Packet Scheduling over a Wireless Channel: AQT-based Constrained Jamming ⋆

Antonio Fernández Anta[1], Chryssis Georgiou[2], and Elli Zavou⋆⋆[1,3]

[1] IMDEA Networks Institute, Madrid, Spain,
[2] University of Cyprus, Nicosia, Cyprus
[3] Universidad Carlos III de Madrid, Madrid, Spain

Abstract. In this paper we consider a two-node setting with a sender transmitting packets to a receiver over a wireless channel. Unfortunately, the channel can be jammed, thus corrupting the packet that is being transmitted at the time. The sender has a specific amount of data that needs to be sent to the receiver and its objective is to complete the transmission of the data as quickly as possible in the presence of jamming.
We assume that the jamming is controlled by a constrained adversary. In particular, the adversary's power is constrained by two parameters, ρ and σ. Intuitively, ρ represents the rate at which the adversary can jam the channel, and σ the length of the largest bursts of jams it can cause. This definition corresponds to the translation of the Adversarial Queuing Theory (AQT) constrains, typically defined for packet injections in similar settings, to channel jamming.
We propose deterministic scheduling algorithms that decide the lengths of the packets to be sent by the sender in order to minimize the transmission time. We first assume all packets being of the same length (uniform) and characterize the corresponding optimal packet length. Then, we show that if the packet length can be adapted, for specific values of ρ and σ the transmission time can be improved.

Keywords: Packet scheduling, Wireless Channel, Unreliable communication, Adversarial jamming, Adversarial Queueing Theory

1 Introduction

1.1 Motivation

The fast transmission of data across wireless channels under different conditions has been an area of investigation for quite some time now [3, 6, 10, 11, 14, 18, 21–25]. However, it presents several challenges depending on the model and applications it focuses on; especially when considering channel jamming.

⋆ This work has been supported in part by the Regional Government of Madrid (CM) grant Cloud4BigData (S2013/ICE-2894, cofunded by FSE & FEDER).
⋆⋆ Partially supported by FPU Grant from MECD

In our work we look at a wireless channel between a single pair of stations (sender and receiver), with the sender's goal to fully transmit a specific amount of data in the most efficient way. As efficiency measures, we look both at the *transmission time* and the *goodput* ratio (successful transmission rate), which are intuitively reversely proportional. Nonetheless, the communication between the sender and the receiver is being "watched" by a malicious entity that sporadically introduces noise in the channel, jamming the packet that happens to be transmitted at that time. More precisely, we model the errors in the channel to be controlled by an adversary with constrained power; defined by parameters ρ and σ. Parameter ρ represents the rate at which the adversary can jam the channel and σ the largest size of a burst of jams that can be caused. A packet that is jammed needs to be retransmitted; hence a feedback mechanism is assumed that informs the sender when a packet was jammed. The sender must transmit data of total size P. Each packet sent contains a *header* of fixed size h and some *payload* whose size, l, is algorithm-depended. Note that this payload counts towards the total size of P to be transmitted. For simplicity and without loss of generality we assume that $h = 1$ and the time to transmit a packet is equal to its length.

The constrained power of the adversary models a jamming entity with limited resource of energy, e.g., military drones [13, 17] or malicious mobile devices [1, 2]. For the adversarial jammer in our model, we consider having a battery of capacity σ units, where each unit can be used to cause one jam. Furthermore, in every $1/\rho$ time the battery is charged by one unit, e.g. with solar cells. More details on the model we consider are given in Section 2.

In a previous work [4], we studied the impact of adversarial errors on packet scheduling, focusing on the long term competitive ratio of throughput, termed *relative throughput*. We explored the effect of feedback delay and proposed algorithms that achieve close to optimal relative throughput under worst-case errors, and adversarial or stochastic packet arrivals. One of the main differences with this work is that the adversary was not constrained. Another difference is the fact that in the current work the packet sizes are to be chosen by the sender in order to send the desired amount of data efficiently. Furthermore, in [4], jammed packets were not retransmitted; the objective was to route packets as fast as possible and not strive to have each packet transmitted. In the current work, the choice of the packet size is precisely the most critical part from the side of the sender. Thus, we focus in devising scheduling algorithms for the decision of packet length to be used and conduct worst-case analysis for the efficiency measures.

1.2 Contributions

First, we introduce an AQT-based adversarial jamming model in wireless networks. To the best of our knowledge, this is the first work that uses such approach to restrict the power of adversarial jamming in such networks. AQT has been widely used for restricting packet arrivals in similar settings (see related work below). However, no research work has considered the possibility of exploring its effects in the intent to "damage" a network. As already mentioned, our approach of constrained adversarial jamming could be used to model battery-operated malicious devices that have bounded battery capacity and specific recharging rate. In Section 2 we formalize the constrained adversarial jamming model we consider.

Then, we present the limitations it imposes on the efficiency of scheduling policies, focusing on the *transmission time* Tr and the *goodput* G as our main performance measures. More precisely, in Section 3 we show bounds on both measures, by focusing on executions with *uniform packet lengths*. We first compute the quasi optimal payload size l^* and show that the optimal transmission time satisfies $Tr \in [LB^*, LB^* + l^* + 1)$, where $LB^* = \frac{[P+(\sigma-1)l^*](l^*+1)}{l^*(1-\rho(l^*-1))}$ and the optimal goodput is $G \in (\frac{P}{LB^*+l^*+1}, \frac{P}{LB^*}]$. We also show, that for uniform packets, as the total amount of data P grows, G is upper bounded by $(1-\sqrt{\rho})^2$, and in infinity ($P \to \infty$) the goodput grows to optimal $G^* = (1-\sqrt{\rho})^2$ regardless of σ.

