

Robotique Mobile - CM1 : robotique collective

Nicolas Monmarché

Département Informatique – Polytech Tours
nicolas.monmarche@univ-tours.fr

24 janvier 2017



Introduction

Plan de l'enseignement (rappel)

- ① Introduction à la Robotique collective
- ② Architecture de robots mobiles
- ③ Capacité de perception
- ④ Planification de trajectoire
- ⑤ Architectures de contrôle

Introduction

Plan

- motivations pour la robotique collective
- exemples de mises en œuvre

Avantages attendus :

- **performance** : une communauté de robots coopératifs est plus performante (en termes de vitesse et d'efficacité) que le même nombre de robots agissants de manière individuelle
- **robustesse** : un ensemble de robots simples est plus robuste qu'un seul robot très élaboré
- **coût** : un ensemble de plusieurs robots identiques est moins coûteux à concevoir et à produire qu'un seul robot ayant la même efficacité
- **adaptation** : la répartition d'une colonie de robots peut s'ajuster dynamiquement au problème traité (par exemple à la densité des objets recherchés)

Problèmes envisageables :

- **interférences** : la communication entre les robots est source de difficultés techniques
- **coût des communications** : communiquer consomme de l'énergie et du temps de calcul, en robotique mobile cela peut-être crucial
- **coopération/compétition** : la frontière peut-être floue entre ces deux comportements au niveau du groupe de robots

Définir un système robotique collectif (*swarm robotics*) :

- plusieurs entités autonomes (mobiles ou non)
- absence de contrôle centralisé : chaque robot prend ses décisions en fonction de ces perceptions
- absence de communication globale : uniquement des interactions locales

Motivations

Sources d'inspiration :

- organisations humaines
- systèmes multi-agents
- systèmes naturels (groupes d'animaux : colonies d'insectes...) ou bioinspirés (réseaux de neurones, algorithmes évolutionnaires)

Exemples

Points abordés par les exemples :

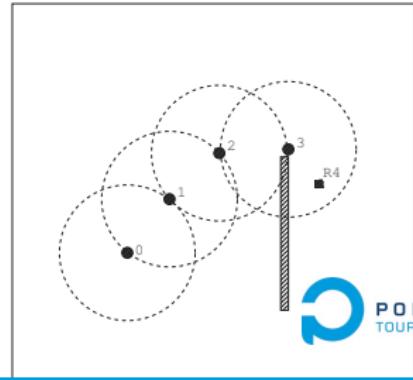
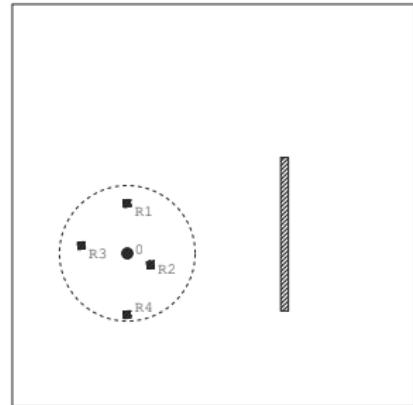
- coordination, communication
- adaptation à un environnement incertain (inconnu et changeant)
- déplacements collectifs

Exemple : communication

Exploration dans un environnement difficile

Goss et Deneubourg (1991)

- objectif : ramasser des objets et les ramener au point central
- contraintes : environnement inconnu, obstacles possibles, communication limitée, pas de carte ni vision
- méthode : mvt aléatoire, les robots se transforment en balises en limite de perception du signal
- expérimentations : simulation + robot khepera (Krieger, Billeter, 1999)



Exemple : communication

Exploration dans un environnement difficile

Fabrication de chaînes de robots : (Nouyan, Dorigo 2006)

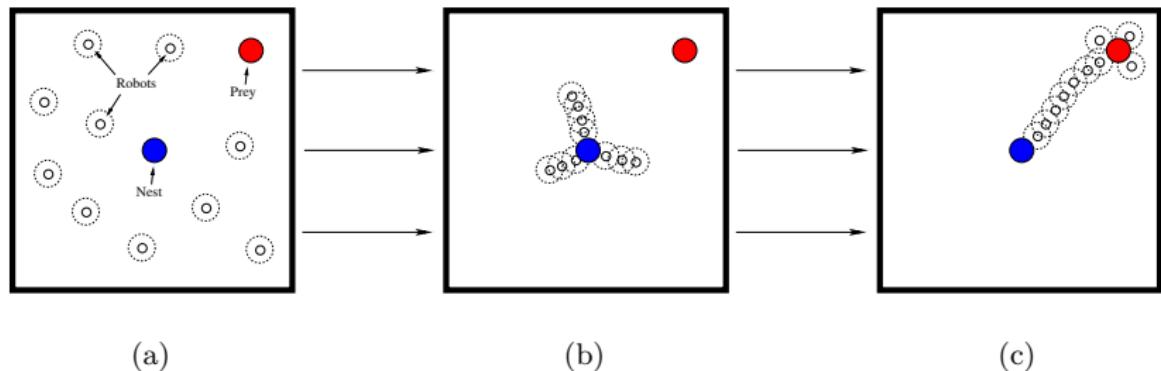


Fig. 1. (a) Initial situation. Robots are indicated by the small white circles. Their limited sensing range is indicated by dashed circles. The task is to form a path between the nest and the prey. (b) The robots search for the nest and once they find it they start self-organizing into randomly oriented chains. (c) When a chain perceives the prey a path is formed.

