Planification de Tâches de Manipulation Fine d'Objets Rigides

Jean-Philippe Saut *, **Anis Sahbani** † et **Véronique Perdereau** ‡
Université Pierre et Marie Curie-Paris6, EA 2385,
LISIF, 3 rue Galilée, Ivry-sur-seine, 94200 France
∗ jean-philippe.saut@lisif.jussieu.fr, † sahbani@ccr.jussieu.fr, ‡ vperd@ccr.jussieu.fr

Résumé : Cet article propose une nouvelle approche du problème de la planification de tâches de manipulation fine d'objets rigides par une main robotisée. A partir de deux configurations du système main+objet, les algorithmes du planificateur déterminent une trajectoire pour les doigts et l'objet, qui permet de relier ces deux configurations, tout en évitant les collisions indésirables et en respectant les contraintes particulières au problème : contraintes cinématiques de non glissement des points de contact et contraintes de stabilité de la prise.
Son originalité est de proposer une nouvelle structuration de l'espace de recherche, qui introduit notamment les sous-espaces de l'ensemble des prises à un nombre donné de doigts, et d'utiliser des chemins directement dans ces sous-espaces au lieu d'employer uniquement des séquences "reconfiguration de prise-déplacement de l'objet" comme cela est fait habituellement.
La complexité du problème impose néanmoins de restreindre l'espace de recherche en émettant des hypothèses simplificatrices. C'est pourquoi nous supposons le système quasi-statique, les effets inertiels n'intervenant pas pour des mouvements suffisamment lents.
Mots-clés : intelligence artificielle, planification de mouvement, manipulation fine.

1 INTRODUCTION

La manipulation d'objets par une main robotisée est un sujet qui passionne depuis longtemps les chercheurs en robotique. Cet engouement s'explique probablement par les grandes possibilités d'interaction avec l'environnement qu'apportent les capacités manuelles et parce que ces capacités entraînent immanquablement leur comparaison avec celles de l'homme, qui est bien souvent vu comme l'idéal à atteindre par les robots.
Pour marquer sa différence avec la manipulation d'un objet maintenu par une prise fixe, faite à l'aide de mouvements de la main obtenus par rotation du poignet ou mouvement du bras complet, on appelle manipulation fine ou dextre la manipulation qui se fait uniquement avec les doigts. Ce type de manipulation est réalisé à l'aide de nombreux petits déplacements de l'objet et de multiples sauts de doigts. Elle offre la possibilité de modifier la pose de l'objet (la donnée d'une position et d'une orientation) à l'aide de mouvements se faisant dans un espace restreint, contrairement aux mouvements de l'objet qui seraient entraînés par un mouvement du bras, doigts immobiles. Par exemple, un humain, après avoir saisi un stylo, peut le réorienter par une succession de petits mouvements de doigts.
La manipulation dextre en robotique implique des domaines de recherches variés tels que la mécanique (conception de la main, étude de la mécanique des contacts), l'automatique (contrôle des doigts en force et/ou en position) et, à un niveau supérieur, la planification de mouvement. C'est à ce dernier domaine que se rattache cet article.
Dans le cas de la manipulation fine, le but de la planification de mouvement est de trouver des trajectoires pour les doigts et pour l'objet, permettant d'amener l'objet d'une pose à une autre, tout en respectant certaines contraintes. Ces contraintes sont de différents types : géométriques (pas de collision entre les différents corps du système main+objet sauf aux points de contact), cinématiques (la vitesse des points de contact doit suivre certaines restrictions imposées par le modèle de contact choisi), et dynamiques (les forces de saisie doivent assurer la stabilité de la prise tout au long du mouvement).

2 ETAT DE L'ART DES METHODES DE PLANIFICATION POUR LA MANIPULATION DEXTRE

Durant les dernières décennies, de nombreux travaux de recherche ont concerné la planification de mouvement de manipulation dextre.
Li et Canny [Li,1989] ont présenté une formulation du problème mais sans y joindre de méthode de résolution. Une autre formulation a été proposée plus récemment dans [Han, 2000]. Trinkle et Hunter [Trinkle,1991] ont proposé une description qualitative des prises, en listant les contacts entre les éléments de l'objet saisi et ceux de la main. Ces éléments sont décrits sous la forme de sommets et d'arêtes. Les descriptions obtenues sont appelées *contact formation* (CF). Après avoir construit un ensemble de CF, une méthode de planification dans l'espace des configurations est utilisée pour relier les CF de cet ensemble et construire un arbre. Cette méthode n'était appliquée qu'à un système de manipulation comportant un faible nombre de degrés de liberté. Par ailleurs, plus les surfaces de l'objet et des doigts sont complexes, plus le nombre de CF est grand et plus le problème est dur à résoudre.

