters are:

- the Computation Time (CT), i.e., the time necessary to determine a solution,
- the Number of Powered On APs (NPO).

For the solution of models OPT and FBAM, which is used in FBH, the parameters `TimeLimit` and `MIPgap` of the solver have been set equal to 1000 s and 0.0001, respectively. `TimeLimit` is the maximum time given to the solver for outputting a solution. When the solver achieves `TimeLimit`, it outputs the best solution found. In this case, the returned solution could be not optimal. The `MIPgap` value refers to the gap value that the solver has to reach before declaring optimality. In the case of LAG, the number of the iterations, i.e., K, has been set to 200. Several preliminary tests with $K = 2000$ have shown that, using this larger value, only in a few cases LAG returns a lower objective function value. In general, the gain is of 1 or 2 additional powered off APs with respect to the solution given with $K = 200$. Furthermore, in about the 50% of the analysed random scenarios, the minimum was found at an iteration $k < 200$. The setting $K = 200$ thus represents a good trade-off between CT and the number of powered on APs.

10.1. CT Analysis

Figures 5 and 6 show the average CTs and the related 95% Confidence Interval (CI) for the considered configurations, related to $N_S = 22$ and $N_S = 44$, respectively. In Figure 5, i.e., considering $N_S = 22$, the curves clearly point out the superiority of FBH and OPT in terms of CT. Furthermore, given the lower time complexity of model FBAM, used by the heuristic FBH, with respect to the one of model OPT, the observed CTs of FBH are lower than those required to solve model OPT.

In Figure 6, i.e., considering $N_S = 44$, we observe two different behaviours. For $|\mathcal{I}| > 200$, OPT can be solved in tens of seconds (as in the case of $N_S = 22$), whereas hundreds of seconds are necessary for the configurations where $|\mathcal{I}| = 100, 200$. Also, FBH shows the worst CT performance for the configurations $|\mathcal{I}| = 100, 200$. On the other hand, in the case of LAG, CT varies in a regular way depending on $|\mathcal{I}|$. This is related to the fact that, in a single iteration, the calculation of x_k requires the solution of subproblem (21)-(22), having a time complexity depending on $|\mathcal{I}|$.

Except for $|\mathcal{I}| = 100$, FBH shows the lowest CT. A deep analysis of the results indicates that the higher average CT value for $|\mathcal{I}| = 100$ is due to the fact that the solver outputs the solution only after achieving the set `TimeLimit`, i.e., 1000 s. Precisely, this behaviour has been observed in the random scenarios 1, 7, 8 and 9 of this configuration.

Recall that, as described in Section 8.2.1, in FBH the assignment STA-APs pair is done using the algorithm ASAP. We report here that the time to perform ASAP was always less than the 50% of the whole CT, and in some cases it was negligible.

To complete the analysis of ASAP, we present some results related to the most critical case, i.e., $N_S = 44$. Indeed, in some scenarios, ASAP did not determine an APs pair for some STAs. By considering the ten random scenarios of each configuration, Table 6 reports the number of the scenarios

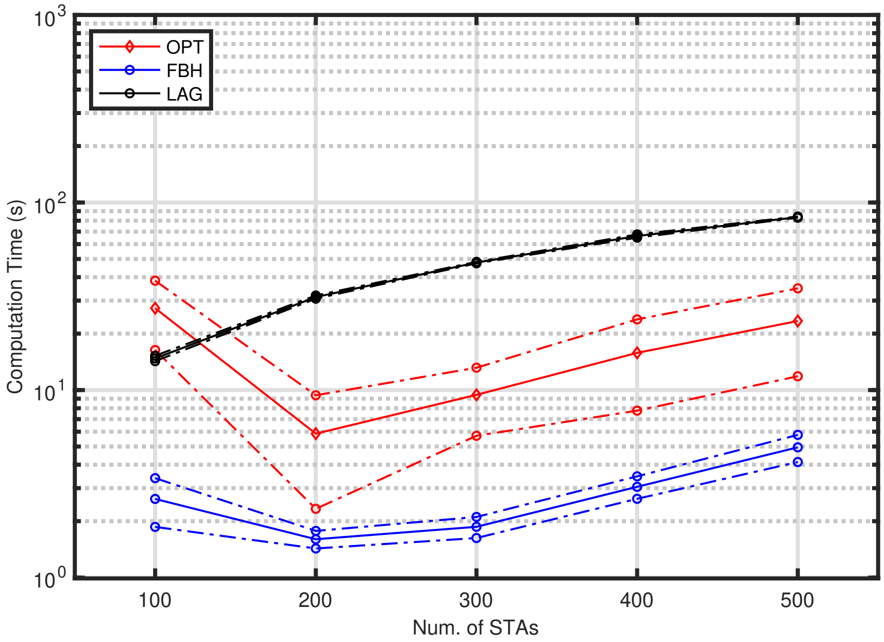


Figure 5: $|\mathcal{I}|$ vs. CT - $N_S = 22$

in which some STAs do not have an assigned APs pair, along with the highest and the average number of STAs without an APs pair. The statistics are not reported for the configurations $|\mathcal{I}| = 100, 200$, since in such configurations an APs pair has been assigned to each STA in all the scenarios.

