



**HAL**  
open science

## Planification de trajectoire d'un chariot autonome à fourche

Julien Pascal, Benoit Thuilot, Paul Checchin

► **To cite this version:**

Julien Pascal, Benoit Thuilot, Paul Checchin. Planification de trajectoire d'un chariot autonome à fourche. CNRIUT 2023 Saint Pierre - La Réunion, IUT de Saint Pierre, Jun 2023, Saint Pierre (La Réunion), La Réunion. <hal-04326931>

**HAL Id: hal-04326931**

**<https://hal.science/hal-04326931v1>**

Submitted on 6 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# Planification de trajectoire d'un chariot autonome à fourche

---

Julien Pascal<sup>2</sup> Benoit Thuilot<sup>2</sup> Paul Checchin<sup>1</sup>

julien.pascal@uca.fr benoit.thuilot@uca.fr paul.checchin@uca.fr

<sup>1</sup> IUT Clermont Auvergne, Université Clermont Auvergne  
Institut Pascal, UMR 6602 UCA/CNRS

<sup>2</sup> Université Clermont Auvergne  
Institut Pascal, UMR 6602 UCA/CNRS

**THÈMES** – Robotique

**RÉSUMÉ** – *Dans le cadre de l'usine du futur, automatiser les transports de produits au sein des ateliers permet à la fois d'accroître la compétitivité et de réduire la pénibilité et/ou la dangerosité de certaines tâches répétitives. Une piste est de rendre plus agiles et autonomes les chariots à fourche qui accostent, soulèvent et déplacent des objets dans les entrepôts existants. Afin d'améliorer sa manœuvrabilité, une configuration de véhicule à deux trains directeurs indépendants offre la possibilité de tout à la fois négocier des virages très serrés (trains avant et arrière braqués en sens opposé) et d'évoluer en crabe (trains avant et arrière braqués dans le même sens) pour se repositionner devant le produit à saisir. Mais le fait qu'une fourche prolonge le véhicule complexifie grandement les algorithmes de commande automatique. C'est l'objet de ces travaux qui sont menés dans le cadre du projet ACCROBOT (ACCostage haute précision ROBOTisé) financé par le Pack Ambition Recherche 2020 de la Région Auvergne-Rhône-Alpes (AURA).*

**MOTS-CLÉS** – *Usine du futur, Transitive, Chariot deux trains directeurs indépendants, Planification de trajectoire.*

## 1 Introduction

Pour envisager le déploiement de plateformes mobiles autonomes pour la manutention de charges lourdes au sein d'usines anciennes, non pensées initialement pour accueillir des robots, le choix technologique d'un chariot à deux trains directeurs indépendants a été fait pour une plus grande agilité (cf. Fig. 1). La planification de trajectoire du robot est rendue difficile car il faut exploiter le mieux possible les degrés de liberté apportés par les deux trains directeurs et gérer un encombrement constant à l'avant dû à une fourche utilisée pour effectuer la manutention. Ces caractéristiques particulières ne permettent pas l'utilisation directe de méthodes existantes. Celles-ci sont évoquées dans la section suivante avant d'introduire notre solution.