Montana [Montana,1995] propose une description complète, dans l'espace des configurations, de la cinématique de la manipulation dextre et présente l'exemple de la génération des mouvements pour la rotation d'un bâton.
Remarquant que les hommes accomplissent toutes leurs tâches manuelles avec un nombre fini de prises, Zhang *et al.* [Zhang,1996] ont identifié un ensemble de "prises canoniques" (*canonical grasps* ou CG) et déterminé celles d'entres elles qui peuvent être reliées par une méthode locale. Les CG servent à construire un graphe dont elles constituent les noeuds. Des lois de contrôle de bas niveau sont conçues pour réaliser les transitions entre les CG. Zhang *et al.* ont expérimenté leur méthode sur une main à trois doigts et six degrés de liberté.
Han et Trinkle [Han, 1998] aussi se sont intéressés au changement de prise et à la trajectographie des doigts. Ils ont présenté une méthode pour planifier la manipulation d'une sphère avec une main à trois doigts. Un doigt doit changer de point de contact quand il est près de la limite de son domaine atteignable ou lorsqu'il ne peut pas assurer la propriété de *force closure* de la prise avec aucun des deux autres doigts. A la même période, Rus [Rus,1997] a mis au point un algorithme appelé *finger tracking*, qui prend en compte toute la dynamique de la manipulation. L'idée principale de cet algorithme est d'utiliser deux doigts dont la position dans le repère du monde est fixe et un troisième doigt dont le mouvement le long de la surface de l'objet entraîne la rotation de celui-ci.
Cherif et Gupta [Cherif, 1999] utilisent ce même principe pour planifier la réorientation d'un objet convexe. Trois doigts sont fixes et le mouvement d'un quatrième fait tourner l'objet.
Plus récemment, Sudsang et Phoka [Sudsang,2003] ont développé une méthode pour planifier les reconfigurations de prise d'une main planaire à quatre doigts manipulant un polygone. Cette méthode s'appuie sur la construction d'un *switching graph*. Chaque noeud de ce graphe est un ensemble de prises particulières appelées *prises concurrentes*. Cette technique élégante permet de construire une séquence de prises permettant d'aller de l'une à l'autre tout en assurant la propriété de *force closure* des prises. Toutefois, cette méthode concerne seulement la reconfiguration de prise et ne s'intéresse pas au mouvement de l'objet ; elle n'a été testée que dans le plan.
Plusieurs auteurs [Goodwine2002, Harmati, 2002] se sont intéressés au problème de la manipulation fine en utilisant des techniques issues de la théorie du contrôle des systèmes stratifiés. Les systèmes stratifiés sont des systèmes dont l'espace des configurations est constitués de différentes surfaces appelées strates, qui s'intersectent pour former d'autres strates de dimension inférieure. Une strate correspondra ici à un type de prise, caractérisé par les doigts qui y participent. Le principe du contrôle des systèmes stratifiés est de définir un espace d'état commun où tous les champs de vecteurs, qui sont définis pour chaque strate, peuvent être considérés.
Yashima [Yashima, 2003] a conçu une architecture de planification probabiliste basée sur le passage entre différents modes de contact. Tous les modes de contact sont considérés (glissement, glissement avec roulement, avec rotation, *etc.*). Un planificateur global construit un arbre dont chaque noeud est une configuration de l'objet, selon la méthode probabiliste RRT [Lavalle, 2000]. Pour tenter de relier deux noeuds de cet arbre, un planificateur local génère d'abord différentes séquences de types de contact pour les différents doigts puis, pour chacune de ces séquences, utilise le problème dynamique inverse pour calculer les couples à exercer sur les articulations des doigts, qui permettent de déplacer l'objet d'un noeud/configuration à l'autre. Cette méthode a l'avantage de garantir la faisabilité, tant cinématique que dynamique, de la solution trouvée, bien que cela se fasse au prix de temps de calcul très importants. Un exemple de résolution de problème est présenté, en simulation, avec une main à trois doigts manipulant une sphère. Cette méthode ne s'intéresse ni au problème de la reconfiguration de prise, indispensable pour la plupart des formes d'objet, ni aux collisions des doigts avec des obstacles ou avec l'objet.
De toutes les méthodes existantes, rares sont celles à proposer une approche globale du problème de la manipulation dextre. En effet, la plupart émettent des hypothèses simplificatrices importantes : on contraint les prises à appartenir à un ensemble discret, les doigts sont considérés comme de simples points, les objets doivent être convexes, *etc*. Celles qui s'intéressent au problème dans sa globalité sont plutôt des méthodes adaptées à des cas particuliers tel que la manipulation d'une sphère avec trois doigts. La méthode que nous proposons dans cet article a pour ambition d'être vraiment globale. Sa seule restriction est de supposer le système "main+objet" comme étant quasi-statique. Elle s'appuie sur des méthodes probabilistes et garantit, en probabilité, qu'une solution sera trouvée si elle existe.
Inspirés par la méthode de planification de tâches de manipulation "simple" de [Sahbani,2002, Siméon,2003], nous suggérons l'utilisation d'une structuration particulière de l'espace des configurations pour guider la recherche d'une solution. Cette structuration de l'espace des configurations constitue le premier apport de cet article. Elle est détaillée dans la section suivante. Le deuxième apport est une approche générique et originale permettant une exploration efficace et guidée de l'espace de configuration. Cette approche est présentée dans la section 4.

3 STRUCTURATION DE L'ESPACE DE RECHERCHE

3.1 Définition du problème de planification de tâches de manipulation fine

Soit une main robotisée $\mathcal{H}$ à n doigts manipulant un objet rigide $\mathcal{O}$ à six degrés de liberté. Le but de la planification de tâches de manipulation fine est de déterminer une séquence de mouvements des doigts et de reconfigurations de prise permettant de déplacer l'objet saisi $\mathcal{O}$

d'une pose initiale à une pose finale. Une pose est la combinaison d'une position et d'une orientation c'est-à-dire un vecteur $\mathbf{p} = (x, y, z, \alpha, \beta, \gamma) \in \mathbb{R}^6$ où (x, y, z) est la position de $\mathcal{O}$ et (α, β, γ) une paramétrisation de son orientation (par exemple, les angles d'Euler). Une prise à k doigts est représentée par un vecteur $\mathbf{g}_k = (u_1, v_1, \ldots, u_k, v_k) \in \mathbb{R}^{2k}$ où (u_i, v_i) sont les coordonnées de la position du i-ème contact –supposé ponctuel– sur la surface de l'objet. Le vecteur de configuration du i-ème doigt est $\mathbf{a}_i = (\theta_{i,1}, \ldots, \theta_{i,m_i}) \in \mathbb{R}^{m_i}$ où $\theta_{i,j}$ est le paramètre articulaire de la j-ème liaison du doigt i et m_i son nombre de liaisons. Nous utilisons la nomenclature *doigt libre* pour un doigt qui ne participe pas à la prise et *doigt lié* pour un doigt participant à la prise.

