



REVUE ALGERIENNE DES SCIENCES

Section -A-

- Sciences de la Nature et de la Vie

- Sciences techniques

<http://univ-eltarf.dz/fr/>

Un modèle de planification multi robots base sur les algorithmes génétiques pour la navigation optimale de robots autonomes

Benmachiche Abdelmdjid ^{a,*}, Bouhadada Tahar^{*,b}, Makhlouf Amina^{a,*}, Maatallah Majda ^a^a Département d'Informatique – Université Chadli Bendjedid El-Tarf, BP 73, (36000) – Algérie.^{*} Laboratoire LRI Université Badji Mokhtar–Annaba BP. 12 Annaba – Algérie^b Département d'Informatique – Université Badji Mokhtar–Annaba BP. 12 (23000) – Algérie.

Informations	Résumé
Mots clés : Robotique mobile, Algorithme génétique, évitement d'obstacles. *Correspondance : benmachiche- abdelmdjid@univ-eltarf.dz Reçu le 28 mai 2018, Révisé le 19 juin, Accepté le 20 juin 2018.	Cet article traite de la problématique de la navigation optimale des robots mobiles. Notre travail consiste à déterminer le chemin optimal de plusieurs robots mobiles à l'aide des algorithmes génétiques dans un environnement dynamique. L'évaluation des individus, est une fonction «fitness» basée sur la longueur du chemin. Cette méthode est implémentée et testée plusieurs cas de figures. Les résultats démontrent sa robustesse ainsi que les performances de notre approche.

1. Introduction

Les La robotique évolutionnaire est un domaine de recherche très actif, une vision assez complète des travaux dans ce domaine est présentée dans les articles de D. Floreano et Boots [1, 2, 3]. Le plus court chemin est un problème d'optimisation très important dans la robotique autonome, c'est un thème de recherche dans des laboratoires de haut niveau. Le but est de trouver le plus court chemin en évitant les obstacles pour un robot du point de départ jusqu'au point d'arrivée. Il existe plusieurs méthodes qui traitent ce problème tel que le champ potentiel dans Zhu [4], les réseaux de neurones dans l'article de Yang [5], les algorithmes génétiques [6, 7] présentés dans les articles de Li et Hu [8, 9]. Dans les dernières années, la navigation collective attire de plus en plus l'attention des chercheurs que la navigation d'un seul robot à cause de son efficacité, sa rapidité, sa robustesse et sa capacité à résoudre des tâches complexes. Exemple : En cas d'une catastrophe naturelle (séisme, incendie) on a besoin de planifier les mouvements des ambulances en prenant en compte plusieurs contraintes : suivre le plus court chemin pour atteindre les blessés, la collision inter-

ambulances et l'évitement des obstacles de la route.

2. Définition du problème

Nous nous concentrons sur la situation suivante :

Dans l'état initial il y a une matrice $m \times m$ qui représente l'environnement avec des obstacles placés de façon aléatoire, Le contenu d'une case de la matrice $t[i][j] \in [0,1]$. L'ensemble des robots doit atteindre un point d'arrivée à partir d'un point de départ, chaque robot a une position de départ (X_i, Y_i) et d'arrivée (X_i', Y_i') ou (i) est la position du robot (i) , les robots ne connaissent pas l'environnement, Les robots doivent se synchroniser (éviter les collisions).

Nous nous sommes intéressés à résoudre les problèmes suivants : Trouver la trajectoire optimale (le plus court chemin) et éviter les obstacles. La collision inter robots.

Ce papier est organisé comme suit. Dans la section 1, une introduction au domaine de la robotique. Nous suivons cela avec une Définition du problème. Dans la section 2, la section 3, nous allons présenter toutes Description de la solution proposée. Les résultats expérimentaux sont illustrés dans la section 4, finalement, les

conclusions et les travaux futurs sont formulés dans la section 5.

