LOG LOG SELECTION WITH HIGH PROBABILITY

Ny Aina ANDRIAMBOLAMALALA
ny-aina.andriambolamalala@irif.fr

March 16^{th} 2018

Institut de Recherche en Informatique Fondamentale
Université de Paris Diderot

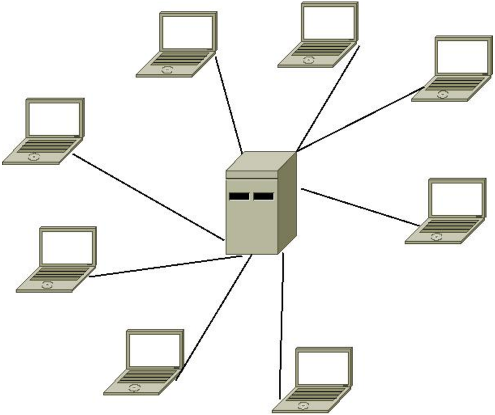
Outline of the talk

1. Distributed Computing
2. Leader election problems in Beeping networks (Radio networks)
3. Related Works
4. Contributions
5. Open problems

Distributed Computing

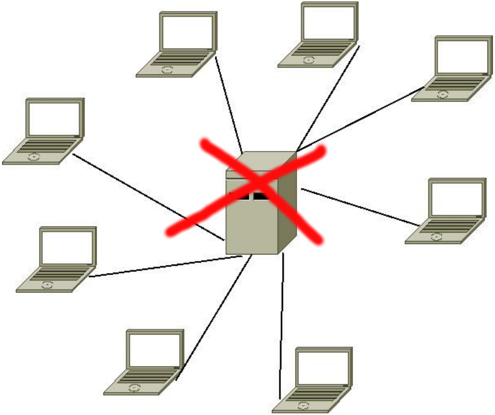
Distributed Computing

- Distribution of tasks by multiple computer working in parallel



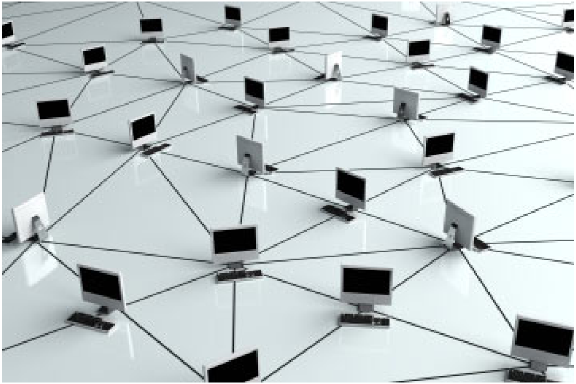
Distributed Computing

- Distribution of tasks by multiple computer working in parallel
- Without central controller



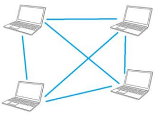
Distributed Computing

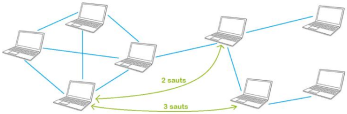
- Distribution of tasks by multiple computer working in parallel
- Without central controller
- Messages sending



Network settings

- Network settings:
  - Single hop (complete graph) and Multi hop networks





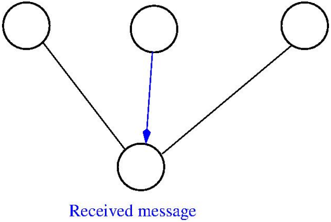
  - Known and Unknown topology networks
    - network size n
    - maximal degree Δ
    - diameter D

Communication in radio networks and beeping model

- Radio networks
  - message of length $\log(n)$, n the number of nodes in the graph

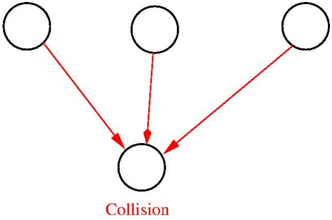
Communication in radio networks and beeping model

- Radio networks
  - message of length log(n), n the number of nodes in the graph
  - Message received if and only if one neighbour sent it overwise SILENCE or COLLISION

