

Amélioration de la conduite de robots mobiles

Alexis Scheuer

► **To cite this version:**

Alexis Scheuer. Amélioration de la conduite de robots mobiles. [Rapport de recherche] RR-3994, INRIA. 2000, pp.18. <inria-00072652>

HAL Id: inria-00072652

<https://hal.inria.fr/inria-00072652>

Submitted on 24 May 2006

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.

Amélioration de la conduite de robots mobiles

Alexis Scheuer

N° 3994

Septembre 2000

THÈME 3

 ***Rapport
de recherche***

Amélioration de la conduite de robots mobiles

Alexis Scheuer*

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Maia

Rapport de recherche n° 3994 — Septembre 2000 — 18 pages

Résumé :

Les travaux présentés dans cet article ont pour objectif l'amélioration de la conduite de robots mobiles, que ceux-ci soient manœuvrables (c'est-à-dire capables de tourner sur place) ou que leurs déplacements soient plus contraints (comme ceux des voitures, par exemple). Pour atteindre ce but, nous avons commencé tout d'abord par une étude des travaux existant dans ce domaine. Cette étude a permis de généraliser plusieurs approches mises en oeuvre pour des problèmes connexes, afin d'une part de définir et d'étudier un nouveau problème de planification de mouvement et d'autre part de proposer une méthode pour résoudre ce problème. Enfin, les solutions obtenues par cette méthode ont été comparées aux solutions fournies par les méthodes classiques, afin d'explicitier leur apport.

Mots-clés : robotique mobile, planification de trajectoires, système non-holonome, optimalité

* Maître de Conférences de l'Université Henri Poincaré (Nancy I) - Alexis.Scheuer@loria.fr - <http://www.loria.fr/~scheuer>

Improving Car-like Robots' Driving

Abstract:

The aim of the works presented in this article is to improve the driving of mobile robots, these robots being manoeuvrable (i.e. able to turn without moving) or having a more constrained motion (for example, as car do). To reach this goal, existing works in the domain have first been reviewed. This review leads us to generalise several approaches, used previously to solve connected problems, in order on one hand to define and study a new motion planning problem, and on the other hand to find a method solving this problem. At last, solutions generated with this method have been compared to those generated by usual methods, showing the benefits of the first ones.

Key-words: mobile robot, trajectory planning, nonholonomic system, optimality

Table des matières

1	Introduction	4
2	Étude de l'existant	4
3	Problème considéré	5
3.1	Plateformes expérimentales	5
3.2	Modèles des robots	6
3.3	Espace des configurations	7
3.4	Trajectoires admissibles	7
3.5	Problème de planification	8
4	Propriétés du problème	8
4.1	Commandabilité	9
4.2	Trajectoires optimales	9
5	Solution proposée	10
5.1	Trajectoires utilisées	10
5.2	Planification locale	11
5.3	Méthode globale	12
6	Résultats obtenus	13
6.1	Robots manœuvrables	13
6.2	Robots de type voiture	13
6.2.1	Complexité du calcul	13
6.2.2	Longueur des chemins	14
6.2.3	Qualité du suivi	14
7	Conclusion et perspectives	14

1 Introduction

Dans les romans de science-fiction, l’an deux mille a souvent été dépeint peuplé d’une foule de robots, conçus pour nous soulager des tâches les plus ingrates. Dans la réalité, les seuls robots de ce genre sont ceux des chaînes de montage, qui ne savent que répéter un même geste dans un environnement réservé (c’est-à-dire dépourvu d’obstacle). Depuis un quart de siècle, les roboticiens cherchent à améliorer les capacités de ces robots, en les dotant de plus d’intelligence. Ces recherches ont principalement permis de mesurer la complexité de raisonnement nécessaire à la définition de robots génériques, c’est-à-dire capables de réagir à une large variété de situations. On préfère maintenant s’intéresser à des robots spécialistes, ou du moins considérer séparément chacune des capacités élémentaires nécessaires au robot.

Ainsi, de nombreux travaux se sont consacrés à la planification de mouvement, c’est-à-dire la capacité de prévoir un déplacement dans un environnement encombré d’obstacles. Dans un premier temps, les robots mobiles ont été conçus de façon à être très manœuvrables : ces robots peuvent tourner sur place, ce qui leur permet (à condition de s’arrêter) de suivre une succession de segments de droite. Ensuite, une nouvelle génération de planificateurs a fournis des chemins plus sophistiqués, contenant en plus des arcs de cercle. Ces chemins peuvent être suivis par les robots manœuvrables avec une vitesse plus importante que les précédents. Ils peuvent aussi être suivis par des robots dont les déplacements sont plus contraints, en particulier les robots de type voiture, dont le rayon de giration est minoré (pour plus de détails, cf. § 3.2).

Le travail présenté dans cet article se situe dans le prolongement de ces travaux : il propose une nouvelle génération de planificateurs, basés sur des chemins encore améliorés. Après avoir présenté quelques travaux portant sur ce sujet (§ 2), le nouveau problème de planification considéré est posé formellement dans la section 3, puis ses propriétés sont étudiées (par rapport à celles du problème de la génération précédente) dans la section 4. Ensuite, une solution permettant de résoudre ce problème est proposée (§ 5), ainsi que les résultats obtenus grâce à cette solution (§ 6). Enfin, cet article se termine par une synthèse des travaux présentés, suivie par les perspectives qu’ils permettent d’envisager (§ 7).

