Analysis of max-consensus algorithms in wireless channels

Franck Iutzeler, Philippe Ciblat, and Jérémie Jakubowicz

Abstract

In this paper, we address the problem of estimating the maximal value over a sensor network using wireless links between them. We introduce two heuristic algorithms and analyze their theoretical performance. More precisely, i) we prove that their convergence time is finite almost surely, ii) we derive an upper-bound on their mean convergence time, and iii) we exhibit a bound on their convergence time dispersion.

I. INTRODUCTION

Wireless Sensor Networks are systems composed of scattered agents with limited power and computational abilities. These agents may acquire some data and communicate through a wireless link to some other agents. Their goal is to auto-organize in order to distributively compute a function of the collected data [1]. For instance, if temperature sensors are deployed in a hostile environment (e.g. mountains) and one wants to know the average temperature in the region by looking at any sensor, a simple idea would be to make the sensors randomly wake up and average their value with another sensor so that they all share the average value of the initial measurements of the network at the end. Namely, the sensors want to make *consensus* over the average value of the initial measurements. This problem was extensively studied in the past few years [2], [3], [4]. However, the average is not always the most useful value to share. Indeed, in some applications, the maximal value may be of greater interest.

For example, if a sensor network has to transmit information periodically (e.g. the average temperature of the region in the previous scheme) through a costly link, it would be of interest to distributively elect the sensor which has the most power resource to operate that communication. To do so, one has to

The authors are with Institut Télécom/Télécom ParisTech ; CNRS LTCI. This work has been partially accepted for publication to Asilomar Conference in November 2011. This work was partially granted by the French Defence Agency (DGA).

estimate the maximal amount of energy left in the sensor battery (along with their ID) in a distributed fashion using the wireless links between some of them. Another useful application is the distributed access control: let us consider that some nodes want to send information to a common access point. In that case, an access control algorithm (close to the CSMA/CD) may be the following one: the agents that want to send information i) draw a number in a common window (e.g. $\{1, .., 32\}$) and then ii) make consensus over the maximal value distributively (and the ID of the associated agent). The sensor with the maximal value then effectively sends its packet to the access point. If the communication does not succeed (typically if two agents drew the same number and hence tried to send their packet at the same time), the procedure is repeated with a window of twice the size until success, mimicking the well-known exponential backoff algorithm. This protocol, close to the CSMA/CD scheme, needs the distributed estimation of a maximum value in a wireless channel context.

A simple way to estimate the maximum value would be to mimic the algorithm introduced for the averaging ([3]). So, the agents wake up randomly and exchange their value with another reachable sensor randomly chosen; then both keep the maximum between their former and received values. Nevertheless, as the communications between the sensors are wireless, it seems more natural for an awaking sensor to broadcast its value and then the sensors which have received the information to update their value accordingly. In this work, we will analyze algorithms i) based on pairwise communications and ii) based on broadcast communications. Notice that an averaging algorithm based on broadcast communications has been proposed in [5] but does not perform well due to the non-conservation of the initial sum. This issue is not raised for estimating the maximum value since the maximum value is preserved.

Distributively estimating the maximum value over a network through wireless communications is thus a relevant problem we propose to address hereafter. We prove the convergence of both above-mentioned algorithms and analyze their convergence speed.

This paper is organized as follows: models and algorithms as well as link with related works are reported in Section II. In Section III, we derive our mathematical results. In Section IV, numerical illustrations are given. Concluding remarks are drawn in Section V.

II. Models and Algorithms

A. Assumptions on the wireless network

Consider a network of N sensors modeled as an undirected graph $G = (V, E)$ where V is the set of agents – the vertices of the graph – and E is the set of links between agents – the edges of the graph. We assume that each link is error-free. To indicate that a couple of agents (v, w) are *neighbors* (we also use the term *adjacent*), we use the notation $v \sim w$. For any set S, we denote its cardinality by $|S|$. Obviously, we have $|V| = N$. The set of neighbors of the agent v is denoted $\mathcal{N}_v$. Each agent v has an initial scalar measure $x_0(v)$. The set of all initial measures is thus stacked in a unique vector $\mathbf{x}_0 \in \mathbb{R}^N$. The network is supposed invariant over time and *connected*.

