



Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Information and Computation 198 (2005) 40–55

Information
and
Computation

www.elsevier.com/locate/ic

Locally guided randomized elections in trees: The totally fair case

Yves Métivier *, Nasser Saheb-Djahromi, Akka Zemmari

LaBRI, Université Bordeaux I—ENSEIRB, 351 cours de la Libération, 33405 Talence, France

Received 30 March 2004; revised 15 January 2005

Available online 19 March 2005

Abstract

We design and analyze a randomized one-passage election algorithm in trees based on a result of Angluin in [Proceedings of the 12th Symposium on Theory of Computing, 1980, pp. 82]. The election process is a distributed elimination scheme which removes leaves one-by-one reducing the tree to a single vertex, called the leader (or elected vertex). We define a locally computable parameter guiding randomly the elimination process. As a particular instance, we provide a parameter assignment in a Markovian type random process in which all vertices have the same chance of being elected.

© 2005 Elsevier Inc. All rights reserved.

Keywords: Distributed algorithm; Election; Probabilistic analysis; Randomized algorithm; Stochastic process

1. Introduction

1.1. The distributed network model

Our model follows the standard models for distributed systems given in [9,5,12]. The communication model is a point-to-point communication network which is represented as a simple

* Corresponding author.

E-mail addresses: metivier@labri.fr (Y. Métivier), saheb@labri.fr (N. Saheb-Djahromi), zemmari@labri.fr (A. Zemmari).

connected undirected graph, where the vertices represent processors and two vertices are linked by an edge if the corresponding processors have a direct communication link. In this paper, we assume that the graph is a tree: a connected graph without cycle. The network is anonymous: unique identities are not available to distinguish processors. Processors communicate by message passing, and each processor knows from which channel it receives a message. Our algorithm is synchronous and timer-based: the processors have access to a physical clock device and as in ([12] page 87):

- (1) we assume that the timer is a global continuous time to which processors have access: a real-valued variable whose value continuously increases in time,
- (2) we make the global time assumption, that is each event is said to take place at a certain time and each event itself is assumed to have the duration 0.

1.2. The problem

The election problem is to choose exactly one element in the set of processors. Thus, starting from a configuration where all processors are in the same state, we must obtain a configuration where exactly one processor is in the state leader and all other processors are in the state lost. The leader can be used subsequently to make decisions or to centralize some information.

The main goal of this paper is the study of chances of being elected for the vertices of a tree in a randomized process described below.

1.3. Known results

The election problem is standard and many solutions are known [10,12]. It was first posed by LeLann [6]. It has been solved under many different assumptions: the graph can be a ring, a tree, a complete graph or a general connected graph. The system can be synchronous or asynchronous. The graph can be directed or undirected. The processors might or might not know the size of the graph. Communications can be synchronous or asynchronous.

In the case of trees, an election process can roughly be viewed as a leaf-removal-sequence, yielding a unique vertex. It was first presented by Angluin ([1] Theorem 4.4). For the sequence viewed as a random process, the probability of being elected for the vertices depends on how one introduces the choice of the vanishing leaves. In [7] the authors consider two elementary approaches. The first one is based on the assumption that *all sequences* of the election process have the *same* probability (the **p**-distribution). The second one assumes that, at each step, *all leaves* have the *same* probability of being removed (the **q**-distribution). In the first approach the computation of the involved probabilities is transformed into a mathematically interesting enumeration problem, which in turn provides a simple characterization of the searched probabilities on the set of vertices. The difficulty, however, resides in that no one-passage-local-randomized elimination process, without a preliminary computation on the tree is available to implement the approach. The second approach can be implemented in a distributive way and the probability of being elected is computed inductively. Its disadvantage is that the generated probability distribution, being of a rigid nature, does not allow any simple characterization.

1.4. The main results

The randomized election we introduce and study here is based on the work of Angluin [1]. A tree T of size n is initially given. The elected vertex is the surviving one, following an $(n - 1)$ -sequence of leaf-eliminations.

We introduce in the algorithm of Angluin a locally computable parameter guiding the elimination process. We use the description of Tel ([12] page 192, Algorithm 6.3); we give the local algorithm of a process whose name is used as a subscript to the variables.

In Algorithm 1, the statement *wait-exponential* ($\lambda(v)$) generates an exponentially distributed random waiting time T of parameter $\lambda(v)$; thus the waiting time T is nothing but

$$(-1/\lambda(v)) \ln(x),$$

where x is an uniform real-valued random variable in $[0, 1]$. Thus, our algorithm uses a random exponentially generated lifetime delay for each leaf.

```

var  $Neigh_v$  : set of nodes (* The neighbors of  $v$  *);
 $rec_v[w]$  for each  $w \in Neigh_v$  : boolean init false;
(*  $rec_v[w]$  is true if  $v$  has received a message from  $w$  *);
 $\lambda(v)$  : the parameter with the same initial value for all vertices;
(* In the uniform case, initially  $\lambda(v) := 1$  *);
begin
  while  $\#\{w : rec_v[w] \text{ is false}\} > 1$  do
    begin
      receive  $\langle \mathbf{tok}, \lambda(w) \rangle$  from  $w$ ;
       $\lambda(v) := F(\lambda(v), \lambda(w))$ ;
      (* $F$  is any function; in the uniform case we take  $F(\lambda(v), \lambda(w)) = \lambda(v) + \lambda(w)$ *)
       $rec_v[w] := true$ 
    end
    (* Now there is one  $w_0$  with  $rec_v[w_0]$  is false; the vertex  $v$  is a leaf *)
    wait-exponential ( $\lambda(v)$ );
    if a message  $\langle \mathbf{tok}, p \rangle$  has arrived then  $state_v := leader$ 
    else
      begin
         $state_v := lost$ ;
        send  $\langle \mathbf{tok}, \lambda(v) \rangle$  to  $w_0$  with  $rec_v[w_0]$  is false
      end
    end

```

Algorithm 1. Randomized Election in a Tree.

