Presentation and Analysis of Some Distributed Randomised Algorithms Habilitation à diriger des recherches

Akka Zemmari

LaBRI - Université Bordeaux 1

October 19, 2009

Research Field

- Design and Analysis of (Randomised) Distributed Algorithms
 - Graph Decomposition
 - Computing by Mobile Agents
 - Computing in Sensor Networks
 - Election Problem
 - Bit Complexity
 - Maximal Independent Set (MIS) Problem
 - Colouring Problem
 - Handshake and Matching Problems

A > < > > < >

Research Field

• Design and Analysis of (Randomised) Distributed Algorithms

- Graph Decomposition
- Computing by Mobile Agents
- Computing in Sensor Networks
- Election Problem
- Bit Complexity
 - Maximal Independent Set (MIS) Problem
 - Colouring Problem
- Handshake and Matching Problems

・ 同 ト ・ ヨ ト ・ ヨ

Outline



2 Bit Complexity of the MIS Problem

3 Colouring Problem

4 Conclusion and Perspectives

→ 3 → 4 3

Introduction



2

Introduction



2

Introduction



2

Introduction



2

Introduction



2

Election in Trees



2

Election in Trees



Akka Zemmari Habilitation à diriger des recherches

2

Election in Trees



2

Election in Trees



2

э

Election in Trees



2

э

Election in Trees



э

Election in Trees

• A tree T is given ;

- while T is not reduced to a leaf do remove a leaf from T
- The *elected vertex* is the surviving one.

Goal of study

- To determine the chances of being elected for each vertex.
- To guide the "ballot" toward a given probability distribution by imposing "locally computable" control structures.

▲□▶ ▲ □▶ ▲ □

Election in Trees

- A tree T is given ;
- while T is not reduced to a leaf do remove a leaf from T
- The *elected vertex* is the surviving one.
- Goal of study
 - To determine the chances of being elected for each vertex.
 - To guide the "ballot" toward a given probability distribution by imposing "locally computable" control structures.

(日) (同) (日) (日)

Election in Trees

- A tree T is given ;
- while T is not reduced to a leaf do remove a leaf from T
- The *elected vertex* is the surviving one.
- Goal of study
 - To determine the chances of being elected for each vertex.
 To guide the "ballot" toward a given probability distribution by imposing "locally computable" control structures.

・ロト ・同ト ・ヨト ・ヨト

Election in Trees

- A tree *T* is given ;
- while T is not reduced to a leaf do remove a leaf from T
- The *elected vertex* is the surviving one.

Goal of study

- To determine the chances of being elected for each vertex.
- To guide the "ballot" toward a given probability distribution by imposing "locally computable" control structures.

Election in Trees

- A tree *T* is given ;
- while T is not reduced to a leaf do remove a leaf from T
- The *elected vertex* is the surviving one.

Goal of study

- To determine the chances of being elected for each vertex.
- To guide the "ballot" toward a given probability distribution by imposing "locally computable" control structures.

Election in Trees

- A tree *T* is given ;
- while T is not reduced to a leaf do remove a leaf from T
- The *elected vertex* is the surviving one.

Goal of study

- To determine the chances of being elected for each vertex.
- To guide the "ballot" toward a given probability distribution by imposing "locally computable" control structures.

Election in Trees

Combinatorial Approach [Y. Métivier and N. Saheb in CAAP'94]

All leaf-removal sequences have the same probability:

$$p_{X}(T) = \frac{C_{T}^{X}}{C_{T}^{\emptyset}}.$$

 C_T^{\times} : the number of leaf-removal sequences yielding to the election of x C_T^{\emptyset} : the total number of leaf-removal sequences.



Election in Trees

Combinatorial Approach [Y. Métivier and N. Saheb in CAAP'94]

All leaf-removal sequences have the same probability:

$$\mathcal{D}_{X}(T) = \frac{C_{T}^{X}}{C_{T}^{\emptyset}}.$$

 C_T^{\times} : the number of leaf-removal sequences yielding to the election of x C_T^{\emptyset} : the total number of leaf-removal sequences.



Election in Trees

Combinatorial Approach

Theorem.

In a tree, the median vertex(ices) has(have) the highest probability of being elected.

・ 同 ト ・ ヨ ト ・ ヨ ト

Election in Trees

Locally Uniform Approach

At each step all leaves have the same probability of being removed.



Election in Trees

Locally Uniform Approach

At each step all leaves have the same probability of being removed.



A > < > > < >

Election in Trees

Locally Uniform Approach

At each step all leaves have the same probability of being removed.