The network is assumed *asynchronous*, meaning that no common clock is available for the agents. Instead, each agent has its own clock and can initiate a communication with its neighborhood at clock ticks. Assuming communication time is small compared to the time between clock ticks, it makes sense (as usually done for other consensus-like algorithms [3], [5]) to consider the absence of collisions between communicating nodes. We also assume that the agent clocks are modeled by independent Poisson processes with intensity λ_v for agent v. It is then equivalent to have a global clock according to a Poisson process with intensity $\lambda = \sum_v \lambda_v$, and that each clock tick is then attributed to a given agent. Then the probability for an agent to wake up is equal to $p_v = \lambda_v/\lambda$. We will assume, for the sake of simplicity that all intensities λ_v are the same, hence $\lambda_v/\lambda = 1/N$. We denote by $x_n(v)$ the value at agent v after n global clock ticks, while $\mathbf{x}_n$ denotes the vector of all values after n global clock ticks.

The goal for the network is to estimate the value $M(\mathbf{x}_0) \triangleq \max_{v\in V} x_0(v)$, in a *distributed* manner, that is, only using communications between adjacent nodes.

B. Algorithms

We propose two algorithms for achieving the task of estimating $M(\mathbf{x}_0)$. Both algorithms are inspired by those already developed to obtain the average-consensus.

The first algorithm is based on the exchange between the current values of two adjacent nodes chosen randomly in the following way.

RANDOM-PAIRWISE-MAX:

1) After the n-th clock tick, a node v_n wakes up.
2) v_n chooses a neighbor w_n uniformly in $\mathcal{N}_{v_n}$.
3) $x_n(v_n) = x_n(w_n) = \max\left(x_{n-1}(v_n), x_{n-1}(w_n)\right)$, and $x_n(v) = x_{n-1}(v)$ otherwise.

This algorithm is suitable for wired networks whereas it is clearly not optimal for wireless networks. Indeed, it does not rely on the broadcasting abilities of the wireless channel. In wireless channel, all the neighbors receive the current value of v_n. Therefore, we propose a second algorithm taking benefit of the broadcast nature of the wireless channel.

RANDOM-BROADCAST-MAX:

1) After the n-th clock tick, a node v_n wakes up.

2) v_n broadcasts its current value to all these neighbors.
3) $x_n(w) = \max\left(x_{n-1}(w), x_{n-1}(v_n)\right)$ for $w \in \mathcal{N}_{v_n}$, and $x_n(w) = x_{n-1}(w)$ otherwise.

Such an algorithm has been already proposed for calculating the average (in that case, the max operator has to be replaced with the average one). Unfortunately, in the context of sum-consensus, such an algorithm does not keep the sum constant along the time which prevents it to converge to the true value. As for the max-consensus, such an algorithm keeps the maximum value and so does not give rise to an undesirable behavior. Therefore the RANDOM-BROADCAST-MAX will be our flagship algorithm.

C. Link with existing works

To our best knowledge, in the framework of distributed computation, only [6] has focused on the max computation. Actually, [6] has developed a general framework to distributively compute a family of functions (including the maximum value) of the nodes measurements. Compared to our set-up, this work has been done under continuous-time and synchronous clocks assumptions. It can nevertheless be adapted to our context (discrete-time and asynchronous clocks), but it will be less powerful since each node goes to the maximum value in an incremental way even if one of its neighbor has the maximum value. To be more precise, let us focus on the following toy example based on a very simple graph with two nodes. The proposed RANDOM PAIRWISE-MAX reaches consensus in one single step, while the algorithm in [6] needs an infinite number of steps because of its incremental nature. Here, [6] pays a price for its generality. Therefore, our proposed algorithms are much more suitable for the max computation.

Even if our work has some connections with the so-called *rumor spreading* issue, it also has some important differences listed below. Actually a rumor spreading or max-consensus algorithm can be distinguished from another one according to three main characteristics: i) who speaks?, ii) with whom and to do what?, and iii) when? In our framework,

i) each node may speak (even if it is unaware of the maximum value). As a consequence, the time spent by each node communication has to be taken into account.
ii) each node speaks with all its neighbors for RANDOM-BROADCAST-MAX, and these neighbors update their value if necessary. As for RANDOM-PAIRWISE-MAX, each node speaks with one of its neighbor randomly chosen and they jointly update their value accordingly.
iii) at each clock tick, only one node randomly chosen speaks. As a consequence, the communication system is collision-free.

