Electing a Leader in Wireless Networks Quickly Despite Jamming

Marek Klonowski
Department of Computer Science at the Faculty of Fundamental Problems of Technology
Wrocław, Poland
marek.klonowski@pwr.wroc.pl

Dominik Pająk
Computer Laboratory
University of Cambridge
Cambridge, UK
dsp39@cl.cam.ac.uk

ABSTRACT

In this paper we present a fast leader election protocol for single-hop wireless networks provably robust against jamming by an external and powerful adversary. A $(T, 1-\varepsilon)$-bounded adversary can jam at most $(1-\varepsilon)w$ out of any $w \geq T$ contiguous time slots, for $0 < \varepsilon < 1$. The network consists of n stations that do not have the knowledge of **any** global parameter n, T, ε. Each station can transmit or listen to the common communication channel. In each slot, all listeners are notified in which of three states the communication channel is in the current slot: no transmitters, exactly one transmitter or at least two transmitters. To the listening stations, a jammed slot is indistinguishable from the case of at least two transmitters.

Our protocol elects a leader, with high probability, in the presence of an arbitrary, adaptive $(T, 1-\varepsilon)$-adversary. For any constant ε and $T = \mathcal{O}(\log n)$, the protocol works in optimal time $\mathcal{O}(\log n)$. The protocol also works for general T and ε in $\mathcal{O}\left(\frac{\log\log(1/\varepsilon)}{\varepsilon^3}\log n\right)$, slots if $T \leq \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$ and $\mathcal{O}\left(\max\left\{\log\log\left(\frac{T}{\varepsilon \log n}\right), \log(1/\varepsilon)\log\log(1/\varepsilon)\right\} T\right)$ otherwise.

General Terms

Algorithms, Reliability, Theory

Keywords

Robustness, Leader election, Wireless network

1. INTRODUCTION

A wireless network is a set of stations working without any central control and communicating using a common wireless interface. Due to the lack of central arbiter, electing a leader in such a distributed system is a challenging task. Despite dozens of protocols devoted to this problem has been presented, some particular yet important from practitioners' point of view, issues are left undressed. In this paper we present and analyze a leader election protocol immune against a powerful adversary that is capable of jamming the common communication channel used by all the stations of the wireless network. Despite similarities to numerous previous results, to the best of our knowledge, our contribution cannot be realized by any of them by any straightforward manner mainly due to very restricted model - on one hand we assume a presence of a very strong, adaptive adversary, and on the other hand the stations have only local knowledge. In particular they do not know the size of the network (or even its bound) nor parameters governing the strength of the adversary.

The assumed model of the adversary covers a significant number of threads spanning from random faults generated by incidental transmissions of coexisting networks to a powerful malicious adversary capable of jamming the network unpredictable number of times in an adaptive manner in order to perform a DoS-like attack. Note that such type of attacks can be launched without any special hardware by listening to the open channel and broadcasting at the same frequency band as the network.

1.1 Our Model

We consider a wireless network consisting of a set of n honest and reliable stations. Each station is placed within transmission (and interference) range of each other - that is, we assume a *single-hop* network. Time is divided into discrete steps called *slots* and nodes are synchronized as they had an access to a global clock. Stations communicate via a single channel as the only medium. In a single slot each station can transmit a message or sense (listen to) the channel. For simplicity, we assume that each station that is not transmitting in a given slot, is listening. If a single station transmits all listening stations receive the message (and the state of the channel is described as Single). If two or more stations transmit the state of the channel is described as Collision. The third state is Null, while the channel is idle in a given slot (i.e. none of stations transmit). We assume that is listening to the channel receives the state of the channel in the current slot. The ability of the stations to distinguish between Null and Collision is sometimes called *collision detection.* We will consider two variants of collision detection: *weak collision detection* (weak-CD) and *strong collision detection* (strong-CD). In strong-CD stations are assumed to be capable of transmitting and listening at the same time. In each slot all stations receive the current state of the channel. On the other hand in weak-CD parallel listening and transmitting is not possible and only stations which are not transmitting are notified about the state of the channel. Clearly even in weak-CD, the transmitters have a partial knowledge about the state of the channel because they know that there is either Single or Collision. Let us stress however that the main result of our paper holds for more demanding weak-CD.

In literature also a model without collision detection (no-CD) is considered. There, the channel can be only in two states: Single in which there is exactly one transmitter or no-Single in which there is either zero or at least two transmitters.

The disruptions of the communication are made by an adversary who knows the entire history of the channel and the protocol executed by the honest stations. Additionally, the real number of stations, which is unknown to the stations in advance, may be known by the adversary. All this knowledge can be used by the adversary to jam particular slots so that his disruptions will inflict the largest possible damage to the algorithm executed by the honest stations. Moreover we do **not** assume that the stations share any secret unknown to the adversary as we would like to avoid special initialization of the system required for distribution of such a secret.

The adversary can jam at most $(1-\varepsilon)w$ out of any $w \geq T$ contiguous slots. Note that the adversary can block even all slots in a short windows of less than T slots. Such a model of adversary is known in literature as $(T, 1-\varepsilon)$-bounded (e.g., [3]). We assume that the stations cannot distinguish between the adversarial jamming or a collision of two or more messages that are sent at the same time by honest stations. The adversary is adaptive - the decision if to jam a given slot may depend of the state of the channel in previous slots. Note however that it has to make a jamming decision before it knows the actions of the nodes in the current slot.

We consider a leader election problem that consist in assigning *leader* or *non-leader* status to each station in such way that exactly one station is a leader and all stations knows its status after completing the protocol. This goal is always realized in a such way that a single station has to transmit and is later notified that has it successfully transmitted.

A specific class of algorithms widely studied in literature are so-called *uniform algorithms* ([21]). In each slots of such an algorithm, each station of the network transmits with the same probability independently of other stations. The probability for each slot may depend on the history of the channel.

1.2 Our Results and Organization of the Paper

In the subsection below we present previous work in leader election and other related problems.