2 Étude de l’existant

Les chemins les plus utilisés actuellement pour des robots de type voiture sont constitués d’arcs de cercle reliés tangentiellement par des segments de droite [13, 8, 17, 26, 24]. Ces chemins sont les plus courts, qu’on se déplace uniquement en marche avant [5] ou en faisant des manœuvres (passage marche avant-marche arrière et réciproquement) [19]. Le principal inconvénient de ces chemins est qu’on ne peut pas les suivre précisément sans s’arrêter à chaque transition arc-segment (et réciproquement), du fait de la discontinuité de la courbure.

Deux solutions ont été proposées pour résoudre ce problème : effectuer un suivi « intelligent » permettant de ne pas s’arrêter (en s’écartant de ces chemins) ou les lisser afin d’obtenir des chemins à courbure continue. La première solution peut poser problème en présence d’obstacles : on risque alors d’effectuer des manœuvres inutiles. Quant à la seconde

solution, elle n'a pour l'instant pas été appliquée avec un lissage pertinent : plusieurs types de courbes ont été proposées pour ce lissage (fonctions polynomiales pour les coordonnées [12, 18] ou pour la courbure [10, 4], mais peu d'entre elles concernent les robots de types voiture [15, 25, 18]. Enfin, les travaux proposant des courbes adaptées aux modèles des robots considérés sont très rares, et portent uniquement sur des robots manœuvrables [14, 1].

Nos recherches se situent dans la lignée de ces derniers travaux : pour un nouveau type de robot manœuvrable et les robots de type voiture, nous avons proposé puis étudié une planification à courbure continue adaptée au modèle du robot.

3 Problème considéré

Le problème considéré dans cet article est celui de la planification de mouvement : on souhaite déterminer, d'une manière automatique, comment un robot peut se déplacer d'une position à une seconde, en évitant toute collision avec un ensemble d'obstacles (fixes ou mobiles) et en respectant des contraintes de déplacement. Dans ce cadre général, nous nous intéressons plus particulièrement aux déplacements de deux types de robots, présentés dans la section 3.1. Nous verrons comment ces robots sont modélisés géométriquement et dynamiquement (§ 3.2), le nouveau point de vue que fournit cette modélisation (§ 3.3), comment les contraintes limitant leurs déplacements sont représentées (§ 3.4), et enfin comment s'exprime formellement le problème de planification que nous devons alors résoudre (§ 3.5).

3.1 Plateformes expérimentales

Les deux types de robots auxquels nous nous sommes intéressés sont assez différents : le premier est un robot manœuvrable (il peut tourner sur place) alors que le second est de type voiture. Nous verrons cependant que leurs déplacements possèdent des points communs.

Le premier robot est un Nomad 200TM (©1998 Nomadic Technologies, Inc.¹), constitué d'une tourelle cylindrique montée sur trois roues (*cf.* figure 1). La particularité de ce robot est que ses roues pivotent en restant parallèles à tout instant : on contrôle ce robot par le roulement et le pivotement de ses roues (commun aux trois). Il existe un autre type de robot manœuvrable, dont les deux roues principales sont fixes par rapport au châssis, les autres roues étant libres. On contrôle un tel robot par le roulement (indépendant) de ces deux roues. Nous verrons (§ 4.2) l'influence de cette différence sur la nature des trajectoires optimales pour le robot.

Le second type de robots que nous avons considéré est beaucoup plus générique : il s'agit de tous les robots mobiles dont le déplacement est similaire à celui d'une voiture. Un tel robot est bien sûr beaucoup plus contraint que les robots manœuvrable : il ne peut résolument pas tourner sur place. Nous verrons cependant que les trajectoires optimales pour ce type de robot ont des points communs avec les trajectoires optimales pour les robots manœuvrables.

1. <http://www.robots.com>



FIG. 1 – un Nomad 200TM, un Cycab.

3.2 Modèles des robots

On suppose que nos robots se déplacent sur un sol plan. Pour les deux types de robots, nous allons considérer deux niveaux de modélisation : un aspect purement géométrique (suffisant pour la détection de collision) et un dynamique (nécessaire pour le déplacement). À ces deux niveaux, on va définir une *configuration*, c'est-à-dire un ensemble de paramètres permettant de représenter le robot. La configuration géométrique représente donc la position du robot, la configuration dynamique représente sa position et sa vitesse.

