Exploration du graphe des états discrétisés pour le robot à deux roues commandées en accélération

Adrien Chan-Hon-Tong

Exploration du graphe des états discrétisés pour le robot à deux roues commandées en accélération

Adrien CHAN-HON-TONG
CEA-LIST-DIASI-LVIC
adrienchanhonton@gmail.com

24 avril 2012

1 Introduction

Cet article s'intéresse à l'étude du problème du guidage optimal en temps du robot à 2 roues commandées en accélération, évoluant dans un plan contenant des obstacles (section 2). Ce problème est un problème bien étudié dans la littérature (section 3). Dans cet article, on s'intéresse au cas particulier où la commande doit être choisie dans un ensemble discret et à des instants discrets régulièrement répartis. Ce problème est alors un problème discret peu étudié dans la littérature, dont on peut trouver la solution optimale par exploration de l'arbre des suites de commande (section 4). Cette exploration étant couteuse, une approche s'appuyant sur les propriétés du graphe des états discrétisés est présentée : dans le cas particulier où l'ensemble des valeurs accessibles sur les 3 dernières composantes du système sont discrètes il est possible de déterminer la solution optimale du problème par l'exploration d'un graphe des états discrétisés (le graphe d'état restant infini) (section 5). Ce nouvel algorithme est complémentaire de l'algorithme basé sur l'exploration de l'arbre des suites de commandes.

2 Modélisation

Le robot à deux roues commandées en accélération est classiquement modélisé par le système suivant :

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\cos(x_3)(x_4+x_5) \\ \frac{1}{2}\sin(x_3)(x_4+x_5) \\ \frac{1}{L}(x_4-x_5) \\ u_1 \\ u_2 \end{pmatrix}$$

Où $U = (u_1, u_2)$ représente la commande, $X = (x_1, x_2, x_3, x_4, x_5)$ représente l'état du système (position du centre, angle, vitesse de chaque roues), et L la distance entre les 2 roues.

La commande doit à chaque instant appartenir à un ensemble $\mathcal{U}$. L'état doit à chaque instant appartenir à un ensemble Ω. On se placera dans le cas où $\Omega = A \times \mathbb{U} \times [-\lambda, \lambda]^2$ (cas d'un robot rond) avec A un ouvert, complémentaire d'un compact de $]-\mu, \mu[^2$. Le but est d'amener le robot d'un état X_I à l'état X_F.

3 Contexte du problème

Il existe deux grands types de problème en automatique. D'une part, on s'intéresse à la recherche d'un chemin permettant d'aller d'un point à un autre en présence d'obstacle. D'autre part, on s'intéresse au guidage optimal de système en l'absence d'obstacle.

Dans le cas général la simple recherche d'un chemin permettant d'aller de X_I à X_F est un problème difficile [8]. Pour résoudre ce type de problème, la méthode la plus commune est de chercher un chemin polygonal inclus dans l'espace libre, car l'existence d'un chemin polygonal est équivalent dans le cas où l'espace libre est un ouvert et que le système est commandable en temps petit à l'existence d'une commande admissible permettant de joindre le point de départ et le point d'arrivée. Cette recherche s'effectue souvent via des méthodes de grille, ou des tirages aléatoire de points [7]. Dans le cas 2D, le problème n'est pas NP-complet. La méthode du graphe de visibilité [6], permet de trouver un chemin polygonal entre deux point dans le plan en présence d'obstacles polygonaux.

D'un autre coté, le problème du guidage optimal en temps de système en l'absence d'obstacle repose généralement sur des théorèmes généraux d'automatique. Le problème du robot à 2 roues commandées en accélération en l'absence d'obstacle est un problème qui a été bien étudié [1, 2, 3, 5] dans la littérature et qui est aujourd'hui résolu [2, 3]. En effet, l'utilisation de théorème généraux d'automatique a permis de démontrer que les trajectoires optimales pour le robot à 2 roues sont "bang-bang" [5] (ce qui reste vrai en cas d'un espace libre ouvert). Les spécificités du problème du robot à deux roues ont alors permis de démontrer que pour tout couple de configuration X_I, X_F il suffit de moins de 4 changement de commande pour aller de X_I à X_F [2, 3].