▲□▶ ▲ □▶ ▲ □

Election in Trees

Locally Guided Election

Each leaf v is active: it has a lifetime L(v) which is exponentially distributed

$$\mathbb{P}r(L(v) \leq t) = 1 - e^{-\lambda(v)t}, \quad \forall t \in \mathbb{R}^+.$$

probability of its death in the interval [t, t + h) is $\lambda(v)h + o(h)$, as $h \rightarrow 0$.

 $\lambda(v)$: death rate of v. It must be determined at the instant where v has become a leaf by the local information received from the removed leaves on its side.

Election in Trees

Question:

- How to choose the λ -assignment to guide the "ballot" towards a given probability distribution over the set of vertices ?
- In particular, is it possible to guide it towards a uniform election ?

▲□ ► < □ ► </p>

Election in Trees

Question:

- How to choose the λ -assignment to guide the "ballot" towards a given probability distribution over the set of vertices ?
- In particular, is it possible to guide it towards a uniform election ?

・ 同 ト ・ ヨ ト ・ ヨ

Election in Trees

An application: Uniform Model

- Initially all vertices ν have the same weight ω(ν) = 1. (Anonymity)
- Once a leaf has been vanished, its father recuperates its weight.
- The death rate for a leaf v is its weight: $\lambda(v) = \omega(v)$.

- < 同 > < 三 > < 三 >

Election in Trees

Theorem.

In the last model of election, all vertices of the tree have the same probability $\frac{1}{n}$ of being elected.

▲ 同 → - ▲ 三

Election in Trees

Election Duration

Let T a tree of size n, and D_n the uniform election duration on T.

Lemma.

The density function of the random variable D_n is

$$f(t) = n(n-1)e^{-2t} (1-e^{-t})^{n-2}$$

Theorem.

The expected value of D_n is $\sum_{i=2}^n \frac{1}{i} = H_n - 1$, where H_n is the *n*th harmonic number.

(日) (同) (日) (日)

Election in Trees

Election Duration

Let T a tree of size n, and D_n the uniform election duration on T.

_emma.

The density function of the random variable D_n is

$$f(t) = n(n-1)e^{-2t} (1-e^{-t})^{n-2t}$$

Theorem.

The expected value of D_n is $\sum_{i=2}^n \frac{1}{i} = H_n - 1$, where H_n is the *n*th harmonic number.

Election in Trees

Generalisation

- To other classes of graphs
- Each vertex v has a weight $\omega(v) \neq 1$.

Image: A (1) → A (

3

Election in Trees

Other classes of graphs

- k-trees
- Polyominoids

(日) (同) (三) (三)

-2
Election in Trees

To more general classes ?

- How to distinguish vertices which can start generating a life duration ?
- When a vertex vanishes, which vertex will collect the transmitted weight ?

・ 同 ト ・ 三 ト ・ 三

Election in Trees

Non uniform weights

- $\omega(v)$ is an integer
- $\omega(\mathbf{v}) \in \mathbb{R}^+_*$.

 \Rightarrow The probability that a vertex v is elected is proportional to its initial weight.

Election in Trees

Non uniform weights

- $\omega(v)$ is an integer
- $\omega(\mathbf{v}) \in \mathbb{R}^+_*$.

 \Rightarrow The probability that a vertex v is elected is proportional to its initial weight.

A (1) > (1) > (1)

Outline



2 Bit Complexity of the MIS Problem

3 Colouring Problem

4 Conclusion and Perspectives

• = • < =</p>

____>

The MIS Problem

Definition.

Let G = (V, E) be a graph. An *independent set* is a subset U of V s.t.,

$$\forall u,v \in U, \ \{u,v\} \notin E,$$

U is *maximal* if $\forall v \in V \setminus U$, $U \cup \{v\}$ is not an independent set.

(日) (同) (三) (三)

The MIS Problem

Definition.

Let G = (V, E) be a graph. An *independent set* is a subset U of V s.t.,

 $\forall u, v \in U, \ \{u, v\} \notin E,$

U is *maximal* if $\forall v \in V \setminus U$, $U \cup \{v\}$ is not an independent set.