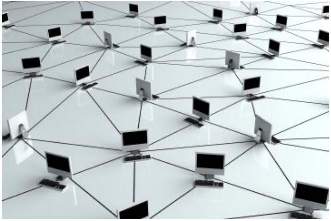


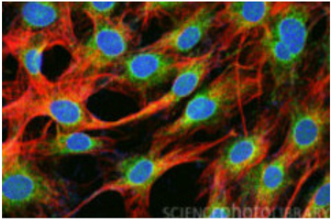
Communication in radio networks and beeping model

- Radio networks
  - message of length log(n), n the number of nodes in the graph
  - Message received if and only if one neighbour sent it overwise SILENCE or COLLISION
  - Collision detection



- Beep Model
  - Biological cellular networks.
  - Synchronized times slots : at any time slot t
    - Send beep (message of length 1)
    - Listen for beep (collision)





Leader election problems in Beeping networks (Radio networks)

Design and Analysis of Communication Algorithms

- Design = correctedness, simplicity of conception

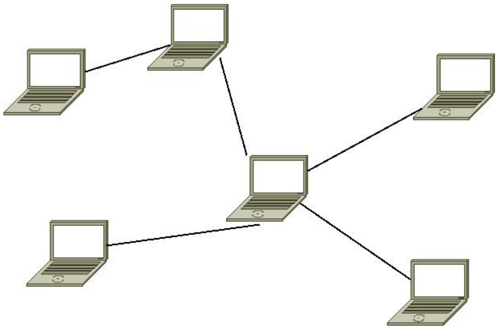
- Analysis = quantifying the running-time and space required of an algorithm : O(x) : number of messages sent during the execution of the algorithm

- Optimization = reaching the lower bound of complexity

- Termination with hight probability = Randomized algorithms must terminates with probability $\geq 1 - O(\frac{1}{n^c})$ where $c > 0$

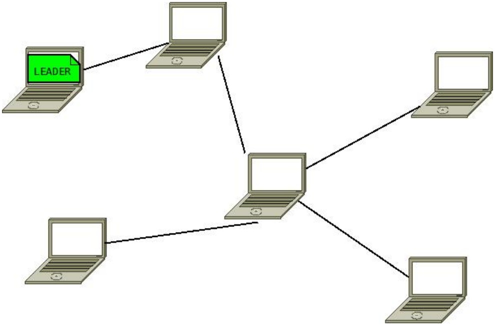
Leader election

- electing a central controller dynamically.



Leader election

- electing a central controller dynamically.



Example of application

Example of application

Example of application

Example of application

Constraints

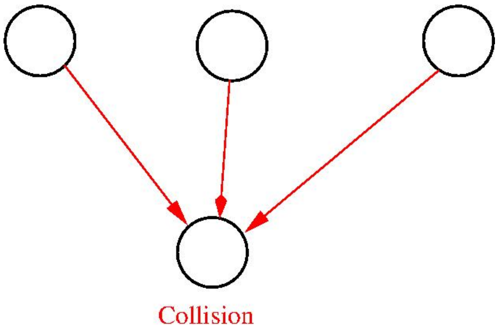
- nodes don't know how many neighbors they have

Constraints

- nodes don't know how many neighbors they have
- nodes don't have information about the topology of the network

Constraints

- nodes don't know how many neighbors they have
- nodes don't have information about the topology of the network
- when collision occurs, nodes don't receive the message

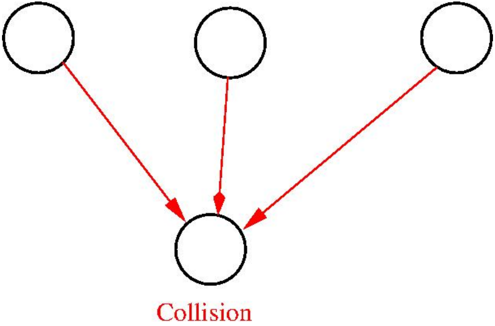


Constraints

- nodes don't know how many neighbors they have
- nodes don't have information about the topology of the network
- when collision occurs, nodes don't receive the message
- nodes can only send message of length $O(\log n)$ or 1 (beep model)

