Vers l'infini et au delà

Quentin Bramas, Stéphane Devismes, Pascal Lafourcade

Vers l'infini et au delà

Quentin Bramas,[1] Stéphane Devismes[2] et Pascal Lafourcade[3]

[1] *Université de Strasbourg, ICUBE, CNRS*
[2] *Université Grenoble Alpes, VERIMAG*
[3] *Université Clermont Auvergne, CNRS UMR 6158, LIMOS*

On considère le problème de l'exploration d'une grille infinie par un nombre fini de robots. Ces robots ont une visibilité finie et exécutent tous le même algorithme de manière synchrone. Les robots ne possèdent pas de système de coordonnées commun, mais sont chiraux (ils savent distinguer leur droite de leur gauche). Les robots possèdent une lumière ayant un nombre fini de couleurs possibles. Mis à part cette lumière, les robots n'ont aucune mémoire, aucun moyen de communiquer explicitement et sont anonymes. Évidemment, la couleur de leur lumière est un moyen de se distinguer, pour casser la symétrie et ainsi exécuter des actions différentes. On montre tout d'abord que cinq robots de ce type sont nécessaires et suffisants pour résoudre le problème de l'exploration de la grille infinie. Pour cela, nous prouvons qu'il n'existe pas d'algorithme n'utilisant que quatre robots, quelle que soit leur visibilité (tant qu'elle est finie). Puis nous donnons un algorithme qui fonctionne avec cinq robots, avec des lumières possédant cinq couleurs, et une visibilité à un saut. Si on se restreint à des robots dont les couleurs ne sont pas modifiables (et toujours avec une visibilité à un saut), on montre que six robots sont nécessaires et suffisants pour résoudre notre problème.

Mots-clefs : robots mobiles, grille infinie, algorithme distribué, exploration

1 Introduction

On étudie comment un nombre fini de robots, ayant une visibilité et une mémoire finies et des capacités de communication restreintes, peuvent réaliser une tâche infinie dans un environnement infini. Pour cela, on considère un ensemble fini de robots équipés de lumières ayant un nombre fini de couleurs possibles. À part ces lumières, les robots n'ont pas de mémoire persistante et aucun moyen de communiquer. Ils exécutent des cycles ≪ *Regarder-Calculer-Se déplacer* ≫ de manière synchrone. Les robots sont placés sur une grille infinie, suivant une configuration bien choisie. Le but est de fournir un algorithme, exécuté par tous les robots, qui permette à chaque point de la grille d'être visité par au moins un robot en temps fini. Dans cette grille, les robots n'ont pas de système de coordonnées commun (c.-à-d. qu'il n'y a *a priori* pas d'accord sur les orientations Nord-Sud et Est-Ouest), mais ont la même chiralité (c.-à-d. une même notion de droite et de gauche).

Dans cet article, nous montrons que cinq robots sont nécessaires et suffisants pour résoudre l'exploration de la grille infinie avec une visibilité à un saut. Si on se restreint à des robots dont la lumière a une couleur fixe (ou, de manière équivalente, à des robots étiquetés de manière non nécessairement distincte), on montre que six robots (toujours avec une visibilité à un saut) sont nécessaires et suffisants pour résoudre notre problème.

Le problème défini par Emek *et al.* [ELS+15] se rapproche le plus de notre modèles. Ils considèrent le problème de la recherche d'un trésor dans une grille de taille non bornée par des agents ayant un système de coordonnées commun, dont la mémoire est finie, et pouvant voir les états des autres agents se trouvant sur le même nœud au même instant. Les hypothèses de visibilité et de système de coordonnées commun rendent le problème solvable à partir de trois robots, alors que notre problème est impossible à résoudre avec quatre robots ou moins.

2 Modèle

Une configuration de n robots dans une grille infinie est représentée par l'ensemble des couples *(position, couleur)* de chacun des robots. Les robots effectuent une infinité de cycles ≪ *Regarder-Calculer-Se Déplacer* ≫ de manière synchrone. À chaque activation, chaque robot *regarde* autour de lui et obtient un

instantané des positions et couleurs des robots situés à distance au plus $\Phi \geq 1$ (distance de Manhattan), lui inclus. Basé uniquement sur cette information visuelle, le robot *calcule* une couleur et une destination (parmi *Haut, Bas, Droite, Gauche, Immobile*), et *se Déplace* vers cette destination en changeant la couleur de sa lumière. Lorsqu'un robot regarde son environnement, l'ensemble des positions et couleurs occupées par les robots à distance au plus Φ est donné selon un système de coordonnées ego-centré dont l'orientation est arbitraire, qui peut être différente à chaque ronde.