From the above, one might wonder whether scheduling uniform packets is in fact the overall best strategy. In Section 4 we show that this is not the case. Focusing on $\sigma = 1$ we show that the optimal goodput derived from uniform packet length transmission, G^*, can be exceeded using an *adaptive algorithm*; an algorithm that decides the length of the packet to be sent next, based on the information provided by a feedback mechanism up to that point in time. In particular, we present the adaptive scheduling algorithm ADP-1 that achieves goodput $G = 1 - \frac{\rho}{2}\left(1 + \sqrt{1 + \frac{8}{\rho}}\right)$, which is greater than G^* for $\rho < \frac{1}{2}(7 - 3\sqrt{5})$. Then, using a parameterized version of ADP-1 and performing case analysis we show its superiority over the uniform packet strategy for $1/\rho > 4$. Specifically, for $1/\rho > 4$ the algorithm achieves greater goodput than G^*.

1.3 Related Work

Adversarial queueing has been used in wireless networks as a methodology for studying their stability under worst case scenarios, removing the stochastic assumptions usually made for the generation of traffic. It concerns the arrival process of packets in the system and it has been introduced by Borodin et al. [7] as a well defined theoretical model since 2001. It has been further studied by Andrews et al. [3] who emphasized the notion of universal stability in such adversarial settings. A variety of works has then followed, using AQT in different network settings, such as on multiple access channels [10, 11] and routing in communication networks [8, 9]. We associate our constrained type of adversarial channel jams with the AQT model for the arrival process of packets in the following way. Classical AQT considers a *window adversary* that accounts packets being injected within a time window w in such a way that they give a total load of at most wr at each edge of the paths they need to follow, where $w \geq 1$ and $r \leq 1$. In our channel jams, for every window of duration $1/\rho$, there is exactly one new error token available for the adversary to use. In a long execution, considering for example a time interval $T > 1/\rho$, there will be up to $T\rho$ new error tokens available to the adversary.

As stated already, several studies have been done on throughput maximization as well as the effects of jamming in wireless channels. For example, Gummandi et al. [16] consider radio frequency interference on 802.11 networks and show that such networks are surprisingly vulnerable. As a method to withstand these vulnerabilities they propose and analyze a channel hopping design. Tsibonis et al. [24] studied the case of scheduling transmissions to multiple users over a wireless channel with time-varying connectivity and proposed an algorithm that focuses on the weighted sum of channel throughputs, considering saturated packet queues. Thuente et al. [23] studied the effects

of different jamming techniques in wireless networks and the trade-off with their energy efficiency. Their study includes from trivial/continuous to periodic and intelligent jamming (taking into consideration the size of packets being transmitted). On a different flavor, Awerbuch et al. [5] design a MAC protocol for single-hop wireless networks that is robust against adaptive adversarial jamming and requires only limited knowledge about the adversary (an estimate of the number of nodes, n, and an approximation of a time threshold T). One of the differences with our work is that the adversary they consider is allowed to jam $(1 - \varepsilon)$-fraction of the time steps. On a later work [21], Richa et al. explored the design of a robust medium access protocol that takes into consideration the signal to interference plus noise ratio (SINR) at the receiver end. In [22] they extended their work to the case of multiple co-existing networks; they proposed a randomized MAC protocol which guarantees fairness between the different networks and efficient use of the bandwidth. Gilbert et al. [15] worked on a theoretical analysis of the damage that can be introduced by a tiny malicious entity having limited power in the sense that it can only broadcast up to β times. Our model can be viewed as a generalization of this restriction, by allowing recharging. What is more, Pelechrinis et al. [18] present a detailed survey of the Denial of Service attacks in wireless networks. They present the various techniques used to achieve malicious behaviors and describe methodologies for their detection as well as for the network's protection from the jamming attacks. Finally, Dolev et al. [12] present a survey of several existing results in adversarial interference environments in the unlicensed bands of the radio spectrum, discussing their vulnerability. However, none of the models studied considers an AQT modeling of the power of the adversarial entity.

As mentioned in Section 1.1, our adversarial jammer has limited sources of energy and can be used to model, for example, military drones or mobile jammers. Drones or *Unmanned Aerial Vehicles (UAV)* are at the peak of their development. As an upcoming technology that is rapidly improving, it has already attracted the colossi of industry, like Google or Amazon, to invest in UAV research and development, creating even commercial models. There have already been a few research works [13, 17] but the area is still being studied; the work in [13] focuses on UAV's risk analysis and the work in [17] focuses in analyzing cellular network coverage using UAV's and software defined radio. Regarding mobile jammers, in the recent years, many companies have made available battery-operated 3G/4G, WiFi or GPS mobile jammers (e.g., [1, 2]); this market can only increase, as wireless communication is becoming the dominating communication technology.

2 Model

2.1 Network setting

We consider a setting of a sending station (sender) that transmits packets to a receiving station (receiver) over an unreliable wireless channel. The sender has some initial data of size P to be transmitted, and follows some *online scheduling* [20, 19] in order to decide the lengths of the packets to be sent in the transmission. The decisions need to be made during the course of the execution, taking into consideration (or not) the channel jams. Each packet p consists of a *header* of a fixed predefined size h and a

payload of length l chosen by the algorithm. The payload represents the useful data to be sent across the channel and is to be chosen by the sender. The total length of the packet is then denoted by $p_{.len} = h + l$. Note that the total payload from all the packets received successfully by the receiver in the execution must sum up to P. For simplicity and without loss of generality we use $h = 1$ throughout our analysis, and hence $p_{.len} = l + 1$. (Note that l needs not be an integer.) Furthermore, we consider constant bit rate for the channel, which means that the transmission time of each packet is proportional to its length (i.e., a packet of size $l + 1$ takes $l + 1$ time units to be transmitted in full).