A⊒ ▶ ∢ ∃

- The graph is supposed anonymous
- We propose a *distributed* and *randomised* algorithm.
- We study:
 - its average complexity,
 - its bit complexity,

◆ 同 → ◆ 三

A General Schema



2

A General Schema





イロン イロン イヨン イヨン

A General Schema







<ロ> <同> <同> < 回> < 回>

A General Schema





2

A General Schema



2

Related Works and Comparisons

	Knowledge	Time (on average)	Message size (number of bits)	Bit complexity (per channel)
Luby (Lynch)	Size of the graph	$O(\log n)$	log n	$O(\log^2 n)$

2

Related Works and Comparisons

	Knowledge	Time	Message size	Bit complexity
		(on average)	(number of bits)	(per channel)
Luby (Lynch)	Size of the graph	$O(\log n)$	log n	$O(\log^2 n)$
Luby (Peleg)	Maximum degree in 2-Neighbourhood	$O(\log^2 n)$	log n	$O(\log^3 n)$

2

Related Works and Comparisons

	Knowledge	Time	Message size	Bit complexity
		(on average)	(number of bits)	(per channel)
Luby (Lynch)	Size of the graph	$O(\log n)$	log n	$O(\log^2 n)$
Luby (Peleg)	Maximum degree in 2-Neighbourhood	$O(\log^2 n)$	log n	$O(\log^3 n)$
Luby (Wattenhofer)	Maximum of neighbours degrees	$O(\log n)$	log n	$O(\log^2 n)$

2

Related Works and Comparisons

	Knowledge	Time	Message size	Bit complexity
		(on average)	(number of bits)	(per channel)
Luby (Lynch)	Size of the graph	$O(\log n)$	log n	$O(\log^2 n)$
Luby (Peleg)	Maximum degree in 2-Neighbourhood	$O(\log^2 n)$	log n	$O(\log^3 n)$
Luby (Wattenhofer)	Maximum of neighbours degrees	$O(\log n)$	log n	$O(\log^2 n)$
Algorithm C	no knowledge	$O(\log n)$	1	$O(\log n)$

2

Algorithm \mathcal{A} : A Real Based Algorithm

- A phase of the algorithm:
 - each vertex u, still in the graph, generates a random variable x(u);
 - u sends x(u) to all its neighbours still in the graph;
 - u receives x(v) from each neighbour v, still in the graph;

u is included in the independent set if its x is a local minimum, i.e., x(u) < x(v) for each neighbour v of u

Analysis of Algorithm \mathcal{A}

Lemma.

In any phase, the expected number of edges removed from the remaining graph G is at least half the number of edges in G.

Corollary.

There are constants k_1 and K_1 such that for any graph G = (V, E) of *n* vertices the number of phases to remove all edges from *G* is:

-] less than k_1 log n on average,
-) less than $K_1 \log n$ with probability $1 o(n^{-1})$.

Analysis of Algorithm \mathcal{A}

_emma.

In any phase, the expected number of edges removed from the remaining graph G is at least half the number of edges in G.

Corollary.

There are constants k_1 and K_1 such that for any graph G = (V, E) of *n* vertices the number of phases to remove all edges from *G* is:

- less than $k_1 \log n$ on average,
- 2 less than $K_1 \log n$ with probability $1 o(n^{-1})$.

▲ □ ▶ ▲ □ ▶ ▲

Algorithm \mathcal{B} : A Bit Based Algorithm

Main idea: simulates exchanges of real numbers by exchanges of bits which define real numbers, most significant first.

One phase of exchange of real numbers is replaced by a phase composed of a sequence of rounds

🗇 🕨 🔺 🖻 🕨 🔺 🖻

Algorithm \mathcal{B} : A Bit Based Algorithm

In each round:

- *u* generates uniformly one bit *b*(*u*) ∈ {0,1};
- u sends b(u) to all its active neighbours;
- u receives b(v) from each active neighbour v,
- *u* makes a decision:
 - u is IN MIS
 - u is NOT IN MIS
 - *u* is *INELIGIBLE*: when a vertex *u* sends this message it means that until the end of the current phase *u* cannot be in the MIS.

Analysis of Algorithm $\mathcal B$

Theorem.

Algorithm \mathcal{B} constructs an MIS for any arbitrary graph of size $n \ge 1$ in $O(\log^2 n)$ exchanges of bits on average and with high probability.

・ロト ・同ト ・ヨト ・ヨト

Algorithm C: An Optimal Bit Complexity Algorithm

Main Idea: desynchronisation of phases between edges in algorithm $\ensuremath{\mathcal{B}}$

If, in a round, u breaks the symmetry with v_1 and does not break the symmetry with v_2 then u considers that a phase with v_1 is completed and starts, in anticipation, a new phase with v_1 and it continues the previous phase with v_2 .

