

# L'intelligence collective

Fabrice Rossi

<http://apiacoa.org/>

Publication originale : Juin 2003

Version actuelle : 24 septembre 2006

Cet article est paru dans le numéro 51 de la revue GNU/Linux Magazine France en juin 2003. Je remercie Denis Bodor et les éditions Diamond pour avoir autorisé la distribution de cet article selon les conditions indiquées dans la section A. L'article est disponible en ligne à l'URL <http://apiacoa.org/publications/2003/lm-ants.pdf>.

## Résumé

L'intelligence artificielle est souvent résumée par un but extrêmement ambitieux, rendre l'ordinateur aussi intelligent que l'homme, et faire en sorte qu'il passe le test de Turing : être indistinguable d'un être humain lors d'une conversation en aveugle. Force est de constater que dans ce domaine, les progrès sont plutôt très lents. Pourtant, dans des domaines moins ambitieux, l'intelligence des ordinateurs progresse constamment. Une des techniques utilisées par les chercheurs depuis quelques années est de s'inspirer de la nature et de l'intelligence **en général**, plutôt que de se focaliser sur l'intelligence **humaine**. Le but de cet article est d'illustrer cette tendance en montrant comment l'intelligence collective qui apparaît dans les colonies d'insectes sociaux est constituée à partir d'une agrégation de comportements individuels élémentaires difficilement qualifiables d'intelligents. Nous verrons comment ces mécanismes peuvent être reproduits informatiquement afin de résoudre des problèmes pour lesquels on propose naturellement des solutions beaucoup plus complexes et délicates à mettre au point.

## 1 Introduction

Quand on évoque l'intelligence artificielle (IA), le lecteur pense presque systématiquement à des algorithmes très complexes comme les moteurs d'inférence (les fameux systèmes experts) ou les réseaux de neurones artificiels. Il faut dire qu'on demande à l'IA de réaliser des tâches très évoluées comme la reconnaissance de visages, de l'écriture manuscrite ou de la parole. On cherche souvent à rendre l'ordinateur aussi intelligent que l'être humain.

La plupart des animaux sont beaucoup moins intelligents que l'homme mais sont cependant capables de réalisations assez évoluées. Considérons par exemple les insectes. Individuellement une fourmi ou une abeille ne sont pas des animaux très évolués. Leur cerveau est minuscule et n'est pas capable de s'adapter à des situations nouvelles. Pourtant, le groupe réalise des tâches très évoluées, comme la construction d'une ruche ou d'une fourmilière, la collecte de la nourriture, l'élevage des larves, etc. Les insectes *sociaux* [1] sont fascinants car ils semblent compenser leurs faiblesses individuelles par une coordination globale qui donne à la colonie une forme d'intelligence bien supérieure à celles de ses membres. Le plus étonnant est que cette coordination n'existe pas matériellement, au sens où la colonie ne comporte aucun

chef. Par des mécanismes élémentaires, les actions individuelles se combinent pour contribuer à la réalisation du but global, sans plan apparent.

Les mécanismes qui permettent le fonctionnement des colonies d'insectes sont très intéressants pour les chercheurs en IA [2]. D'ailleurs, cette discipline s'est souvent basée sur une forme d'imitation de la nature pour progresser. Les moteurs d'inférence sont par exemple une informatisation des mécanismes de raisonnement logique mis en oeuvre par les humains. Les réseaux de neurones artificiels sont issus de modèles mathématiques élémentaires des neurones humains. Les systèmes dits *multi-agents* sont eux inspirés de l'intelligence collective des insectes sociaux. Ces systèmes s'inspirent des colonies en se basant sur des agents simples (le modèle informatique d'un insecte) qui interagissent afin de produire un comportement global intelligent. Les gros avantages de l'approche multi-agents sont les suivants :

- les agents utilisent des algorithmes assez simples et donc en général plus faciles à mettre au point que les programmes des autres méthodes de l'IA ;
- l'intelligence globale est souvent croissante avec le nombre d'agents, ce qui permet d'adapter relativement aisément la puissance du système aux ressources disponibles ;
- l'absence de contrôle central autorise une implémentation sous forme de système réparti, par exemple sur plusieurs ordinateurs. On peut ainsi profiter d'un *cluster* de machines, ce qui renforce la souplesse du modèle et ses possibilités d'adaptation aux tâches demandées ;
- les communications et actions sont en général locales, c'est-à-dire qu'un agent peut seulement interagir avec son environnement immédiat, ce qui facilite une implémentation répartie efficace.

On compte de très nombreuses applications pratiques de l'intelligence collective, en particulier dans les problèmes d'optimisation. Des concepts assez proches de ceux qui seront exposés dans cet article (en particulier les mécanismes de marquage par phéromones) ont été utilisés pour proposer des solutions heuristiques au problème du voyageur de commerce. Ce problème consiste à trouver le chemin le plus court passant par un ensemble fixé de villes. Il fait partie de la classe des problèmes NP-complets pour lesquels on ne connaît pas (actuellement) de solution exacte en temps polynomial. Depuis les travaux de Marco Dorigo et de ses collègues [3], l'approche inspirée du comportement des fourmis a été appliquée à divers problèmes d'optimisation combinatoire (en productique et en télécommunication par exemple). On compte aussi des applications en robotique, en analyse des données, etc.

L'intelligence collective est un domaine extrêmement riche, impossible à couvrir en un article, et même d'ailleurs en un numéro entier de LinuxMag. Le but de cet article est donc de décrire un modèle multi-agents élémentaire afin de montrer comment l'intelligence collective émerge naturellement par combinaison de comportements très simples. L'exemple choisi est celui d'une colonie de fourmis car il est introduit les concepts de coopération par l'utilisation de phéromones qui sont à la base des applications concrètes issues des travaux de Marco Dorigo et de ses collègues.

## 2 Une colonie de fourmis

### 2.1 La récolte de nourriture par des fourmis

Nous allons étudier le problème du ramassage de nourriture par les fourmis d'une unique fourmilière [4]. La fourmilière est le point de départ des fourmis qui explorent le monde qui les entoure. Quand une fourmi découvre de la nourriture, elle prélève le plus gros morceau qu'elle

est capable de porter, puis elle le rapporte à la fourmilière. Une fois la nourriture stockée en lieu sûr, la fourmi retourne au travail.

Le modèle est simplifié à l'extrême afin de faciliter son étude. Par exemple, nous ne prendrons pas en compte la compétition entre fourmilières. De même nous ne tiendrons pas compte de la fatigue des fourmis, du fait que la nourriture se présente parfois sous forme d'éléments difficiles à séparer et que plusieurs fourmis doivent s'associer pour la transporter.

## 2.2 Modèle informatique

Nous adoptons un modèle informatique discret : comme l'illustre la figure 1, le monde est constitué d'une grille de taille finie (les bords sont infranchissables) dont chaque case peut contenir une ou plusieurs fourmis, ainsi qu'un nombre arbitraire d'unités de nourriture. Certaines cases peuvent contenir des obstacles infranchissables. La fourmilière occupe une case unique.

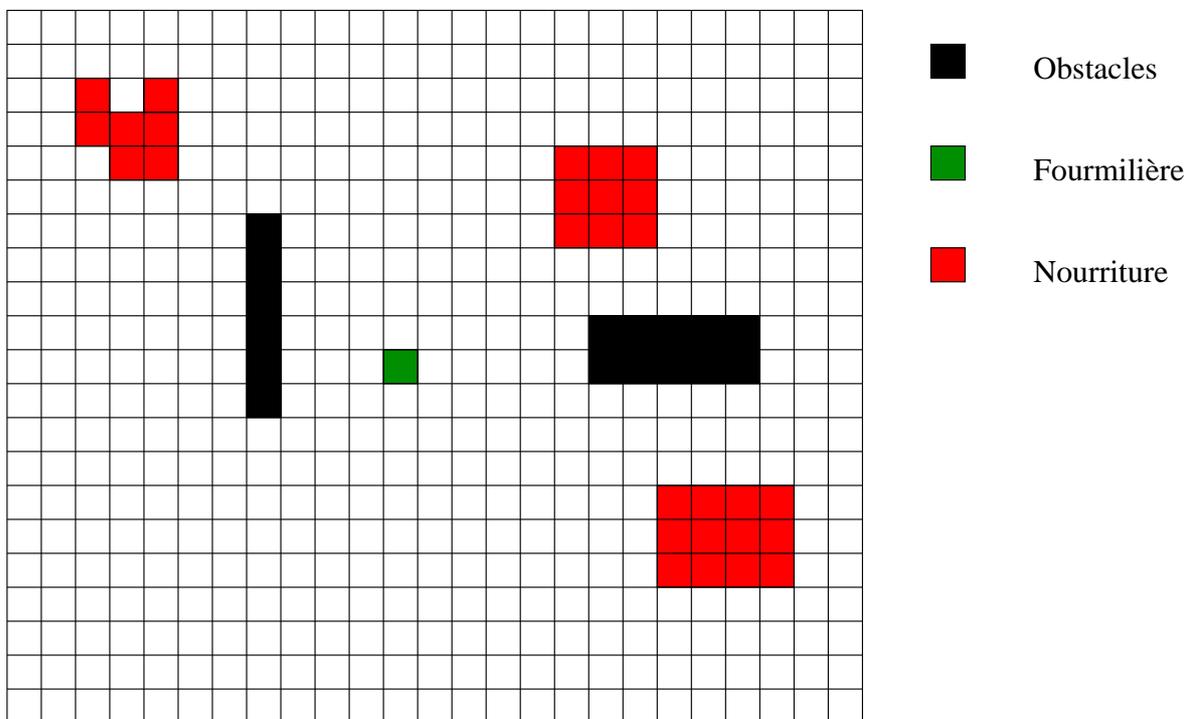


FIG. 1 – Dessin du monde

Quand une fourmi se déplace, elle passe directement d'une case à une des cases voisines. Le temps s'écoule de façon discrète, c'est-à-dire qu'on passe de l'instant  $t$  directement à l'instant  $t+1$  en réalisant séquentiellement les mouvements de toutes les fourmis étudiées.