3. Description de la solution proposée

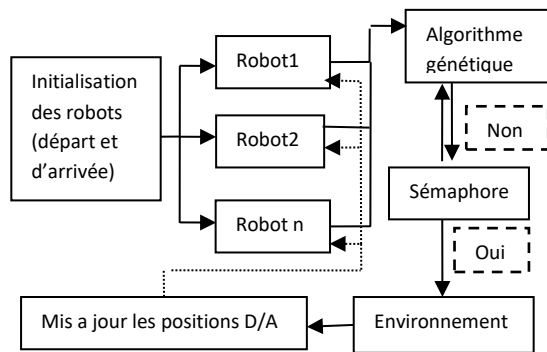


Fig. 1: Architecture générale du système

- 1) le robot se déplace à partir de point de départ suivant les 4 directions au hasard (**Fig. 1**).
- 2) Le robot détecte les obstacles et enregistre les détails de l'environnement
- 3) Trouve un chemin qui relie le point de départ et d'arrivée.
- 4) Redémarre une autre fois pour construire la population initiale.
- 5) Minimise les chemins trouvés.
- 6) Trouve le chemin le plus court.
- 7) Attend la fin de calcul pour les autres robots.
- 8) Exécution parallèle pour les robots.
- 9) Traitement des cas spéciaux (inter-blocage).

3.1 Le module de raisonnement génétique

Le module de raisonnement génétique est le module principal, il définit les mouvements physiques au robot dans son environnement (**Fig. 2**).

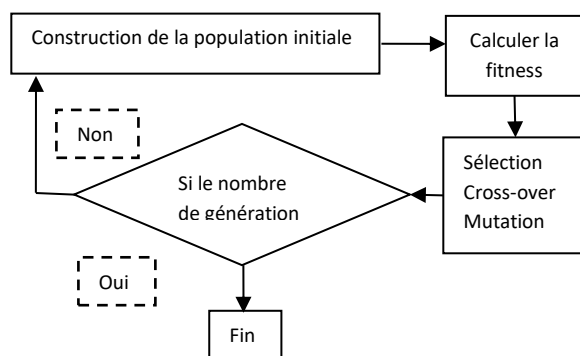


Fig. 2 : Architecture de l'algorithme génétique proposé.

3.1.1 Codage

Un chemin est modélisé sous la forme suivante : $\text{chemin} = \{X_{i1}, X_{i2}, X_{i3}, \dots, X_{in}\}$ où i : est le numéro du robot, l'indice 1..n, où (n) est la longueur du chemin et X un entier $\in \{0, 1, 2, 3\}$.

Exemple :

$\text{chemin} = \{0000133333222211\}$ Où la longueur du chemin $n=15$.

3.1.2 Population initiale

Elle est générée aléatoirement et correspond aux chemins possibles pour atteindre le point d'arrivée. Avoir une solution acceptable (voir table1) évoque d'envisager 100 chemins possibles afin d'aboutir à la destination.

Pour résoudre le problème de convergence prématuré (minimum local) l'algorithme génère un ensemble de solutions (chemins).

La population initiale est présentée par une liste de listes pour donner plus de convivialité car la longueur du chemin trouvé n'est pas toujours constante.

$$\text{Pop-in} = \begin{bmatrix} \text{ch}[1,1], \text{ch}[1,2], \dots, \text{ch}[1,n] \\ \text{ch}[2,1], \text{ch}[2,2], \dots, \text{ch}[2,n] \\ \vdots \\ \text{ch}[x,1], \text{ch}[x,2], \dots, \text{ch}[x,n] \end{bmatrix}$$

Où n est le nombre des chemins de chaque génération et x est une variable qui désigne la longueur du chemin.

3.1.3 Fitness

C'est la somme des distances entre 2 points consécutifs appartenant au chemin.

Telle que :

$$F = \sum \text{dis}(\text{pos}(X_i, Y_i), \text{pos}(X'_i, Y'_i)). \quad (1)$$

Où $\text{pos}(X_i, Y_i)$ est la position courante du robot et $\text{pos}(X'_i, Y'_i)$ est la position suivante. Le résultat de cette fonction est le nombre de déplacements pour joindre les points de départ et d'arrivée, c'est-à-dire la longueur du chemin [11].