Ainsi, pour le Nomad 200TM, un point (la projection R de l'axe de la tourelle) suffit à caractériser sa position, alors qu'il faut rajouter l'orientation θ des roues et leurs vitesses respectivement de roulement v et de pivotement ω pour caractériser son déplacement (cf. figure 2). On a donc comme configurations géométrique p_N et dynamique q_N :

$$p_N = (x, y), q_N = (x, y, \theta, v, \omega)$$

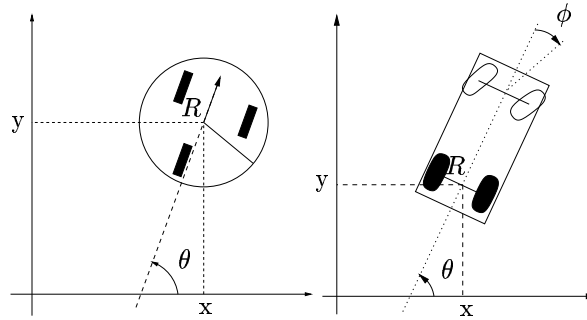


FIG. 2 – modèles des robots considérés.

Par contre, un point ne suffit pas pour caractériser la position d'un robot de type voiture : il faut au moins un point (par exemple le milieu R de l'axe des roues arrière) et une orientation θ (celle de l'axe principal du véhicule). Pour caractériser le déplacement de ce robot, il faut rajouter à cela la vitesse instantanée v de R et la courbure κ de la courbe qu'il

suit (représentant l'orientation ϕ des roues avants) (cf. figure 2). On a donc :

$$p_V = (x, y, \theta), q_V = (x, y, \theta, v, \kappa)$$

3.3 Espace des configurations

Utilisant cette formalisation, on peut considérer l'ensemble des configurations comme un espace, celui des configurations géométriques pouvant être plongé dans celui des configurations dynamiques (pour chaque type de robot). Un chemin correspond alors à une courbe dans l'espace des configurations géométriques, et une trajectoire à une courbe dans l'espace des configurations dynamiques.

On dit de plus qu'une configuration est *libre* lorsqu'elle correspond à une position pour laquelle le robot n'est pas en collision (ou en contact) avec un obstacle. Une trajectoire est donc sans collision si et seulement si elle correspond à une courbe dans l'espace des configurations dynamiques libres. Planifier une trajectoire sans collision revient donc à déterminer une courbe dans l'espace des configurations dynamiques libres.

En pratique, cependant, le calcul de l'espace des configurations dynamiques libres est trop coûteux pour être effectué : les calculs de collision sont effectués après transposition de l'espace des configurations dans l'espace réel. L'espace des configurations est uniquement utilisé comme outil formel.

3.4 Trajectoires admissibles

Toute courbe de l'espace des configurations dynamiques ne correspond pas forcément à une trajectoire *admissible*, c'est-à-dire que le robot peut suivre. Rappelons en effet que les déplacements de nos robots sont contraints. La première de ces contraintes, commune aux deux types de robot, est nommée *contrainte d'orientation* et s'écrit :

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0$$

Elle signifie que les roues noires des schémas de la figure 2 doivent rouler sans glisser : leur orientation θ contraint le mouvement du point R . Cette contrainte est dite *non-holonome* : elle limite le déplacement du robot sans limiter les positions qu'il peut atteindre.

Plus généralement, le déplacement d'un robot, représenté par la dérivée de sa configuration dynamique, peut être contraint par une équation de la forme :

$$\dot{q} = f(q, u), q \in C, u \in U(q)$$

On distingue alors trois catégories de contraintes de déplacement :

- les contraintes holonomes, qui limitent l'espace C des configurations dynamiques q atteignables,
- les contraintes non-holonomes portant sur les paramètres de contrôle u (qui se bornent à limiter les valeurs possibles, éventuellement en fonction de q) et

- les contraintes non-holonomes de déplacement proprement dites, qui détermine la variation de configuration en fonction de la configuration actuelle et du contrôle appliqué. Dans notre cas, on a alors pour le Nomad 200TM :

$$\begin{aligned} q_N &= (x, y, \theta, v, \omega), & C_N &\subset W_N \times S^1 \times V_N, \\ \dot{q}_N &= (v \cos \theta, v \sin \theta, \omega, a, \gamma), & u_N &= (a, \gamma) \in U_N \end{aligned}$$

où W_N est l'environnement dans lequel se déplace le robot, V_N est l'intervalle des vitesses autorisées (de la forme $[v_{\min}, v_{\max}] \times [-\omega_{\max}, \omega_{\max}]$) et U_N celui des contrôles ($[a_{\min}, a_{\max}] \times [-\gamma_{\max}, \gamma_{\max}]$).

De même, pour les robots de type voiture, on a :

$$\begin{aligned} q_V &= (x, y, \theta, v, \kappa), & C_V &\subset W_V \times S^1 \times V_V, \\ \dot{q}_V &= (v \cos \theta, v \sin \theta, v \kappa, a, \sigma), & u_V &= (a, \sigma) \in U_V \end{aligned}$$

où W_V est l'environnement dans lequel se déplace le robot, V_V est l'intervalle des vitesses autorisées (de la forme $[v_{\min}, v_{\max}] \times [-\kappa_{\max}, \kappa_{\max}]$) et U_V celui des contrôles ($[a_{\min}, a_{\max}] \times [-\sigma_{\max}, \sigma_{\max}]$).

On remarque que les formulations sont tout à fait similaires, même si le contrôle se fait dans le premier cas sur les accélérations de roulement a et de pivotement γ , et dans le second cas sur l'accélération de translation a et la variation de courbure σ .