Section 2 is devoted to a basic leader election protocol called `AWDP` in a simplified model, wherein we assume strong-CD model and the global knowledge of the parameter ε. `AWDP` works in time $O(\max\{T, \log(1/\varepsilon)/\varepsilon^3 \log n\})$ with high probability and is immune against any $(T, 1-\varepsilon)$-bounded adversary. We also show a lower bound of $\Omega(\max\{T, 1/\varepsilon \log n\})$ for leader election algorithms working with high probability. This shows that `AWDP` is optimal for any T and any constant ε. In Section 2.3 we show, that `AWDP` can be modified to work with unknown ε for the price of a small overhead. We introduce algorithm `UnknownParameters` which works for any unknown ε and T in time $\mathcal{O}\left(\frac{\log\log(1/\varepsilon)}{\varepsilon^3} \log n\right)$, if $T \leq \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$ and $\mathcal{O}\left(\max\left\{\log\log\left(\frac{T}{\varepsilon \log n}\right), \log(1/\varepsilon)\log\log(1/\varepsilon)\right\} T\right)$ otherwise.

In Section 3 we show a method of translating our results from previous section into a weak-CD model with only a constant-factor overhead. Both algorithms `AWDP` with known ε and `UnknownParameters` with unknown ε can be applied to weak-CD using this method.

In Section 4 we conclude the paper and present some open problems in the area of robust algorithms in wireless networks.

1.3 Related and Previous Work

There is a well developed body of literature devoted to jamming and leader election. Our model of the network and adversary is the same as the one in [3]. Authors of [3] present a general Medium Access Control (MAC) protocol achieving a constant throughput in single-hop wireless networks under $(T, 1-\varepsilon)$-bounded adaptive adversary jamming the communication channel. Leader election is one of the applications of their general framework. Algorithm from [3] was the first evidence that it is feasible to solve classical problems in wireless networks even when the fraction of non-jammed slots can be arbitrarily close to 0. In our paper we would like to show that it is even possible to solve such problems efficiently. In particular for arbitrary constant $\varepsilon < 1$ and $T = O(\log n)$ our protocol needs only $O(\log n)$ slots to elect a leader with high probability whereas the algorithm from [3] has proven runtime of $O(\log^4 n)$. For constant ε and very large T our protocol needs time $O(T \log\log T)$ improving the known bound of $O(T \log T)$ [3]. More importantly, our protocol does not need any information about global parameters, while in the previous paper the parameter $\gamma = O(1/(\log\log n + \log T)$ is explicitly used by the stations. On the other hand we do not present the analysis of energy-efficiency of the presented methods - we expect, however that the energetic efficiency of our protocol should be similar to the leader election from [3].

Leader election is a fundamental problem of distributed systems. In the context of wireless radio networks (or equivalent/similar models) it has been described in many notable papers including [2, 19, 18, 21, 25, 15]. In [13] authors discus leader election from perspective of energy consumption. A survey of classic leader election protocols for radio networks can be found in [20] or in Chapter 8 of [11]. The leader election is closely linked to problem of *selection resolution* in wireless networks. In this problem, the goal of the stations is to obtain the first Single in the channel. In CD model, the selection resolution can be solved in expected time $O(\log\log n)$ [25] and this time is optimal [23]. With probability at least $1 - 1/f$, time $O(\log\log n + \log f)$ is necessary and sufficient [18]. In no-CD model, the problem can be solved with high probability in time $O(\log^2 n)$ [19] and a lower bound $\Omega(\log n \log f)$ holds for any algorithm working with probability at least $1 - 1/f$ [9]. The selection resolution in strong-CD model is equivalent to leader election, as the station that transmits successfully can be chosen to be a leader. Also without an adversary, any algorithm working in strong-CD can be simulated using weak-CD by executing the algorithm in odd time slots and using the even time slots to notify about each Single. However in the presence of an adversary, an algorithm for selection resolution in weak-CD does not immediately translate to an algorithm for leader election.

The problem of leader election is considered under various adversarial model. In [10, 14] authors consider *inner* adversary controlling some stations. The aim of the adversary is to force the network to elect one of adversarial stations as the leader. Authors present an algorithm with execution time $\Theta(n^3)$ that partially protects against such adversary, where n is the size of the network. One can also point some other papers about leader election or some wider class of protocols in wireless network under adversary jamming the communication. Some theoretical and experimental results can be found in [24, 1, 7]. In all those papers the assumed model is different than in our paper (or in [3]) or the analysis is based on experimental results. In [16] authors demonstrate that in practice it is very hard to detect jamming adversary. Moreover in [4] authors present how a standard IEEE 802.11 protocols can be efficiently attacked by an adaptive adversary with moderately limited energy and without any special hardware. Some general countermeasures against jamming based on physical methods can be found in [22, 17, 5]. However it turned out that such approach cannot be efficiently combined with popular technologies like 802.11 protocols. In [8] one can find some more theoretical approach to this issue. The leader election problem has been also considered from self-stability perspective (i.e., [1, 6]) It turned out however that the described approach does not provide immunity against adversarial jamming.

Some further references can be found in [26, 3]

2. LEADER ELECTION IN STRONG-CD

In this section we present an algorithm for $(T, 1-\varepsilon)$ -bounded adversary for an (easier) strong-CD model.

In [3], the authors present a solution that is immune against any (T, ε)-bounded adversary. Stations in their algorithm ignore all Collisions and the decisions are made based only on Nulls and Singles. Such an approach is immune against an adversary since we can be certain that slots with Nulls and Singles were not influenced by the adversary. We would like propose a different approach. We will make use of the Collisions but we will value Nulls much more than Collisions. Informally speaking, each Collision will be "worth" 1 and each Null will be "worth" $1/\varepsilon$. Clearly to use this intuition, our protocol will need to initially know the value of ε. Moreover we assume that successfully transmitting stations immediately receives a feedback from the channel (i.e. we assume strong *collision detection* model). In the next sections both assumptions can be relaxed for the price of a small overhead.

2.1 Protocol description

In this section we present algorithm `AWDP` for leader election in strong-CD with known ε. The following procedure `Broadcast` is a basic primitive in our algorithm.

Function 1 `Broadcast`(u)

```
transmit with probability 2^{-u}
return the status of the channel
```

In algorithm `AWDP` we will maintain variable u which will serve as an estimate of $\log_2 n$. In each slot each station will transmit independently with probability 2^{-u}. Our objective is to make the value of u close to $\log_2 n$ as the transmission with probability $1/n$ results in Single with highest probability. Intuitively, when each station transmits with probability 2^{-u} and the state of the channel is Null then one can suspect that the estimate is too big and it should be decreased to increase the probability of Single. Conversely, each Collision should yield the increase of estimate u. However a $(T, 1-\varepsilon)$-bounded adversary with $\varepsilon < 1/2$ can jam more than a half of the slots. Thus to make the protocol robust even against such a powerful adversary we cannot use symmetric changes or the adversary could force the estimate u to diverge to infinity. Instead, knowing that roughly at most ε fraction of all slots is not-jammed, our algorithm increases the estimate u by only $\varepsilon/8$ upon each Collision. Then each Null that decreases the estimate by 1 is sufficient to "neutralize" around $8/\varepsilon$ jammed slots. Indeed, we take advantage of the fact that the adversary cannot induce a Null to the channel and only "one-side errors" are possible.