La configuration du système "$\mathcal{O}$ saisi par $\mathcal{H}$ avec k doigts" est complètement déterminée par la connaissance de $\mathbf{p}$, $\mathbf{g}_k$ et des $\mathbf{a}_i$ des doigts libres. Ce vecteur de configuration est suffisant pour connaître l'état du système complet sous réserve que les modèles géométriques des doigts aient une solution unique (si ce n'est pas le cas, on peut toujours les restreindre arbitrairement ou selon un critère de qualité de prise, par exemple).

Soit $\mathcal{CS}_{free}$ l'espace de toutes les configurations libres. Une configuration est libre si elle n'induit pas de contact entre $\mathcal{O}$ et $\mathcal{H}$ ailleurs que sur le bout des doigts (pour assurer une prise), ni de contact entre $\mathcal{O}$ ou $\mathcal{H}$ et des obstacles de l'environnement, ni de contact entre deux corps de $\mathcal{H}$ qui ne sont pas liés par une articulation.

Une caractéristique importante du système $\mathcal{H} + \mathcal{O}$ est la stabilité d'un point de vue mécanique de ses configurations. En effet, certaines prises ne permettent pas à $\mathcal{H}$ de maintenir l'objet. Ceci est dû au fait que la main et l'objet ne sont pas soudés au niveau des contacts mais que les forces de contact doivent obéir à certaines contraintes dépendant du modèle de contact. Nous choisissons le modèle, couramment utilisé, du contact ponctuel avec frottement obéissant à la loi de Coulomb. Soit $\overrightarrow{f}$ la force de contact exercée par un solide S_1 sur un solide S_2. Pour que le contact ne soit pas rompu par un glissement, $\overrightarrow{f}$ doit respecter les inégalités suivantes :

$$\overrightarrow{f} \cdot \overrightarrow{n} > 0 \tag{1}$$

$$\sqrt{f_{t_1}^2 + f_{t_2}^2} \leq \mu f_n \tag{2}$$

où $\overrightarrow{n}$ est la normale au point de contact, dirigée vers S_2, f_{t_1} et f_{t_2} sont les composantes tangentielles de $\overrightarrow{f}$, f_n sa composante normale et μ le coefficient de frottement du contact. Géométriquement, cette inégalité est équivalente à l'appartenance du vecteur $\overrightarrow{f}$ au cône d'axe $\overrightarrow{n}$ et de demi-angle d'ouverture $atan(\mu)$ (figure 1). La nature ponctuelle des contacts entraîne qu'ils ne peuvent transmettre que des forces et pas de moments. Par conséquent, trois contacts ponctuels avec frottement, au moins, sont nécessaires pour assurer la stabilité de la prise.

La stabilité de la prise est un problème dynamique mais étant donné la lenteur des mouvements mis en jeu pendant la manipulation d'un objet, on peut le considérer

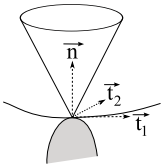

FIG. 1 – Cône de frottement du contact entre deux solides.

comme quasi-statique. Dans ce cas, la stabilité d'une prise ne dépend que de sa configuration. Une propriété fondamentale à prendre en compte est alors la propriété de *force closure*, abondamment traitée dans la littérature (par exemple dans [Bicchi, 1995, Liu, 1999]). La propriété de *force closure* dépend de la position des contacts et de leurs normales par rapport à l'objet. Une prise est *force closure* si et seulement si elle peut exercer n'importe quel torseur dynamique (résultante et moment) en appliquant des forces de contact appropriées. Une telle prise permet de résister à n'importe quelle perturbation exercée sur l'objet, pour peu que les doigts puissent appliquer des forces suffisamment importantes. Pour être sûr que ces forces peuvent être appliquées, on peut par exemple introduire une limite supérieure aux amplitudes des forces que peuvent fournir les doigts dans le test de *force closure* ([Bicchi, 1995]). On considérera qu'une prise est stable si elle vérifie la propriété de *force closure*.

Nous pouvons maintenant définir le problème planification en manipulation fine :

Un problème de planification de tâches de manipulation fine, consiste à trouver un ensemble continu de configurations appartenant à $\mathcal{CS}_{free}$, reliant une configuration initiale à une configuration finale, telles que les prises associées soient stables. Un tel ensemble de configurations est appelé *chemin de manipulation fine*.

3.2 Une nouvelle structuration de l'espace des configurations

Dans le cadre de la manipulation fine, les configurations admissibles de $\mathcal{CS}_{free}$ sont celles qui correspondent à des prises de l'objet, les autres correspondant à des états instables. Par conséquent, nous introduisons un sous-espace fondamental, le sous-espace GS_k (pour Grasp Subspace k), qui est le sous-espace de toutes les prises de $\mathcal{O}$ par une combinaison de k doigts liés. La dimension de ce sous-espace est $2k + 6 + \sum_{i=1}^{n-k} m_i$: chacun des k doigts liés a deux degrés de liberté, $\mathcal{O}$ en a 6 et les $(n - k)$ doigts libres en ont chacun m_i. L'espace des configurations du système $\mathcal{H} + \mathcal{O}$ est donc composé des différents sous-espaces GS_k avec $k \in [\![0, \ldots, n]\!]$. Comme une prise à $k + 1$ doigts est un cas particulier de prise à k doigts (un des doigts libres se positionne sur la surface de l'objet), on a $GS_{k+1} \subset GS_k, \forall k \in [\![0, \ldots, n-1]\!]$.