Table 6: FBH: Statistics on STAs Without APs pair (SWAP) - Case $N_S = 44$.

Parameter	300	400	500
Number of Scenarios with STAs without APs pair	4	5	4
Maximum Number of STAs without APs pair	11	6	8
Average Number of STAs without APs pair	2.2	1.9	2.0

Table 7 presents a set of results, always related to FBH, aimed at showing the enhancements given by the last steps of ASAP applied to the solution of model FBAM.

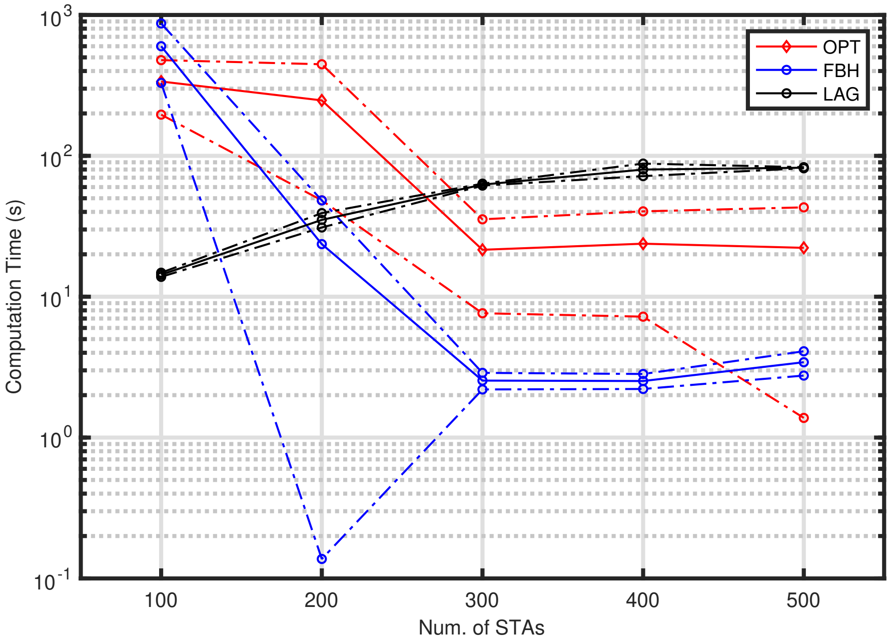


Figure 6: $|\mathcal{I}|$ vs. CT - Case $N_S = 44$.

Until the end of Step 3, ASAP uses only the APs which are powered on according to the solution of model FBAM. As emphasized in Section 8.2.1, these resources do not necessarily allow the assignment of an APs pair to each STA. For each configuration, the first three rows of the table show some statistics for the STAs Without an APs pair (SWAP) observed in the ten random scenarios at the end of Step 3. Such statistics are, respectively, the average and the maximum number of SWAP. Moreover, the third row shows the maximum percentage of SWAP. In all the configurations, the maximum number of SWAP is higher than 14% of $|\mathcal{I}|$, with some scenarios where the percentage of SWAP achieves 23% (e.g., $|\mathcal{I}| = 300$). The performance improvements given by Step 4 are highlighted by the fifth row of the table, reporting the maximum number of SWAP at the end of this step. This row shows that the execution of Step 4 completely solves the SWAP problem for $|\mathcal{I}| = 100, 200$, by powering on additional APs. For these two configurations, the sixth row of the table indicates that the maximum number of APs powered on by Step 4 are 2 and 4 for $|\mathcal{I}| = 100$ and 200, respectively.

For the other configurations, the fourth row of the table shows the high

Table 7: FBH: Improvements given by Steps 3 and 4 of ASAP - Case $N_S = 44$.

Parameter	100	200	300	400	500
Average Num. of SWAP - Step 3	3	19	45.20	42.90	66.00
Max. Num. of SWAP - Step 3	14	34	69	74	80
Max. Num. of SWAP (%)- Step 3	14.00	17.00	23.00	18.50	16.00
Average Num. of SWAP Solved by Step 4	3	19	43	41	64
Max. Num. of SWAP - Step 4	0	0	3	4	1
Max. Num. of AP powered on by Step 4	2	4	12	11	11

number of STAs for which Step 4 finds an APs pair on average. However, as shown in the fifth row of the table, at the end of Step 4 there are still a few SWAP (at most 4 for $|\mathcal{I}| = 400$). The cost of this enhancement is the number of additional powered on APs, which is 11 or 12 at the most.

By comparing the CT of OPT and FBH, the reported computational results do not justify the design and the use of the proposed heuristic in place of solving the exact model, because the CTs are order of tens of seconds except in the scenarios with $|\mathcal{I}| = 100, 200$ for $N_S = 44$.