Function 2 `AWDP` (ε)

```
a ← 8/ε
u ← 0
state ← Collision
repeat
   state ← Broadcast(u)
   if status = Null then
      u ← max(u − 1, 0)
   else if status = Collision then
      u ← u + 1/a
until state ≠ Single
```

Observe that algorithm `AWDP` is a *uniform* one as in each slot each station is transmitting with the same probability 2^{-u}.

2.2 Analysis

Let us note that the variable u in consecutive slots is performing a form of a biased random walk on a discrete subset of real numbers. The walk (and the algorithm) is completed if finally a single station transmits and becomes a leader. In our analysis we would like to show that with high probability, the value of u is in a close proximity of $\log_2 n$ for a significant number of slots, independently how the adversary is acting.

Let us consider an execution of the algorithm such that the leader is not chosen in the first t slots. We divide the t slots of the execution into several groups depending on the value of u at the beginning of each slot and the state of the channel. Let us define the groups and the respective *counters* of the slots belonging to each group. Let $u_0 = \log_2 n$ denote the exact value of the estimator.

IS - is the number of *irregular silences*, i.e. slots, such that $u \leq u_0 - \log_2(2 \ln a)$ and the state of the channel is Null.

IC - is the number of *irregular collisions*, i.e. slots such that $u \geq u_0 + \frac{1}{2}\log_2 a$ and the state of the channel is Collision and the adversary does **not** jam the channel.

CS - is the number of *correcting silences*, i.e. slots, such that $u \geq u_0 + \frac{1}{2}\log_2 a + 1$ and the state of the channel is Null.

CC - is the number of *correcting collisions*, i.e. slots, such that $u \le u_0 - \log_2(2\ln a)$ and the state of the channel is Collision and the adversary does **not** jam the channel.

E - number of slots jammed by the adversary.

R - all other, called **regular** slots.

Note that the value of u at the beginning of each regular slot satisfies $u_0 - \log_2(2\ln a) \le u \le u_0 + \frac{1}{2}\log_2 a + 1$.

In the following auxiliary lemma we prove several relations between the probability of Null, Single and Collision in a slot and the probability p of transmission by each station. Recall that in a fixed time slot, each station transmits to the channel independently with the same probability p.

LEMMA 2.1. *Let $p = \frac{1}{xn}$ be the probability of transmission in a single slot by each station for some $n > 1$ and $x > 0$.*

1. $\mathbb{P}[\mathsf{Null}] \le e^{-\frac{1}{x}}$.
2. $\mathbb{P}[\mathsf{Collision}] \le \frac{1}{x^2}$.
3. $\mathbb{P}[\mathsf{Single}] \ge \frac{1}{x}e^{-\frac{1}{x}}$.
4. $\mathbb{P}[\mathsf{Single}] \ge \frac{1}{x} - \frac{1}{x^2}$.

PROOF. Knowing that $\mathbb{P}[\mathsf{Null}] = (1-p)^n$, $\mathbb{P}[\mathsf{Single}] = np(1-p)^{n-1}$, $\mathbb{P}[\mathsf{Collision}] = 1 - \mathbb{P}[\mathsf{Null}] - \mathbb{P}[\mathsf{Single}]$, all above statements can be derived using the following two double inequalities:

$$\left(1 - \frac{1}{k}\right)^k \le \frac{1}{e} \le \left(1 - \frac{1}{k}\right)^{k-1},$$

$$1 + ky \le (1+y)^k \le 1 + ky + \frac{ky^2}{2},$$

which are true for $k \ge 1$ and $y > -1$. The latter double inequality can be shown by a straightforward induction. □

Using the previous lemma we can show upper bounds on the probability of a slot being an irregular silence or an irregular collision.

LEMMA 2.2. *For any number of stations n and any slot of algorithm AWDP:*

1. *the slot is an irregular silence with probability at most $1/a^2$, independently of the state of the channel in previous rounds,*
2. *the slot is an irregular collision with probability at most $1/a$, independently of the state of the channel in previous rounds .*

PROOF. Observe that a slot is an irregular silence if $u \le u_0 - \log_2(2\ln a)$ thus the transmission probability p of each station satisfies $p \ge 2\ln a/n$. The first point now follows from Lemma 2.1 (p.1). A slot is an irregular collision if $u \ge u_0 + \frac{1}{2}\log_2 a$. The probability of transmission satisfies in this case $p \le 1/(n\sqrt{a})$ and the second point follows from Lemma 2.1 (p.2). □

In the following lemma we show relations between the counters of slots of each type.

LEMMA 2.3. *Following observations hold*

1. $t = IS + IC + CS + CC + E + R$,
2. *IS is stochastically dominated by $Bin(t, 1/a^2)$,*
3. *IC is stochastically dominated by $Bin(t, 1/a)$,*
4. $CS \le \frac{IC+E}{a}$,
5. $CC \le IS \cdot a + u_0 \cdot a$.

PROOF. The first point is implied by the fact that each slot before the leader is chosen falls into exactly one category. Points 2 and 3 follow from Lemma 2.2.

To prove 4 observe that a slot s is a correcting silence if the initial estimate u at the beginning of s satisfies $u \ge u_0 + \log_2\sqrt{a} + 1$ and the result of the slot is Null. Recall that a slot is an irregular collision if $u \ge u_0 + \log_2\sqrt{a}$ and the transmissions result in Collision. In a correcting silence the estimate is decreased from u to $u-1$. Consider steps before step s in which the estimate was increased from $u-1$ to u. Since $u - 1 \ge u_0 + \log_2\sqrt{a}$ then each such a slot was either an irregular collision or was a slot jammed by the adversary. Since each Collision causes the estimate to increase by $1/a$ we can exclusively assign exactly a irregular collision or slots with adversarial jamming to each correcting silence.

Similarly we can justify point 5. The total number of correcting collisions cannot be greater then a times number of steps caused by irregular silence plus slots necessary to reach $u_0 - 2a$ from 0 (note that the initial value of u is 0). Clearly each irregular silence can cause decrementing u by at most 1. □

At the beginning of each regular slot, the estimate u is close to u_0 thus each such a slot yields a significant probability of Single.