La nécessité pour les différents corps de $\mathcal{H}$ et de $\mathcal{O}$ d'avoir des déplacements continus dans l'espace, rend impossible la connexion de certaines configurations du

même sous-espace GS_k sans passer par la restriction de ce sous-espace à GS_{k+1} ou par GS_{k-1} et ce quelque soit le type de chemin considéré, qui doit assurer cette continuité. Ceci est illustré par la figure 2 qui montre, pour l'exemple d'une main à cinq doigts, que pour connecter deux configurations de GS_4, il faut passer par des configurations intermédiaires dans $GS_5 \backslash GS_4$ ou GS_3.

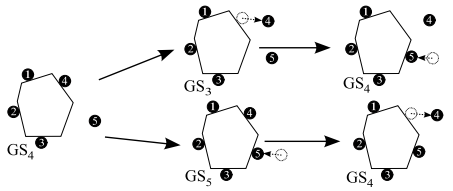

FIG. 2 – Passage d'une classe de prise à 4 doigts à une autre pour une main à 5 doigts : la continuité des mouvements des doigts impose le passage intermédiaire à une prise à 5 doigts ou à 3 doigts.

Il y a autant de choix pour les doigts liés d'une prise à k doigts que de combinaisons de k éléments parmi n. Chaque sous-espace $GS_k \backslash GS_{k+1}$ a donc C_n^k composantes connexes et GS_k possède k composantes connexes de $GS_{k+1} \backslash GS_{k+2}$.
En réalité C_n^k est le nombre minimum de composantes connexes de chacun des sous-espaces $GS_k \backslash GS_{k+1}$ puisque la présence d'obstacles, de butées articulaires ou d'autres contraintes entraîne la décomposition en composantes connexes plus nombreuses.
L'existence de contraintes sur les contacts (non glissement) nous mène à introduire deux types fondamentaux de chemin local, qui correspondent à des sous-tâches de manipulation élémentaires : la reconfiguration de prise et le déplacement de l'objet. Nous appellerons le premier *chemin de ressaisie* et le second *chemin de transfert*.
Pendant un chemin de ressaisie, l'objet est immobile et maintenu par au moins trois doigts (figure 3) –le nombre de contacts minimum permettant d'assurer la stabilité de la prise dans le cas de contacts ponctuels avec frottement. Un chemin de ressaisie permet de changer la prise en maintenant la pose de l'objet constante. Le chemin de ressaisie correspond au passage d'un sous-espace GS_k à un sous-espace GS_{k-1} qui est de dimension supérieure (un doigt libre a plus de degrés de liberté qu'un doigt lié) puis au retour à GS_k, pour chacun des doigts qui doit changer de point de contact.
Pendant un chemin de transfert (figure 3), l'objet change de configuration tandis que la prise reste inchangée. La pose de l'objet entre les configurations initiale et finale est choisie le long d'un chemin linéaire les reliant. Les configurations des doigts sont calculées à partir de leurs modèles géométriques inverses, ce que l'on peut faire car on connaît les positions des contacts sur l'objet et la pose de celui-ci. Un chemin de transfert est un type de chemin particulier dans $GS_k \backslash GS_{k+1}$.
Deux configurations $q_1 = (\mathbf{p}_1, \mathbf{g}_{k_1})$ et $q_2 = (\mathbf{p}_2, \mathbf{g}_{k_2})$, toutes deux appartenant à GS_k, peuvent donc être connectées par un chemin de transfert seulement si $\mathbf{g}_{k_1} = \mathbf{g}_{k_2}$ et par un chemin de ressaisie seulement si $\mathbf{p}_1 = \mathbf{p}_2$.

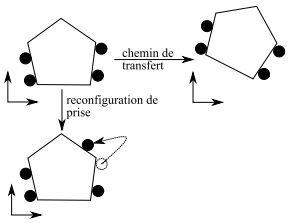

FIG. 3 – Exemples de chemins de ressaisie et de transfert pour une main à 4 doigts dans le plan.

Les méthodes de planification de tâches de manipulation dextre présentées jusqu'à aujourd'hui, discrétisent $\mathcal{CS}_{free}$ en choisissant un nombre fini de configurations puis essaient de les relier par des méthodes locales.
Une telle démarche conduit à un sur-échantillonnage de $\mathcal{CS}_{free}$ et par conséquent à des temps de calcul très longs et un nombre excessif de reconfigurations de prise. En effet, ces méthodes imposent de changer de sous-espace même si on veut relier entre elles deux configurations d'une même composante connexe GS_k.
Nous proposons à la place une nouvelle approche permettant de tenir compte de la continuité des sous-espaces GS_k (donc de la continuité de la prise) et de relier directement deux configurations qui appartiennent à une même composante connexe de ces sous-espaces. Nous expliquons cette approche dans la section suivante.

