

SDRP & MA

Elections locales probabilistes

A. Zemmari
zemmari@labri.fr
www.labri.fr/visidia/

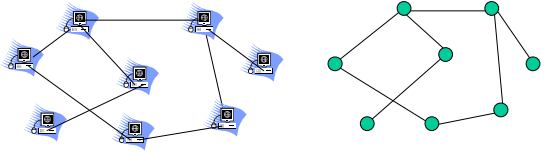
03/12/2004

1

SDRP & MA

Le modèle

- Un réseau asynchrone de processus anonymes;
- Les processus communiquent par échange de messages en mode asynchrone;
- Modélisation : un graphe.



03/12/2004

2

SDRP & MA

Présentation

- **Pourquoi les élections locales :**
 - Brique de base pour les algorithmes distribués codés par les calculs locaux.
- **Pourquoi des élections probabilistes :**
 - Permettent de réaliser l'exclusion mutuelle et d'éviter les conflits (voir chapitre suivant).

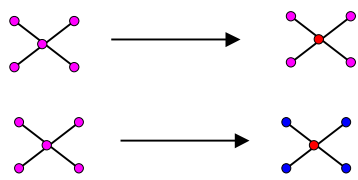
03/12/2004

3

SDRP & MA

Présentation (2)

- Un sommet change son étiquette en fonction des étiquettes de ses voisins :



03/12/2004

4

SDRP & MA

Types de synchronisations locales

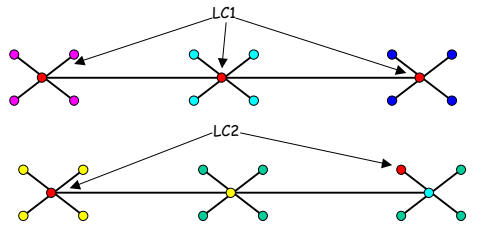
- **RdV**: dans une étape de calcul, les étiquettes de deux sommets voisins changent.
- **LC1**: dans une étape de calcul, seule l'étiquette du sommet centre de la boule change, et ceci en fonction des étiquettes des autres sommets de la boule.
- **LC2**: dans une étape de calcul, toutes les étiquettes des sommets de la boule changent

03/12/2004

5

SDRP & MA


Types de synchronisations locales (2)



03/12/2004

6

SDRP & MA




Pourquoi LC1

- Un algorithme de détection des propriétés stables (d'après Szymanski, Shi et Prywes) :
 - Un ensemble de processus qui réalisent chacun une tâche. Chaque processus est capable de détecter localement sa terminaison.
 - L'algorithme SSP permet de détecter à quel moment tous les processus ont terminé.

03/12/20047

SDRP & MA




Pourquoi LC1 (2)

- A chaque sommet v , on associe :
 - $P(v)$ un prédicat qui vaut vrai si la tâche de v est terminée, $P(v)$ est initialisé à *faux*.
 - $a(v)$ un entier qui code la distance jusqu'à laquelle v sait que les sommets ont terminé, $a(v)$ est initialisé à -1 .
- Dans chaque calcul local, v change la valeur de son entier $a(v)$ en fonction des valeurs $a(w)$, pour tout w voisin de v :
 - Si $P(v) = \text{faux}$, alors $a(v) = -1$;
 - Si $P(v) = \text{vrai}$, alors $a(v) = 1 + \text{Min}\{a(w) \mid w \text{ voisin de } v\}$
- **Théorème** : (Szymanski, Shi et Prywes) L'algorithme décrit ci-dessus permet à tout sommet v de savoir que le prédicat P est vrai pour tous les sommets dans la boule de centre v et de rayon $a(v)$.

03/12/20048

SDRP & MA




Pourquoi LC1 (3)

- L'algorithme SSP est codé par des calculs locaux utilisant des LC1;
- Un sommet ne change que son étiquette et non celles de ses voisins;
- On doit interdire que deux sommets voisins puissent changer leurs étiquettes en même temps (exclusion lors du changement de l'étiquette).

