Optimal Power Allocation and Scheduling Under Jamming Attacks

Salvatore D'Oro, Eylem Ekici, and Sergio Palazzo

***Abstract*— In this paper, we consider a jammed wireless scenario where a network operator aims to schedule users to maximize network performance while guaranteeing a minimum performance level to each user. We consider the case where no information about the position and the triggering threshold of the jammer is available. We show that the network performance maximization problem can be modeled as a finite-horizon joint power control and user scheduling problem, which is NP-hard. To find the optimal solution of the problem, we exploit dynamic programming techniques. We show that the obtained problem can be decomposed, i.e., the power control problem and the user scheduling problem can be sequentially solved at each slot. We investigate the impact of uncertainty on the achievable performance of the system and we show that such uncertainty leads to the well-known exploration-exploitation tradeoff. Due to the high complexity of the optimal solution, we introduce an approximation algorithm by exploiting state aggregation techniques. We also propose a performance-aware online greedy algorithm to provide a low-complexity sub-optimal solution to the joint power control and user scheduling problem under minimum quality-of-service requirements. The efficiency of both solutions is evaluated through extensive simulations, and our results show that the proposed solutions outperform other traditional scheduling policies.**

***Index Terms*— Scheduling, power control, jamming, QoS.**

I. Introduction

MANY resource allocation problems in the networking domain can be modeled as network utility maximization problems where cross-layer optimization techniques at both PHY and MAC layers are used to maximize network performance.

Frequently, due to the time-varying behavior of both network conditions and resource availability, the allocation of network resources has to be performed within a fixed and finite temporal window, i.e., the horizon of the optimization problem.

The resource allocation problem is further exacerbated with the inclusion of *Quality-of-Service* (QoS) constraints and third-party entities such as jamming attackers over which network operators have no control. In particular, reactive jamming attacks in wireless communications have been proven to be one of the most threatening and harmful attacks as they can completely or partially disrupt ongoing communications [1]. Reactive jammers continuously monitor one or multiple channels, searching for ongoing transmission activities by means of *energy detectors* [2]. Then, only when a transmission is detected, the jammer starts its attack. The attacker node is able to transmit jamming signals that interfere with those transmitted by legitimate users, thus causing a drop in the *Signal-to-Interference-plus-Noise Ratio* (SINR) of such users.

Reactive jamming attacks reach a high jamming efficiency and can even improve the energy-efficiency of the jammer in several application scenarios [2], [3]. Also, they can easily and efficiently be implemented on COTS hardware such as USRP radios [1], [4], [5]. But, more importantly, reactive jamming attacks are harder to detect due to the attack model, which allows jamming signal to be hidden behind transmission activities performed by legitimate users [4], [6], [7]. In this paper, we adopt this more general and efficient attack model. For a more detailed discussion on jamming and anti-jamming techniques we refer the reader to [8]–[11] and the references therein.

Main challenges in dealing with reactive jamming attacks are 1) to detect the presence of a jammer and 2) to develop proper anti-jamming mechanisms to avoid the disrupting actions of the jammer. It has been shown that detecting the presence of a jammer requires complex or time-consuming operations, such as statistical estimations and continuous packet monitoring [6], [7], [12]. Moreover, customized solutions to counteract jamming attacks are non-trivial as the exact behavior of the jammer is generally unknown and unpredictable.

It is clear that under such hostile conditions, network performance can be significantly reduced. Without proper anti-jamming techniques, it is not easy to optimize network performance within a finite-horizon. Most anti-jamming techniques need either a priori information on jammer's behavior and channel conditions or a large number of temporal slots to become effective. Also, guaranteeing a minimum performance level to all users in the network can be even more challenging, especially if service guarantees must be met in a limited temporal window. As an example, a capacity maximization problem is investigated in [13] where authors aim to maximize the overall number of successful transmissions of a jammed network consisting of several users by exploiting distributed no-regret learning techniques. However, this approach does not guarantee a minimum performance level to network users.

Manuscript received April 23, 2016; revised October 15, 2016; accepted October 23, 2016; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor X. Liu.

S. D'Oro and S. Palazzo are with the University of Catania, 95125 Catania, Italy (e-mail: sdoro@dieei.unict.it; spalazzo@dieei.unict.it).

E. Ekici is with The Ohio State University, Columbus, OH 43210 USA (e-mail: ekici.2@osu.edu).

Digital Object Identifier 10.1109/TNET.2016.2622002

Also, to find the optimal solution, the iterative learning algorithm proposed in [13] requires a large number of iterations, a condition that cannot always be satisfied.

Solutions to a variety of QoS provisioning problems in unjammed scenarios have been proposed in the literature. As an example, in [14] and [15] authors propose optimal solutions for the capacity maximization problem under minimum performance guarantee constraints. However, such solutions do not consider possible jamming attack, and it is shown that the proposed approaches converge to the optimal solution when the horizon tends to infinity. Therefore, it is hard to apply these solutions to the problem we address in this work.

In this paper, we consider a multi-user multi-carrier time-slotted cellular network where the network operator has to schedule network users to maximize achievable network performance while guaranteeing a minimum performance level to each user. We also consider the worst case scenario where a reactive jammer is deployed within the base station (BS) coverage range. Each node is affected by its jamming activity and the transmission power of each user cannot exceed a given maximum threshold. We further assume that channel conditions vary in time according to the block-fading model.

We show that the above problem can be modeled as a finite-horizon joint power control and user scheduling problem. Also, we prove that finding an optimal solution is NP-hard. We formulate the problem by exploiting techniques from Dynamic Programming (DP). The DP formulation allows us to show that the joint power control and user scheduling is a decomposable problem. That is, at each optimization step we can sequentially solve the power control and the user scheduling problems. We show that, under some conditions, it is possible to identify the optimal power control policy, i.e., conservative, exploratory or aggressive. To avoid the curse of dimensionality of the DP approach, we exploit state aggregation techniques to propose an approximated solution and study its complexity.

We also propose a Performance-aware Online Greedy Algorithm (POGA) to provide a low-complexity sub-optimal solution for the joint power control and scheduling problem under minimum QoS requirements. Accordingly, we present an algorithmic implementation of POGA and we show that its computational complexity is polynomial in the number of users in the system.

We evaluate the achievable performance of both the approximated solution and POGA through extensive simulation results. Also, we investigate the impact of jammer's behavior and position on achievable system's performance. Finally, we compare the performance of the above proposed solutions with those achieved by traditional scheduling policies. We show that both our solutions outperform other considered approaches.

The rest of the paper is organized as follows. Related work is presented in Section II. In Section III, we first illustrate the system model and, then, we formulate the joint power control and user scheduling problem. In Section IV, we propose a decomposable DP formulation of the original problem. A discretized version of the original problem is proposed in Section V, and POGA is presented in Section VI. In Section VII, we evaluate the achievable performance of the proposed solutions through numerical simulations. Finally, Section VIII concludes the paper.

II. Related Work

Providing jamming-proof communications is an interesting topic and several solutions have been proposed in the literature. For example, spread-spectrum techniques are commonly used to avoid the jammer and its attacks [16]–[19]. However, to be effective, such techniques need to either share or establish a secret among network users. For this reason, such techniques cannot be applied in all wireless scenarios [20], [21].

In [20], [22], and [23] a *trigger-identification* approach is presented. First, nodes whose transmission trigger the reactive jammer, i.e., the triggering nodes, are identified. Then, optimal routing paths are established, which exclude triggering nodes from the routing process. However, such solutions are designed for large sensor networks where multi-hop communication is feasible. Such solutions fail in scenarios where the whole network is under attack and any node can potentially be a trigger node.

To detect transmission activities, the jammer has to first sense ongoing communications. Then, when an activity is detected the jammer starts its attack. Thus, there is a delay, i.e., activation time, between the detection and attack phases. All bits transmitted during the activation time escape jamming and can be exploited to establish communications under reactive jamming attacks [21].

Another approach is proposed in [24]–[26] where radio silences are exploited to establish secure communications. As the reactive jammer does not attack when no transmission activities are performed by users, it is possible to encode the information to be transmitted in silences between consecutive packets. Accordingly, although transmitted packets can be completely damaged, the radio silence between consecutive packets can be modulated to convey data by mapping bit sequences and silence period durations. However, the above solutions neither aim to maximize network performance nor provide any minimum QoS service level to network users.

Power control has been recently proposed to overcome possible reactive jamming attacks [27], [28]. The intuition is that by controlling the transmission power of users, it is possible to let the received power at the jammer remain under the triggering threshold [28]. In [27], authors consider a single sender-receiver pair and propose a joint *frequency hopping* (FH) and power control scheme to avoid reactive jamming attacks. The proposed solution consists in selecting user's transmission power such that the sender's power at the receiver side is higher than that of the jammer. However, the latter approach fails when dealing with systems with multiple senders and where FH is not possible such as the one we consider in this paper.

III. System Model

A. Network Model

We consider a multi-carrier slotted wireless system where a set of users access the network and communicates with a shared BS through several non-interfering channels. Let $\mathcal{N}$ be

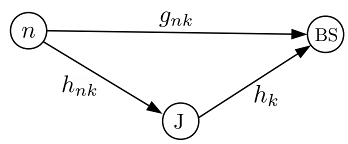

Fig. 1. Channel model.

TABLE I
NOTATION

Variable	Description
$\mathcal{N}, \mathcal{K}, \mathcal{H}$	Sets of users, channels and slots
P	Maximum transmission power level of users
P_J, P_{th}	Maximum transmission power level and triggering threshold of the jammer
g_{nk}	Channel gain between user n and the BS on channel k
H	Horizon duration
α_{nk}	Triggering function for n transmitting on k
$\theta_{nk}(j)$	Allocation indicator for n on k at slot j
$p_{nk}(j)$	Power control variable for n on k at slot j
$p^m_{nk}(j)$	Maximum transmission power of n on k that has not triggered the jammer up to slot j
$p^M_{nk}(j)$	Minimum transmission power of n on k that has triggered the jammer up to slot j
$\pi_{nk}(j)$	The history of n on k up to slot j
$E_\alpha\{R_{nk}(p)\|\pi_{nk}(j)\}$	Expected achievable rate of n on k for a given $\pi_{nk}(j)$
$\Pi_{nk}(j)$	Feasible transmission power set given $\pi_{nk}(j)$
$\Delta_{nk}(j)$	Lebesque measure of $\Pi_{nk}(j)$
$p^{\mathrm{OPT}}_{nk}(j)$	Optimal transmission power level for n on k at slot j
γ_i	Step-size used in (22)
$\rho_n(j)$	Residual performance variable for n at slot j
ξ_p	Power quantization step

the set of users and $\mathcal{K}$ be the set of the channels accessed by these users. We assume that users are power-constrained and are equipped with single-antenna transceivers. Their instantaneous transmission power is limited to a maximum power level P and they can transmit on at most one channel at a time. Also, we assume that a given slot on a given channel can be assigned to only one user. In our paper, we focus on the uplink scheduling problem.

In our model, we assume *Additive White Gaussian Noise* (AWGN) channels with channel gains defined as i.i.d. random variables. As shown in Fig. 1, let h_{nk} be the channel gain coefficients between user n and the jammer on channel k, while h_k indicates the channel gain coefficient between the jammer and the BS on channel k. Finally, g_{nk} indicates the channel gain between user n and the BS on channel k. Relevant parameters and variables are summarized in Table I.

We assume block fading, that is, channel gain coefficients remain constant for a fixed number of slots before they change. Let H be the number of slots where channel gain coefficients remain constant. Therefore, the optimal scheduling problem has to be periodically performed every H slots. We refer to such time period as the *scheduling cycle*. Accordingly, H is the finite-horizon of the optimization problem.[1]

B. Attack Model

We assume that a malicious user, i.e., the jammer, aims to disrupt ongoing communications between legitimate users and the BS. More specifically, in our model we consider a reactive jamming attacker where the malicious user continuously monitors all channels in $\mathcal{K}$ searching for transmission activities and jams only those channels where the received signal power is higher than a given threshold. In our study, we assume that when a transmission activity is detected in a given slot, the jammer emits a jamming signal whose duration is equal to the slot duration.[2] To switch from the monitoring to the transmission phases causes a delay also referred to as the activation time. Such delay has been shown to be small [1], and are negligible compared to the duration of each slot. Thus, we assume that the jammer is able to instantaneously switch between RX and TX front-ends.