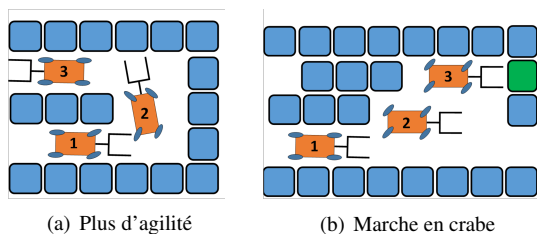


FIGURE 1 – Cinématique à deux trains directeurs

## 2 Planification de trajectoire en robotique mobile

Les algorithmes d'échantillonnage type *Rapidly Exploring Random Tree* (RRT) [1] ou *Probabilistic RoadMap* (PRM) [2], largement utilisés dans la littérature, ne sont pas retenus en raison de leur caractère non déterministe. Les méthodes fondées sur le *Machine Learning* sont performantes, mais nécessitent de grands jeux de données pour entraîner les modèles. Une revue détaillée de ces méthodes est donnée dans [3]. La décomposition en cellules [4] de la carte de l'environnement fournit un chemin qui est rarement réalisable par le robot (trop anguleux et nécessite des traitements supplémentaires pour le lisser). Il existe enfin les méthodes par champs de potentiel [5], dont le chemin s'obtient après une descente de gradient sur une carte de potentiels bien définis (répulsifs pour les obstacles, attractifs pour la solution). Parmi ces méthodes, le fast-marching a été privilégié. C'est un algorithme de recherche du plus court chemin dans un graphe, similaire à celui de Dijkstra [6]. En revanche, il utilise l'équation eikonale pour calculer les distances. Cette équation aux dérivées partielles, empruntée au monde de la physique, décrit le phénomène de propagation des ondes. Pour notre application, des chemins lisses sont naturellement obtenus (cf. Fig. 3(c)).

Dans notre variante de l'algorithme du fast-marching, l'espace de travail est discrétisé selon une grille. Chaque cellule se voit assigner un coût représentant la difficulté à

la traverser (infini pour les obstacles, puis décroissant à mesure que l'on s'en éloigne). Ce coût équivaut aux pondérations rencontrées dans un graphe sur lequel on applique un algorithme de plus court chemin. Le chemin final est obtenu en effectuant une descente de gradient. Pour garantir que le chemin soit réalisable par le robot, le gradient guide le modèle cinématique virtuel du véhicule (cf. Fig. 2) :

$$\begin{cases} dx = \tau \cos(\theta + \delta_R) \\ dy = \tau \sin(\theta + \delta_R) \\ d\theta = \tau \cos(\delta_R) \frac{\tan(\delta_F) - \tan(\delta_R)}{L} \end{cases}$$

avec  $\delta_R$  et  $\delta_F$  les angles de braquage arrière et avant des roues,  $L$  l'empattement et  $\tau$  le pas de descente. La configuration du robot est caractérisée par le triplet  $(x, y, \theta)$  avec  $(x, y)$  le centre de l'essieu arrière et  $\theta$  son orientation par rapport au repère de référence de l'environnement.

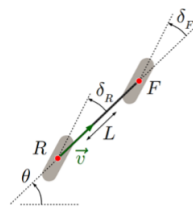


FIGURE 2 – Modèle de véhicule à deux trains directeurs

## 3 Cas d'un chariot à fourche

La gestion de l'encombrement du robot est problématique. En général, on inscrit le robot dans un cercle, puis les obstacles sont dilatés du rayon de ce cercle. Dans notre cas, avec une emprise au sol fortement allongée par la fourche, cette approche serait bien trop conservatrice. Aussi, en plus des positions  $(x, y)$ , les orientations  $\theta$  sont aussi discrétisées pour construire un espace 3D. La construction de l'environnement se fait en testant toutes les configurations  $(x, y, \theta)$  échantillonnées et en déterminant celles qui présentent ou non une collision avec un obstacle. Cette méthode *a priori* coûteuse se parallélise très bien sur GPU (une carte de  $50 \times 100$  positions  $(x, y)$  et 50 positions angulaires est traitée en 0,7 secondes sur GPU Nvidia T1200).

La méthode se décompose en trois grandes étapes (cf. Fig. 3) : partant d'une carte en deux dimensions avec la position des obstacles et des points de départ et d'arrivée, les positions angulaires sont ensuite échantillonnées pour effectuer le posé de robot et obtenir l'espace 3D navigable. La méthode du fast-marching est appliquée pour obtenir la carte des distances et la descente de gradient pour obtenir la trajectoire.

Différentes méthodes ont été testées pour identifier comment orienter les roues en fonction du gradient. Si on utilise l'essieu arrière pour optimiser la position  $(x, y)$  du robot

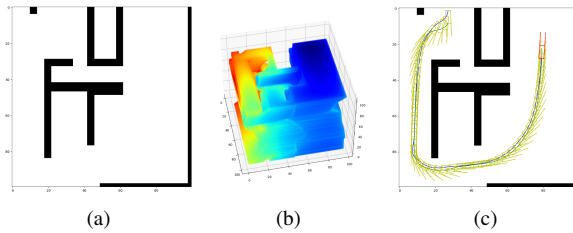


FIGURE 3 – Etapes de la méthode : (a) modélisation de l’environnement en grille binaire, (b) Posé de robot pour obtenir une carte en 3D et création d’un champ de potentiel (potentiel élevé loin de l’arrivée), (c) trajectoire finale obtenue par descente de gradient

