Exploration Optimale Probabiliste d'un Anneau par des Robots Semi-Synchrones et Amnésiques†

Stéphane Devismes[1] Franck Petit[2] et Sébastien Tixeuil[3]

[1] *VERIMAG UMR 5104, Université Joseph Fourier, Grenoble 1 (supporté par le projet ANR* SHAMAN*)*
[2] *INRIA, LIP UMR 5668, Université de Lyon / ENS Lyon (supporté par le projet ANR* R-Discover*)*
[3] *LIP6 UMR 7606, Université Pierre et Marie Curie, Paris 6 (supporté par les projets ANR* SHAMAN *et* R-Discover*)*

Nous considérons une cohorte de k robots mobiles identiques, amnésiques et semi-synchrones, capables de percevoir leur environnement mais pas de communiquer, qui évoluent sur des chemins contraints. Les résultats précédents dans ce contexte montre que les situations initiales symétriques induisent des bornes inférieures élevées quand les problèmes doivent être résolus par des robots *déterministes*. Nous initions l'étude des bornes et solutions *probabilistes* dans le même contexte, et considérons le problème de l'exploration d'anneaux anonymes et non orientés de taille n quelconque. Il est connu que $\Theta(\log n)$ robots sont nécessaires et suffisants pour résoudre le problème avec k robots déterministes, quand k et n sont premiers entre eux. Nous montrons que *quatre* robots probabilistes identiques sont nécessaires et suffisants pour résoudre le même problème, tout en supprimant la contrainte de coprimalité. Nos résultats positifs sont constructifs.

1 Introduction

Nous considérons des cohortes de robots dotés de capteurs visuels (mais incapables de communiquer explicitement) et d'actionneurs de mouvements. Les robots évoluent par *cycles* qui comprennent chacun des phases de *vision*, *calcul* et *déplacement*. Dans un cycle, un robot observe tout d'abord son environnement (phase vision). Basé sur cette observation, le robot décide – de manière probabiliste ou déterministe – de se déplacer vers une destination ou de rester sur place (phase calcul). Enfin, quand il décide d'un déplacement, le robot se déplace vers sa destination (phase déplacement). Nous supposons que les robots ont des capacités très faibles : ils sont *anonymes* (ils n'ont aucun moyen de se distinguer l'un de l'autre), *uniformes* (ils exécutent tous le même protocole), *amnésiques* (leur mémoire volatile s'efface entre deux cycles), et ne savent pas s'orienter (et ne savent pas s'accorder sur une direction ou une orientation commune).

Nous considérons le modèle discret où l'espace est partitionné en un nombre *fini* de lieux. On représente alors l'espace par un graphe dont les nœuds représentent les lieux, et les arêtes la possibilité pour un robot de se déplacer d'un lieu à un autre. Nous nous intéressons au problème de l'exploration, où k robots explorent collectivement un anneau de taille n, avant de stopper leurs déplacements. Le fait de contraindre les robots à s'arrêter une fois l'exploration de tous les lieux effectués oblige les robots à « se souvenir » des régions du graphe déjà explorées, *i.e.* d'être capable de distinguer les différentes étapes du processus d'exploration, puisque les robots ne disposent pas de mémoire persistente. Comme deux configurations ne peuvent être distinguées que par les positions des robots, la mesure principale de complexité est le nombre de robots requis pour explorer un graphe donné. Le nombre important de situations symétriques induit un nombre élevé de robots pour résoudre le problème. Pour les réseaux en arbre, [FIPS08] montre que $\Omega(n)$ robots sont requis pour la plupart des arbres de taille n, et qu'un nombre sublinéaire de robots (plus précisément, $\Theta(\log n/\log\log n)$) ne peut être suffisant que si le degré maximum de l'arbre est 3. Dans les anneaux uniformes, [FIPS07] prouve que $\Theta(\log n)$ robots sont nécessaires et suffisants, et donne un algorithme qui

†Cet article est un résumé étendu de [DPT09], cf. `http://hal.inria.fr/inria-00360305/fr/`

suppose que le nombre k de robots et la taille n de l'anneau sont premiers entre eux. Notons que toutes les approches précédentes dans le modèle discret utilisent des robots *déterministes*.

Dans cet article, nous considérons le modèle *semi-synchrone*. Il est clair que les conditions nécessaires et les bornes présentées dans [FIPS08] pour l'exploration déterministe dans un anneau anonyme reste valable dans le modèle semi-synchrone. Nous proposons ici d'adopter une approche *probabiliste* afin de lever les conditions nécessaires et obtenir de meilleures bornes. Ainsi, contrairement à l'approche déterministe, nous montrons que *quatre* robots probabilistes identiques sont nécessaires et suffisants pour résoudre l'exploration d'un anneau anonyme de taille $n > 8$, tout en supprimant la contrainte de coprimalité. Notre résultat négatif (Section 3) montre que pour tout anneau de taille au moins 4, il ne peut exister aucun protocole d'exploration utilisant 3 robots, même si ces robots disposent de primitives probabilistes. Notre résultat positif (Section 4) est constructif : nous présentons un protocole probabiliste avec quatre robots évoluant sur un anneau de taille supérieure à huit.