03/12/20049

SDRP & MA




Pourquoi LC2

- Algorithme de Mazurkiewicz pour la numérotation des sommets d'un graphe. (A voir ultérieurement)

03/12/200410

SDRP & MA




Implémentation

- **LC1** :
 chaque sommet v répète indéfiniment les actions suivantes :
 v tire un nombre aléatoire $rand(v)$;
 v envoie $rand(v)$ à tous ses voisins;
 v reçoit tous les nombres envoyés par ses voisins;
 (* v est localement élu dans $B(v,1)$ si $rand(v)$ est strictement plus grand que $rand(w)$, pour tout w voisin de v *)

03/12/200411

SDRP & MA



Implémentation (2)

- **LC2** :
 Chaque sommet v répète indéfiniment les actions suivantes :
 v tire un nombre aléatoire $rand(v)$;
 v envoie $rand(v)$ à tous ses voisins;
 v reçoit les nombres envoyés par ses voisins;
 soit m_w le max des nombres reçus par v de ses voisins différents de w , v envoie m_w à tous ses voisins différents de w ;
 v reçoit les nombres envoyés par tous ses voisins;
 (* v est localement élu dans $B(v,2)$ si $rand(v)$ est strictement plus grand que tous les nombres que reçoit v *)

03/12/200412

SDRP & MA

Questions posées

- Quel est le nombre moyen de synchronisations LC1 (resp. LC2) dans le graphe ?
- Quelle est la probabilité de succès (au même sens que pour le problème du rendez-vous)?
- Quelle est la performance des algorithmes probabilistes utilisés (en comparaison avec un algorithme « idéale ») ?

03/12/200413

SDRP & MA

Probabilité d'élection pour un sommet

- Chaque sommet v tire un nombre aléatoire et uniforme d'un ensemble $\{1, 2, \dots, N\}$.
- Soit $X \subseteq V$ tel que $v \in X$, soit $k = |X|$,

(1)
$$\Pr(\text{rand}(v) > \text{rand}(w), \forall w \in X \setminus \{v\}) = \frac{1}{N} \sum_{i=2}^k \binom{k-1}{i-1}$$

- Et

(2) $\forall v, w \in X, \Pr(\text{rand}(v) = \text{rand}(w)) = 1/N$.

- Preuve : [Exercice](#).

03/12/200414

SDRP & MA

Probabilité d'élection pour un sommet (2)

- **Propositions :**
 - La probabilité pour un sommet v d'être localement élu dans une L1-élection est donnée par :
$$p_1(v) = \frac{1}{d(v)+1}$$
 - La probabilité pour un sommet v d'être localement élu dans une L2-élection est donnée par :
$$p_2(v) = \frac{1}{N_2(v)}$$

où $N_2(v) = |B(v, 2)|$

- **Preuve :** application directe de (1)

03/12/200415

SDRP & MA

Probabilité d'élection pour un sommet (3)

- **Corollaire :** Soit v un sommet de G . Le temps moyen d'attente pour qu'il soit localement élu dans une L1-élection (resp. L2-élection) est $d(v)+1$ (resp. $N_2(v)$).
- **Exemples :**
 - Si G est un anneau de taille n , alors pour tout v $p_1(v) = 1/3$ et $p_2(v) = 1/5$.
 - Si G est un graphe complet de taille n , alors pour tout v $p_1(v) = p_2(v) = 1/n$.

03/12/200416

Probabilité de succès

- Probabilité d'au moins une élection locale dans le graphe = 1
- **Preuve :** Soient v, w deux sommets voisins,

$$\Pr(\text{rand}(v) = \text{rand}(w)) \rightarrow 0 \text{ si } N \rightarrow \infty \text{ (d'après (2))}$$

03/12/200417

SDRP & MA

Espérance du nombre d'élections locales

- **Notations :** $M_1(G)$ (resp. $M_2(G)$) le nombre moyen de sommets localement élus dans une L1-élection (resp. L2-élection).
- **Proposition :**
$$M_1(G) = \sum_{v \in V} \frac{1}{d(v)+1} \text{ et } M_2(G) = \sum_{v \in V} \frac{1}{N_2(v)}$$
- **Exemples :**
 - $M_1(C_n) = n/3$ et $M_2(C_n) = n/5$
 - $M_1(K_n) = M_2(K_n) = 1$

03/12/200418

SDRP & MA

Espérance du nombre d'élections locales (2)

- **Définition** : Soit k un entier positif. On définit la k -densité de $\mathcal{G}=(V,E)$ par :

$$D_k(G) = \frac{\sum_{v \in V} d(v)^k}{|V|}$$
- En particulier :

$$D_1(G) = \frac{\sum_{v \in V} d(v)}{|V|} = 2 \frac{|E|}{|V|}$$

03/12/2004 19

SDRP & MA

Espérance du nombre d'élections locales (3)

- **Lemme** : Soit $\mathcal{G}=(V,E)$ un graphe connexe.

$$\sum_{v \in V} N_2(v) \leq \sum_{v \in V} d(v)^2 + |V|$$
 - En plus, si \mathcal{G} est un arbre, alors l'inégalité devient une égalité.
- **Preuve** : voir le tableau.