Une fourmi possède une orientation, c'est-à-dire qu'elle regarde une des 8 cases voisines. Cette orientation est importante car la fourmi ne peut se déplacer que vers la case observée. En d'autres termes, pour se déplacer, la fourmi commence par s'orienter, puis avance dans la direction choisie, comme l'illustre la figure 2.

Une fourmi peut prendre une décision de mouvement en fonction de son environnement. Par exemple, elle peut décider de se déplacer vers de la nourriture si elle en voit dans les cases qui l'entourent.

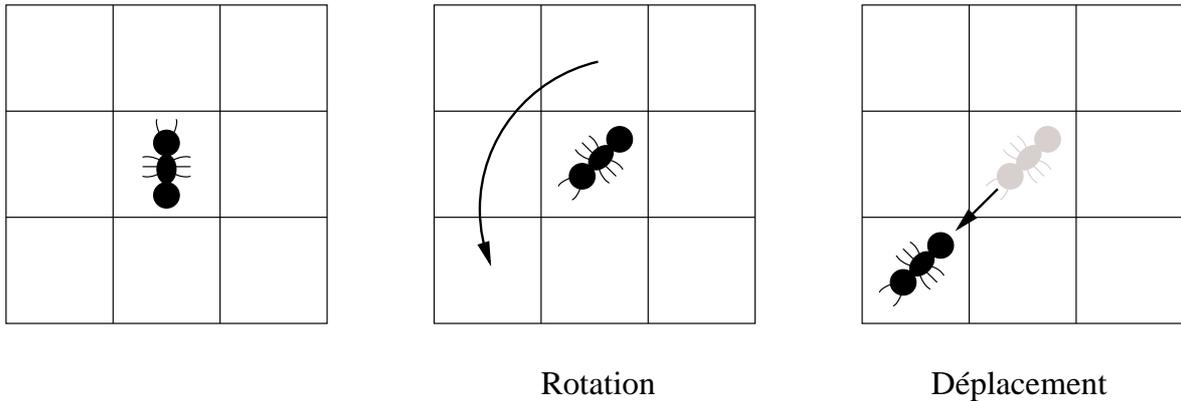


FIG. 2 – Mouvement d'une fourmi

Pour évaluer l'efficacité des fourmis dans leur collecte de nourriture, on peut tracer l'évolution de la quantité de nourriture stockée dans la fourmilière au cours du temps. Cette évolution sera d'autant plus rapide que la collecte est efficace.

### 2.3 Implémentation

Le lecteur trouvera sur le site de l'auteur (<http://apiacoa.org/software/ants/>) une implémentation en Java du modèle proposé. Cette implémentation est sous licence GPL et toute collaboration est la bienvenue, d'autant que l'aspect graphique du logiciel est pour l'instant très basique, comme le montre la capture d'écran de la figure 3.

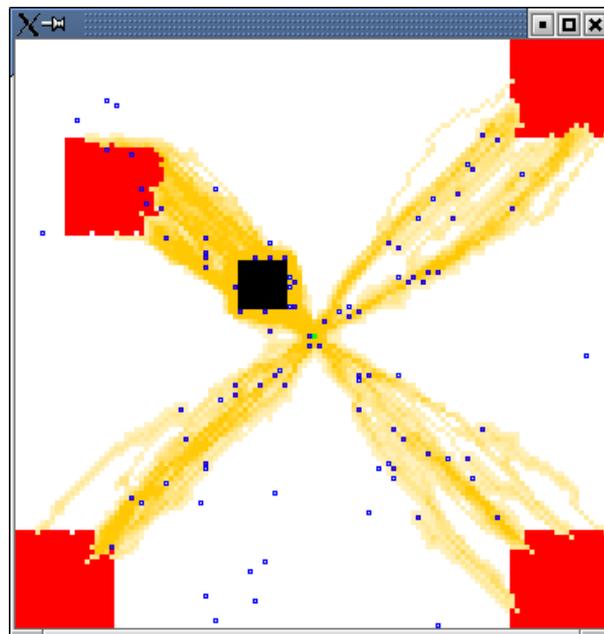


FIG. 3 – Capture d'écran

### 3 Des fourmis stupides

#### 3.1 Modèle de base

Dans un premier temps, nous étudions des fourmis très stupides. Leur mémoire est tellement limitée qu'elle ne contient que l'emplacement de la fourmilière, ce qui permet à une fourmi de revenir à celle-ci dès qu'elle a trouvé de la nourriture. La fourmi possède un cerveau qui est soit en mode *recherche*, soit en mode *retour à la fourmilière*. En mode *recherche*, la fourmi applique l'algorithme suivant :

1. choisir la prochaine direction au hasard
2. s'orienter dans cette direction
3. avancer d'un pas
4. si la case atteinte contient de la nourriture :
  - (a) ramasser la nourriture
  - (b) passer en mode *retour à la fourmilière*
5. sinon, retourner en 1

En mode *retour à la fourmilière*, la fourmi applique l'algorithme suivant :

1. se tourner vers la fourmilière
2. avancer d'un pas
3. si la case atteinte est la fourmilière :
  - (a) déposer la nourriture
  - (b) faire demi-tour
  - (c) passer en mode *recherche*
4. sinon, retourner en 1

Les algorithmes appliqués sont donc extrêmement simples. Le but du demi-tour dans le mode *retour à la fourmilière* consiste pour la fourmi à repartir dans la direction d'où elle vient, et donc à essayer de rejoindre l'amas de nourriture dans lequel elle a prélevé la nourriture qu'elle rapporte.

Dans le mode *recherche*, la fourmi doit choisir au hasard sa prochaine direction. Pour éviter des mouvements trop désordonnés, la nouvelle direction est choisie relativement à l'ancienne, c'est-à-dire sous forme d'une rotation. Les probabilités retenues auront une influence sur le comportement exploratoire des fourmis. Par exemple, si on donne une probabilité élevée aux rotations d'angle faible, la fourmi va avoir tendance à progresser tout droit, alors que des probabilités uniformes conduisent à un mouvement plus désordonné. Nous étudierons l'influence des probabilités sur la vitesse d'exploration dans les sections suivantes.

Dans le mode *retour à la fourmilière*, la fourmi ne choisit pas toujours la direction optimale (celle de la fourmilière). On considère en effet qu'elle n'est pas assez intelligente pour se déplacer sans erreur. On modélise l'erreur de direction par une rotation aléatoire par rapport à la direction optimale, comme dans le mode *recherche*. Bien entendu, les probabilités doivent être adaptées afin que la fourmi puisse rentrer à la fourmilière en un temps raisonnable. Notons d'autre part qu'il faut prévoir un algorithme permettant d'éviter les obstacles. Ce type d'algorithme est assez difficile à mettre au point. En fait, il existe des solutions connues et optimales, mais elles sont totalement incompatibles avec la notion d'agent élémentaire. En

général, ces solutions ne sont pas locales (la fourmi doit voir à distance) ou nécessitent la mémorisation des déplacements antérieurs. On a donc choisi un algorithme très simple qui ne fonctionne pas dans le cas général, mais seulement pour des cas élémentaires comme celui présenté dans la section suivante. L'idée de la méthode est de choisir la direction admissible (i.e., qui mène à une case libre) la plus proche de la direction de la fourmière. Pour éviter que les fourmis prennent toutes le même chemin, chaque fourmi a un sens de rotation préféré et choisira donc la direction la plus proche, soit dans le sens des aiguilles d'une montre, soit dans le sens contraire.

### 3.2 Premiers résultats

Pour évaluer la qualité de la collecte de nourriture, on choisit une configuration test du monde avec les caractéristiques suivantes :

- le monde est constitué d'un carré de 120x120 cases ;
- la fourmière est au centre du monde (le point vert dans le dessin) ;
- quatre tas de nourriture sont placés dans les quatre coins. Le tas en haut à gauche est rapproché de la fourmière (les zones rouges) ;
- il y a au total 1600 unités de nourriture, chaque tas couvrant une zone de 20x20 cases, avec une unité par case ;
- un obstacle carré est placé entre la fourmière et le tas en haut à gauche (le carré noir) ;
- on considère une population de 100 fourmis ;
- enfin, on étudie l'évolution de la quantité de nourriture collectée sur 4000 unités de temps.