The algorithm is initiated by leaves at time 0. Each vertex waits for messages until it has received messages from all but one of its neighbors. Each vertex v , which becomes a leaf has an exponentially distributed lifetime of parameter $\lambda(v)$ determined at the time when v becomes a leaf. A leaf v which has expired its lifetime and has not received a message disappears with the incident edge: its state

is *lost*; it sends to its father a message with the value $\lambda(v)$. The parameter $\lambda(v)$ of a vertex v which receives a message $\langle \mathbf{tok}, \alpha \rangle$ with a value α becomes equal to $F(\lambda(v), \alpha)$. A vertex is elected if it receives a message before the end of its lifetime. We assume that communications take no time, thus the delay between the elimination of a leaf and the notification to its father is equal to 0. The time is continuous so that two messages are never sent simultaneously; hence the elected vertex is the unique vertex which received a message from all neighbors (see [12] page 193).

According to the choice of F we obtain different distributions of chances of being elected. The election process is modeled by a time-continuous death process as follows. Each vertex v , which becomes a leaf has an exponentially distributed lifetime of parameter $\lambda(v)$, supposed to be determined at the time when v becomes a leaf. This assumption is equivalent to the one that the death probability of v in the time interval $[t, t + h]$ is $h\lambda(v) + o(h)$, as $h \rightarrow 0$ at any time t , and this is independent of what is going on elsewhere and of what happened in the past. This assumption seems to us *natural* in the framework of *distributed* algorithms. A leaf which has expired its lifetime disappears with the incident edge. The process continues until the tree is reduced to a single vertex. This general model leaves open the choice of $\lambda(v)$ depending on F . All we require is that $\lambda(v)$ must be locally determined when v becomes a leaf; it can be computed as and when vertices are removed. The intuitive idea behind the model is to control chances of election by the λ -assignment: if $\lambda(v)$ increases then the probability of v being elected decreases.

If the parameter λ is the same for any vertex, then to each leaf-removal at each step is assigned the same probability as in the second approach in [7] and, therefore, the probability of being elected will be the \mathbf{q} -distribution. Consequently, the present model extends the second approach introduced in [7] (regarding the generated probability distribution). It is, however, possible to assign different death rates $\lambda(v)$ to vertices according to their initial position in the tree. Clearly, different considerations lead to different probability distributions in the election process.

A naturally arising question will be how to choose the λ s, and give all vertices of T the same chance of being elected, a task which a priori seems difficult, since the interior vertices of a tree are more covered than the peripheral ones.

Our main result, Theorem 1, states precisely that this particular instance of the general model, called the uniform model, can be achieved by equating $\lambda(v)$ to the size of the subtree of T constituted of the root v and all its removed successor vertices. This assignment can be performed in a distributed way as follows. Initially, all vertices of T have the same weight 1 : $w(v) = 1$. For a leaf v , the parameter $\lambda(v)$ is its weight: $\lambda(v) = w(v)$. As a leaf is removed its father collects its weight, i.e., the weight of the father is incremented by the weight of the removed son: $F(\lambda(v), w(v)) = \lambda(v) + w(v)$.

Finally, we can note that the same algorithm can be used for any connected network as soon as a spanning tree of the network is available.

1.5. Tools

The election process on a tree is viewed as an irreversible continuous-time Markov process. The behavior of such processes is generally ruled by a system of differential equations subject to an initial condition. The analysis of the general model can theoretically be achieved in terms of the solution of the system. In particular, the probability of being elected is nothing but the absorption probability. However, the system does not allow an explicit solution except in very simple cases.

The study of the uniform model brings into play some elementary though powerful tools from probability theory, such as evaluation techniques for the sum and maximum of independent random variables. We believe that the basic assumption of exponentially distributed waiting times separating local events, linked to Poisson processes, is a natural assumption in this context and, thus, variants of the present model may be adopted in the analysis of randomized distributed algorithms where the randomization affects local operations duration (random delays).

1.6. Basic notions and notation

A *tree graph* (or simply a *tree*) T is a connected undirected graph containing no cycles, see [2,11]. Its *size* n is the number of its vertices. The vertices of degree less than or equal to 1 are called *leaves* (in the case of trees of size $n \geq 2$ the leaves are of degree 1). If v is a leaf in the tree T and w is the neighbor vertex of v , then w is the *father* of v or v is a *son* of w . If the tree is reduced to two vertices each of them are both son and father of the other vertex. A *factor* T' of a tree T is a subgraph of T which is a tree.