However, in Section 7 we pointed out that the exact model has $|\mathcal{J}| + O(|\mathcal{I}||\mathcal{J}|^2)$ variables and $(|\mathcal{I}| + |\mathcal{J}|)$ constraints. Therefore, a large amount of memory and time could be necessary to load the problem description to the solver. In particular, for the considered configurations, Figure 7 shows the memory used by the implementations of models OPT and FBAM (recall that the solution of FBAM is required by the heuristic FBH). The figure shows the average and the 95% CI calculated over the ten random network scenarios of each configuration. The used memory depends only on the geometry of the problem, i.e., on $|\mathcal{J}|$, $|\mathcal{I}|$, and on the manner the STAs are randomly distributed in the considered area. Such a distribution has an impact on the number of the zero elements of matrix Ψ, which can be used to save a lot of memory. In fact, MATLAB allows one to define a `sparse` matrix, which is obtained by converting a full matrix into a sparse form by squeezing out all the zero elements. If a matrix contains many zeros, by converting the matrix to sparse thus saves memory. In the figure, the 95% CI are almost indistinguishable from the average curves. Such results outline a difference of more than one order of magnitude between the two models in terms of used memory, by thus supporting the study and the practical use of the heuristic FBH.

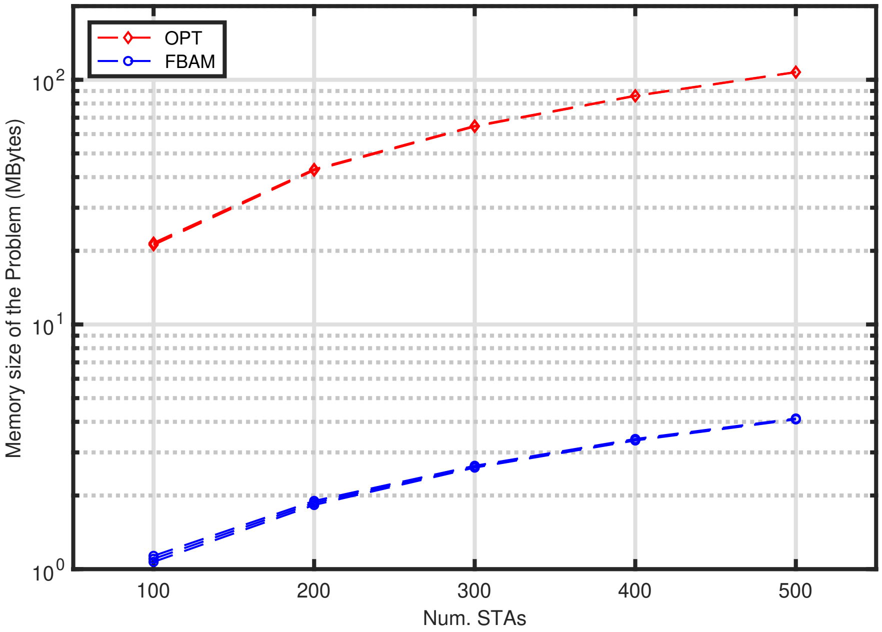


Figure 7: Memory used by the model implementation vs. $|\mathcal{I}|$: Average and 95% CI.

10.2. NPO Analysis

Tables 8 and 9 report the average and the 95% CI of NPO for $NS = 44$ and $NS = 22$, respectively. By analysing Table 8, the results related to the case $NS = 44$ point out that, for $|\mathcal{I}| = 100, 200$, the heuristic LAG powers on the highest number of APs. On the contrary, for $|\mathcal{I}| > 200$, FBH provides the worst performance. By taking into account that, in some of these scenarios, FBH is not able to determine an APs pair for all the STAs, as shown in Table 6, we can conclude that, although the CT of FBH is an order of magnitude lower than the one of LAG, LAG is preferable to FBH for the configurations where $|\mathcal{I}| > 200$. On the other hand, for $|\mathcal{I}| = 100$, in most of the random scenarios FBH outperforms LAG both in terms of CT and NPO, whereas for $|\mathcal{I}| = 200$ the two heuristics provide similar performance.

Table 8: Number of powered on APs: average value and 95% CI vs. $|\mathcal{I}|$ - Case $N_S = 44$.

$\lvert\mathcal{I}\rvert$	OPT	FBH	LAG
100	8.00 ± 0.00	8.80 ± 0.56	12.00 ± 0.67
200	10.10 ± 0.23	13.00 ± 0.67	15.20 ± 0.56
300	14.00 ± 0.00	20.00 ± 2.08	19.20 ± 0.56
400	19.00 ± 0.00	25.40 ± 1.88	23.60 ± 0.50
500	23.00 ± 0.00	31.40 ± 1.40	27.60 ± 0.50

Table 8 also shows that model OPT outputs NPO values with null 95% CI, except for $|\mathcal{I}| = 200$. Indeed, for all the random scenarios related to the same configuration, we observed the same value of NPO. Only for $|\mathcal{I}| = 200$ we observed one random scenario, i.e., Scenario 7, which was not solved within the set `TimeLimit`. For Scenario 7, in particular, 11 APs have been powered on instead of the 10 APs which are powered on in the other scenarios of this configuration. We emphasize that, for the scenarios solved within the `TimeLimit`, OPT returns the minimum number of APs to open, since it is an exact formulation of the optimization problem addressed. Clearly, such optimal values are usually obtained at the expense of a large time and memory consumption, as previously reported.