Le monde obtenu est représenté par la figure 4.

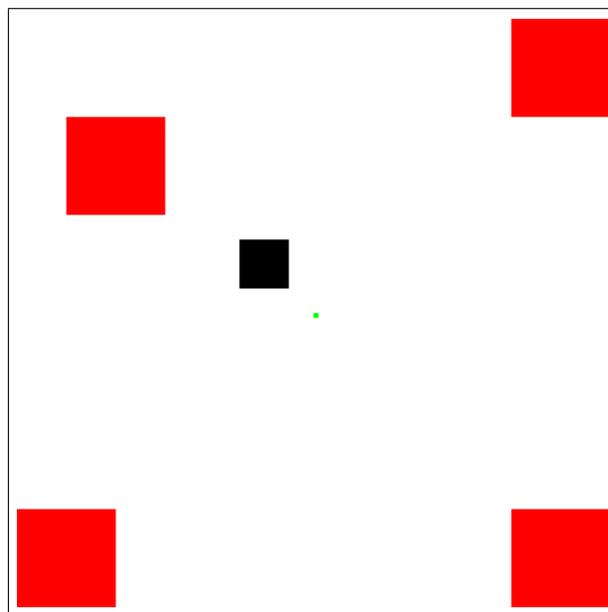


FIG. 4 – Monde de test

On peut étudier dans un premier temps l'effet de la directivité des mouvements, c'est-à-dire du choix des probabilités de rotation d'une fourmi en mode *recherche*. On se donne par exemple trois distributions de probabilité résumées dans le tableau suivant :

Rotation	0	+/- 1	+/- 2	+/-3	+/-4
Distribution 1	$\frac{5}{24}$	$\frac{8}{24}$	$\frac{6}{24}$	$\frac{4}{24}$	$\frac{1}{24}$
Distribution 2	$\frac{12}{19}$	$\frac{4}{19}$	$\frac{2}{19}$	$\frac{1}{19}$	0
Distribution 3	$\frac{12}{16}$	$\frac{2}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	0

Le tableau se lit de la façon suivante : quand une fourmi regarde vers la case juste au dessus d'elle, elle a 4 chances sur 19 de se tourner d'une case vers la gauche ou vers la droite avec la distribution 2. On constate que la distribution 3 est très directive (la fourmi a tendance à continuer tout droit), alors que la distribution 2 est moyennement directive. La distribution 1 au contraire est très peu directive car la fourmi n'a que 5 chances sur 24 de continuer tout droit. La figure 5 représente l'évolution de la quantité de nourriture dans la fourmilière en fonction de la distribution choisie.

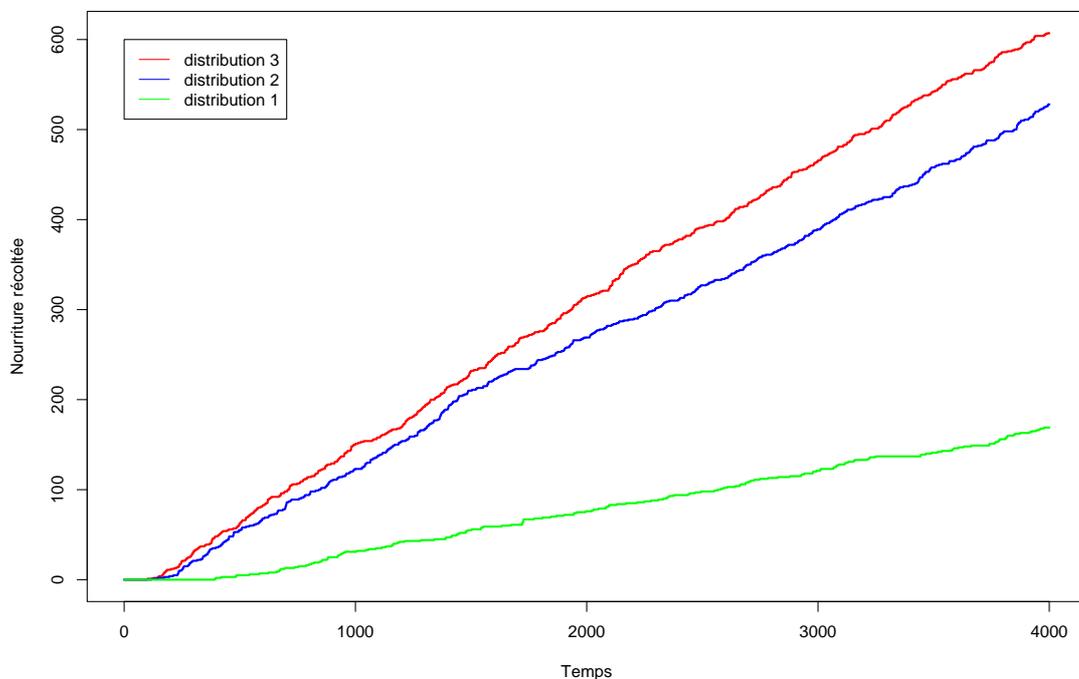


FIG. 5 – Comparaison des distributions : collecte de nourriture

On constate que la distribution directive est beaucoup plus efficace que les autres sur l'exemple étudié. Pour comprendre les raisons de cette efficacité, on peut observer l'emplacement des fourmis après 100 déplacements (figure 6). On constate que pour la distribution 1 (image de gauche), les fourmis (en bleu) sont restées relativement concentrées autour de la fourmilière, alors que pour la distribution 3 (image de droite), elles ont parcouru une plus grande distance et ont donc exploré plus efficacement le monde.

Dans la suite du texte, nous utiliserons donc la distribution 3. Il faut noter que pour une autre configuration du monde, une exploration moins directive aurait pu donner de meilleurs résultats. De plus, on pourrait aussi s'intéresser aux probabilités utilisées dans le mode *retour à la fourmilière*. Cependant, le comportement de chaque fourmi n'étant pas le thème principal de cet article, nous nous contenterons de donner la distribution utilisée pour le retour :

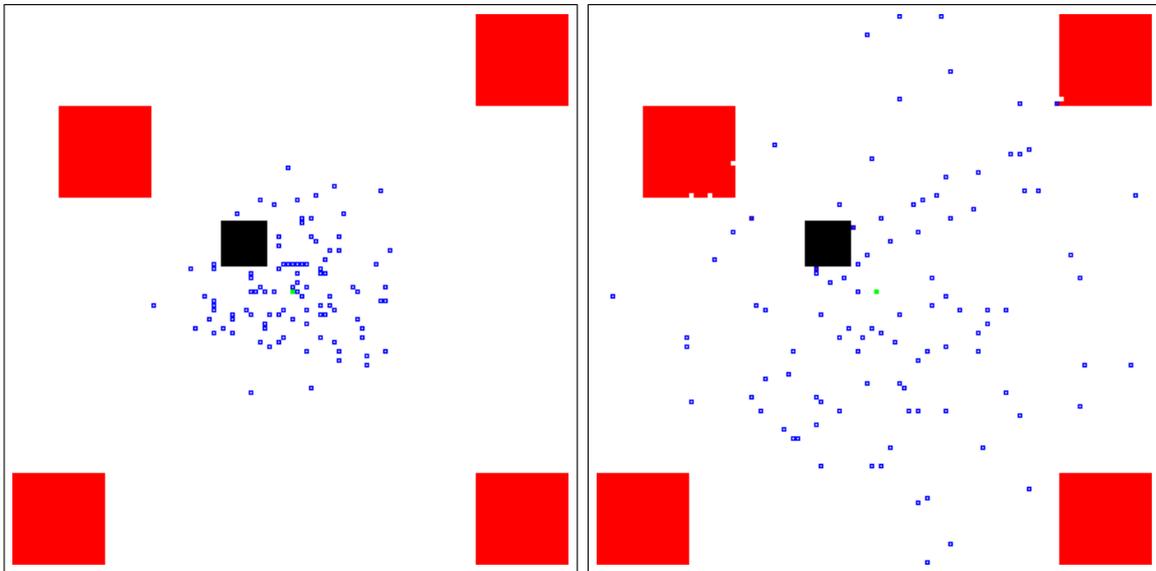


FIG. 6 – Comparaison des distributions : amplitude des déplacements

Rotation	0	+/- 1	+/- 2	+/-3	+/-4
Distribution 1	$\frac{80}{91}$	$\frac{10}{91}$	$\frac{1}{91}$	0	0

On constate que la distribution retenue est très directive. Contrairement au mode *recherche*, la rotation ne s'effectue pas ici par rapport à la direction précédente, mais par rapport à la direction optimale (celle de la fourmilière).