2 Modèle

Nous considérons une cohorte de k robots évoluant sur un anneau de n nœuds (avec $k \leq n$), $u_0, \ldots, u_{n-1}$, *i.e.*, u_i est connecté à u_{i-1} et à u_{i+1} (les calculs sur les indices s'effectuant modulo n). Les indices sont utilisés uniquement à des fins de notation : les nœuds sont *anonymes* et l'anneau n'est *pas orienté*, *i.e.*, étant donnés deux nœuds voisins u et v, il n'y a aucun étiquetage explicite ou implicite qui permette de déterminer que u est à la droite ou à la gauche de v. Les robots sont anonymes, semi-synchrone, amnésiques et uniformes.

Chaque robot exécute infiniment des cycles *vision-calcul-déplacement*. Chaque cycle est exécuté atomiquement : tout robot activé à l'instant t exécute un cycle complet entre t et $t+1$. Lors de la phase de vision, un robot peut déterminer si plusieurs robots sont situés sur le même nœud (*détection de la multiplicité*). Dans la suite, $d_i(t)$ dénote le nombre de robots positionnés sur le nœud u_i à l'instant t. Si $d_i(t) \geq 2$, alors une *tour* est présente sur u_i à l'instant t. Le nœud u_i est *libre* à l'instant t si $d_i(t) = 0$ et *occupé* sinon. Étant donnés une orientation arbitraire de l'anneau et un nœud u_i, $\gamma^{+i}(t)$ (resp., $\gamma^{-i}(t)$) dénote la séquence $\langle d_i(t) d_{i+1}(t) \ldots d_{i+n-1}(t) \rangle$ (resp., $\langle d_i(t) d_{i-1}(t) \ldots d_{i-(n-1)}(t) \rangle$). La séquence $\gamma^{-i}(t)$ est le *mirroir* de $\gamma^{+i}(t)$ et vice versa. Puisque l'anneau n'est pas orienté, un accord sur une seule des séquences $\gamma^{+i}(t)$ ou $\gamma^{-i}(t)$ est impossible. La paire $\{\gamma^{+i}(t), \gamma^{-i}(t)\}$ représente la *vue* du nœud u_i à l'instant t. La vue de u_i est *symétrique* quand $\gamma^{+i}(t) = \gamma^{-i}(t)$, et *asymétrique* sinon. Par convention, la *configuration* du système à l'instant t est $\gamma^{+0}(t)$. Une configuration à partir de laquelle il y a une probabilité nulle qu'un robot se déplace est dite *terminale*. Soit $\gamma = \langle x_0 x_1 \ldots x_{n-1} \rangle$ une configuration, la configuration $\langle x_i x_{i+1} \ldots x_{i+n-1} \rangle$ est obtenue par *rotation* de γ $i \in [0 \ldots n-1]$ positions. Deux configuration γ et γ' sont *indistinguables* quand γ' peut être obtenue par rotation de γ ou de son mirroir. Dans le cas contraire, elles sont dites *distinguables*.

Un *ordonnanceur* est un prédicat qui définit l'ensemble des exécutions admissibles. Nous supposons un ordonnanceur *distribué* (à chaque instant, tout sous-ensemble non vide de robots peut être activé) et *équitable* (tout robot est activé infiniment souvent dans une exécution). Notons qu'un ordonnanceur *séquentiel* (qui active à chaque instant exactement un robot) est un cas particulier de l'ordonnanceur distribué. Pendant la phase de vision, il se peut que les deux arêtes adjacentes à un nœud v occupé par un robot semblent identiques, *i.e.* la vue de v est symétrique. Dans ce cas, si le robot décide de bouger, il peut traverser l'une ou l'autre des arêtes : nous considérons le pire cas où le choix de l'arête traversée est décidé par un adversaire.

Un protocole *déterministe* (resp. *probabiliste*) $\mathcal{P}$ résoud le problème de l'exploration si et seulement si toute exécution e de $\mathcal{P}$ partant d'une configuration sans tour satisfait : (*i*) e atteint une configuration terminale en *temps fini* (resp. *temps moyen fini*) ; (*ii*) chaque nœud est visité par au moins un robot au cours de e. Cette spécification implique que toute configuration initiale doit être *sans tour*.