2.2 Packet failures

We model the unavailability of the channel to be controlled by the adversary (σ, ρ)-$\mathcal{A}$, which is defined by its two "restrictive" parameters, $\rho \in [0, 1]$ and $\sigma \geq 1$ as follows. The adversary has a token bucket of size σ where it stores "error tokens" and is initially full. From the beginning of the execution and up to a time t, within interval $T = [0, t]$, there will be $\lfloor \rho T \rfloor$ such error tokens created, where ρ is the rate at which they become available to the adversary. In other words, a new error token becomes available at times $1/\rho, 2/\rho, \ldots$. Note that the values of the adversary parameters are given to it (are not chosen by it) and it can only use them in a "smart" way in order to control the packet jams in the channel. If there is at least one token in the bucket, the adversary can introduce an error in the channel and jam the current packet, consuming one token. If the token bucket if full (i.e., there are already σ error tokens in the bucket) and a new token arrives, then one token is lost (this models the fact that a fully charged battery cannot be further charged). As a worst case analysis, we consider that the adversary jams some bit in the header of the packets in order to ensure their destruction. Therefore, adversary (σ, ρ)-$\mathcal{A}$ defines the error pattern as a collection of jamming events on the channel, jamming the packet that is being transmitted in that instant.

2.3 Efficiency measure

For the efficiency of a scheduling algorithm, we look at the *total transmission time*, Tr; that is the time from the beginning of an execution to the moment that the complete payload P has been successfully received. We also look at the *goodput rate*, G; that is the ratio of the total amount of payload successfully transmitted over time, despite the jams in the channel. Note that the goodput rate will eventually be maximized in the long-run, assuming infinite amount of data P. Note also, that in most of our analysis we avoid using *floors* and *ceilings* in order to keep the readability of our results as simple as possible for the reader. Nonetheless, this does not affect the correctness of our results since when being applied on large enough time intervals and data, the "losses" become negligible.

2.4 Feedback mechanism

As for the feedback mechanism, instantaneous feedback to the sender about a packet being received is being considered, as in [4]. We also assume that the notification packets

cannot be jammed by the errors in the channel because of their relatively small size. In particular, we consider notification / acknowledgement messages sent for every packet that is received successfully. If such a message is not received by the sender, then it considers the packet to be jammed.

3 Uniform Packet Length

In this section we explore the case in which all packets are of the same length. Nonetheless, we first make the following observation, which bounds the error availability rates used, being such that they permit some data transmission (this holds also for non-uniform packet lengths).

Observation 1 *Let c be the smallest packet size used by an algorithm (i.e., $\forall p, p_{.len} \geq c$). For any error rate $\rho \geq 1/c$, no goodput larger than zero can be achieved.*

Proof. If the error rate is $\rho \geq 1/c$, a new error token arrives during the transmission of any packet (recall that packets are of size at least c). Hence, there are error tokens in the bucket at all times for the adversary to corrupt all packets being transmitted. Using an error token every c time, is sufficient to keep the goodput at zero. □

From this observation, it can be derived that algorithms that only use packets of length $p_{.len} \geq 1/\rho$ are not interesting. In particular, since in this section we consider an algorithm that systematically sends packets of the same length, we asume that the packets used satisfy $p_{.len} < 1/\rho$.

The main goal for the algorithms to be designed is to minimize the transmission time needed to successfully transmit the total amount of data P to the receiver. Knowing both adversarial parameters, ρ and σ, and considering uniform packets of size $p_{.len} = l + 1 < 1/\rho$, we can find the quasi optimal value for the length of the payload l in each packet that minimizes the transmission time. For simplicity, we will assume that the total length of the data to be transmitted P is a multiple of the payload length l. (For large values of P the error introduced by this assumption is negligible.) Then, the objective is that P/l packets arrive successfully at the receiver.

Let us now derive a *lower bound* on the transmission time that can be achieved using uniform packets. We denote with $Tr(l)$ the transmission time with packets of uniform payload l. Let r be the number of packets jammed and retransmitted by the sender. Then,

$$Tr(l) = (P/l + r)(l + 1). \tag{1}$$

Observe that the last packet transmitted was correctly received, since otherwise the data would have been completely transmitted by time $Tr(l)-(l+1)$, which contradicts the fact that $Tr(l)$ is the transmission time. Hence, the number of packets jammed and retransmitted is upper bounded as

$$r \leq \lceil (Tr(l) - (l + 1))\rho \rceil - 1 + \sigma, \tag{2}$$

where we apply the fact that the last error used by the adversary must have been available before time $Tr(l) - (l + 1)$. We claim that the number of packets jammed by the

adversary and retransmitted is in fact equal to the bound of Eq. 2. Otherwise, the adversary could have jammed the last packet sent (at time $Tr(l) - (l+1)$), achieving a longer transmission time. Hence,

$$r = \lceil (Tr(l) - (l+1))\rho \rceil - 1 + \sigma. \tag{3}$$

Moreover, since the adversary could not jam the last packet sent, it must also hold that $r + 1 \geq Tr(l)\rho + \sigma = (P/l + r)(l+1)\rho + \sigma$, from which we can bound the value of r as

$$r \geq \frac{P\rho(l+1) + (\sigma - 1)l}{l - l\rho(l+1)}. \tag{4}$$

Let us define the lower bound of the transmission time when packets of uniform payload l are used, as function $LB(l)$. Then,

Lemma 1. *Using uniform packets of payload l, the lower bound of the transmission time is*

$$Tr(l) \geq LB(l) = \frac{P + (\sigma - 1)l}{l(1 - \rho(l+1))}(l+1).$$

Proof. Replacing the lower bound of r (Eq. 4) in Eq. 1 we have

$$Tr(l) \geq \left(\frac{P}{l} + \frac{P\rho(l+1) + (\sigma - 1)l}{l - l\rho(l+1)} \right)(l+1) = \frac{P + (\sigma - 1)l}{l(1 - \rho(l+1))}(l+1),$$

which when combined with the definition of $LB(l)$, completes the proof. □

Using Calculus, we can find the payload length l^* that minimizes $LB(l)$, which yields the following theorem.