### 3.3 Perception

Les fourmis que nous venons d'étudier sont vraiment très basiques, car elles ne perçoivent même pas leur environnement. On peut donc introduire une forme de perception de la nourriture pour tenter d'améliorer la collecte. Cette forme d'intelligence élémentaire modifie le comportement de la fourmi en mode *recherche*. Au lieu de choisir au hasard sa nouvelle direction, la fourmi peut se tourner automatiquement vers une case contenant de la nourriture si une telle case fait partie des huit positions voisines de la sienne. On constate sur la figure 7 que l'amélioration de la collecte reste cependant quasi nulle avec une telle forme de perception.

## 4 L'intelligence collective

### 4.1 Les phéromones

Dans la nature, les fourmis exhibent un comportement collectif en utilisant un outil de communication très efficace, les phéromones. Une phéromone est une hormone émise dans le milieu extérieur, c'est-à-dire une substance chimique dont les propriétés peuvent altérer le comportement du récepteur. Les hormones sont un mécanisme de communication chimique très utilisé en interne par les êtres vivants. Chez l'être humain, par exemple, l'hypophyse (un élément du cerveau) produit l'hormone de croissance qui est véhiculée par le sang jusqu'à certaines cellules cibles afin de stimuler la croissance verticale du corps. Les phéromones

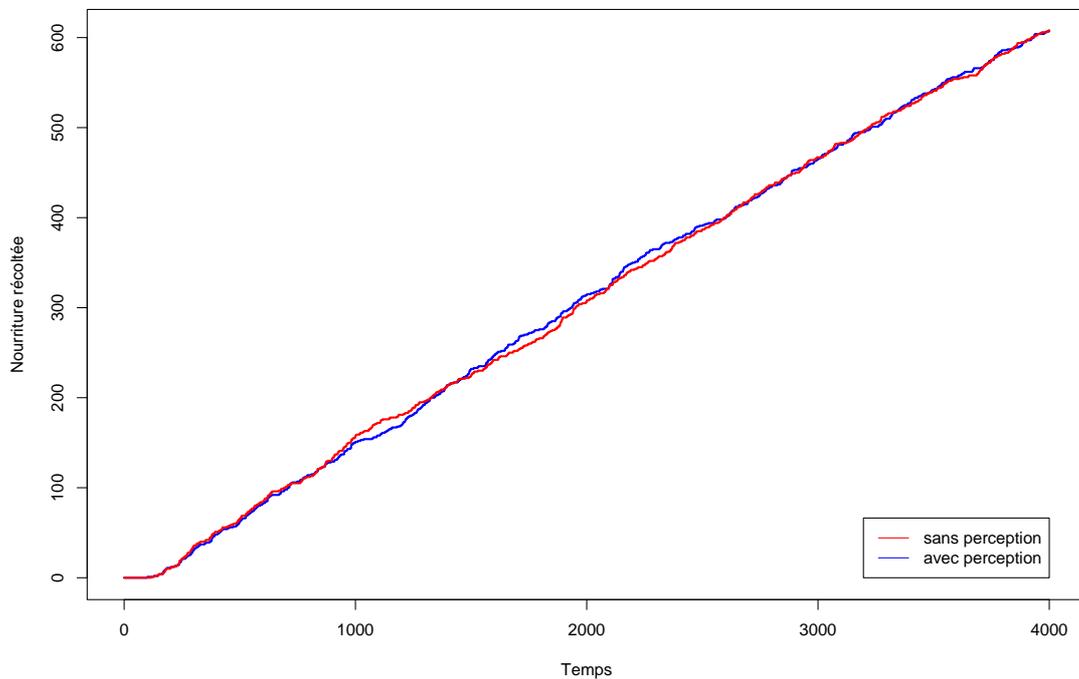


FIG. 7 – Collecte de nourriture : avec ou sans perception

(vraisemblablement spécifiques au règne animal) sont émises dans le milieu extérieur et permettent à des animaux de communiquer entre eux. Elles entrent notamment en jeu dans les mécanismes de reproduction. Dans le cas des insectes sociaux, elles sont aussi utilisées pour coordonner la colonie.

Les fourmis utilisent en effet des phéromones pour laisser des pistes que les autres fourmis vont suivre pour s'orienter. Plus précisément, on suppose qu'une fourmi peut déposer des gouttes de phéromones sur le sol. C'est ce qu'elle va faire en mode *retour à la fourmilière*, l'algorithme devenant donc :

1. déposer une goutte de phéromones
2. se tourner vers la fourmilière
3. avancer d'un pas
4. si la case atteinte est la fourmilière :
  - (a) déposer la nourriture
  - (b) faire demi-tour
  - (c) passer en mode *recherche*
5. sinon, retourner en 1

Une fourmi qui porte de la nourriture fabrique donc une piste entre l'emplacement du bloc de nourriture et la fourmilière. Les fourmis en mode *recherche* peuvent alors profiter de la piste ainsi obtenue pour se diriger vers la nourriture. De plus, le niveau de phéromones dans une case du monde est une indication du nombre de fourmis transportant de la nourriture

qui sont passées par cette case. Les fourmis en *recherche* ont donc intérêt à se diriger vers des cases qui contiennent beaucoup de phéromones. Pour tenir compte de ce comportement, la nouvelle direction d'une fourmi en mode *recherche* est déterminée par l'algorithme suivant :

1. si au moins une des 8 cases voisines contient de la nourriture :
  - (a) déterminer les cases contenant le plus de nourriture
  - (b) se tourner aléatoirement vers une de ces cases
2. sinon, si au moins une des 8 cases voisines contient des phéromones :
  - (a) déterminer la case contenant des phéromones la plus éloignée de la fourmilière
  - (b) en cas d'ex-aequo, choisir la case contenant le plus de phéromones
  - (c) se tourner vers la case choisie
3. choisir la prochaine direction au hasard à partir de la direction actuelle

On constate que l'algorithme est assez évolué, mais possède des caractéristiques très intéressantes :

- l'algorithme est local : la fourmi étudie seulement les cases qui l'entourent. On peut bien sûr critiquer l'utilisation de la distance à la fourmilière, mais c'est en fait une façon simple d'exprimer le fait qu'une fourmi va naturellement suivre une piste de phéromones dans le sens qui l'éloigne de la fourmilière. Comme la fourmi connaît la direction de la fourmilière, on pourrait exprimer l'algorithme en terme de cette direction. Il serait alors moins lisible mais équivalent. Notons de plus que l'information donnant la position de la fourmilière peut être maintenue dynamiquement par la fourmi en prenant en compte ses propres déplacements ;
- la communication entre fourmis est indirecte : toute la communication est basée sur les phéromones, il n'y a pas de liaison entre les fourmis ;
- une fourmi n'a pas de mémoire, elle ne sait pas par où elle est déjà passée, où elle a déjà ramassé de la nourriture, etc. En fait, le sol joue le rôle d'une mémoire globale et répartie pour l'ensemble de fourmis.

Pour éviter que les pistes persistent quand une source de nourriture est épuisée, on donne une durée de vie aux phéromones, c'est-à-dire que celles-ci s'évaporent progressivement. Pour qu'une piste soit conservée, il faut donc qu'elle soit entretenue, c'est-à-dire utilisée par des fourmis portant de la nourriture.

## 4.2 Résultats

La figure 8 compare la collecte de nourriture avec et sans phéromone. On constate que l'amélioration induite par les phéromones est radicale, en particulier pour le début de la collecte. La collaboration entre les fourmis permet en effet d'optimiser les déplacements de celles qui sont en mode recherche.

La figure 9 illustre comment les phéromones assurent une meilleure efficacité globale. Elle correspond à l'état du monde après 500 unités de temps. Les phéromones sont représentées par les couleurs : plus la case est orange plus elle contient de phéromones. On constate tout d'abord que les phéromones forment bien des routes vers les quatre zones de nourriture. De plus, les fourmis sont presque toutes sur les routes de phéromones, ce qui explique les performances de collecte. En effet, en l'absence de phéromones, les fourmis sont réparties de façon plus uniforme dans le monde. Certaines d'entre elles sont éloignées de toute nourriture et risquent de devoir chercher pendant un temps important une des zones intéressantes. En