Un algorithme est décrit par une configuration initiale globale I, un ensemble de couleurs Cl et un ensemble de règles. Une règle associe une vue locale à une destination, quelle que soit son orientation. Pour simplifier, on suppose ici que la destination ne peut pas être ambiguë [†]. Cela signifie que (1) un algorithme ne contient qu'une règle pour chaque vue, quelle que soit l'orientation et (2) si la vue est symétrique, la destination de la règle ne peut être qu'*Immobile*. Sous ces conditions, *l'exécution*, c.-à-d., la suite des configurations $(C_i)_{i\in\mathbb{N}}$ formées par les robots à chaque ronde, ne dépend pas de l'orientation des vues et est donc uniquement déterminée par la configuration initiale $C_0 = I$. Notre modèle impose aussi que chaque nœud de la grille ne peut contenir qu'au plus un robot. Si en plus un algorithme vérifie que les robots ne se croisent pas sur une arête de la grille (deux robots adjacents n'échangent pas leur position), alors on dit qu'il est *exclusif*.

Un algorithme résout *l'exploration de la grille infinie* si, dans l'exécution $E = (C_i)_{i\in\mathbb{N}}$ avec $C_0 = I$, pour chaque nœud u de la grille, il existe un configuration $C_i \in E$ dans laquelle u est occupée par un robot.

La section suivante présente deux théorèmes d'impossibilité, le premier dans le cas général et le deuxième restreint au cas des robots qui ne change pas de couleur durant l'exécution (couleurs fixes). Puis nous présentons deux algorithmes optimaux en nombre de robots dans chacun des cas.

3 Théorèmes d'impossibilité

Lemme 1. *Si un algorithme résout le problème de l'exploration de la grille infinie, alors la distance entre les deux robots les plus éloignés de chaque configuration tend vers l'infini.*

Démonstration. Supposons par contradiction qu'il existe une borne B telle qu'infiniment souvent tous les robots sont à distance au plus B. Comme le nombre de couleurs et le nombre de positions relatives entre les robots sont finis, alors il existe deux configurations telles que l'une est la translatée de la seconde. Cela implique que l'exécution est périodique, et donc que les robots ne partent que dans une direction, ce qui conduit à une contradiction car une infinité de robots dans la direction opposée ne sont jamais visités. □

Théorème 1. *Le problème de l'exploration infinie de la grille n'est pas solvable avec 4 robots, ou moins, quel que soit le rayon de visibilité et le nombre de couleurs.*

Schéma de preuve. Supposons qu'un algorithme $\mathcal{A}$, exécuté par au moins deux mais au plus quatre robots (l'exploration d'une grille infinie étant trivialement impossible avec un seul robot), permet d'explorer la grille infinie. D'après le Lemme 1, la distance entre les deux robots les plus éloignés tends vers l'infini. On appelle *extrémités*, les robots les plus éloignés d'une configuration. Si $\mathcal{A}$ fonctionne avec deux robots, alors les deux extrémités restent isolées, donc immobiles, une contradiction. On considère maintenant que $\mathcal{A}$ fonctionne avec trois robots. Si une extrémité reste isolée indéfiniment alors les deux autres robots explorent seuls la grille, une contradiction. Donc un robot doit à un moment quitter une extrémité pour rejoindre l'autre extrémité. Dans ce cas, les trois robots finissent par être isolés et donc immobiles pour toujours, une contradiction. Considérons maintenant le cas à quatre robots. Si un robot à une extrémité reste seul indéfiniment, il est forcément immobile et les trois autres robots explorent la grille sans lui, une contradiction. Donc il y a infiniment souvent un groupe de deux robots (les *voyageurs*) qui se déplacent entre deux extrémités (les extrémités et les robots voyageurs ne sont pas forcément toujours les mêmes). Cependant, un groupe de robots qui voyage sur une distance arbitraire exécute forcément une suite de mouvements périodique, donc lorsqu'ils atteignent l'autre extrémité, ils ne savent pas quelle distance ils ont parcouru. Cela implique que les mouvements effectués par les extrémités sont périodiques, car ils ne dépendent que d'un nombre fini de paramètres. Donc chaque extrémité se déplace dans une direction unique. Les nœuds

†. Le cas général, les algorithmes et les preuves de tous théorèmes sont disponibles dans le rapport technique en ligne [BDL19].

qui sont entre les extrémités de la configuration peuvent être visités par le groupe voyageur mais quelles que soient les directions prises par les extrémités, il y a forcément des nœuds qui ne sont jamais visités. □

La preuve du théorème ci-dessous reprend les mêmes principes que celle du théorème 1. On a deux cas principaux : si le groupe de voyageurs est composé de trois robots, on revient au cas précédent ; si le groupe de voyageurs contient deux robots, alors, dû au fait que les couleurs sont fixes et que le rayon de visibilité est un, on prouve qu'il ne peut que se déplacer en ligne droite lorsqu'il est isolé.