In the sequel, we have to borrow some simple concepts from directed graphs. An *arborescence* A (or *rooted tree*) is a tree in which a distinguished vertex r is considered as the *root* and the edges (or arcs) are directed in such a way that there exists a path (in A) from the root to all vertices. If A is an arborescence then for a given vertex v of A the subgraph of A induced by all vertices of A , which are accessible to v , is the *subtree of A rooted at v* . A *forest* is a set of pairwise disjoint arborescences. In an arborescence or in a forest a leaf denotes a vertex without son (successor).

1.7. Summary

The paper is organized as follows. In Section 2 we give simple properties which are common in the general model. We introduce a particular instance of the model and prove the uniformity property in Section 3. The distribution of the uniform election duration is studied in Section 4 for two particular instances of trees: chain graphs and star graphs. Section 5 is devoted to some open questions and to new directions to complete the present study.

This paper is an improved version of the extended abstract [8].

2. Simple properties of the general model

Once a vertex v has become a leaf, its death happens according to a Markovian process with a given parameter $\lambda(v)$, which is supposed to have been determined by some specific rules detailed in the sequel. Each leaf v of the residual tree behaves *independently* and, under the assumption that it is alive at time t , the probability of its death in the interval $[t, t + h]$ is given by $\lambda(v)h + o(h)$, as $h \rightarrow 0$. We refer to $\lambda(v)$ as the *death rate* on v . The quantity $1/\lambda(v)$ is the mathematical expectation of the lifetime of v (from the instant when it becomes a leaf). It should be noted that $\lambda(v)$ is *not* a fixed parameter initially assigned to v ; it can be a function of values transmitted to v (by the vanished adjacent vertices). In particular, if v has several neighbor vertices, at the time v becomes a leaf only one neighbor is surviving and, $\lambda(v)$ computed step-by-step, may depend on the disappeared neighbor vertices.

We first recall basic properties associated with Markovian processes (see [3], Chapter XVII).

Proposition 1. *Let v be a vertex of a tree T becoming a leaf at some instant t . A death rate $\lambda(v)$ is assigned to v at this instant. Let $L(v)$ be the lifetime of v (from this instant). Then*

(i) $L(v)$ is an exponentially distributed r.v. (random variable) with parameter $\lambda(v)$

$$Pr(L(v) > x) = e^{-\lambda(v)x}, \quad \forall x \geq 0.$$

(ii) $L(v)$ has the mathematical expectation $\frac{1}{\lambda(v)}$.

(iii) $L(v)$ satisfies the “memoryless” property

$$Pr(L(v) > x + y \mid L(v) > x) = Pr(L(v) > y) = e^{-\lambda(v)y}, \quad \forall x, y \geq 0.$$

At some instant t , the tree T of size n is reduced to a factor tree T' of size n' , $2 \leq n' \leq n$, with leaves $v_1, \dots, v_k, k \geq 2$, having death rates $\lambda(v_1), \dots, \lambda(v_k)$, respectively. Let $L(v_i)$ be the r.v. denoting the lifetime of $v_i, 1 \leq i \leq k$. The death rate for the tree T' is $\lambda(T') = \sum_{i=1}^k \lambda(v_i)$; meaning that, since $L(v_i)$ are totally independent, the probability of a death in $[t, t + h]$ is $h \sum_{i=1}^k \lambda(v_i) + o(h)$, as $h \rightarrow 0$. Therefore, we may consider $\lambda(T')$ as the rate of parallelism on the factor tree T' . This tree has the lifetime $L(T')$ (from the time T' appears, till the time when one of its leaves disappears) which is a r.v. defined by $L(T') = \min_{1 \leq i \leq k} L(v_i)$, since the next reduction will take place at the time $t + \min_{1 \leq i \leq k} L(v_i)$. This r.v. is characterized by its distribution function

$$Pr(L(T') \leq x) = 1 - Pr(L(T') > x) = 1 - e^{-x\lambda(T')}, \quad \forall x \geq 0.$$

Thus, $L(T')$ is an exponentially distributed r.v. with mathematical expectation $\frac{1}{\lambda(T')}$.

If we have to compute the probability $q_T(v)$ of being elected for the vertex v in the tree T , we should sum the probabilities of transitions beginning with the initial tree T , yielding at the end the irreducible vertex v according to the introduced irreversible random process. Let us reconsider the case of the factor T' . Identifying the state of the process at some instant t by the residual factor tree at this instant, the next configuration will be one of the factor trees $T_i = T' \setminus \{v_i\}$ obtained by removing the leaf v_i and the incident edge, for $i = 1, \dots, k$. Each of these “next-state” has a probability given by:

Proposition 2. *Let T' be the state of the election process at some instant t . Let $v_1, \dots, v_k, k \geq 2$, be the leaves of T' . Then the state of the process is one of the trees T_i obtained from T' (by removing v_i and the incident edge), with the transition probability*

$$p(T', T_i) = \frac{\lambda(v_i)}{\lambda(T')} = \frac{\lambda(v_i)}{\sum_{1 \leq j \leq k} \lambda(v_j)}, \quad 1 \leq i \leq k.$$

Proof. According to the previous proposition the random variables $L(v_i)$ are memoryless and, therefore, the distribution of the extra-lifetime from a given instant t , subject to the condition of the survival up to this instant, is the same as the initial distribution for the lifetime. On the other hand $L(v_i)$, for $1 \leq i \leq k$, are independent and thus

$$\begin{aligned}
p(T', T_i) &= \Pr\left(\min\{L(v_1), \dots, L(v_k)\} = L(v_i)\right) \\
&= \Pr\left(L(v_i) \leq L(v_j) \mid 1 \leq j \leq k\right) \\
&= \int_0^{+\infty} \lambda(v_i) e^{-\lambda(v_i)x_i} \left(\prod_{j=1, j \neq i}^k \int_{x_i}^{+\infty} \lambda(v_j) e^{-\lambda(v_j)x_j} dx_j \right) dx_i \\
&= \frac{\lambda(v_i)}{\sum_{1 \leq j \leq k} \lambda(v_j)}. \quad \square
\end{aligned}$$

Remark 3. In the case where all vertices have the same assigned parameter λ , the probability of being elected is the \mathbf{q} -distribution given in [7], since according to Proposition 2, all transitions from a given factor tree T' have the same probability.