3.5 Problème de planification

Le problème de planification que nous souhaitons résoudre est le suivant. Étant données deux configurations dynamiques q_d et q_f et un ensemble d'obstacles O , on cherche une trajectoire (c'est-à-dire une courbe de l'espace C des configurations dynamiques) qui :

- relie la configuration de départ q_d à la configuration finale q_f ,
- est admissible (et donc respecte les contraintes définies dans la section précédente), et
- est sans collision avec les obstacles de O .

Nous allons proposer une méthode pour résoudre les deux instances de ce problème, concernant respectivement le Nomad 200TM et les robots de type voiture (§ 5). Auparavant, nous allons étudier les caractéristiques de ces deux instances, en ce qui concerne l'existence de solutions, et l'existence et la nature de solutions optimales.

4 Propriétés du problème

Nous nous intéressons donc ici à ces deux propriétés du problème : quand existe-t-il des solutions au problème de planification précédent (§ 4.1), et existe-t-il des solutions optimales et de quels types de courbe sont-elles constituées (§ 4.2) ?

4.1 Commandabilité

Le Nomad 200TM pouvant tourner sur place, il est évidemment *commandable*, c'est-à-dire qu'il existe toujours une solution au problème de planification lorsque O est vide. Il est même *commandable en temps petit*, c'est-à-dire qu'il existe une solution à ce problème dès que les configurations à relier sont dans la même partie connexe de l'intérieur de l'ensemble des configurations libres [23, § 3.3]. Autrement dit, une solution existe dès qu'il est possible de relier les configurations par une trajectoire (pas forcément admissible mais) sans collision avec les obstacles : la prise en compte des contraintes de déplacement n'a donc aucune incidence sur l'existence d'une solution [13].

Les mêmes propriétés ont été démontrées pour les robots de type voiture, lorsqu'on considère des trajectoires dont la courbure peut être discontinue. Ces trajectoires pouvant être modifiées afin de rendre continue leur courbure (en ajoutant des portions où la courbure est modifiée à l'arrêt), la commandabilité sans continuité de la courbure garantit celle avec continuité de la courbure. Ce résultat peut aussi être obtenu directement [22, § 4.1] [20, § II.2.1] en utilisant un théorème algébrique de commandabilité [16, th. IV.3].

4.2 Trajectoires optimales

L'existence de trajectoires optimales est un problème algébrique, et peut être démontré (entre autre) par le théorème de Filippov [3, th. 5.1.ii]. Cette existence est cependant conditionnée par une condition : on doit imposer une distance minimale entre le robot et les obstacles, de manière à ce que l'espace des configurations libres soit un compact, et non un ouvert.

En revanche, la détermination des trajectoires optimales (en temps) est un problème trop complexe pour être résolu en totalité. En effet, on ne dispose que d'un théorème (algébrique, encore une fois) caractérisant des conditions que doivent nécessairement vérifier ces trajectoires : le Principe du Maximum de Pontryagin, ou PMP [3, th. 5.1.i]. L'utilisation de ce théorème permet de déterminer les types de courbe dont les trajectoires sont composées, mais ne permet pas de savoir précisément quelle est la trajectoire optimale pour un problème donné.

En particulier, l'application du PMP dans les cas qui nous intéressent a permis de montrer que les trajectoires optimales correspondent à un contrôle *bang-bang* : les paramètres de contrôle (c'est-à-dire les accélérations ou la variation de la courbure) ont toujours des valeurs minimales, nulles ou maximales, la valeur nulle n'étant prise que lorsque la vitesse correspondante ou la courbure est elle-même minimale, nulle ou maximale.

Pour le Nomad 200TM, ce résultat généralise les résultats obtenus pour l'autre type de robot manœuvrable [14], et permet d'affirmer que les trajectoires optimales sont constituées de segments de droite, d'arcs de cercle, de rotations sur place, de portions de clothoïde² et

². Une clothoïde est une courbe dont la courbure κ varie linéairement en fonction de son abscisse curviligne s [14].

d'anti-clothoïde³, et de morceaux d'une courbe dont les coordonnées combinent celles du cercle et de la clothoïde.

En ce qui concerne les robots de type voiture, les trajectoires optimales sont constituées de segments de droite, d'arcs de cercle, de portions de clothoïde et de morceaux d'une spirale similaire aux clothoïdes. Ce résultat généralise celui obtenu pour un robot manœuvrable dont la dérivé de la courbure reste bornée [1, prop. 1]. Cependant, comme dans cet autre cas, les trajectoires optimales ne seront pas utilisables : elles sont constituées d'une infinité de morceaux dès qu'elles contiennent (sans être limitées à) un segment de droite [1, prop. 2].

5 Solution proposée

Ayant étudié les caractéristiques du problème de planification qui nous intéresse, nous allons maintenant en proposer une solution. Nous présenterons d'abord les trajectoires que nous avons utilisées et nous justifierons leur utilisation par rapport à celle des trajectoires optimales (§ 5.1). Nous montrerons ensuite comment de telles trajectoires sont calculées localement, c'est-à-dire sans prendre en compte l'évitement d'obstacle (§ 5.2), puis comment cette planification locale est utilisée pour résoudre le problème de planification (§ 5.3).