4 UNE NOUVELLE APPROCHE DE PLANIFICATION POUR LA MANIPULATION FINE

Au lieu de travailler avec un ensemble discret de prises, nous utilisons la continuité des sous-espaces GS_k en introduisant des degrés de liberté virtuels pour représenter cette continuité. Ces degrés de liberté correspondent aux placements continus des points de contacts sur la surface de l'objet saisi. Cette représentation correspond à la description des prises de la partie 3.1. Elle permet de choisir des configurations qui respectent la fermeture des chaînes cinématiques formées par les doigts et l'objet et, également, d'envisager des déplacements dans une même composante connexe d'un sous-espace GS_k. On peut ainsi définir un chemin linéaire (figure 4) entre deux configurations de GS_k en reliant linéairement les deux vecteurs de configuration augmentés (par les variables modélisant les degrés de liberté ajoutés pour représenter la continuité de la prise).
L'exploration des composantes connexes des différents GS_k se fera directement en utilisant des chemins inclus dans ces sous-espaces. Ainsi, on pourra connecter directement dans GS_k deux configurations qui correspondent à la fois à des poses de l'objet et à des prises différentes,

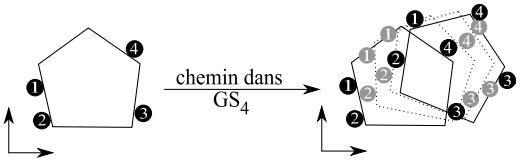

FIG. 4 – Un chemin linéaire dans GS_4 pour une main à 4 doigts dans le plan.

sans avoir à passer par des configurations intermédiaires (ce qui reviendrait à échantillonner inutilement l'espace de recherche). La figure 5 montre, dans le cas d'une main à quatre doigts, le nombre minimum de configurations intermédiaires nécessaires pour simplement passer d'une prise à quatre doigts à une autre si les quatre contacts changent de position. En restant dans GS_4, on peut relier directement les deux configurations. Il est important de noter que l'intersection de tous les GS_3 n'est autre que GS_4 (figure 5). Il suffit donc de déterminer la connectivité de GS_4 pour résoudre le problème de manipulation. Le passage d'une composante connexe de GS_4 à une autre se traduit par une opération de reconfiguration (ou saut d'un doigt).

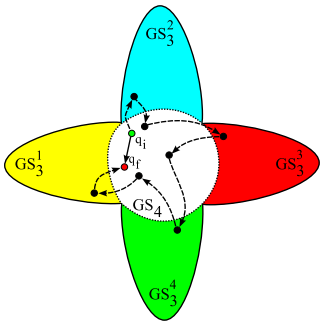

FIG. 5 – Une reconfiguration de prise par une suite de chemins de ressaisie (chemin en pointillé) passant par les différentes composantes connexes de $GS_3 \backslash GS_4$ et la même en restant dans GS_4 (chemin en trait plein).

Plus généralement, si on cherche à relier deux configurations de GS_k, il est inutile d'explorer la totalité de $\mathcal{CS}_{free}$; il suffit de déterminer la connexité de GS_k. Nous proposons donc de déterminer, dans un premier temps, les différentes composantes connexes de GS_k puis de les connecter par des chemins de transfert/ressaisie.
On construit ainsi un graphe de manipulation fine. La figure 6 montre un exemple de graphe de manipulation fine dans le cas d'une main à quatre doigts en présence d'obstacles (objet dans l'environnement ou butée articulaire).
Les chemins dans GS_k sont presque toujours cinématiquement irréalisables en raison de la modification de la prise par glissement des doigts, qui a lieu en même temps que les déplacements de l'objet. Cependant, ils peuvent être décomposés en séquence

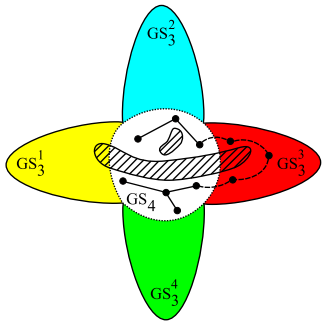

FIG. 6 – Un exemple de graphe de manipulation fine. Un passage par $GS_3 \backslash GS_4$ est nécessaire pour contourner un obstacle (zone hachurée) et relier les deux composantes de GS_4.

multiple de ressaisie-transfert ou transfert-ressaisie grâce à la propriété de réduction établie et prouvée dans [Alami, 1994] (pour un système à deux dimensions mais on peut l'étendre directement au cas de la manipulation fine). Les seuls chemins dans GS_k qui sont réalisables correspondent au cas particulier des chemins de transfert. La section suivante donne des détails sur l'implémentation de la méthode proposée et décrit notamment ses algorithmes fondamentaux.

5 PLANIFICATEUR DE TACHES DE MANIPULATION FINE

Nous proposons dans cette section un planificateur de tâches de manipulation conçu sur le modèle des méthodes probabilistes [Kavraki, 1994, Overmars, 1994]. Il travaille par conséquent en trois phases :

- une phase d'apprentissage
- une phase de recherche
- une phase de lissage

Le graphe de manipulation fine est construit lors de la phase d'apprentissage. Ce graphe dépend de la géométrie de la main robotisée, de l'objet et des obstacles statiques de l'environnement. Ainsi, dans un même environnement et pour un même objet à manipuler, le même graphe peut être utilisé pour résoudre plusieurs tâches de manipulation. Lors de la phase de recherche, les configurations initiales et finales sont connectées au graphe de manipulation. La recherche d'un chemin de manipulation dans le graphe est ensuite menée à l'aide d'un algorithme de type A*. Une fois le chemin de manipulation trouvé, une phase de lissage est lancée. Elle est composée de trois étapes ; la première étape consiste à transformer les chemins dans GS_k en un nombre fini de chemins de transfert et de ressaisie. Elle se fait à l'aide d'un algorithme dichotomique. Dans la seconde étape, on cherche à minimiser le nombre de reconfigurations de prise. Finalement, au cours de la troisième étape, les chemins de transfert/ressaisie composant le chemin de manipulation sont lissés. Un algorithme

probabiliste est utilisé pour effectuer ce lissage comme décrit dans [Kavraki, 1994].