3.1.4 Sélection

Le cycle de vie des individus se révèle à chaque génération, sur une population de n individus, 80% y seront conservés, ces derniers peuvent se reproduire (le pourcentage de sélection idéal est entre 70-95%). Pour garder les meilleurs individus de chaque génération, notre algorithme utilise une phase d'élitisme : tirage des chemins selon leur longueur dans un ordre croissant puis on garde 10% de la population courante pour les copier dans la nouvelle génération.

3.1.5 Croisement (cross-over)

Le croisement a pour but d'enrichir la diversité de la population en manipulant la structure des chromosomes. Tout d'abord, les points de rencontre des deux chemins sont déterminés, si les deux chemins ne se rencontrent pas, le fils généré sera identique au père ; sinon on recherche le point de rencontre qui optimise

le trajet : chemin identique à celui du père de départ jusqu'au point de rencontre, puis chemin identique à celui de la mère du point de rencontre à la fin du chemin maternel. On définit de plus la reproduction d'un individu avec lui-même, le fils étant alors identique au père. Cela permet de conserver de façon sûre les meilleurs parents dans la génération suivante.

3.1.6 Mutation

Pour simuler la présence de mutations, juste après la reproduction, l'individu généré subit aléatoirement des mutations. La probabilité de mutation est de 1% (ce taux faible est justifié par la probabilité élevée de tomber sur des faux chemins ou des obstacles en mutant les cases c'est-à-dire la probabilité d'avoir un chemin rompu est de 75%, et 25 % pour un chemin complet), chacune des mutations modifie le chemin concerné en lui additionnant un nombre aléatoire, puis en prenant le résultat de la congruence modulo 4 qui représente les 4 directions.

3.2 Le module de Synchronisation

Il joue le rôle d'un gestionnaire d'accès aux cases, il utilise comme outil les sémaphores de type mutex dans les livres [11, 12] (un seul robot à la fois dans une case).

3.2.1 L'inter-blocage

Dans le cas d'inter-blocage où le robot 1 demande la case du robot 2 et le contraire, nous avons choisi un robot de façon aléatoire et on considère l'autre comme un obstacle puis le robot choisi fait un appel à l'algorithme génétique et recalcule le chemin en prenant en compte les changements de l'environnement, puis le deuxième robot (considéré comme obstacle) reprend son chemin (Fig. 3).

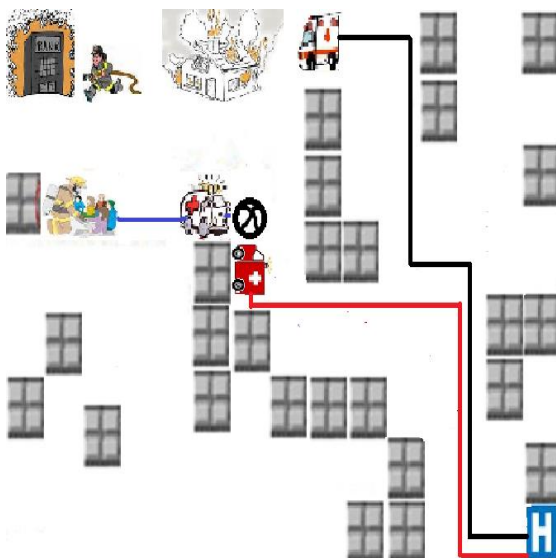


Fig. 3: Synchronisation entre les robots

4. Résultats et discussions

Afin de tester la performance de notre solution, on a essayé de la comparer avec des autres algorithmes avec les mêmes conditions : une matrice 15*15, 40 obstacles, le départ {Xd=0,Yd=0} et l'arrivée {Xa=14,Ya=14}.

Les résultats obtenus sont présentés dans le tableau 1 (Tableau. 1).

	Longueur du chemin	Taux de convergence
AG	28	100%
RN	31	91%
A*	28	100%
CF	29	96%

Tableau. 1 : Comparaison des résultats de l'algorithme génétique proposés (AG), réseau de neurone (RN), colonie de fourmis (CF) et l'algorithme astar (A*) (voir le livre [10]).