5.1 Trajectoires utilisées

Comme nous venons de le voir, les trajectoires optimales sont constituées d'une large variété de courbes de types différents. Par conséquent, le nombre de trajectoires possibles pour relier deux configurations données est très élevé. La possibilité de faire des manœuvres (passage de la marche avant à la marche arrière, et réciproquement) augmente encore ce nombre. Résoudre le problème de planification risque donc d'être assez lent, ou même impossible, les calculs étant trop complexes. Pour résoudre notre problème (et ce, en un temps raisonnable), nous avons préféré utiliser d'autres trajectoires que les trajectoires optimales.

Tout d'abord, nous avons utilisé le principe de la *décomposition chemin-vitesse* [11] : nous choisissons un chemin le long duquel nous calculons un profil de vitesse, l'ensemble des deux définissant une trajectoire. Ce principe (une application du classique "divide and conquer", « diviser pour régner ») consiste à décomposer le problème pour le simplifier.

Les chemins que nous avons choisis sont une généralisation des chemins proposés d'abord par Dubins dans le cas de la marche avant uniquement [5], puis par Reeds et Shepp lorsqu'on fait des manœuvres [19]. Nous avons seulement rajouté à ces chemins des portions de clothoïde de manière à connecter les segments de droite et les arcs de cercle avec une variation continue de la courbure (*cf.* figure 3).

Ainsi, les virages, constitués uniquement d'un arc de cercle pour Dubins ou Reeds et Shepp, ont été remplacés par des virages à courbure continue, ou virages CC, formés d'une portion de clothoïde, un arc de cercle (optionnel) et d'une portion de clothoïde symétrique à la première [21, § 4.2] [20, § VII.2.]. Nous appelons les chemins alors obtenus des *chemins*

³. Une anti-clothoïde est la courbe parcourue lorsque la vitesse de translation v varie linéairement et la vitesse de rotation ω reste constante [14].

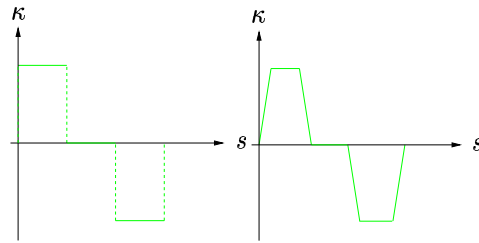


FIG. 3 – profils de courbure (Dubins / CSCC).

*sous-optimaux à courbure continue (CSCC)*⁴ : nous en avons montré la sous-optimalité, c'est-à-dire que la différence entre ces chemins et les chemins optimaux de mêmes extrémités peut être bornée [22, § 6.2] [20, annexe D.2].

Un chemin de ce type, ou formé d'une séquence de CSCC dans la même direction (avant ou arrière), étant obtenu, on calcule un profil de vitesse optimal le long de ce chemin. Ce profil commence par une partie dont l'accélération est maximale, jusqu'à atteindre la vitesse maximale ou le milieu du chemin, et finit par une partie symétrique (décélération maximale jusqu'à l'arrêt), la partie médiane (optionnelle) étant parcourue à vitesse maximale.

5.2 Planification locale

Il s'agit de calculer le plus court CSCC permettant de relier deux configurations données. Nous avons généralisé pour cela les méthodes utilisées pour calculer des chemins de Dubins ou de Reeds et Shepp. Le principal problème vient de la complexité des virages CC (à courbure continue). En effet, ces méthodes utilisent le fait que, partant d'une configuration q_0 fixée (associant une position et un vecteur donnant l'orientation), un virage en arc de cercle mène forcément à une configuration q située sur un des deux cercles contenant la position de la configuration d'origine, et tangent au vecteur de cette configuration (*cf.* figure 4).

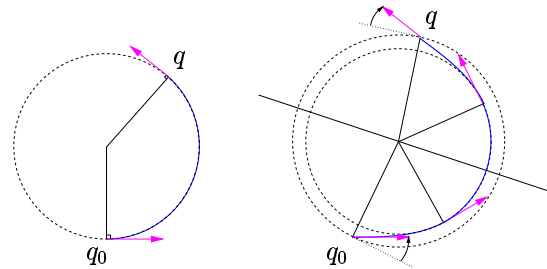


FIG. 4 – virage en arc de cercle et virage CC.

4. De tels chemins sont utilisés pour le tracé des autoroutes ou des lignes de TGV.

Dans le cas des virages CC, on a pu montrer que cette propriété se généralisait : si le virage est assez long (pour contenir un arc de cercle), sa courbure symétrique rend son tracé symétrique et, partant d'une configuration q_0 , sa configuration finale reste sur un cercle, les vecteurs d'orientation des configurations initiale et finale faisant des angles opposés avec la tangente au cercle issue de la position de leur configuration (*cf.* figure 4). Dans le cas où le virage est trop court, on peut déterminer un virage CC permettant de rejoindre le même cercle, les portions de clothoïde n'ayant pas alors une dérivée de la courbure maximale (en valeur absolue). Le rayon des cercles et l'angle avec la tangente ne dépendent que des valeurs de κ_{\max} et de σ_{\max} , et tendent respectivement vers $1/\kappa_{\max}$ et 0, les valeurs correspondant aux cas de Dubins et de Reeds et Shepp, lorsque σ_{\max} tend vers l'infini, c'est-à-dire quand on admet des discontinuités de la courbure [21, § 4.2] [20, § VII.2].

