

# Exploration d'arbres par des équipes de robots asynchrones et amnésiques

Paola Flocchini<sup>1</sup>   **David Ilcinkas**<sup>1,2,3</sup>   Andrzej Pelc<sup>2</sup>  
Nicola Santoro<sup>3</sup>

<sup>1</sup>University of Ottawa, Canada

<sup>2</sup>Université du Québec en Outaouais, Canada

<sup>3</sup>Carleton University, Canada

AlgoTel '07, 31 mai 2007

# Problème

## Modèle

- Equipe de robots
  - **identiques, asynchrones et amnésiques**
  - procédant par **cycles Look-Compute-Move**
- Graphes **anonymes**.

## Objectifs

- Chaque nœud doit être **visité** par au moins un robot.
- Tous les robots doivent **s'arrêter**.

# Modèle des robots

## Look-Compute-Move

- **Look** : snapshot (enraciné) du graphe et de ses robots.
- **Compute** : choix d'un voisin (ou du nœud courant).
- **Move** : mouvement instantané vers ce voisin.

## Caractéristiques

- **Asynchrones** : entrelacement arbitraire des actions.
- **Identiques** : indifférenciables et algorithmes identiques.
- **Amnésiques** : Compute uniquement basé sur le Look correspondant.
- **Non-communicants** : aucune communication, même locale, entre les robots.

# Modèle des robots

## Look-Compute-Move

- **Look** : snapshot (enraciné) du graphe et de ses robots.
- **Compute** : choix d'un voisin (ou du nœud courant).
- **Move** : mouvement instantané vers ce voisin.

## Caractéristiques

- **Asynchrones** : entrelacement arbitraire des actions.
- **Identiques** : indifférenciables et algorithmes identiques.
- **Amnésiques** : Compute uniquement basé sur le Look correspondant.
- **Non-communiquants** : aucune communication, même locale, entre les robots.

# Travaux connexes

Modèle précédemment utilisé

dans le plan

- Ando, Oasa, Suzuki, Yamashita, **Distributed memoryless point convergence algorithm for mobile robots with limited visibility**, IEEE Trans. on Rob. and Aut. 1999.
- ...
- Czyzowicz, Gasieniec, Pelc, **Gathering few fat mobile robots in the plane**, OPODIS'2006.

dans les graphes

- Klasing, Markou, Pelc, **Gathering asynchronous oblivious mobile robots in a ring**, ISAAC 2006.

# Travaux connexes

Modèle précédemment utilisé

dans le plan

- Ando, Oasa, Suzuki, Yamashita, **Distributed memoryless point convergence algorithm for mobile robots with limited visibility**, IEEE Trans. on Rob. and Aut. 1999.
- ...
- Czyzowicz, Gasieniec, Pelc, **Gathering few fat mobile robots in the plane**, OPODIS'2006.

dans les graphes

- Klasing, Markou, Pelc, **Gathering asynchronous oblivious mobile robots in a ring**, ISAAC 2006.

# Précisions sur le modèle

## Configurations initiales

Quelconques mais avec **un seul robot par nœud**.  
(Nécessaire pour la terminaison.)

En cas de symétrie

- Compute : choix d'une classe d'équivalence de voisins.
- Choix effectif : fait par l'adversaire (pire cas).

Détection de multiplicité

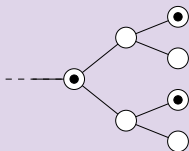
0, 1 ou plusieurs robots

# Précisions sur le modèle

## Configurations initiales

Quelconques mais avec **un seul robot par nœud**.  
(Nécessaire pour la terminaison.)

## En cas de symétrie



- **Compute** : choix d'une **classe d'équivalence de voisins**.
- Choix effectif : fait par l'adversaire (pire cas).

Détection de multiplicité  
0, 1 ou plusieurs robots

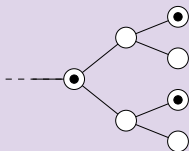


# Précisions sur le modèle

## Configurations initiales

Quelconques mais avec **un seul robot par nœud**.  
(Nécessaire pour la terminaison.)