Exemple : communication

Exploration dans un environnement difficile

Fabrication de chaînes de robots : (Nouyan, Dorigo 2006)

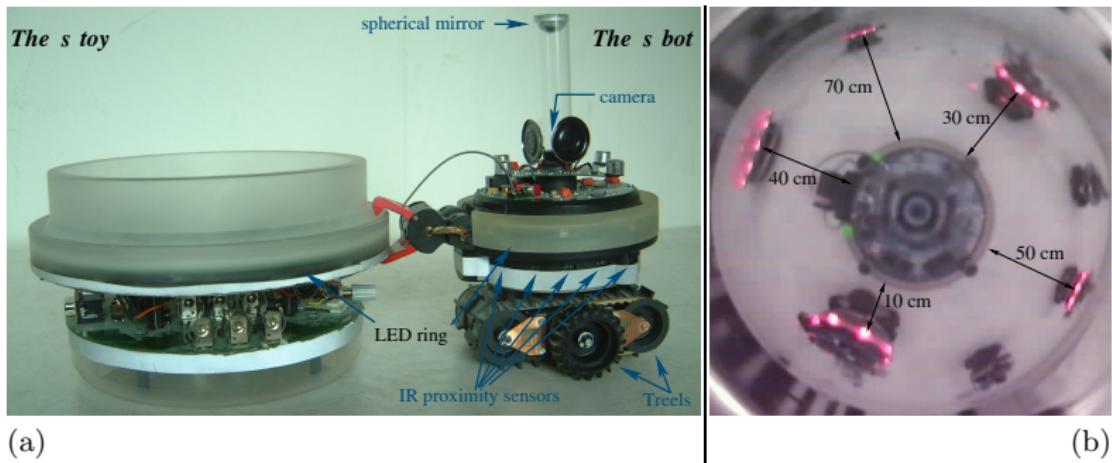
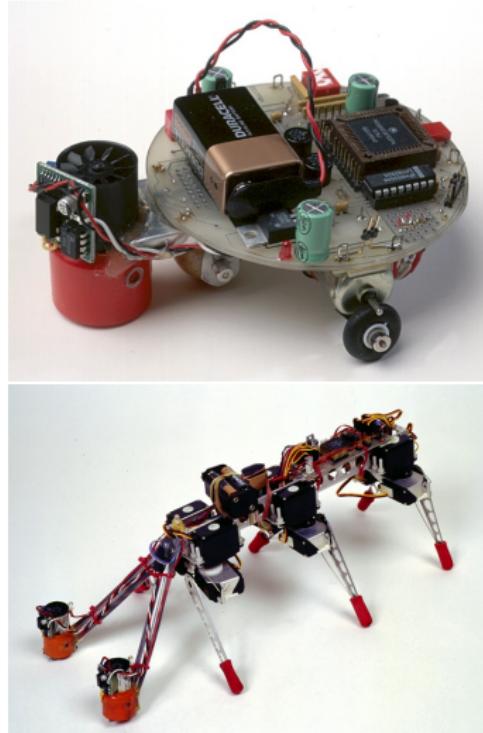
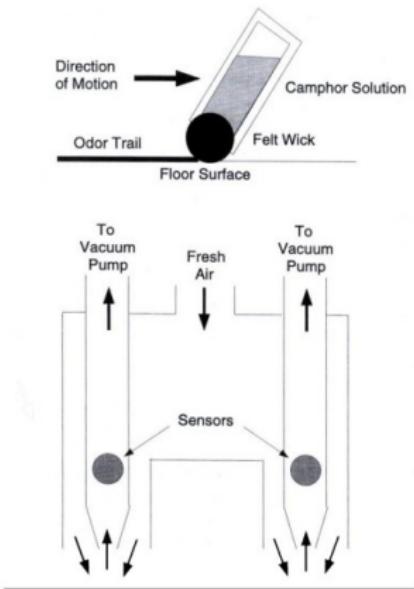


Fig. 2. The hardware. (a) The *s-toy* and the *s-bot*. (b) An image taken with the omnidirectional camera of the *s-bot*. It shows other *s-bots* and an *s-toy* activating their red LEDs at various distances.