Theorem 1. *Using uniform packets the transmission time is lower bounded as*

$$Tr \geq LB(l^*) = \frac{P + (\sigma - 1)l^*}{l^*(1 - \rho(l^* + 1))}(l^* + 1)$$

and the goodput is upper bounded as

$$G \leq \frac{P}{LB(l^*)} = \frac{Pl^*(1 - \rho(l^* + 1))}{(P + (\sigma - 1)l^*)(l^* + 1)},$$

where

$$l^* = \frac{\sqrt{P(P\rho + (\sigma - 1)(1 - \rho))} - P\rho}{P\rho + \sigma - 1}.$$

Obviously, when P tends to ∞, so does the transmission time Tr. However, we can derive in this case an upper bound on the goodput as follows.

Corollary 1. *Using uniform packets, the goodput is upper bounded as $G \leq (1 - \sqrt{\rho})^2$, and in the limit as the value of P grows,*

$$G^* = \lim_{P \to \infty} G = (1 - \sqrt{\rho})^2$$

Proof. Using Calculus it can be shown that the upper bound of G obtained in Theorem 1 grows with P. Observe that $\lim_{P\to\infty} G = l^*(1-\rho(l^*+1))/(l^*+1)$ and $\lim_{P\to\infty} l^* = (\sqrt{\rho}-\rho)/\rho = 1/\sqrt{\rho}-1$. Replacing the latter in the former the claims follow. □

We now show a corresponding *upper bound* on the transmission time. We start by combining Eqs. 3 and 1 as follows:

$$\begin{aligned} r &= \lceil (Tr(l)-(l+1))\rho \rceil - 1 + \sigma \\ &< (Tr(l)-(l+1))\rho + \sigma \\ &= ((P/l+r)(l+1)-(l+1))\rho + \sigma \\ &= (P/l+r)(l+1)\rho + \sigma - (l+1)\rho. \end{aligned}$$

This allows us to find an upper bound of r as

$$r < \frac{P\rho(l+1)+(\sigma-(l+1)\rho)l}{l-l\rho(l+1)}. \tag{5}$$

Let us now define the upper bound of the transmission time when packets of payload l are used, as function $UB(l)$. Then,

Lemma 2. *Using uniform packets of payload l, the upper bound of the transmission time is*

$$Tr(l) < UB(l) = \frac{P+(\sigma-(l+1)\rho)l}{l(1-\rho(l+1))}(l+1).$$

Proof. Replacing the upper bound of r (Eq. 5) in Eq. 1 we have

$$Tr(l) < \left(\frac{P}{l} + \frac{P\rho(l+1)+(\sigma-(l+1)\rho)l}{l-l\rho(l+1)}\right)(l+1) = \frac{P+(\sigma-(l+1)\rho)l}{l(1-\rho(l+1))}(l+1),$$

which when combined with the definition of $UB(l)$, completes the proof. □

From Observation 1, $\rho < 1/(l+1)$ must hold. Then, $(l+1)\rho < 1$ and the bound obtained in the above lemma is strictly bigger than the lower bound presented in Lemma 1, as expected. In fact, the gap between bounds can be obtained as shown in the following lemma.

Lemma 3. *Using uniform packets of payload l, the transmission time satisfies* $Tr(l) \in [LB(l), LB(l)+l+1)$.

Proof. Recall that the lower bound $LB(l)$ is obtained in Lemma 1. Subtracting this expression from the upper bound $UB(l)$ presented in Lemma 2, we have

$$\begin{aligned} UB(l) - LB(l) &= \frac{P+(\sigma-(l+1)\rho)l}{l(1-\rho(l+1))}(l+1) - \frac{P+(\sigma-1)l}{l(1-\rho(l+1))}(l+1) \\ &= \frac{l(1-\rho(l+1))}{l(1-\rho(l+1))}(l+1) = l+1. \end{aligned}$$

From the above and the fact that $Tr(l) < UB(l)$ the claim follows. □

Corollary 2. *Using uniform packets of payload l, $Tr(l)$ is the only multiple of $l+1$ that falls in the interval $[LB(l), LB(l)+l+1)$.*

Finally, combining Lemma 3 with Theorem 1 we derive the following theorem.

Theorem 2. *Consider l^* as defined in Theorem 1. Then*

- *the transmission time $Tr(l^*)$ observed is less that l^*+1 (one packet) longer that the optimal. I.e., $Tr(l^*) < Tr + l^* + 1$.*
- *the goodput $G(l^*)$ converges to the optimal goodput G as P grows. Additionally, when P goes to infinity the goodput matches the optimal G^*, i.e. $\lim_{P\to\infty} G(l^*) = \lim_{P\to\infty} G = (1-\sqrt{\rho})^2$.*

Proof. The first claim follow directly from Lemma 3, since the value of l^* is the one that minimizes $LB(l)$. For the second, recall that $G(l^*) = \frac{P}{Tr(l^*)}$. Hence, observing again Lemma 3 we get that

$$G(l^*) > \frac{P}{LB(l^*)+l^*+1} = \frac{1}{\frac{LB(l^*)}{P} + \frac{l^*+1}{P}}.$$

As P grows $\frac{l^*+1}{P}$ tends to 0, making $G(l^*)$ converge to $P/LB(l^*)$ which is an upper bound on the optimal goodput. Finally, as shown in Corollary 1, when P tends to infinity, $P/LB(l^*)$ tends to $(1-\sqrt{\rho})^2$, which completes the proof. □

4 Adaptive Packet Length

As we have shown in the previous section, if all packets have the same size, more precisely size l^*+1, then there is an upper bound on the achievable goodput $G^* = (1-\sqrt{\rho})^2$. In this section, *focusing on the case $\sigma = 1$*, we lift the restriction on uniform packet length and consider an algorithm that adapts the packet length it uses as a function of the observed jams. We show that by using this approach it is possible to achieve a goodput greater than $(1-\sqrt{\rho})^2$, under the restriction of $\rho < 1/4$.