Néanmoins, il existe peu de travaux essayant de résoudre conjointement recherche de chemin en présence d'obstacles et recherche de la commande temps optimal. [4] propose une méthode approchée pour déterminer un chemin en temps optimal en présence d'obstacle. Cette méthode est basée sur l'exploration d'un graphe discrétisé pour un système quelconque. Dans la suite de cet article, on montre que dans le certain cas et pour le robot à deux roues commandées en accélération, la méthode présentée dans [4] peut être modifiée pour produire non plus une solution approchée mais une solution exacte.

Pour cela on introduit une nouvelle contrainte par rapport à la modélisation présenté en section 2 : on suppose qu'on ne peut choisir la commande qu'aux temps $\{n\delta_t / n \in \mathbb{N}\}$ et que $\mathcal{U}$ est un ensemble fini. De plus on relâche la contrainte sur l'état d'arrivée : l'objectif est d'arriver dans un état "proche" de X_F.

4 Résolution par exploration en largeur de l'arbre des suites de commandes

La contrainte qui impose de ne pouvoir changer de commande qu'aux temps $\{n\delta_t / n \in \mathbb{N}\}$ et que la commande doive appartenir à un ensemble fini donne au problème un caractère discret en ce sens qu'une solution n'est plus une fonction qui a chaque instant associe une commande, mais seulement une suite finie de commandes de durée δ_t.

On notera pour tout point $X \in \Omega$ $\mathcal{U}_X$ l'ensemble des suites de commandes admissibles depuis le point X. Ces ensembles sont définis récursivement par les 2 conditions :

1. U est admissible depuis X si l'application de U pendant une durée δ_t depuis le point X ne fait jamais sortir la trajectoire de Ω
2. si U est admissible depuis X et que l'application de U depuis X pendant un temps δ_t conduit au point $\widehat{X}$ alors pour toutes suites $U_2, ..., U_r$ dans $\mathcal{U}_{\widehat{X}}$, la suite concaténée $U, U_2, ..., U_r$ est dans $\mathcal{U}_X$

On notera $(U_1, ..., U_r)(X)$ pour désigner le point résultant de l'application de la suite de commande $(U_1, ..., U_r)$ depuis le point X pour toutes suites $(U_1, ..., U_r) \in \mathcal{U}_X$.

L'objectif est donc de trouver $(U_1, ..., U_p) \in \mathcal{U}_{X_I}$ une suite admissible depuis l'état initial tel que $(U_1, ..., U_p)(X_I) = X_F$ (ou $(U_1, ..., U_p)(X_I) \approx X_F$ si on relache cette contrainte) et tel que p soit le plus petit possible. Or il n'existe qu'un nombre fini de suite de la forme $(U_1, ..., U_p)$. Il est donc possible de trouver une solution optimale $(U_1, ..., U_p)$ par l'exploration en largeur de l'arbre des suites de commandes (le sommet est X_I, chaque noeud appartient à Ω, chaque arrête est une commande admissible depuis le noeud, on cherche une feuille étiquetté par X_F).

Cet algorithme peut être modifier pour retourner une solution optimale en temps qui de plus maximise la marge vis à vis des obstacles et de l'arrivée. Une trajectoire possède une marge de ν si et seulement si elle est toujours une solution si on augmente les obstacles de ν et qu'on diminue la boule d'arrivée de ν (dans le cas où l'on cherche à être proche de X_F et non égal à X_F). On notera A_ν l'espace libre correspondant à A quand les obstacles sont augmentés de ν (classiquement si O est l'ensemble des obstacles $O \oplus B_f(0, \nu)$ est l'ensemble des obstacles augmentés de ν). Remarquons que la marge est nécessairement strictement positive car A est un ouvert.

Cet algorithme a une complexité temporelle de la forme $O(|\mathcal{U}|^p)$ où p est la solution optimale du problème. Cette exploration est donc couteuse et on va s'intéresser dans la suite à une approche permettant de déterminer la solution optimale en un temps polynomiale en p.

5 Graphes des états discrétisés

5.1 Heuristiques associés

Un graphe infini est naturellement associé au problème étudié : l'ensemble des états du graphe est constitué de l'ensemble des états de Ω et l'ensemble des arcs est constitué de l'ensemble des couples $\left(X, \widehat{X}\right)$ tel qu'il existe $U \in \mathcal{U}_X$ tel que $(U)(X) = \widehat{X}$. La solution optimale du problème étudié est associé à un plus court chemin dans ce graphe entre X_I et X_F. Mais le plus court chemin dans ce graphe n'est pas calculable avec les algorithmes classiques d'exploration de graphe. De plus, même le sous-graphe constitué des sommets atteignable depuis X_I peut être infini.