LEMMA 2.4. *In each regular slot, the probability of* Single *is at least $C = \frac{\ln a}{a^2}$.*

PROOF. By the definition in each regular slot the variable u satisfies the following double inequality

$$u_0 - \log_2(2\ln a) \le u \le u_0 + \log_2(\sqrt{a}).$$

Thus the probability of transmitting of each station is $\frac{2\ln(a)}{n} \ge p \ge \frac{1}{n\sqrt{a}}$. Let us recall that the probability of Single in a slot in which all n stations transmit independently with probability p is $f(p) = n \cdot p(1-p)^{n-1}$. Since f has a single local maximum for $p \in [0,1]$ then the minimal value of $f(p)$ for $p \in \left[\frac{2\ln(a)}{n}, \frac{1}{n\sqrt{a}}\right]$ is $\min\{f(\frac{2\ln(a)}{n}), f(\frac{1}{n\sqrt{a}})\}$.

By Lemma 2.1 (p.3) we have that $f(\frac{2\ln(a)}{n}) > \frac{2\ln a}{a^2}$ and from (p.4) we get that $f(\frac{1}{n\sqrt{a}}) > \frac{1}{\sqrt{a}} - \frac{1}{a}$ Let us observe that $\frac{2\ln a}{a^2} < \frac{1}{\sqrt{a}} - \frac{1}{a}$ for all $a \ge 8$. □

To prove the next lemma we need following version of the Chernoff bound.

FACT 1. *Let $X \sim Bin(n,p)$, then for any $0 \le \delta < 3/2$*

$$Pr[X > (\delta+1)np] \le exp\left(-\frac{\delta^2 np}{3}\right).$$

PROOF. This inequality is obtained by substituting $t = \delta np$ to Thm. 2.1, formula 2.5 in [12]. □

In the following lemma we show upper bounds, that hold with high probability, on the numbers of irregular silences and irregular collisions.

LEMMA 2.5. *For any* $\beta \geq 1$

$$\mathbb{P}\left[IS(1+a) > \frac{2t}{a^2}(1+a)\right] \leq \frac{1}{3n^\beta},$$
$$\mathbb{P}\left[\frac{9}{8}IC > \frac{9}{4}\frac{t}{a}\right] \leq \frac{1}{3n^\beta}.$$

for $t > 3a^2 \log(3n^\beta)$.

PROOF. Since IS, IC are bounded by a binomially distributed random variable (Lemma 2.3 points 2 and 3) we can apply a standard Chernoff bound. The proof follows directly from Fact 1 for $\delta = 1$. □

Finally, using all the previous lemmas we can show bound on the runtime of AWDP algorithm.

THEOREM 2.6. *For any* $0 < \varepsilon < 1$ *and any constant* $\beta \geq 1$ *AWDP chooses a leader in strong-CD model with knowledge of* ε*, in time*

$$t = \mathcal{O}\left(\max\left\{T, \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}\right\}\right)$$

with probability at least $1 - 1/n^\beta$ *in the presence of any* $(T, 1-\varepsilon)$*-adversary with known* ε *and unknown* T.

PROOF. From the Lemma 2.4 we have that in each regular slot we have probability of Single at least $C = \frac{2\ln a}{a^2}$ independently of other slots. One can easily prove that it suffice to have at least $\frac{\ln(3n^\beta)}{C}$ regular slots to get at least one Single with probability at least $1 - 1/(3n^\beta)$.

We will demonstrate that the number of regular slots is big with high probability independently on the way the adversary is acting.

From Lemma 2.3 and since $a \geq 8$ we have

$$\begin{aligned} R &= t - IS - IC - CS - CC - E \\ &\geq t - IS - IC - \frac{IC + E}{a} - IS \cdot a - u_0 \cdot a - E \\ &\geq t - IS(1+a) - \frac{9}{8}IC - u_0 \cdot a - (1 + \frac{1}{a})E = \star \end{aligned}$$

Let us assume that $> T$. Since, by the definition of the adversary $E < (1-\varepsilon)t$, then $(1 + \frac{1}{a})E < (1-\varepsilon)(1 + \frac{\varepsilon}{8})t \leq (1 - \frac{7}{8}\varepsilon)t$.

Applying this observation and a bound for u_0 to $\star$ we get

$$R > \frac{7}{8}\varepsilon t - IS(1+a) - \frac{9}{8}IC - a \cdot \log n - 1. \quad (1)$$

From Lemma 2.5 we have that

$$\begin{aligned} R &> \frac{7}{8}\varepsilon t - \frac{2t}{a^2}(1+a) - \frac{9t}{4a} - a \cdot \log n - 1 \\ &= \frac{7}{8}\varepsilon t - \frac{17t}{4a} - \frac{2t}{a^2} - a \cdot \log n - 1 \end{aligned}$$

for $t > 3a^2 \log(3n)$ with probability at least $1 - 2/3n^\beta$. Since $a \geq 8$

$$\begin{aligned} R &> \frac{7}{8}\varepsilon t - \frac{9t}{2a} - a \cdot \log n - 1 \\ &\geq \frac{7}{8}\varepsilon t - \frac{9}{16}\varepsilon t - a \cdot \log n - 1 = \frac{5}{16}\varepsilon t - a \cdot \log n - 1. \end{aligned}$$

Finally, it is enough to take t such that

$$R > \frac{5}{16}\varepsilon t - a \cdot \log n - 1 > \frac{a^2 \ln(3n^\beta)}{2\ln a}$$

or equivalently,

$$t > \frac{16}{5\varepsilon}\left(\frac{a^2 \ln(3n^\beta)}{2\ln a} + a \cdot \log n + 1\right) = \mathcal{O}\left(\frac{\log(n^\beta)}{\varepsilon^3 \log(1/\varepsilon)}\right).$$

to complete AWDP with probability at least $1 - 1/n^\beta$. □

LEMMA 2.7. *Any leader election algorithm working with probability at least* $1 - 1/n$ *in presence of a* $(T, 1-\varepsilon)$*-bounded adversary requires time* $\Omega(\max\{T, 1/\varepsilon \log n\})$*, for any* T, n, ε*, where* $0 < \varepsilon < 1$.