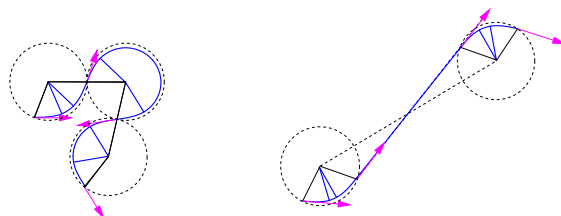


FIG. 5 – deux exemples de CSCC.

La méthode de Dubins, concernant des déplacements en marche avant uniquement [21, § 4.3] [20, § VII.3], puis celle de Reeds et Shepp [6], a alors été transformée en modifiant les cercles considérés (changement du rayon et de la position) et considérant non plus des droites tangentes au cercle mais celles les coupant en faisant un certain angle avec la tangente (*cf.* figure 5).

5.3 Méthode globale

Cette méthode de planification locale nous permet donc de calculer un chemin admissible reliant deux configurations. Pour trouver un chemin admissible sans collision reliant ces deux configurations, il suffit de déterminer une séquence de configurations intermédiaires permettant d'éviter les obstacles. Il existe pour cela plusieurs méthodes, dont certaines sont *génériques* (c'est-à-dire indépendantes de la planification locale).

Dans notre cas, nous avons utilisé une telle méthode, nommée *Probabilistic Path Planning method* [26] : il s'agit en fait de créer un graphe pour recouvrir l'espace des configurations libres, les noeuds de ce graphe étant des configurations libres et ses arêtes des CSCC obtenus par la planification locale. Le problème de planification est résolu lorsqu'on sait relier, grâce à la planification locale, les configurations de départ et d'arrivée à deux noeuds de la même composante connexe du graphe : un parcours du graphe nous permet alors de relier les configurations de départ et d'arrivée par une succession de CSCC, dont on déduit une trajectoire.

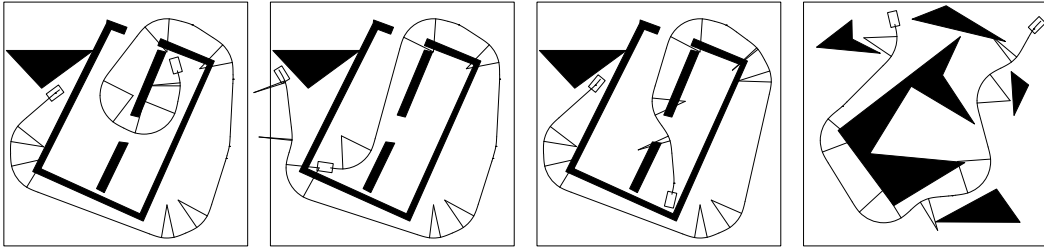


FIG. 6 – exemples de successions de CSCC.

6 Résultats obtenus

Bien que la méthode utilisée soit la même pour les deux applications, les résultats expérimentaux ne peuvent être considérés ensemble. Nous allons donc présenter ceux-ci séparément, suivant qu'ils concernent les robots manœuvrables (§ 6.1) ou les robots de type voiture (§ 6.2).

6.1 Robots manœuvrables

Les seuls résultats que nous ayons obtenus pour le Nomad 200TM sont ceux concernant les caractéristiques du problème et montrant sa similitude avec celui concernant les robots de type voiture (*cf.* § 3 & 4). En effet, il n'existe pas à notre connaissance d'autres travaux étudiant la planification de trajectoires adaptées aux contraintes dynamiques du Nomad 200TM, ni de résultats auxquels nous aurions pu comparer les nôtres. D'autre part, plusieurs problèmes techniques n'ont pas permis de mettre en pratique les résultats théoriques obtenus : précision déplorable du contrôleur du robot (contrairement à ce que promettait le manuel de l'utilisateur), manque de temps et d'informations pour développer un autre contrôleur, défection mécanique du robot [23], etc...

6.2 Robots de type voiture

Par contre, pour les robots de type voiture, nous avons pu comparer nos résultats avec ceux obtenus en utilisant la méthode de Dubins. La comparaison entre les résultats obtenus avec notre méthode et celle de Reeds et Shepp n'a pas encore été effectuée, mais elle a toutes les chances d'apporter les mêmes conclusions.

Les conclusions que nous présentons ici portent sur trois points : la complexité de la méthode de planification utilisée et la qualité des chemins obtenus, représentée par leur longueur et la facilité avec laquelle ils sont suivis [22, § 6] [20, § VIII.2].

6.2.1 Complexité du calcul

La méthode de planification locale que nous avons utilisée n'est qu'une généralisation de celles de Dubins et de Reeds et Shepp. Elles sont toutes d'ordre constant, ce qui implique

que les planifications globales sont de même ordre, quelle que soit la planification locale qu'on utilise.