Pour contourner cela, il est possible de discrétiser l'espace des états en construisant un graphe fini $G = (V, E)$ où V est une partition fini de Ω et où E contient au moins tous les arcs $(\Upsilon, \Gamma) \in V^2$ vérifiant $\exists X \in \Upsilon / \exists U \in \mathcal{U}_X / (U)(X) \in \Gamma$. Tous les graphes construit sur ce modèle ont en effet la propriété suivant : $\forall \Upsilon, \Gamma \in V \ \left(X, \widehat{X}\right) \in \Upsilon \times \Gamma$ le fait qu'il existe une suite de commandes $U_1, ..., U_r \in \mathcal{U}_X$ tel que $(U_1, ..., U_r)(X) = \widehat{X}$ implique qu'il existe un chemin de longueur inférieur où égale à r dans le graphe V reliant Υ, Γ. Ainsi dans un graphe construit sur ce modèle, et où il existe 2 partitions Υ, Γ avec $X_I \in \Upsilon$ et $X_F \in \Gamma$ alors la longueur du plus court chemin entre Υ, Γ (en arc) dans le graphe est une heuristique minorante de la longueur de la plus petite suite de commande permettant d'aller de X_I à X_F.

5.2 Invariance de la dynamique aux translations

Le système du robot à 2 roues commandées en accélération est vérifie une propriété d'invariance sur ses 2 premières composantes. En effet pour tous nombres a, b et pour tous états X, on a la propriété suivante : si $(U_1, ..., U_r) \in \mathcal{U}_X \cap \mathcal{U}_{X+(a,b,0,0,0)}$ alors $(U_1, ..., U_r)(X + (a, b, 0, 0, 0)) = (U_1, ..., U_r)(X) + (a, b, 0, 0, 0)$.

Dans le cas particulier où simultanément l'ensemble des nombres de $\mathcal{U}$ et de $Q = \{(X_{I,4} - X_{I,5})\delta_t\} \cup \{\pi\} \cup \left\{\frac{u-v}{L}\delta_t^2 / u, v \in \mathcal{U}\right\}$ sont commensurables entre eux (dans chaque ensemble), le système a la propriété que l'ensemble des valeurs atteignables par les 3 dernières composantes des vecteurs d'états par application de l'ensemble de toutes les suites de commandes admissibles est discret (et donc fini car Ω force ces valeurs à être bornée). En effet les valeurs des deux dernières composantes sont de la forme : $X_{I,i} + (u_1 + u_2 + ... + u_r)\delta_t$ avec $u_1, ..., u_r \in \mathcal{U}$ et les valeurs de la troisième composante sont de la forme $X_{I,4-5} + (X_{I,4} - X_{I,5})r\delta_t + \left(\frac{u_1-v_1}{L} + ... + \frac{u_r-v_r}{L}\right)\frac{\delta_t^2}{2} + 2k\pi$. Aussi dans le cas où tous les éléments de $\mathcal{U}$ de Q sont commensurables, ces ensembles se réécrivent $X_{I,4-5} + pgcd(\mathcal{U})\delta_t\mathbb{Z}$ et $X_{I,3} + pgcd(Q)\mathbb{Z}$ c'est à dire des ensembles discrets.

Dans ce cas particulier, seul les deux premières composantes ont un ensemble de valeurs accessibles infini. Or comme on l'a rappelé la dynamique est invariante à ces deux composantes. Il est alors naturel de munir Ω d'une nouvelle norme

$\|\|$ définit par

$$\left\|X-\widehat{X}\right\| = \begin{cases} \infty & \text{si } \sum\limits_{i=3}^{5}\left(X_i-\widehat{X}_i\right)^2 \neq 0 \\ \max\left\{\left|X_1-\widehat{X}_1\right| + \left|X_2-\widehat{X}_2\right|\right\} & \text{sinon} \end{cases}$$

Muni de cette norme, le système vérifie la propriété suivante : si $(U_1,...,U_r) \in \mathcal{U}_X \cap \mathcal{U}_{\widehat{X}}$ alors $\left\|(U_1,...,U_r)(X) - (U_1,...,U_r)\left(\widehat{X}\right)\right\| = \left\|X-\widehat{X}\right\|$.