Théorème 2. *Le problème de l'exploration infinie de la grille n'est pas solvable avec 5 robots, ou moins, avec un rayon de visibilité 1, lorsque les robots ont des couleurs fixes.*

4 Algorithme avec Six Robots de Couleur Fixe

Il est conseillé de lire les explications qui suivent en observant l'exécution de notre algorithme à l'aide de l'animation fournie en ligne [‡].

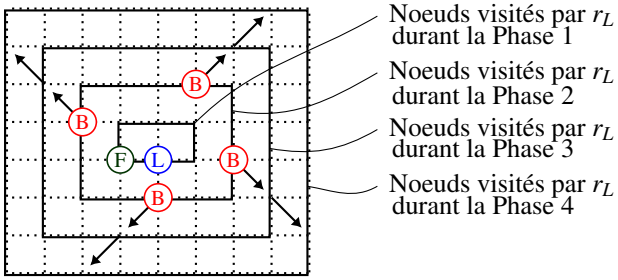

FIGURE 1: Nœuds visités après quatre phases de l'algorithme $\mathcal{A}_1^{fixed}$.

Comme les robots n'ont pas de mémoire autre que la couleur de leur lumière, il faut que la configuration *contienne* l'information de ce qui a été visitée. Formellement, cela signifie que certains robots vont jouer le rôle de balise délimitant indirectement la zone déjà explorée. Nos algorithmes fonctionnent par phase, et à chaque phase, un groupe de robots voyageur va parcourir les limites de la zone en se déplaçant de balise en balise, jusqu'à faire un tour complet. À chaque fois que le groupe voyageur rencontre une balise, un *ajustement* se produit. Cet ajustement à pour conséquence de déplacer la balise (généralement en diagonale) et d'orienter le groupe voyageur afin qu'il continue son périple vers la balise suivante. Un fois le tour complet terminé (c.-à-d. lorsqu'une phase est réalisée), la zone déjà explorée est plus grande, et les balises ont été déplacées pour délimiter cette nouvelle zone. On peut voir sur la Figure 1 la configuration initiale de notre premier algorithme, avec les quatre balises en rouge notées B et les robots vert et bleu notés F (pour *Follower*) et L (pour *Leader*) formant le groupe voyageur. Les balises délimitent le rectangle strictement contenu dans le plus petit rectangle englobant les balises, qui sera visité par le leader durant la prochaine phase. Le plus petit rectangle dessiné est donc visité par le leader durant la première phase. Les flèches indiquent les déplacement effectués par les balises lors des ajustements durant les deux premières phases.

Nous présentons maintenant les règles de notre algorithme $\mathcal{A}_1^{fixed}$. La Figure 2 présente les règles exécutées par le groupe voyageur lui permettant de se déplacer en ligne droite. On voit notamment qu'un robot L qui voit un unique voisin F, se dirige à la gauche de ce robot. Inversement un robot F qui voit un unique robot L se déplace à la droite de celui-ci. Les règles sont présentées comme si le groupe voyageur se déplaçait vers le haut, mais les même règles définissent aussi les déplacements vers la gauche, le bas et la droite (en fonction de la position relative des deux robots).

Lorsque le groupe voyageur rencontre un robot balise, le robot L est le premier à le voir. Comme notre algorithme n'a pas de règle pour la vue d'un robot L possédant un voisin F et un voisin B, ce robot L reste immobile. Le robot F par contre n'a pas encore vu le robot B, il continue donc son déplacement. Lorsque le robot F voit le robot B, il a la vue de la règle $\mathcal{R}_{\mathcal{V}rnF1}$, illustrée Figure 3. B exécute règle $\mathcal{R}_{\mathcal{V}rnB1}$, et les robots échangent leurs positions. Ensuite le groupe voyageur est reformé, mais a effectué une rotation d'angle $\pi/2$. La règle $\mathcal{R}_{\mathcal{V}rnB2}$, illustrée Figure 4, indique au robot F de se déplacer exactement comme si le robot B n'était pas présent. La règle $\mathcal{R}_{\mathcal{V}rnF2}$ permet de terminer l'ajustement de la balise. Durant les deux rondes d'ajustement, la balise s'est déplacée en diagonale (d'abord à droite, puis en haut, si on utilise la même orientation).