Table 9: Number of powered on APs: average value and 95% CI vs. $|\mathcal{I}|$ - Case $N_S = 22$.

$\lvert\mathcal{I}\rvert$	OPT	FBH	LAG
100	10.00 ± 0.00	14.30 ± 1.35	13.60 ± 0.37
200	19.00 ± 0.00	24.00 ± 1.39	22.10 ± 0.71
300	28.00 ± 0.00	33.50 ± 1.79	31.70 ± 0.48
400	37.00 ± 0.00	42.70 ± 0.89	40.80 ± 0.66
500	46.00 ± 0.00	51.50 ± 1.44	50.00 ± 0.48

As far as $NS = 22$ is concerned, Table 9 clearly highlights the improved performance of LAG with respect to FBH. Indeed, in this case, LAG returns average values of NPO (as well as values of CI) lower than the ones returned by FBH in all the configurations.

10.3. Load distribution analysis

In order to show how the N_S SPs in an AP are used, we define a *saturated AP* as an AP such that the number of the STAs assigned to it is equal to N_S. In this case, no preallocated SPs are available in the AP. Figure 8 shows the average number of saturated APs, in percentage, for the different configurations in the case $N_S = 44$. The percentage is calculated over the number of the powered on APs averaged over the scenarios of the considered configuration. The figure shows a similar behaviour for OPT and LAG, although LAG has lower values. For $|\mathcal{I}| \geq 300$, more than 70% of used APs are saturated. In particular, the optimal solutions found by OPT concentrate the STAs in the few powered on APs. On the contrary, the curve related to FBH indicates that ASAP tends to distribute the STAs among the activated APs in a balanced way, by maintaining the percentage of the saturated AP less than 25% also in the worst configuration. This behaviour can be due to the first two steps of ASAP, where the strategy for assigning the STAs to the APs pairs does not favour the concentration of the STAs in a few APs.

Figure 9, instead, shows the minimum number of saturated APs observed in the scenarios, always in the case $N_S = 44$. This figure confirms the different behaviour of FBH with respect to OPT and LAG. In particular, for the configurations with $|\mathcal{I}| < 500$, the figure highlights that FBH has at least one scenario with no saturated AP (i.e., 0%). On the contrary, except for $|\mathcal{I}| = 100$, OPT and LAG have always saturated APs in the generated scenarios. Furthermore, OPT shows minimum values higher than 50% of the number of powered on APs. In other words, all the scenarios with $|\mathcal{I}| > 200$ have more than 50% of powered on APs that are saturated.

The analysis of the case $N_S = 22$ confirms the behaviour presented in Figures 8 and 9. For the sake of brevity, we do not report the corresponding results.

10.4. Analysis of the SPS Assignment Algorithm

We evaluated the performance of Algorithm 7 by using the results returned by OPT, LAG and FBH. In all the cases, the proposed SPS assignment algorithm was able to assign to each STA the same SP slot in the two associated APs. Figure 10 shows the links between the STAs and the related APs pair by considering four different SPSs, in the case $N_{SP} = 44$ and

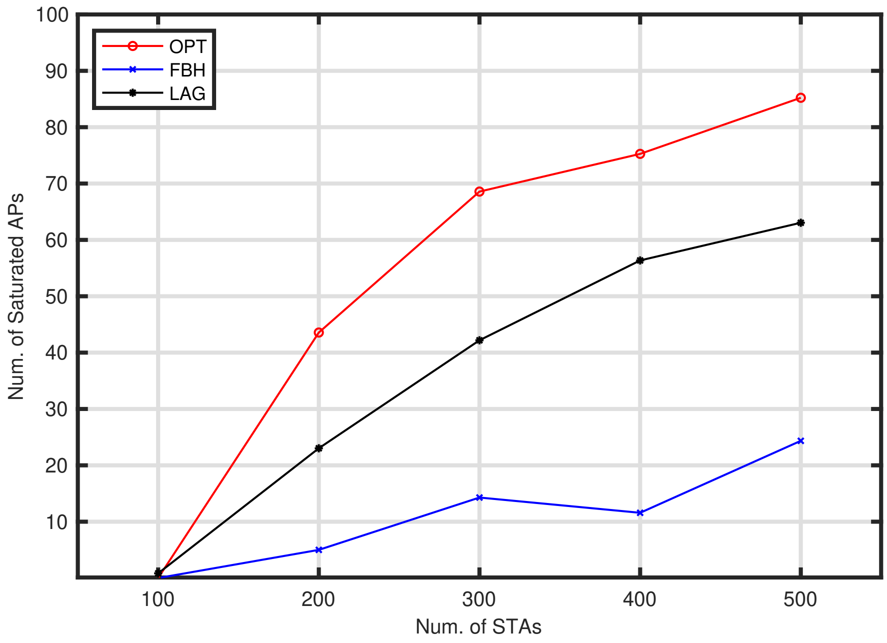


Figure 8: Average number of saturated APs vs. $|\mathcal{I}|$ - Case $N_S = 44$.