5.3 Résolution par exploration du graphe des états discrétisés

Dans le cas particulier présenté à la section 4.2 l'espace des états peut se réecrire : $\Omega = A \times \Theta$ où Θ est un ensemble fini. Ω est muni de la norme $\|\|$ défini à la section 4.2 et l'objectif est d'aller depuis X_I dans $B_o(X_F, \delta_F)$ c'est à dire d'arriver dans un état X tel que $\|X - X_F\| < \delta_F$. Supposons qu'il existe une suite de commande permettant de remplir notre objectif, notons p le plus petit nombre de commande qui permettent de remplir l'objectif et notons δ_X la marge maximale parmis les suites de commande de taille p solutions du problème. Cela signifie qu'il existe toujours une suite de commande de taille p permettant de remplir notre objectif même si tous les obstacles sont augmenté de δ_X et l'arrivée réduite de δ_X c'est à dire qu'on arrive dans $B_o(X_F, \delta_F - \delta_X)$.

Soit $\epsilon = \frac{1}{2}\frac{\delta_X}{p+1}$ et soit l'ensemble de sommet

$$V = \left\{[a\epsilon, a\epsilon+\epsilon] \times [b\epsilon, b\epsilon+\epsilon] \subset A_{\frac{\delta_X}{2}} / (a,b) \in \mathbb{Z}\right\} \times \Theta$$

Pour tout sommet $u \in V$ de la forme $([a\epsilon, a\epsilon+\epsilon] \times [b\epsilon, b\epsilon+\epsilon], \theta)$, on note $X_u = \left(a\epsilon + \frac{\epsilon}{2}, b\epsilon + \frac{\epsilon}{2}, \theta\right)$.

Soit l'ensemble d'arc $E \subset V^2$ tel que l'arc (u,v) appartient à E si et seulement si il existe une commande U appartenant à $\mathcal{U}_{X_u}$ tel que $(U)(X_u) \in v$.

Notons G le graphe (V,E), v_I le sommet du graphe qui contient X_I et v_F l'ensemble de sommet du graphe inclus dans $B_o\left(X_F, \delta_F - \frac{\delta_X}{2}\right)$.

Nous allons montrer que l'existance d'une suite de commande solution de taille p et de marge δ_X implique l'existance d'un chemin de taille p entre v_I et v_F et réciproquement que l'existance d'un chemin de taille p entre v_I et v_F dans ce graphe permet de récupérer une solution de taille p (sans garantie sur la marge de la solution obtenue).

5.3.1 Existance d'un chemin

Soit $(U_1,...,U_p)$ une solution de taille p et de marge δ_X, Soit les points $X_1,...,X_{p+1}$ définis par : $X_1 = X_I$ et $X_{i+1} = (U_i)(X_i) = (U_1,...,U_i)(X_I)$. Soit les sommets $v_1,...,v_{p+1}$ et les points $X_{v_1},...,X_{v_{p+1}}$ définis par : $v_1 = v_I$ et pour tous $i \in \{1,...,p\}$, v_{i+1} est le sommet contenant $(U_i)(X_{v_i})$. Cette suite est bien définie, car par une récurrence immédiate, on établit que pour tous

$i \in \{1, ..., p+1\}$ $\|X_{v_i} - X_i\| \leq i\epsilon \leq \frac{\delta_X}{2}$ et cela implique que $U_i \in \mathcal{U}_{X_{v_i}}$ puisque $(U_1, ..., U_p)$ a une marge de δ_X, et que pour tous $i \in \{1, ..., p\}$, v_{i+1} est bien dans le graphe puisque X_{v_i} est a une distance $\frac{\delta_X}{2}$ de tous obstacles.