Exemple : communication

Communication indirecte : Utilisation de phéromones (camphre)

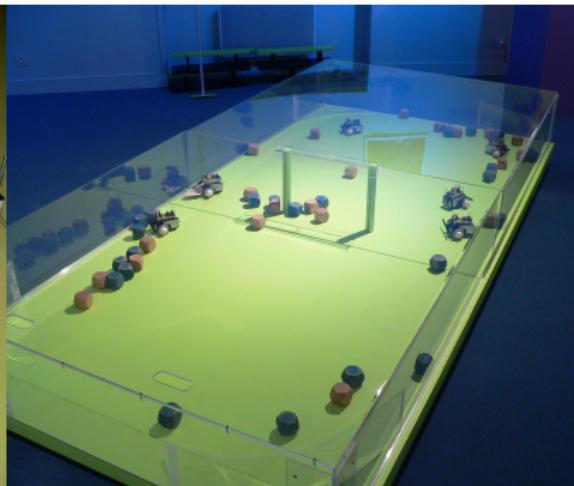
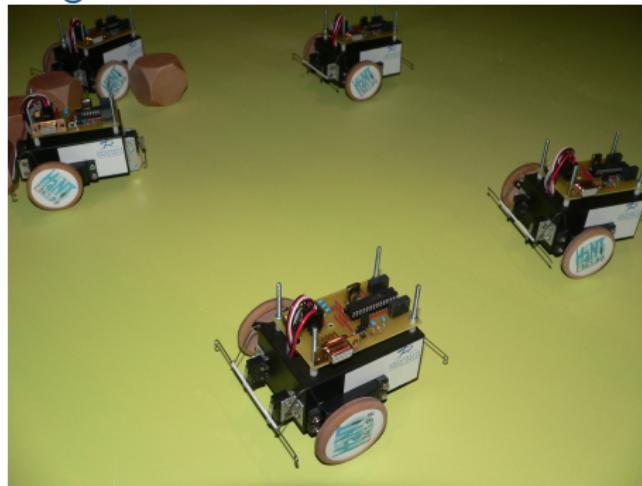


A. Russell, Thiel D., and A. Mackay-Sim. Sensing odour trails for mobile robot navigation. In *IEEE POLYTECH' International Conference on Robotics and Automation*, pages 2672–2677, San Diego, CA, may 1994. A.

Exemple : communication

Communication indirecte : tri collectif par une colonie d'Arabots

La communication peut se faire par la modification de l'environnement à travers l'action même des robots. Ici, il s'agit de ranger des cubes.



Exemple : communication

Communication indirecte : tri collectif par une colonie d'Arabots

Algorithme simple (point de vue du robot) :

- tant que possible, j'avance
- si je ne détecte plus de mouvement par rapport au sol alors reculer un peu et changer de direction au hasard

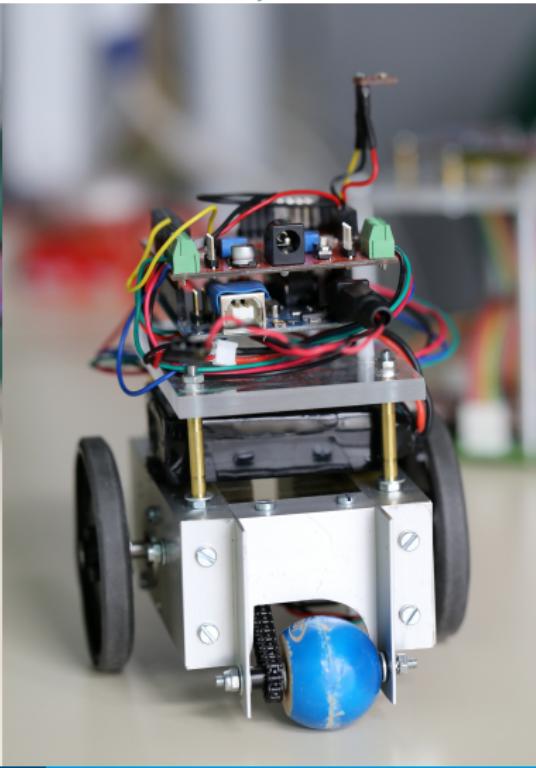
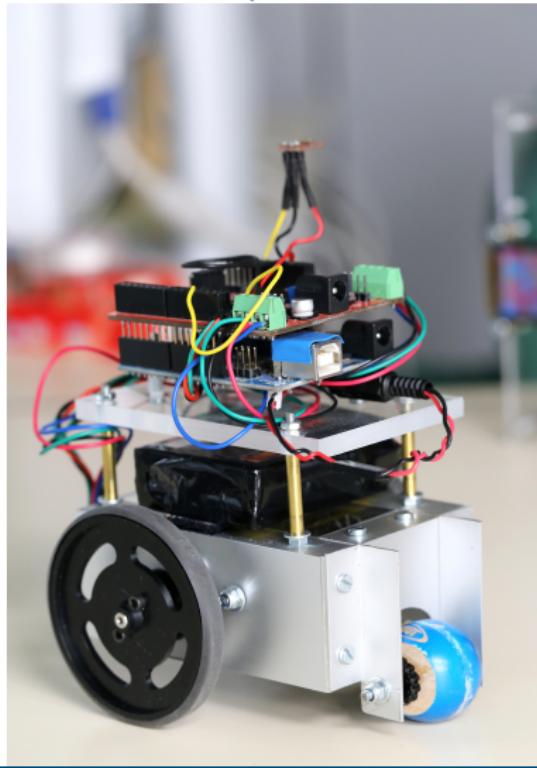
Propriétés :

- pas de différence entre le contact avec un autre robot, le bord et un tas d'objets trop lourd à déplacer
- finalement, au bout d'un certain temps tous les cubes sont « rangés »
- si un robot tombe en panne, le travail continue : robustesse