PROOF. To choose a leader considered model, even without an adversary with probability at least $1 - 1/n$, one needs $\Omega(\log n)$ slots [18]. The adversary can simply jam the first $\lfloor (1-\varepsilon)T \rfloor$ slots out of each T consecutive time steps ensuring that in order to have $c \log n$ non-jammed steps, for any constant c, the algorithms needs to work for $\Omega(\max\{T, 1/\varepsilon \log n\})$ slots. □

2.3 Extension for unknown ε

In previous section we presented algorithm AWDP (ε), which is immune against any $(T, 1-\varepsilon)$-bounded adversary, but needs the knowledge of parameter ε. In this section we will propose an algorithm, that estimates ϵ and therefore works without the knowledge of this parameter. In order to perform leader election without knowledge of ε we would like to execute algorithm AWDP ($\hat{\varepsilon}$) multiple times with different values of $\hat{\varepsilon}$, for example $\hat{\varepsilon} = 1/2, 1/4, 1/8, \ldots$. But to do so we need to know the estimate of the runtime of AWDP ($\hat{\varepsilon}$) to stop its execution if $\hat{\varepsilon}$ is not the correct estimate of ε. Thus we are looking for an estimate of value $\max\{\log n, T\}$. We will show that using following function Estimation we can obtain a value that is between $\log\log n - 1$ and $\max\{\log\log n, \log T\} + 1$ w.h.p.

Function 3 Estimation(L)

```
for round ← 1, 2, ... do
    Repeat 2^round times Broadcast(2^round)
    nulls ← number of Nulls in this round
    if nulls ≥ L then
        return round
```

LEMMA 2.8. *For any* $\varepsilon > 0$*, if* $n \geq 115$ *then with probability at least* $1 - 2/n^2$ *function Estimation(2) in the presence of* $(T, 1-\varepsilon)$*-adversary obtains* Single *or returns value* i *satisfying* $\log\log n - 1 \leq i \leq \max\{\log\log n, \log T\} + 1$ *in time* $\mathcal{O}(\max\{\log n, T\})$

PROOF. Assume that Single did not occur in none of the slots of procedure Estimation. Denote by p, the transmission probability in a given round r. Observe that the following facts are true by Lemma 2.1:

$$\mathbb{P}\left[\textsf{Null in } r|p \geq \frac{1}{\sqrt{n}}\right] \leq e^{-\sqrt{n}}, \quad (2)$$

$$\mathbb{P}\left[\textsf{Collision in } r|p \leq \frac{1}{n^2}\right] \leq \frac{1}{n^2}. \quad (3)$$

To show that $\log\log n - 1 \leq i$ observe that in rounds $1, 2, \ldots \lfloor\log\log n\rfloor - 1$ there is a total number of at most $\log n$ time slots. And in each of these time slots, the probability of transmission is at least $2^{-2^{\lfloor\log\log n\rfloor - 1}} \geq \frac{1}{\sqrt{n}}$. Thus by (2) and the Union Bound we obtain that the probability that in any of these slots there was a Null is at most $\log n \cdot e^{-\sqrt{n}}$. Since $n \geq 115$ we have $\log n \cdot e^{-\sqrt{n}} \leq 2/n^2$, thus with probability at least $1 - 2/n^2$ we have $i \geq \log\log n - 1$.

Now we want to show that $i \leq \max\{\lceil\log\log n\rceil + 1, \lceil\log T\rceil + 1\}$. In i-th round there are at least $2^{\lceil\log T\rceil + 1} \geq 2T$ slots. Among T slots at most $(1-\varepsilon)T$ slots can be jammed, thus, since $\varepsilon > 1$, then at least 1 slot is not-jammed. Thus in i-th round we have at least 2 not-jammed slots. By (3), and using the Union Bound we obtain that with probability at least $1 - 2/n^2$ in both of these rounds we obtain Null. Thus with probability at least $1 - 2/n^2$ the returned value is at most $\max\{\lceil\log\log n\rceil + 1, \lceil\log T\rceil + 1\}$. □

Now let us denote by `AWDP` $(\hat{\varepsilon}, t)$ the execution of procedure `AWDP` $(\hat{\varepsilon})$ terminated after exactly t steps. Let c be such a constant that execution `AWDP` $(\hat{\varepsilon}, c\max\{T, \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}\})$ results in a Single with probability at least $1 - 1/n^2$ in the presence of any $(T, 1-\varepsilon)$-bounded adversary, where $\varepsilon/2 \leq \hat{\varepsilon} \leq \varepsilon$. Such a constant exists by Theorem 2.6.

Function 4 `UnknownParameters`

$\varepsilon_i \leftarrow 2^{-i/3}$
$t_0 \leftarrow c \cdot 2^{1+\texttt{Estimation(2)}}$
$t_i \leftarrow \frac{t_0}{\varepsilon_i^3 \log(1/(\varepsilon_i))}$
for $i \leftarrow 1, 2, \ldots$ **do**
  for $j \leftarrow 1, 2, \ldots, i$ **do**
    `AWDP` $\left(\varepsilon_j, t_i \cdot \frac{i}{j}\right)$

Theorem 2.9. *If $n \geq 115$, then algorithm* `UnknownParameters` *elects a leader in strong-CD model in time:*

1. $\mathcal{O}\left(\frac{\log\log(1/\varepsilon)}{\varepsilon^3} \log n\right)$, *if* $T \leq \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$,

2. $\mathcal{O}\left(\max\left\{\log\log\left(\frac{T}{\varepsilon \log n}\right), \log(1/\varepsilon)\log\log(1/\varepsilon)\right\} T\right)$, *if* $T > \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$,

with probability at least $1 - 1/(3n)$ in the presence of any $(T, 1-\varepsilon)$-bounded adversary with unknown ε and T.

Proof. First assume that the value returned by function `Estimation`(2) is between $\log\log n - 1$ and $\max\{\log\log n, \log T\} + 1$. This happens with probability at least $1 - 2/n^2$ by Lemma 2.8. Thus variable t_0 satisfies $4c\max\{\log n, T\} \geq t_0 \geq c\log n$.