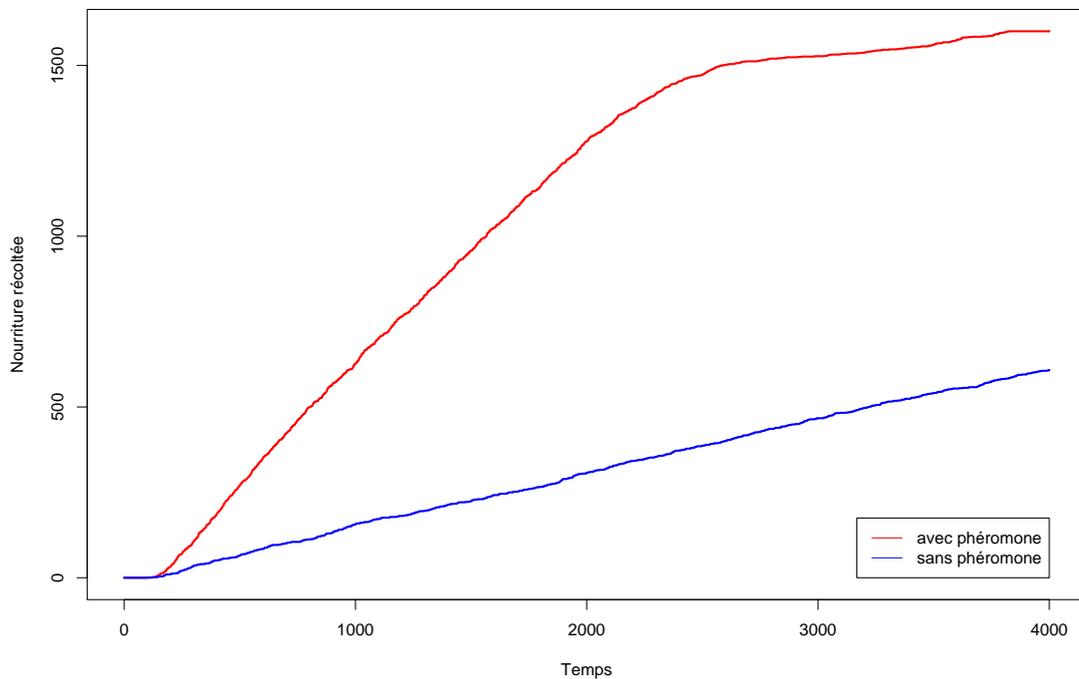


FIG. 8 – Collecte de nourriture : avec ou sans phéromones

présence de phéromones, au contraire, après 500 unités de temps, presque toutes les fourmis ont été "recrutées". Si elles ne sont pas en mode *retour à la fourmilière*, elles sont en train de suivre une piste de phéromone et retournent donc efficacement vers une zone de nourriture.

On peut expliquer le tassement en performances autour de 2000 unités de temps par le sur-recrutement de fourmis pour le tas le plus proche de la fourmilière. En effet, comme ce tas est proche, il est découvert plus vite. De ce fait, les pistes de phéromones qui mènent à ce tas sont plus rapidement mise en place que celles qui mènent aux autres tas. Beaucoup de fourmis sont donc recrutées pour aller collecter la nourriture du tas. Malheureusement, l'algorithme de suivi des pistes est très élémentaire et quand le tas s'épuise, les fourmis restent bloquées à l'extrémité de la piste. On observe ce phénomène sur la figure 10, à la fois pour le tas le plus proche (après 2000 unités de temps) et pour les autres tas (après 3000 unités de temps).

Deux solutions sont envisageables pour améliorer la situation. On peut soit réduire la durée de vie des phéromones, au risque de perdre les effets bénéfiques de celles-ci, soit améliorer l'algorithme de suivi des pistes. Dans les exemples présentés au dessus, une goutte de phéromones persiste 110 unités de temps pour la zone centrale, beaucoup moins sur les bords (la goutte occupe 8 cases), à savoir 14 unités de temps. Si on réduit la durée de vie à 75 unités de temps (9 pour les bords de la goutte), on améliore notablement les résultats, comme l'illustre la figure 11. Sur cette courbe, les phéromones lentes correspondent à une durée de vie de 450 unités de temps pour une goutte (28 sur les bords). Si on continue à réduire la durée de vie, par exemple en passant à 56 unités de temps, les performances commencent à se dégrader.

Le problème principal de la solution par réglage de l'évaporation des phéromones est que, dans la réalité, les fourmis ne restent pas coincées à la fin d'une piste ! Quand elles se

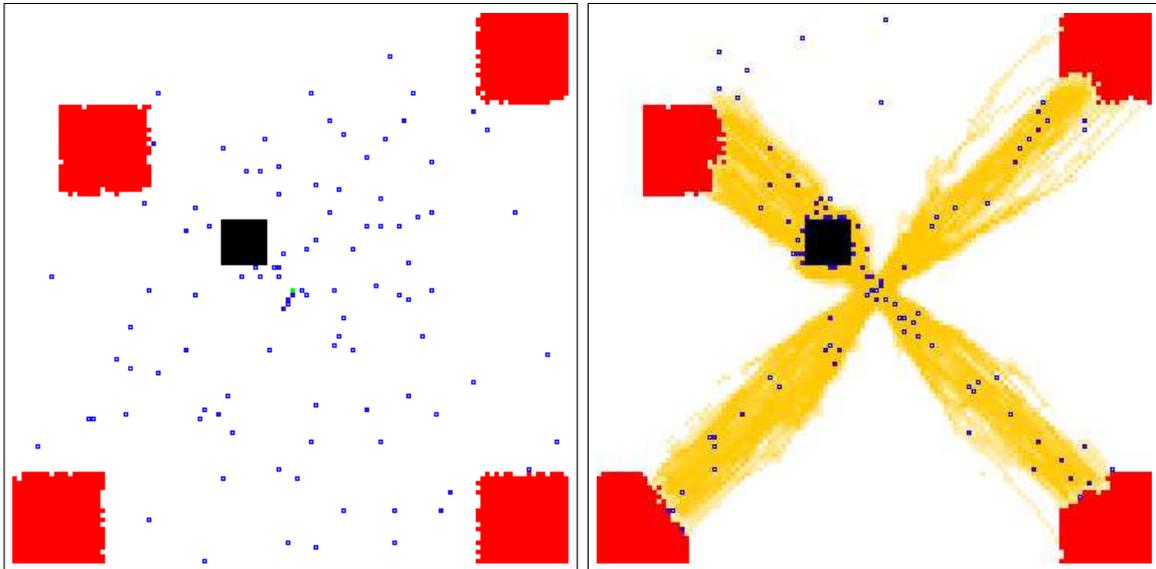


FIG. 9 – Déplacements sans phéromone (à gauche) ou avec (à droite)

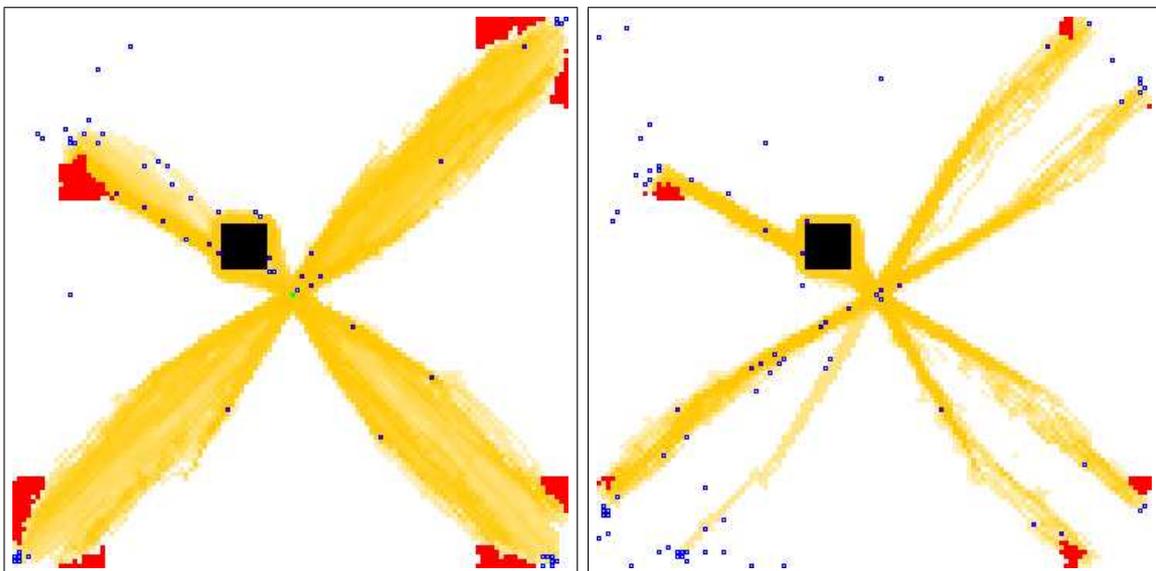


FIG. 10 – Evolution du monde : après 2 000 unités de temps à gauche, puis après 3 000 unités de temps à droite