We divide the execution into consecutive periods of length $1/\rho$. In particular, the i^{th} period, $i = 1, 2, \ldots$, spans the time interval $I_i = \left[\frac{i-1}{\rho}, \frac{i}{\rho}\right)$. Note that since error tokens arrive at time instants $1/\rho, 2/\rho, \ldots$ and $\sigma = 1$, at most one packet can be jammed by the adversary in each period. For simplicity, and since we focus on periods of fixed length $1/\rho$, we will use the *useful payload* sent in the period as one of the goodness metrics used, denoted UP. Observe that UP $= G/\rho$ and therefore, the upper bound on the useful payload that can be achieved with uniform packets is UP$^* = (1-\sqrt{\rho})^2/\rho$.

4.1 Algorithm ADP-1 for $\rho < \frac{1}{2}(7-3\sqrt{5})$.

We start by proposing the following algorithm, to be used for small values of ρ (and $\sigma = 1$).

Algorithm ADP-1 Description: Each period starts by scheduling packets of decreasing length $p_{i.len} = Z - i$ for $i = 0, 1, 2, 3 \ldots$. If a packet p_j is jammed during the period, this transmission sequence is stopped, and after p_j, a single more packet is scheduled by the algorithm whose length spans the rest of the period.

We will now show that for ρ small enough, we can specify the parameter Z such that the useful payload achieved in each period is at least UP_u.

Theorem 3. *Adaptive algorithm ADP-1, with* $Z = \frac{1}{2}\left(\sqrt{1+\frac{8}{\rho}} - 1\right)$*, achieves goodput* $G = 1 - \frac{\rho}{2}\left(1 + \sqrt{1+\frac{8}{\rho}}\right)$*. This value is larger than the upper bound for the uniform case if* $\rho < \frac{1}{2}(7 - 3\sqrt{5}) \approx 0.1459$.

Proof. There are two cases to be considered in a period:

(a) If the adversary jams a packet p_j, the useless data sent in the period adds to $Z + 1$. This number comes from the j headers of the packets sent before p_j, plus the length $p_{j.len} = Z - j$ of the packet jammed, plus the header of the last packet sent in the period (which cannot be jammed). Hence, in this case, the useful payload of the period is $1/\rho - (Z + 1)$.

Otherwise, (b) if no packet is jammed, the useless data sent in the period correspond only to the headers of the packets sent. Then, if the last packet sent in the interval is p_k, the useless data is $k + 1$, and the corresponding useful payload is $1/\rho - (k + 1)$. The value Z is chosen so that the total length of the packets sent in this case is equal the length of the interval. From this property, $\sum_{i=0}^{k} p_{i.len} = \frac{1}{\rho}$, the value of Z must satisfy $Z(k+1) - \frac{k(k+1)}{2} = \frac{1}{\rho}$ and hence

$$Z = \frac{k}{2} + \frac{1}{\rho(k+1)}. \tag{6}$$

In a given period the choice of whether case (a) or (b) occurs is up to the adversary, since she can decide which packet to jam, if any. This means that the useful payload achieved will be the minimum of the two cases, $\text{UP} = \min\{1/\rho - (Z + 1), 1/\rho - (k + 1)\}$. Observe from this Eq. 6 that the length Z of the initial packet increases if the number of packets k decreases. Additionally, it must hold that $Z \geq k$ and therefore UP is maximized when when $Z = k$. Hence, the optimal k is the suitable solution of the equation $k = \frac{k}{2} + \frac{1}{\rho(k+1)}$, which is $k = \frac{1}{2}\left(\sqrt{1+\frac{8}{\rho}} - 1\right) = Z$.

The useful payload achieved is then $\text{UP} = \frac{1}{\rho} - \left(\frac{1}{2}\sqrt{1+\frac{8}{\rho}} - \frac{1}{2} + 1\right) = \frac{1}{\rho} - \frac{1}{2}\left(\sqrt{1+\frac{8}{\rho}} + 1\right)$, which is more that $\text{UP}^* = (1 - \sqrt{\rho})^2/\rho$ for $\rho < \frac{1}{2}(7 - 3\sqrt{5}) \approx 0.1459$. The corresponding goodput is $G = \frac{\text{UP}}{1/\rho} = 1 - \frac{\rho}{2}\left(\sqrt{1+\frac{8}{\rho}} + 1\right)$. □

Corollary 3. *Adaptive algorithm ADP-1, with* $Z = \frac{1}{2}\left(\sqrt{1+\frac{8}{\rho}} - 1\right)$ *achieves transmission time* $Tr = \frac{2P}{2-\rho-\sqrt{\rho(\rho+8)}}$.

4.2 Exhaustive Case Study for $\rho \geq \frac{1}{2}(7 - 3\sqrt{5})$.

From the above results, we see that in the case of $\sigma = 1$, instead of using packets of uniform length $l^* + 1$, it is better to use an adaptive algorithm. More precisely, we have shown that for $\rho < \frac{1}{2}(7 - 3\sqrt{5})$, ADP-1 achieves a better useful payload and goodput rate than the optimal uniform packet algorithm (the one that uses packet length $p_{.len} = l^* + 1$). We now explore the case of $\rho \geq \frac{1}{2}(7 - 3\sqrt{5})$. As before, we look at periods of length $1/\rho$, which means that the length of the period is at most $\frac{2}{7-3\sqrt{5}} \approx 6.85 < 7$. Hence, we consider only periods of such lengths.