Let P_{th} and P_J denote the triggering threshold and the transmission power of the jammer, respectively. We assume that the jammer's attack strategy is independent of channels and users; that is, the values of P_{th} and P_J are constant and equal for all $k \in \mathcal{K}$ and $n \in \mathcal{N}$. However, it is worth noting that the approach proposed in this paper also applies to the more general case where the values of P_{th} and P_J independently vary accross users and channels. That is, our model can be also applied to the case where the triggering threshold and the transmission power of the jammer for user $n \in \mathcal{N}$ and channel $k \in \mathcal{K}$ are P^{th}_{nk} and P^J_{nk}, respectively.

To model the triggering mechanism that regulates the jammer, we define the *triggering function* $\alpha_{nk}(p) : \mathbb{R} \rightarrow \{0,1\}$ for user n transmitting on channel k. More specifically,

$$\alpha_{nk}(p) = \begin{cases} 1 & \text{if } ph_{nk} \geq P_{th} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where p is the transmission power for n on channel k. Intuitively, according to eq. (1), an attack is performed only when the received power at the jammer side (ph_{nk}) is greater than or equal to the triggering threshold P_{th}. Clearly, $\alpha_{nk}(0) = 0$. We consider the worst case scenario where $\alpha_{nk}(P) = 1$, i.e., all nodes are jammed by the jammer when they transmit with full power. However, the more general case where $\alpha_{nk}(P) = 0$ can be similarly treated by exploiting the same techniques presented in this paper.

Let us consider the generic channel $k \in \mathcal{K}$ and let p^m and p^M be two feasible transmission power levels for a given user $n \in \mathcal{N}$ such that $0 \leq p^m < p^M \leq P$. We further assume that $\alpha_{nk}(p^m) = 0$ and $\alpha_{nk}(p^M) = 1$, that is, the system knows that when user n transmits with power p^m on channel k it does not trigger the jammer, while transmitting with power p^M on the same channel activates the jammer and consequently causes a decrease in the SINR.[3] Therefore, the probability of triggering the jammer when transmitting with power p given the values

[1] Our model also applies to the case of a mobile jammer attacks the network. Since the jammer wants to be undetectable and unpredictable, it moves and changes its position every H slots. It follows that channel gain coefficients vary in time and the network operator has to periodically find the optimal scheduling policy every H slots.

[2] To consider the case where the duration T_J of the jamming signal is shorter than the slot duration T_p, the model here proposed has to be slightly modified. Specifically, the expected rate of the system has to be rewritten as $\tilde{R}(p) = \left(1 - \frac{T_J}{T_p}\right) \log\left(1 + \frac{gp}{\sigma^2}\right) + \frac{T_J}{T_p} E_\alpha\{R(p)|\pi(j)\}$, where $E_\alpha\{R(p)|\pi(j)\}$ is defined in eq. (7). However, it can be easily shown that the results presented in this work also apply to this specific case.

[3] The existence of both p^m and p^M is always guaranteed by our assumptions and in Section IV we provide proper mechanisms to identify the values of both parameters.

of both p^m and p^M can be written as

$$\mathrm{F}_{nk}(p) = \Pr\{ph_{nk} \geq P_{th}|\pi_{nk}\} = \Pr\left\{\frac{P_{th}}{h_{nk}} \leq p|\pi_{nk}\right\} \quad (2)$$

where $\pi_{nk} = \left(p^m, p^M\right)$ is a tuple that represents the *history* (or knowledge) of the system. As shown in eq. (2), the probability of triggering the jammer depends on the ratio P_{th}/h_{nk} between the triggering threshold of the jammer and the channel gain coefficient between the jammer and the transmitter. Although in reality the position and the triggering threshold of the jammer are unknown and, thus, the exact value of the ratio P_{th}/h_{nk} is unknown to the network operator, the information contained in the history $\pi_{nk} = \left(p^m, p^M\right)$ is still available. Note that given π_{nk}, the exact value of the ratio P_{th}/h_{nk} can be any value in the range $(p^m, p^M]$. Therefore, to model such uncertainty on the knowledge of such parameters we assume that the ratio P_{th}/h_{nk} is modeled as a *uniformly distributed* random variable[4] over the interval $(p^m, p^M]$. Accordingly, we rewrite eq. (2) as follows:

$$\mathrm{F}_{nk}(p) = \begin{cases} 0 & \text{if } p \leq p^m \\ \dfrac{p - p^m}{p^M - p^m} & \text{if } p^m < p < p^M \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

C. Problem Formulation

As a consequence of the AWGN assumption, for any given user $n \in \mathcal{N}$ scheduled on channel $k \in \mathcal{K}$, the SINR received at the BS side is

$$\mathrm{SNR}_{nk}(p) = \frac{g_{nk}p}{\sigma^2 + \alpha_{nk}(p)h_k P_J} \quad (4)$$

where p is the transmission power and σ^2 is the variance of the AWGN which we assume to be equal for all $n \in \mathcal{N}$ and $k \in \mathcal{K}$. We assume that the channel gain coefficients g_{nk} can be accurately estimated. Since the transmission power of each user is chosen by the centralized entity and all g_{nk} are known, the product $h_k P_J$ can be accurately obtained by the BS by comparing the expected received signal with the actual received signal.

Let $\mathcal{H} = \{1, 2, \ldots, H\}$ be the set of slots in a scheduling cycle. Accordingly, we define the achievable rate $R_{nk}(p)$ at slot $j \in \mathcal{H}$ as follows:

$$R_{nk}(p(j)) = \log\left(1 + \frac{g_{nk}p(j)}{\sigma^2 + \alpha_{nk}(p(j))h_k P_J}\right) \quad (5)$$

Let $\theta_{nk}(j)$ and $p_{nk}(j)$ be the *allocation indicator* and *power control variable*, respectively. If user n is allocated to channel k at slot j, the allocation indicator is set to one, i.e., $\theta_{nk}(j) = 1$, otherwise it is set to zero, i.e., $\theta_{nk}(j) = 0$. Similarly, $p_{nk}(j)$ denotes the transmission power which must take values in the range $[0, P]$ due to the power constraint we have discussed in Section III-A. Let $\boldsymbol{\theta}(j) = (\theta_{nk}(j))_{n,k}$ and $\boldsymbol{p}(j) = (p_{nk}(j))_{n,k}$ be the *scheduling policy* and *power control policy* at slot j, respectively. Clearly, if $\theta_{nk}(j) = 0$, then we set $p_{nk}(j) = 0$. Also, for any $n \in \mathcal{N}$ and $k \in \mathcal{K}$, let $\pi_{nk}(j) = \left(p^m_{nk}(j), p^M_{nk}(j)\right)$ be the history up to slot j, where

$$\begin{aligned} p^m_{nk}(j) &= \max\ \{p_{nk}(l) \in \tilde{\mathbf{p}}(j) \colon \alpha_{nk}\left(p_{nk}(l)\right) = 0,\ l < j\} \\ p^M_{nk}(j) &= \min\ \{p_{nk}(l) \in \tilde{\mathbf{p}}(j) \colon \alpha_{nk}\left(p_{nk}(l)\right) = 1,\ l < j\} \end{aligned} \quad (6)$$

and $\tilde{\mathbf{p}}(j) = (p_{nk}(l))_{n,k,l<j}$ is the set of all the power control decisions taken up to slot j. By assumption, we have $\pi_{nk}(1) = (0, P)$ for all $n \in \mathcal{N}$ and $k \in \mathcal{K}$. Intuitively, at each slot the system keeps track of the reaction of the jammer to different policies chosen in the past.

To evaluate (5), we need to know the triggering function exactly. Unfortunately, the reaction of the jammer, i.e., the outcome of the triggering function $\alpha_{nk}(p(j))$, is known only at the end of each slot. Accordingly, from eqs. (1), (3) and (5), the expected achievable rate for user n on channel k is

$$\begin{aligned} &E_\alpha\left\{R_{nk}(p)|\pi_{nk}(j)\right\} \\ &\quad = \frac{p - p^m_{nk}(j)}{p^M_{nk}(j) - p^m_{nk}(j)} \log\left(1 + \frac{g_{nk}p}{\sigma^2 + h_k P_J}\right) \\ &\qquad + \left(1 - \frac{p - p^m_{nk}(j)}{p^M_{nk}(j) - p^m_{nk}(j)}\right) \log\left(1 + \frac{g_{nk}p}{\sigma^2}\right) \end{aligned} \quad (7)$$

where the expectation is taken w.r.t. the output of the triggering function given that the history of the system at slot j is $\pi_{nk}(j)$.

We define the following finite-horizon joint power control and scheduling problem with minimum performance guarantee under jamming attacks (Problem **A**).

$$(\mathbf{A}): \max_{\boldsymbol{\theta}, \mathbf{p}}\ \mathrm{E}_{\boldsymbol{\alpha}}\left\{\sum_{j\in\mathcal{H}}\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} \theta_{nk}(j) R_{nk}(p_{nk}(j))\right\}$$

$$\text{s.t.} \sum_{k\in\mathcal{K}} \theta_{nk}(j) \leq 1, \quad \forall n \in \mathcal{N},\ j \in \mathcal{H} \quad (8)$$

$$\sum_{n\in\mathcal{N}} \theta_{nk}(j) \leq 1, \quad \forall k \in \mathcal{K},\ j \in \mathcal{H} \quad (9)$$

$$\mathrm{E}_{\boldsymbol{\alpha}}\left\{\sum_{j\in\mathcal{H}}\sum_{k\in\mathcal{K}} R_{nk}(p_{nk}(j))\right\} \geq R^*_n, \quad \forall n \in \mathcal{N} \quad (10)$$

$$\theta_{nk}(j) \in \{0, 1\}, \quad \forall n \in \mathcal{N}, \quad \forall k \in \mathcal{K},\ j \in \mathcal{H} \quad (11)$$

$$p_{nk}(j) \in [0, P], \quad \forall n \in \mathcal{N}, \quad \forall k \in \mathcal{K},\ j \in \mathcal{H} \quad (12)$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}(1), \boldsymbol{\theta}(2), \ldots, \boldsymbol{\theta}(H))$; $\boldsymbol{p} = (\boldsymbol{p}(1), \boldsymbol{p}(2), \ldots, \boldsymbol{p}(H))$ are the decision variables; $\boldsymbol{\alpha} = (\alpha_{nk}(p_{nk}(j)))_{n,k,j}$ is the set of all outcomes of the triggering function according to the actual power control policy $\boldsymbol{p}(j)$ at slot j; and R^*_n is the minimum rate requirement of user n. In Problem (**A**), constraint (8) guarantees that at any given time a user can be allocated to only one slot. On the other hand, constraint (9) ensures that only one user can be allocated on a given slot, thus avoiding possible collisions and/or interferences among users. The minimum performance constraint is imposed by the non-linear constraint (10) which ensures that the expectation of the rate achieved by any user at the end of the optimization horizon is higher than or equal to the performance requirement R^*_n. Finally, constraints (11) and (12) guarantee the feasibility of the decision variables.

[4]Note that the uniform distribution assumption is well-suited to capture this worst-case scenario where the actual value of the triggering threshold is unknown to the network operator and all values in the range $(p^m, p^M]$ are equiprobable.

D. Hardness of the Problem

$(\mathbf{A})$ is a non-linear (concave) combinatorial optimization problem with both discrete (i.e., $\theta_{nk}(j)$) and continuous (i.e., $p_{nk}(j)$) decision variables. In this section, we prove that $(\mathbf{A})$ is NP-hard by showing that the *Multiprocessor Scheduling* is polynomially reducible to a subproblem of $(\mathbf{A})$. The multiprocessor scheduling is known to be NP-complete [29] and it is stated as follows: given m processors, a deadline D and a set $\mathcal{X}$ of jobs where each job $x_n \in \mathcal{X}$ has length l_n, is there a m-processor scheduling that schedules all jobs and meets the overall deadline D?