Expérimentalement, on a pu constater un rapport compris entre 1,5 et 2 entre le temps de calcul des CSCC sans manœuvre et les chemins de Dubins, sur un millions d'exemples tirés aléatoirement. Nous allons voir que cette légère perte de temps est largement compensée par l'amélioration du suivi.

6.2.2 Longueur des chemins

Nous avons déjà vu que les CSCC sont sous-optimaux. Cependant, la différence de longueur entre les CSCC et les chemins optimaux n'a pu qu'être largement majorée, du fait de la complexité des calculs. Cette différence peut aussi être majorée par la différence de longueur entre les CSCC et les chemins de Dubins. Cette dernière ne peut être calculée dans tous les cas, mais elle peut être évaluée expérimentalement pour un éventail de cas. Sur le millions d'exemples dont nous avons parlé, nous avons constaté que 82% des différences n'excèdent pas 10% de la longueur du chemin de Dubins, bien que le ratio des longueur puisse tendre vers l'infini pour certains cas pathologiques (le ratio maximum constaté sur ce millions d'exemples est 8,3). Il est important de noter que cet allongement du chemin ne fait qu'anticiper celui du chemin réellement suivi par le robot lorsqu'on ne souhaite pas s'arrêter.

6.2.3 Qualité du suivi

Les points précédents sont largement compensés par l'amélioration du suivi constatée entre les CSCC et les chemins de Dubins. Cette amélioration a été évaluée en mesurant l'écart entre le chemin planifié et celui réellement suivi, en utilisant une méthode classique de suivi [9]. Cet écart est divisé par près de 100 lorsqu'on se déplace à vitesse réduite (1 m.s^{-1}), et reste divisé par plus de 15 lorsque la vitesse s'accroît (3 m.s^{-1}). Nous avons même constaté, à cette dernière vitesse, des cas où le chemin de Dubins engendre une situation de blocage, alors que le CSCC est suivi avec une précision approchant 15 cm.

Ce planificateur a été utilisé, en collaboration avec un contrôleur [7], pour piloter un véhicule expérimental (*cf.* figure 7).

7 Conclusion et perspectives

Dans cet article, nous avons montré comment il est possible d'améliorer la conduite de robots mobiles, que ceux-ci soient très manœuvrables ou que leur déplacements soient plus limités (comme ceux d'une voiture, par exemple). Nous avons pour cela utilisé de nombreux outils mathématiques (algèbre, analyse et géométrie), dont certains spécifiques au domaine considéré (la robotique). Un nouveau problème de planification de mouvement ayant été défini, ces outils ont permis de caractériser ce problème (par rapport aux problèmes classiques), en ce qui concerne l'existence de solutions et l'existence et la nature de solutions



FIG. 7 – une Ligier.

optimales. Ils ont aussi permis de proposer une méthode pour résoudre ce problème, de façon à obtenir une solution dont la qualité a été comparée à celle des solutions classiques.

Les perspectives actuelles ouvertes par ce travail sont les suivantes :

- on envisage tout d’abord de remplacer les clothoïdes par d’autres courbes, afin d’obtenir des trajectoires infiniment dérivables et non simplement C^2 ; le seul problème concerne le choix de ces courbes, le principe de planification restant le même ;
- on peut aussi vouloir améliorer la planification locale ; pour l’instant, elle construit plusieurs chemins reliant deux configurations et choisit le plus court d’entre eux ; il est possible de construire une partition de l’espace des configurations en fonction du type du chemin le plus court [2] ;
- on pourra enfin essayer de nouvelles méthodes de planification globale, afin d’en trouver des plus rapides ; en particulier, on pourra ne prendre en compte que progressivement les contraintes de déplacement [24].

Remerciements : cet article présente une synthèse des recherches que j’ai effectuées au cours de ma thèse de doctorat, qui s’est déroulée dans le projet Sharp, commun au laboratoire Gravir de l’Imag et à l’Inria Rhône-Alpes, ainsi que de celles de mon stage post-doctoral dans le “Vision and Control” Strategic Research Program de l’école de Mechanical and Production Engineering de la Nanyang Technological University, à Singapour. Je tiens à remercier toutes les personnes qui ont rendu ces recherches à la fois passionnantes et agréables.

Références bibliographiques

- [1] J.-D. Boissonnat, A. Cerezo, and J. Leblond. A note on shortest paths in the plane subject to a constraint on the derivative of the curvature. Research Report 2160, Inst. Nat. de Recherche en Informatique et en Automatique, January 1994. <http://www.inria.fr/RRRT/RR-2160.html>.