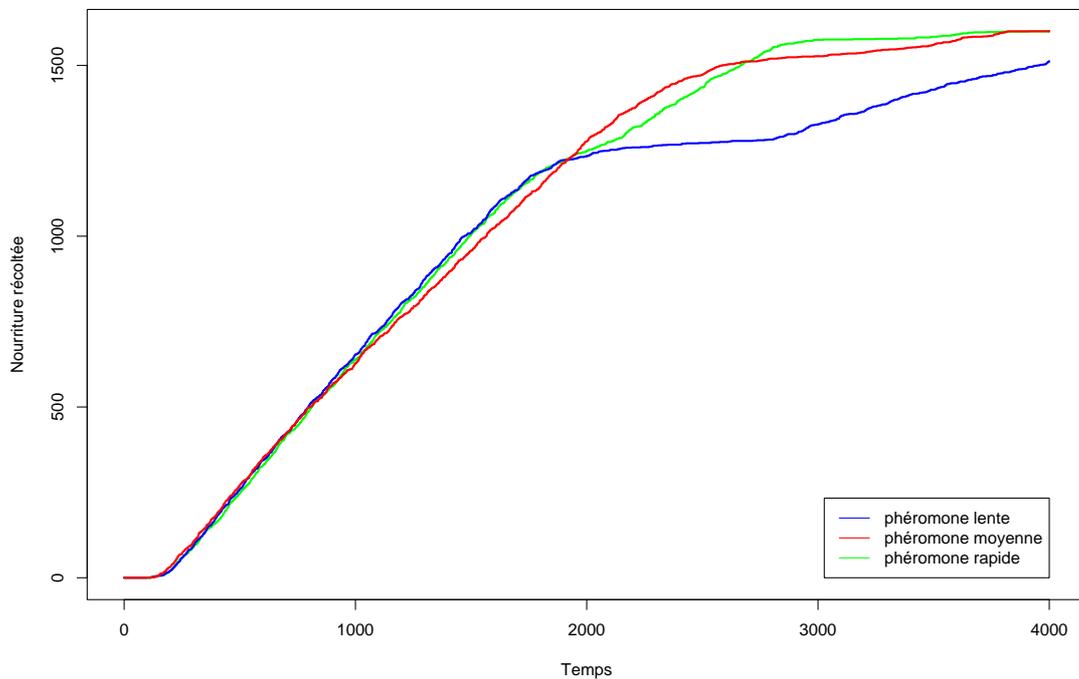


FIG. 11 – Persistance des phéromones

retrouvent dans une telle situation, elles reprennent en général une recherche classique, sans tenir compte des phéromones pendant un certain temps (ce qui revient à s'éloigner de la piste avant de prendre de nouveau en compte les phéromones). On peut donc proposer l'algorithme suivant, quand la fourmi est en mode *recherche* :

1. si la fourmi est réfractaire aux phéromones depuis plus de 50 unités de temps, redevenir sensible aux phéromones
2. si au moins une des 8 cases voisines contient de la nourriture :
  - (a) déterminer les cases contenant le plus de nourriture
  - (b) se tourner aléatoirement vers une de ces cases
  - (c) choisir la prochaine direction au hasard à partir de la direction actuelle
  - (d) avancer d'un pas
3. sinon, si au moins une des 3 cases situées devant la fourmi contient des phéromones :
  - (a) déterminer, parmi les trois cases, celle qui contient des phéromones et qui est la plus éloignée de la fourmilière
  - (b) en cas d'ex-aequo, choisir la case contenant le plus de phéromones
  - (c) se tourner vers la case choisie
  - (d) avancer d'un pas
4. sinon, devenir réfractaire aux phéromones et avancer comme dans le mode recherche classique

5. si la case atteinte contient de la nourriture :
  - (a) ramasser la nourriture
  - (b) passer en mode *retour à la fourmilière*
6. sinon, retourner en 1

L'algorithme est beaucoup plus évolué que le suivi standard de phéromones, mais conserve les caractéristiques importantes de ce dernier : la localité et l'absence de communications directes. La figure 12 compare les phéromones volatiles avec le suivi efficace (en présence de phéromones à durée de vie moyenne). On constate que les deux solutions sont comparables. Comme la solution avec suivi efficace est plus conforme à la réalité, nous la retenons pour la suite de l'article.

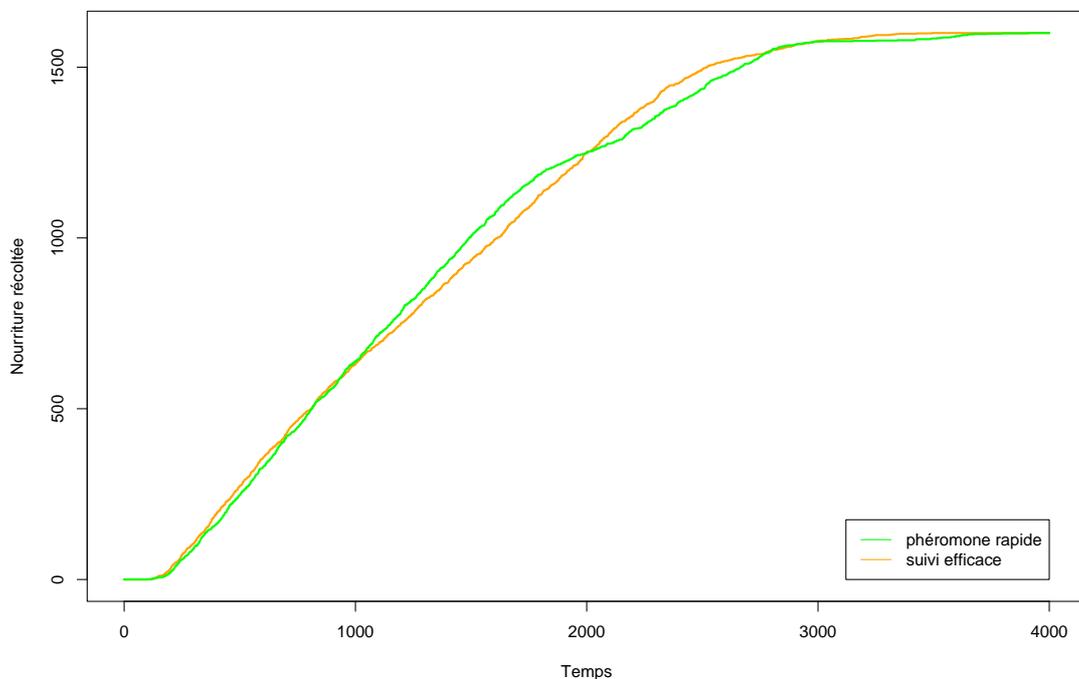


FIG. 12 – Suivi efficace

## 5 L'intelligence individuelle

### 5.1 Mémorisation

Les fourmis étant capables de mémoriser la direction de leur fourmilière, il est naturel de penser qu'une fourmi doit aussi être capable de mémoriser la direction de la dernière zone de nourriture qu'elle a découverte. On augmente de cette façon l'intelligence individuelle de chaque fourmi. L'algorithme de la fourmi en mode *recherche* devient alors le suivant :

1. si la fourmi se souvient d'un emplacement de nourriture :
  - (a) se tourner vers cet emplacement

- (b) si l'emplacement mémorisé est atteint, l'oublier
- 2. sinon :
  - (a) choisir la prochaine direction au hasard
  - (b) s'orienter dans cette direction
  - (c) avancer d'un pas
- 3. si la case atteinte contient de la nourriture :
  - (a) ramasser la nourriture
  - (b) mémoriser l'emplacement de la nourriture
  - (c) passer en mode *retour à la fourmilière*
- 4. sinon, retourner en 1

L'idée est donc simplement pour une fourmi de revenir dans une zone dans laquelle elle a découvert de la nourriture. Quand la fourmi est revenue dans cette zone, elle reprend une recherche aléatoire classique.

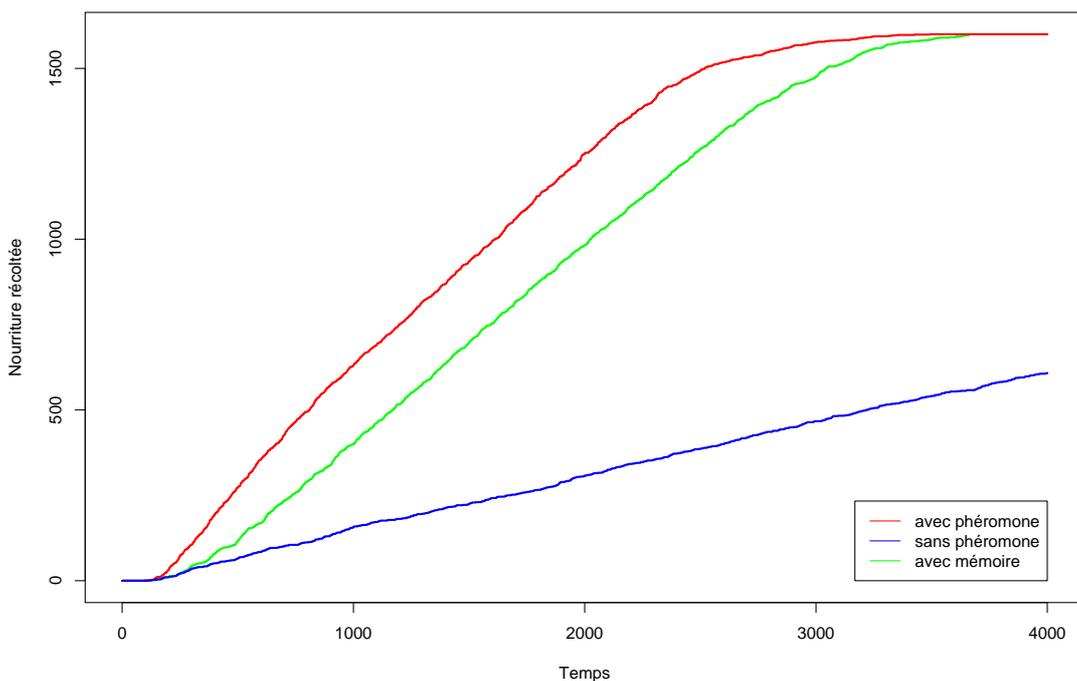


FIG. 13 – Influence de la mémorisation

On constate sur la figure 13 l'efficacité de cette stratégie individuelle. Cependant, la méthode reste moins efficace que l'intelligence collective obtenue grâce aux phéromones. En particulier, la collecte complète de nourriture se fait en 3493 unités de temps quand on utilise des phéromones, contre 3665 avec la mémorisation. La différence entre les deux intelligences réside essentiellement dans la phase de recrutement. En effet, les pentes des deux courbes sont comparables, ce qui signifie que la mémoire collective construite grâce aux phéromones