Theorem 1 (NP-Hardness): Problem $(\mathbf{A})$ is NP-hard.

Proof: Let $D = H$, $\mathcal{X} = \mathcal{N}$, $x_n = n$, $m = |\mathcal{K}|$. Let us assume $P_{th} = 0$, i.e., any user transmission triggers the jammer and $\alpha_{nk}(p) = 1$ for all $p > 0$, $n \in \mathcal{N}$ and $k \in \mathcal{K}$. Let $p^*_{nk}(j)$ be the optimal transmission power level. In this specific instance of Problem $(\mathbf{A})$, it is straightforward that $p^*_{nk}(j) = P$ for all $n \in \mathcal{N}$, $k \in \mathcal{K}$ and $j \in \mathcal{H}$. Now let us assume that the minimum performance requirement R^*_n is such that $R^*_n < \min_{k\in\mathcal{K}} \{R_{nk}(P)\}$. Accordingly, we define the length of job x_n as $l_n = \frac{R^*_n}{\max_{k\in\mathcal{K}}\{R_{nk}(P)\}} T$, where $T = 1$ is the duration of a single slot. The above subproblem of $(\mathbf{A})$ requires us to find the rate-optimal scheduling of users (i.e., the jobs) among the available channels (i.e., the processors) while satisfying their minimum performance requirement (i.e., the job's length) within the considered horizon (i.e., the deadline). This way, we have built a reduction of the multiprocessor scheduling to an instance of a subproblem of $(\mathbf{A})$. Thus, the above instance is NP-complete by reduction [29]. Also, since this reduction can be made in polynomial time, it follows that $(\mathbf{A})$ is NP-hard and, unless P=NP, it cannot be solved in polynomial-time. ■

IV. Optimal Solution

In Theorem 1, we have shown that $(\mathbf{A})$ is NP-hard. However, how to find an optimal solution still remains unsolved. In this section, we show that $(\mathbf{A})$ is a dynamic problem. Moreover, we show that at each slot the joint power control and user scheduling problem is decomposable. That is, at each slot it is possible to separately solve the power control and the user scheduling problems.

A. Problem Dynamism and Decomposability

In previous sections, we have shown that to maximize the achievable performance of the system, i.e., the overall transmission rate, the scheduler has to evaluate the expected achievable rate for all users at each slot. Eq. (7) shows that the expected achievable rate for a given user on a given channel depends on the history parameter $\pi_{nk}(j)$. In turn, eq. (6) shows that the history $\pi_{nk}(j)$ depends on actions taken in the past. Therefore, the knowledge and the state of the system dynamically evolve at each slot. Intuitively, a DP approach is well-suited to model and solve the considered problem.

Another important issue is whether or not the problem is decomposable. To maximize the overall achievable rate of network, the scheduling problem requires us to first estimate the achievable performance of each user. On the contrary, as shown in eq. (7), to maximize the single-slot expected rate, the power control problem only needs to know the history parameter $\pi_{nk}(j)$ for all n and k. Recall that $\pi_{nk}(j)$ does not depend on the scheduling policy at the actual slot, but only depends on the scheduling decisions taken in the past. Hence, at each slot, the power control problem can be solved independently of the actual scheduling policy. However, the user scheduling problem needs the output of the power control problem. Therefore, as shown in Fig. 2, we first solve the power control problem and find the optimal transmission power level for each user on each channel. Then, we solve the scheduling problem.

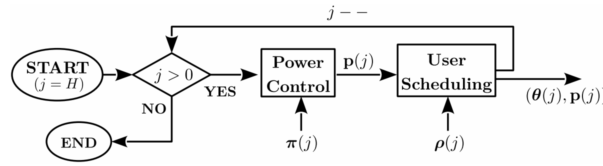

Fig. 2. Structure of the dynamic programming problem.

It is of extreme importance to remark that the decomposition of the problem does not invalidate the optimality of the solution. As we have already pointed out above, to find the optimal power control policy does not depend on the actual scheduling policy but only on the decisions taken in the past. The converse is not true as the scheduling problem preliminarily requires to calculate the achievable rate of all users, which implies that the power control problem has to be solved first. Accordingly, the decomposition of the problem still provides the optimal solution to the joint power control and user scheduling problem.

Parameters shown in Fig. 2 and their importance will be described in the two following sections.

B. Optimal Power Control

To solve the single-slot power control problem, we must find the optimal transmission power level for all users on each channel. Constraints (8) and (9) imply that no collision may occur and we can separately solve the power control problem for any individual user.

For each slot j and channel $k \in \mathcal{K}$, we define the single-user power control problem (Problem $\mathbf{B}$) as follows:

$$(\mathbf{B})\colon \max\left\{\max_{p\in\Pi_{nk}(j)} \mathrm{E}_\alpha\left\{R_{nk}(p)|\pi_{nk}(j)\right\}, R_{nk}(P)\right\}$$

where $\Pi_{nk}(j) = [p^m_{nk}(j), p^M_{nk}(j)]$.

The challenges of $(\mathbf{B})$ are twofold as: i) the problem evolves according to past choices; ii) the reaction of the jammer is observed only at the end of the slot. Therefore, we must consider all possible realizations of the triggering function $\alpha_{nk}(p)$. Let $p^*_{nk}(j)$ be defined as follows:

$$p^*_{nk}(j) = \arg\max_{p\in\Pi_{nk}(j)} E_\alpha\left\{R_{nk}(p)|\pi_{nk}(j)\right\} \tag{13}$$

From eq. (7), eq. (13) can be rewritten as follows:

$$p^*_{nk}(j) = \arg\max_{p\in\Pi_{nk}(j)} \frac{p - p^m_{nk}(j)}{\Delta_{nk}(j)} \log\left(1 + \frac{g_{nk}p}{\sigma^2 + h_k P_J}\right) + \left(1 - \frac{p - p^m_{nk}(j)}{\Delta_{nk}(j)}\right)\log\left(1 + \frac{g_{nk}p}{\sigma^2}\right)$$

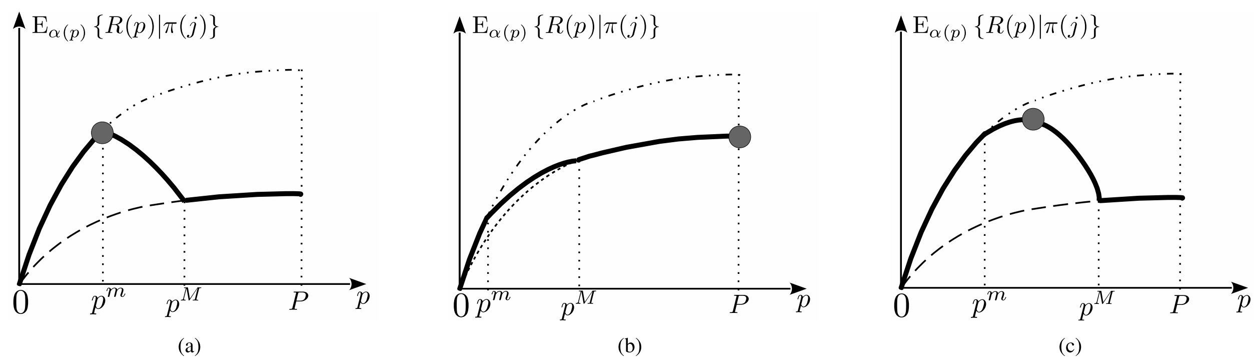

Fig. 3. Comparison between different realizations of the expected achievable rate (solid lines) as a function of the transmission power p. Gray dots represent optimal transmission power policies: (a) conservatory; (b) aggressive; and (c) exploratory. Dashed-dotted lines show the achievable rate when no attacks are performed ($\alpha(p) = 0$). Dashed lines show the achievable rate when the user is under attack ($\alpha(p) = 1$).

where $\Delta_{nk}(j) = p^M_{nk}(j) - p^m_{nk}(j)$ is the *Lebesgue measure* of $\Pi_{nk}(j)$. In Proposition 1, we show that (**B**) admits a unique optimal solution.

*Proposition 1: Problem (**B**) always admits a unique solution $p^{\mathrm{OPT}}_{nk}(j)$ defined as*

$$p^{\mathrm{OPT}}_{nk}(j) = \begin{cases} p^*_{nk}(j) & \textit{if } \mathrm{E}_\alpha\{R_{nk}(p^*_{nk}(j))\} \geq R_{nk}(P) \\ P & \textit{otherwise} \end{cases} \quad (14)$$

Proof: The proof consists in showing that the function $\mathrm{E}_{\alpha_{nk}(p)}\{R_{nk}(p)|\pi_{nk}(j)\}$ is strictly concave and, therefore, it admits a unique maximizer. Let $y = h_k P_J$. By looking at the second order derivative of $\mathrm{E}_{\alpha_{nk}(p)}\{R_{nk}(p)|\pi_{nk}(j)\}$, it can be shown that strict concavity holds if

$$\frac{gy(p-p^m)[y+2(\sigma^2+gp)]}{\sigma^2+gp+y} < g(p^M-p^m)(\sigma^2+gp+y) + 2y(\sigma^2+gp) \quad (15)$$

where, for the sake of simplicity, we have omitted all subscripts and the slot index. However, if we remove the negative term in the left hand side of eq. (15), i.e., we consider a maximization of the first term in eq. (15), and do some algebra, it follows that

$$\frac{gy(p)[y+2(\sigma^2+gp)]}{\sigma^2+gp+y} < g(p^M-p^m)(\sigma^2+gp+y) + 2y(\sigma^2+gp)$$

yields to $y^2 < 2y^2$, which always holds if $y \neq 0$. Furthermore, it is straightforward to prove that also $y = 0$, i.e., no jamming attack is performed by the attacker, leads to strictly concavity. To prove the second part of the proposition, it suffices to note that $\mathrm{E}_{\alpha_{nk}(p)}\{R_{nk}(p^*_{nk}(j))|\pi_{nk}(j)\} > R_{nk}(P)$ implies that the optimal transmission power level is $p^{\mathrm{OPT}}_{nk}(j) = p^*_{nk}(j)$. Instead, if $\mathrm{E}_{\alpha_{nk}(p)}\{R_{nk}(p^*_{nk}(j))|\pi_{nk}(j)\} = R_{nk}(P)$, both $p^*_{nk}(j)$ and P are optimal. Without losing in generality, if both policies are optimal, then we assume that $p^{\mathrm{OPT}}_{nk}(j) = p^*_{nk}(j)$. Finally, if $\mathrm{E}_{\alpha_{nk}(p)}\{R_{nk}(p^*_{nk}(j))|\pi_{nk}(j)\} < R_{nk}(P)$, it follows that $p^{\mathrm{OPT}}_{nk}(j) = P$. ■

Proposition 1 suggests that there are some scenarios where transmitting with the highest power, i.e., P, and triggering the jammer is the optimal power control solution. For example, if the jammer is in proximity of a user but far away from the BS, it is reasonable to assume that even low transmission power levels can trigger the jammer. Therefore, to transmit with a low power level to avoid the jammer can be inefficient. It follows that transmitting with the highest power P and triggering the jammer can be the only optimal policy. Clearly, $p^{\mathrm{OPT}}_{nk}(j)$ depends on the values of several parameters such as P_J, channel gain coefficients and $\pi_{nk}(j)$. Therefore, it is hard to know a priori the optimal policy chosen by the scheduler. However, to better understand the dynamics that regulate the power control problem, we define three different policies:

- *conservative*: is a policy where $p^{\mathrm{OPT}}_{nk}(j) = p^m_{nk}(j)$. Recall that $\alpha_{nk}(p^m_{nk}(j)) = 0$. Therefore, under such a policy (Fig. 3), the scheduler chooses to avoid the jammer by choosing a safe strategy which ensures that the jammer will not be triggered;
- *aggressive*: is a policy where the optimal power control policy consists in transmitting with full power, i.e., $p^{\mathrm{OPT}}_{nk}(j) = P$. In general, such a policy is optimal when jamming activities does not affect the performance of the system significantly (Fig. 3);

$$\delta^C_{nk}(\pi_{nk}(j)) = \frac{\sigma^2 + p^m_{nk}(j)g_{nk}}{g_{nk}} \cdot \left[\log\left(1+\frac{h_kP_J}{\sigma^2}\right) - \log\left(1+\frac{h_kP_J}{\sigma^2+p^m_{nk}(j)g_{nk}}\right)\right] \quad (16)$$

$$\hat{\delta}^C_{nk}(\pi_{nk}(j)) = p^M_{nk}(j) - P\sigma^2/(\sigma^2+h_kP_J) \quad (17)$$

$$\delta^A_{nk}(\pi_{nk}(j)) = \frac{\sigma^2 + p^M_{nk}(j)g_{nk} + h_kP_J}{g_{nk}} \cdot \left[\log\left(1+\frac{h_kP_J}{\sigma^2}\right) - \log\left(1+\frac{h_kP_J}{\sigma^2+p^M_{nk}(j)g_{nk}}\right)\right] \quad (18)$$

$$\hat{\delta}^A_{nk}(\pi_{nk}(j)) = (p^*_{nk}(j) - p^m_{nk}(j)) \cdot \frac{\log\left(1+\frac{h_kP_J}{\sigma^2}\right) - \log\left(1+\frac{h_kP_J}{\sigma^2+g_{nk}p^*_{nk}(j)}\right)}{\log\left(1+\frac{h_kP_J}{\sigma^2}\right) - \log\left(\frac{\sigma^2+h_kP_J+g_{nk}P}{\sigma^2+g_{nk}p^*_{nk}(j)}\right)} \quad (19)$$

- *exploratory*: in this case, a transmitting power $p \in (p_{nk}^m(j), p_{nk}^M(j))$ is optimal (Fig. 3). The scheduler decides to take the risk by exploring new transmitting power levels to which jammer's reaction is unknown.

The above policies are in line with the vast body of literature on the *exploration-exploitation* trade-off [30], where the decision maker has to choose between gathering new information by exploring new actions, or exploit the already explored actions whose system's reactions are already known. When an exploratory policy is chosen, the scheduler takes the risk and explores new power control policies, even though such decision could trigger the jammer. When conservative and aggressive policies are chosen, the reaction of the jammer, together with the achievable performance under such policies, can be exactly predicted. Therefore, conservative and aggressive policies are exploitation decisions. In the rest of the paper, we will refer to the exploration of new power control policies as the learning dynamics (or learning process) of the system.

Since the system is able to detect the presence or the absence of a jamming attack only when a slot ends, the achievable rate and the jammer's reaction to a given policy are known only at the end of each slot. Therefore, the history of the system is updated at the beginning of each slot according to the reaction of the jammer to decisions taken in the previous slot. Note that the history of the system is updated only when a user is scheduled on a given channel. That is, if $\sum_{k\in\mathcal{K}} \theta_{nk}(j) = 0$ for a given $j \in \mathcal{H}$ and $n \in \mathcal{N}$, we have that $\pi_{nk}(j+1) = \pi_{nk}(j)$. Instead, when a user is scheduled on a given channel and $j > 1$, the history of the system is updated according to the history update dynamics in eq. (20).

$$\pi_{nk}(j+1) = \begin{cases} \left(p_{nk}^m(j), p_{nk}^{OPT}(j)\right) & \text{if } \alpha_{nk}(p_{nk}^{OPT}(j)) = 1 \\ & \wedge p_{nk}^{OPT}(j) \neq P \\ \left(p_{nk}^m(j), p_{nk}^M(j)\right) & \text{if } \alpha_{nk}(p_{nk}^{OPT}(j)) = 1 \\ & \wedge p_{nk}^{OPT}(j) = P \\ \left(p_{nk}^{OPT}(j), p_{nk}^M(j)\right) & \text{otherwise} \end{cases} \quad (20)$$

where "$\wedge$" is the logical AND operator and we recall that $\pi_{nk}(1) = (0, P)$.

For any $\pi_{nk}(j)$, let $\Delta_{nk}(\pi_{nk}(j)) = p_{nk}^M(j) - p_{nk}^m(j)$ denote the *measure* of the interval $[p_{nk}^m(j), p_{nk}^M(j)]$ generated by the history $\pi_{nk}(j)$. Furthermore, let $\delta_{nk}^C(\pi_{nk}(j))$, $\hat{\delta}_{nk}^C(\pi_{nk}(j))$, $\delta_{nk}^A(\pi_{nk}(j))$ and $\hat{\delta}_{nk}^A(\pi_{nk}(j))$ be defined as in eqs. (16)-(19). In Proposition 2, we illustrate how system's learning dynamics impact the choice of the optimal power control policy.

Proposition 2: For each $n \in \mathcal{N}$, $k \in \mathcal{K}$ and $j, l \in \mathcal{H}$, let $\delta_{nk}^C(j) = \min\left\{\delta_{nk}^C(\pi_{nk}(j)), \hat{\delta}_{nk}^C(\pi_{nk}(j))\right\}$, and $\delta_{nk}^A(j) = \min\left\{\delta_{nk}^A(\pi_{nk}(j)), \hat{\delta}_{nk}^A(\pi_{nk}(j))\right\}$. We have that 1) if $\Delta_{nk}(j) \leq \delta_{nk}^C(j)$ a conservative policy is optimal, 2) if $\Delta_{nk}(j) \geq \delta_{nk}^A(j)$ an aggressive policy is optimal, and 3) if either conservative or aggressive policies are optimal at slot j, these policies will be optimal for any $l > j$.

Proof: To prove the first part of this proposition, we look at the first order derivative, say $r_{nk}(p)$, of the function $\mathrm{E}_{\alpha_{nk}(p)}\left\{R_{nk}(p)\right\}$ w.r.t. the variable $p \in [p_{nk}^m(j), p_{nk}^M(j)]$. For a conservative strategy to be optimal, i.e., $p_{nk}^{OPT}(j) = p_{nk}^m(j)$, it must hold that i) $r_{nk}(p_{nk}^m(j)) \leq 0$; and ii) $R_{nk}(p_{nk}^m(j)) \geq R_{nk}(P)$. It can be easily shown that i) holds if $\Delta_{nk}(\pi_{nk}(j)) \leq \delta_{nk}^C(\pi_{nk}(j))$, and ii) holds if $\Delta_{nk}(\pi_{nk}(j)) \leq \hat{\delta}_{nk}^C(\pi_{nk}(j))$. Therefore, a conservative policy for user n on channel k at slot j is optimal if $\Delta_{nk}(\pi_{nk}(j)) \leq \delta_{nk}^C(j)$.

On the contrary, an aggressive policy is optimal if a) $r_{nk}(p_{nk}^M(j)) \geq 0$; or b) $\mathrm{E}_{\alpha_{nk}(p_{nk}^*(j))}\left\{R_{nk}(p_{nk}^*(j))\right\} < R_{nk}(P)$. It is possible to show that a) holds if $\Delta_{nk}(\pi_{nk}(j)) \geq \delta_{nk}^A(\pi_{nk}(j))$, while b) holds if $\Delta_{nk}(\pi_{nk}(j)) \geq \hat{\delta}_{nk}^A(\pi_{nk}(j))$. Thus, an aggressive policy is optimal if $\Delta_{nk}(\pi_{nk}(j)) \geq \delta_{nk}^C(j)$. Clearly, an exploratory policy is optimal if neither conservative nor aggressive policies are optimal.

The third and last part of the proof is a direct consequence of the history update mechanism in eq. (20). When $p_{nk}^{\mathrm{OPT}}(j)$ is conservative ($p_{nk}^{OPT}(j) = p_{nk}^m(j)$) or aggressive ($p_{nk}^{OPT}(j) = P$), it follows that $\pi_{nk}(j+1) = \pi_{nk}(j)$. Thus, the same policy will be still optimal for all $l > j$, i.e., $\pi_{nk}(j) = \pi_{nk}(l) = \cdots = \pi_{nk}(H)$. ■

Proposition 2 gives us an important insight on the learning dynamics of the system. We have shown that $\pi_{nk}(j) = \pi_{nk}(j')$ for all $j > j'$ if either conservative or aggressive policies are chosen at slot j'. That is, anytime that either conservative or aggressive policies are optimal for a given user on a given channel, the learning process for that user on that considered channel is stopped.

Although we proved that the optimal power control policy is always unique and we have shown the dynamics regulating the choice of the optimal policy, how to obtain the value of $p_{nk}^{\mathrm{OPT}}(j)$ still remains unsolved. A closed-form for $p_{nk}^{\mathrm{OPT}}(j)$, can be derived only by solving eq. (13), which is not possible for our problem. To find the solution of eq. (13), we exploit techniques from *stochastic approximation theory* and *exponential mappings*. For the sake of simplicity, in the following of this section we omit the subscripts n, k and the slot index j. Let us define $\tilde{R}(p) = \mathrm{E}_{\alpha(p)}\left\{R(p)\right\}$, and let $r(p)$ denote the first derivative of $\tilde{R}(p)$ w.r.t. p. From eq. (13), $r(p)$ can be written as

$$\begin{aligned} r(p) = \frac{d\tilde{R}(p)}{dp} &= \frac{g}{\sigma^2 + gp} + \frac{1}{p^M - p^m} \\ &\times \left[\log\left(1 + \frac{hP_J}{\sigma^2 + gp}\right) - \log\left(1 + \frac{hP_J}{\sigma^2}\right)\right] \\ &- \frac{p - p^m}{p^M - p^m} \cdot \frac{ghP_J}{(\sigma^2 + gp)(\sigma^2 + gp + hP_J)} \end{aligned} \quad (21)$$

We also assume $p^m = 0$ and $p^M = P$. However, the more general case where $0 < p^m < p^M < P$ can be treated similarly.[5]

The measure of the feasible power level set is $\Delta = p^M - p^m = P$. Finally, in (22) we consider the discrete-time stochastic approximation algorithm with exponential mappings

$$\begin{cases} z[i+1] = z[i] + \gamma_i r(p[i]) \\ p[i+1] = \Delta \dfrac{e^{z[i+1]}}{1 + e^{z[i+1]}} \end{cases} \quad (22)$$

where i is the iteration index and γ_i is a variable step-size [31]. It is worth remarking that, at any iteration of (22), the value

[5]Note that we can define an auxiliary variable $p' = p - p^m$, $p' \in [0, p^M - p^m]$. Our results still hold as $r(p) = \frac{d\tilde{R}(p)}{dp} = \frac{d\tilde{R}(p'+p^m)}{dp'} = r(p')$.

of $p[i]$ is always bounded in $[0, \Delta]$. That is, the proposed algorithm in (22) always generates feasible transmission power updates.[6] Let p^* be the optimal solution of eq. (13). Now, we first derive the continuous-time version of (22). Then, in Proposition 3, we prove that (22) converges to p^* for any possible feasible initial condition.

Let us define the continuous-time dynamics of (22) as

$$\begin{cases} \dot{z} = r(p) \\ p = \Delta \dfrac{e^z}{1+e^z} \end{cases} \tag{23}$$

Proposition 3: Let the step-size γ_i be defined such that $\sum_i \gamma_i^2 < \sum_i \gamma_i = +\infty$. Then, for any feasible initial condition, the discrete-time algorithm (22) always converges to the optimal solution of eq. (13).