Let T be a tree of size $n \geq 2$. Consider the sequence $\sigma = \langle \sigma_1, \sigma_2, \dots, \sigma_n \rangle$, with $\sigma_1 = T$, of factor trees, such that each σ_i is obtained from σ_{i-1} by a transition (i.e., removing a leaf), for $2 \leq i \leq n$. Clearly, the probability of such a sequence should be defined by $p(\sigma) = \prod_{i=2}^n p(\sigma_{i-1}, \sigma_i)$. Denoting by $\Omega(T)$ the set of all sequences, the probability of being elected for a given vertex v is $\sum_{\sigma \in \Omega(T), \sigma_n = \{v\}} p(\sigma)$.

Another parameter of interest, which may be subject of further studies, is the expected time in the election process.

For a given $\sigma = \langle \sigma_1, \sigma_2, \dots, \sigma_n \rangle \in \Omega(T)$, its duration $D(\sigma)$ is the sum of the lifetimes of disappearing leaves in σ . The expected duration of the election process is $\sum_{\sigma \in \Omega(T)} p(\sigma) E(D(\sigma))$, where $E(D(\sigma))$ is the mathematical expectation of $D(\sigma)$. In Section 4, we will study the election duration for a particular instance of the model in the case of chain and star graphs.

The election process over T is in fact a Markov continuous-time process. The set of states is the set of factor trees of T , the initial state is T and the transitions are defined by the leaf-removal process, with probabilities assigned as defined above. According to this point of view we can state the following proposition, which *in principle*, allows to represent mathematically all the parameters of interest.

Proposition 4. Let T' be a factor. Let $P_{T'}(t)$ denote the probability that the state of the election at time t is T' . We have

- (i) $\frac{dP_T(t)}{dt} = -\lambda(T)P_T(t)$,
- (ii) for all factor tree $T' \neq T$ of size at least 2, $\frac{dP_{T'}(t)}{dt} = -\lambda(T')P_{T'}(t) + \sum_v \lambda(v)P_{T' \cup \{v\}}(t)$, where the summation is extended to all vertices v adjacent to T' and which do not belong to T' , and
- (iii) for all vertices v , $\frac{dP_{\{v\}}(t)}{dt} = \sum_{v'} \lambda(v')P_{\{v, v'\}}(t)$,

with the initial condition $P_T(0) = 1$.

Proof. The following proof is a straightforward adaptation of a well-known method (see [3], Chapter XVII, Section 5), based on analyzing the probability of transitions from neighbor states to the state T' in the time interval $[t, t + h]$.

Consider the evolution of the process in the interval $[t, t + h]$. Let T' be a tree factor of T . Let us compute the probability of being in the state T' at time $t + h$ in terms of the state at time t .

- For $T' \neq T$ and not reduced to a leaf, we have

$$P_{T'}(t + h) = \sum_v P_{T' \cup \{v\}}(t) \pi_{T' \cup \{v\}, T'}(h) + P_{T'}(t) \pi_{T', T'}(h) + o(h),$$

where $\pi_{S, T'}(h)$ is the probability of a (direct) transition from S to T' in a time interval of length h ; the summation is extended to all vertices v adjacent to T' and not belonging to it. We get

$$P_{T'}(t + h) = h \sum_v \lambda(v) P_{T' \cup \{v\}}(t) + P_{T'}(t) [1 - \lambda(T')] + o(h).$$

Hence

$$\frac{P_{T'}(t + h) - P_{T'}(t)}{h} = -\lambda(T') P_{T'}(t) + \sum_v \lambda(v) P_{T' \cup \{v\}}(t) + \frac{o(h)}{h}.$$

This proves (ii).

- To prove (i), we remark that in the case of $P_T(t + h)$, the above sum ($\sum_v \dots$) on the right vanishes, since T has no preceding factor trees. This establishes (i).
- To show (iii), it suffices to remark that the state $\{v\}$ is an absorbing one and, therefore, $\pi_{\{v\}, \{v\}}(h) = 1$. Thus in $P_{\{v\}}(t + h)$, the negative term vanishes. A simple computation yields (iii). \square

The last proposition is mathematically very strong, since it provides a general solution for the probability of being elected and the duration of the election in terms of probability of states given as the solution of a differential equation system. However, no solution seems to be available, except in obvious instances and more techniques should be developed for an efficient computation.