In general, we are going to deal with subintervals of the period of length $1/\rho$. We will denote with $T = [t, t')$ an interval in the execution (subinterval of the period) such that t is an instant at which the adversary has one error token in the error bucket, and t' the time instant at which the next error token becomes available. Hence, the adversary has one error token (and only one) to be used in T. We use $|T|$ to denote the length of the interval, and UP_T to denote the useful payload that has been sent and correctly received by the receiver during T.

Let us first make the following observation.

Observation 2 *If there is at most one packet p of length $p_{.len} > 1$ sent in an interval T, then $UP_T = 0$.*

Proof. Since the adversary has one error token at the beginning of the interval, it uses it to jam packet p. The rest of packets (if any) have length 1 and carry no payload. □

We consider now different cases depending on the length of the interval, $|T|$, to be explored. We use the following algorithm for any interval T.

Algorithm ADP-1$_T$ Description: As a base case, if $|T| < 2$ then ADP-1$_T$ simply sends a packet that spans the whole interval. Otherwise, let i the integer such that $|T| \in [i, i+1)$. Then ADP-1$_T$ sends a packet p whose length depends on i. If p is jammed, it sends a packet p' that spans the rest of the interval T. Otherwise, it applies recursively algorithm ADP-1$_{T'}$ to the interval $T' = [t + p_{.len}, t')$. Observe that $|T'| < i$.

Lemma 4. *If $|T| < 2$, then $UP_T = 0$.*

Proof. For any packet sent, the header requires 1 unit of length. Since $|T| < 2$, it means that only one packet can be sent within T. Hence, $\text{UP}_T = 0$ from Observation 2. □

Lemma 5. *If $|T| \in [2, 3)$, Algorithm ADP-1$_T$ uses uniform packets with $p_{.len} = |T|/2$ and achieves useful payload $UP_T = \frac{|T|}{2} - 1$. The packets used in such interval are uniform.*

Proof. First observe that the algorithm essentially sends two packets of length $|T|/2$. This in fact achieves useful payload $\text{UP}_T = \frac{|T|}{2} - 1$, since the adversary has only one error token to be used in T, and it jams only one packet. No matter which one is jammed, the payload of the unjammed packet, whose length is $\frac{|T|}{2} - 1$, is received correctly.

We show now that this is in fact the best possible useful payload that ADP-1$_T$ can achieve for period T. Since $|T| < 3$ and the header has length one, the algorithm cannot send more than 2 packets. Consider an algorithm ALG that:

– First sends a packet p of length larger than $|T|/2$. Then, the adversary jams p. Since the length of the rest of the interval is $|T| - p_{.len} < |T|/2$, the useful payload $\text{UP}_T < \frac{|T|}{2} - 1$.
– First sends a packet p of length smaller than $|T|/2$ (but at least 1). Then, the adversary does not jam p. After sending p, until the end of T there is a subinterval T' of length $|T'| = |T| - p_{.len} < 2$. From Lemma 4, the useful payload of T' is $\text{UP}_{T'} = 0$. Then, the useful payload of T is $\text{UP}_T = p_{.len} - 1 < \frac{|T|}{2} - 1$.

In both cases the useful payload of ALG is smaller than the one achieved by the algorithm proposed. Hence, the algorithm proposed gives the best possible useful payload for an interval T, where $|T| \in [2, 3)$. □

Lemma 6. *If $|T| \in [3, 4)$, Algorithm ADP-1$_T$ uses uniform packets with $p_{.len} = |T|/2$ and achieves useful payload $UP_T = \frac{|T|}{2} - 1$. The packets used in such interval are uniform.*

Proof. The proof is similar to that of the previous lemma, with a small difference. In the case that algorithm ALG sends a packet with length $p_{.len} < |T|/2$, the adversary does not jam p and after it is received, there is a subinterval T' of length $|T'| = |T| - p_{.len} < 3$ until the end of T. From Lemmas 4 and 5, the useful payload of T' is upper bounded as $\text{UP}_{T'} \leq \frac{|T'|}{2} - 1 = \frac{|T|-p_{.len}}{2} - 1$. Then, the useful payload of T is $\text{UP}_T \leq p_{.len} - 1 + \frac{|T|-p_{.len}}{2} - 1 = \frac{|T|+p_{.len}}{2} - 2 < \frac{|T|+|T|/2}{2} - 2$, which is smaller than $\frac{|T|}{2} - 1$ for $|T| < 4$. Hence, the algorithm proposed gives the best possible useful payload for an interval T, where $|T| \in [3, 4)$. □

Lemma 7. *If $|T| \in [4, 5)$, Algorithm ADP-1$_T$ with $p_{.len} = (|T| + 2)/3$ achieves useful payload $UP_T = \frac{2|T|-5}{3}$. The packets used in the whole interval are not uniform in this case.*

Proof. Let Algorithm ADP-1$_T$ send first packet p with $p_{.len} = (|T| + 2)/3$. If it is jammed, a packet p' of length $|T| - (|T| + 2)/3$ is sent successfully. Then, in this case the useful payload is $\text{UP}_T = |T| - (|T| + 2)/3 - 1 = \frac{2|T|-5}{3}$. Otherwise, observe that $|T'| = |T| - p_{.len} \in [2, 4)$. Then, form Lemmas 5 and 6 the $\text{UP}_{T'} = \frac{|T'|}{2} - 1 = \frac{|T|-p_{.len}}{2} - 1$. Hence, $\text{UP}_T = p_{.len} - 1 + \frac{|T|-p_{.len}}{2} - 1 = \frac{2|T|-5}{3}$.