Only few papers ([7], [8], [9], [10], [11]) have taken into account the broadcasting nature of the medium in the rumor spreading problematic. But in all these papers, the communications are synchronous and so the main issue deals with the collision between the transmissions. Moreover, only the informed nodes wake up. As a consequence, their set-up is different from ours, and their results do not hold in our context. In [12], the broadcasting nature of the channel is also considered in the so-called FLOOD-MAX algorithm. But the context is much simpler than ours since all the nodes wake simultaneously and the communication is collision-free. Result obtained for this algorithm is clearly unsuitable for our analysis.

All other papers dealing with rumor spreading ([13], [14], [15], [16], [17], [18], [19]) focused on pairwise communication and so does not take benefit of the broadcasting nature of the channel. Consequently, their works and results can not be applied for the RANDOM-BROADCAST-MAX. In contrast, the proposed RANDOM-PAIRWISE-MAX is closely related to them. Actually, in most of these papers, only the informed nodes wake up and propagate its information to a randomly chosen neighbor which differs significantly from our algorithm. However one algorithm, the so-called PUSH-PULL is more closely related to our algorithm. Indeed, at each clock tick, every informed node propagates its information to one of its neighbor randomly chosen (*push* step) whereas every uninformed node asks one of its neighbor for the information (*pull* step) [20], [21]. The update step is then clearly equivalent to those of RANDOM-PAIRWISE-MAX. Nevertheless one fundamental difference exists and prevents us to re-use results on the PUSH-PULL. Indeed, each node is active at each clock tick in the PUSH-PULL set-up whereas, in our set-up, one node randomly chosen is active per clock tick. Consequently one node is active every N clock ticks *in average*. This implies to use few different tools for analyzing the convergence.

Our problem and the proposed algorithms are thus novel, then deserve our theoretical convergence analysis hereafter given.

III. PERFORMANCE ANALYSIS

We define the convergence time τ as the first time when all the nodes share the same value, *i.e.*,

$$\tau \triangleq \inf\{n \in \mathbb{N} : \forall v \in V,\ x_n(v) = M(\mathbf{x}_0)\}. \tag{1}$$

Given an undirected graph $G = (V, E)$ with N nodes, one can define its $N \times N$ adjacency matrix A_G with entries: $a_G(v, w) = 1$ if $v \sim w$ and 0 otherwise. It is a symmetric matrix. We also introduce the $N \times N$ diagonal matrix D_G where the i-th diagonal entry is the degree of the node v_i, *i.e.*, $|\mathcal{N}_{v_i}|$. We denote $d_{\max}$ the maximum degree. The symmetric matrix $L_G = D_G - A_G$ is called the *Laplacian* of the graph G. Its eigenvalues are non-negative and its kernel has dimension 1 whenever the graph is connected. We denote by $\lambda_1, \ldots, \lambda_N$ the eigenvalues of L_G sorted by increasing order. The *diameter* of graph G is given by $\Delta_G = \max\{\ell(v, w) : (v, w) \in V^2\}$ where $\ell(v, w) = \inf\{m \in \mathbb{N} : [A_G]^m(v, w) > 0\}$ corresponds the minimum number of edges needed to connect v to w.

A. Random-Broadcast-Max

Theorem 1 asserts all the sensors will share the maximum value after a finite number of clock ticks.

Theorem 1. *For* RANDOM-BROADCAST-MAX, *we have* $\tau < \infty$ *with probability* 1.

The proof is reported in Appendix A. The previous result is not surprising at all, and we would like now to have more information about the behavior of τ and, especially, about its mathematical expectation $\mathbb{E}[\tau]$ versus some characteristics of the operating graph.

Theorem 2. *For* RANDOM-BROADCAST-MAX, *one has*

$$\mathbb{E}[\tau] \leq \beta, \quad \textit{where } \beta = N\Delta_G + N(\Delta_G - 1)\log\left(\frac{N-2}{\Delta_G - 1}\right).$$

The proof is reported in Appendix B. Note that the upper bound of Theorem 2 is reached when G is the complete graph since the time needed for propagating the *max* is the time needed for the *max* node to wake up and communicate its value to all other nodes using only one broadcast communication, hence N in expectation. Moreover, for the ring graph, we can prove that $\mathbb{E}[\tau] = (N^2 - N)/2$ while the bound is equal to $N^2(1 + \log(2))/2$. By neglecting the term proportional to N, we observe that the mean and its bound are both scaled in N^2.