Proof: Let $p(t)$ be a solution orbit of (23). Our proof consists in 1) showing that $p(t)$ converges to p^* as $t \to +\infty$, and 2) showing that (22) is an *asymptotic pseudo-trajectory* (APT) [32] for (23) which converges to p^* when some conditions on the step-size are satisfied. From Proposition 1, we have that $R(p)$ is strictly concave, therefore $r(p)(p - p^*) < 0$ for all $p \in [0, \Delta]$ by definition. Thus, we can exploit the latter result to build a Lyapunov-candidate-function for (23). It is easy to show that the function $V(p)$ defined as

$$V(p) = \Delta \log\left(\frac{\Delta - p^*}{\Delta - p}\right) + p^* \log\left(\frac{p^*}{p} \cdot \frac{\Delta - p}{\Delta - p^*}\right) \tag{24}$$

is a strict Lyapunov function for (23). One can show that this statement is true by observing that $\dot{V} = dV(p)/dt = r(p)(p - p^*) < 0$, $V(p^*) = 0$ and $V(p) > 0$ for all $p \neq p^*$. Furthermore, by decoupling z and p, we obtain $z = \log\left(\frac{p}{\Delta - p}\right)$; and by rewriting $V(p)$ in terms of z, we obtain $V(z)$ which can be proved to be *radially unbounded*. Therefore, we can assert that the optimal solution p^* is also *globally asymptotically stable* (GAS), which leads to the conclusion that $p(t)$ converges to p^* as $t \to +\infty$. Now, we prove the second part of the proposition. That is, we prove that also the discrete-time algorithm converges to the optimal solution as $i \to +\infty$. By decoupling (23), we get

$$\dot{p} = dp/dt = p\left(1 - \frac{p}{\Delta}\right) r(p) \tag{25}$$

This result will be useful to show that the discrete-time algorithm tracks the continuous-time system up to a bounded error that asymptotically tends to 0 as i increases.

A second-order Taylor expansion of (22) leads to

$$p[i+1] = p[i] + \gamma_i p[i]\left(1 - \frac{p[i]}{\Delta}\right) r(p[i]) + \frac{1}{2} M \gamma_i^2 \tag{26}$$

for some bounded M. Note that M is bounded as $\frac{d}{dp} r(p)$ is bounded by definition. Intuitively, eq. (26) is the discrete version of eq. (25) up to a bounded error. Since, by assumption, $\sum_i \gamma_i^2 < \sum_i \gamma_i = +\infty$, results contained in [32] show that $p[n]$ is an APT for (23).

It still remains to prove that $p[n] \to p^*$. By considering a Taylor expansion of $V(z)$, we obtain

$$V(z[i+1]) = V(z[i]) + \gamma_i (p[i] - p^*) r(p[i]) + \frac{1}{2} M' \gamma_i^2 \tag{27}$$

for some bounded $M' > 0$. Recall that $(p[i] - p^*) r(p[i]) < 0$ by definition and p^* is GAS. It follows that $\mathcal{W} = [0, \Delta]$ is a basin of attraction for p^*. Therefore, there must exist a compact set $\mathcal{L} \subset \mathcal{W}$ containing p^*. If we prove that there also exists a large enough i' such that $p[i'] \in \mathcal{L}$, then, the proof is concluded. Assume ad absurdum that such i' does not exists. Therefore, some $\beta > 0$ must exist such that $(p[i] - p^*) r(p[i]) \leq -\beta$ for all i. It follows that

$$V(z[i+1]) \leq V(z[i]) - \gamma_i \beta + \frac{1}{2} M' \gamma_i^2 \tag{28}$$

which, by telescoping, yields to

$$V(z[i+1]) \leq V(z[0]) - \beta \sum_i \gamma_i + \frac{1}{2} M' \sum_i \gamma_i^2 \tag{29}$$

Since we assumed that $\sum_i \gamma_i^2 < \sum_i \gamma_i = +\infty$, the latter equation leads to $V(z[i+1]) \leq -\infty$, which is a contradiction as $V(z)$ is lower bounded by construction. Therefore, [32] ensures that there must exist a large i' such that $p[i'] \in \mathcal{L}$ and $\lim_{i \to +\infty} p[i] = p^*$, which concludes our proof. ■

C. Optimal User Scheduling

In this section, we define the finite-horizon DP-based algorithm to solve (**A**). To properly define the DP framework, we must take into account the uncertainty introduced by the jammer's behavior and its impact on the dynamics of the problem. Thus, in the language of DP, we define:

- *System state*: we define the system state at slot j as the tuple $(\boldsymbol{\pi}(j), \boldsymbol{\rho}(j))$, where $\boldsymbol{\pi}(j) = (\pi_{nk}(j))_{n,k}$ denotes the *history vector* at slot j. At each slot, $\pi_{nk}(j)$ is updated according to eq. (20). On the other hand, $\boldsymbol{\rho}(j) = (\rho_n(j))_n$ denotes the *residual performance vector*. Each $\rho_n(j)$ specifies the remaining amount of performance that has to be allocated to user n from slot j to the horizon H to satisfy its minimum performance request R_n^*. At each slot, $\rho_n(j)$ is updated as follows:

$$\rho_n(j+1) = \begin{cases} R_n^* & \text{if } j = 1 \\ \rho_n(j) - \displaystyle\sum_{k \in \mathcal{K}} \theta_{nk}(j) \mathrm{E}_\alpha \{R_{nk}(p_{nk}^{\mathrm{OPT}}(j))\} & \text{otherwise} \end{cases} \tag{30}$$

- *Action*: at each slot j, the actions of the scheduler are the optimal scheduling and power control policies, i.e., $\boldsymbol{\theta}(j)$ and $\mathbf{p}(j)$, respectively. For any scheduling policy $\theta_{nk}(j) \in \boldsymbol{\theta}(j)$, it must hold that $\theta_{nk}(j) = \{0, 1\}$. Instead, $\mathbf{p}(j)$ contains the optimal power control policies chosen when the history of the system is $\boldsymbol{\pi}(j)$. The generic element $p_{nk}(j) \in \mathbf{p}(j)$ is trivially defined as $p_{nk}(j) = 0$ if $\theta_{nk}(j) = 0$ and $p_{nk}(j) = p_{nk}^{\mathrm{OPT}}(j)$ if $\theta_{nk}(j) = 1$, where each $p_{nk}^{\mathrm{OPT}}(j)$ is given in eq. (14);
- *Single-Slot Reward*: we define the function $\Phi(\boldsymbol{\pi}(j), \boldsymbol{\rho}(j), \boldsymbol{\theta}(j), \mathbf{p}(j), j)$ as the reward, i.e., the total transmission rate, that the system achieves at slot

[6] The same result also holds for the general case where $p[i] \in [p^m, p^M]$.

j when policy $(\boldsymbol{\theta}(j), \mathbf{p}(j))$ is chosen and the state is $(\boldsymbol{\rho}(j), \boldsymbol{\pi}(j))$. Therefore,

$$\Phi(\boldsymbol{\pi}(j), \boldsymbol{\rho}(j), \boldsymbol{\theta}(j), \mathbf{p}(j), j) = \sum_{k\in\mathcal{K}} \sum_{n\in\mathcal{N}} \theta_{nk}(j) \Gamma_{nk}(\boldsymbol{\pi}(j), \mathbf{p}(j), j)$$

where $\Gamma_{nk}(\boldsymbol{\pi}(j), \mathbf{p}(j), j) = \mathrm{E}_\alpha \left\{R_{nk}(p_{nk}^{OPT}(j))\right\}$ and $p_{nk}^{OPT}(j) \in \mathbf{p}(j)$ and it is calculated in eq. (14). Thus, the dependence of $\Gamma_{nk}(\boldsymbol{\pi}(j), \mathbf{p}(j), j)$ from $\boldsymbol{\pi}(j)$ is implicit in the definition of $p_{nk}^{OPT}(j)$.

To solve (**B**), we write the Bellman's equation [33]:

$$J(\boldsymbol{\pi}(j), \boldsymbol{\rho}(j), j) = \max_{\boldsymbol{\theta}(j), \mathbf{p}(j)} \Phi(\boldsymbol{\pi}(j), \boldsymbol{\rho}(j), \boldsymbol{\theta}(j), \mathbf{p}(j), j) + \mathrm{E}\left\{J(\boldsymbol{\pi}(j+1), \boldsymbol{\rho}(j+1), j+1) \,|\, \boldsymbol{\pi}(j), \boldsymbol{\rho}(j)\right\} \tag{31}$$

$$\text{s.t.} \sum_{i=j}^{H} \sum_{k\in\mathcal{K}} \theta_{nk}(j) \mathrm{E}_\alpha \left\{R_{nk}\left(p_{nk}^{OPT}(i)\right)\right\} \geq \rho_n(j) \tag{32}$$

$$\sum_{k\in\mathcal{K}} \theta_{nk}(j) \leq 1, \quad \forall n \in \mathcal{N} \tag{33}$$

$$\sum_{n\in\mathcal{N}} \theta_{nk}(j) \leq 1, \quad \forall k \in \mathcal{K} \tag{34}$$

$$\theta_{nk}(j) \in \{0,1\},\ p_{nk}(j) \in [0, P], \quad \forall n \in n, \quad \forall k \in \mathcal{K} \tag{35}$$

where we set $J(\boldsymbol{\pi}(j), \boldsymbol{\rho}(j), H+1) = 0$ for all $\boldsymbol{\pi}(j)$ and $\boldsymbol{\rho}(j)$.

At each slot, the Bellman's equation consists in a (binary) integer linear programming (ILP) problem. ILP problems are known to be NP-Complete and their exact solution can be computed through standard *Branch-and-Bound* methods. Finally, by using backwards induction [33], we solve the Bellman's equation and find the optimal solution to (**B**).

In Fig. 2, we show the building blocks of our proposed solution. Backward induction implies that we start from slot $j = H$ and go backwards in time. At each slot, we first solve the power control problem by exploiting the system state parameter $\boldsymbol{\pi}(j)$. The solution of the power control problem consists in the power control vector $\mathbf{p}(j)$. To solve the scheduling problem, the system requires $\mathbf{p}(j)$ and $\boldsymbol{\rho}(j)$. The scheduler solves the single-slot reward maximization problem and provides the optimal scheduling policy $\boldsymbol{\theta}(j)$. Finally, each element in $\mathbf{p}(j)$ is modified such that each $p_{nk}(j) \in \mathbf{p}(j)$ is set to $p_{nk}(j) = 0$ if $\theta_{nk}(j) = 0$ and the optimal joint power control and user scheduling policy $(\boldsymbol{\theta}(j), \mathbf{p}(j))$ is found.

V. Approximation of the Optimal Solution

In the previous sections, we have shown that DP can solve our considered problem optimally. However, it is well known that DP suffers from the "curse of dimensionality" [33]. That is, when the state space and the number of variables increase, the number of possible combinations that we need to solve considerably increases. As an example, the power control variables $\boldsymbol{p}(j)$ are defined over the continuous set $[0, P]$. It follows that the number of possible combinations of both the system state and the feasible actions is infinite. Furthermore, both the history $\boldsymbol{\pi}(j)$ and the residual performance vector $\boldsymbol{\rho}(j)$ depend on $\boldsymbol{p}(j)$, which contributes to further increase the dimension of the problem.

DP can theoretically still provide an optimal solution to such continuous space problem. However, from a practical point of view, it is unrealistic to implement the Bellman's equation on discrete-time systems.

To avoid the "curse of dimensionality", we show that an approximation of the optimal solution can be obtained by discretizing the power control variable. Specifically, we exploit *state aggregation* techniques [33] which allow to aggregate one or more spaces of the original problem to create several lower dimension abstract spaces. Thus, we can aggregate spaces by quantizing the power control action space to create a discretized version of it.

In the following of this section, we show that by discretizing both the power control variable and the residual performance vector it is possible to significantly reduce the complexity of the problem while guaranteeing user QoS requirements.

A. State Aggregation Approach

Let ξ_p be the power quantization step. Without losing in generality, we assume that the power quantization step is chosen such that ξ_p is a divisor for P.