Observe that for a fixed i we execute procedures `AWDP` $(\varepsilon_1, t_i \cdot i)$, `AWDP` $(\varepsilon_2, t_i \cdot i/2)$, ..., `AWDP` (ε_i, t_i), where $t_i = 2^i t_0/(i/3 - 1) = 3 \cdot 2^i t_0/i$. Thus the total time of all procedures executed for a fixed i equals

$$\sum_{j=1}^{i} t_i \frac{i}{j} \leq 3 \cdot 2^i c t_0 \cdot \sum_{j=1}^{i} \frac{1}{j} \leq 3 \cdot 2^i (\ln i + 1) t_0.$$

Thus if I is the largest value of variable i in algorithm `UnknownParameters` then the total time of the algorithm is

$$\sum_{i=1}^{I} 3 \cdot 2^i (\ln i + 1) t_0 \leq 3 \cdot 2^{I+1} (\ln I + 1) t_0 \quad (4)$$

We will consider two cases:

First assume that $T \leq \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$. Take $i^* = \lceil 3\log_2(1/\varepsilon)\rceil$ and consider procedure `AWDP` $(\varepsilon_{i^*}, t_{i^*})$. Observe that $\varepsilon/2 \leq \varepsilon_{i^*} \leq \varepsilon$. We have $t_{i^*} = \frac{1}{\varepsilon_{i^*}^3 \log(1/(\varepsilon_{i^*}))} \cdot t_0$ and

$$t_{i^*} \geq \frac{c\log n}{\varepsilon^3 \log(1/\varepsilon)},$$

thus by the choice of constant c, with probability at least $1 - 1/n^2$ procedure `AWDP` $(\varepsilon_{i^*}, t_{i^*})$ results in a Single. The total time in this case is by Equation (4) at most:

$$2^{i^*+1}(\ln i^* + 1)t_0 = \mathcal{O}\left(\frac{\log\log(1/\varepsilon)}{\varepsilon^3} \log n\right).$$

Consider the second case of $T > \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$. Take $i^* = \lceil\log(cT/t_0)\rceil + \lceil\log_2(\log_2(1/\varepsilon))\rceil$ and $j^* = \lceil 3\log_2(1/\varepsilon)\rceil$. Assume that $j^* \leq i^*$. Consider procedure `AWDP` $(\varepsilon_{j^*}, t_{i^*}\frac{i^*}{j^*})$. Observe that, similarly as in the previous case, $\varepsilon/2 \leq \varepsilon_{j^*} \leq \varepsilon$, and

$$t_{i^*}\frac{i^*}{j^*} = \frac{2^{\lceil\log(cT/t_0)\rceil} \cdot 2^{\lceil\log_2(\log_2(1/\varepsilon))\rceil} 3 \cdot t_0}{\lceil 3\log_2(1/\varepsilon)\rceil} \geq cT.$$

Thus procedure `AWDP` $(\varepsilon_{j^*}, t_{i^*}\frac{i^*}{j^*})$ results in Single with probability at least $1 - 1/n^2$. The total time in this case by Equation (4) is at most:

$$2^{i^*+1}(\ln i^* + 1)t_0 = \mathcal{O}\left(\log\log\left(\frac{T}{\varepsilon\log n}\right) T\right).$$

On the other hand if $j^* > i^*$, we consider procedure `AWDP` (j^*, j^*). We have

$$t_{j^*} = \frac{2^{j^*} 3t_0}{j^*} \geq \frac{2^{i^*} 3t_0}{j^*} \geq cT.$$

Since $\varepsilon/2 \leq \varepsilon_{j^*} \leq \varepsilon$, then procedure `AWDP` $(\varepsilon_{j^*}, t_{j^*}\frac{i^*}{j^*})$ results in Single with probability at least $1 - 1/n^2$, and the total time is in this case

$$2^{j^*+1}(\ln j^* + 1)t_0 = \mathcal{O}\left(\frac{\log\log(1/\varepsilon)}{\varepsilon^3} t_0\right) =$$
$$= \mathcal{O}\left(\log(1/\varepsilon)\log\log(1/\varepsilon) T\right),$$

because since $j^* > i^*$, then $1/\varepsilon^3 \geq cT\log(1/\varepsilon)/(4t_0)$.

The probability of failure of procedure `Estimation` is at most $2/n^2$ and the probability of failure of the selected `AWDP` procedure is at most $1/n^2$. By the union bound, algorithm `UnknownParamters` successfully elects a leader in the correct time with probability at least $1 - 3/n^2 \geq 1 - 1/(3n)$. □

3. LEADER ELECTION IN WEAK-CD

In the previous section we presented an algorithm that succeeds with high probably to elect a leader in strong-CD model by obtaining a Single in the channel. It is usually unrealistic in practice to assume that a station can simultaneously transmit and listen thus we would like to propose a solution working in the weak-CD model.

In our solution, each station that transmits assumes that the transmission results in a Collision. The pseudocode of function `Broadcast` in weak-CD is as follows:

Function 5 `Broadcast`(u)

```
transmit with probability 2^{-u}
if transmitted then
    return Collision
else
    return the status of the channel
```

Using such a modified `Broadcast` function we can deploy our algorithms for strong-CD from previous sections in weak-CD and they will give the same result until the first Single. Thus any leader election in strong-CD can be immediately used as a selection resolution algorithm in weak-CD. After the first Single all the stations, except the transmitter, are aware that the leader is elected. But the transmitter itself does not know that it became the leader and the procedure should be finished. Thus we need a procedure to notify the transmitter. With no adversary a simple notification mechanism is sufficient: one can perform the algorithm only in odd time slots and whenever a successful transmission occurs, the stations that heard the transmission, broadcast in the corresponding even time slot. Using this mechanism, the leader can realize that it had become a leader and therefore ensuring the termination of the algorithm. However even a simple adversary can disrupt such algorithm by jamming some even time slot. In this section we will propose a new mechanism that will allow us to notify the leader and terminate the algorithm with only a constant factor overhead in the presence of any $(T, 1-\varepsilon)$-bounded adversary.

Instead of dividing the slots into sets of odd and even we will use a partition into three sets $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3$. The partition is defined as follows:

$$\mathcal{C}_1^i = \{3 \cdot 2^i - 3, 3 \cdot 2^i - 2, \dots, 4 \cdot 2^i - 4\}$$

$$\mathcal{C}_2^i = \{4 \cdot 2^i - 3, 4 \cdot 2^i - 2, \dots, 5 \cdot 2^i - 4\}$$

$$\mathcal{C}_3^i = \{5 \cdot 2^i - 3, 5 \cdot 2^i - 2, \dots, 6 \cdot 2^i - 4\}$$

$$\mathcal{C}_1 = \bigcup_{i=1}^{\infty} \mathcal{C}_1^i,\ \mathcal{C}_2 = \bigcup_{i=1}^{\infty} \mathcal{C}_2^i,\ \mathcal{C}_3 = \bigcup_{i=1}^{\infty} \mathcal{C}_3^i$$

Sets $\mathcal{C}_j^i$ will be called intervals because they are composed of consecutive time steps. Each interval $\mathcal{C}_j^i$, for $j \in \{1, 2, 3\}$ has size 2^i, thus for $i \geq \log_2 T$, the adversary cannot jam the entire interval.