‡. https://doi.org/10.5281/zenodo.2625730

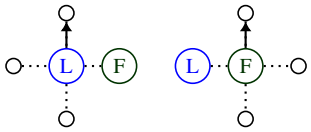

FIGURE 2: $\mathcal{R}_{strL}$ et $\mathcal{R}_{strF}$.

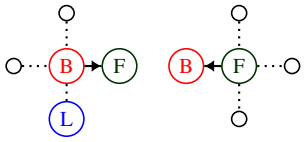

FIGURE 3: $\mathcal{R}_{trnB1}$ et $\mathcal{R}_{trnF1}$.

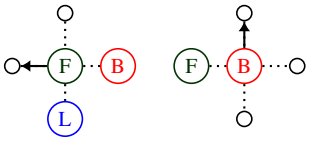

FIGURE 4: $\mathcal{R}_{trnB2}$ et $\mathcal{R}_{trnF2}$.

Théorème 3. *L'algorithme $\mathcal{A}_1^{fixed}$ explore la grille infinie avec six robots de couleur fixe. Il est optimal en nombre de robot mais n'est pas exclusif.*

5 Algorithme avec Cinq Robots

Le principe de l'algorithme $\mathcal{A}_1^{modifiable}$ est similaire au précédent, avec des balises qui délimitent un triangle à la place d'un rectangle. Les balises sont les robots de couleurs verte (G), rouge (R) et jaunes (Y, le plus à droite). Le groupe voyageur est formé des robots bleu (B) et jaune (Y sous le robot bleu). Les flèches indiquent les déplacement effectués par les balises lors des ajustements, durant les deux premières phases. Les triangles représentent les nœuds visités lors des trois première phases.

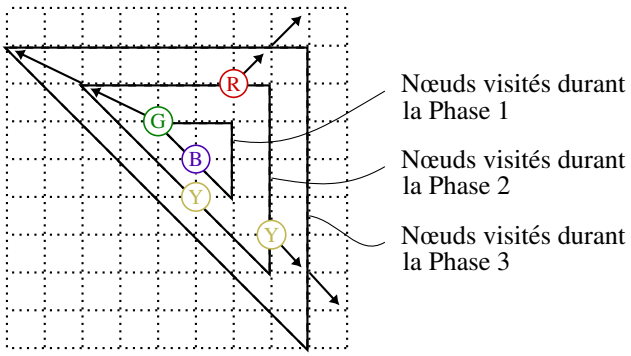

FIGURE 5: Nœuds visités après trois phases de l'algorithme $\mathcal{A}_1^{modifiable}$.

Nous présentons maintenant certaines règles exécutées par les robots qui permettent une telle exécution. La Figure 6 présente les règles exécutées par le groupe voyageur lors du déplacement en diagonale (les autres règles étant similaires à l'algorithme précédent). On voit notamment qu'un robot jaune qui possède un unique voisin bleu, se dirige à la droite de ce robot et met sa couleur à violet (P). Aussi, un robot de couleur bleu suit toujours un robot violet ou un robot jaune. La Figure 7 présente la vue globale du déplacement en diagonal.

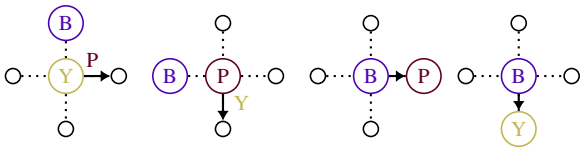

FIGURE 6: Règle de l'algorithme $\mathcal{A}_1$ permettant un déplacement en diagonal du groupe voyageur.

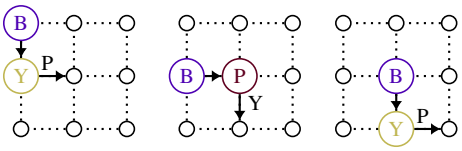

FIGURE 7: Vue globale du déplacement en diagonal du groupe voyageur.

Théorème 4. *L'algorithme $\mathcal{A}_1^{modifiable}$ explore de manière exclusive la grille infinie avec cinq robots et cinq couleurs modifiables. Il est optimal en nombre de robot.*

Références

[BDL19] Quentin Bramas, Stéphane Devismes, and Pascal Lafourcade. Infinite grid exploration by disoriented robots. *CoRR*, abs/1905.09271, 2019.

[ELS+15] Yuval Emek, Tobias Langner, David Stolz, Jara Uitto, and Roger Wattenhofer. How many ants does it take to find the food ? *Theoretical Computer Science*, 608(P3) :255–267, December 2015.