

Sur la connaissance a priori nécessaire et suffisante pour résoudre un problème distribué

David ILCINKAS

CNRS & Université de Bordeaux (LaBRI)

Réunion Aladdin, Arcachon

6 mai 2011

Exploration de graphes avec terminaison

Exploration de graphes...

Une entité mobile (robot/agent) doit **visiter** chaque sommet d'un graphe **anonyme**.

avec terminaison

L'entité mobile doit s'arrêter au bout d'un temps fini, i.e. détecter la terminaison.

Anonyme :

- sommets sans identifiant
- étiquetage local des arêtes (numéros de port)

Exploration de graphes avec terminaison

Exploration de graphes...

Une entité mobile (robot/agent) doit **visiter** chaque sommet d'un graphe **anonyme**.

...avec terminaison

L'entité mobile doit s'arrêter au bout d'un temps fini, i.e. **détecter la terminaison**.

Anonyme :

- sommets sans identifiant
- étiquetage local des arêtes (numéros de port)

Exploration de graphes avec terminaison

Exploration de graphes...

Une entité mobile (robot/agent) doit **visiter** chaque sommet d'un graphe **anonyme**.

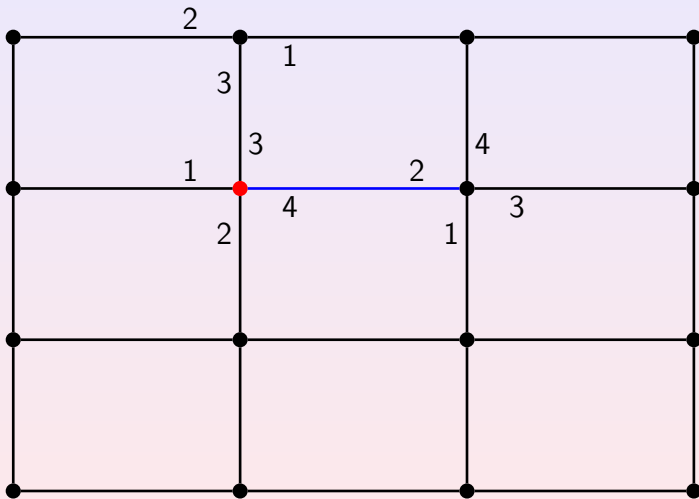
...avec terminaison

L'entité mobile doit s'arrêter au bout d'un temps fini, i.e. **détecter la terminaison**.

Anonyme :

- **sommets sans identifiant**
- étiquetage local des arêtes (numéros de port)

Exemple d'un graphe anonyme



Résultat positif

Théorème

Il existe un agent **capable de résoudre le problème d'exploration avec terminaison** s'il connaît une borne supérieure \hat{n} sur la taille du graphe à explorer.

Preuve

Explorer tous les chemins de longueur \hat{n} partant du sommet de départ puis s'arrêter.

Corollaire

La connaissance d'une borne supérieure sur n est suffisante pour résoudre l'exploration avec terminaison.

Cette connaissance est-elle nécessaire ?

Résultat positif

Théorème

Il existe un agent **capable de résoudre le problème d'exploration avec terminaison** s'il connaît une borne supérieure \hat{n} sur la taille du graphe à explorer.

Preuve

Explorer tous les **chemins de longueur \hat{n}** partant du sommet de départ puis s'arrêter.

Corollaire

La connaissance d'une borne supérieure sur n est suffisante pour résoudre l'exploration avec terminaison.

Cette connaissance est-elle nécessaire ?

Résultat positif

Théorème

Il existe un agent **capable de résoudre le problème d'exploration avec terminaison** s'il connaît une borne supérieure \hat{n} sur la taille du graphe à explorer.

Preuve

Explorer tous les **chemins de longueur \hat{n}** partant du sommet de départ puis s'arrêter.

Corollaire

La **connaissance d'une borne supérieure** sur n est **suffisante** pour résoudre l'exploration avec terminaison.

Cette connaissance est-elle nécessaire ?

Résultat positif

Théorème

Il existe un agent **capable de résoudre le problème d'exploration avec terminaison** s'il connaît une borne supérieure \hat{n} sur la taille du graphe à explorer.

Preuve

Explorer tous les **chemins de longueur \hat{n}** partant du sommet de départ puis s'arrêter.