To prove that this is the best approach for the choice of the packet length, consider an algorithm ALG that

– First sends a packet p of length larger than $(|T| + 2)/3$. Then, the adversary jams p. Since the length of the rest of the interval is $|T| - p_{.len} < |T| - (|T| + 2)/3$, the useful payload $\text{UP}_T < |T| - (|T| + 2)/3 = \frac{2|T|-5}{3}$.
– First sends a packet p of length smaller than $(|T| + 2)/3$, but at least 1. Then, the adversary does not jam p. After p there is a subinterval T' of length $|T'| = |T| - p_{.len} < 4$. Then, from Lemmas 4, 5, and 6, the useful payload of T' is upper bounded as $\text{UP}_{T'} \leq \frac{|T'|}{2} - 1 = \frac{|T|-p_{.len}}{2} - 1$. Then, the useful payload of T is $\text{UP}_T = p_{.len} - 1 + \frac{|T|-p_{.len}}{2} - 1 < \frac{2|T|-5}{3}$.

In both cases the useful payload is smaller than the ones achieved by the algorithm proposed. Hence, the algorithm proposed with the packet length chosen, gives the best possible useful payload in an interval T, where $|T| \in [4, 5)$. □

Lemma 8. *If $|T| \in [5, 6)$, Algorithm ADP-1$_T$ with $p_{.len} = (|T|+2)/3$ achieves useful payload UP$_T = \frac{2|T|-5}{3}$. The packets used in the whole interval are not uniform in this case.*

Proof. The proof is similar to that of lemma 7, with some small differences. The main difference is in the case that algorithm ALG sends a packet with length $p_{.len} < (|T| + 2)/3$. As above, the adversary will not jam p and after sending it successfully, there will be a subinterval T' of length $|T'| = |T| - p_{.len} < 5$ until the end of T. Then, from Lemmas 4 to 7, the useful payload of T' is upper bounded as $\text{UP}_{T'} \leq \frac{2|T'|-5}{3} = \frac{2(|T|-p_{.len})-5}{3}$. Hence, the useful payload of T becomes $\text{UP}_T \leq p_{.len} - 1 + \frac{2(|T|-p_{.len})-5}{3}$ which is smaller than $\frac{2|T|-5}{3}$ for $p_{.len} < 3$. The latter holds, since $p_{.len} < (|T| + 2)/3$ and $|T| < 6$. Hence again, the algorithm proposed with the packet length chosen, gives the best possible useful payload in an interval T, where $|T| \in [5, 6)$. □

Lemma 9. *If $|T| \in [6, 7)$, Algorithm ADP-1$_T$ with $p_{.len} = (|T|+2)/3$ achieves useful payload UP$_T = \frac{2|T|-5}{3}$. The packets used in the whole interval are not uniform in this case either.*

Proof. The proof follows the same exact logic as lemmaa 7 and 8. The only difference is in the case that algorithm ALG sends a packet with length $p_{.len} < (|T| + 2)/3$. As above, the adversary will not jam p and after sending it successfylly, the subinterval T' that remains is of length $|T'| = |T| - p_{.len} < 6$. Then, from Lemmas 4 to 8, the useful payload of T' is upper bounded as $\text{UP}_{T'} \leq \frac{2|T'|-5}{3} = \frac{2(|T|-p_{.len})-5}{3}$. Hence, the useful payload of T becomes $\text{UP}_T \leq p_{.len} - 1 + \frac{2(|T|-p_{.len})-5}{3}$ which is smaller than $\frac{2|T|-5}{3}$ for $p_{.len} < 3$. The latter holds, since $p_{.len} < (|T| + 2)/3$ and $|T| < 7$. Hence, the algorithm proposed with the packet length chosen, gives the best possible useful payload in an interval T, where $|T| \in [6, 7)$. □

Putting all these results together, and fixing $|T| = 1/\rho$, we get the following theorem.

Theorem 4. *For $\sigma = 1$, $\rho \geq \frac{1}{2}(7-3\sqrt{5})$ and $1/\rho \in [4, 7)$, adaptive algorithm ADP-1$_T$ has goodput $G = \frac{2-5\rho}{3}$. This is achieved using first packet p with length $p_{.len} = \frac{1}{3\rho} + \frac{2}{3}$; the packets used are not of uniform length.*

Note that for $1/\rho > 4$, the goodput achieved is bigger than the upper bound of the uniform packet approach, $G > G^*$, and for $1/\rho = 4$ it is equal to the upper bound, $G = G^*$.

5 Conclusions

In this paper we have applied Adversarial Queuing Theory (AQT), a well known theoretical modeling tool, for the first time to restrict adversarial packet jamming on wireless networks. We have chosen to study a constrained adversarial entity, considering a

bounded error-token capacity σ and an error-token availability rate ρ. This model could be applied in various battery-operated malicious devices such as drones or mobile jammers. We have first shown upper and lower bounds on transmission time and goodput by exploring the case of uniform packet lengths. Then, focusing on $\sigma = 1$, we have shown that an adaptive algorithm that changes the packet length based on feedback received for jammed packets, can achieve better goodput and transmission time. What might seem surprising is that even for the "simple" case of $\sigma = 1$, the analysis of the adaptive algorithm is nontrivial, and imposes constraints also on ρ.

An intriguing open question is whether it is still possible to obtain better efficiency than the uniform packet lengths "policy" for adaptive algorithms with $\sigma > 1$. Considering for example $\sigma = 2$ seems to already be a challenging task. Another interesting future direction is to investigate the case where one or both parameters ρ and σ are not known; here one will need to monitor the history of the observed jams in an attempt to estimate these parameters. On the other hand, the adversary will try to "hide" the true value of these parameters, yielding an interesting gameplay between the adversary and an algorithm. Another direction to follow would be to consider in addition the channel errors due to congestion and transmission rate.

Acknowledgments. The authors would like to thank Dariusz Kowalski and Joerg Widmer for many fruitful discussions. We would also like to thank the anonymous reviewers for their constructive comments and suggestions that helped us improve the presentation of our work.