3 Résultat Négatif

Nous montrons que l'exploration est impossible à résoudre quand on dispose de moins de quatre robots, même si le protocole est probabiliste (Théorème 1). La preuve se fait en deux étapes principales. La première étape est basée sur le fait que l'amnésie des robots conduit tout protocole d'exploration à construire une mémoire implicite au moyen des configurations du système. Nous montrons que si l'ordonnanceur se comporte de manière séquentielle, il est impossible de particulariser suffisament les configurations — sauf dans un cas précis — pour se souvenir des nœuds qui ont été visités. La deuxième étape consiste à exclure le cas problématique précité.

Première étape. Pour concevoir un protocole d'exploration avec k robots dans un anneau de $n > k$ nœuds, il faut être capable de construire un sous-ensemble $\mathcal{S}$ d'au moins $n-k+1$ configurations tel que : (*i*) pour tout couple de configurations différentes de $\mathcal{S}$, (γ,γ'), on a γ et γ' qui sont distinguables ; (*ii*) toute configuration de $\mathcal{S}$ contient une tour de moins de k robots. En utilisant ce résultat, nous obtenons que $\forall k, 0 \leq k < 3, \forall n > k$, il n'existe aucun protocole d'exploration (même probabiliste) d'un anneau de n nœuds avec k robots.

Considérons maintenant le cas de 3 robots. La taille de l'ensemble des configurations distinguables contenant une tour de moins de 3 robots est $\lfloor n/2 \rfloor$. Donc il ne peut exister de protocole quand l'inégalité $\lfloor n/2 \rfloor \geq n-k+1$ n'est pas vérifiée. Le seul cas possible restant pour $k < 4$ est $k = 3$ et $n = 4$, que nous considérons lors de la deuxième étape.

Deuxième étape. Il suffit d'étudier tous les protocoles possibles pour $k = 3$ et $n = 4$. Chaque cas amène à l'une des contradictions suivantes : (*i*) les choix adverses de l'ordonnanceur permettent la construction d'un exécution qui ne termine jamais avec probabilité 1 ; (*ii*) pour chaque configuration terminale possible (*i.e.* chaque configuration comprenant une tour), il existe une exécution qui atteint la configuration terminale sans explorer tous les nœuds.

Theorem 1 *$\forall k, 0 \leq k < 4, \forall n > k$, il n'existe aucun protocole d'exploration d'un anneau de taille n par une cohorte de k robots.*

4 Résultat Positif

Dans cette section, nous proposons un protocole d'exploration probabiliste pour une cohorte de $k = 4$ robots dans un anneau de taille $n > 8$.

Définitions. Un *segment* (resp. *trou*) de *longueur* x (ou x-segment, resp. x-trou) est un chemin élémentaire non-vide et maximal de x nœuds consécutifs occupés (resp. libres). Un *nœud isolé* est un 1-segment. Dans le trou $h = u_i, \ldots, u_k$ ($k \geq i$), les nœuds u_i et u_k constituent les *extrémités* de h. Un nœud v est *voisin* du trou h si v n'appartient pas à h mais qu'il est voisin d'une de ses extrémités (h est alors un *trou voisin* de v). Par extension, un robot placé sur v est aussi un voisin du trou h. Une *flèche* est un chemin maximal élémentaire $u_i, \ldots, u_k$ de longueur au moins 4 tel que (*i*) u_i et u_k sont occupés par un robot ; (*ii*) $\forall j \in [i+1 \ldots k-2]$, u_j est libre et (*iii*) il existe une tour de deux robots sur u_{k-1}. Le nœud u_i est la *queue* de la flèche et le nœud u_k est la *tête* de la flèche. La *taille* d'une flèche est définie par l'ensemble des nœuds libres qui la compose. Notons que la taille minimale d'une flèche est 1 et sa taille maximale est $n-3$. Pour $k = 4$ robots, s'il existe une flèche dans une configuration, celle-ci est unique. Une flèche est *primaire* si elle est de taille 1 et *finale* si elle est de taille $n-3$.

Protocole. Dans notre protocole, les robots se déplacent de manière déterministe autant que possible. Un comportement probabiliste est utilisé dans certains cas : quand le système se trouve dans une configuration symétrique, l'ordonnanceur pourrait utiliser l'exécution simultanée de plusieurs robots pour préserver cette symétrie. Pour briser la symétrie malgré les choix de l'ordonnanceur, un robot « lance une pièce » (avec une probabilité uniforme) durant sa phase de calcul : si le résultat est pile, le robot décide un déplacement, sinon il décide de rester sur place. Un déplacement conditionné par un lancer de pièce (*i.e.* un déplacement probabiliste) est noté p-**Déplacement** dans l'algorithme, alors qu'un déplacement inconditionnel (*i.e.* un déplacement déterministe) est noté **Déplacement**.