Exemple : communication

Communication indirecte : tri collectif par une colonie d'Arabots

Version 2013 (moteurs pas-à-pas + odométrie) :



Exemple

Transport collectif

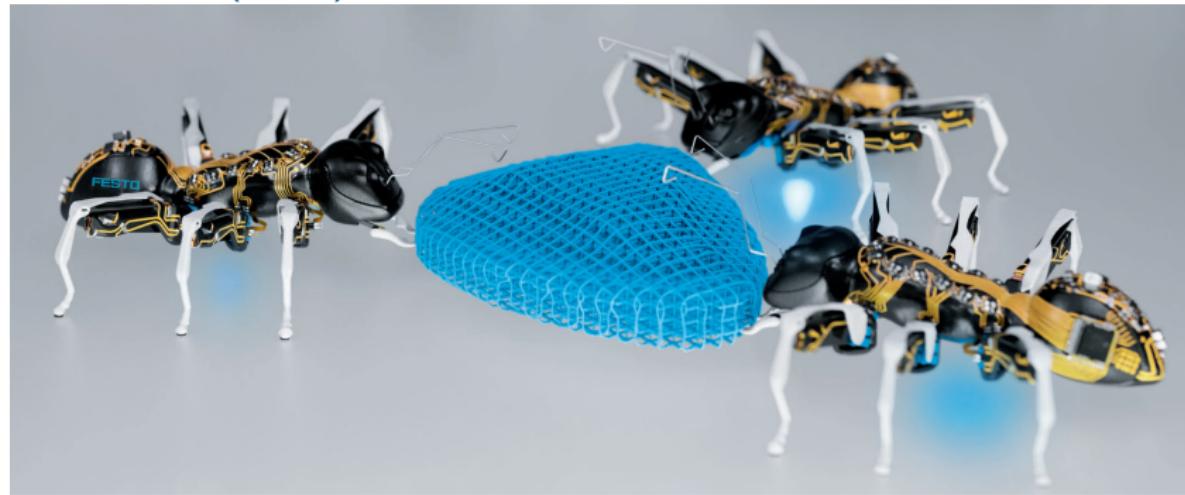
Principe :

- chaque robot perçoit un **renforcement** de son action (tenter de bouger dans une direction et abandonner si cela ne provoque pas de mouvement)
- la communication est **indirecte** : c'est le mouvement de l'objet qui représente le message
- si les robots ont des actions contradictoires : l'objet ne bouge pas \Rightarrow pas de renforcement, le robot change d'action en introduisant du **hasard**

Exemple

Transport collectif

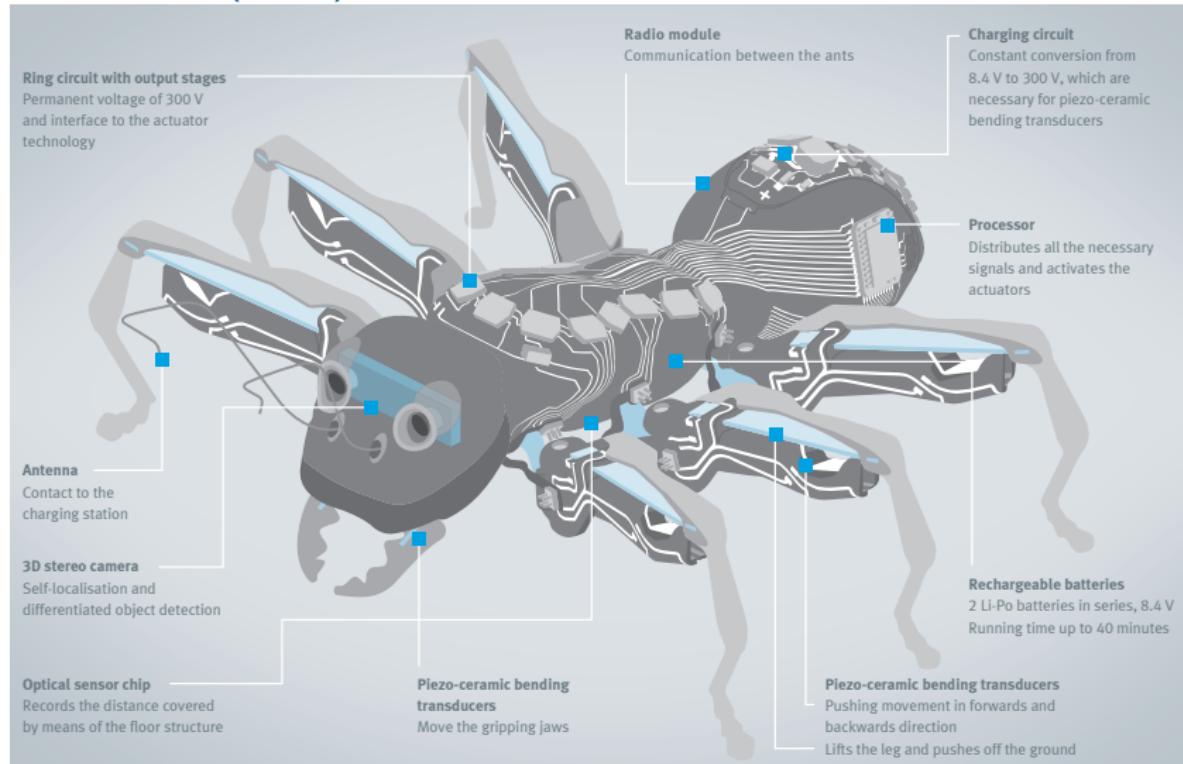
BionicANTs (Festo)



