INSPIRE: Distributed Bayesian Optimization for ImproviNg SPatIal REuse in Dense WLANs

Anthony Bardou
anthony.bardou@ens-lyon.fr
Univ Lyon, ENS de Lyon
Université Claude Bernard Lyon 1, CNRS, LIP
Lyon, France

Thomas Begin
thomas.begin@ens-lyon.fr
Univ Lyon, ENS de Lyon
Université Claude Bernard Lyon 1, CNRS, LIP
Lyon, France

ABSTRACT

WLANs, which have overtaken wired networks to become the primary means of connecting devices to the Internet, are prone to performance issues due to the scarcity of space in the radio spectrum. As a response, IEEE 802.11ax and subsequent amendments aim at increasing the spatial reuse of a radio channel by allowing the dynamic update of two key parameters in wireless transmission: the transmission power (TX_POWER) and the sensitivity threshold (OBSS_PD). In this paper, we present INSPIRE, a distributed online learning solution performing local Bayesian optimizations based on Gaussian processes to improve the spatial reuse in WLANs. INSPIRE makes no explicit assumptions about the topology of WLANs and favors altruistic behaviors of the access points, leading them to find adequate configurations of their TX_POWER and OBSS_PD parameters for the "greater good" of the WLANs. We demonstrate the superiority of INSPIRE over other state-of-the-art strategies using the ns-3 simulator and two examples inspired by real-life deployments of dense WLANs. Our results show that, in only a few seconds, INSPIRE is able to drastically increase the quality of service of operational WLANs by improving their fairness and throughput.

CCS CONCEPTS

• **Networks → Wireless local area networks**; **Network performance analysis**; • **Theory of computation → Bayesian analysis**; **Reinforcement learning**; **Gaussian processes**; **Distributed algorithms**.

KEYWORDS

Machine Learning; Gaussian Process; Spatial Reuse; IEEE 802.11; Power Control

1 INTRODUCTION

Since their introduction in the late 1990s, WLANs (Wireless Local Area Networks) have rapidly overtaken wired networks to become the primary means of connecting devices to the Internet. According to Cisco [9], they will account for 57% of the Internet traffic in 2022, compared to 22% and 21% for mobile and wired networks, respectively. The current WLAN architecture is defined by the IEEE standard 802.11 (commercially known as Wi-Fi). APs (Access Points) are the centerpiece of this setup, serving as relays for wireless devices; we refer to the latter as STAs (Stations) throughout this paper. Typically, each AP is equipped with a wired interface giving access to the LAN and then the Internet as well as a wireless interface providing connectivity to nearby STAs through radio communications.

Space on the radio spectrum is a scarce resource as it is often shared by multiple WLANs. The radio bands used by the IEEE 802.11 standard (currently 2.4 and 5 GHz, soon to be joined by 6 GHz) are divided into channels. Different APs can then be assigned to different, orthogonal channels enabling the APs to transmit at the same time without interfering with each other. Equally important thanks to the limited radio range of radio waves, APs configured on the same radio channel can transmit concurrently provided that they are sufficiently far away from each other. This ability was central to the success of WLANs and it is commonly known as the spatial reuse of radio channels.

However, the spatial reuse of radio channels as performed by today's WLANs may be reaching its limit. This is particularly true in places where WLAN deployments are very dense, such as offices, shopping malls and train stations. This is because, in these areas the distance between APs is small, so that an AP is more likely to be blocked by the transmissions of one or several nearby APs operating on the same channel. This will in turn take a hefty toll on the WLANs' performance.

A solution to this issue can be found in the 2021 amendment to 802.11 known as 802.11ax [2], which enables the dynamic configuration of two key parameters at each AP: TX_PWR and OBSS_PD. The former parameter specifies the power level (in dBm) at which the AP transmits its data. The latter parameter defines the sensitivity threshold (in dBm). If the energy received is below this level, this indicates to the AP that the radio channel is clear and thus available for transmission. Otherwise, the AP must defer its transmissions. While prior amendments to 802.11 held these TX_PWR and OBSS_PD parameters constant (typically 20 dBm and -82 dBm respectively), 802.11ax has made them dynamic with their values spanning from 1 to 21 dBm for the former and from -82 to -62 dBm for the latter. Adjusting the configurations of TX_PWR and OBSS_PD can help overcome the limitations of spatial reuse in dense environments by allowing APs that are close to each other to transmit on the same channel. Figure 1 depicts a simple example of two APs operating on the same radio channel and illustrates how different configurations of the TX_PWR and OBSS_PD parameters can lead to different performance.

Despite the potential of 802.11ax to improve the spatial reuse of radio channels, finding an adequate configuration of TX_PWR and OBSS_PD for the APs in a WLAN is a complex problem. First, an adequate configuration is very topology-specific. In other words, knowing a suitable configuration for a given scenario is of no value for another scenario. Second, a distributed solution would be more appreciated than a centralized solution. Not only does this avoid the search in an otherwise very high dimensional space but this
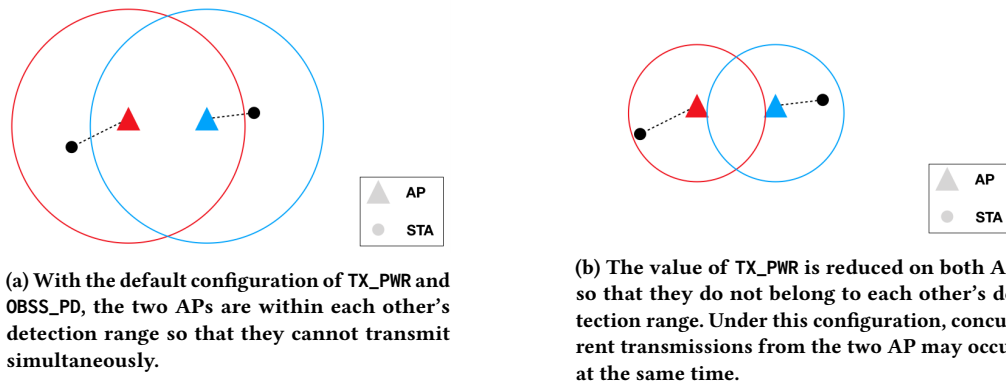
(a) With the default configuration of `TX_PWR` and `OBSS_PD`, the two APs are within each other's detection range so that they cannot transmit simultaneously.

(b) The value of `TX_PWR` is reduced on both AP so that they do not belong to each other's detection range. Under this configuration, concurrent transmissions from the two AP may occur at the same time.

Figure 1: Adequately configuring the `TX_PWR` parameter of APs can significantly improve the spatial reuse of radio channels in WLANs. Note that concurrent transmissions of the two APs could also be attained by increasing `OBSS_PD` at each AP. While similar, reducing `TX_PWR` and increasing `OBSS_PD` may affect the WLANs' performance differently (see Table 2 of [20] for more details).

avoids the assumption of having a centralized entity (e.g., a controller) deciding the configurations of all APs. This assumption is acceptable if all the interfering APs belong to the same WLAN but unrealistic if they belong to concurrent WLANs. Third, forecasting the performance of WLANs with an analytical model that can subsequently help establish "optimal" configurations is difficult. The degree of details in the models will be either too coarse and thus inapplicable, or adequate but unscalable when scenarios involve multiple APs and STAs. A solution to this is that APs can apply new configurations of their parameters, measure the effect of these changes on their performance, and exchange their experience with surrounding APs. This paves the way for the use of online and reinforcement learning techniques in a distributed manner.