3. Uniform election model

We study a particular instance of the general model. Let $T = (V, E)$ be a tree. Initially, all vertices have the same *weight* 1 : $w(v) = 1, \forall v \in V$. As a leaf vanishes, its father collects its weight, adding it to its current weight. At the time where a vertex v becomes a leaf in a residual tree T' , its weight is the number of vanished vertices on its side. The lifetime for v , according to this model, is a r.v. having an exponential distribution of parameter $\lambda(v) = w(v)$. As we shall see this strategy leads to a *totally* “fair” randomized election : in a tree of size n all vertices have the same probability $\frac{1}{n}$ of being elected. Let us first consider an example.

Example 5. Let T be the tree of Fig. 1. Initially, all vertices have the same weight 1.

There are two transition sequences leading to the choice of the vertex labeled a . These two sequences may be identified by the corresponding leaf-removal sequences $\langle e, d, c, b \rangle$ and $\langle d, e, c, b \rangle$. Consider the first one. Initially, there are 3 leaves of weight 1 and thus the probability of the e -removal is $\frac{1}{3}$ (Proposition 2). In the reduced tree there are now two leaves of weight 1 and thus the

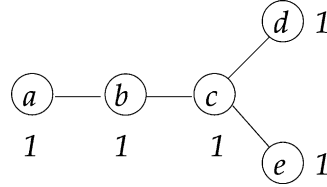


Fig. 1. A tree with initial weights.

probability of the d -removal is $\frac{1}{2}$. Now in the factor tree on a, b, c the leaf a is of weight 1 and c of weight 3. Since the parameter λ for a leaf is nothing but its weight, according to Proposition 2, the probability of the c -removal is 3 times the probability of the a -removal and is therefore $\frac{3}{4}$. The same argument applied to the tree factor consisting of a and b , assigns the probability $\frac{4}{5}$ to the removal of b . Hence the probability of the first sequence is $\frac{1}{3} \frac{1}{2} \frac{3}{4} \frac{4}{5} = \frac{1}{10}$. Clearly, the second sequence $\langle d, e, c, b \rangle$ has the same probability and therefore the vertex a is elected with probability $\frac{1}{5}$.

A similar computation shows that any other vertex has the same probability $\frac{1}{5}$ of being elected.

To prove this main assertion of uniformity, we first introduce a slight modification of the model introduced in the section. Maybe, the following proof is not the shortest one. It has, however, the advantage of translating the model into a variant on directed trees which in turn seems to be interesting. Let $F = (V, A)$ be a forest of arborescences. We consider a randomized election process over F as follows. At any time interval $[t, t + h]$, any leaf v (recall that a leaf in an arborescence has no successor) may vanish with a probability proportional to its weight (initially all vertices are weighted 1). If a vanishing leaf has a father (i.e., a predecessor vertex) it transmits its weight to the father whose weight will be added to the weight of the vanishing son. The process goes on within the reduced forest obtained from F by removing this vanishing leaf and the incident arc (if it exists), until the forest is totally removed. Since the considered graphs are no longer connected, we preferably use terms such as eliminating (or death) process, beating vertex here (the last surviving vertex).

Example 6. Consider the forest of Fig. 2. Let us compute the probability that x beats other vertices of the forest.

This event is the consequence of one among the eight sequences of leaf-removal: $\langle b, c, a, d \rangle$, $\langle b, c, d, a \rangle$, $\langle b, d, c, a \rangle$, $\langle d, b, c, a \rangle$, $\langle c, b, a, d \rangle$, $\langle c, b, d, a \rangle$, $\langle c, d, b, a \rangle$, and $\langle d, c, b, a \rangle$. Consider the first sequence $\langle b, c, a, d \rangle$. Since initially there are 4 leaves with weight 1 each one, the b -removal has probability $\frac{1}{4}$. In the reduced forest the probability of the c -removal is $\frac{1}{3}$. Now, in the remaining forest there are 3 leaves; x and d have weight 1 and a has weight 3. Hence the probability of the a -removal

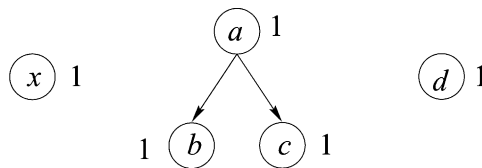


Fig. 2. An example of a weighted forest.

is $\frac{3}{5}$. There remains finally x and d of weight 1 each one and the probability that x survives to d becomes $\frac{1}{2}$. The probability of the sequence $\langle b, c, a, d \rangle$ is therefore $\frac{1}{4} \frac{1}{3} \frac{3}{5} \frac{1}{2} = \frac{1}{40}$. The other probabilities are computed similarly, yielding the value $\frac{1}{5}$ for x to beat other vertices.

As we shall see further this new death process, related to the uniform election model, has the advantage of admitting an inductive reasoning. One can write a system of differential equations for the process as we did in Proposition 4 for the general model on trees. We do not go, however, in this direction. Instead, we find the distribution of the waiting time for a forest to disappear. Then we prove that the probability that a given forest survives another one depends on nothing but the sizes of each forest. We derive finally the uniformity property of the model introduced in this section.

Given a forest F , let $W(F)$ denote the time it takes to disappear, which is a positive real-valued r.v. Given two forests F_1 and F_2 , F_1 survives (or beats) F_2 if $W(F_1) > W(F_2)$.

The following proposition may seem surprising, it asserts that the distribution of $W(F)$ depends only on the size of F .