Let us consider the previous works on max propagation by using the broadcasting nature of the medium ([7], [8], [9], [10], [11]). Even if the framework is strongly different (see Section II-C), it is of interest to compare the performance bounds. When all the informed nodes wake up simultaneously and thus collide to each other, the best convergence time behaves like $\Delta_G \log(N/\Delta_G)$ [9]. Surprisingly, this is almost the same shape as ours up to a factor N.

Having an upper-bound on the expected convergence time is very useful, but does not provide information about the outliers, *i.e.*, the event for which the convergence time is extremely long. Therefore, in Theorem 3, we provide concentration-like result.

Theorem 3. *For* RANDOM-BROADCAST-MAX, *with probability* $1 - \varepsilon$,

$$\tau \leq \beta + N\Delta_G\left(\log\left(\frac{\Delta_G}{\varepsilon}\right) - 1\right)$$

The proof is reported in Appendix C. Let us focus on the ”toy” example considering the complete graph. The extra time cost is equal to $N\log(1/\varepsilon)$, *i.e.*, $N\log N$ if $\varepsilon = 1/N$. Surprisingly, [15] obtained similar results although both frameworks are strongly different.

B. Random-Pairwise-Max

A similar work can be done for the RANDOM-PAIRWISE-MAX. Actually the convergence can be proven by following the same approach as those given in Appendix A. In contrast, the proofs about mean convergence time and concentration rely on quite different tools and thus are introduced hereafter.

Theorem 4. *For* RANDOM-PAIRWISE-MAX, *one has*

$$\mathbb{E}[\tau] \leq \alpha, \quad \textit{where } \alpha = N d_{\max} . \frac{h_{N-1}}{\lambda_2},$$

with the n-th harmonic number $h_n = \sum_{k=1}^{n} 1/k$.

The proof is reported in Appendix D. In order to illustrate the upper-bound given in Theorem 4, let us focus on the case where G is a complete graph. For such a graph, $d_{\max}/\lambda_2$ is of order $\mathcal{O}(1)$, hence our bound is of order $\mathcal{O}(N \log N)$. In the standard rumor spreading context, the bound is of order $\mathcal{O}(\log N)$ [22]. Once again, we pay an extra factor of order N for not knowing which nodes are informed or not.

Theorem 5. *For* RANDOM-PAIRWISE-MAX, *with probability* $1-\varepsilon$,

$$\tau \leq \alpha \left(1 + \log\left(\frac{N}{\varepsilon}\right) . \left(1 + \sqrt{1 + \frac{1}{\log\left(\frac{N}{\varepsilon}\right)}} \right) \right).$$

The proof is reported in Appendix E. Note that for small ε, the RHS of Theorem 5 can be replaced with $\overline{T}_{\mathrm{RPM}}(1+2\log(N/\varepsilon))$. By taking $\varepsilon = 1/N$ (which is usual in the literature), we obtain $\overline{T}_{\mathrm{RPM}}(1+4\log(N))$. In [23], it is proven that τ for the PUSH-PULL is $\mathcal{O}(\alpha_G^{-1}\log^2(N)\sqrt{\log(N)})$ with probability $(1-1/N)$ where α_G is the vertex expansion. In Theorems 4 and 5, $\overline{T}_{\mathrm{RPM}}$ can be actually replaced with $\overline{T}'_{\mathrm{RPM}} = N d_{\max} h_{N-1} \alpha_G^{-1}/2$ by applying the definition of α_G in Eq. (3). Therefore, τ for the RANDOM-PAIRWISE-MAX is $\mathcal{O}(N d_{\max}\alpha_G^{-1}\log^2(N))$ with probability $(1-1/N)$. Up to the factor N (essentially due to our communication protocol), the trends offer strong similarities.