In a previous work [5], we introduced a centralized solution based on the sampling of Gaussian mixtures and Thompson sampling to improve the spatial reuse of WLANs by adequately configuring `TX_PWR` and `OBSS_PD` parameters at each AP. In this paper, we present a distinct contribution named `INSPIRE`, which is a distributed online learning solution performing local Bayesian optimizations based on GPs (Gaussian Processes) to address the issue of improving the spatial reuse of WLANs. `INSPIRE` makes no explicit assumptions about the topology of WLANs or the radio environments and thus can apply to any WLANs. Additionally, it can operate even when the APs to be configured belong to different concurrent WLANs. The contributions of this paper are as follows:

- We demonstrate the ability of GPs at approximating the reward function, which reflects the performance of WLANs, and at exploring efficient AP configurations;
- We establish the superiority of a divide-and-conquer approach to handle the complex problem of setting the `TX_PWR` and `OBSS_PD` parameters at each AP;
- We introduce `INSPIRE` a distributed solution that lets the APs of concurrent WLANs automatically adapt their internal parameters' setting in their own interest as well as in the interest of obtaining a more efficient spatial reuse of radio channels;

- We evaluate the efficiency of `INSPIRE` on real-life inspired case studies using a detailed network discrete-event simulator and compare its performance with several state-of-the-art solutions (centralized or distributed).

The remainder of this paper is organized as follows. The next section discusses the related work. Section 3 describes the proposed strategy and the main theoretical results obtained to address the issue of spatial reuse of a radio channel in WLANs. Its performance are evaluated in Section 4.2. Section 5 concludes this paper.

2 RELATED WORK

The release of the IEEE 802.11ax amendment [1] in late 2021 marks a new era for the spatial reuse of radio channels of WLANs: Nodes can dynamically adjust their transmission power (`TX_PWR`) and sensitivity threshold (`OBSS_PD`) parameters. For a detailed explanation of how this new feature is implemented, we refer the interested reader to [21], which also provides simple scenarios to illustrate its potential benefits.

Years before IEEE released the 802.11ax amendment, the idea of dynamically updating `TX_PWR` and `OBSS_PD` has been explored by some researchers. The pioneering work of [24] presents an analytical model that, based on the current radio channel conditions, dynamically configures `OBSS_PD` on each node of a Wi-Fi-based mesh network. Concurrently, [14] established that adapting `TX_PWR` can lead to increased throughput and reduced energy consumption. More recently, in 2020, [17] casts the issues of positioning the APs of a WLAN and choosing their `TX_PWR` as an optimization problem. The authors provides a solution to this problem that delivers a static configuration of `TX_PWR` for a WLAN. But their solution does no account for the number of STAs nor the type of traffic in the WLAN.

The difficulty of accurately modeling the dependency between the configuration parameters of a large WLAN and its performance is a strong hurdle to the development of spatial reuse strategies based on analytical models. As a result, most of the proposed strategies are data-driven. Adaptive by construction, they seem to constitute promising candidates in the search for configurations that

improve the spatial reuse of a radio channel and the performance of WLANs.

Machine learning (ML) techniques are natural candidates for addressing problems requiring a data-driven approach, and the spatial reuse problem is no exception. [4] addresses the problem of configuring TX_PWR and OBSS_PD with a two-scale solution using artificial neural networks (ANN). In their strategy, STAs and APs first adjust their value of OBSS_PD to minimize interference. Then, an ANN, which was trained offline through simulation, is used to increase the fairness between STAs in terms of attained throughput. However, given the vast diversity of WLAN topologies, the offline learning of the ANN appears as a clear limitation to the generalization of this strategy. An online learning procedure is proposed by [5], which uses reinforcement learning and more precisely the Multi-Armed Bandit (MAB) framework to find the optimal configuration of TX_PWR and OBSS_PD in a WLAN. The approach comprises two agents with one sampling promising configurations through a multivariate normal distribution, and the other identifying the best configuration among those already sampled with Thompson sampling and Normal-Gamma priors. These two ML solutions [4, 5] were tested on the network simulator ns-3 and lead to significant WLAN improvements. However, in order to perform their optimization, they both assume the presence of a central controller that has access and control over all the APs in the WLANs. By construction, these approaches are centralized, and hence cannot be applied to cases where concurrent WLANs managed by different owners interfere with others.

Distributed approaches are undisputedly better fit than centralized approaches to handle cases with a set of concurrent WLANs. [3] introduces a distributed algorithm named Dynamic Sensitive Control which is run on every STAs of a WLAN. In short, each STA tries to dynamically reduce its value of OBSS_PD to favor concurrent transmissions while keeping it high enough to ensure a high quality signal reception. Similarly, [15] proposes Link-aware Spatial Reuse (LSR), a distributed algorithm designed for the APs. In LSR, each AP chooses another AP, which is allowed to transmit concurrently, and then prescribes a value of TX_PWR for the selected AP. These two algorithms rely on a single measurement metric reflecting the quality of the received signal, namely the Received Signal Strength, to choose the nodes configuration. More recently, strategies using distributed MAB approaches have been proposed [20, 22]. They both use Thompson sampling with Gaussian priors to find the best couple of TX_PWR and OBSS_PD at each AP. In [22], each AP seeks to maximize the throughput of its associated STAs. On the other hand, in [20], the authors assume that every AP has access to the performance of all other APs in the WLAN; then each AP attempts to maximize a global reward that takes into account the performance of all the other nodes. Both strategies [20, 22] solutions were evaluated in a self-made simulator with simple random scenarios.

Table 1 summarizes the main characteristics of the data-driven strategies discussed above. It shows that, out of the six considered strategies, two (namely [3, 4]) only focus on the configuration of the OBSS_PD parameter (keeping the TX_PWR parameter fixed). To help in the comparison of the different strategies, we introduce two concepts: "pull area" and "push area". The pull area indicates the area on which each node is assumed to obtain information (this typically includes parameter configurations and performance

measurements). Depending on the strategy being considered, the pull area can include just the node itself, the surrounding nodes, or the whole set of nodes in the WLANs. The push area designates the area which each AP can influence typically through the prescription of parameter configurations. In the case of centralized strategies (e.g., [4, 5]), the pull and push areas naturally cover the whole set of APs. We distinguish partially distributed strategies (e.g., [20]) wherein either the pull or push area includes the whole set of APs with fully distributed strategies (e.g., [3, 15, 22]) in which both the pull and push areas differ from the whole set of APs. We observe in Table 1 that only three out of the six state-of-the-art strategies can be considered as fully distributed. The last four columns of Table 1 pertain to the performance evaluation used to validate each of these strategies. It appears that most strategies were evaluated without considering the dynamical selection of the Modulation Coding Scheme (MCS) for the speed of the wireless links, nor bidirectional (with upstream and downstream) traffic. This can be seen as a strong limitation since this overlooks some associated trade-offs. For instance, increasing the value of TX_PWR certainly enables the communication to operate with a faster data rate (larger MCS), but at the same time, it increases the level of interference with surrounding APs. Additionally, most strategies were evaluated on relatively simple scenarios (with a few APs and a limited number of radio channels), often using a self-made network simulator.

In this paper, we propose a fully distributed strategy to address the problem of the spatial reuse of radio channels in WLANs. The proposed strategy can be applied to any arrangement of WLANs and its novelties are mostly twofold. First, to the best of our knowledge, it is the first strategy making use of Gaussian Processes to explore promising WLAN configurations in the quest of discovering the optimal one. Gaussian processes are recognized tools to deal with the exploration vs. exploitation dilemma (see [7, 18]) which is at the center of the spatial reuse problem. Second, unlike the existing fully distributed strategies, INSPIRE allows each AP to account for its surroundings thanks to pull and push areas broader than a single node. Through the use of a simple consensus method, APs of the WLANs achieve to behave altruistically selecting configurations for the "greater good" of the WLANs. We also introduce realistic scenarios, inspired by real-life WLANs, with dynamic MCS and bidirectionnal traffic, to evaluate and compare the efficiency of all the considered strategies.