remplace sans trop de difficulté la mémoire individuelle. Par contre, le recrutement des fourmis par les pistes de phéromones n'existe pas en cas d'intelligence individuelle et les fourmis mettent donc plus de temps à découvrir leur première source de nourriture. La figure 14 donne l'état du monde après 500 unités de temps pour les deux intelligences (suivi efficace dans le cas de la phéromone) et confirme cette interprétation : il y a plus de fourmis "perdues" dans le cas de la mémoire individuelle.

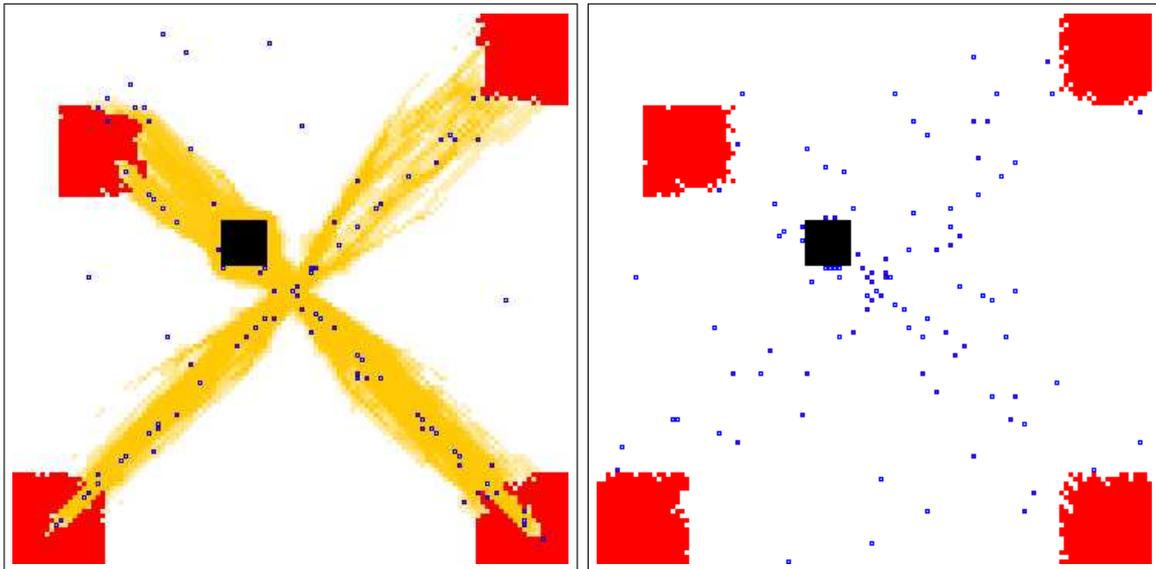


FIG. 14 – Déplacements avec phéromone (à gauche) ou avec mémoire (à droite)

## 5.2 Phéromones

Il est tentant de combiner l'utilisation des phéromones (qui favorisent le recrutement) avec la mémoire individuelle (il n'y a pas mieux pour revenir à un tas déjà découvert). Cependant, la figure 15 montre que l'amélioration obtenue est marginale par rapport à l'utilisation seule des phéromones. On accélère la collecte, puisque toute la nourriture est obtenue en 3044 unités de temps (contre 3493), mais cette accélération ne concerne que les toutes dernières unités de nourriture collectées.

Par contre, l'amélioration par rapport à l'utilisation seule de la mémoire individuelle est importante, ce qui confirme une fois de plus l'efficacité des phéromones pour le recrutement des fourmis.

## 5.3 Un monde plus complexe

Quand le monde est plus complexe, en particulier quand le nombre d'obstacles augmente, la situation devient en général moins favorable pour l'intelligence collective, car les chemins de retour vers la nourriture peuvent devenir assez tortueux et les phéromones insuffisantes pour résoudre le problème. Le monde de la figure 16 est un bon exemple de ce type de situations délicates.

Malgré la difficulté de la tâche, l'utilisation des phéromones donne des résultats très satisfaisants. Vers le milieu de la collecte, l'intelligence individuelle domine, vraisemblablement

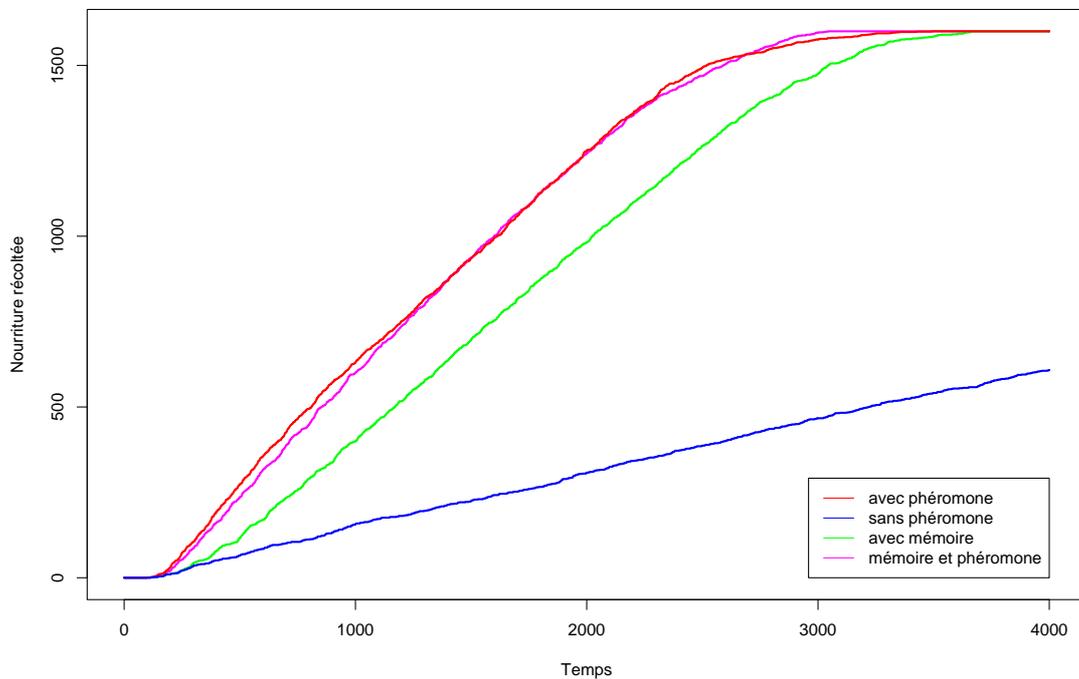


FIG. 15 – Combinaison de méthodes

pour des raisons liées à la complexité des chemins menant de la fourmilière aux zones de nourriture. Sur la fin, les phéromones reprennent le dessus. Comme l'illustre la figure 17, la méthode combinée donne cette fois les meilleurs résultats.

## 6 Conclusion

Le modèle élémentaire de collecte de nourriture présenté dans cet article montre qu'avec un concept très simple, celui de marques (les phéromones) laissées sur une mémoire collective (le sol), la colonie de fourmis remplace avantageusement une mémoire individuelle (retour vers la zone de nourriture) et une coordination centralisée (recrutement de nouvelles fourmis).

Pour obtenir un tel comportement de façon centralisée, il faudrait supposer que les fourmis sont capables d'explorer le monde puis de rapporter à une autorité centrale (la reine par exemple) les informations issues de l'exploration (une sorte de carte de la région explorée). La reine pourrait alors combiner les cartes, puis calculer les chemins optimaux depuis la fourmilière vers les zones de nourriture. Elle planifierait ensuite la collecte de façon optimale en envoyant les fourmis vers les tas de nourriture. Chaque fourmi devrait être capable de suivre le plan établi par la reine. Bien entendu, tout ceci relève de la pure fiction et les travaux de Jean-Louis Deneubourg [1] ont montré que les fourmis utilisent en réalité des mécanismes très proches de ceux que nous avons décrits dans l'article.