IV. NUMERICAL ILLUSTRATIONS

The proposed upper-bound for the expected convergence time and the convergence time dispersion have been evaluated on Random Geometric Graphs (RGG) which are well suited for modelling Wireless Sensor Networks. They consist in uniformly choosing N points (representing the nodes/sensors) in the unit square and then drawing an edge between two sensors closer than a pre-defined radius r. By choosing $r = \sqrt{8\log(N)/N}$, connectedness is ensured with high probability [24], [4]

In Figure 1, we plot the (empirical) mean number of communications for reaching convergence and the associated upper-bounds (given by Theorems 2 and 4) for each proposed algorithm versus the number of sensors N. We observed that the RANDOM-BROADCAST-MAX outperforms the RANDOM-PAIRWISE-MAX. When the network size increases, the upper-bounds become quite pessimistic due to the various used simplifications (in the case of RANDOM-BROADCAST-MAX, we rely on the spanning tree instead of the whole graph and we broadcast the information layer per layer; in the case of RANDOM-PAIRWISE-MAX, we use Cheeger's inequality and the approximation $1/d_{\max}$). .

As the RANDOM-BROADCAST-MAX is much more interesting in terms of performance, we hereafter only focus on it. In Figure 2, we plot the histograms of the convergence time as well as the upper-bounds for the convergence with probability $1-1/N$ (given in Theorem 3) when $N=40$.

V. CONCLUSION

We have proposed two algorithms for estimating the maximum value in wireless sensor networks. The convergence times of these algorithms have been analyzed in depth.

APPENDIX A
PROOF OF THEOREM 1

Let us denote by K_n the set of nodes sharing $M(\mathbf{x}_0)$ at time n, *i.e.*, we have $K_n = \{v \in V : x_n(v) = M(\mathbf{x}_0)\}$. If the algorithm does not have converged at time n, it still exists at least one node $v_n^{(0)}$ in K_n such that one $w_n^{(0)} \in \mathcal{N}_{v_n^{(0)}}$ is not in K_n, *i.e.*, $\mathcal{N}_{v_n^{(0)}} \cap K_n^c = \emptyset$ where the superscript $(.)^c$ denotes the complementary subset in V. As K_n is a non-decreasing family of non-empty subsets of V, $|K_n|$ is also a non-decreasing integer sequence. At time $n+1$, if $v_n^{(0)}$ wakes up, then the probability of this event is lower-bounded by $1/N$ and thus by $1/Nd_{\max}$. So $|K_{n+1}|$ will be strictly greater than $|K_n|$. As $|K_n|$ is upper-bounded (by N) and is a monotonic sequence, it converges to a certain value c. To have $c < N$, $|K_n|$ has to be constant (but different from N) for any n large enough. At each clock tick, the sequence has a probability less than $(1-1/Nd_{\max})$ to remain constant, and so $(1-1/Nd_{\max})^k$ after k clock ticks. Therefore, $|K_n|$ can not converge to $c < N$ almost surely which concludes the proof.

APPENDIX B
PROOF OF THEOREM 2

We assume for the sake of simplicity that one single node, say $v^{(0)}$, has the maximum at time $n=0$. Let us partition the set V according to nodes' distances from $v^{(0)}$:

$$L_i = \{v \in V : d(v^{(0)}, v) = i\}, \quad k \in \mathbb{N}$$

One has $V = \cup_{i=0}^{\Delta_G} L_i$ and $L_i \cap L_j = \emptyset$ for $i \neq j$. Define the random times: $t_0 = 0$, and $t_i = \inf\{n \geq t_{i-1} : \forall v \in L_i, x_n(v) = M(\mathbf{x}_0)\}$. We denote by $\mathcal{F}_n$ the σ-algebra spanned by the nodes sharing the maximum values at time n. Using the same proof framework as in the standard coupon collector problem (see, e.g. [25]), it is easy to show that $\mathbb{E}[t_{i+1} - t_i | \mathcal{F}_{t_i}] \leq N h_{|L_i|}$. The term $\mathbb{E}[t_{i+1} - t_i | \mathcal{F}_{t_i}]$ corresponds to the duration to fill up completely the layer $(i+1)$ given that the nodes sharing the maximum value at time t_i, *i.e.*, given at least that the layer i was already filled up. Therefore we have

$$\mathbb{E}[\tau] \leq \sum_{i=0}^{\Delta_G - 1} \mathbb{E}[t_{i+1} - t_i | \mathcal{F}_{t_i}] \leq \sum_{i=0}^{\Delta_G - 1} N h_{|L_i|} \ .$$

By using the inequality $h_n \leq \log(n) + 1$ and the fact $|L_0| = 1$, we obtain

$$\mathbb{E}[\tau] \leq N \left(\Delta_G + \sum_{i=1}^{\Delta_G - 1} \log |L_i| \right) \ .$$

Using $\sum_{i=1}^{n-1} \log x_i \leq (n-1) \log(\frac{1}{n-1} \sum_{i=1}^{n-1} x_i)$, with $x_i = |L_i|$ and $n = \Delta_G$ concludes the proof.