3 PROPOSED SOLUTION

3.1 WLANs under study

Let $\mathcal{W}$ denote the set of concurrent WLANs under study, each of which being comprised of one or more APs. We let V be the set of APs in $\mathcal{W}$ that operate on the radio channel of interest. We denote by N the number of APs, by s_i the set of STAs associated with AP i and by S the total number of STAs in the considered radio channel of $\mathcal{W}$. Thus, we have: $S = \sum_{i=1}^{N} |s_i|$. Finally, we use $\mathcal{N}_i$ to designate the set of APs that are within the communication range of AP i (when every AP is under the default configuration of the TX_PWR and OBSS_PD parameters). Note that AP i itself belongs to $\mathcal{N}_i$. We refer to the APs in $\mathcal{N}_i$ as the surroundings of AP i.

We make no assumptions on $\mathcal{W}$, including on the specific arrangement of its APs and STAs, other than the three detailed below.

Table 1: Comparison of the state-of-the-art data-driven strategies. The last column refers to the size of the scenarios involved in the performance evaluation of the strategy. For instance, 216/18 means the evaluation comprises 216 APs distributed over 18 radio channels.

Proposed solution	Tuning of TX_PWR	Degree of centralization		Dynamic MCS	Traffic Up/Down	Simulator	APs / channels
		Pull area	Push area				
WCNC'15 [3]		Associated STAs	Associated STAs		Up	Self-made	100/3
WCNC'21 [15]	✓	AP itself	AP itself	✓	Down	ns-3	6/1
Globecom'20 [4]		All APs	All APs		Up/Down	ns-3	3/1
ADHOC'19 [22]	✓	AP itself	AP itself		Down	Self-made	8/1
JNCA'19 [20]	✓	All APs	AP itself		Down	Self-made	8/1
MSWiM'21 [5]	✓	All APs	All APs		Down	ns-3	10/1
INSPIRE	✓	Surrounding APs	Surrounding APs	✓	Up/Down	ns-3	216/18

First, we assume that every AP i is able to exchange control frames (possibly through its beacon frames) with its surrounding APs (i.e., the ones in $\mathcal{N}_i$). By the same token, we suppose that at least one AP i has another AP in its communication range (i.e., $\exists i \in [\![1, N]\!], \mathcal{N}_i \neq \{i\}$), otherwise the spatial reuse of the radio channel would already be at its apex.

Second, we assume that the N APs have their TX_PWR and OBSS_PD parameters configurable (as it is the case since the introduction of the 802.11ax amendment). We use x_i^t to denote the configuration of AP i with regards to its two TX_PWR and OBSS_PD parameters at time t. Analogously, x^t represents the configuration of the N APs from $\mathcal{W}$ at time t. Thus, we have: $x_i^t \in C = [\![-82, -62]\!] \times [\![1, 21]\!]$ dBm and $x^t \in C^N$.

Lastly, we assume that each AP in $\mathcal{W}$ can periodically run performance tests and obtain, in return, the mean throughput attained by each of its STAs over a short time interval Δt. More formally, we use the vector $T^t \in \mathbb{R}^{+S}$ to denote the throughput attained by the S STAs of $\mathcal{W}$ given the WLAN configuration x^t at time t. Throughout this paper, we sometimes refer to T^t as $T(x^t)$ to explicitly show the dependency between the STAs' throughputs and APs' configurations.

In this work, we seek to discover an adequate configuration x^* of the N APs composing $\mathcal{W}$ that improves the collective experience of the S STAs through a better reuse of their radio channel. We address this problem as a reinforcement learning task in which, at regular time intervals t, the APs collect measurements T^t associated to their current configuration x^t, and need to decide their next configuration x^{t+1}. The obstacles towards that objective are mostly threefold. (i) We need to define a meaningful objective function that APs will attempt to optimize collectively; (ii) We are facing the well-known exploration vs. exploitation dilemma since the search for an adequate configuration of the WLANs should be as seamless as possible (without disrupting the STAs). This leads us to cast the problem as a MAB problem where the arms are the WLANs' configurations. Following the MAB terminology, we refer to the objective function as the reward function; (iii) We are looking for a strategy that can be applied in a distributed way since it would be in general unrealistic to assume that (concurrent) APs have a fine knowledge beyond their surroundings.

3.2 Reward function

We need to define a reward function R that appraises the "goodness" of a configuration x with regards to the WLANs performance. Because multiple criteria may be considered in the definition of R, there is no universal definition. However, assessing the quality of a configuration x can be derived from the STAs throughputs $T(x)$ obtained with APs configured with x. Among all easily computable reward functions, $R(x) = \prod_{T_i \in T(x)} T_i$ is called the proportional fairness (PF) and provides a convenient trade-off between fairness and cumulated throughput. However, PF is often criticized for its high variability, since $\frac{\partial R}{\partial T_i} = \prod_{T_j \in T(x), j \neq i} T_j$.

To overcome this drawback, we consider the logarithm of PF. This lowers its variability, which becomes: $\frac{\partial R}{\partial T_i} = \frac{1}{T_i}$ (note that T_i is typically much larger than 1). This also emphasizes the contribution of STAs with low throughputs in the computation of R and provides a pleasant closed-form to optimize. For an arbitrary set of APs X, we can define $R_X = \log \prod_{\substack{i \in X \\ j \in s_i}} T_j(x)$. Then, our *global* reward function R is:

$$R(x) = R_V(x) = \log \prod_{\substack{i \in V \\ j \in s_i}} T_j(x)$$
$$= \sum_{\substack{i \in V \\ j \in s_i}} \log T_j(x) \tag{1}$$

However, to compute Equation 1, an AP must have a complete knowledge of the performance attained by the STAs of all APs or, at least, be able to communicate with all the APs in $\mathcal{W}$. This is in contradiction with our assumption that APs only have a partial knowledge of $\mathcal{W}$, limited to their surrounding APs. To design a reward function compatible with the distributed case, we proceed as follows. Each AP i applies Equation 1 but restricted to the set of its associated STAs and obtains in return a "selfish" reward denoted by $R_{\{i\}}$. Previous work [22] have showed that considering such selfish rewards may have a positive but limited impact on the WLANs performance. Therefore, we introduce a more altruistic reward, denoted by R_i that accounts not only for the "selfish" reward of AP i (i.e., $R_{\{i\}}$) but also for the rewards of the surrounding APs (i.e., the ones in $\mathcal{N}_i$). The "altruistic" local reward of AP i is computed as:

$$R_i(x) = \sum_{j \in \mathcal{N}_i} \frac{R_{\{j\}}(x)}{|\mathcal{N}_j|} \tag{2}$$

Note that R_i defined with Equation 2 ensures that a configuration x maximizing the local rewards is also a maximum for the global reward R since we have: $\sum_{i=1}^{N} R_i(x) = R(x)$.

PROOF. This is a straightforward property since

$$\sum_{i=1}^{N} R_i(x) = \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i} \frac{R_{\{j\}}(x)}{|\mathcal{N}_j|} \tag{3}$$

By noticing that $i \in \mathcal{N}_j \iff j \in \mathcal{N}_i$, we can permute the indices and use the definition of $R_{\{j\}}$:

$$\sum_{j=1}^{N} \frac{R_{\{j\}}(x)}{|\mathcal{N}_j|} \sum_{i \in \mathcal{N}_j} 1 = \sum_{j=1}^{N} \sum_{k \in s_j} \log T_k(x) = R(x) \tag{4}$$

□

3.3 Local reward maximization

For the sake of clarity and since all variables in this section are relative to an AP i, we often omit the subscript i in the notations.