5.1 Construction du graphe de manipulation

La construction du graphe de manipulation se fait en alternant deux étapes :
- exploration de GS_k
- connexion de ses composantes connexes par des chemins de transfert/ressaisie

Nous présentons l'algorithme de construction du graphe (voir Algorithme 1). q_{init} et q_{goal} sont respectivement la configuration initiale et la configuration finale désirée du système, $\mathcal{GM}$ le graphe de manipulation, $\mathcal{CC}$ l'ensemble des composantes connexes qui ont été trouvées par l'algorithme. l_1 et l_2 sont les premières composantes connexes trouvées par l'algorithme, à l'issue de l'ajout de q_{init} et q_{goal} à $\mathcal{GM}$.

$\mathcal{GM} = \{q_{init}, q_{goal}\}$ $\mathcal{CC} = \{l_1, l_2\}$
si $l_1 \equiv l_2$ **alors**
    fin de l'algorithme
sinon
    tant que $l_1 \neq l_2$ **faire**
        choisir aléatoirement $\alpha \in \{0; 1\}$
        suivant α **faire**
            cas où $\alpha = 0$
                explorer(q_{init},q_{goal},$\mathcal{GM}$,$\mathcal{CC}$)
            cas où $\alpha = 1$
                connecter composantes($\mathcal{GM}$,$\mathcal{CC}$)
        fin
    fin
fin

Algorithme 1 : Algorithme de construction du graphe de manipulation

5.1.1 *Exploration de GS_k*

L'exploration de GS_k doit permettre de déterminer ses différentes composantes connexes. GS_k est exploré en tant que tel sans utiliser de chemins qui sortent de ce sous-espace. Des degrés de liberté sont ajoutés entre chaque doigt participant à la prise et l'objet, modélisant à la fois la continuité de la prise et la fermeture des chaînes cinématiques. Explorer ainsi GS_k apparaît comme un problème de planification de mouvement d'un système comportant plusieurs chaînes cinématiques fermées. Pour résoudre ce problème, on utilise l'algorithme RLG [Cortés, 2002]. La chaîne est divisée en une partie active et une partie passive. La partie active est tirée aléatoirement dans le domaine d'accessibilité de la partie passive. Cette partie passive est calculée en utilisant les modèles géométriques inverses. La stabilité de la prise est vérifiée pour chaque configuration générée.

5.1.2 *Connexion dans GS_k*

Pour explorer GS_k, il est important de pouvoir connecter deux configurations de ce sous-espace. Dans les travaux de [Sahbani,2002], traitant de la manipulation à l'aide d'un bras robotisé, les auteurs ont eu à modéliser une variation continue de prise. Ils ont simplifié le problème en considérant le cas d'une pince à mors parallèles saisissant une barre parallélépipédique et en supposant que la prise ne se fait que sur une des paires de faces opposées de la barre. Dans le cas de la manipulation fine, le problème est bien plus complexe. Pour avoir une approche générale, il est important de disposer d'une paramétrisation de la prise qui permette des changements continus. Comme la prise est caractérisée par la position sur la surface de l'objet des points de contact, il est nécessaire d'avoir une paramétrisation de cette surface pour pouvoir calculer des chemins dans GS_k. Jusqu'ici nous avons supposé qu'il était toujours possible d'avoir une paramétrisation de la surface de l'objet manipulé. En vérité, la plupart des objets ont des surfaces non paramétrables et toutes les paramétrisations ne conviennent pas à notre problème. Les points de contact doivent idéalement se déplacer d'un point de la surface à un autre selon le plus court chemin. Si, par exemple, on fait varier linéairement les paramètres sphériques (coordonnées sphériques) de la position d'un point sur une sphère, la trajectoire du point sur la surface ne sera pas le plus court chemin.
Ce problème peut être contourné car ce qui est vraiment nécessaire est de pouvoir tirer au hasard des points n'importe où sur la surface de l'objet et de pouvoir calculer un chemin continu sur la surface de l'objet reliant deux de ces points. On peut y arriver en approximant la surface de l'objet par un polyèdre, ce qui n'est pas vraiment une limitation car on peut approximer n'importe quelle surface, avec la précision que l'on veut, par un polyèdre.
Un algorithme de calcul de géodésique est utilisé pour trouver le chemin le plus court possible entre deux points donnés de la surface, sous forme d'une suite de segments connectés les uns à la suite des autres, qui constitue une trajectoire continue. Le choix d'un point de contact est effectué par un tirage aléatoire d'une des faces du polyèdre avec un biais dépendant de son aire puis d'une position sur cette face.

5.1.3 *Connexion par des chemins de transfert/ressaisie*

Pour construire le graphe de manipulation fine, il nous reste à connecter les différentes composantes connexes de GS_k par des chemins de transfert/ressaisie. Pour relier deux configurations $q_1 = (\mathbf{p}_1, \mathbf{g}_{k_1})$ et $q_2 = (\mathbf{p}_2, \mathbf{g}_{k_2})$ appartenant à des composantes connexes de GS_k distinctes, *via* une séquence minimale de chemins de transfert et de ressaisie, il existe deux possibilités, nécessitant toutes deux l'ajout d'un noeud/configuration intermédiaire au graphe de manipulation :
- $q_1 \overset{ressaisie}{\longrightarrow} (\mathbf{p}_1, \mathbf{g}_{k_2}) \overset{transfert}{\longrightarrow} q_2$
- $q_1 \overset{transfert}{\longrightarrow} (\mathbf{p}_2, \mathbf{g}_{k_1}) \overset{ressaisie}{\longrightarrow} q_2$