Let $\lceil\cdot\rceil$ and $\lfloor\cdot\rfloor$ be the ceiling and floor operators, respectively. For any $p \in [0, P]$, let $\overline{p} = \left\lceil \frac{p}{\xi_p} \right\rceil \xi_p$ and $\underline{p} = \left\lfloor \frac{p}{\xi_p} \right\rfloor \xi_p$ be the higher and lower quantized power level values, respectively.

In the discretized version of the problem, we use the proposed quantization-based state aggregation to discretize $p_{nk}^*(j)$. That is, we calculate its quantized equivalents $\overline{p_{nk}^*}$ and $\underline{p_{nk}^*}$. Let $\hat{p}_{nk}^*(j)$ be defined as follows:

$$\hat{p}_{nk}^*(j) = \begin{cases} p_{nk}^*(j) & \text{if } \overline{p_{nk}^*} = \underline{p_{nk}^*} \\ \arg\max_{p=\{\overline{p_{nk}^*}, \underline{p_{nk}^*}\}} \mathrm{E}_{\alpha(p)}\left\{R_{nk}(p)\right\} & \text{otherwise} \end{cases}$$

Thus, $\hat{p}_{nk}^{\mathrm{OPT}}(j) = \arg\max_{p=\{\hat{p}_{nk}^*(j), P\}} \mathrm{E}_{\alpha(p)}\left\{R_{nk}(p)\right\}$ is the solution to the discretized version of eq. (14). [7]

Let $\mathbf{p}(j)$ and $\boldsymbol{\theta}(j)$ be the optimal power control and scheduling policy at slot j, respectively. Similarly to the continuous space problem, for any $p_{nk} \in \mathbf{p}(j)$ we have $p_{nk} = \hat{p}_{nk}^{\mathrm{OPT}}(j)$ iff $\theta_{nk}(j) = 1$. Otherwise, $\hat{p}_{nk}^{\mathrm{OPT}}(j) = 0$. At each slot, the history of the system $\pi_{nk}(j)$ is updated according to eq. (20).

So far, we have discretized the state of power control variable. However, from eqs. (5) and (30), it is clear that both the achievable rate and the residual performance vector have a continuous state space.

Let ξ_r be the performance quantization step. We assume that the network operator forces each user to submit a minimum performance requirement, R_n^*, such that the latter is an integer multiple of ξ_r. To overcome the high-dimensionality caused by the definition of the residual performance vector, we modify

[7] Note that even though $\hat{p}_{nk}^*(j)$ is optimal for the discretized version of eq. (13), it is sub-optimal for the continuous space problem, unless that $p_{nk}^*(j) = \hat{p}_{nk}^*(j)$.

the update dynamic of $\boldsymbol{\rho}(j)$ as follows:

$$\rho_n(j+1) = \begin{cases} R_n^* & \text{if } j=1 \\ \left\lfloor \dfrac{\rho_n(j)}{\xi_r} - \dfrac{\sum_{k\in\mathcal{K}} \theta_{nk} \mathrm{E}_{\boldsymbol{\alpha}}\left\{R_{nk}(p_{nk})\right\}}{\xi_r} \right\rfloor \xi_r & \text{otherwise} \end{cases} \tag{36}$$

where $p_{nk} \in \mathbf{p}(j)$.

Let N_p and N_r denote the maximum number of power and transmission rate quantized levels, respectively. Trivially, $N_p = (P/\xi_p + 1)$ and $N_r = (R^{max}/\xi_r + 1)$, where $R^{max} = \max\left\{R_1^*, R_2^*, \ldots, R_{|\mathcal{N}|}^*\right\}$.

Now, we apply the Bellman's equation to the discretized problem and find its optimal solution.

At each slot j, the number of possible combinations of $\boldsymbol{\rho}(j)$ and $\boldsymbol{\pi}(j)$ are $N_r{}^{|\mathcal{N}|}$ and $(\frac{N_p(N_p-1)}{2})^{|\mathcal{N}||\mathcal{K}|}$, respectively. Finding the maximum of the single-slot reward maximization problem has complexity $\mathcal{O}\left((\max\{|\mathcal{K}|,|\mathcal{N}|\})^{\min\{|\mathcal{K}|,|\mathcal{N}|\}}\right)$. Finally, the number of possible combinations in eq. (31) is $\mathcal{O}(|\mathcal{K}||\mathcal{N}|H)$. Therefore, the overall complexity of the Bellman's equation is $\mathcal{O}\left(\omega|\mathcal{K}||\mathcal{N}|\left(N_r\frac{N_p^{|\mathcal{K}|}(N_p-1)^{|\mathcal{K}|}}{2^{|\mathcal{K}|}}\right)^{|\mathcal{N}|}H\right)$ where $\omega = (\max\{|\mathcal{K}|,|\mathcal{N}|\})^{\min\{|\mathcal{K}|,|\mathcal{N}|\}}$. However, if $|\mathcal{K}| < |\mathcal{N}|$, by exploiting constraints (33) and (34) we can reduce the complexity of the single-slot maximization problem to $\mathcal{O}\left(\frac{|\mathcal{N}|!}{(|\mathcal{N}|-|\mathcal{K}|)!}\right)$. Thus, the complexity becomes $\mathcal{O}\left(|\mathcal{K}||\mathcal{N}|\frac{|\mathcal{N}|!}{(|\mathcal{N}|-|\mathcal{K}|)!}\left(N_r\frac{N_p^{|\mathcal{K}|}(N_p-1)^{|\mathcal{K}|}}{2^{|\mathcal{K}|}}\right)^{|\mathcal{N}|}H\right)$. In the special case where $|\mathcal{N}| = \mathcal{O}(\log H)$, the proposed algorithm has polynomial complexity. That is, to solve the problem in polynomial time, we should consider either a low number of users that scales as the logarithm of the horizon H, or a large value of H. In all other cases, the complexity of the algorithm exponentially increases with the number of users in the system.

VI. Performance-Aware Online Greedy Algorithm (POGA)

In previous sections, we have proposed both optimal and approximated solutions to the joint power control and scheduling problem. Unfortunately, due to complexity issues, those solutions may not find application to those scenarios where the number of users and channels are large. Furthermore, solutions proposed in Sections IV and V require a priori knowledge of the users in the network. However, there are some scenarios where users dynamically access and leave the network, i.e., the number of users in the network vary in time. Under the above assumptions, to apply both our optimal or approximated solutions is not possible. Accordingly, to account for the above issues, in this section we focus on the design of a low-complexity online algorithm.

Let $\mathcal{N}(j)$ be the set of users at slot $j \in \mathcal{H}$ and $K = |\mathcal{K}|$ be the number of wireless channels. Accordingly, $\mathcal{N}^U(j) = \{n \in \mathcal{N}(j) : \rho_n(j) = 0\}$ indicates the set of unsatisfied users whose minimum QoS requirement has not been satisfied yet. Consequently, $\mathcal{N}^S(j) = \mathcal{N}(j) \setminus \mathcal{N}^U(j)$ represents the set of satisfied users up to slot j.

Algorithm 1 Performance-Aware Online Greedy Algorithm (POGA)

Input: System state $(\boldsymbol{\pi}(j), \boldsymbol{\rho}(j))$, Satisfied users $\mathcal{N}^S(j)$, Unsatisfied users $\mathcal{N}^U(j)$.
Output: A greedy power control and scheduling policy $(\boldsymbol{\theta}(j), \mathbf{p}(j))$.

```
Calculate power control policy p(j) from eq. (22)
Find the greedy scheduling θ^U(j) among unsatisfied users in N^U(j)
if There are unscheduled channels then
    Find the greedy scheduling θ^S(j) among satisfied users in N^S(j)
    end
end
θ(j) ← θ^U(j) ∨ θ^S(j)
p(j) ← θ(j) ∘ p(j)
Update (π(j+1), ρ(j+1))
return (θ(j), p(j))
```

Since no information about future user arrival is available, and to reduce the complexity of the problem, in Algorithm 1 we propose a Performance-aware Online Greedy Algorithm (POGA) to schedule user transmissions. We exploit the decomposability of the problem that we have introduced in Section IV-A. Specifically, we first calculate the power control policy $\mathbf{p}(j)$ as in eq. (14) by exploiting the discrete-time stochastic approximation algorithm with exponential mappings that we have introduced in eq. (22).

Then, to account for minimum QoS requirements, we give priority to unsatisfied users in $\mathcal{N}^U(j)$ and we schedule their transmissions by finding the greedy scheduling policy $\boldsymbol{\theta}^U(j)$. If all channels have been assigned to unsatisfied users, i.e., $\sum_{n\in\mathcal{N}^U(j)}\sum_{k\in\mathcal{K}}\theta_{nk}(j) = K$, we set $\boldsymbol{\theta}(j) = \boldsymbol{\theta}^S$ and $\mathbf{p}(j) = \boldsymbol{\theta}(j) \circ \mathbf{p}(j)$, where "$\circ$" is the Hadamard product operator. Otherwise, if some channels have not been allocated in $\boldsymbol{\theta}^U(j)$, i.e., $\sum_{n\in\mathcal{N}^U(j)}\sum_{k\in\mathcal{K}}\theta_{nk}(j) < K$, we exploit those unused channels to schedule transmissions of already satisfied users in $\mathcal{N}^S(j)$ and calculate the greedy scheduling policy $\boldsymbol{\theta}^S(j)$. Thus, we set $\boldsymbol{\theta}(j) = \boldsymbol{\theta}^U(j) \vee \boldsymbol{\theta}^S(j)$ and $\mathbf{p}(j) = \boldsymbol{\theta}(j) \circ \mathbf{p}(j)$, where "$\vee$" is the logical OR operator. Finally, we update $(\boldsymbol{\pi}(j+1), \boldsymbol{\rho}(j+1))$ according to eqs. (20) and (30) and POGA returns the sub-optimal scheduling solution $(\boldsymbol{\theta}(j), \mathbf{p}(j))$.

Let $N = \max_{j\in\mathcal{H}}\{\mathcal{N}(j)\}$ be the maximum number of users in the system. It can be shown that the complexity of the power control algorithm is $\mathcal{O}(NK)$ [34]. By exploiting results derived in V-A, it can be shown that to find the greedy scheduling policies $\boldsymbol{\theta}^S(j)$ and $\boldsymbol{\theta}^U(j)$ has complexity $\mathcal{O}(\frac{|\mathcal{N}|!}{(|\mathcal{N}|-|\mathcal{K}|)!}) = \mathcal{O}(N^{K-1})$. Therefore, the complexity of executing Algorithm 1 for H slots is $\mathcal{O}(HN^K)$. That is, the overall complexity of POGA is polynomial in the number N of users in the system.

VII. Numerical Results

In this section, we evaluate the achievable performance of the solutions above proposed through extensive numerical simulations. Specifically, in Section VII-A we assess the performance of the approximated solution to the joint power

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

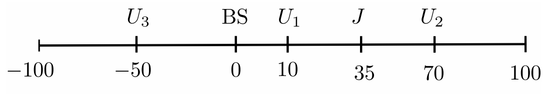

Fig. 4. Topology of the simulated 1-dimensional scenario.

TABLE II
SIMULATION SETTING

Parameter	Value
Carrier frequency	$f_c = 2.4\,\text{GHz}$
Channel bandwidth	$B = 10.93\,\text{KHz}$
Noise spectral density	$\sigma^2 = 584\,\mu\text{W}$
Maximum transmission power of users	$P = 0.6\,\text{W}$
Triggering threshold	$P_{th} = 0.5\,\mu\text{W}$
Edge of the simulated square area	$L = 200\,\text{m}$
Horizon duration	$H = 10$

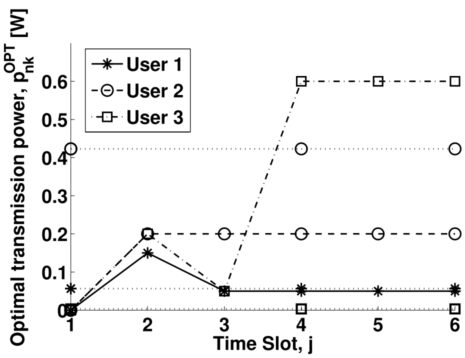

Fig. 5. Learning process on a single channel. Dotted lines show the lowest transmission power that triggers the jammer.

control and user scheduling problem. Instead, the achievable performance of POGA and its comparison with other traditional scheduling policies are discussed in Section VII-B.