Now that each AP i has its own local reward function, we need a model of the knowledge of AP i about R_i in order to find its argmax. We represent the beliefs of AP i about R_i by defining a prior distribution on the reward functional space with a Gaussian Process (GP).

Gaussian process. In our case, a GP can be defined as a collection of random variables indexed by configurations of APs in $\mathcal{N}_i$: $\{Y_c; c \in C^{|\mathcal{N}_i|}\}$ such that every finite collection $(Y_{c_1}, \cdots, Y_{c_n}) \sim \mathcal{N}(\mu, \Sigma)$. Without loss of generality, we assume the GP to have zero mean so that it is entirely determined by its covariance function $\Sigma : C^{|\mathcal{N}_k|} \times C^{|\mathcal{N}_k|} \to \mathbb{R}^+$. As shown by [23], GPs can be used as priors on a function space. We use X_t to denote the $t \times 2|\mathcal{N}_k|$ features matrix gathering the tested configurations $(x^1, \cdots, x^t)^T$ and Y_t to denote the $t \times 1$ label vector gathering the corresponding local reward values $(R_i(x^1), \cdots, R_i(x^t))^T$. Given X_t and Y_t, we can infer the distribution of the reward value for an arbitrary configuration x, $R_i(x)$, from Bayes' theorem: $R_i(x)|X_t, Y_t \sim \mathcal{N}(\mu(x), \sigma^2(x))$ with $\mu(x)$ and $\sigma^2(x)$ defined in Equations 5 and 6, respectively.

$$\mu(x) = \Sigma(x, X_t)\Sigma(X_t, X_t)^{-1} Y_t \tag{5}$$

$$\sigma^2(x) = \Sigma(x, x) - \Sigma(x, X_t)\Sigma(X_t, X_t)^{-1}\Sigma(X_t, x) \tag{6}$$

Since GPs can be used as a prior on a functional space, they are useful to solve regression problems as well as maximization tasks. In our case, the AP i uses a GP to model R_i and to assist the exploration of promising configurations of the APs in $\mathcal{N}_i$, maximizing R_i in a Bayesian way.

Choosing the covariance function Σ is a critical step when designing a GP as it determines some key features such as its isotropy and smoothness. Since the reward function, which quantifies the quality of spatial reuse in $\mathcal{N}_i$, is likely to exhibit threshold effects, we choose a covariance function that decreases rapidly as the distance between two considered configurations increases. Thus, the regularity constraint is not too restrictive on the modeled function. Because we have no incentive to prefer any particular direction over another, we let the covariance function $\Sigma(x, x')$ depend only

on $||x - x'||$ to ensure the isotropy of the GP. This leads us to use a Matérn kernel [10] with parameter $v = \frac{3}{2}$, which is defined as

$$\Sigma(x, x') = s^2 \left(1 + \frac{\sqrt{3}||x - x'||}{\rho}\right) e^{-\frac{\sqrt{3}||x - x'||}{\rho}} \tag{7}$$

where s^2 and ρ are two hyperparameters whose values are approximated by maximizing the likelihood of Y_t (which is Gaussian) during the learning process.

As discussed before, each AP i faces the exploitation vs exploration dilemma in its attempt to find the optimal configuration. A common way in the MAB framework to appraise a given strategy π is then to consider the cumulative regret $\Gamma(\pi)$. In our problem, $\Gamma(\pi)$ is expressed with Equation 8 for an episode of D steps, since it is expressed as the cumulative sum of the differences between the best reward that the AP can get and $R_i(\pi(t))$, which is the actual reward obtained at time t for the strategy π.

$$\Gamma(\pi) = \sum_{t=1}^{D} \max_{x \in C^{|\mathcal{N}_i|}} R_i(x) - R_i(\pi(t)) \tag{8}$$

Minimizing the cumulative regret with GP models is usually done by defining a strategy π from the maximization of an acquisition function A: $\pi(t) = \arg\max_{x \in C^{|\mathcal{N}_i|}} A_t(x)$. However, this assumes that our search space $C^{|\mathcal{N}_i|}$ is continuous. Since each AP i deals with discrete configurations of APs in $\mathcal{N}_i$, we systematically round the recommendation of the GP to the nearest valid WLAN configuration. Many acquisition functions exist, such as Knowledge Gradient (KG) [11], GP-UCB [19] or the Expected Improvement (EI) [13]. We choose EI over KG (whose computational cost can rapidly become prohibitive) and GP-UCB (which was found to be less efficient on our examples). The EI acquisition function is expressed as $A_t(x) = \mathbb{E}\left[(\mu_{t+1}(x) - \max_{1 \le k \le t} R_i(x_k))^+\right]$ given that $X_{t+1} = (X_t, x)$, $Y_{t+1} = (Y_t, R_i(x))$. Since $R_i(x) \sim \mathcal{N}(\mu(x), \sigma^2(x))$, we can derive a convenient closed-form for EI, as shown in Equation 9.

$$EI(x) = (\mu(x) - R_{i,t}^*)\Phi(Z) + \sigma(x)\phi(Z) \tag{9}$$

with $R_{i,t}^* = \max_{1 \le k \le t} R_i(x_k)$, $Z = \frac{\mu(x) - R_{i,t}^*}{\sigma(x)}$, Φ and ϕ being respectively the CDF and the PDF of a standard Gaussian distribution.

Then, the AP i can try to maximize Equation 9 by differentiating it and performing a gradient ascent. By applying its strategy $\pi_i(t) = \arg\max_{x \in C^{|\mathcal{N}_i|}} EI(x)$ and classical gradient ascent techniques on Equation 9, AP i provides promising configurations for its surrounding APs in $\mathcal{N}_i$.

3.4 Aggregation of local prescriptions

In the previous sections, we have described how each AP i computes its local reward and relies on its model $\mathcal{GP}_i$ to explore promising configurations for the APs in $\mathcal{N}_i$.

However, more coordination between APs is required. By construction, the collection $\mathcal{F} = (\mathcal{N}_k)_{1 \le k \le N}$ is a cover of the set of APs in $\mathcal{W}$ but not a partition. In fact, if $\mathcal{F}$ had only null intersections (i.e., $\forall j, k, \mathcal{N}_j \cap \mathcal{N}_k = \emptyset$), then the spatial reuse of the radio channel would already be at its apex and there is no need for improvement. Figure 2 illustrates an example with 5 APs in which the collection $\mathcal{F} = (\mathcal{N}_1, \mathcal{N}_2, \mathcal{N}_3, \mathcal{N}_4, \mathcal{N}_5)$ exhibits multiple non-null intersections. As a result, most APs will receive *a set* of *different* prescriptions for
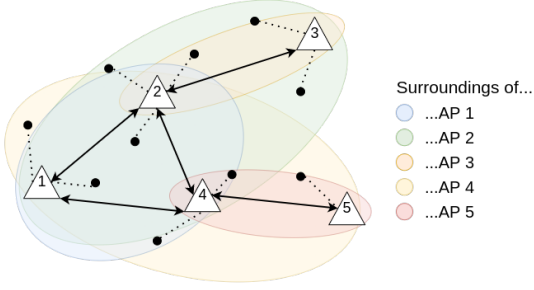
Figure 2: A WLAN represented by a graph with APs depicted as labelled triangles and STAs as black dots. An edge exists between two APs when they are in the communication range of each other. We use different colors to illustrate the surroundings of each AP in $\mathcal{F}$.

the configuration of their TX_PWR and OBSS_PD parameters at their next iteration. For instance, AP 1 will receive prescriptions from APs 2 and 4 in addition to its own prescription. Since APs can only test one configuration at a time, one of those prescriptions must be chosen, or preferably, a consensus between them must be made.