Exemple

Transport collectif

BionicANTs (Festo)



Exemple

Rapprochement de plaques

Modèle biologique : les fourmis tisserandes (*Weaver ants*) :



Rajan Chameer

Exemple

Rapprochement de plaques

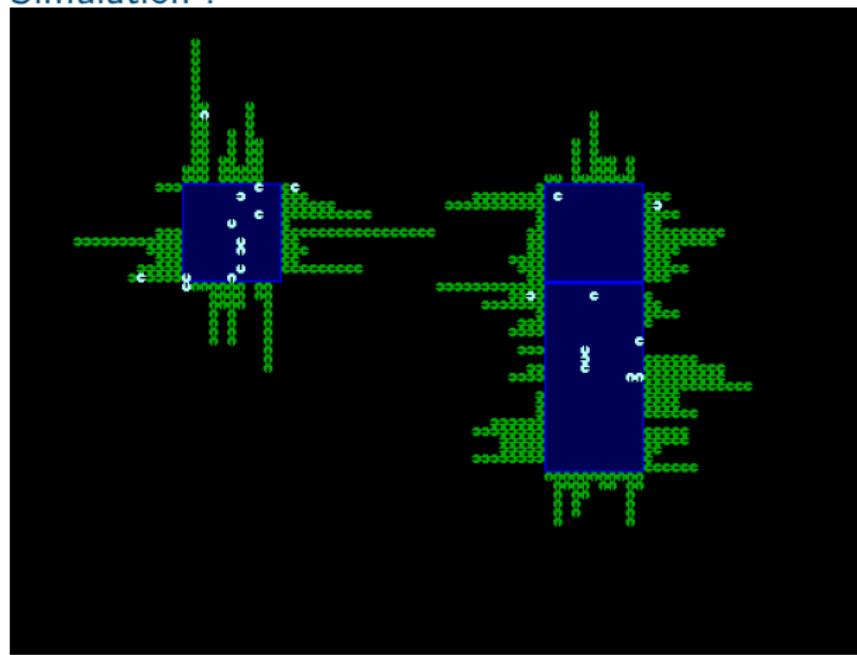
Modélisation :

- chaque robot se déplace au hasard sur la feuille
- lorsqu'il trouve une chaîne de robots, il a une probabilité p_1 de s'y engager
- lorsqu'un robot atteint le bout d'une chaîne, il s'y accroche avec une probabilité p_2
- à chaque pas de temps, le robot en bout de chaîne a une probabilité p_3 d'abandonner et de retourner sur la feuille en remontant la chaîne
- si un robot perçoit qu'un des robots à l'une de ses extrémités abandonne la chaîne, il l'abandonne à son tour
- si un robot qui arrive en bout de chaîne se connecte des deux côtés, il se met à tirer
- si un robot en milieu de chaîne sent une traction, il se met à son tour à tirer

Exemple

Rapprochement de plaques

Simulation :



Exemple

Coopération de robots

Morphogénèse (SwarmBot)