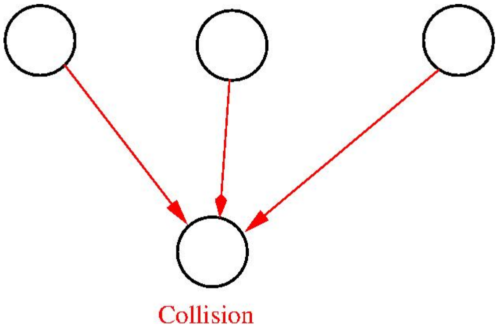
Constraints

- in beep model, if a node hears beep at time t, how to know if there was one or many nodes beeping



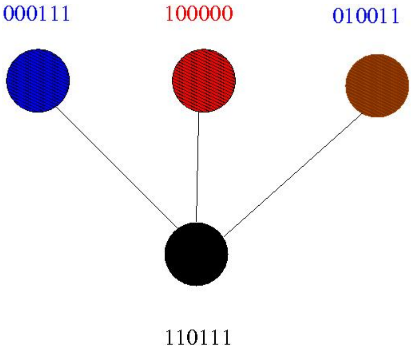
Constraints

- in beep model, if a node hears beep at time t, how to know if there was one or many nodes beeping
- how does a node know if it's the only one beeping at time t



Constraints

- in beep model, if a node hears beep at time t, how to know if there was one or many nodes beeping
- how does a node know if it's the only one beeping at time t
- if nodes send a message of length $O(\log n)$ bit by bit, how to know if one node received the original message



Related Works

Beep model and Radio Networks

Authors	Problem	Single-hop Network Model	Complexity (w.h.p.)	References
Willard	randomized Leader Election	RN with collision detection	$O(\log n)$	SIAM J. of Comp. (1986)
Kushilevitz-Mansour	randomized Leader Election	RN No collision detection	$O(\log^2 n)$	SIAM J. of Comp. (1998)
Nakano · Olariu	randomized Leader Election	Beep with collision detection	$\mathrm{O}(\log n)$	IEEE TPDS (2002)
Ghaffari · Lynch · Sastry	randomized Leader Election	RN with collision detection	$\mathrm{O}(\log n)$	Dist. Comp. (2012)

Existing randomized algorithms principle

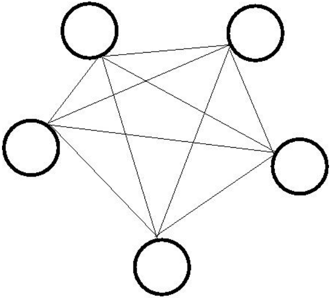
- Time computation subdivided into synchronized time slots t

Existing randomized algorithms principle

- Time computation subdivided into synchronized time slots t
- At each time slot t, all nodes have to beep with a given probability p_t

Existing randomized algorithms principle

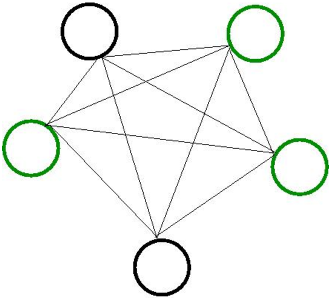
- Time computation subdivided into synchronized time slots t
- At each time slot t, all nodes have to beep with a given probability p_t
- p_t increases or decreases depending on what happens in $t-1$



p ++

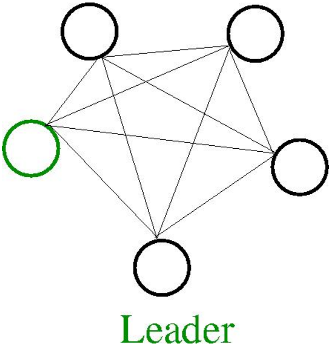
Existing randomized algorithms principle

- Time computation subdivided into synchronized time slots t
- At each time slot t, all nodes have to beep with a given probability p_t
- p_t increases or decreases depending on what happens in $t - 1$



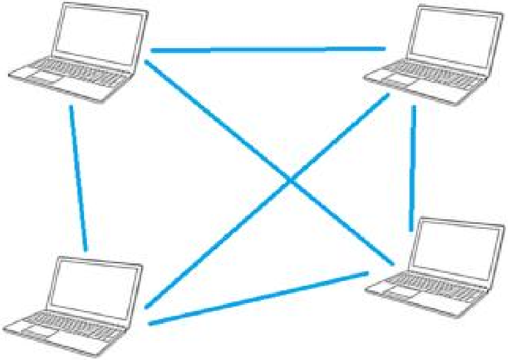
Existing randomized algorithms principle