In general, maximizing local rewards is very likely to lead to a sub-optimal situation since, for non-linear optimization problems, individual interests are often not aligned with the global interest (e.g., the famous Tragedy of the Commons [12]). Without more information on the relation between the configuration of the APs and the measured throughputs of STAs, it seems difficult to provide an expression for the argmax of the global reward function R given the argmax of the local reward functions R_i. However, recall that $\sum_{i=1}^{N} R_i(x) = R(x)$. We can leverage this property to provide guarantees.

THEOREM 1. *Let $\{R_i\}_{1 \leq i \leq N}$ be a set of Lipschitzian functions, $w \in [0,1]^N, ||w||_1 = 1$ be a weight vector and $\{x_j^i\}_{i \in \mathcal{N}_j}$ be the prescriptions received by the AP j, with $x^i = \arg\max_{x \in C^{|\mathcal{N}_i|}} R_i(x)$, then the weighted marginal median of the received sets of prescriptions is a maximin optimum $\tilde{x}$ of $\sum_{i=1}^{N} R_i(\tilde{x}_{\mathcal{N}_i})$:*

$$\tilde{x}_j = med\left(\{\left(x_j^i, w_i\right)\}_{i \in \mathcal{N}_j}\right) \tag{10}$$

PROOF. Let x^i be the prescription of AP i, with x_j^i the prescription of AP i for AP j and $w \in [0,1]^N$ a normalized weight vector ($||w||_1 = 1$). Consider the quantities $R^* = \sum_{i=1}^{N} w_i R_i(x^i)$ and $\tilde{R} = \sum_{i=1}^{N} w_i R_i(\tilde{x}_{\mathcal{N}_i})$ for a given consensus $\tilde{x} \in C^N$. If $x^i = \arg\max_{x \in C^{|\mathcal{N}_i|}} R_i(x)$ then it follows that $\forall \tilde{x} \in C^N, R^* - \tilde{R} > 0$. We want to minimize this difference. If all the functions R_i are λ-Lipschitzian, we have:

$$R^* - \tilde{R} = \sum_{i=1}^{N} w_i R_i(x^i) - \sum_{i=1}^{N} w_i R_i(\tilde{x}_{\mathcal{N}_i})$$

$$= \sum_{i=1}^{N} w_i(R_i(x^i) - R_i(\tilde{x}_{\mathcal{N}_i}))$$

$$\leq \lambda \sum_{i=1}^{N} w_i ||x^i - \tilde{x}_{\mathcal{N}_i}||_1$$

$$\leq \lambda \sum_{i=1}^{N} w_i \sum_{j \in \mathcal{N}_i} \sum_{d=1}^{D} |x_{j,d}^i - \tilde{x}_{j,d}| = \Psi(\tilde{x}) \tag{11}$$

by expliciting the L_1 norm over the D dimensions of C. By rearranging the indices and splitting the absolute values, we have:

$$\Psi(\tilde{x}) = \lambda \sum_{d=1}^{D} \sum_{j=1}^{N} \sum_{i \in \mathcal{N}_j} w_i |x_{j,d}^i - \tilde{x}_{j,d}|$$

$$= \lambda \sum_{d=1}^{D} \sum_{j=1}^{N} \left(\sum_{\substack{i \in \mathcal{N}_j \\ x_{j,d}^i < \tilde{x}_{j,d}}} w_i(\tilde{x}_{j,d} - x_{j,d}^i) + \sum_{\substack{i \in \mathcal{N}_j \\ x_{j,d}^i \geq \tilde{x}_{j,d}}} w_i(x_{j,d}^i - \tilde{x}_{j,d}) \right) \tag{12}$$

We want to minimize $\Psi(\tilde{x})$. This is equivalent to finding $\tilde{x}$ so that $\nabla \Psi$ is 0, which can be written as, $\forall(j,d) \in [\![1,N]\!] \times [\![1,D]\!]$:

$$\frac{\partial \Psi}{\partial \tilde{x}_{j,d}} = \lambda \left(\sum_{\substack{i \in \mathcal{N}_j \\ x_{j,d}^i < \tilde{x}_{j,d}}} w_i - \sum_{\substack{i \in \mathcal{N}_j \\ x_{j,d}^i \geq \tilde{x}_{j,d}}} w_i \right) = 0 \tag{13}$$

The only value of $\tilde{x}_{j,d}$ which ensures this relation is the weighted median of the sample $\{x_{j,d}^i\}_{i \in \mathcal{N}_j}$, provided that the weights can cancel out. It remains to verify that this critical point is a minimum by considering the coefficients of the Hessian matrix. We can rewrite the partial derivative of Ψ:

$$\frac{\partial \Psi}{\partial \tilde{x}_{j,d}} = -\lambda \sum_{i \in \mathcal{N}_j} w_i \left(x_{j,d}^i - \tilde{x}_{j,d}\right)$$

$$= -\lambda \sum_{i \in \mathcal{N}_j} w_i \left(2H_{\frac{1}{2}}\left(x_{j,d}^i - \tilde{x}_{j,d}\right) - 1\right) \tag{14}$$

with $H_{\frac{1}{2}}$ the Heaviside function. Then, we obtain the Hessian matrix coefficients:

$$\frac{\partial \Psi}{\partial \tilde{x}_{j,d} \partial \tilde{x}_{j',d'}} = \begin{cases} 0 & \text{if } (j,d) \neq (j',d'), \\ 2\lambda \sum_{i \in \mathcal{N}_j} w_i \delta\left(x_{j,d}^i - \tilde{x}_{j,d}\right) & \text{otherwise.} \end{cases} \tag{15}$$

with δ the Dirac impulsion. These coefficients define a positive definite matrix if $\tilde{x}_{j,d} \in \{x_j^i\}_{i \in \mathcal{N}_j}$, which is necessarily the case if $\tilde{x}_{j,d}$ is a weighted median of the sample. However, we must not create new data: if two elements are eligible to be the weighted

median, we must select one of them and not taking the average between the two.

Since the weighted marginal median of the prescriptions minimizes the upper bound of $R^* - \tilde{R}$, it is a maximin optimum for $\tilde{R} = \sum_{i=1}^{N} w_i R_i(\tilde{x}_{\mathcal{N}_i})$. $\qquad\qquad\qquad\qquad\qquad\square$

Since, by definition of the local reward functions, $\sum_{i=1}^{N} R_i(x) = R(x)$ (see Eq. 4), we can apply Theorem 1 with, $\forall i \in [\![1, N]\!], w_i = \frac{1}{N}$ and state that taking the marginal median of the prescriptions is a good way to reach high values of the global reward R.

3.5 Algorithm and complexity

Algorithm 1 INSPIRE run at each AP i

Input: subset $\mathcal{N}_i$ of APs

1: Initialize the Gaussian Process $\mathcal{GP}_i$
2: **while true do**
3:     Find a prescription $x^i = \arg\max_{x \in C^{|\mathcal{N}_i|}} EI_i^t(x)$ by gradient ascent
4:     Broadcast x^i to APs in $\mathcal{N}_i$
5:     Receive the prescriptions x_i^j from AP j, $j \neq i, j \in \mathcal{N}_i$
6:     Compute the consensus $\tilde{x}_i^{t+1}$ with Equation 10
7:     Test x_i^{t+1} for Δt seconds and compute its selfish reward $R_{\{i\}}$ with Equation 1 applied only to AP i
8:     Broadcast $R_{\{i\}}$, $|\mathcal{N}_i|$ and $\tilde{x}_i^{t+1}$ to APs in $\mathcal{N}_i$
9:     Receive $R_{\{j\}}$, $|\mathcal{N}_j|$ and $\tilde{x}_j^{t+1}$ from AP j, $j \neq i, j \in \mathcal{N}_i$
10:     Compute the local reward R_i with Equation 2 and the local configuration $\tilde{x}_{\mathcal{N}_i}^{t+1}$
11:     Add the pattern $\left(\tilde{x}_{\mathcal{N}_i}^{t+1}, R_i\right)$ to $\mathcal{GP}_k$
12: **end while**

Algorithm 1 summarizes the main steps of our proposed strategy INSPIRE run on each AP of the WLANs.