Appendix C

Proof of theorem 3

Let $A_i^v(t)$ be the event that the node v (belonging to layer L_i) is not switched on after t iterations. So $\mathbb{P}[A_i^v(t)] = \left(\frac{N-1}{N}\right)^t$. When the event $t_{k+1} - t_k \geq t$ occurs, we know that the event $\cup_{v \in L_{i+1}} A_i^v(t)$ also occurs. Therefore $\mathbb{P}[t_{i+1} - t_i \geq t] \leq \mathbb{P}(\cup_{v \in L_{i+1}} A_i^v(t))$. By using the Union bound and the fact that $0 \leq 1 - y \leq \exp(-y)$ for $y \in [0,1]$, one can prove the probability that after t iterations, some of the nodes of L_i have still not talked is as follows

$$\mathbb{P}[\cup_{v \in L_{i+1}} A_i^v(t)] \leq \sum_{v \in L_i} \exp\left(-\frac{t}{N}\right). \tag{2}$$

For any $\varepsilon > 0$, by choosing $t_\varepsilon = N \log |L_i| + N \log(\Delta_G / \varepsilon)$, we then get

$$\mathbb{P}\left[t_{i+1} - t_i \geq N \log |L_i| + N \log\left(\frac{\Delta_G}{\varepsilon}\right)\right] \leq \frac{\varepsilon}{\Delta_G}.$$

By using once again the Union's bound, we find the final result.

Appendix D

Proof of Theorem 4

The definition of K_n is given at the beginning of Appendix A. In the context of RANDOM-PAIRWISE-MAX, one has $|K_n| \leq |K_{n+1}| \leq |K_n| + 1$. Here, our objective is to exhibit a tight evaluation of the probability that the sequence $|K_n|$ is strictly increasing at time n.

$$\mathbb{P}[|K_{n+1}| = |K_n| + 1 \mid K_n] = \mathbb{P}[v_n \in K_n, w_n \notin K_n | K_n, v_n \sim w_n] = \mathbb{P}[\{v_n, w_n\} \in \partial K_n | K_n, v_n \sim w_n].$$

The selection algorithm of an edge is as follows: choose v_n uniformly over V, then w_n uniformly over $\mathcal{N}_{v_n}$ and independently of the past or vice-versa. Therefore, for any edge e, we have $\mathbb{P}[\{v_n, w_n\} = e] \geq 2/Nd_{\max}$ which implies that

$$\mathbb{P}[\{v_n, w_n\} \in \partial K_n \mid K_n, v_n \sim w_n] \geq 2\frac{|\partial K_n|}{Nd_{\max}}.$$

For any subset S of V, the following inequality, called Cheeger's inequality, holds

$$\frac{|\partial S|}{|S|} \geq \lambda_2.\left(1 - \frac{|S|}{N}\right) \tag{3}$$

where $\partial S \triangleq \big\{\{v, w\} \in E : v \in S, w \notin S\big\}$ is the boundary of S. More details are available in [26].

Using Cheeger's inequality, we obtain

$$\mathbb{P}[\{v_n, w_n\} \in \partial K_n \mid K_n, v_n \sim w_n] \geq \frac{2\lambda_2}{N^2 d_{\max}}(N - |K_n|)|K_n| \tag{4}$$

As in Appendix B, assuming, for the sake of simplicity, that initially one single node has the maximum value, consider the stopping times: $\tau_i = \inf\{n \in \mathbb{N} : |K_n| = i\}$, so that $\tau_1 = 0$ and $\tau = \sum_{i=1}^{N-1}(\tau_{i+1} - \tau_i)$ (if more than one node have the maximum value at time 1, one just has to start at $i > 1$). Let L_n be equal to the random variable $|K_{n+1}| - |K_n|$ given $|K_n|$. L_n is a Bernoulli distribution of parameter p_n. From Eq. (4), we have $p_n \geq (2\lambda_2/N^2 d_{\max}).(N - |K_n|)|K_n|$. As $(\tau_{i+1} - \tau_i)$ is the number of iterations needed to increment $|K_n|$ when $|K_n| = i$, or equivalently, the number of trials on L_n for obtaining the value 1 when $|K_n| = i$, the random variable $(\tau_{i+1} - \tau_i)$ is geometrical distributed with parameter $p_i \geq \pi_i$ with $\pi_i = (2\lambda_2/N^2 d_{\max}).(N - i)i$. As a consequence, $\mathbb{E}[\tau_{i+1} - \tau_i] \leq 1/\pi_i$. We thus have