Exemple

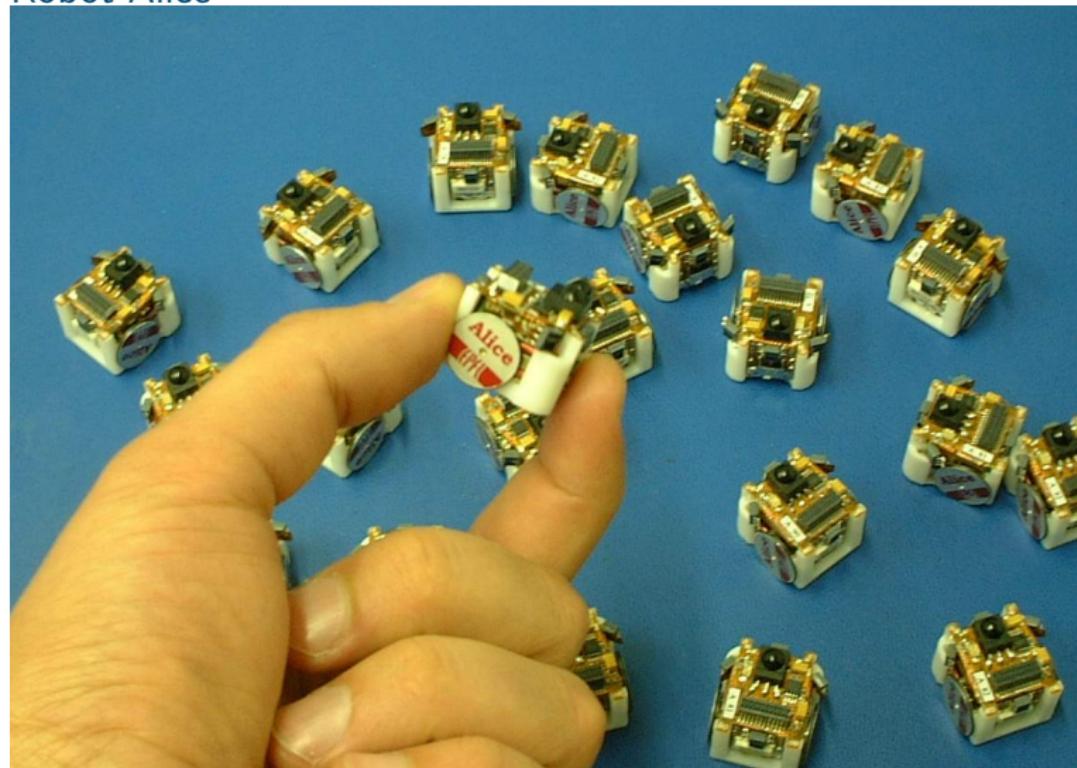
Collaboration/spécialisation

Projet Swarmanoid

Exemple

Grandes colonies

Robot Alice



Exemple

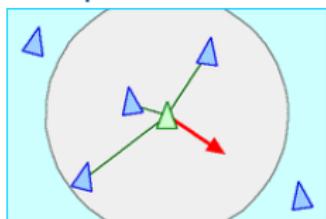
Apprentissage de comportements distribués

Algorithme évolutionnaires distribués

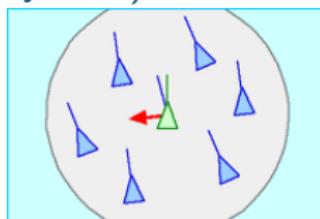
Exemple

Déplacements collectifs

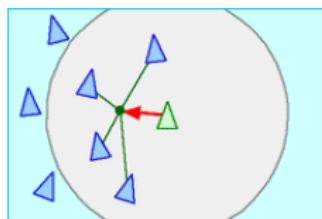
Principes des Boids (Reynolds)



séparation
le boids évite la collision avec ses voisins



alignement
le boids aligne sa vitesse sur la vitesse moyenne de ses voisins



cohésion
le boids se rapproche du centre de gravité de ses voisins

Exemple

Déplacements collectifs

Simulations :



Exemple

Déplacements collectifs

- déplacement en groupe
- construction collective

⇒Colonies de quadricoptères

Exemple

Colonie de robots volants

eMotionButterflies (Festo)