Contrary to what one might think, the most resource intensive operation in Algorithm 1 is not the inversion of the $\Sigma(X_t, X_t)$ matrix. Since, at step t, we already know the Cholesky decomposition of $\Sigma(X_{t-1}, X_{t-1}) = LL^T$, it is very easy to compute the Cholesky decomposition of $\Sigma(X_t, X_t)$. The most resource-intensive operation is the maximization of the acquisition function through gradient ascent. Since this requires computing many matrix-vector multiplications for at most m steps, the computational complexity of Algorithm 1 at time t is $O\left(mt^2\right)$.

It is worth noting that the dimensionality of the problem (i.e. $\dim(C^{|\mathcal{N}_i|}) = |\mathcal{N}_k| \dim C$) does not appear in the expression of the asymptotic computational complexity of INSPIRE. This interesting property results from the use of a kernel function by GPs to compare WLANs configurations. This gives INSPIRE the ability to handle arbitrarily dense WLANs, or to optimize more parameters than just TX_PWR and OBSS_PD, without taking a hefty toll on its execution time. In fact, the real burden to the execution time of INSPIRE is t. This compels us to bound the size of X_t and to find a balance between the amount of collected data on the WLANs' performance and configuration and a quick execution time. We keep the possibility of approximation methods to reduce the computational complexity of INSPIRE for future works. As for now, we

recommend using windowing methods (such as a moving window) to bound the size of X_t and so the computational complexity of INSPIRE.

4 PERFORMANCE EVALUATION

4.1 Experimental settings

To evaluate the ability of INSPIRE at improving the spatial reuse of a radio channel through the configuration of the TX_PWR and OBSS_PD parameters, we consider two distinct scenarios.

The first scenario is inspired by the WLAN deployment of Cisco in their offices in San Francisco. In [8], Cisco provides the location of 60 APs that together deliver wireless connectivity to their employees on a floor. To account for the WLANs' activity from other floors, we consider a three-floor building and we replicate on each floor the same arrangement of APs as in Cisco's offices. This leads us to a total number of 180 APs spanned over three floors. Assuming 18 independent radio channels, we run a radio channel allocation algorithm to determine the radio channel used by each AP. For our first scenario, we consider the subgraph resulting from the channel allocation with the highest density. We use **T1** to refer to this topology (i.e., arrangements of APs and STAs), which is illustrated in Figure 3a. **T1** exhibits a total of 10 APs and we associate a number of 5 STAs per AP.

The second scenario addresses the case of many single-AP WLANs deployed and operated independently in a relatively limited area. This is typically the case in housing units where each apartment is equipped with its own AP so that the APs are often only a few meters away from a number of others. More specifically, we consider a nine-story building with 216 apartments of 25 m² each. We randomly position an AP within each apartment as well as 4 STAs per AP. Then, similarly to the first scenario, we apply a radio channel allocation algorithm given a total of 18 radio channels, to obtain the topology of interest denoted by **T2**. Note that **T2** consists of 14 APs and 56 STAs. Figure 3b depicts the topology **T2**.

For each scenario, we consider heavily loaded conditions. APs attempt to transmit frames to each of their associated STAs at a rate of 50 Mbps while the latter attempt to send their frames to the AP at a lower rate of 3.33 Mbps. These assumptions are in line with the downstream traffic largely exceeding the upstream traffic in WLANs. Given the speed of wireless links in 802.11ax, the buffers of the APs will always be full of frames waiting to be sent. More generally, considering APs in saturation represents undoubtedly the most difficult case when dealing with the spatial reuse of a radio channel. Therefore, if INSPIRE manages to significantly improve the WLANs' performance under these circumstances, then it can only do better under normal conditions.

To better appraise the quality of INSPIRE, we also consider a control strategy as well as several state-of-the-art solutions, which were discussed in Section 2 and briefly summarized here:

- DEFAULT: Every AP keeps its default configuration for the TX_PWR and OBSS_PD parameters (i.e., $(-82, 20)$ dBm);
- WCNC'15: Each AP implements a simple distributed algorithm to dynamically update its OBSS_PD parameter [3];
- JNCA'19: Each AP solves a MAB problem using Thompson sampling to dynamically update their TX_PWR and OBSS_PD parameters [20];
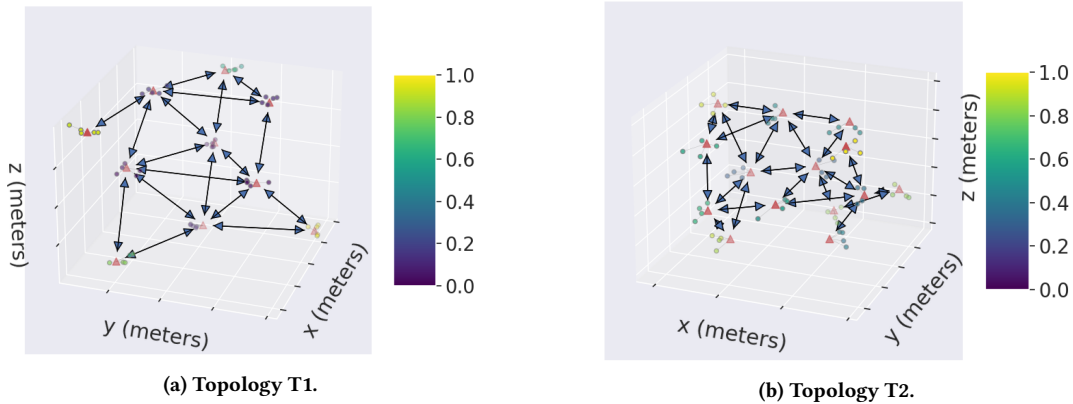
(a) Topology T1.

(b) Topology T2.

Figure 3: The two considered topologies. APs are shown as red triangles, connected by a arrow if they lie in each other's communication range. Associated STAs are shown as dots colored according to their throughputs: warm, yellowish colors indicate that the STA has enough throughput most of the time while, on the contrary, cool, blueish colors indicate that the STA has mostly not enough throughput under the default configuration of 802.11: 20 dBm for `TX_PWR` and -82 dBm for `OBSS_PD`.

- `MSWiM'21`: Similar to `JNCA'19`, except that the sampling of new configurations is performed through a multivariate Gaussian mixture, and that the solution is centralized [5].

We implemented `INSPIRE` (based on the open-source Gaussian process library LibGP [6]) as well as the four strategies described above in the open-source network simulator ns-3 [16]. ns-3 is a well-established realistic discrete-event simulator that implements most of the network protocols involved in the WLANs communication from the Physical up to the Application layer. We report in Table 2 the simulation parameters used in the rest of this section. Unlike previous works (e.g., [3–5, 20, 22])) with the exception of [15], our simulations incorporate the mechanism of rate adaptation that let APs and STAs dynamically vary the speed of their wireless links (through the use of different Modulation Coding Scheme (MCS)) in response to the quality of the received signal. This is particularly important for the sake of our study since changing the value of `TX_PWR` necessarily affects the quality of the received signal and thus the MCS. Since our simulated WLANs take place in buildings, we choose an appropriate path loss by combining the models `ItuR1238` and `InternalWallsLoss`, both implemented by ns-3. With these propagation models, the signal is decreased by an additional attenuation coefficient each time it goes through a floor or a wall. The attenuation coefficients are respectively -4 dBm (which is the default value in `ItuR1238`) and -8 dBm.