Pour calculer les chemins de transfert et de ressaisie, nous utilisons la méthode de planification des *Rapidly-exploring Random Trees* (RRT, [Lavalle, 2000]).
Remarque :
La fréquence des tentatives de connexion des composantes connexes de GS_k par des chemins de trans-

fert/ressaisie dépend du paramètre α de l'algorithme 1. Le réglage de ce paramètre est donc très délicat. En effet, plus la probabilité d'avoir $\alpha = 1$ est grande, plus l'échantillonnage de $\mathcal{CS}_{free}$ est dense, ce qui ralentit considérablement l'algorithme et revient alors à effectuer une discrétisation de prises continues. Au contraire, plus la probabilité d'avoir $\alpha = 1$ est faible, plus on favorise l'exploration de GS_k en tant que domaine continu (chaînes fermées). Il s'agit là d'un compromis à trouver. Le choix de α dépendra aussi du problème de manipulation dextre à résoudre et de l'encombrement de l'environnement.

6 RESULTATS DE SIMULATION

Pour vérifier l'efficacité de la méthode proposée, nous avons développé (en C++) un planificateur , qui utilise la bibliothèque de fonctions de détection de collision PQP développée à l'UNC (*University of North Carolina*) [Gottschalk, 1996]. La main simulée comporte quatre doigts à trois degrés de liberté. Nous présentons les résultats obtenus pour trois exemples de tâches de manipulation dextre, sur un PC équipé d'un processeur Athlon64 3500+ et de 1Go de RAM. Le premier exemple est l'exemple classique de la réorientation d'une sphère (figure 7). Il est résolu en moins de vingt secondes.

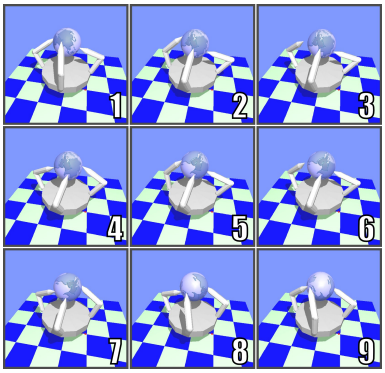

FIG. 7 – Etapes de la réorientation d'une sphère.

Le deuxième exemple présente la modification de la pose d'un objet parallélépipédique (figure 8). L'objet a une forme moins régulière que dans l'exemple précédent ; il y a donc plus de composantes connexes dans GS_4, ce qui rend le problème plus long à résoudre.

Le troisième exemple est la manipulation d'un ellipsoïde en présence d'un obstacle (figure 9), qui va nécessiter un lever de doigt supplémentaire.

Nous regroupons dans le tableau 1 différents résultats concernant la résolution de ces problèmes pour plusieurs exécutions. On peut remarquer comme les temps de calcul augmentent avec la difficulté du problème, liée au nombre de composantes connexes de GS_4, lui-même lié à la complexité de la forme de l'objet et au nombre d'obstacles présents.

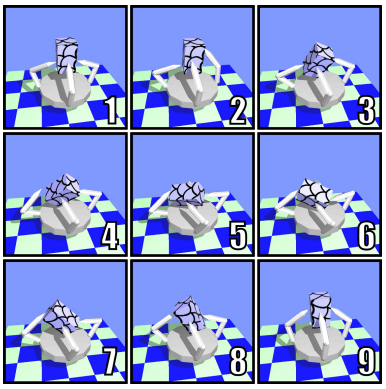

FIG. 8 – Etapes de la manipulation d'une boîte.

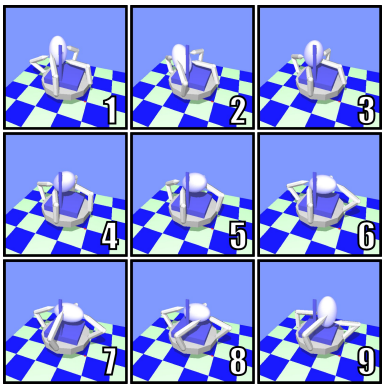

FIG. 9 – Etapes de la manipulation d'un ellipsoïde en présence d'un obstacle.

exemple	1	2	3
temps de résolution moyen (en secondes)	17.5	27	128
nombre moyen de noeuds générés	196	113	543
nombre moyen de noeuds du graphe	23	27	65

TAB. 1 – Performances du planificateur.

6.1 Comparaison avec les autres approches existantes

Parmi les techniques existantes, la comparaison avec la méthode proposée ne peut se faire qu'avec celles émettant des hypothèses similaires. Ainsi les méthodes considérant la dynamique complète du système requièrent logiquement des temps de calcul considérablement plus longs. La méthode de [Yashima, 2003], par exemple, trouve une solution au problème de la réorientation d'une sphère au bout de plusieurs heures. En revanche, elle ne calcule pas seulement les trajectoires des doigts et de l'objet mais aussi les consignes d'effort des doigts. Les auteurs des approches ne prenant pas en compte la dynamique ne présentent pas les temps de calcul de leurs résultats. Comme elles sont moins complètes puisqu'elles partent d'une trajectoire connue de l'objet [Goodwine2002, Harmati, 2002] ou ne traitent pas le problème de l'accessibilité des doigts [Cherif, 1999],

elles sont probablement plus rapides mais offrent moins de garantie de convergence.