$|\mathcal{I}| = 100$. In the figure, the small black dots denote the random locations of the STAs, whereas the red points are the locations of the APs which have been powered on according to the solution found by model OPT. The big black points, in particular, emphasize those STAs which have been assigned to the considered SPS, i.e., $SPS = 1, 3, 9, 20$. Moreover, the red lines show the assignment between such STAs and the corresponding APs pairs. Specifically, in the four different subfigures, we show the communications which are permitted in the four considered SPSs, i.e., $SPS = 1, 3, 9, 20$. This set of SPSs has been selected since all the powered on APs are involved, and since it illustrates the diverse number of simultaneous communications that may arise in the area, i.e., 2 when considering SPS=20, 3 for SPS=1, and 4 for SPS=3 and SPS=9.

10.5. Algorithm comparison

The results previously presented have outlined some interesting features of the proposed approaches, which may allow one to suggest what is the more suitable algorithm to use depending on the characteristics of the instances

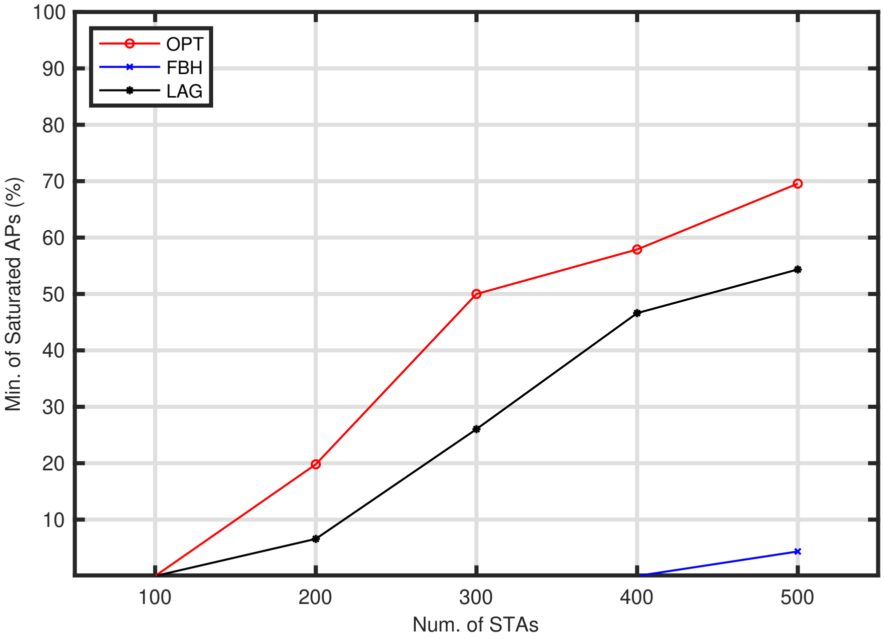


Figure 9: Minimum number of saturated APs vs. $|\mathcal{I}|$ - Case $N_S = 44$.

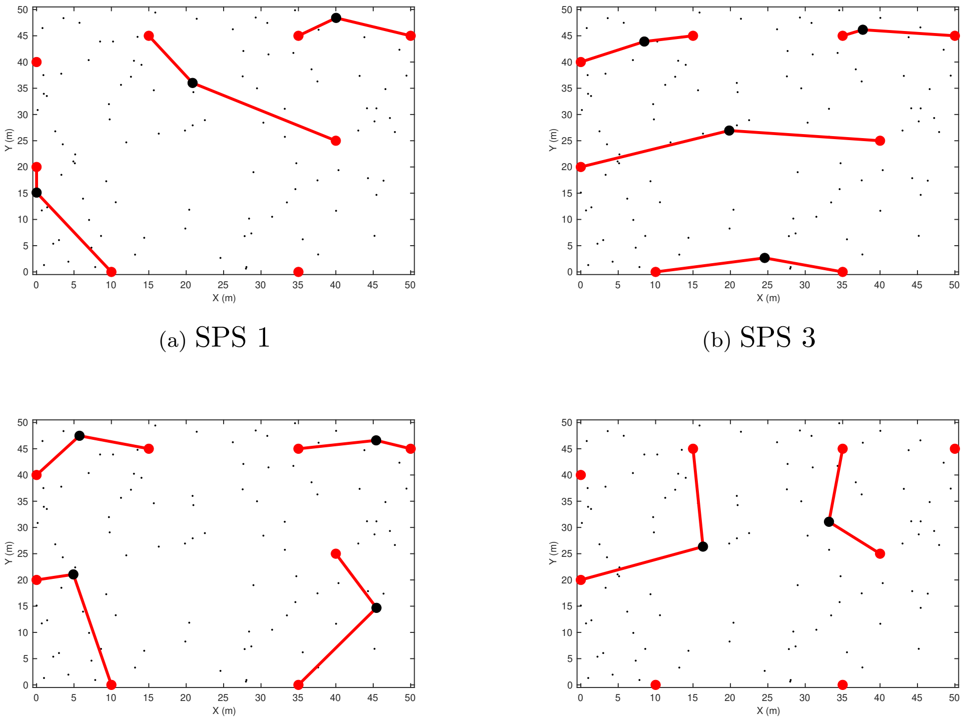
to solve.