- Time computation subdivided into synchronized time slots t
- At each time slot t, all nodes have to beep with a given probability p_t
- p_t increases or decreases depending on what happens in $t-1$



Contributions

LEADER ELECTION SINGLE HOP BEEPING NETWORK



- each node i takes a random number $x_i \in [0, n]$ using a probability
  $p_k = \mathbb{P}[x_i = k] = \mathrm{e}^{-\sqrt{k}} - \mathrm{e}^{-\sqrt{k+1}}$

COMBINATORIC PART

- each node i takes a random number $x_i \in [0, n]$ using a probability $p_k = \mathbb{P}[x_i = k] = \mathrm{e}^{-\sqrt{k}} - \mathrm{e}^{-\sqrt{k+1}}$
  - $\forall i, x_i < 4(\log n)^2$ with probability at least $1 - O(\frac{1}{n^c})$

- each node i takes a random number $x_i \in [0, n]$ using a probability
  $p_k = \mathbb{P}[x_i = k] = e^{-\sqrt{k}} - e^{-\sqrt{k+1}}$
  - $\forall i, x_i < 4(\log n)^2$ with probability at least $1 - O(\frac{1}{n^c})$
  - $\#max = 1$ with probability at least $1 - O(\frac{1}{n^c})$

CORRECTION

- each node i takes a random number $x_i \in [0, n]$ using a probability p_k

$$\begin{cases} (e^{-\sqrt{k}} - e^{-\sqrt{k+1}}) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k < (\frac{1+\epsilon}{2} \log n)^2 \\ (e^{-\sqrt{k}} - e^{-\sqrt{k+1}} + \int_{k+1}^{k+2} \frac{e^{-\sqrt{t}}}{t} dt) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k \geq (\frac{1+\epsilon}{2} \log n)^2 \end{cases}$$

CORRECTION

- each node i takes a random number $x_i \in [0, n]$ using a probability p_k

$$\begin{cases} (e^{-\sqrt{k}} - e^{-\sqrt{k+1}}) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k < (\frac{1+\epsilon}{2}\log n)^2 \\ (e^{-\sqrt{k}} - e^{-\sqrt{k+1}} + \int_{k+1}^{k+2} \frac{e^{-\sqrt{t}}}{t} dt) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k \geq (\frac{1+\epsilon}{2}\log n)^2 \end{cases}$$

  - $\forall i, x_i < 4(\log n)^2$ with probability at least $1 - O(\frac{1}{n^c})$

- each node i takes a random number $x_i \in [0, n]$ using a probability p_k

$$\begin{cases} (e^{-\sqrt{k}} - e^{-\sqrt{k+1}}) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k < (\frac{1+\epsilon}{2}\log n)^2 \\ (e^{-\sqrt{k}} - e^{-\sqrt{k+1}} + \int_{k+1}^{k+2} \frac{e^{-\sqrt{t}}}{t} dt) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k \geq (\frac{1+\epsilon}{2}\log n)^2 \end{cases}$$

  - $\forall i, x_i < 4(\log n)^2$ with probability at least $1 - O(\frac{1}{n^c})$
  - $\#max = 1$ with probability at least $1 - O(\frac{1}{n^c})$

Algorithm SINGLE HOP LEADER ELECTION(S, x_s)

$\forall$ nodes $s \in S$ at any time slot t:
  set $SUP \leftarrow 4(\log n)^2$, $INF \leftarrow 0$, $last_{INF} \leftarrow INF$, $status \leftarrow NULL$
  take a random number $x_i \in [0, n]$ using a probability $p_k = \mathbb{P}[x_i = k]$
  while s ***status*** $\notin \{LEADER, NON\_LEADER\}$ **do**
    if $X_s \in]INF, SUP]$ **then**
      BEEP, $last_{INF} \leftarrow INF$, $INF \leftarrow \frac{INF+SUP}{2}$
    else
      LISTEN
    end
    if *hear BEEP* **then**
      $status \leftarrow NON\_LEADER$
    else
      $INF \leftarrow last_{INF}$, $SUP \leftarrow INF$
    end
    if $|]INF, SUP]| = 1$ **then**
      if s *BEEPS* **then**
        $status \leftarrow LEADER$
      else
        $status \leftarrow NON\_LEADER$
      end
    end
  end
  s having $status = LEADER$ send X_s bit by bit