A. Performance Assessment of the Approximated Solution

We consider a wireless network consisting of $N = 3$ legitimate users. Unless otherwise stated, we consider $K = 2$ channels whose gain coefficients g_{nk}, h_{nk} and h_k are generated according to the path-loss model. As depicted in Fig. 4, we consider a 1-dimensional scenario where users (i.e., U_1, U_2, U_3), the BS and the jammer (i.e., J) are located along the same axis.[8] Other relevant simulation parameters are reported in Table II. Finally, unless otherwise stated, we set the maximum transmission power for the jammer to $P_J = 0.6\,W$, which is a common setting of many commercial mobile jammers.

We start by illustrating how the learning process on a single channel evolves over time. In Fig. 5, we show the evolution of the learning process, i.e., the calculation of p_{nk}^{OPT} over time, as a function of P_J. For all users, dotted lines represents the lowest transmission power that triggers the jammer. The exploratory policy chosen by U_1 at $j = 2$ causes the triggering

[8]This is just an illustrative example. However, our approach is independent of the actual wireless network topology.

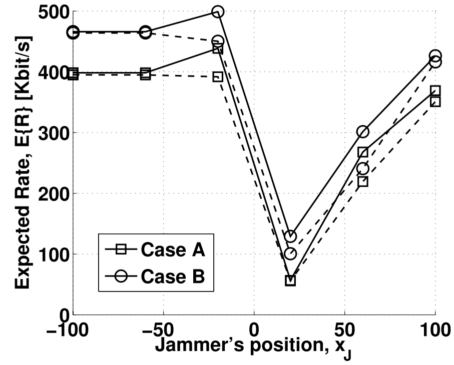

Fig. 6. Expected rate of the system as a function of the position of the jammer x_J for different QoS requirement settings and values of P_J (Solid lines: $P_J = 0.5$W; Dashed lines: $P_J = 1$W).

of the jammer. Thus, U_1 reduces its transmitting power but it again triggers the jammer. This is because the triggering threshold for user U_1 (dotted-starred lines) is low. Hence, U_1 chooses an aggressive policy even though it triggers the jammer. Similarly, the exploratory policy of U_2 at $j = 2$ does trigger the jammer. Therefore, U_2 reduces its transmission power and explores a new transmission power policy. The new explored policy is found to not trigger the jammer. Accordingly, U_2 evaluates eq. (14) and realizes that no further improvements are needed, a conservative policy is chosen and the learning process for U_2 is stopped. On the other hand, U_3 is far away from the jammer. Therefore, as shown in Fig. 5, the triggering threshold for this user is high. As a consequence, the first exploratory policy does not trigger the jammer. However, the risk of triggering the jammer is high. It follows that eq. (14) suggests to choose a conservative policy.

To investigate the impact of the position x_J of the jammer on the achievable performance of the network, in Fig. 6 we evaluate the expected transmission rate as a function of x_J under two different minimum QoS requirements. Specifically, we consider the case where all users request an identical minimum QoS level $R_n^* = 10.93\,\text{Kbit/s}$ (Case A), and the case where no requirements are submitted by users (Case B). Fig. 6 shows that when the jammer is at the border of the considered scenario, its attacks have limited effect on network performance. Instead, when the jammer approaches the BS and the users, the achievable rate of the network considerably decreases. Also, when no QoS constraints are considered and rate maximization is the only objective of the network operator, system performance are higher than those achieved in Case A.

In Figs. 7 and 8 we compare the performance of the system when different scheduling policies are considered. Specifically, we compare the proposed approximated solution to random, round-robin and greedy scheduling policies without minimum QoS requirements. Note that the greedy scheduling policy without QoS requirements we consider in Figs. 7 and 8 is different from POGA. POGA is performance-aware and aims at satisfying user QoS requirements. On the contrary, a greedy scheduling policy without QoS requirements only schedules those users whose transmission rate is the highest.

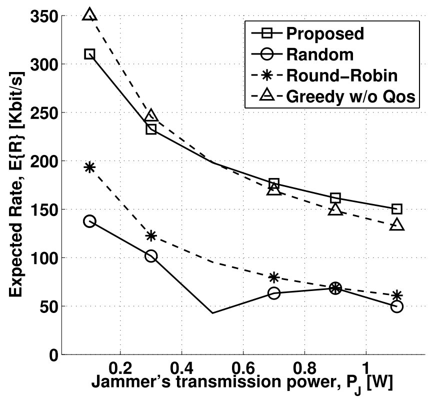

Fig. 7. Expected rate of the system under different scheduling policies as a function of P_J.

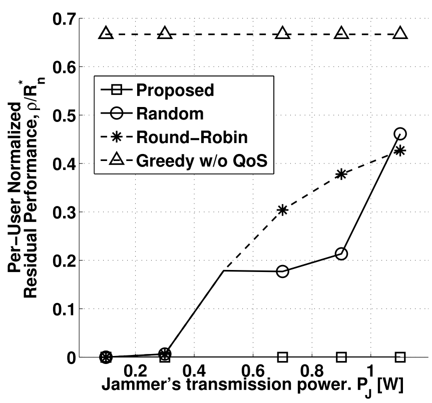

Fig. 8. Average per-user normalized residual performance at the end of the joint power control and user scheduling cycle as a function of P_J.

A comparison between POGA and a greedy scheduling policy without QoS requirements is in Section VII-B.

Fig. 7 shows that our proposed solution reaches high transmission rates and it outperforms random and round robin policies. The achievable performance of the network under the proposed approximated solution and the greedy without QoS requirements are comparable. However, in Fig. 8 we plot the per-user normalized residual performance variable $\bar{\rho}$ at the horizon H defined as

$$\bar{\rho} = \frac{1}{N} \sum_{n \in \mathcal{N}} \frac{\rho_n(H)}{R_n^*} \quad (37)$$

$\bar{\rho}$ represents the QoS-gap, i.e., the per-user amount of performance that has not been provided to users at the end of the scheduling cycle. Thus, the desirable value is $\bar{\rho} = 0$, while $\bar{\rho} > 0$ indicates infeasible solutions. Our solution is the only one that guarantees $\bar{\rho} = 0$, i.e., all minimum QoS requirements are satisfied. Therefore, even though a greedy approach without QoS requirements allows to achieve high performance, it does not guarantee minimum QoS levels. In the considered simulation setting, Fig. 8 also shows that $\bar{\rho}$ for the greedy approach without QoS requirements is constant and high. On the contrary, the value of $\bar{\rho}$ under random and round robin policies is low for small values of the transmission power of the jammer, but it increases when the value of P_J increases as well.

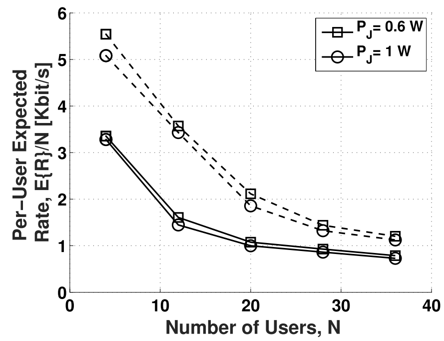

Fig. 9. Per-user expected rate of the system under POGA as a function of the number N of users (Solid lines: $K = 4$; Dashed lines: $K = 8$).

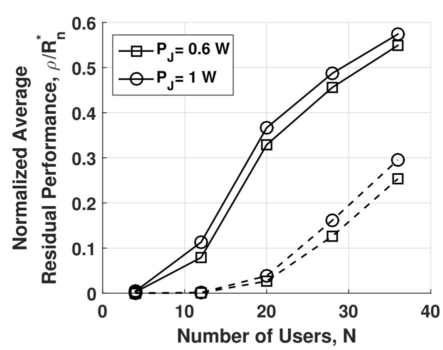

Fig. 10. Average per-user normalized residual performance under POGA as a function of the number N of users (Solid lines: $K = 4$; Dashed lines: $K = 8$).

B. Performance Assessment of POGA

In this section, we assess the performance of the online algorithm POGA. The results being presented are averaged over 200 simulations. At each simulation run, the positions of the users and the jammer are randomly generated within a square area of edge $L = 200$ according to a uniform distribution.

In Fig. 9, we assume $H = 5$ and we show the per-user expected rate of the system under POGA as a function of the number N of users when $K = 4$ (solid lines) and $K = 8$ (dashed lines). POGA first aims at satisfying all users. Then, it greedily maximizes the rate of the system by scheduling the users that have been already satisfied. Therefore, as N increases, the per-user expected rate under POGA decreases. Also, it is worth noting that the per-user expected rate when $K = 8$ approaches that of $K = 4$ as the number of users increases as well. Such a result stems from the fact that when the available channels are shared by a large number of users, the improvement provided by increasing the number K of channels reduces. It is worth noting that by increasing the number K of channels it is possible to considerably improve the rate of the system. Instead, worse performance are achieved when the jammer attacks with higher transmission power P_J.

In Fig. 10, we show the average per-user normalized residual performance variable defined in eq. (37) under POGA as a function of the number N of users when $K = 4$ (solid lines) and $K = 8$ (dashed lines). It is shown that, as N increases, the average per-user normalized residual performance increases

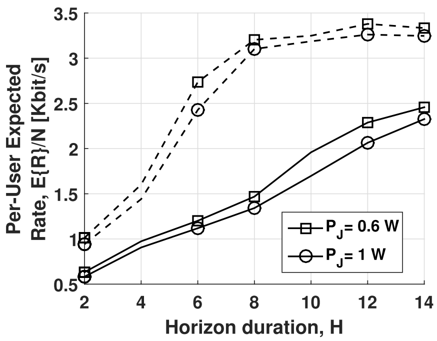

Fig. 11. Per-user expected rate of the system under POGA as a function of the horizon duration H (Solid lines: $K = 4$; Dashed lines: $K = 8$).

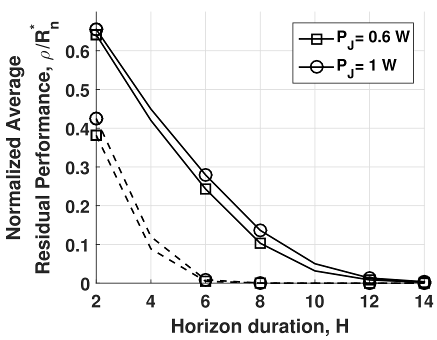

Fig. 12. Average per-user normalized residual performance under POGA as a function of the horizon duration H (Solid lines: $K = 4$; Dashed lines: $K = 8$).

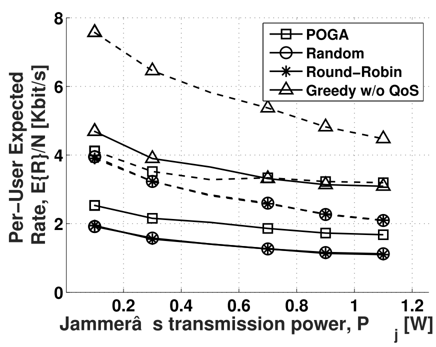

Fig. 13. Expected rate of the system under different scheduling policies as a function of P_J.

as well. Such a result is due to the fact that it is hard to accommodate all minimum QoS requirements when a large number of users is under attack. It is worth noting that, by increasing the number of channels, it is possible to improve the performance of POGA and reduce the average per-user normalized residual performance. As expected, smaller values of P_J result in better performance of the system.

To investigate the impact of the horizon duration H on the achievable performance of the system, in Figs. 11 and 12 we consider $N = 20$ users and we show the per-user expected rate of the system and the average per-user normalized residual performance, respectively.