## En cas de symétrie



- **Compute** : choix d'une **classe d'équivalence de voisins**.
- Choix effectif : fait par l'adversaire (pire cas).

## Détection de multiplicité

0, 1 ou **plusieurs** robots

# Nos résultats

## Résultat principal

Arbres de degré maximum 3 :

- $\Theta(\log n / \log \log n)$  robots

## Justification des restrictions

- $\Theta(n)$  robots dans certains arbres de degré maximum 4 (arbres ternaires complets)
- $\Theta(\log n)$  robots dans les cycles  
(#robots | #nœuds  $\Rightarrow$  exploration impossible)

# Nos résultats

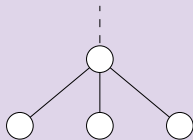
## Résultat principal

Arbres de degré maximum 3 :

- $\Theta(\log n / \log \log n)$  robots

## Justification des restrictions

- $\Theta(n)$  robots dans certains arbres de degré maximum 4 (arbres ternaires complets)



- $\Theta(\log n)$  robots dans les cycles  
(#robots | #nœuds  $\Rightarrow$  exploration impossible)

# Nos résultats

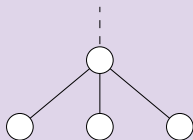
## Résultat principal

Arbres de degré maximum 3 :

- $\Theta(\log n / \log \log n)$  robots

## Justification des restrictions

- $\Theta(n)$  robots dans certains arbres de degré maximum 4 (arbres ternaires complets)



- $\Theta(\log n)$  robots dans les cycles  
( $\#\text{robots} \mid \#\text{nœuds} \Rightarrow$  exploration impossible)

# Borne inférieure : $\Omega(\log n / \log \log n)$ robots

## Idées principales

Arbres binaires complets, cas synchrone

- peu de robots  $\Rightarrow$  peu de snapshots différents
- peu de snapshots différents  $\Rightarrow$  peu de sommets explorés avant de boucler

# Borne supérieure

## Casser la symétrie

$$k \equiv 5 \pmod{6}$$

## Garder des traces de la progression

- Multiplicité pour marquer le début de l'exploration
- Arrangement des robots pour former un cerveau (compteur)

## Pourquoi $\Theta(\log n / \log \log n)$ robots ?

- Chemin de longueur  $L$  et  $O(1)$  robots codent  $\log L$  bits
- Dans un arbre de degré maximum 3, on peut trouver  $\Theta(\log n / \log \log n)$  chemins de longueur  $L = \Theta(\log n)$

# Borne supérieure

Casser la symétrie

$$k \equiv 5 \pmod{6}$$

Garder des traces de la progression

- Multiplicité pour marquer le début de l'exploration
- Arrangement des robots pour former un cerveau (compteur)

Pourquoi  $\Theta(\log n / \log \log n)$  robots ?

- Chemin de longueur  $L$  et  $O(1)$  robots codent  $\log L$  bits
- Dans un arbre de degré maximum 3, on peut trouver  $\Theta(\log n / \log \log n)$  chemins de longueur  $L = \Theta(\log n)$

# Borne supérieure

## Casser la symétrie

$$k \equiv 5 \pmod{6}$$

## Garder des traces de la progression

- Multiplicité pour marquer le début de l'exploration
- Arrangement des robots pour former un cerveau (compteur)

## Pourquoi $\Theta(\log n / \log \log n)$ robots ?

- Chemin de longueur  $L$  et  $O(1)$  robots codent  $\log L$  bits
- Dans un arbre de degré maximum 3, on peut trouver  $\Theta(\log n / \log \log n)$  chemins de longueur  $L = \Theta(\log n)$



# Perspectives

- Graphes arbitraires (de degré maximum 3)
- Impact de la détection de multiplicité
- Visibilité restreinte