Algorithms Seminar 2002–2004,
F. Chyzak (ed.), INRIA, (2005), pp. 97–100.

Available online at the URL
`http://algo.inria.fr/seminars/`.

Quasi-Optimal Leader Election Algorithms in Radio Networks with Log-Logarithmic Awake Time Slots

Jean-François Marckert
Lama, University of Versailles (France)

April 28, 2003

Summary by Marianne Durand

Abstract

This summary presents two leader election protocols for radio networks, in both telephone and walkie-talkie model, where the number of radios is unknown. Those randomized protocols are shown to elect a leader in $O(\log n)$ expected time, and to be energy-efficient due to a small total awake time of the radio stations.

1. Introduction

A radio network consists in n radio stations with a graph of connection. The graph is assumed to be complete, so that all the stations are directly connected. All the stations are identical, and indistinguishable, and it is assumed that they are synchronized, and thus have a common notion of time. All stations can talk and listen. In a first model called *telephone model*, they can do both at the same time, and in a second called *walkie-talkie model* they can listen or talk but not both. When no station talk, there is a natural noise, when more than two stations talk, there is interference noise, whereas when only one station talks, all the other listening stations hear its message. The model of *collision detection* studied here is that a station that listens can distinguish between 1 and $\mathbb{N}/1$ for the number of emitting stations. The radio stations are supposed to run on batteries, and a station can be asleep in order to save them. In this case the station does not listen nor talk, and cannot be waken up by neighbours.

The question answered to in this summary, is *how to elect a leader?* That is how to elect a station such that all the n stations know that this station is the leader. This process should be fast and cheap in total awake time slots of the stations. If the total number of stations n is known, there is a trivial solution for the telephone model (that can be adapted for the walkie-talkie model) where each station broadcasts with probability $1/n$, and when only one station broadcasts, it is elected. This gives an election in $O(1)$ time. So from now on, n is supposed to be totally unknown.

The leader-election problem is fundamental in distributed systems, and has various applications in military communication and cellular phones. Algorithms that answer that problem can be found in [3], and Kushilevitz and Mansour in [1] gave a lower bound of $\Omega(\log n)$ for the average time of an election. The next section provides a class of efficients algorithms, that elect a leader in $O(\log n)$ expected time, with no station being awake for more than $O(\log\log n)$ time slots. Moreover, these algorithms feature a tuning parameter α that may be adjusted in order to reduce the election time, but with a larger total awake time for the stations, or at the opposite, reduce the energy consumption, but allowing a longer election. This summary is based on the paper [2].

2. Algorithms and Main Results

2.1. **Telephone.** This algorithm relies on the intuition that in general a station does not need to be awake when it does not broadcast, since each station that broadcasts listens, and can thus testify later of what happened. This suggests that each station must choose to be awake or asleep during a predetermined sequence of time slots, and then that all stations should wake up at a predetermined moment to hear the news about the election.

The general idea is that stations broadcast with probability $1/2, 1/4, \ldots$ until only one station broadcasts, but as this may never happen, we plan rounds of predetermined length, and at the end of each round, stations wake up to hear about the possible termination. The variable $\alpha > 1$ is a tuning parameter in the algorithm presented here.

Algorithm 'Tel'

---

round $\leftarrow 1$;
Repeat
  For k from 1 to $\lceil \alpha^{round} \rceil$ **do**
    Each station wakes up independently with probability $1/2^k$ to broadcast and to listen;
    If a unique station broadcasts **then** it becomes a *candidate* station;
  EndFor
  At the end of each round, all stations wake up and all candidates stations broadcast.
  If there is a unique candidate **then** it is *elected*
  round $\leftarrow$ *round* $+ 1$;
until a station is elected.

A station is a candidate if it is the only station to broadcast at a given time, and the algorithm ends if there is only one candidate in a round. The quality of this algorithm is based on the fast decrease inside a round of the broadcasting probability. Namely, if this probability 2^{-k} is much greater than $1/n$, for instance $k \leq \log_2 n - 3$, the probability to have more than two broadcasting stations is high. On the other side, if $k \geq \log_2 n + 3$, then the probability to have no broadcasting station is high. This shows that there is few time slots that may with good probability see a candidate. So that when $round \geq \log_\alpha \log_2 n$, the probability to have an election is significant (this probability is studied in the last section).

Theorem 1. *Let $q = 0.6308$, and let c be the function defined by*

$$c(\alpha, q) = \frac{q\alpha^3}{(\alpha - 1)(1 - \alpha(1 - q))}, \tag{1}$$

then, on average, Algorithm 'Tel' elects a leader in at most $c(\alpha, q)\log_2 n$ time slots, with no station being awake for more than $2\log_\alpha \log_2 n(1 + o(1))$ mean time slots.

2.2. **Walkie-talkie.** For the walkie-talkie model, the algorithm has to face the problem that no candidate station can listen to its own message, therefore, the role of witness is introduced. A witness listens to the candidate station, and is later able to testify. Algorithm 'WT', that is close to Algorithm 'Tel', has a candidate iff at a time there is only one broadcasting station and only one witness. There is an election if there is only one candidate during the round.