We instrumented ns-3 to collect and compute a number of performance metrics. At the end of each iteration, the quality of the spatial reuse is assessed with Equation 1, although distributed strategies may internally use the local reward defined in Equation 2. Then, we compute the classical metric to analyze the efficiency of a strategy at dealing with a MAB problem: (i) The cumulative regret (with Equation 8 using a normalized version of the global reward in Equation 1). We also collect the following performance metrics to reflect the effect of each strategy on the behavior of the WLANs and of their STAs: (ii) The number of starving STAs, which we define as STAs experiencing a very low throughput (namely, less than 10%

Table 2: ns-3 parameters.

Parameter	Value
ns-3 version	3.31
Number of repetitions	22
Simulation duration	30 s
Duration of an iteration (Δt)	75 ms
Packet size	1,464 bytes
Downlink traffic	50.0 Mbps
Uplink traffic	3.33 Mbps
Channel size	20 MHz
Frequency band	5 GHz
A-MDPU Aggregation	4
Path loss	`HybridBuildings` (`ItuR1238` + `InternalWallsLoss`)
Wi-Fi Manager	`IdealWifiManager`

of their attainable throughput) and (iii) The cumulated throughput, which simply sums all STAs' throughput.

Each simulation lasts 30 seconds and we replicated them independently 22 times to obtain and visualize their first, second, and third quartiles. When the quartiles of a performance metric vary too much within a single simulation, we apply an exponential moving average (with $\alpha = 0.04$) to extract the underlying trends of the quartiles sequences. The metrics are collected throughout the whole duration of the simulation. At the end of each iteration, we compute all the performance metrics and then we refer to the current strategy to decide what will be the next configuration of the WLANs. Since an iteration lasts Δt = 75 ms and a simulation lasts 30 seconds, the quality of each solution is assessed over 400 iterations.

4.2 Numerical results

Figure 4 illustrates the performance metrics delivered by the ns-3 simulator for each strategy in the case of topology **T1**. The cumulative regret, represented in Figure 4a, indicates which strategy has performed the best at any time of the simulation. INSPIRE is found to be the most efficient strategy, reducing the cumulative regret by 70% more than DEFAULT and by over 50% than WCNC'15, which happens to be the most efficient state-of-the-art strategy. We now look at the other performance metrics to better understand how much INSPIRE is able to improve the behavior of the WLAN and of its STAs. Taking DEFAULT as baseline, Figure 4b shows that INSPIRE reduces the number of STAs in starvation by 80% while Figure 4c demonstrate that our proposed solution manages to increase the cumulated throughput (+400%).

We now turn to the case of topology **T2**. First, we observe in Figure 5a that among the four considered strategies, INSPIRE is the one that manages to decrease the most the cumulative regret with a decline of about 36% compared to the DEFAULT configuration at the end of the simulation. The proposed solution also outperforms MSWiM'21, which is found to be the best state-of-the-art strategy on this topology, by a margin of 14%. Looking at the performance of WLANs and of their STAs, Figure 5b shows that INSPIRE is able to limit the number of STAs starving from throughput by a degree of 36% when compared to the DEFAULT configuration. Similarly, the cumulated throughput of STAs have their value increased by 28% and nearly doubled with INSPIRE (see Figure 5c).

Overall, through the study of topologies **T1** and **T2**, INSPIRE demonstrates its superiority over the other state-of-the-art strategies. The significant improvements brought by our proposed solution on all performance metrics are permanently obtained after 100 iterations only (7.5 seconds). In other words, in less than 10 seconds, INSPIRE manages to significantly improve the behavior of the WLANs and of the associated STAs thanks to a better spatial reuse of the radio channel. This efficiency in searching and finding an adequate configuration of the TX_PWR and OBSS_PD parameters at each AP of the WLANs mostly results from the distributed, altruistic use of Gaussian processes which we further discuss in the next section.

5 CONCLUSIONS

In this work, we have presented INSPIRE, a reinforcement learning method to improve the spatial reuse of radio channels in WLANs by configuring two parameters of APs: the transmission power (TX_PWR) and the sensitivity threshold (OBSS_PD), that can be dynamically configured with the latest Wi-Fi amendments. To address the difficult problem of sharing efficiently and fairly the resource of a radio channel, INSPIRE works as a distributed solution where each AP solves a local Multi-Armed Bandit problem with the help of information and actions limited to its surrounding APs (*i.e.* within its communication range). The development of the solution includes (i) an intuitive quantification (based on STAs throughputs) of the "goodness" of a configuration of TX_PWR and OBSS_PD for concurrent APs in WLANs, both at local and global scales, (ii) the use of an acquisition function and Gaussian processes to find local configurations that maximize approximations of local reward functions, and (iii) an altruistic behavior facilitated by prescriptions to

surrounding APs along with a consensus method which aggregates the prescriptions of surrounding APs for the "greater good" of the WLANs.

INSPIRE has been evaluated and compared with other state-of-the-art strategies addressing the same problem, using the open-source network simulator ns-3 that implements all the layers of the network stack. The different strategies were compared on two examples inspired by real-life deployments of dense WLANs in both professional and domestic environments. INSPIRE was found to outperform other state-of-the-art strategies by significantly reducing the number of STAs in starvation and increasing the cumulated throughput of the WLANs in only a few seconds.

As future work, we plan to assess the quality of INSPIRE for a specific class of WLANs where STAs are mobile (e.g., customers in a shopping mall). Another natural follow-up would be to experiment INSPIRE with real material on a testbed.

REFERENCES

[1] 2021. 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. *IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016)* (2021).

[2] 2021. 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 1: Enhancements for High-Efficiency WLAN. *IEEE Std 802.11ax-2021 (Amendment to IEEE Std 802.11-2020)* (2021), 1–767. https://doi.org/10.1109/IEEESTD.2021.9442429

[3] M. Shahwaiz Afaqui, Eduard Garcia-Villegas, Elena Lopez-Aguilera, Graham Smith, and Daniel Camps. 2015. Evaluation of dynamic sensitivity control algorithm for IEEE 802.11ax. In *IEEE Wireless Communications and Networking Conference (WCNC'15)*. 1060–1065. https://doi.org/10.1109/WCNC.2015.7127616

[4] Elif Ak and Berk Canberk. 2020. FSC: Two-Scale AI-Driven Fair Sensitivity Control for 802.11ax Networks. In *IEEE Global Communications Conference (GLOBECOM'20)*. 1–6. https://doi.org/10.1109/GLOBECOM42002.2020.9322153

[5] Anthony Bardou, Thomas Begin, and Anthony Busson. 2021. Improving the Spatial Reuse in IEEE 802.11ax WLANs: A Multi-Armed Bandit Approach. In *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM'21)*.

[6] Manuel Blum. 2016. LibGP. GitHub, https://github.com/mblum/libgp.

[7] Sayak Ray Chowdhury and Aditya Gopalan. 2017. On kernelized multi-armed bandits. In *International Conference on Machine Learning*. PMLR, 844–853.

[8] Cisco. 2020. High Density Wi-Fi Deployments. https://documentation.meraki.com/Architectures_and_Best_Practices/Cisco_Meraki_Best_Practice_Design/Best_Practice_Design_-_MR_Wireless/High_Density_Wi-Fi_Deployments. Accessed: 2022-01-21.