Algorithm C: An Optimal Bit Complexity Algorithm

Main Idea: desynchronisation of phases between edges in algorithm $\ensuremath{\mathcal{B}}$

If, in a round, u breaks the symmetry with v_1 and does not break the symmetry with v_2 then u considers that a phase with v_1 is completed and starts, in anticipation, a new phase with v_1 and it continues the previous phase with v_2 .

A > < > > < >

Analysis of Algorithm C

Theorem.

The randomised distributed MIS Algorithm C for arbitrary graphs of size n halts in time $O(\log n)$ with probability $1 - o(n^{-1})$, each message containing 1 bit.

Corollary.

The bit complexity per channel of Algorithm C is $O(\log n)$.

(日) (同) (日) (日)

Analysis of Algorithm ${\cal C}$

Theorem

The randomised distributed MIS Algorithm C for arbitrary graphs of size n halts in time $O(\log n)$ with probability $1 - o(n^{-1})$, each message containing 1 bit.

Corollary.

The bit complexity per channel of Algorithm C is $O(\log n)$.

A (1) > A (1) > A

Outline



2 Bit Complexity of the MIS Problem

3 Colouring Problem

4 Conclusion and Perspectives

~ 프 > ~ 프

< 67 ▶

The Problem

Definition.

Let G = (V, E) be a simple connected undirected graph. A proper vertex colouring for G is an assignment of a colour c(v) to each vertex v, such that any two adjacent vertices have a different colour, i.e., $c(v) \neq c(u)$ for every $\{u, v\} \in E$.

- 4 同 6 4 日 6 4 日 6

The Problem

Definition.

Let G = (V, E) be a simple connected undirected graph. A proper vertex colouring for G is an assignment of a colour c(v) to each vertex v, such that any two adjacent vertices have a different colour, i.e., $c(v) \neq c(u)$ for every $\{u, v\} \in E$.



< (17) > < (2)

Related Works



Related Works



Related Works



€ {0,1,<mark>2</mark>}

Related Works

3



Related Works

Johansson's Algorithm



Akka Zemmari Habilitation à diriger des recherches

Related Works

Johansson's Algorithm



Akka Zemmari Habilitation à diriger des recherches

Related Works

Johansson's Algorithm



Akka Zemmari Habilitation à diriger des recherches
Related Works

Theorem.

Johansson's Algorithm runs in $O(\log n)$ rounds on average and w.h.p.

Remark.

Messages are of size $O(\log n)$, and hence, its average bit complexity is $O(\log^2 n)$.

(日) (同) (三) (三)

Related Works

Theorem

Johansson's Algorithm runs in $O(\log n)$ rounds on average and w.h.p.

Remark.

Messages are of size $O(\log n)$, and hence, its average bit complexity is $O(\log^2 n)$.

- < 同 > < 三 > < 三 >

Related Works

Theorem. [Kothapalli et al.]

If only one bit can be sent along each edge in a round, then every distributed vertex colouring algorithm (in which every node has the same initial state and initially only knows its own edges) needs at least $\Omega(\log n)$ rounds w.h.p.

Algorithm *Fast_Colour*



< ロ > < 回 > < 回 > < 回 > < 回 > .

Algorithm Fast_Colour



イロン イロン イヨン イヨン

Algorithm *Fast_Colour*



2

Algorithm Fast_Colour





2

< 同 > < 三 > < 三

Algorithm Fast_Colour



æ

- (E

Algorithm Fast_Colour



▲ 同 ▶ → ● 目

Algorithm Fast_Colour



æ

3

▲ 同 → - ▲ 三

Analysis of the Algorithm

Expected Time Complexity

Lemma.

In any phase of the algorithm, the expected number of edges removed from the residual graph G is half the number of its edges.

Corollary.

There are constants k_1 and K_1 such that for any graph G of n vertices, the number of phases to remove all edges from G is:

- less than k₁ log n on average,
- less than $K_1 \log n$ w.h.p.

(日) (同) (日) (日)

Analysis of the Algorithm

Expected Time Complexity

Lemma.

In any phase of the algorithm, the expected number of edges removed from the residual graph G is half the number of its edges.

Corollary.

There are constants k_1 and K_1 such that for any graph G of n vertices, the number of phases to remove all edges from G is:

- less than k₁ log n on average,
- less than K₁ log n w.h.p.

Analysis of the Algorithm

Theorem.

Algorithm Fast_Colour computes a colouring for any arbitrary graph of size n in time $O(\log n)$ w.h.p., each message containing 1 bit.