L'histogramme présenté (Fig. 4) montre la variation de la fitness en fonction de l'évolution des générations ; on remarque le tracé descendant qui traduit la minimisation de la fonction de fitness.

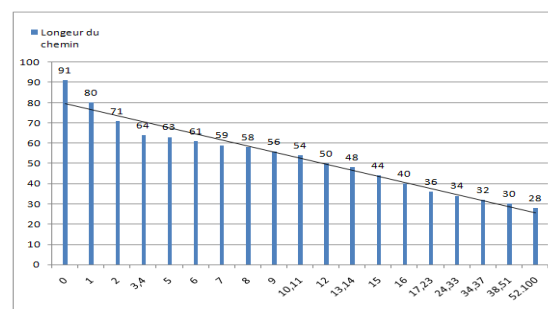


Fig. 4 : Histogramme présentant la variation du fitness

Les avantages de notre solution sont de deux ordres : On peut créer un nombre illimité de robots ce qui rend cette architecture dynamique et extensible et le multithreading qui réduit le temps d'exécution et augmente les performances de notre système.

5. Conclusion

Nous avons exposé notre modèle de planification multi-robot basé sur les algorithmes génétiques, représentant les robots, la synchronisation afin de construire le plan de navigation. Nous avons illustré notre modèle par un exemple de planification distribuée dans un environnement avec des obstacles. Cet exemple met en avant le dialogue entre les robots dans un

but de synchronisation et de répondre à l'objectif initial.

Référence :

- [1] Floreano, D., Kato, T., Marocco, D., Sauser, E. (2004) Coevolution of Active Vision and Feature Selection. *Biological Cybernetics*, vol. 90, no. 3, pages 218–228.
- [2] Boots, B., Nundy, S., Purves, D. (2007) Evolution of Visually Guided Behavior in Artificial Agents. *Network: Computation in Neural Systems*, vol. 18, no. 1, pages 11–34.
- [3] Maatallah, M., Seridi-Bouchelaghem, H. (2015) A fuzzy hybrid approach to enhance diversity I, top-N recommendations. *International Journal of business information systems*, 19(4), pp 505-530.
- [4] Zhu, Q., Yan, Y., Xing, Z. (2006) Robot Path Planning Based on Artificial Potential Field Approach with Simulated Annealing. *Sixth International Conference on Intelligent Systems Design and Applications*, Jinan, China, 2, pp.622-627.
- [5] Yang, S., Meng, M. (2000) An efficient neural network approach to dynamic robot motion planning. *Neural Networks*, 13(2), pp.143-150.
- [6] Makhoulouf, A., Lazli, L., Bensaker, B. (2016) Structure evolution of Hidden Markov Models FOR Audiovisual Arabic Speech recognition. *International Journal of signal and Imaging Systems Engineering*, IJSISE, 9(1), pp.55-66.
- [7] Benmachiche, A., Bouhadada, T., Laskri, M.T., Zendi, A. (2016) A dynamic navigation for autonomous mobiles robots. *Iospress: Intelligent Decision Technologies*, ISSN 1875-8843 (E), Volume 10(1).
- [8] Li, Q., Zhang, W., Yin, Y., Wang, Z. (2006) An Improved Genetic Algorithm of Optimum Path Planning for Mobile Robots. *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications*, 2, pp. 637 - 642.
- [9] Hu, Y., Yang, S.A. (2004) Knowledge Based Genetic Algorithm for Path Planning of a Mobile Robot. *Proceedings of the IEEE International Conference on Robotics & Automation*, pp. 4350-4355.
- [10] Michel, R. (1986) *Algorithms for Mutual Exclusion*, MIT Press, ISBN 0-262-181193.
- [11] M12. Sunil, R., Pradip, D., Srimani, K. (1991) *Distributed Mutual Exclusion Algorithms*, IEEE Computer Society, ISBN 0818633808.
- [12] Russell, S.J., Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Upper Saddle River, N.J.: Prentice Hall. pp. 97–104. ISBN 0-13-790395-2.