Notre protocole est composé de trois phases distinctes :

- **Phase** I **:** En partant d'une configuration sans tour, les robots se déplacent le long de l'anneau de telle sorte que : (*i*) ils ne forme aucune tour ; (*ii*) ils forment un segment unique (*i.e.* un 4-segment) en temps moyen fini. Cette phase est la plus complexe et est décrite dans l'algorithme 1. Dans les configurations asymétriques, les robots se déplacent de manière déterministe, et dans les configurations symétriques, les robots se déplacent de manière probabiliste (avec p-**Déplacement**). Dans tous les cas, nous garantissons qu'aucune tour n'est formée en utilisant la contrainte suivante : un robot ne peut se déplacer à travers un x-trou voisin h que si $x > 1$ ou si l'autre robot voisin du x-trou ne peut s'y rendre.

- **Phase** II **:** En partant d'une configuration comprenant un seul 4-segment $u_i, u_{i+1}, u_{i+2}, u_{i+3}$, le système atteint ultimement une configuration où une flèche primaire est formée par les nœuds u_i, u_{i+1}, u_{i+2}, u_{i+3}. Pour ce faire, nous procédons comme suit : soient $\mathcal{R}_1$ et $\mathcal{R}_2$ les robots situés sur les nœuds u_{i+1} et u_{i+2} du 4-segment. $\mathcal{R}_1$ et $\mathcal{R}_2$ effectuent alors un déplacement probabiliste vers u_{i+2} et u_{i+1}, resp. Ultimement un seul de ces robots se déplace et une flèche primaire est constituée.
- **Phase** III **:** En partant d'une configuration où les 4 robots constituent une flèche primaire, la queue de la flèche se déplace de manière déterministe vers la tête de la flèche de telle sorte que la taille de la flèche de décroît jamais. Le protocole termine quand les robots forment une flèche finale. Une fois la flèche finale atteinte, tous les nœuds ont été visités par au moins un robot.

Algorithm 1 Procédure *Phase* I.

```
si la configuration contient un 3-segment alors
    si je suis un robot isolé alors Déplacement vers le 3-segment par le trou le plus petit ;
sinon
    si la configuration contient un unique 2-segment alors        /* deux robots sont isolés */
        si je suis à la plus proche distance du 2-segment alors
            Déplacement vers le 2-segment par le trou dont une extrémité du 2-segment et moi sommes voisins ;
        fin si
    sinon
        si la configuration contient exactement deux 2-segments alors
            si je suis voisin d'un trou le plus long alors p-Déplacement vers l'autre 2-segment par mon trou voisin ;
        sinon        /* les quatre robots sont isolés */
            Soit l_max la longueur du trou le plus grand ;
            si tout robot est voisin d'un l_max-trou alors
                p-Déplacement par un l_max-trou voisin ;
            sinon
                si trois robots sont voisins d'un l_max-trou alors
                    si je suis voisin d'un seul l_max-trou alors
                        Déplacement vers le robot qui n'est voisin d'aucun l_max-trou par mon trou voisin de taille minimale ;
                    fin si
                sinon        /* deux robots sont voisins de l'unique l_max-trou */
                    si je suis un voisin de l'unique l_max-trou alors Déplacement par mon trou voisin de taille minimale ;
                fin si
            fin si
        fin si
    fin si
fin si
```

5 Conclusion

Nous avons prouvé que pour l'exploration des anneaux uniformes, introduire des primitives probabilistes pour le robot pouvait réduire la complexité intrinsèque de $\Theta(\log n)$ à $\Theta(1)$. Etudier l'apport probabiliste pour d'autres problèmes constitue un sujet intéressant, mais nous souhaitons pointer deux questions ouvertes immédiates soulevées par notre travail : (i) trouver des protocoles spécifiques pour les cas où $n \in \{5, \ldots, 8\}$; (ii) calculer la complexité en temps en fonction du nombre k ($4 \leq k < n$) de robots utilisés pour l'exploration.

Références

[DPT09] Stéphane Devismes, Franck Petit, and Sébastien Tixeuil. Optimal probabilistic ring exploration by asynchronous oblivious robots. Technical Report inria-00360305, INRIA, February 2009.

[FIPS07] Paola Flocchini, David Ilcinkas, Andrzej Pelc, and Nicola Santoro. Computing without communicating : Ring exploration by asynchronous oblivious robots. In *OPODIS*, pages 105–118, 2007.

[FIPS08] Paola Flocchini, David Ilcinkas, Andrzej Pelc, and Nicola Santoro. Remembering without memory : Tree exploration by asynchronous oblivious robots. In Alexander A. Shvartsman and Pascal Felber, editors, *SIROCCO*, volume 5058 of *Lecture Notes in Computer Science*, pages 33–47. Springer, 2008.