Proposition 7. *Let F be a forest of size n then the distribution function $G_F(t)$ of the r.v. $W(F)$ is given by*

$$G_F(t) = Pr(W(F) \leq t) = (1 - e^{-t})^n, \quad \forall t \geq 0.$$

Proof. By induction on n . If F is a vertex, then the proposition holds (the lifetime for a single vertex is an exponentially distributed r.v. with parameter 1). Suppose that the proposition holds for forests of size less than n and let us prove it for a forest F of size n ($n \geq 2$).

- (i) Suppose that F consists of forests F_1, F_2, \dots, F_k with $k \geq 2$. Let $n = n_1 + n_2 + \dots + n_k$, where n_i is the size of F_i , $1 \leq i \leq k$. In this case $W(F_i)$, $1 \leq i \leq k$, are mutually independent r.v. and hence by the induction hypothesis

$$\begin{aligned} Pr(W(F) \leq t) &= \prod_{i=1}^k Pr(W(F_i) \leq t) \\ &= \prod_{i=1}^k (1 - e^{-t})^{n_i} \\ &= (1 - e^{-t})^n. \end{aligned}$$

- (ii) Otherwise, consider that F has size n and consists of a unique root r and arborescences A_1, A_2, \dots, A_k . Now, let F' be the forest consisting of A_1, A_2, \dots, A_k . F' has size $n - 1$ and, by the induction hypothesis, $(1 - e^{-t})^{n-1}$ is the distribution function of $W(F')$. But $W(F)$ is the sum of two independent r.v. $W(F')$ and the lifetime of r . The last one is an exponential r.v. of parameter n (weight of r). Thus $W(F)$ has the distribution function (see [4], p. 142, Theorem 2) given by

$$G_F(t) = \int_0^t G_{F'}(t - x) d(1 - e^{-nx}).$$

Hence

$$\begin{aligned}
 G_F(t) &= \int_0^t n[1 - e^{-(t-x)}]^{n-1} e^{-nx} \, dx \\
 &= \int_0^t n[e^{-x} - e^{-t}]^{n-1} e^{-x} \, dx \\
 &= [-(e^{-x} - e^{-t})^n]_{x=0}^{x=t} \\
 &= (1 - e^{-t})^n.
 \end{aligned}$$

The proposition follows. \square

As an easy consequence of the proposition, we get the following results on the probability of survival.

Corollary 8.

- (i) Let F_1 and F_2 be two forests of sizes n_1 and n_2 , respectively. Then the probability that F_1 beats F_2 is $\frac{n_1}{n_1+n_2}$. In particular, the probability that an isolated vertex x beats a forest of size n is $\frac{1}{n+1}$.
- (ii) Given a forest F of arborescences A_1, A_2, \dots, A_k . Then, for any root r_i of A_i , the probability of being elected is proportional to the inverse of the size of A_i ($1 \leq i \leq k$).

Proof. We show only (i); (ii) is proved similarly. The probability that F_1 beats F_2 is $\Pr(W(F_1) \geq W(F_2))$. On the other hand, $W(F_1)$ and $W(F_2)$ are independent r.v. with the distribution functions $G_{F_1}(t_1)$ and $G_{F_2}(t_2)$ given in Proposition 4. A simple computation yields

$$\Pr(W(F_1) \geq W(F_2)) = n_1 \int_0^\infty (1 - e^{-t_1})^{n_1+n_2-1} e^{-t_1} \, dt_1 = \frac{n_1}{n_1 + n_2}. \quad \square$$

Remark 9. Returning to the case of undirected trees, we have to point out that the equivalent of the previous propositions does not hold for trees. Indeed, given a tree T of size n and an isolated vertex x not belonging to T , if one considers the election process, as introduced in Section 2 (each leaf has a lifetime which is an exponentially distributed r.v. with the parameter $\lambda(v) = w(v)$), the probability of being elected for the isolated x is not $\frac{1}{n+1}$. In fact it is conjectured to be $\frac{2(n-1)}{n^2}$. It should be noted that in this case, it is supposed that, since at each step the weight of a leaf may change (once it becomes isolated), the extra lifetime is generated according to the last weight.

We are ready to prove the main result of this section.

Theorem 10. Let T be a given tree of size n and let v a given vertex of T . If $q_T(v)$ denotes the probability of being elected for v in T according to the uniform model, then $q_T(v) = \frac{1}{n}$.

Proof. If $n = 1$ or $n = 2$ the theorem is obviously true. Otherwise suppose a_1, \dots, a_k be adjacent vertices to v . Let A_1, \dots, A_k be disjoint arborescences rooted at a_1, \dots, a_k of sizes n_1, \dots, n_k , respectively. Then the vertex v is beaten in T if and only if one of the roots a_i in A_i beats v_i in the residual

arborescence rooted at v to the vertices which are not in A_i —call this arborescence B_i —(for some $i \in \{1, \dots, k\}$). But these events are pairwise disjoint for different i 's. Hence

$$1 - q_T(v) = \sum_{i=1}^k q(a_i; \{A_i, B_i\}).$$

According to Corollary 1, $q(a_i; \{A_i, B_i\})$ is equal to $\frac{n_i}{n}$ and, therefore, $1 - q_T(v) = \frac{n-1}{n}$, since $\sum_{i=1}^k n_i = n - 1$. The theorem follows. \square

4. Election duration in the uniform model

A parameter of interest which comes immediately after the probability distribution over vertices is the election duration. It may be considered as a measure of the time complexity. This is the absorption time for the continuous-time Markovian process considered in Section 2. Theoretically, Proposition 3 determines the probability of being in the state T' (recall T' is a factor tree of the initial tree T) at time t . However, the differential equation systems yielding these probabilities are quite difficult to solve. The study seems interesting since it must be common to large classes of pure death processes. No substantial result in this direction is known to the authors. We consider here two very special instances of the problem for which an elementary computation provides the probability of being in the state T' .