$$\mathbb{E}[\tau] \leq \frac{N^2 d_{\max}}{2\lambda_2} \sum_{i=1}^{N-1} \frac{1}{(N-i)i} = \frac{N d_{\max}}{\lambda_2} \sum_{i=1}^{N-1} \frac{1}{i} ,$$

which after some simple algebra leads to the result.

Appendix E
Proof of theorem 5

Remind the notations used in Appendix D. The random variable $\tau_{i+1} - \tau_i$ is geometric-distributed with parameter $p_i \geq \pi_i$. As a consequence, $\tau_{i+1} - \tau_i$ is stochastically dominated by a geometric distribution with parameter π_i denoted by Y_i, which means that the cdf of $\tau_{i+1} - \tau_i$ is smaller than the cdf of Y_i at any point [27]. By using Chernoff's bound for geometric random variable, we have, for any $\delta > 0$,

$$\mathbb{P}\left[\tau_{i+1} - \tau_i \geq \frac{1+\delta}{\pi_i}\right] \leq \mathbb{P}\left[Y_i \geq \frac{1+\delta}{\pi_i}\right] \leq \exp\left(-\frac{\delta^2}{2(1+\delta)}\right).$$

Let ε be any positive value. Selecting δ_ε as the smallest positive term such that $\exp\left(-\delta_\varepsilon^2/(2(1+\delta_\varepsilon))\right) \leq \varepsilon/N$ leads to $\delta_\varepsilon = \log(N/\varepsilon)(1 + \sqrt{1 + 1/\log(N/\varepsilon)})$. So, we have $\mathbb{P}[\tau_{i+1} - \tau_i \geq (1+\delta_\varepsilon)/\pi_i] \leq \varepsilon/N$. Then, by using the Union's bound, we have $\mathbb{P}[\tau \geq (1+\delta_\varepsilon)(\sum_i 1/\pi_i)] \leq \varepsilon$ which concludes the proof.

REFERENCES

[1] J.N. Tsitsiklis, *Problems in decentralized decision making and computation*, Ph.D. thesis, M. I. T., Dept. of Electrical Engineering and Computer Science, 1984.

[2] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Analysis and optimization of randomized gossip algorithms," in *Proc. CDC Decision and Control 43rd IEEE Conf*, 2004, vol. 5, pp. 5310–5315.

[3] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.

[4] A. D. G. Dimakis, A. D. Sarwate, and M. J. Wainwright, "Geographic Gossip: Efficient Averaging for Sensor Networks," *IEEE Transactions on Signal Processing*, vol. 56, no. 3, pp. 1205–1216, 2008.

[5] T. C. Aysal, M. E. Yildiz, A. D. Sarwate, and A. Scaglione, "Broadcast Gossip Algorithms for Consensus," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2748–2761, 2009.

[6] J. Cortés, "Distributed algorithms for reaching consensus on general functions," *Automatica*, vol. 44, no. 3, pp. 726–737, 2008.

[7] I. Chlamtac and S. Kutten, "On Broadcasting in Radio Networks–Problem Analysis and Protocol Design," *IEEE Transactions on Communications*, vol. 33, no. 12, pp. 1240–1246, 1985.

[8] O. Bar-Yehuda, R. Goldreich and A. Itai, "On the time-complexity of broadcast in radio networks: an exponential gap between determinism randomization," *Journal of Computer and System Science*, vol. 45, no. 1, pp. 104–126, 1992.

[9] E. Kushilevitz and Y. Mansour, "An $\Omega(D \log(N/D))$ lower bound for broadcast in radio networks," *SIAM Journal on Computing*, vol. 27, no. 3, pp. 702–712, 1998.

[10] L. Chrobak, M. Gasieniec and W. Rytter, "Fast broadcasting and gossiping in radio networks," *Journal of Algorithms*, vol. 43, no. 2, pp. 177–189, 2002.