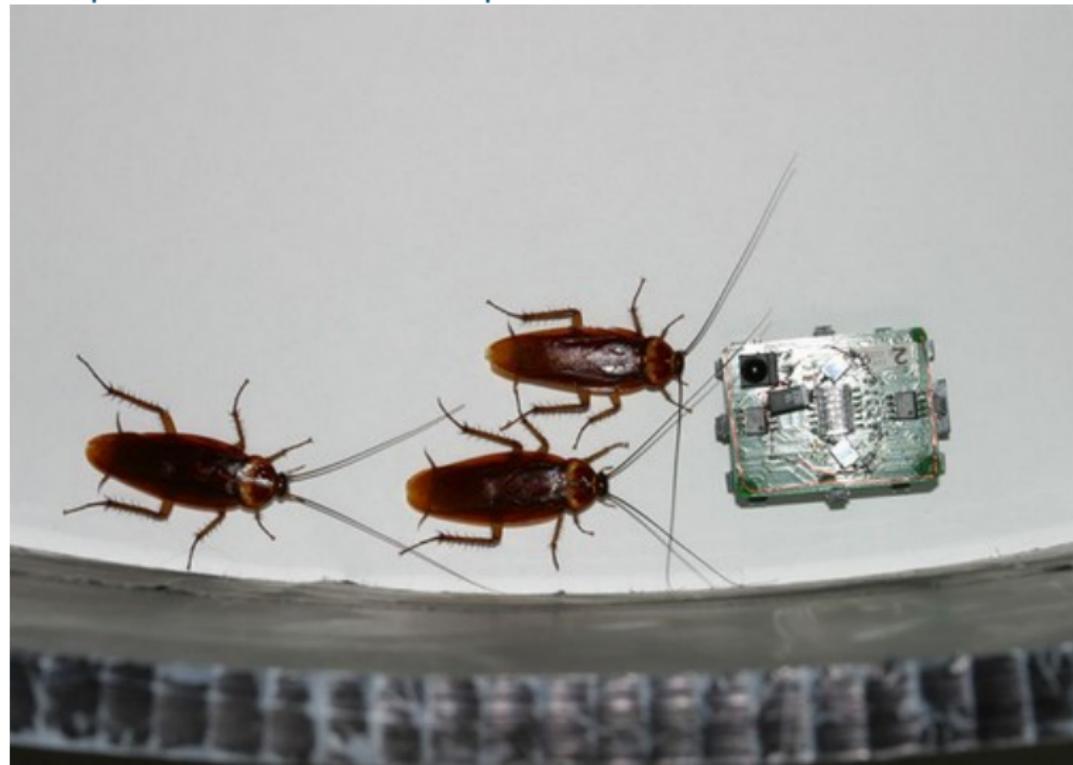


Coordinated flying thanks to indoor GPS

Exemple

Adaptation à un environnement incertain

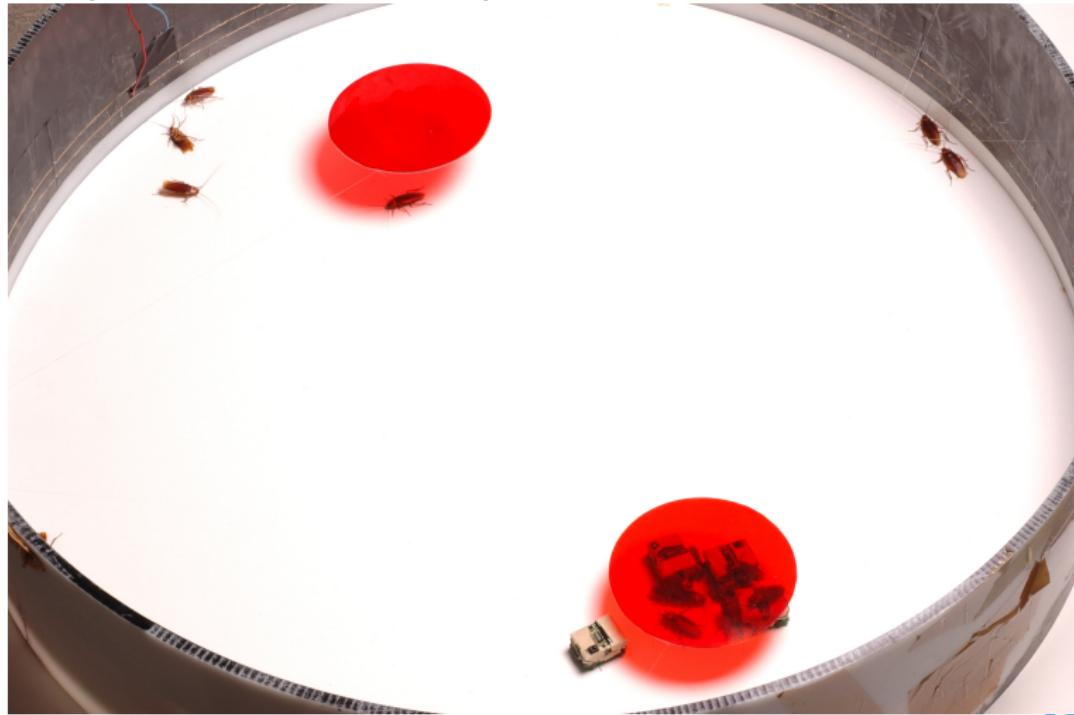
Comportement des blattes qui se cachent...



Exemple

Adaptation à un environnement incertain

Comportement des blattes qui se cachent...