First, the computation time (CT) of both OPT and FBH is a critical issue for high values of N_S and low values of $|\mathcal{I}|$. For example, for $N_S = 44$ and $|\mathcal{I}| = 100$, assigning the STAs to a few APs appear to be more complex than in the other cases. Hence, probably this is why the solver spends more time in finding a solution. Moreover, the high complexity of model OPT causes a high memory requirement, which affects the time needed to load the model description into the solver.

Regarding the CT of LAG, it shows an almost exponential dependence on $|\mathcal{I}|$. Indeed, we observed an almost linear curve of CT vs. $|\mathcal{I}|$ with log-linear axes.

By comparing the approaches, for $N_S = 44$ and $|\mathcal{I}| = 100, 200$, FBH outperforms LAG in terms of CT and NPO. In the other cases, instead, LAG takes more time to find a solution, but it guarantees a value of NPO which is lower than the one of FBH. That is, less APs are powered on by LAG.

Regarding the quality of the computed solutions, both OPT and LAG



(a) SPS 1

(b) SPS 3

(c) SPS 9

(d) SPS 20

Figure 10: Communications allowed in four different SPSs - The red dots are the APs powered on by OPT, the big black points and the red lines represent the STAs and the pair links used during the SPS, while the small black points represent the other STAs of the network scenario.

generate solutions with a high concentration of STAs in the selected APs. This characteristic leads to a high number of saturated APs. FBH, instead, does not always guarantee the assignment of an APs pair to each STA.

In summary, the proposed approaches present some complementary features, especially in terms of CT, NPO and number of saturated APs, also depending on the values of parameters N_S and $|\mathcal{I}|$. Furthermore, the presence of SWAP, i.e., STAs without an assigned APs pair, in the final output of FBH may represent a disadvantage of this method, although it shows an interesting performance in terms of CT and of number of saturated APs.

10.6. Model limitations

The proposed model, and the two heuristics which rely on it, have certain limitations essentially due to some of the assumptions made, the most notable being the assumption of static STAs. Consequently, the model is certainly helpful in designing the infrastructure which is necessary for the TSN support, but it needs to be periodically solved in case of mobile STAs, in order to take into account changes in their location. Due to the computation effort required for the model solution, such a period cannot be shorter than a few hundred seconds. Moreover, in order to mitigate possible issues related to the reconfiguration of the wired part, and to frequent changes in the state of the APs between successive computations (i.e., from power on to power off and vice-versa), some additional constraints should be included into the formulation. Also observe that, in the current form, the proposed model is not designed to quickly address issues related to AP failures. In fact, the model minimizes the number of the powered on APs, and this may cause a lack of alternative APs (or APs pairs) in the event of some AP failures. To overcome this limitation, the presented model could be used as a starting point to formulate robust models, such as the one in [44], which account for AP failures, errors in radio channel estimation (e.g., due to environmental dynamics), or both. Finally, observe that the proposed model could also be used as a benchmark for studying other approaches to the problem, such as dynamic programming or machine learning algorithms.

11. Conclusions

In this paper we have discussed the setting of the IEEE802.11ad MAC parameters which allow one to guarantee a low latency by exploiting the isochronous SPs allocation to BI of 1 ms. Then, we have formulated a design model aimed at minimizing the number of deployed APs for enhancing the system reliability under the FRER mechanism, while considering, for each AP, the constraints on the maximum number of SPs dedicated to the low latency traffic. Due to the complexity of the exact model, two heuristics have been presented. One, named LAG, is based on an iterative procedure involving the solution of a Lagrangian Dual problem. The other one, called FBH, is based on the combination of an approximated flow-based model, less complex to solve than the exact model, and of ASAP, i.e., an algorithm used to define the STA-APs pair association and reduce the SWAP problem, i.e., the issue of STAs without an assigned APs pair. The performance comparison among the presented approaches has shown complementary features of the proposed methods, and highlighted some key pros and cons of the proposed heuristics. It is worth noting that, in operational scenarios where constraints on the SPS assignment are present, it would be preferable to state and solve a global problem that incorporates the presented design problem (12)-(14), aimed at deciding the location of the powered-on APs and the STA-APs pair assignment, as well as the SPS assignment problem, jointly addressing the related decisions. Indeed, the approach presented in this paper relies on the splitting of such a global problem into two subproblems and their consecutive resolution. The reported simulation results show that, in a simple operational scenario where no constraints on the SPS assignment are given, the proposed splitting works well. The formulation and study of the global problem, for various operational scenarios, represent an interesting avenue of research that we plan to investigate in future works.

Acknowledgments

This work was partially supported by the Italian Ministry of Education and Research (MIUR) in the framework of the CrossLab project and FoReLab project (Departments of Excellence), and by the project PRA 2022-2023, Research Project "POIANA: Platform for Observations In Agricultural and eNvironmentAl fields" funded by the University of Pisa, Italy.