[11] A. Czumaj and W. Rytter, "Broadcasting algorithms in radio networks with unknown topology," in *Proc. 44th Annual IEEE Symp. Foundations of Computer Science*, 2003, pp. 492–501.

[12] N.A. Lynch, *Distributed Algorithms*, Morgan Kaufmann Series In Data Management Systems, 1996.

[13] P.J. Slater, E.J. Cockayne, and S.T. Hedetniemi, "Information dissemination in trees," *SIAM Journal on Computing*, vol. 10, pp. 692, 1981.

[14] B Pittel, "On spreading a rumor," *SIAM Journal on Applied Mathematics*, pp. 213–223, 1987.

[15] U. Feige, D. Peleg, P. Raghavan, and E. Upfal, "Randomized broadcast in networks," *Random Structures & Algorithms*, vol. 1, no. 4, pp. 447–460, 1990.

[16] W.R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, 1999, pp. 174–185.

[17] R. Karp, C. Schindelhauer, S. Shenker, and B. Vocking, "Randomized rumor spreading," in *FOCS*, 2000, p. 565.

[18] B. Doerr, T. Friedrich, and T. Sauerwald, "Quasirandom rumor spreading: Expanders, push vs. pull, and robustness," *Automata, Languages and Programming*, pp. 366–377, 2009.

[19] N. Fountoulakis, A. Huber, and K. Panagiotou, "Reliable Broadcasting in Random Networks and the Effect of Density," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.

[20] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry, "Epidemic algorithms for replicated database maintenance," in *Proceedings of the sixth annual ACM Symposium on Principles of distributed computing*. ACM, 1987, pp. 1–12.

[21] T. Sauerwald and A. Stauffer, "Rumor spreading and vertex expansion on regular graphs," in *Proc. 22nd ACM-SIAM Symp. on Discrete Algorithms (SODA)*, 2011, pp. 462–475.

[22] A. Frieze and G. Grimmett, "The shortest-path problem for graphs with random arc-lengths," *Discrete Applied Mathematics*, vol. 10, no. 1, pp. 57–77, 1985.

[23] G. Giakkoupis and T. Sauerwald, "Rumor spreading and vertex expansion," in *Proc. 23rd ACM-SIAM Symp. on Discrete Algorithms (SODA)*, 2012, pp. 1623–1641.

[24] M. Penrose, *Random geometric graphs*, Oxford University Press, 2003.

[25] R. Motwani and P. Raghavan, *Randomized algorithms*, Cambridge University Press, 1995.

[26] N. Biggs, *Algebraic graph theory*, vol. 67, Cambridge Univ Pr, 1993.

[27] H. Thorisson, *Coupling, stationarity, and regeneration*, Springer Verlag, 2000.

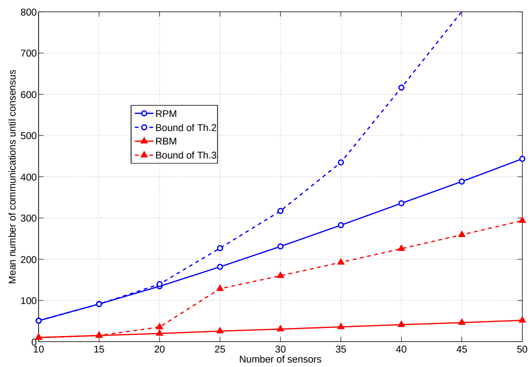

Fig. 1. (Empirical) mean number of communications for reaching convergence and the associated upper-bounds versus N.

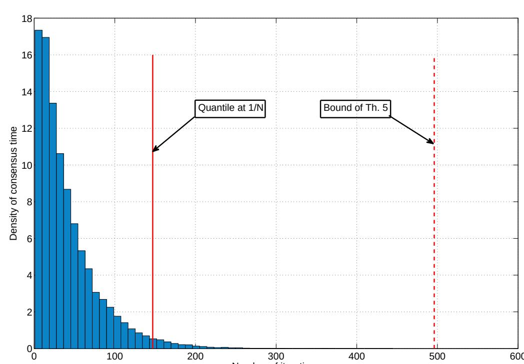

Fig. 2. Histogram of the convergence time and upper-bounds associated with probability $(1 - 1/N)$ for the RANDOM-BROADCAST-MAX when $N = 40$.