Results presented in this section are simple. Indeed, the motivation is the presentation of the following conjecture. Although, nothing is known at present about the election duration and its expectation in general trees, the results on the mathematical expectation of the duration in chains and stars suggest that:

Conjecture 11. *The average duration in the uniform model is likely to be logarithmic in the size of the tree.*

4.1. Election duration in a chain graph

We reconsider the uniform model introduced in the previous section. As a simple example we suppose that the tree T in which the election process takes place is a chain of length n . For the positive integer $1 \leq m \leq n$, let $P_m(t)$ denote the probability that at time t the residual chain has the size m . It should be noted that this identification of states differs from the one used in the context of Proposition 3, where the state of the process at time t was initially defined as the residual factor tree. By contrast, herein and in the following subsections, the state of the process at time t is defined as the *size* of the residual subgraph (chain, star, etc.) We have :

Theorem 12. *Under the above assumptions, we have*

(i) $P_{n-k}(t) = (k + 1)e^{-(k+2)t}(e^t - 1)^k$, if $0 \leq k \leq n - 2$ and

(ii) $\frac{dP_1(t)}{dt} = (n - 1)e^{-nt}(e^t - 1)^{n-2} = n(n - 1)e^{-2t}(1 - e^{-t})^{n-2}$.

Proof. Since we identify the state of the Markov process with the size of the residual chain, any chain of size $n - k$, $1 \leq k \leq n - 1$, is the result of a chain of size $n - k + 1$ and produces in turn a chain of size $n - k - 1$. The same fact holds for a chain of size n and a chain of size 1 with the exception that for the first one no preceding chain exists and that for the second one no following one exists. Using the same technique as in Proposition 1, we get

- (a) $\frac{dP_n(t)}{dt} = -2P_n(t)$, since $\lambda_n = 2$.
- (b) $\frac{dP_{n-k}(t)}{dt} = -(k+2)P_{n-k}(t) + (k+1)P_{n-(k-1)}(t)$, for $1 \leq k \leq n - 2$, and
- (c) $\frac{dP_1(t)}{dt} = nP_2(t)$, for $k = n - 1$, $\lambda_1 = n$.

with the initial condition $P_n(0) = 1$. The first equation subject to the initial condition yields $P_n(t) = e^{-2t}$, which asserts that (i) holds for $k = 0$. Let (i) be true for $k = l$ and prove it for $k = l + 1$ whenever $l + 1 \leq n - 2$. Thus, according to (b), we have

$$\frac{dP_{n-(l+1)}(t)}{dt} = -(l+3)P_{n-(l+1)}(t) + (l+2)(l+1)e^{-(l+2)t}(e^t - 1)^l.$$

The last linear equation is easily solved, yielding

$$P_{n-(l+1)}(t) = (l+2)e^{-(l+3)t}(e^t - 1)^{l+1}.$$

This establishes (i). Now setting $P_2(t) = (n-1)e^{-nt}(e^t - 1)^{n-2}$ in (c), we derive (ii). \square

Let D_n denote the election duration in a chain of size n . It is a positive real-valued r.v. which is determined by (c) in the previous theorem. Indeed, we have :

Corollary 13. *The election duration D_n has the density function*

$$f(t) = \begin{cases} n(n-1)e^{-2t}(1 - e^{-t})^{n-2} & \text{if } t > 0 \\ 0 & \text{otherwise.} \end{cases}$$

It is linked to a beta distribution: the r.v. $X_n = e^{-D_n}$ has the density $\beta_{n-1,2}$ (see [4], II.4). Its mathematical expectation is $H_n - 1 = \sum_{i=2}^n \frac{1}{i}$, where H_n is the n th harmonic number.

Proof. We have clearly $\Pr(D_n \leq t) = P_1(t)$ for $t \geq 0$. Thus, D_n has a distribution function which is the solution of (ii) in the previous theorem. It has, therefore, the density function $f(t)$. To compute the density of X_n , we use the change of variable $x = e^{-t}$, getting

$$g(x) = f(t(x)) \left| \frac{dt}{dx} \right| = n(n-1)(1-x)^{n-2}x,$$

with $x \in (0, 1)$. This is the beta density

$$\beta_{n-1,2}(x) = \frac{\Gamma(n+1)}{\Gamma(n-1)\Gamma(2)}(1-x)^{n-2}x, \quad x \in (0, 1).$$

A standard but long computation yields the mathematical expectation of D_n , which is $-\int_0^1 \ln(x) \beta_{n-1,2}(x) dx$. A simple method to calculate it is to sum the mathematical expectations of the time

for transitions from a chain of size i to a chain of size $i - 1$, for $i = n, n - 1, \dots, 2$. Since the rate of the transition $i \rightarrow i - 1$ is $n - i + 2$, the expected time of this transition is $\frac{1}{n-i+2}$. The sum of these fractions is $H_n - 1$. \square

4.2. Election duration in a star graph

As the second instance of the study, let us consider the election duration of the uniform model in a star graph G of size n . The star graphs and the chain graphs constitute two extreme instances of trees. This is why these very particular cases seem interesting.