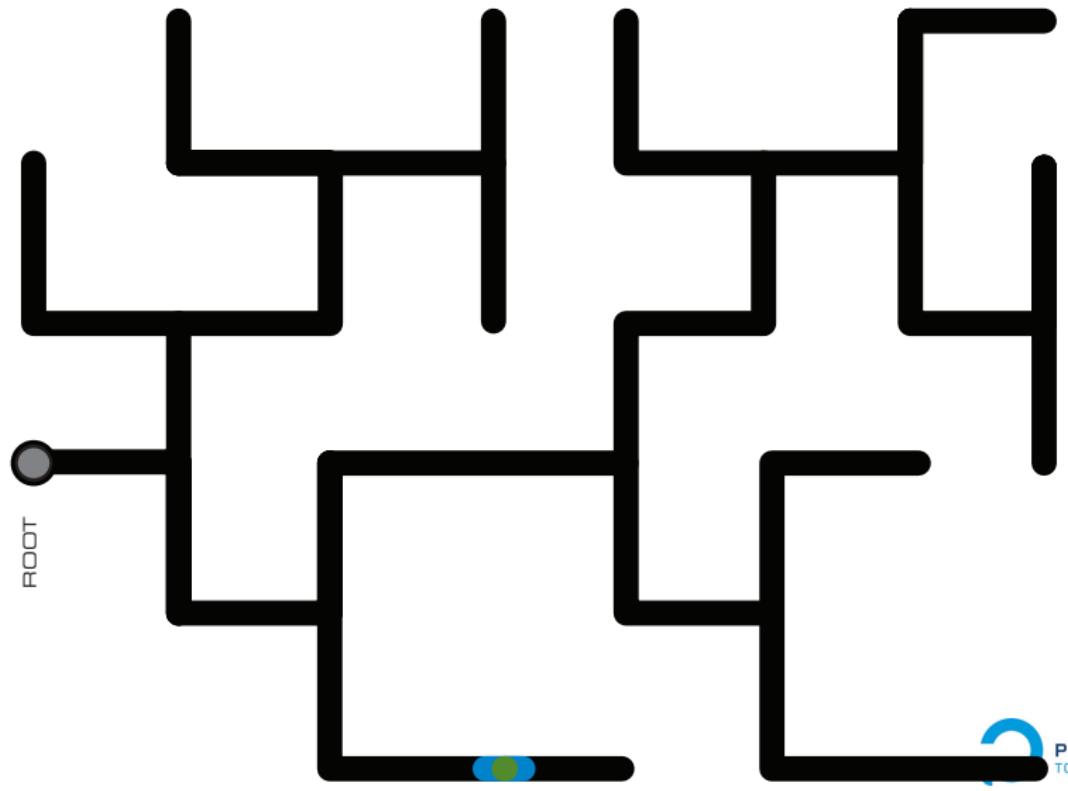


Exemple

Robots collectifs artistiques

Exemple

Miniaturisation/éducation : l'exemple Ozobot



TREE MAZE

Exemple

Miniaturisation/éducation : l'exemple Ozobot

The screenshot shows the OzoBlockly interface with a program for an Ozobot. The program starts by setting a variable `leafCount` to 0. It then enters a loop that repeats 10 times. Inside the loop, the Ozobot follows a line to the next intersection or line end, changes `leafCount` by 1, and then performs a searchSubtree function. This searchSubtree function checks if there is a "line end". If yes, it picks a "back" direction, follows the line to the next intersection or line end, and then picks a "right" direction. It then calls the searchSubtree function again. If there is no "line end", it picks a "straight" direction, calls the searchSubtree function again, picks a "right" direction, and then follows the line to the next intersection or line end. Finally, the program terminates and turns the Ozobot off.

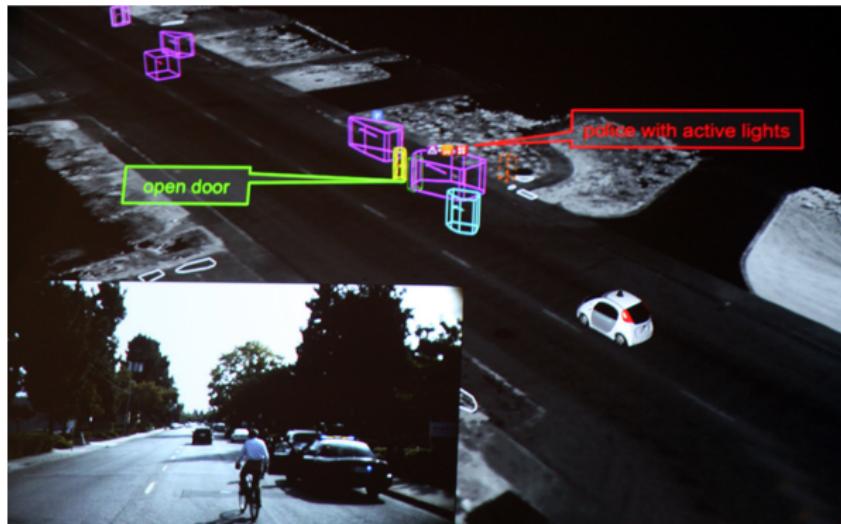
JavaScript Preview

Real-time preview of your OzoBlockly program using JS syntax.

```
var leafCount;  
  
// Depth First traversal in a binary tree  
function searchSubtree() {  
    followLine();  
    if (isThereLineEnd()) {  
        // Robot keeps track of visited leaves (line ends)  
        leafCount = (typeof leafCount == 'number' ?  
leafCount : 0) + 1;  
        pickDirection(BACKWARD);  
        followLine();  
    } else {  
        pickDirection(RIGHT);  
        // recursive call  
        searchSubtree();  
        pickDirection(FORWARD);  
        searchSubtree();  
        pickDirection(RIGHT);  
        followLine();  
    }  
  
    leafCount = 0;  
    wait(1);  
    setLEDColorRGB(LED_TOP, 0, 127, 0);  
    searchSubtree();  
    // When the tree traversal is finished, robot  
    signalizes number of leaves in the binary tree.  
    for (var count = 0; count < leafCount; count++) {  
        wait(1);  
        setLEDColorRGB(LED_TOP, 127, 127, 0);  
        delay(50);  
        setLEDColorRGB(LED_TOP, 0, 127, 0);  
    }  
    terminate(EXIT_POWER_DOWN);  
}
```

Conclusion

Google Car : "Nos voitures ne sont pas encore assez humaines" (Chris Urmson)



South by Southwest Festival, 2016 (Morgane Tual, Bernard Monasterolo, LeMonde.fr) « Ce qui leur manque surtout, c'est la compréhension de l'imprévisibilité de l'humain... Et parfois de sa bêtise.

Les ingénieurs ont mis en place des équipes pour simuler des situations absurdes, stupides ou incroyables rencontrées par les voitures, mais il semble que sur ce plan, l'être humain ait une forme d'avance difficilement rattrapable : « Nous avons eu des groupes de jeunes qui traversaient la rue à quatre pattes, en sautant comme des grenouilles, ou encore un jour une femme en chaise roulante qui poursuivait des canards sur la route... »