03/12/2004 20

SDRP & MA

Espérance du nombre d'élections locales (4)

- **Théorème** : $\mathcal{G}=(V,E)$ un graphe connexe avec $|V|=n$ et $|E|=m$.

$$(1) M_1(G) \geq \frac{n}{D_1(G)+1} = \frac{n^2}{2m+n} \quad \text{Et} \quad (2) M_2(G) \geq \frac{n}{D_2(G)+1}$$
- **Preuves** : (1) au tableau, (2) [Exercice](#).
- **Corollaire** : \mathcal{G} un graphe de degré maximum d .

$$M_1(G) \geq \frac{n}{d+1} \quad \text{et} \quad M_2(G) \geq \frac{n}{d^2+2d+1}$$
- **Corollaire** : T un arbre de taille n .

$$M_1(T) \geq \frac{n^2}{3n-2} > \frac{n}{3}$$

03/12/2004 21

SDRP & MA

Analyse de performance

- **Définitions** :
 - $\alpha(\mathcal{G})$ le cardinal du plus grand ensemble de sommets indépendants contenu dans \mathcal{G} .
 - $\beta(\mathcal{G})$ le cardinal du plus grand ensemble de sommets à distance au moins 3 les uns des autres.
 - Soit A un algorithme probabiliste quelconque permettant de réaliser des L1-élections. On définit l'efficacité de A par

$$\Delta_A(G) = \frac{M_1(G)}{\alpha(G)}$$
 - Soit B un algorithme probabiliste quelconque permettant de réaliser des L2-élections. On définit l'efficacité de B par

$$\Delta_B(G) = \frac{M_2(G)}{\beta(G)}$$

03/12/2004 22

SDRP & MA

L1 et L2

- **Théorème** : Si T est un arbre. Alors l'efficacité de l'algorithme de LC1 est strictement supérieure à $1/3$.
- **Preuve** : $M_1(T) \geq n/3$ et $\alpha(T) < n$ donc $\Delta_{LC1}(T) > 1/3$.
- **Théorème** : Si T est un arbre. Alors l'efficacité de l'algorithme de LC2 est strictement supérieure à $1/4$.
- **Preuve** : par induction sur la taille de l'arbre T .
- **Conjecture** : $\Delta_{LC2}(T) > 1/3$

03/12/2004 23

SDRP & MA

L1 et L2 (2)

- **Remarque** : Si \mathcal{G} n'est pas un arbre, alors les deux algorithmes peuvent avoir une efficacité très mauvaise. En effet, il suffit de considérer le graphe $\mathcal{G}_{2n} = (V,E)$ avec :
 - $V = \{u_1, u_2, \dots, u_n\} \cup \{v_1, v_2, \dots, v_n\}$
 - E est tel que les sommets $\{u_1, u_2, \dots, u_n\}$ forment un graphe complet et $\forall i, j \{u_i, v_j\} \in E$
- **Exercice** : calculer $\Delta_{LC1}(\mathcal{G}_{2n})$ et $\Delta_{LC2}(\mathcal{G}_{2n})$

03/12/2004 24



Efficacité de l'algorithme du Rendez-vous

- **Définition** : Soit $\mathcal{G}=(V,E)$ un graphe et $K(\mathcal{G})$ la taille d'un couplage maximal dans \mathcal{G} . On définit l'efficacité de tout algorithme R permettant de réaliser les rendez-vous par :

$$\Delta_R(\mathcal{G}) = \frac{M_R(\mathcal{G})}{K(\mathcal{G})}$$

- **Exemples** :
 - Si \mathcal{G} est un graphe complet, alors $\Delta_R(\mathcal{G}) = 1/(n-1)$.
 - Si \mathcal{G} est une étoile, alors $\Delta_R(\mathcal{G}) = 1/1 = 1$.

03/12/2004

25



Efficacité de l'algorithme du Rendez-vous (2)

- **Théorème** : L'efficacité de l'algorithme dans le cas des arbres est supérieure à $1/3$.
- **Preuve** : Par induction sur la taille de l'arbre.

03/12/2004

26



Généralisation

- **k-élection** : un sommet est élu si et seulement si il tire le plus grand nombre sur une rayon k.
- **Utilisation** : décomposition de graphes (voir plus loin).

03/12/2004

27