Corollaire

La **connaissance d'une borne supérieure** sur n est **suffisante** pour résoudre l'exploration avec terminaison.

Cette connaissance est-elle **nécessaire** ?

Résultat négatif

Théorème

La **connaissance d'une borne supérieure** sur n est **nécessaire** pour résoudre l'exploration avec terminaison.

Preuve

- Supposons qu'il existe un agent explorant sans connaissance d'une borne supérieure sur n .
- Agent sur un triangle : arrêt au temps k
- Agent sur cycle C_{2k} : arrêt avant l'exploration

Êtes-vous convaincu ?

Est-ce une preuve satisfaisante ? Non : parité de n .

Résultat négatif

Théorème

La **connaissance d'une borne supérieure** sur n est **nécessaire** pour résoudre l'exploration avec terminaison.

Preuve

- Supposons qu'il existe un agent explorant **sans connaissance d'une borne supérieure** sur n .
- Agent sur un triangle : arrêt au temps k
- Agent sur cycle C_{2k} : arrêt avant l'exploration

Êtes-vous convaincu ?

Est-ce une preuve satisfaisante ? Non : parité de n .

Résultat négatif

Théorème

La **connaissance d'une borne supérieure** sur n est **nécessaire** pour résoudre l'exploration avec terminaison.

Preuve

- Supposons qu'il existe un agent explorant **sans connaissance d'une borne supérieure** sur n .
- Agent sur un triangle : arrêt au temps k
- Agent sur cycle C_{2k} : arrêt avant l'exploration

Êtes-vous **convaincu** ?

Est-ce une preuve satisfaisante? Non : parité de n .

Résultat négatif

Théorème

La **connaissance d'une borne supérieure** sur n est **nécessaire** pour résoudre l'exploration avec terminaison.

Preuve

- Supposons qu'il existe un agent explorant **sans connaissance d'une borne supérieure** sur n .
- Agent sur un triangle : arrêt au temps k
- Agent sur cycle C_{2k} : arrêt avant l'exploration

Êtes-vous **convaincu** ?

Est-ce une **preuve satisfaisante** ? Non : parité de n .

Résultat négatif

Théorème

La **connaissance d'une borne supérieure** sur n est **nécessaire** pour résoudre l'exploration avec terminaison.

Preuve

- Supposons qu'il existe un agent explorant **sans connaissance d'une borne supérieure** sur n .
- Agent sur un triangle : arrêt au temps k
- Agent sur cycle C_{2k} : arrêt avant l'exploration

Êtes-vous **convaincu** ?

Est-ce une **preuve satisfaisante** ? Non : parité de n .

Tentative de formalisation

Information a priori / conseil

- L'agent doit explorer le graphe G .
- Pour l'aider, un oracle donne un conseil au robot sous la forme d'une chaîne binaire $\mathcal{O}(G)$.

\mathcal{O} induit une partition (F_0, F_1, \dots) de la famille de graphes \mathcal{G} :
 $F_i = \{G \in \mathcal{G} \mid \mathcal{O}(G) = i\}$

Connaitre une borne supérieure sur n

$$\forall i \exists \hat{n}_i, \forall G \in F_i \quad n(G) \leq \hat{n}_i$$

Tentative de formalisation

Information a priori / conseil

- L'agent doit explorer le graphe G .
- Pour l'aider, un oracle donne un conseil au robot sous la forme d'une chaîne binaire $\mathcal{O}(G)$.

\mathcal{O} induit une partition (F_0, F_1, \dots) de la famille de graphes \mathcal{G} :
 $F_i = \{G \in \mathcal{G} \mid \mathcal{O}(G) = i\}$

Connaitre une borne supérieure sur n

$$\forall i \exists \tilde{n}_i, \forall G \in F_i : n(G) \leq \tilde{n}_i$$

Tentative de formalisation

Information a priori / conseil

- L'agent doit explorer le graphe G .
- Pour l'aider, un oracle donne un conseil au robot sous la forme d'une chaîne binaire $\mathcal{O}(G)$.

\mathcal{O} induit une partition (F_0, F_1, \dots) de la famille de graphes \mathcal{G} :
 $F_i = \{G \in \mathcal{G} \mid \mathcal{O}(G) = i\}$

Connaître une borne supérieure sur n

$$\forall i \exists \hat{n}_i \forall G \in F_i \quad n(G) \leq \hat{n}_i$$

Résultat négatif (bis)

Théorème

Un agent sans connaissance d'une borne supérieure sur la taille ne peut pas explorer tous les cycles.