[9] VNI Cisco. 2019. Cisco visual networking index: Forecast and trends, 2017–2022 white paper. *Cisco Internet Report* 17 (2019), 13.

[10] Marc G. Genton. 2002. Classes of kernels for machine learning: a statistics perspective. *Journal of Machine Learning Research* 2 (3 2002).

[11] Shanti S. Gupta and Klaus J. Miescke. 1996. Bayesian look ahead one-stage sampling allocations for selection of the best population. *Journal of Statistical Planning and Inference* 54, 2 (1996), 229–244. https://doi.org/10.1016/0378-3758(95)00169-7

[12] Garrett Hardin. 2009. The Tragedy of the Commons. *Journal of Natural Resources Policy Research* 1, 3 (2009), 243–253. https://doi.org/10.1080/19390450903037302

[13] Donald R. Jones, Matthias Schonlau, and William J. Welch. 1998. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization* 13 (1998), 455–492. https://doi.org/10.1023/A:1008306431147

[14] Youngsoo Kim, Jeonggyun Yu, and Sunghyun Choi. 2004. SP-TPC: a self-protective energy efficient communication strategy for IEEE 802.11 WLANs. In *IEEE Vehicular Technology Conference (VTC'04)*.
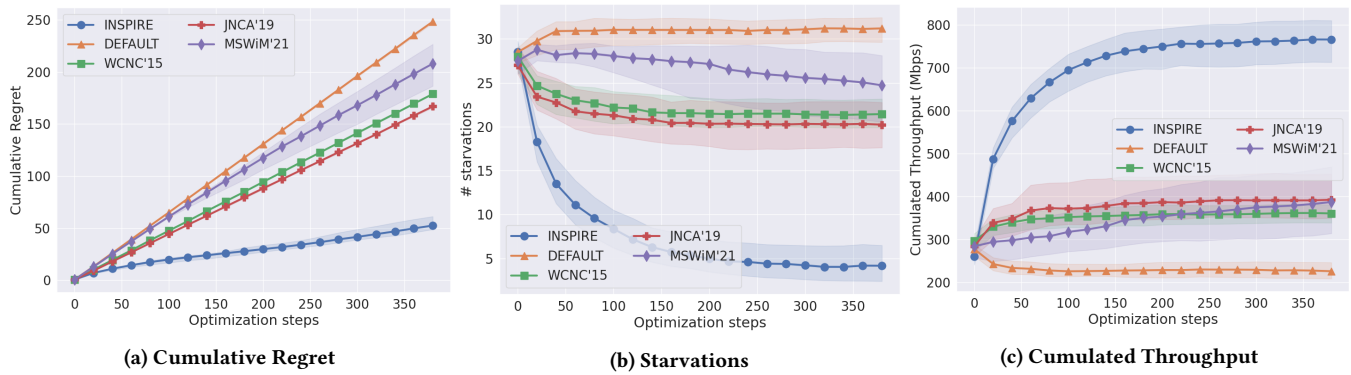
(a) Cumulative Regret

(b) Starvations

(c) Cumulated Throughput

Figure 4: Performance metrics delivered on topology T1 by each strategy.

(a) Cumulative Regret

(b) Starvations

(c) Cumulated Throughput
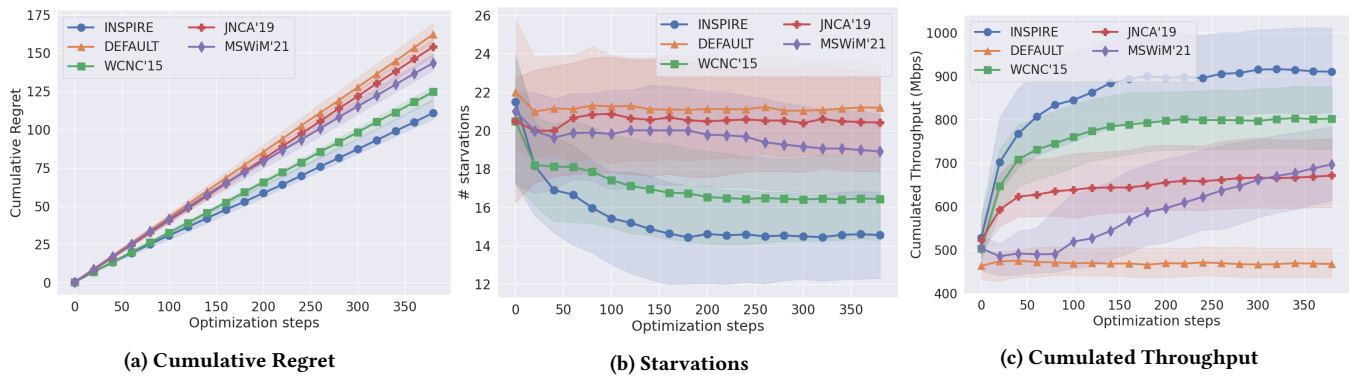
Figure 5: Performance metrics delivered on topology T2 by each strategy.

[15] Hyunjoong Lee, Hyung-Sin Kim, and Saewoong Bahk. 2021. LSR: Link-aware Spatial Reuse in IEEE 802.11ax WLANs. In *IEEE Wireless Communications and Networking Conference (WCNC'21)*. 1–6. https://doi.org/10.1109/WCNC49053.2021.9417353

[16] ns-3.31. 2020. The Network Simulator ns-3. https://www.nsnam.org/. Accessed: 2022-01-21.

[17] Shuwei Qiu, Xiaowen Chu, Yiu-Wing Leung, and Joseph Kee-Yin Ng. 2020. Joint Access Point Placement and Power-Channel-Resource-Unit Assignment for 802.11ax-Based Dense WiFi with QoS Requirements. In *IEEE Conference on Computer Communications (INFOCOM'20)*. 2569–2578. https://doi.org/10.1109/INFOCOM41043.2020.9155490

[18] Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias Seeger. 2009. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995* (2009).

[19] Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. 2012. Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting. *IEEE Transactions on Information Theory* 58, 5 (2012), 3250–3265. https://doi.org/doi:10.1109/tit.2011.2182033

[20] Francesc Wilhelmi, Sergio Barrachina-Muñoz, Boris Bellalta, Cristina Cano, Anders Jonsson, and Gergely Neu. 2019. Potential and pitfalls of Multi-Armed Bandits for decentralized Spatial Reuse in WLANs. *Journal of Network and Computer Applications* 127 (2019), 26–42. https://doi.org/10.1016/j.jnca.2018.11.006

[21] Francesc Wilhelmi, Sergio Barrachina-Muñoz, Cristina Cano, Ioannis Selinis, and Boris Bellalta. 2021. Spatial Reuse in IEEE 802.11ax WLANs. *Computer Communications* 170 (2021), 65–83. https://doi.org/10.1016/j.comcom.2021.01.028

[22] Francesc Wilhelmi, Cristina Cano, Gergely Neu, Boris Bellalta, Anders Jonsson, and Sergio Barrachina-Muñoz. 2019. Collaborative Spatial Reuse in wireless networks via selfish Multi-Armed Bandits. *Ad Hoc Networks* 88 (2019), 129–141. https://doi.org/10.1016/j.adhoc.2019.01.006

[23] Christopher K. I. Williams and Carl Edward Rasmussen. 1995. Gaussian Processes for Regression. In *Conference on Neural Information Processing Systems (NeurIPS'95)*.

[24] Jing Zhu, Xingang Guo, L. Lily Yang, W. Steven Conner, Sumit Roy, and Mousumi M. Hazra. 2004. Adapting physical carrier sensing to maximize spatial reuse in 802.11 mesh networks. *IEEE Wireless Communications and Networking Conference (WCNC'04)*. (2004).