References

1. http://alljammer.com/ (last accessed: April 8, 2015).
2. http://www.jammer-store.com/ (last accessed: April 8, 2015).
3. Matthew Andrews, Baruch Awerbuch, Antonio Fernández, Tom Leighton, Zhiyong Liu, and Jon Kleinberg. Universal-stability results and performance bounds for greedy contention-resolution protocols. *Journal of the ACM (JACM)*, 48(1):39–69, 2001.
4. Antonio Fernández Anta, Chryssis Georgiou, Dariusz R Kowalski, Joerg Widmer, and Elli Zavou. Measuring the impact of adversarial errors on packet scheduling strategies. In *Structural Information and Communication Complexity (SIROCCO)*, pages 261–273. Springer, 2013.
5. Baruch Awerbuch, Andrea Richa, and Christian Scheideler. A jamming-resistant mac protocol for single-hop wireless networks. In *Proceedings of the Twenty-seventh ACM Symposium on Principles of Distributed Computing*, PODC '08, pages 45–54, New York, NY, USA, 2008. ACM.
6. Pravin Bhagwat, Partha Bhattacharya, Arvind Krishna, and Satish K Tripathi. Enhancing throughput over wireless lans using channel state dependent packet scheduling. In *INFOCOM'96. Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation. Proceedings IEEE*, volume 3, pages 1133–1140. IEEE, 1996.
7. Allan Borodin, Jon Kleinberg, Prabhakar Raghavan, Madhu Sudan, and David P Williamson. Adversarial queuing theory. *Journal of the ACM (JACM)*, 48(1):13–38, 2001.
8. Bogdan S Chlebus, Vicent Cholvi, and Dariusz R Kowalski. Stability of adversarial routing with feedback. In *Networked Systems*, pages 206–220. Springer, 2013.

9. Bogdan S Chlebus, Vicent Cholvi, and Dariusz R Kowalski. Universal routing in multi hop radio networks. In *Proceedings of the 10th ACM international workshop on Foundations of mobile computing*, pages 19–28. ACM, 2014.
10. Bogdan S Chlebus, Dariusz R Kowalski, and Mariusz A Rokicki. Adversarial queuing on the multiple-access channel. In *Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, pages 92–101. ACM, 2006.
11. Bogdan S Chlebus, Dariusz R Kowalski, and Mariusz A Rokicki. Stability of the multiple-access channel under maximum broadcast loads. In *Stabilization, Safety, and Security of Distributed Systems*, pages 124–138. Springer, 2007.
12. Shlomi Dolev, Seth Gilbert, Rachid Guerraoui, Dariusz R Kowalski, Calvin Newport, Fabian Kohn, and Nancy Lynch. Reliable distributed computing on unreliable radio channels. In *Proceedings of the 2009 MobiHoc S 3 workshop on MobiHoc S 3*, pages 1–4. ACM, 2009.
13. Michelle S Faughnan, Brian J Hourican, G Collins MacDonald, Megha Srivastava, JA Wright, YY Haimes, E Andrijcic, Zhenyu Guo, and JC White. Risk analysis of unmanned aerial vehicle hijacking and methods of its detection. In *Systems and Information Engineering Design Symposium (SIEDS), 2013 IEEE*, pages 145–150. IEEE, 2013.
14. Zhenghua Fu, Petros Zerfos, Haiyun Luo, Songwu Lu, Lixia Zhang, and Mario Gerla. The impact of multihop wireless channel on tcp throughput and loss. In *INFOCOM 2003. Twenty-second annual joint conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1744–1753. IEEE, 2003.
15. Seth Gilbert, Rachid Guerraoui, and Calvin Newport. Of malicious motes and suspicious sensors: On the efficiency of malicious interference in wireless networks. *Theoretical Computer Science*, 410(6):546–569, 2009.
16. Ramakrishna Gummadi, David Wetherall, Ben Greenstein, and Srinivasan Seshan. Understanding and mitigating the impact of rf interference on 802.11 networks. *ACM SIGCOMM Computer Communication Review*, 37(4):385–396, 2007.
17. Michal Jakubiak. Cellular network coverage analysis using uav and sdr. Master's thesis, Tampere University of Technology, 2014.
18. Konstantinos Pelechrinis, Marios Iliofotou, and Srikanth V Krishnamurthy. Denial of service attacks in wireless networks: The case of jammers. *Communications Surveys & Tutorials, IEEE*, 13(2):245–257, 2011.
19. Kirk Pruhs. Competitive online scheduling for server systems. *ACM SIGMETRICS Performance Evaluation Review*, 34(4):52–58, 2007.
20. Kirk Pruhs, Jiri Sgall, and Eric Torng. Online scheduling. *Handbook of scheduling: algorithms, models, and performance analysis*, pages 15–1, 2004.
21. Andrea Richa, Christian Scheideler, Stefan Schmid, and Jin Zhang. Towards jamming-resistant and competitive medium access in the sinr model. In *Proceedings of the 3rd ACM workshop on Wireless of the students, by the students, for the students*, pages 33–36. ACM, 2011.
22. Andrea Richa, Christian Scheideler, Stefan Schmid, and Jin Zhang. Competitive and fair throughput for co-existing networks under adversarial interference. In *Proceedings of the 2012 ACM symposium on Principles of distributed computing*, pages 291–300. ACM, 2012.
23. David Thuente and Mithun Acharya. Intelligent jamming in wireless networks with applications to 802.11 b and other networks. In *Proc. of MILCOM*, volume 6, 2006.
24. Vagelis Tsibonis, Leonidas Georgiadis, and Leandros Tassiulas. Exploiting wireless channel state information for throughput maximization. *Information Theory, IEEE Transactions on*, 50(11):2566–2582, 2004.
25. Elif Uysal-Biyikoglu, Balaji Prabhakar, and Abbas El Gamal. Energy-efficient packet transmission over a wireless link. *IEEE/ACM Trans. Netw.*, 10(4):487–499, August 2002.