- < 同 > < 三 > < 三 >

Analysis of the Algorithm

Local Complexity

Proposition

Let G = (V, E) be a connected graph and $v \in V$ with $d(v) = d \rightarrow \infty$. Let L_d denote the number of bits generated by v before it obtains its final colour. Then

$$\mathbb{E}\left(L_d\right) = \log_2 d + \frac{1}{2} + \frac{\gamma}{\log 2} + Q\left(\log_2 d\right) + O\left(d^{-2}\right),$$

where $Q(u) = -\frac{1}{\log 2} \sum_{k \in \mathbb{Z} \setminus \{0\}} \Gamma\left(\frac{2ik\pi}{\log 2}\right) e^{-2ik\pi u}$ is a Fourier series with period 1 and with an amplitude which does not exceed 10^{-6} .

・ロト ・同ト ・ヨト ・ヨト

Analysis of the Algorithm

Local Complexity

Lemma.

If $d
ightarrow \infty$, then we have

$$\mathbb{V}ar(L_d) = \left(\frac{1}{\log 2} - 1\right)\log_2 d + \frac{1}{12} + \frac{1}{6}\frac{\pi^2}{(\log 2)^2} - P(\log_2 d) + O(d^{-2}),$$

where $P(u) = Q(u)^2 + \left(2u + \frac{2\gamma}{\log 2} - \frac{2}{\log 2}\right)Q(u).$

Proposition

The ratio R_d between L_d and $\log_2 d$ tends in probability to 1 as d tends to ∞ .

(日) (同) (三) (三)

Analysis of the Algorithm

Local Complexity

Lemma.

If
$$d
ightarrow \infty$$
, then we have

$$\mathbb{V}ar(L_d) = \left(\frac{1}{\log 2} - 1\right)\log_2 d + \frac{1}{12} + \frac{1}{6}\frac{\pi^2}{(\log 2)^2} - P(\log_2 d) + O(d^{-2}),$$

where $P(u) = Q(u)^2 + \left(2u + \frac{2\gamma}{\log 2} - \frac{2}{\log 2}\right)Q(u).$

Proposition

The ratio R_d between L_d and $\log_2 d$ tends in probability to 1 as d tends to ∞ .

(日) (同) (三) (

Outline



2 Bit Complexity of the MIS Problem

3 Colouring Problem



→ 3 → < 3</p>

____>

Conclusion and Perspectives

Election Problem

- is there any "totally fair" algorithm for any class of graph ?
- can we design a one-pass distributed algorithm which implements the *p*-distribution ?

• Bit Complexity

- of the 2-MIS problem ?
- of the distance 2 colouring problem ?
- Colouring Problem
 - the main challenge: reduce the number of colours

Conclusion and Perspectives

• Election Problem

- is there any "totally fair" algorithm for any class of graph ?
- can we design a one-pass distributed algorithm which implements the *p*-distribution ?

Bit Complexity

- of the 2-MIS problem ?
- of the distance 2 colouring problem ?
- Colouring Problem
 - the main challenge: reduce the number of colours

Conclusion and Perspectives

- Election Problem
 - is there any "totally fair" algorithm for any class of graph ?
 - can we design a one-pass distributed algorithm which implements the *p*-distribution ?
- Bit Complexity
 - of the 2-MIS problem ?
 - of the distance 2 colouring problem ?
- Colouring Problem
 - the main challenge: reduce the number of colours

Related Publications

Election Problem

- Y. Métivier, N. Saheb and A. Zemmari. Locally Guided Randomized Elections in Trees: the Totally Fair Case. *Information and Computation*. 2005.
- J.-F. Marckert, N. Saheb, A. Zemmari.*Election Algorithms with Random Delays in Trees.* (FPSAC 2009).
- A. El Hibaoui, N. Saheb-Djahromi and A. Zemmari. *Polyominoids and Uniform Election*. (FPSAC 2005).
- Y. Métivier, N. Saheb and A. Zemmari. *A Uniform Randomized Election in Trees (Extended Abstract)*. (SIROCCO 2003).

A (1) > A (1) > A

Related Publications

MIS Problem

- Y. Métivier, J.M. Robson, N. Saheb-Djahromi, A. Zemmari. An Optimal Bit Complexity Randomised Distributed MIS Algorithm. (SIROCCO 2009).
- Y. Métivier, N. Saheb and A. Zemmari. Randomized Local Elections. *Information Processing Letters*. 2002.
- Colouring Problem
 - Y. Metivier, J.- M. Robson, N. Saheb-Djahromi, A. Zemmari. An analysis of an optimal bit complexity randomised distributed vertex colouring algorithm. (OPODIS 2009).

Thank You

Akka Zemmari Habilitation à diriger des recherches

-2