Take any algorithm $\mathcal{A}$ which obtains the first Single in the channel in time $t(n)$ with probability at least $1 - 1/(3n)$ in the presence of a $(T, 1-\varepsilon)$-bounded adversary. Algorithm `Notification` will transform any such algorithm $\mathcal{A}$ into a leader election algorithm working in time $O(t(n))$ with probability at least $1 - 1/n$ immune against the same adversary.

In algorithm `Notification` we will execute algorithm $\mathcal{A}$ twice, once in slots from $\mathcal{C}_1$ and once in slots from $\mathcal{C}_2$. We use a specific way of executing the algorithm in these sets. Assume that we want to execute $\mathcal{A}$ in $\mathcal{C}_1$. Then we will perform 2^i first steps of the algorithm in interval $\mathcal{C}_1^i$, then all stations will revert all variables associated with algorithm $\mathcal{A}$ to their initial state and will perform 2^{i+1} first steps in interval $\mathcal{C}_1^{i+1}$. For randomized algorithms $\mathcal{A}$ we perform new random choices after the restart. Thus the first 2^i steps of $\mathcal{A}$ in $\mathcal{C}_1^{i+1}$ might produce different result than 2^i steps in $\mathcal{C}_1^i$.

Such execution can last for example until Single in $\mathcal{C}_1$ (however we need to remember that such Single will not be heard by the transmitter) or until Single in $\mathcal{C}_2$.

Function 6 `Notification`

```
leader ← undefined
perform algorithm A in C_1 until a Single in C_1 or C_2
if status(C_1) = Single then
    leader ← false
    stop the algorithm in C_1
    perform algorithm A in C_2 until a Single in C_2 or C_3
if status(C_2) = Single then
    if leader = false then
        transmit in each step in C_1 until a Single in C_3
        return
    else if leader = undefined then
        leader ← true
        transmit in each step in C_3 until a Null in C_1
        return
```

Lemma 3.1. *If $n \geq 3$ and algorithm $\mathcal{A}$ obtains the first* Single *in the channel in weak-CD in time $t(n)$ with probability at least $1 - 1/(3n)$ and $t(n)$ is non-decreasing, then*

1. *algorithm `Notification` elects a leader in weak-CD in time $\mathcal{O}(t(n))$ with probability at least $1 - 1/n$,*

2. *if $\mathcal{A}$ is immune against any $(T, 1-\epsilon)$-bounded adversary for any T and any ϵ then `Notification` is also immune against any $(T, 1-\epsilon)$-bounded adversary.*

Proof. Take $i = \lceil \log_2 t(n) \rceil$ and observe that, based on the definition of `Notification` set $\mathcal{C}_1^i$ contains $2^i \geq t(n)$ consecutive time slots. Thus algorithm $\mathcal{A}$ succeeds in obtaining the first single in $\mathcal{C}_1^i$ with probability at least $1 - 1/(3n)$.

Assume that it happens and the Single is obtained in a time slot from $\mathcal{C}_1^i$. Then all stations, except the transmitter, will stop performing the algorithm in $\mathcal{C}_1$ and start a new execution of $\mathcal{A}$ in $\mathcal{C}_2$. Only the successfully transmitter from $\mathcal{C}_1$ will continue performing algorithm $\mathcal{A}$ in $\mathcal{C}_1$. Since $t(n)$ is non-decreasing then $t(n-1) \leq t(n)$ and with probability at least $1 - 1/(3n-3)$, the second Single is obtained in a time slot from $\mathcal{C}_2^i \subset \mathcal{C}_2$. Now, the transmitter from $\mathcal{C}_1$ will notice this Single, set its leader value to *true* and will start transmitting in $\mathcal{C}_3$. Now observe that the adversary cannot jam all slots from $\mathcal{C}_3^i$, or otherwise algorithm $\mathcal{A}$ with time complexity $t(n)$ could not exist. Thus conditioned that Single was obtained in $\mathcal{C}_1^i$ and in $\mathcal{C}_2^i$, we will also obtain Single in $\mathcal{C}_3^i$.

Let us denote by l, the station that transmitted in $\mathcal{C}_1^i$, by s the station that transmitted in $\mathcal{C}_2^i$ and by R the rest of the stations. After the first Single, all stations from $R \cup \{s\}$ set their values of variable *leader* to false. The only station that does not hear the first Single is l. Notice that l does not participate in algorithm $\mathcal{A}$ in $\mathcal{C}_2^i$ and thus will hear the second Single. Thus l will be the only station with variable *leader* set to *true*. After the successful transmission in $\mathcal{C}_3^i$, all stations from $R \cup \{s\}$ will terminate the algorithm. Clearly in $\mathcal{C}_1^{i+1}$ there will be at least one Null since the adversary cannot jam 2^{i+1} consecutive slots. After the first Null in $\mathcal{C}_1^{i+1}$, station l will also terminate. Thus algorithm `Notification` is a correct LeaderElection algorithm and it works in time at most $8t(n)$ with probability at least $(1 - 1/(3n)) \cdot (1 - 1/(3n - 3)) \geq 1 - 1/n$ in the presence of a $(1 - \epsilon, T)$-bounded adversary.

Observe that if $\mathcal{A}$ executed in $\mathcal{C}_1$ or $\mathcal{C}_2$ will take longer than $t(n)$ to obtain the Single, the algorithm `Notification` will still obtain the correct result and terminate. □

By applying the procedure `Notification` to `AWDP` we obtain an algorithms working in weak-CD with the knowledge of ε. Lemma 3.1 immediately implies bound on the time complexity of such an algorithm.

THEOREM 3.2. *There exists an algorithm electing a leader in a wireless network of n stations in weak-CD model in time $O(\max\{T, \log(1/\varepsilon)/\varepsilon^3\})$ with probability at least $1 - 1/n$ in in the presence of any $(T, 1 - \varepsilon)$-bounded adversary for any known ε, unknown T and unknown $n \geq 3$.*

Finally we apply `Notification` to `UnknownParameters` to obtain an algorithm in weak-CD without the knowledge of any global parameter. By Lemma 3.1 we immediately obtain the following theorem.