Enfin $\left\|X_{v_{p+1}} - X_{p+1}\right\| \leq \frac{\delta_X}{2}$ or on a l'inégalité $\|X_F - X_{p+1}\| \leq \delta_F - \delta_X$ donc $\left\|X_F - X_{v_{p+1}}\right\| \leq \delta_F - \frac{\delta_X}{2}$ donc $v_{p+1} \in v_F$. Ainsi $v_1, ..., v_{p+1}$ est bien un chemin de v_I à v_F de taille p dans le graphe (V, E)

5.3.2 Extraction d'une solution à partir d'un chemin

Réciproquement si $v_I = v_1, ..., v_{p+1} \in v_F$ est un chemin dans (V, E) associé aux arcs $U_1, ..., U_p$ alors on peut construire la suite de point $X_1, ..., X_{p+1}$ avec $X_1 = X_I$ et $X_{i+1} = (U_i)(X_i)$ car par le même raisonnement que dans la section 4.3.1 $\forall i \in \{1, ..., p\}$ on a $\left\|X_{i+1} - X_{v_{i+1}}\right\| \leq \|(U_i)(X_i) - (U_i)(X_{v_i})\| + \left\|(U_i)(X_{v_i}) - X_{v_{i+1}}\right\|$ on établit donc l'inégalité $\left\|X_{i+1} - X_{v_{i+1}}\right\| \leq \|X_i - X_{v_i}\| + \epsilon$ et donc $\left\|X_{i+1} - X_{v_{i+1}}\right\| \leq (i+1)\varepsilon \leq \frac{\delta_X}{2}$. Or on a l'égalité $\|X_{p+1} - X_F\| \leq \left\|X_{p+1} - X_{v_{p+1}}\right\| + \left\|X_{v_{p+1}} - X_F\right\| \leq \delta_F$. Ainsi $U_1, ..., U_p$ est une solution de taille p du problème de commande optimale.

5.4 Complexité

Une simple exploration du graphe (V, E) construit dans la section 4.3 permet de déterminer une solution au problème posé en un temps en $O\left(|V|^3\right)$. La taille de l'ensemble V peut s'exprimer en fonction des données du problème, ainsi la complexité de l'algorithme basé sur l'exploration de graphe est en $O\left(|\Theta|^3\left(\frac{\mu}{\delta_X}(p+1)\right)^6\right)$. Cette complexité dépend fortement de δ_X, $|\Theta|$ et de μ mais elle n'est que polynomiale en p.

6 Conclusion

Dans cet article, on a montré que si certaines grandeurs intrinsèques du problème sont commensurables, l'ensemble des valeurs accessibles par les 3 dernières composantes du système est un ensemble fini. Cette propriété associée avec le fait que la dynamique du système est indépendante des valeurs des deux premières composantes, permet de résoudre le problème par l'exploration d'un graphe d'états discrétisés. L'algorithme de calcul de la commande optimale par l'exploration d'un graphe d'états discrétisés que nous présentons dans cet article, est complémentaire à l'algorithme trivial d'exploration exhaustive de l'ensemble des suites de commandes. En effet l'algorithme d'exploration des suites de commandes dépend exponentiellement de la taille de la solution optimale, alors que l'algorithme d'exploration d'un graphe d'états discrétisés ne dépend que polynomialement de la taille de la solution optimale. Il dépend cependant de paramètre supplémentaire comme la marge de la solution optimale.

Dans de futurs travaux, nous chercherons à trouver un moyen de trouver une borne inférieure à la marge de la solution optimale en fonction de la configuration des obstacles, une telle borne permettrait de pouvoir borner a priori le temps de calcul de l'algorithme.

Références

[1] Jong-Suk Choi and Byung Kook Kim, Near-Time-Optimal Trajectory Planning for Wheeled Mobile Robots with Translational and Rotational Sections, IEEE International Conference on Robotics and Automation, 2001

[2] David B. Reister, Time optimal trajectories for a two wheeled robot, Technical Report, 1990

[3] David B. Reister and François G. Pin, Time-Optimal Trajectories for Mobile Robots With Two Independently Driven Wheels, The International Journal of Robotics Research, 1994

[4] Jean-Claude Latombe, A Fast Path Planner For a Car-Like Indoor Mobile Robot, AAAI, 1991

[5] J. P. Laumond and S. Sekhavat and F. Lamiraux, Guidelines in nonholonomic motion planning for mobile robots, Robot Motion Plannning and Control, 1998

[6] J.-C. Latombe and R. Motwani and P. Raghavan, Nonholonomic path planning for pushing a disk among obstacles, PIEEE International Conference on Robotics and Automation, 1997

[7] J. P. Laumond, Probabilistic path planning, Robot Motion Planning and Control, 1998

[8] J. F. Canny, The complexity of robot motion planning, 1988