References

[1] Y. Liu, M. Kashef, K. B. Lee, L. Benmohamed, and R. Candell, "Wireless network design for emerging IIoT applications: Reference framework and use cases," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1166–1192, 2019.

[2] N. Finn, "Introduction to time-sensitive networking," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 22–28, 2018.

[3] IEEE, "IEEE Time-Sensitive Networking Task Group," https://1.ieee802.org/tsn/, (Accessed: 2023-01-15).

[4] M. Eisen, M. M. Rashid, A. Ribeiro, and D. Cavalcanti, "Scheduling low latency traffic for wireless control systems in 5G networks," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[5] A. Seferagić, J. Famaey, E. De Poorter, and J. Hoebeke, "Survey on wireless technology trade-offs for the Industrial Internet of Things," *Sensors*, vol. 20, no. 2, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/2/488

[6] D. Cavalcanti, J. Perez-Ramirez, M. M. Rashid, J. Fang, M. Galeev, and K. B. Stanton, "Extending accurate time distribution and timeliness capabilities over the air to enable future wireless industrial automation systems," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1132–1152, 2019.

[7] C. S. V. Gutiérrez, L. U. S. Juan, I. Z. Ugarte, and V. M. Vilches, "Time-sensitive networking for robotics," 2018. [Online]. Available: https://arxiv.org/abs/1804.07643

[8] I. Val, O. Seijo, R. Torrego, and A. Astarloa, "IEEE 802.1AS clock synchronization performance evaluation of an integrated wired–wireless TSN architecture," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 2986–2999, 2022.

[9] M. Gundall, C. Huber, and S. Melnyk, "Integration of IEEE 802.1AS-based time synchronization in IEEE 802.11 as an enabler for novel industrial use cases," *arXiv*, 2021. [Online]. Available: https://arxiv.org/abs/2101.02434

[10] A. M. Romanov, F. Gringoli, and A. Sikora, "A precise synchronization method for future wireless TSN networks," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3682–3692, 2021.

[11] T. Jeffree, "802.1Qbv - enhancements for scheduled traffic," 2016. [Online]. Available: https://www.ieee802.org/1/pages/802.1bv.html

[12] A. De la Oliva, X. Wang, R. Yang, and R. Gazda, "IEEE 802.11-19/1223r0: Improving WLAN reliability," 2019. [Online]. Available: https://mentor.ieee.org/802.11/dcn/19/11-19-1223-00-00be-improving-wlan-reliability-joint-tsn-11be-session.pdf

[13] T. Adame, M. Carrascosa-Zamacois, and B. Bellalta, "Time-Sensitive Networking in IEEE 802.11be: On the way to low-latency WiFi 7," *Sensors*, vol. 21, no. 15, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/15/4954

[14] D. Cavalcanti and G. Venkatesan, “802.1 TSN over 802.11 with updates from developments in 802.11be,” 2020. [Online]. Available: https://www.ieee802.org/1/files/public/docs2020/new-Cavalcanti-802-1TSN-over-802-11-1120-v02.pdf

[15] E. Park, D. Lim, J. Kim, and J. Choi, “IEEE 802.1-19/0779r1: Performance investigation on Multi-AP transmission,” 2019. [Online]. Available: https://mentor.ieee.org/802.11/dcn/19/11-19-0779-01-00be-performance-investigation-on-multi-ap-transmission.pptx

[16] A. Titus, R. Bansal, T. V. Sreejith, A. A. Kherani, and N. Akhtar, “Decision problems for joint transmission in Multi-AP coordination framework of IEEE 802.11be,” in *2021 International Conference on COMmunication Systems and NETworkS (COMSNETS)*, 2021, pp. 326–333.

[17] “IEEE standard for information technology–telecommunications and information exchange between systems–local and metropolitan area networks–specific requirements-part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications amendment 3: Enhancements for Very High Throughput in the 60 GHz Band,” *IEEE Std 802.11ad-2012 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012 and IEEE Std 802.11aa-2012)*, pp. 1–628, 2012.

[18] P. Zhou, K. Cheng, X. Han, X. Fang, Y. Fang, R. He, Y. Long, and Y. Liu, “IEEE 802.11ay-based mmWave WLANs: Design challenges and solutions,” *IEEE Communications Surveys and Tutorials*, vol. 20, no. 3, pp. 1654–1681, 2018.

[19] C. Cano, G. H. Sim, A. Asadi, and X. Vilajosana, “A channel measurement campaign for mmWave communication in industrial settings,” *IEEE Trans. Wireless. Comm.*, vol. 20, no. 1, p. 299–315, jan 2021. [Online]. Available: https://doi.org/10.1109/TWC.2020.3024709

[20] M. Mackay, A. Raschella, and O. Toma, “Modelling and analysis of performance characteristics in a 60 GHz 802.11ad wireless mesh backhaul network for an urban 5G deployment,” *Future Internet*, vol. 14, no. 2, 2022. [Online]. Available: https://www.mdpi.com/1999-5903/14/2/34

[21] J. Struye, F. Lemic, and J. Famaey, "Towards ultra-low-latency mmWave Wi-Fi for multi-user interactive virtual reality," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[22] C. Pielli, T. Ropitault, N. Golmie, and M. Zorzi, "An analytical model for CBAP allocations in IEEE 802.11ad," *IEEE Transactions on Communications*, vol. 69, no. 1, pp. 649–663, 2021.