THEOREM 3.3. *There exists an algorithm electing a leader in a wireless network of n stations in weak-CD in time*

1. $\mathcal{O}\left(\frac{\log\log(1/\varepsilon)}{\varepsilon^3}\log n\right)$, *if* $T \leq \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$,
2. $\mathcal{O}\left(\max\left\{\log\log\left(\frac{T}{\varepsilon \log n}\right), \log(1/\varepsilon)\log\log(1/\varepsilon)\right\} T\right)$, *if* $T > \frac{\log n}{\varepsilon^3 \log(1/\varepsilon)}$,

with probability at least $1-1/n$ in the presence of any $(T, 1-\varepsilon)$-bounded adversary for any unknown T, ε and unknown $n \geq 115$.

Observe that for unknown constant ε and $T = \omega(\log n)$ our algorithm has complexity $O(T \log\log(T))$.

4. CONCLUSIONS

In this paper we presented a fast leader election protocol robust against a very intensive jamming caused by an adaptive adversary. The presented protocol offers an optimal (with respect to size of the network n) execution time for a practical combination of parameters. Moreover it is also close to optimal with respect to parameter T. However the problem of obtaining an algorithm that for unknown constant ε and very large $T \gg \log n$ is working in time $O(T)$ is left open.

The most important is, however, that our protocol does not need any knowledge about global parameters of the model - including T, ε that describe the adversary. That is - our protocol has a purely **local** character. We believe that some of the presented procedures can be also used as building blocks in constructions of other protocols including size approximation, k-selection or fair use of the wireless channel.

Despite a well developed body of literature many vital questions are left unanswered. In particular it is not clear what countermeasures against a jammer can be constructed for the communication model without collision detection.

5. ACKNOWLEDGMENTS

This paper is supported by Polish National Science Center - decision number 2013/09/B/ST6/02258.

6. REFERENCES

[1] G. Antonoiu and P. K. Srimani. A self-stabilizing leader election algorithm for tree graphs. *J. Parallel Distrib. Comput.*, 34(2):227–232, 1996.

[2] B. Awerbuch. Optimal distributed algorithms for minimum weight spanning tree, counting, leader election and related problems (detailed summary). In A. V. Aho, editor, *STOC*, pages 230–240. ACM, 1987.

[3] B. Awerbuch, A. W. Richa, C. Scheideler, S. Schmid, and J. Zhang. Principles of robust medium access and an application to leader election. *ACM Transactions on Algorithms*, 10(4):24, 2014.

[4] E. Bayraktaroglu, C. King, X. Liu, G. Noubir, R. Rajaraman, and B. Thapa. On the performance of IEEE 802.11 under jamming. In *INFOCOM*, pages 1265–1273, 2008.

[5] T. X. Brown, J. E. James, and A. Sethi. Jamming and sensing of encrypted wireless ad hoc networks. In S. Palazzo, M. Conti, and R. Sivakumar, editors, *MobiHoc*, pages 120–130. ACM, 2006.

[6] S. Cai, T. Izumi, and K. Wada. Space complexity of self-stabilizing leader election in passively-mobile anonymous agents. In *SIROCCO*, volume 5869 of *LNCS*, pages 113–125. Springer, 2009.

[7] J. T. Chiang and Y. Hu. Cross-layer jamming detection and mitigation in wireless broadcast networks. In *MOBICOM*, pages 346–349, 2007.

[8] S. Dolev, S. Gilbert, R. Guerraoui, D. Kowalski, C. Newport, F. Kuhn, and N. Lynch. Reliable distributed computing on unreliable radio channels. *Proc. 2009 MobiHoc S3 Workshop*, 2009.

[9] M. Farach-Colton, R. J. Fernandes, and M. A. Mosteiro. Bootstrapping a hop-optimal network in the weak sensor model. In *ESA*, pages 827–838, 2005.

[10] Z. Golebiewski, M. Klonowski, M. Koza, and M. Kutylowski. Towards fair leader election in wireless networks. In *ADHOC-NOW*, pages 166–179, 2009.

[11] J. Hromkovic, R. Klasing, A. Pelc, P. Ruzicka, and W. Unger. *Dissemination of Information in Communication Networks - Broadcasting, Gossiping, Leader Election, and Fault-Tolerance.* Texts in Theoretical Computer Science. An EATCS Series. Springer, 2005.

[12] S. Janson, T. Luczak, and A. Rucinski. *Random Graphs.* Addison-Wesley, 2000.

[13] M. Kardas, M. Klonowski, and D. Pajak. Energy-efficient leader election protocols for single-hop radio networks. In *ICPP*, pages 399–408, 2013.

[14] M. Klonowski, M. Koza, and M. Kutylowski. Repelling sybil-type attacks in wireless ad hoc systems. In *ACISP*, pages 391–402, 2010.

[15] E. Korach, S. Kutten, and S. Moran. A modular technique for the design of efficient distributed leader finding algorithms. *ACM Trans. Program. Lang. Syst.*, 12(1):84–101, 1990.

[16] M. Li, I. Koutsopoulos, and R. Poovendran. Optimal jamming attacks and network defense policies in wireless sensor networks. In *INFOCOM*, pages 1307–1315, 2007.

[17] X. Liu, G. Noubir, R. Sundaram, and S. Tan. SPREAD: foiling smart jammers using multi-layer agility. In *INFOCOM*, pages 2536–2540, 2007.

[18] K. Nakano and S. Olariu. A randomized leader election protocol for ad-hoc networks. In *SIROCCO*, pages 253–267, 2000.

[19] K. Nakano and S. Olariu. Randomized leader election protocols in radio networks with no collision detection. In *ISAAC*, pages 362–373, 2000.

[20] K. Nakano and S. Olariu. A survey on leader election protocols for radio networks. In *ISPAN*, page 71, 2002.

[21] K. Nakano and S. Olariu. Uniform leader election protocols for radio networks. *IEEE Trans. Parallel Distrib. Syst.*, 13(5):516–526, 2002.

[22] V. Navda, A. Bohra, S. Ganguly, and D. Rubenstein. Using channel hopping to increase 802.11 resilience to jamming attacks. In *INFOCOM*, pages 2526–2530, 2007.

[23] C. C. Newport. Radio network lower bounds made easy. In *DISC*, pages 258–272, 2014.

[24] A. W. Richa, C. Scheideler, S. Schmid, and J. Zhang. Competitive and fair medium access despite reactive jamming. In *ICDCS*, pages 507–516, 2011.

[25] D. E. Willard. Log-logarithmic selection resolution protocols in a multiple access channel. *SIAM J. Comput.*, 15(2):468–477, 1986.

[26] J. Zhang. Robust and efficient medium access despite jamming. *PhD. Thesis*, 2012.