- [2] X.-N. Bui, P. Souères, J.-D. Boissonnat, and J.-P. Laumond. Shortest path synthesis for Dubins non-holonomic robot. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 2–7, San Diego, CA (US), May 1994. http://doc.laas.fr:8889/ows-bin/owa/publishis.rech_doc?langage=fr&clef=9245.
- [3] L. Cesari. *Optimization—theory and applications*, volume 17 of *Application of Mathematics*. Springer-Verlag, 1983.
- [4] H. Delingette, M. Hébert, and K. Ikeuchi. Trajectory generation with curvature constraint based on energy minimization. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 206–211, Osaka (JP), November 1991. http://www-sop.inria.fr/epidaure/personnel/delingette/biblio1.html#delingette_iros.
- [5] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79:497–516, 1957.
- [6] Th. Fraichard, A. Scheuer, and R. Desvigne. From Reeds and Shepp’s to continuous-curvature paths. In *Proc. of the IEEE Int. Conf. on Advanced Robotics*, pages 585–590, Tokyo (JP), October 1999. <http://www.inrialpes.fr/sharp/people/frchard/publications-f.html>.
- [7] Ph. Garnier and Th. Fraichard. A fuzzy motion controller for a car-like vehicle. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 3, pages 1171–1178, Osaka (JP), November 1996. <http://www.inrialpes.fr/sharp/people/frchard/publications-f.html>.
- [8] P. E. Jacobs and J. Canny. Planning smooth paths for mobile robots. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 2–7, Scottsdale, AZ (US), May 1989.
- [9] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi. A stable tracking control method for a non-holonomic mobile robot. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 3, pages 1236–1241, Osaka (JP), November 1991.
- [10] Y. Kanayama and N. Miyake. Trajectory generation for mobile robots. In *Proc. of the Int. Symp. on Robotics Research*, pages 16–23, Gouvieux (FR), 1985.
- [11] K. Kant and S. Zucker. Toward efficient trajectory planning: the path-velocity decomposition. *Int. Journal of Robotics Research*, 5(3):72–89, Fall 1986.
- [12] K. Komoriya and K. Tanie. Trajectory design and control of a wheel-type mobile robot using B-spline curve. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, pages 398–405, Tsukuba (JP), September 1989.
- [13] J.-P. Laumond. Feasible trajectories for mobile robots with kinematic and environment constraints. In *Proc. of the Int. Conf. on Intelligent Autonomous Systems*, pages 346–354, Amsterdam (NL), December 1986.

- [14] J.-P. Laumond, P. E. Jacobs, M. Taïx, and R. M. Murray. A motion planner for non-holonomic mobile robots. *IEEE Trans. Robotics and Automation*, 10(5):577–593, October 1994. http://doc.laas.fr:8889/ows-bin/owa/publishis.rech_doc?langage=FR&clef=6903.
- [15] R. Liscano and D. Green. Design and implementation of a trajectory generator for an indoor mobile robot. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, pages 380–385, Tsukuba (JP), September 1989.
- [16] C. Lobry. Contrôlabilité des systèmes non linéaires. In I. D. Landau, editor, *Outils et modèles mathématiques pour l'automatique, l'analyse de systèmes et le traitement du signal*, volume 1, pages 187–214. CNRS, 1981.
- [17] B. Mirtich. Using skeletons for nonholonomic motion planning. Research Report ESRC 92-16/RAMP 92-6, Department of Electrical Engineering and Computer Science, University of California, Berkeley, June 1992.
- [18] O. Pinchard, A. Liégeois, and F. Pougnet. Generalized polar polynomials for vehicle path generation with dynamic constraints. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 915–920, Minneapolis, MN (US), April 1996.
- [19] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145(2):367–393, 1990.
- [20] A. Scheuer. *Planification de chemins à courbure continue pour robot mobile non-holonyme*. Thèse de doctorat, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), January 1998. http://www.loria.fr/~scheuer/Research/Publishis/index.html#scheuer_these_98.
- [21] A. Scheuer and Th. Fraichard. Continuous-curvature path planning for car-like vehicles. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 997–1003, Grenoble (FR), September 1997. http://www.loria.fr/~scheuer/Research/Publishis/index.html#scheuer_fraichard_iros_97.
- [22] A. Scheuer and Ch. Laugier. Planning sub-optimal and continuous-curvature paths for car-like robots. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 1, pages 25–31, Victoria, BC (CA), October 1998. http://www.loria.fr/~scheuer/Research/Publishis/index.html#scheuer_laugier_iros_98.
- [23] A. Scheuer and M. Xie. Continuous-curvature trajectory planning for manoeuvrable non-holonomic robots. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 3, pages 1675–1680, Kyongju (KR), October 1999. http://www.loria.fr/~scheuer/Research/Publishis/index.html#scheuer_xie_iros_99.
- [24] T. Siméon, S. Leroy, and J.-P. Laumond. Computing good holonomic collision-free paths to steer nonholonomic mobile robots. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 1004–1009, Grenoble (FR), September 1997. http://doc.laas.fr:8889/ows-bin/owa/publishis.rech_doc?langage=FR&clef=19391.

- [25] S. J. Singh and D. H. Shin. Position based path tracking for wheeled mobile robot. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, pages 386–391, Tsukuba (JP), September 1989.
- [26] P. Švestka and M. H. Overmars. Probabilistic path planning. Technical Report UU-CS-1995-22, Utrecht University, P.O.Box 80.089, 3508 TB Utrecht, the Netherlands, May 1995. <http://www.cs.ruu.nl/~markov/research.html>.



Unité de recherche INRIA Lorraine
LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399