COMPLEXITY

- Complexity of $O(\log \log n)$ with high probability

COMPLEXITY

- Complexity of $O(\log \log n)$ with high probability
  - dichotomy in $\log(4(\log n)^2 = O(\log \log n)$

COMPLEXITY

- Complexity of $O(\log \log n)$ with high probability
  - dichotomy in $\log(4(\log n)^2 = O(\log \log n)$
  - sending x_{max_i} in $\log(4(\log n)^2 = O(\log \log n)$

Improvement

- Instead of using p_k

$$\begin{cases} (e^{-\sqrt{k}} - e^{-\sqrt{k+1}}) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k < (\frac{1+\epsilon}{2}\log n)^2 \\ (e^{-\sqrt{k}} - e^{-\sqrt{k+1}} + \int_{k+1}^{k+2} \frac{e^{-\sqrt{t}}}{t} dt) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k \geq (\frac{1+\epsilon}{2}\log n)^2 \end{cases}$$

Improvement

- Instead of using p_k

$$\begin{cases} (e^{-\sqrt{k}} - e^{-\sqrt{k+1}}) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k < (\frac{1+\epsilon}{2}\log n)^2 \\ (e^{-\sqrt{k}} - e^{-\sqrt{k+1}} + \int_{k+1}^{k+2} \frac{e^{-\sqrt{t}}}{t} dt) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k \geq (\frac{1+\epsilon}{2}\log n)^2 \end{cases}$$

- More general form p_k

$$\begin{cases} (e^{-k^{\frac{1}{a}}} - e^{-k+1^{\frac{1}{a}}}) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k < (\frac{1+\epsilon}{2}\log n)^a \\ (e^{-k^{\frac{1}{a}}} - e^{-k+1^{\frac{1}{a}}} + \int_{k+1}^{k+2} \frac{e^{-t^{\frac{1}{a}}}}{t} dt) \times \frac{1}{1+n^{-\frac{(1+\epsilon)}{2}}} & k \geq (\frac{1+\epsilon}{2}\log n)^a \end{cases}$$

Where $a = 1 + \lambda, 0 < \lambda \leq 1$

Improvement

- Using this new p_k
  - $\forall i, x_i \leq (2 \log n)^a$ w.h.p
  - *max* is unique w.h.p
- We can use these properties to design an algorithm electing a leader in $O(\log n)^{1+\lambda}, \lambda \in]0,1]$, on Radio networks without collision detection $w.h.p$

Open problems

- We use a discrete probability distribution where $\forall i, x_i \leq 4(\log n)^2$ w.h.p

- We use a discrete probability distribution where $\forall i, x_i \leq 4(\log n)^2$ w.h.p
- Is there any discrete probability distribution such that The maximum of n copies is unique (w.h.p) and of order $O((\log \log n)^c)$?

- We use a discrete probability distribution where $\forall i, x_i \leq 4(\log n)^2$ w.h.p
- Is there any discrete probability distribution such that The maximum of n copies is unique (w.h.p) and of order $O((\log\log n)^c)$?
- $\Omega(?)$

- We use a discrete probability distribution where $\forall i, x_i \leq 4(\log n)^2$ w.h.p
- Is there any discrete probability distribution such that The maximum of n copies is unique (w.h.p) and of order $O((\log \log n)^c)$?
- $\Omega(?)$
- How about energy efficiency

- We use a discrete probability distribution where $\forall i, x_i \leq 4(\log n)^2$ w.h.p
- Is there any discrete probability distribution such that The maximum of n copies is unique (w.h.p) and of order $O((\log \log n)^c)$?
- $\Omega(?)$
- How about energy efficiency
- Find any way to simulate this algorithm on Multi-hop Radio Networks

- We use a discrete probability distribution where $\forall i, x_i \leq 4(\log n)^2$ w.h.p
- Is there any discrete probability distribution such that The maximum of n copies is unique (w.h.p) and of order $O((\log\log n)^c)$?
- $\Omega(?)$
- How about energy efficiency
- Find any way to simulate this algorithm on Multi-hop Radio Networks
- Is there any discrete probability distribution such that The maximum of n copies is in $[\log n - c, \log n + c]$ (w.h.p)?

Thanks