7 CONCLUSION

Nous avons présenté une nouvelle approche de planification globale pour le problème de la manipulation dextre d'un objet rigide, reposant sur une nouvelle formulation du problème. Cette formulation s'appuie sur la définition des sous-espaces GS_k, qui sont les sous-espaces de l'ensemble des prises à k doigts. La méthode de résolution proposée construit un graphe probabiliste dont les noeuds sont des configurations choisies dans les GS_k. Ces sous-espaces sont explorés indépendamment puis les différentes composantes connexes obtenues sont reliés via des chemins appropriés. Des simulations de tâches de manipulation d'objets de différentes formes ont été réalisées qui ont confirmé la pertinence de la démarche.
Des améliorations peuvent être envisagées, concernant l'optimisation des chemins de manipulation calculés.

BIBLIOGRAPHIE

[Alami, 1994] Alami, R., Laumond, J.-P., Siméon, T. "Two manipulation planning algorithms", Algorithmic Foundations of Robotics WAFR94, Springer Verlag.

[Bicchi, 1995] Bicchi, A., "On the closure properties of robotic grasping", The International Journal of Robotics Research, Vol. 14, p. 319-33.

[Cherif, 1999] Cherif, M., Gupta, K., "Planning Quasi-Static Motions for Reconfiguring Objects with a Multi-fingered Robotic Hand", IEEE Transactions on Robotics and Automation, Vol. 4, p. 491-503.

[Cortés, 2002] Cortés, J., Siméon, T., Laumond, J.-P., "A Random Loop Generator for Planning the Motions of Closed Kinematic Chains using PRM Methods", Proceedings of the IEEE International Conference on Robotics and Automation.

[Goodwine2002] Goodwine,B., Burdick,J., "Motion Planning for Kinematic Stratified Systems with Application to Quasi-Static Legged Locomotion and Finger Gaiting", IEEE Transactions on Automatic Control, Vol. 18, p. 209-222

[Gottschalk, 1996] Gottschalk, S., Lin, M.C., Manocha,D., "OBBTree : A Hierarchical Structure for Rapid Interference Detection", Proceedings of ACM Siggraph'96, Nouvelle Orléans.

[Han, 1998] Han, L., Trinkle, J.C., "Dextrous Manipulation by Rolling and Finger Gaiting", Proceedings of the IEEE International Conference on Robotics and Automation, p. 730-735, Louvain.

[Han, 2000] Han, L., Li, Z., Trinkle, J.C., Qin, Z., Jiang, S., "The Planning and Control of Robot Dextrous Manipulation", Proceedings of the IEEE International Conference on Robotics and Automation, p. 263-269, San Francisco.

[Harmati, 2002] Harmati,I., Lantos,B., Payandeh,S., "On Fitted Stratified and Semi-Stratified Geometric Manipulation Planning with Fingertip Relocations", The International Journal of Robotics Research, Vol. 21, p. 489-510, juin 2002.

[Kavraki, 1994] Kavraki, L., Latombe, J.C., "Randomized Preprocessing of Configuration Space for Fast Path Planning", Proceedings of the IEEE International Conference on Robotics and Automation, p. 2138-2139, San Diego.

[Lavalle, 2000] LaValle, S.M., Kuffner, J.J., "Randomized kinodynamic planning", International Journal of Robotics Research, Vol. 20, p. 378-400.

[Li,1989] Li, Z., Canny, J., Sastry, S., "On Motion Planning for Dexterous Manipulation, Part I : The Problem Formulation", Proceedings of the IEEE Conference on Robotics and Automation, p. 775-780, Scottsdale.

[Liu, 1999] Liu, Y.-H., "Qualitative test and force optimization of 3D frictional form-closure grasps using linear programming", IEEE Transactions on Robotics and Automation, Vol. 15, p. 163-173.

[Montana,1995] Montana,D. "The Kinematics of Multi-Fingered Manipulation", IEEE Transactions on Robotics and Automation, Vol. 11, p. 491-503.

[Overmars, 1994] Overmars, M., Svestka, P., "A Probabilistic learning approach to motion planning", Algorithmic Foundations of Robotics, Springer-Verlag.

[Rus,1997] Rus, D., "In-hand Dexterous Manipulation of 3D piecewise-smooth objects" International Journal of Robotics Research.

[Sahbani,2002] Sahbani, A., Siméon, T., Cortés, J., "A probabilistic algorithm for manipulation planning under continuous grasps and placements", Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Lausanne.

[Siméon,2000] Siméon, T., Laumond, J.-P., Nissoux, C., "Visibility-based probabilistic roadmaps for motion planning", Journal of Advanced Robotics, Vol. 14, p. 477-494.

[Siméon,2003] Siméon, T., Cortés, J., Sahbani, A., Laumond, J.-P., "A General Manipulation Task Planner", Algorithmic Foundations of Robotics, Springer-Verlag, Vol. 5, p. 311-328.

[Sudsang,2003] Sudsang, A., Phoka, T., "Regrasp Planning for a 4-Fingered Hand Manipulating a Polygon", Proceedings of the IEEE International Conference on Robotics and Automation, Vol. 2, p. 2671-2676.

[Trinkle,1991] Trinkle, J.C., Hunter, J., "A Framework For Planning Dexterous Manipulation", Proceedings of the IEEE Conference on Robotics and Automation, p. 775-780, Sacramento.

[Yashima, 2003] Yashima, M., Shiina, Y., Yamaguchi, H., "Randomized Manipulation Planning for A Multi-Fingered Hand by Switching Contact Modes", Proceedings of the 2003 IEEE International Conference on Robotics and Automation, Taiwan.

[Zhang,1996] Zhang, H., Tanie, K., Maekawa, H., "Dextrous Manipulation Planning by Grasp Transformation", Proceedings of the IEEE Conference on Robotics and Automation, Minneapolis.