En algorithmique en général, et en IA en particulier, nous avons cependant comme réflexe naturel de mettre au point des solutions centralisées très proches (dans leur esprit) de notre fourmilière futuriste avec sa reine omnisciente. Les résultats obtenus ne sont pas toujours très

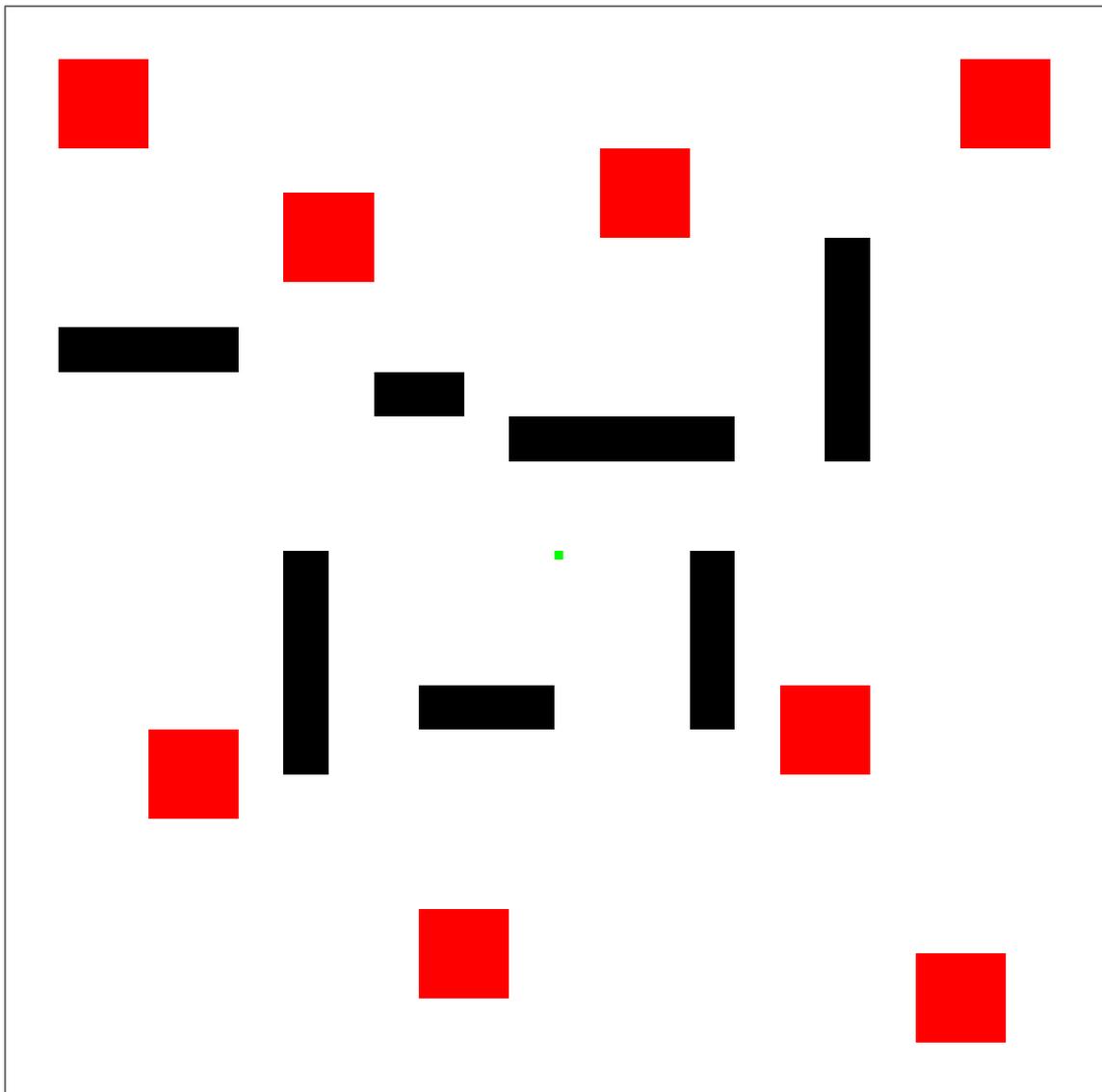


FIG. 16 – Un monde plus complexe

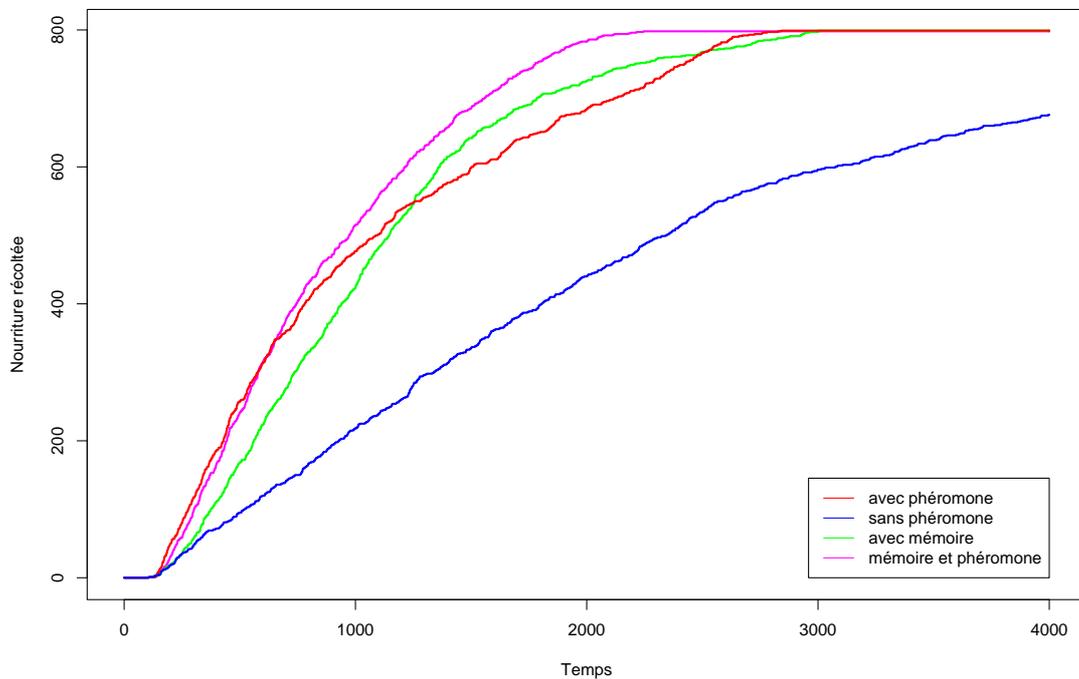


FIG. 17 – Combinaison de méthodes

satisfaisants, c'est pourquoi, comme nous le disions en introduction, des chercheurs ont adapté les méthodes des insectes sociaux pour résoudre des problèmes concrets. L'équipe de Marco Dorigo a ainsi proposé un algorithme d'optimisation qui utilise des fourmis et des phéromones pour résoudre des problèmes d'optimisation combinatoire. Les principes de l'algorithme sont très proches de ceux mis en oeuvre dans cet article : les fourmis explorent une sorte de monde qui représente le problème, en laissant des traces de phéromones. Les traces vont peu à peu faire apparaître par renforcement une solution optimale.

L'intelligence collective des colonies d'insectes offre donc un modèle alternatif pour la construction de programmes d'intelligence artificielle. Les algorithmes obtenus sont intéressants à plus d'un titre car ils sont en général assez simples, décentralisés et intrinsèquement parallèles. Le domaine de l'intelligence collective, comme celui des systèmes multi-agents qui en est très proche, est en développement constant depuis une dizaine d'années et est devenu, au fil du temps, une source de solutions crédibles pour des problèmes réels.

## Références

- [1] Jean-Louis Deneubourg. Individuellement, les insectes sont bêtes collectivement, ils sont intelligents... *Le Temps stratégique*, 65, Septembre 1995. <http://www.archipress.org/ts/deneubourg.htm>.
- [2] Eric Bonabeau and Guy Théraulaz. L'intelligence en essaim. *Pour La Science*, 271, Mai 2000.

- [3] Marco Dorigo, Vittorio Maniezzo, and Alberto Coloni. Positive feedback as a search strategy. Technical Report 91-016, Politecnico di Milano, Milan, Italie, Juin 1991. <ftp://iridia.ulb.ac.be/pub/mdorigo/tec.reps/TR.01-ANTS-91-016.ps.gz>.
- [4] Site du projet ants (évaluation comparative de modèles de cognition auto-organisée dans des systèmes naturels et artificiels). [http://www.irit.fr/ACTIVITES/EQ\\_SMI/SMAC/PROJETS/Projet\\_ANTS.html](http://www.irit.fr/ACTIVITES/EQ_SMI/SMAC/PROJETS/Projet_ANTS.html).

## **A Licence**

Cette création est mise à disposition selon le Contrat Paternité - Pas d'Utilisation Commerciale - Pas de Modification disponible en ligne <http://creativecommons.org/licenses/by-nc-nd/2.0/fr/> ou par courrier postal à Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

## **B Historique**

23 Septembre 2006 Mise en ligne avec quelques corrections par rapport au texte d'origine  
(merci à Nicolas Schoonbroodt pour ses remarques)  
Juin 2003 Publication par le GNU/Linux Magazine France