et l’essieu avant pour optimiser son orientation, de nombreuses marches en crabe apparaissent ainsi que des oscillations lorsque l’on introduit les limites des angles de braquage. Du point de vue d’une application industrielle, ce comportement n’est pas envisageable. Pour remédier au problème, nous avons créé deux comportements. Le premier est identique au précédent lors de situations complexes, proches des obstacles ou pour manœuvrer en fin de parcours. Le second est un comportement plus proche de celui d’un véhicule à un train directeur lorsque l’on est loin des obstacles et de l’arrivée. Ce travail fait encore l’objet d’études complémentaires, mais le comportement général du robot est beaucoup plus proche de ce qu’un conducteur humain effectuerait.

L’ensemble des tests précédents sur la planification ont été effectués dans un environnement de simulation simple, développé pour l’occasion. Dans le but de valider notre solution au problème de planification et pour les futurs travaux, un test sur robot réel est prévu. L’INRAe a développé sur le site clermontois une plateforme robotique à deux trains directeurs : Adap2e (cf. Fig. 4). Cette plateforme a d’ores et déjà été mise en œuvre ce qui a conduit à adapter certaines parties de la méthode pour accepter un chemin sous la forme d’une suite de configurations  $(x, y, \theta)$  et non pas une suite de positions  $(x, y)$ . Les lois de commandes ont de même été adaptées pour vérifier la bonne prise en compte de cette information supplémentaire. Avec ce développement dédié, il est possible de passer très facilement de la simulation au robot réel, c’est-à-dire du poste informatique du bureau à la plateforme expérimentale sur le terrain.

## 4 Conclusions

Une approche de planification de trajectoire d’un chariot à fourche à deux trains indépendants a été présentée. Il faudra la valider avec un suivi de trajectoire du robot réel nécessitant des lois de commande adaptées, mais qui restent à implémenter sur le robot. La suite des travaux concerne la partie liée à l’accostage. Il faudra déterminer la position



FIGURE 4 – Robot Adap2e

optimale que le robot doit atteindre pour effectuer la saisie. L’aspect perception de l’environnement du robot est une autre difficulté du projet. Il faudra coupler celui-ci à la commande pour réussir à stabiliser le robot sur la configuration cible. Les charges déplacées dans le cadre industriel visé sont lourdes et déportées à l’extrémité du chariot. Aussi, on ne peut pas ignorer les effets inertiels qui entrent en jeu. Il sera donc nécessaire de compléter les lois de commandes pour intégrer explicitement ces effets inertiels et les glissements.

## 5 Remerciements

Le projet ACCROBOT (ACCostage haute précision ROBOTisé - Chantier Transitiqque du laboratoire FACTOLAB) est financé par le Pack Ambition Recherche 2020 de la Région Auvergne-Rhône-Alpes (AURA). Les travaux sont effectués au sein de l’Institut Pascal dans le cadre de FACTOLAB, laboratoire commun issu de la collaboration avec le groupe Michelin dédié à l’industrie du futur.

## Références

- [1] LaValle, S. M. (1998). *Rapidly-exploring random trees : A new tool for path planning*.
- [2] Kavraki, L. E., Svestka, P., Latombe, J. C., and Overmars, M. H. (1996). *Probabilistic roadmaps for path planning in high-dimensional configuration spaces*. IEEE Transactions on Robotics and Automation, 12(4), 566-580
- [3] Zhou, Chengmin, Bingding Huang, and Pasi Fränti, *A review of motion planning algorithms for intelligent robots*, Journal of Intelligent Manufacturing (2021), p. 1-38
- [4] Gonzalez, Ramon, Marius Kloetzer, and Cristian Mahulea, *Comparative study of trajectories resulted from cell decomposition path planning approaches*, 2017 21st International Conference on System Theory, Control and Computing (ICSTCC), IEEE, 2017
- [5] Hwang, Yong Koo, and Narendra Ahuja, *A potential field approach to path planning*, IEEE Transactions on Robotics and Automation, 8.1 (1992), p. 23-32
- [6] Sethian, J. A., *A fast marching level set method for monotonically advancing fronts*, Proceedings of the National Academy of Sciences, Applied Mathematics, Vol. 93 (1996), p. 1591-1595