(Rappel : conn. borne supérieure $\forall i \exists \hat{n}_i \forall G \in F_i \quad n(G) \leq \hat{n}_i$)

Preuve

- Supposons qu'il existe un agent explorant sans connaissance d'une borne supérieure sur n .
- $\exists i \forall N \exists G_N \in F_i \quad n(G_N) > N$
- Agent sur G_1 : arrêt au temps N
- Agent sur G_N : arrêt avant l'exploration

Résultat négatif (bis)

Théorème

Un agent sans connaissance d'une borne supérieure sur la taille ne peut pas explorer tous les cycles.

(Rappel : conn. borne supérieure $\forall i \exists \hat{n}_i \forall G \in F_i \quad n(G) \leq \hat{n}_i$)

Preuve

- Supposons qu'il existe un agent explorant **sans connaissance d'une borne supérieure** sur n .
- $\exists i \forall N \exists G_N \in F_i \quad n(G_N) > N$
- Agent sur G_1 : arrêt au temps N
- Agent sur G_N : arrêt avant l'exploration

Fin de l'histoire ?

Ensemble des cycles \subset ensemble des graphes

Corollaire

La connaissance d'une borne supérieure sur la taille est également nécessaire pour explorer tous les graphes.

FAUX!

Pas de monotonie...

$\mathcal{O}(G) = 0$ si G est un chemin et $n(G)$ sinon

Fin de l'histoire ?

Ensemble des cycles \subset ensemble des graphes

Corollaire

La connaissance d'une borne supérieure sur la taille est également nécessaire pour explorer tous les graphes.

FAUX!

Pas de monotonie...

$\mathcal{O}(G) = 0$ si G est un chemin et $n(G)$ sinon

Comparaison entre oracles/conseils

Idée : s'abstraire des informations que l'agent peut **calculer**
seul

Définition : $\mathcal{O}' \succeq \mathcal{O}$

Il existe un algo \mathcal{A} tq $\mathcal{O} + \mathcal{A}$ peut **simuler** la sortie de \mathcal{O} .

Théorème (nouvel essai)

Pour tout \mathcal{O} permettant de résoudre l'exploration, il existe \mathcal{O}' donnant une borne sup, tq $\mathcal{O} \succeq \mathcal{O}'$.

Preuve

\mathcal{O} et l'algo d'exploration correspondant $\mathcal{A}_{\mathcal{O}}$ donne une borne supérieure sur la taille de G : le temps de terminaison de $\mathcal{A}_{\mathcal{O}}$

Comparaison entre oracles/conseils

Idée : s'abstraire des informations que l'agent peut **calculer** seul

Définition : $\mathcal{O}' \succeq \mathcal{O}$

Il existe un algo \mathcal{A} tq $\mathcal{O} + \mathcal{A}$ peut **simuler** la sortie de \mathcal{O} .

Théorème (nouvel essai)

Pour tout \mathcal{O} permettant de résoudre l'exploration, il existe \mathcal{O}' donnant une borne sup, tq $\mathcal{O} \succeq \mathcal{O}'$.

Preuve

\mathcal{O} et l'algo d'exploration correspondant $\mathcal{A}_{\mathcal{O}}$ donne une borne supérieure sur la taille de G : le temps de terminaison de $\mathcal{A}_{\mathcal{O}}$

Comparaison entre oracles/conseils

Idée : s'abstraire des informations que l'agent peut **calculer** seul

Définition : $\mathcal{O}' \succeq \mathcal{O}$

Il existe un algo \mathcal{A} tq $\mathcal{O} + \mathcal{A}$ peut **simuler** la sortie de \mathcal{O} .

Théorème (nouvel essai)

Pour tout \mathcal{O} permettant de résoudre l'exploration, il existe \mathcal{O}' donnant une borne sup, tq $\mathcal{O} \succeq \mathcal{O}'$.

Preuve

\mathcal{O} et l'algo d'exploration correspondant $\mathcal{A}_{\mathcal{O}}$ donne une borne supérieure sur la taille de G : le temps de terminaison de $\mathcal{A}_{\mathcal{O}}$