Algorithm 'WT'

$round \leftarrow 1$;
Repeat
  For k from 1 to $\lceil \alpha^{round} \rceil$ **do**
    Each station wakes up independently with probability $1/2^k$;
    with probability 1/2 each awake station decides *either* to broadcast *or* to listen;
    a listening station that gets a message is a *witness*;
  EndFor
  At time $\lceil \alpha^{round} \rceil + 1$, each witness and each station that has broadcasted wakes up;
  Each witness broadcasts its received message;
  If there is a unique witness, the station that was witnessed is elected;
  At time $\lceil \alpha^{round} \rceil + 2$, all stations are listening;
    If a leader has been elected
      then the leader broadcasts and all the stations are aware of its status;
  $round \leftarrow round + 1$;
until a station is elected.

Theorem 2. *On the average, Algorithm 'WT' elects a leader in at most* $c(\alpha, q') \log_2 n$ *time slots, with no station being awake for more than* $2.5 \log_\alpha \log_2 n$ *mean time slots, with* $q' = 0.6176$.

3. Analysis

To analyze the properties of Algorithm 'Tel', like the average election time, we first study the probability to have an election at round j.

Proposition 1. *Let* p_j *denote the probability that a leader is elected at round* j*, and* $j(n)$ *be a sequence of integers such that* $n/2^{\alpha^{j(n)}} \to 0$*, then there exists* N *such that for all* $n > N$

$$p_{j(n)} \geq 0.3693.$$

The proof of this proposition is in two steps, first find a proper expression for p_j, second, find a lower bound.

There is an election at round j if there is only one candidate during this round, say at step k. So p_j is the sum on all possible k of the probability to have only one broadcasting station at step k, and no candidates during the other steps. The probability to have a candidate at step k is the probability that one station broadcasts (2^{-k}) times the probability that the others don't ($(1-2^{-k})^{n-1}$) times the number of stations (n). So p_j is written as

$$\begin{aligned} p_j &= \sum_{k=1}^{\lceil \alpha^j \rceil} \frac{n}{2^k} \left(1 - \frac{1}{2^k}\right)^{n-1} \prod_{i \neq k} \left(1 - \frac{n}{2^i}\left(1 - \frac{1}{2^i}\right)^{n-1}\right) \\ &= \sum_{m=0}^{\infty} \sum_{k=1}^{\lceil \alpha^j \rceil} \left(\frac{n}{2^k}\left(1 - \frac{1}{2^k}\right)^{n-1}\right)^{m+1} \prod_{i=1}^{\lceil \alpha^j \rceil} \left(1 - \frac{n}{2^i}\left(1 - \frac{1}{2^i}\right)^{n-1}\right) \end{aligned}$$

Name s_j the product in the equation above, then by thin upper bounds, and numerical calculations, we get an upper bound for s_j. More precisely, using the notations of Proposition 1, we have that $s_{j(n)} \geq 0.1883$.

For the other terms in p_j, we first bound $(1-2^{-k})^{n-1} = e^{(n-1)\log(1-2^{-k})} \leq e^{-(n-1)/2^k}$, and then the Mellin transform gives us that

$$\sum_{k=1}^{\lceil \alpha^j \rceil} \left(\frac{n}{2^k}\right)^{m+1} e^{-(m+1)n/2^k} \sim \frac{(m+1)!}{(m+1)^{m+2}\log 2},$$

up to minor fluctuations. Then, summing on m and doing numerical calculations give the result of Proposition 1.

Proof of Theorem 1. The value $j^* = \lceil \log_\alpha \log_2 n \rceil$ plays a crucial role in the analysis. The probability of having an election before the round j^* is small, so we assume that no election happens before this round. Then we observe that $n/2^{\alpha^{j^*+1}} \to 0$ when $n \to \infty$, so that the results of Proposition 1 applies. As the probability to have an election after j^*+1 has a constant lower bound q, the average time of the election is smaller than q^{-1}. Finally we obtain that the average number of rounds n_1 needed to elect a leader has an average smaller than $j^* + q^{-1}$. The round number i last for a time $\lceil \alpha^i \rceil$, so if T_1 denotes the time of the election, we have

$$\mathbf{E}(T_1) \leq \mathbf{E} \sum_{i=1}^{n_1} \lceil \alpha^i \rceil \leq \sum_{k=1}^{\infty} \sum_{i=1}^{j^*+k} (1+\alpha^i) q (1-q)^{k-1} \leq c(\alpha, q) \log_2 n + O(\log\log n).$$

This proves the first part of Theorem 1. For the second part, simply observe that a station is awake less than once in a row on average, so as the number of rounds is bounded by $2\log_\alpha \log_2 n$, the total number of awakening time slots for a station is smaller than $2\log_\alpha \log_2 n$. ☐

Proof of Theorem 2. The proof of Theorem 2 follows the same main lines. It expresses the probability to have an election at round j, and gives a lower bound for this quantity. Then this bound is translated into an upper bound for the average election time. ☐

4. Conclusion

The two algorithms presented here offer a quasi-optimal (up to a constant factor) way to answer the leader election problem, in the telephone or walkie-talkie model. Moreover, they are energy-saving protocols.

The tuning parameter α allow to optimize the average time complexity or the awake time slots of the n stations.

Bibliography

[1] Kushilevitz (Eyal) and Mansour (Yishay). – An $\omega(d\log(n/d))$ lower bound for broadcast in radio networks. In *SIAM Journal on Computing*, vol. 27, pp. 702–712. – 1998.

[2] Lavault (Christian), Marckert (Jean-François), and Ravelomanana (Vlady). – Quasi optimal leader election algorithms in radio networks with log-logarithmic awake time slots. – 2003. To appear.

[3] Willard (Dan). – Log-logarithmic selection resolution protocols in a multiple access channel. *SIAM Journal of computing*, vol. 15, n° 2, 1986, pp. 468–477.