[23] M. P. R. S. Kiran and P. Rajalakshmi, "Saturated throughput analysis of IEEE 802.11ad EDCA for high data rate 5G-IoT applications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4774–4785, 2019.

[24] S. Sudhakaran, K. Montgomery, M. Kashef, D. Cavalcanti, and R. Candell, "Wireless Time Sensitive Networking impact on an industrial collaborative robotic workcell," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 10, pp. 7351–7360, 2022.

[25] O. Seijo, J. A. López-Fernàndez, and I. Val, "w-SHARP: Implementation of a high-performance Wireless Time-Sensitive Network for low latency and ultra-low cycle time industrial applications," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3651–3662, 2021.

[26] O. Seijo, I. Val, M. Luvisotto, and Z. Pang, "Clock synchronization for Wireless Time-Sensitive Networking: A march from microsecond to nanosecond," *IEEE Industrial Electronics Magazine*, vol. 16, no. 2, pp. 35–43, 2022.

[27] A. Sahoo, W. Gao, T. Ropitault, and N. Golmie, "Admission control and scheduling of isochronous traffic in IEEE 802.11ad MAC," in *Proceedings of the 24th International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 125–134. [Online]. Available: https://doi.org/10.1145/3479239.3485698

[28] C. Hemanth and T. G. Venkatesh, "Performance analysis of Service Periods (SP) of the IEEE 802.11ad hybrid MAC protocol," *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1224–1236, 2016.

[29] M. Lecci, M. Drago, A. Zanella, and M. Zorzi, “Exploiting scheduled access features of mmWave WLANs for periodic traffic sources,” in *2021 19th Mediterranean Communication and Computer Networking Conference (MedComNet)*, 2021, pp. 1–8.

[30] T. Azzino, T. Ropitault, and M. Zorzi, “Scheduling the data transmission interval in IEEE 802.11ad: A reinforcement learning approach,” in *2020 International Conference on Computing, Networking and Communications (ICNC)*, 2020, pp. 602–607.

[31] P. F. Smulders, “Statistical characterization of 60-GHz indoor radio channels,” *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 10, pp. 2820–2829, 2009.

[32] C. Gustafson, *60 GHz Wireless Propagation Channels: Characterization, Modeling and Evaluation.* PhD Theses, Lund University, 2014.

[33] T. S. Rappaport, G. R. MacCartney, M. K. Samimi, and S. Sun, “Wideband millimeter-wave propagation measurements and channel models for future wireless communication system design,” *IEEE Transactions on Communications*, vol. 63, no. 9, pp. 3029–3056, Sep 2015. [Online]. Available: http://dx.doi.org/10.1109/TCOMM.2015.2434384

[34] *IEEE 802.11-09/0334r8, Channel Models for 60 GHz WLAN Systems.* IEEE P802.11, 2010.

[35] D. Solomitckii, A. Orsino, S. Andreev, Y. Koucheryavy, and M. Valkama, “Characterization of mmWave channel properties at 28 and 60 GHz in factory automation deployments,” in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 2018, pp. 1–6.

[36] M. Teubal, “Heavy and light industry in economic development,” *The American Economic Review*, vol. 63, no. 4, pp. 588–596, 1973. [Online]. Available: http://www.jstor.org/stable/1808850

[37] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, “Millimeter wave mobile communications for 5G cellular: It will work!” *IEEE Access*, vol. 1, pp. 335–349, 2013.

[38] *TR 22.804 V16.3.0: Study on Communication for Automation in Vertical Domains.* 3GPP, 2020.

[39] X. Lu, M. Lema, T. Mahmoodi, and M. Dohler, "Downlink data rate analysis of 5G-U (5G on Unlicensed Band): Coexistence for 3GPP 5G and IEEE802. 11ad WiGig," in *European Wireless 2017; 23th European Wireless Conference; Proceedings of.* VDE, 2017, pp. 1–6.

[40] S. Vitturi, C. Zunino, and T. Sauter, "Industrial communication systems and their future challenges: Next-Generation Ethernet, IIoT, and 5G," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 944–961, 2019.

[41] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)*, first edition ed. W. H. Freeman, 1979.

[42] Z. Drezner and H. W. Hamacher, *Facility Location: Applications and Theory*, second edition ed. Springer Science Business Media, 2004.

[43] L. Wolsey, *Integer Programming, second edition*, ser. Discrete Mathematics and Optimization. John Wiley & Sons, 2020.

[44] R. G. Garroppo, M. G. Scutellà, and F. D'Andreagiovanni, "Robust green wireless local area networks: A matheuristic approach," *Journal of Network and Computer Applications*, vol. 163, 2020.