Fig. 11 shows that higher rates are achieved when the horizon duration is large. Analogously, Fig. 12 illustrates how the average per-user normalized residual performance variable decreases, and asymptotically tends to zero, as H increases. That is, large values of the horizon duration H allow

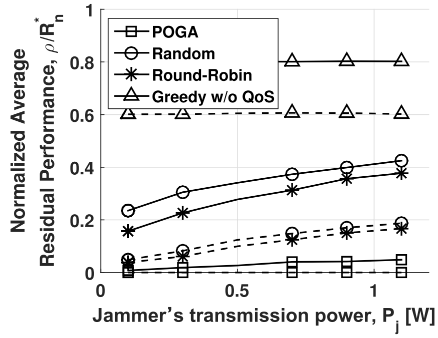

Fig. 14. Average per-user normalized residual performance at the end of the joint power control and user scheduling cycle as a function of P_J.

to schedule and accommodate more users, which eventually results in better performance.

Finally, in Figs. 13 and 14 we compare POGA to other scheduling policies. We assume an horizon duration of $H = 10$ and we consider $N = 20$ users. Similarly to what has been shown in Figs. 7 and 8, a greedy policy without QoS requirements achieves the highest transmission rate. However, it poorly performs in terms of guaranteeing a minimum QoS level to users. Instead, Fig. 13 shows that POGA outperforms random and round-robin policies in terms of per-user expected rate. And, as illustrated in Fig. 14, POGA achieves the lowest per-user normalized residual performance. That is, among the considered scheduling policies, POGA is the best suited to achieve high transmission rate while guaranteeing user QoS requirements. Furthermore, both Figs. 13 and 14 show that by increasing the number K of channels, all scheduling policies achieve higher performance.

VIII. Conclusions

In this paper, we studied the joint power control and scheduling problem in jammed networks under minimum QoS constraints. By assuming that no information on jammer's behavior and position is available, we proved that the problem is NP-Hard. However, we showed that it can be decomposed and modeled as a dynamic problem whose complexity is further reduced by exploiting state aggregation techniques. Learning through observation of jammer's reactions to transmission decisions taken in the past has been exploited to identify optimal transmission policies at future slots and maximize network performance. We also proposed and discussed a low-complexity Performance-aware Online Greedy Algorithm (POGA). Simulation comparison showed that the approaches proposed in this paper outperform other traditional scheduling policies.

References

[1] M. Wilhelm, I. Martinovic, J. B. Schmitt, and V. Lenders, "Short paper: Reactive jamming in wireless networks: How realistic is the threat?" in *Proc. 4th ACM Conf. Wireless Netw. Secur.*, 2011, pp. 47–52.

[2] M. Pajic and R. Mangharam, "Spatio-temporal techniques for anti-jamming in embedded wireless networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2010, Apr. 2010, Art. no. 51.

[3] A. D. Wood, J. A. Stankovic, and G. Zhou, "DEEJAM: Defeating energy-efficient jamming in IEEE 802.15.4-based wireless networks," in *Proc. 4th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw. (SECON)*, Jun. 2007, pp. 60–69.

[4] D. Nguyen, C. Sahin, B. Shishkin, N. Kandasamy, and K. R. Dandekar, "A real-time and protocol-aware reactive jamming framework built on software-defined radios," in *Proc. ACM Workshop Softw. Radio Implement. Forum*, New York, NY, USA, 2014, pp. 15–22.

[5] E. Bayraktaroglu *et al.*, "On the performance of IEEE 802.11 under jamming," in *Proc. IEEE INFOCOM*, Apr. 2008, 1939–1947.

[6] A. Marttinen, A. M. Wyglinski, and R. Jäntti, "Statistics-based jamming detection algorithm for jamming attacks against tactical MANETs," in *Proc. IEEE Military Commun. Conf. (MILCOM)*, Oct. 2014, pp. 501–506.

[7] M. Strasser, B. Danev, and S. Čapkun, "Detection of reactive jamming in sensor networks," *ACM Trans. Sensor Netw.*, vol. 7, no. 2, Sep. 2010, Art. no. 16.

[8] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: Attack and defense strategies," *IEEE Netw.*, vol. 20, no. 3, pp. 41–47, May/Jun. 2006.

[9] R. A. Poisel, *Modern Communications Jamming Principles and Techniques*. Norwood, MA, USA: Artech House, 2011.

[10] K. Grover, A. Lim, and Q. Yang, "Jamming and anti–jamming techniques in wireless networks: A survey," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 17, no. 4, pp. 197–215, 2014.

[11] Z. Lu, W. Wang, and C. Wang, "Modeling, evaluation and detection of jamming attacks in time-critical wireless applications," *IEEE Trans. Mobile Comput.*, vol. 13, no. 8, pp. 1746–1759, Aug. 2014.

[12] D. Giustiniano, V. Lenders, J. B. Schmitt, M. Spuhler, and M. Wilhelm, "Detection of reactive jamming in DSSS-based wireless networks," in *Proc. 6th ACM Conf. Secur. Privacy Wireless Mobile Netw.*, 2013, pp. 43–48.

[13] J. Dams, M. Hoefer, and T. Kesselheim, "Jamming-resistant learning in wireless networks," in *Automata, Languages, and Programming* (Computer Science), vol. 8573. Berlin, Germany: Springer, 2014, pp. 447–458.

[14] X. Liu, E. K. P. Chong, and N. B. Shroff, "Opportunistic transmission scheduling with resource-sharing constraints in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 10, pp. 2053–2064, Oct. 2001.

[15] X. Liu, E. K. P. Chong, and N. B. Shroff, "A framework for opportunistic scheduling in wireless networks," *Comput. Netw.*, vol. 41, no. 4, pp. 451–474. 2003.

[16] C. Popper, M. Strasser, and S. Capkun, "Anti-jamming broadcast communication using uncoordinated spread spectrum techniques," *IEEE J. Sel. Areas Commun.,* , vol. 28, no. 5, pp. 703–715, Jun. 2010.

[17] E.-K. Lee, S. Y. Oh, and M. Gerla, "Randomized channel hopping scheme for anti-jamming communication," in *Proc. IFIP Wireless Days (WD)*, Oct. 2010, pp. 1–5.

[18] B. DeBruhl, C. Kroer, A. Datta, T. Sandholm, and P. Tague, "Power napping with loud neighbors: Optimal energy-constrained jamming and anti-jamming," in *Proc. ACM Conf. Secur. Privacy Wireless Mobile Netw.*, 2014, pp. 117–128.

[19] S. Khattab, D. Mosse, and R. Melhem, "Jamming mitigation in multiradio wireless networks: Reactive or proactive?" in *Proc. 4th Int. Conf. Secur. Privacy Commun. Netw.*, 2008, Art. no. 27.

[20] I. Shin, Y. Shen, Y. Xuan, M. T. Thai, and T. Znati, "A novel approach against reactive jamming attacks," *Ad Hoc Sensor Wireless Netw.*, vol. 12, nos. 1–2, pp. 125–149, 2011.

[21] Y. Liu and P. Ning, "BitTrickle: Defending against broadband and high-power reactive jamming attacks," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 909–917.

[22] Y. Xuan, Y. Shen, N. Nguyen, and M. Thai, "A trigger identification service for defending reactive jammers in WSN," *IEEE Trans. Mobile Comput.*, vol. 11, no. 5, pp. 793–806, May 2012.

[23] I. Shin, Y. Shen, Y. Xuan, M. T. Thai, and T. Znati, "Reactive jamming attacks in multi-radio wireless sensor networks: An efficient mitigating measure by identifying trigger nodes," in *Proc. 2nd ACM Int. Workshop Found. Wireless Ad Hoc Sensor Netw. Comput.*, 2009, pp. 87–96.

[24] W. Xu, W. Trappe, and Y. Zhang, "Anti-jamming timing channels for wireless networks," in *Proc. 1st ACM Conf. Wireless Netw. Secur.*, New York, NY, USA, 2008, pp. 203–213.

[25] R. D. Pietro and G. Oligeri, "Silence is golden: Exploiting jamming and radio silence to communicate," *ACM Trans. Inf. Syst. Secur.*, vol. 17, no. 3, pp. 9-1–9-24, Mar. 2015.

[26] S. D'Oro *et al.*, "Defeating jamming with the power of silence: A game-theoretic analysis," *IEEE Trans. Wireless Commun.*, vol. 14, no. 5, pp. 2337–2352, May 2015.

[27] K. Xu, Q. Wang, and K. Ren, "Joint UFH and power control for effective wireless anti-jamming communication," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 738–746.

[28] W. Xu, "On adjusting power to defend wireless networks from jamming," in *Proc. 4th Annu. Int. Conf. Mobile Ubiquitous Syst., Netw. Services, (MobiQuitous)*, Aug. 2007, pp. 1–6.

[29] M. R. Garey and D. S. Johnson, *Computers and Intractability A Guide to the Theory of NP-Completeness*. San Francisco, CA, USA: Freeman, 1990.

[30] J. G. March, "Exploration and exploitation in organizational learning," *Org. Sci.*, vol. 2, no. 1, pp. 71–87, 1991.

[31] H. J. Kushner and G. G. Yin, *Stochastic Approximation Algorithms and Applications* (Applications of Mathematics), vol. 35. New York, NY, USA: Springer, 1997.

[32] M. Benaïm, "Dynamics of stochastic approximation algorithms," in *Seminaire de Probabilites XXXIII* (Mathematics), vol. 1709. Berlin, Germany: Springer, 1999.

[33] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA, USA: Athena Scientific, 1995.

[34] S. D'Oro, P. Mertikopoulos, A. L. Moustakas, and S. Palazzo, "Interference-based pricing for opportunistic multicarrier cognitive radio systems," *IEEE Trans. Wireless Commun.*, vol. 14, no. 12, pp. 6536–6549, Dec. 2015.

Salvatore D'Oro (S'12) received the B.S. degree in computer engineering, the M.S. degree in telecommunications engineering, and the Ph.D. degree from the University of Catania in 2011, 2012, and 2015, respectively. From 2013 to 2015, he was a Visiting Researcher with Université Paris-Sud 11, Paris, France, and The Ohio State University, Columbus, OH, USA. He is currently a Post-Doctoral Research Fellow with the University of Catania. In 2015, he organized the First Workshop on COmpetitive and COoperative Approaches for 5G networks (COCOA) at European Wireless 2016. He served on the Technical Program Committee (TPC) of the CoCoNet8 workshop at the IEEE ICC 2016. In 2013, he served on the TPC of the 20th European Wireless Conference (EW2014).

Eylem Ekici (S'99–M'02–SM'11) received the B.S. and M.S. degrees in computer engineering from Boğaziçi University, Istanbul, Turkey, in 1997 and 1998, respectively, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2002. He is currently a Professor with the Department of Electrical and Computer Engineering, The Ohio State University. His current research interests include cognitive radio networks, vehicular communication systems, and next-generation wireless systems, with a focus on algorithm design, medium access control protocols, resource management, and analysis of network architectures and protocols. He is the TPC Co-Chair of the IEEE INFOCOM 2017. He is an Associate Editor of the IEEE Transactions on Mobile Computing and *Computer Networks*. He is a former Associate Editor of the IEEE/ACM Transactions on Networking.

Sergio Palazzo (M'92–SM'99) received the degree in electrical engineering from the University of Catania, Catania, Italy, in 1977. Since 1987, he has been with the University of Catania, where is currently a Professor of telecommunications networks. His current research interests include mobile systems, wireless and satellite IP networks, and protocols for the next generation of the Internet. He is a member of the MobiHoc Steering Committee. He has also been a TPC Co-Chair of some other conferences, including IFIP Networking 2011, IWCMC 2013, and European Wireless 2014. He has been the General Chair of some ACM conferences, including MobiHoc 2006 and MobiOpp 2010. He currently serves on the Editorial Board of *Ad Hoc Networks*. He was an Editor of the IEEE Wireless Communications Magazine, the IEEE/ACM Transactions on Networking, the IEEE Transactions on Mobile Computing, *Computer Networks*, and *Wireless Communications and Mobile Computing*.