Using the same notation as in the case of chains, we have

Theorem 14. *For a given integer k , $1 \leq k \leq n$, the probability $P_k(t)$ of being in the state k (i.e., a star of size k) is*

$$P_k(t) = \binom{n-1}{k-1} (1 - e^{-t})^{n-k} e^{-(k-1)t}.$$

Proof. A computation similar to the one in the case of chains yields

- $\frac{dP_n(t)}{dt} = -(n-1)P_n(t)$,
- for $2 \leq k \leq n$, $\frac{dP_k(t)}{dt} = -(k-1)P_k(t) + kP_{k+1}(t)$, and
- $\frac{dP_1(t)}{dt} = 2P_2(t)$

with the initial condition $P_n(0) = 1$. Now a straightforward computation, using a decreasing induction over k proves the theorem. \square

The theorem characterizes in particular the uniform election duration D_n in a star graph of size n :

Corollary 15. *The r.v. D_n has a distribution admitting the following density function*

$$f(t) = \begin{cases} (n-1)(1 - e^{-t})^{n-2} e^{-t} & \text{for } t \geq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Its mathematical expectation is $H_n - 1 = \sum_{i=2}^n \frac{1}{i}$.

Proof. Clearly D_n has the distribution function $P_n(t)$, given by the above theorem. Hence it admits the given density function. The same technique used for the chain graphs can be applied to obtain the mathematical expectation $H_n - 1$. \square

Remark 16. It is also possible to use a more direct method to prove the last theorem and its corollary. Indeed, the probability of having a star of size k at time t is the probability that exactly $n - k$ leaves of the initial n -star have vanished at time t . Since they behave independently, the probability of this event is

$$\binom{n-1}{n-k} (1 - e^{-t})^{n-k} e^{-(k-1)t},$$

which is the same as the expression of $P_k(t)$ given by the theorem.

Remark 17. We saw in the case of death process on forests (with an upward elimination) the absorption time W is the maximum of n independent identical exponentially distributed r.v. It should be noted that W includes the waiting time for the last vertex (or elected one) to vanish.

Although D and W are not identical, the studied cases yield distribution functions for D which are ‘similar’ to that of W . There may be a simple common feature for the distribution of the election duration in general.

5. Further investigations

So far, we have proposed a typical instance of a general model of “guided elections” in a tree for which we have proposed a mathematical framework. In such a setting, many interesting questions arise about the power and limitation of the general model. Can any probability distribution over the set of vertices of a tree be generated by an instance of the model, by using only one passage for computing λ as soon as neighboring leaves vanish? The answer seems to be negative, since no system is available that uses a locally computable one-passage λ leading to the \mathbf{p} -distribution (the probability distribution assigning the same probability to all leaf-removal-sequences, see [7]) is available. If such is the case, investigating the characterizations of the probability which can be generated seems desirable.

A study based on simulations suggests that if the expected lifetime of a vertex is chosen to be its weight (instead of the inverted weight in the uniform model), then the election process favors the median vertices (i.e., the medians have the greatest probability of being elected). The same technique suggests that the choice of a greater distance from the removed leaves plus one for the expected lifetime of a given vertex favors centers. However, up until now, there is no available analytical confirmation provided which favors such directions of research.

Acknowledgments

The authors are grateful to Philippe Duchon and Bernard Ycart for helpful discussions. Additionally, they like to express their gratitude to the referees for their careful reading.

References

- [1] D. Angluin, Local and global properties in networks of processors, in: *Proceedings of the 12th Symposium on Theory of Computing*, 1980, pp. 82–93.
- [2] C. Berge, *Graphs and Hypergraphs*, North-Holland, Amsterdam, 1985.
- [3] W. Feller, *An Introduction to Probability Theory and Its Application*, vol. I, second ed., Wiley, New York, 1960.
- [4] W. Feller, *An Introduction to Probability Theory and Its Application*, vol. II, second ed., Wiley, New York, 1966.
- [5] C. Lavault, *Évaluation des algorithmes distribués*, Hermes, 1995.
- [6] G. LeLann, Distributed systems: towards a formal approach, in: B. Gilchrist (Ed.), *Information processing’77*, North-Holland, Amsterdam, 1977, pp. 155–160.
- [7] Y. Métivier, N. Saheb, Probabilistic analysis of an election algorithm in a tree, in: *Colloquium on Trees in Algebra and Programming*, vol. 787 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, 1994, pp. 234–245.

- [8] Y. Métivier, N. Saheb, A. Zemhari, A uniform randomized election in trees, in: *International Colloquium on Structural Information and Communication Complexity*, 2003, pp. 261–274.
- [9] D. Peleg, *Distributed Computing—A Locality-Sensitive Approach*, SIAM, 2000.
- [10] M. Raynal, *Networks and Distributed Computation: Concepts, Tools, and Algorithms*, MIT Press, Cambridge, MA, 1988.
- [11] K.H. Rosen (E.d.), *Handbook of discrete and combinatorial mathematics*, CRC Press, Boca Raton, 2000.
- [12] G. Tel, *Introduction to distributed algorithms*, Cambridge University Press, Cambridge, 2000.