



**HAL**  
open science

# Planification de mouvements avec prise en compte explicite des incertitudes géométriques

Mario Fernando de La Rosa-Rosero

► **To cite this version:**

Mario Fernando de La Rosa-Rosero. Planification de mouvements avec prise en compte explicite des incertitudes géométriques. Autre [cs.OH]. Institut National Polytechnique de Grenoble - INPG, 1996. Français. NNT: . tel-00004986

**HAL Id: tel-00004986**

**<https://theses.hal.science/tel-00004986>**

Submitted on 23 Feb 2004

**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.

THESE

présentée par

Mario Fernando DE LA ROSA-ROSERO

pour obtenir le grade de DOCTEUR

de l'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

(Arrêté ministériel du 30 mars 1992)

(Spécialité : Informatique)

---

**PLANIFICATION DE MOUVEMENTS AVEC PRISE EN COMPTE  
EXPLICITE DES INCERTITUDES GEOMETRIQUES**

---

Date de soutenance : le 22 novembre 1996

Composition du jury :

Président : M. Roger MOHR

Rapporteurs : M. Rachid ALAMI  
M. Dominique MEIZEL

Directeur : M. Christian LAUGIER

Examineur : Mme. Jocelyne TROCCAZ

Thèse préparée au sein du projet SHARP  
GRAVIR - INRIA Rhône-Alpes  
655 Avenue de l'Europe  
38330 Montbonnot Saint Martin – France

*A mes parents,  
à ma grand-mère Maxima,  
à mes frères, et  
à tous ceux qui m'ont aidé*

## Remerciements

Je tiens à remercier Monsieur Roger MOHR Professeur à l'Institut National Polytechnique de Grenoble d'avoir présidé mon jury de thèse.

Je remercie Monsieur Rachid ALAMI Chargé de recherche CNRS au Laboratoire d'Analyse et d'Architecture des Systèmes (Toulouse) ainsi que Monsieur Dominique MEIZEL Professeur à l'Université de Technologie de Compiègne d'avoir accepté d'être les rapporteurs de ma thèse et apporté des remarques constructives concernant le mémoire.

Je suis reconnaissant à Madame Jocelyne TROCCAZI Chargé de recherche CNRS au Laboratoire TIMC (La Tronche) de l'intérêt qu'elle a porté à mon travail.

Je tiens à remercier profondément mon directeur de thèse Monsieur Christian LAUGIER Directeur de recherche INRIA (Unité de recherche INRIA Rhône-Alpes) m'avoir permis d'intégrer le projet de recherche SHARP et d'enrichir par les discussions que nous avons eu ma problématique de thèse.

J'adresse une pensée toute particulière aux membres du projet SHARP pour leur amitié leur soutien et les moments partagés dans la vie quotidienne et spécialement à José Philippe Moëz Juan Manuel Nicolas Alberto Remis Isabelle Thierry et Cyril.

Je remercie Philippe GARNIER Jérôme GENSEL Thierry FRAICHARD Isabelle MAZON et Nicolas MOULY pour leurs commentaires et corrections portant sur mon mémoire de thèse.

Je remercie également Christian BARD et Anton DEGUET qui ont assuré le bon fonctionnement du matériel informatique dans le projet SHARP et également Daniele HERZOG pour son aide dans le secrétariat.

Je remercie l'équipe RIA du LAAS-CNRS (Toulouse) pour m'avoir accueilli afin de réaliser des expériences réelles avec le robot mobile Hilare II. Je tiens à remercier plus particulièrement d'une part Messieurs Michel DEVY et Raja CHATILA pour m'avoir permis de réaliser cette partie expérimentale et d'autre part Messieurs Nicolas SIMEON Bertrand BOUILLY et Maher KHATIB pour m'avoir aidé dans la réalisation pratique des expériences avec le robot.

Je remercie le Centre Régional des Oeuvres Universitaires et Scolaires (CROUS) qui a assuré le financement de mes études doctorales en France.

Je remercie enfin "el grupo de amigos" avec qui j'ai partagé des moments très agréables en dehors de mes études. Je considère que ces moments m'ont aussi apporté

---

une expérience personnelle très enrichissante dans ma vie. Je tiens à remercier particulièrement : RodrigoΓMarioΓCathyΓHaroldΓLauraΓMaraΓPaoloΓSalvadorΓClaudiaΓMiguel SantanaΓConstanzaΓBrunoΓGloriaΓJorge MarioΓMaria CamilaΓMaria ConsueloΓFabioΓNathalieΓSylvieΓJean ClaudeΓAmeliaΓGuillermoΓRubbyΓPierreΓOlgaΓMaryΓCarmenΓXavierΓCarlosΓPascalΓVicenteΓJosé NájeraΓSaloméΓJosé EscamillaΓJérômeΓDenisΓVeroniqueΓJuan ManuelΓAlbertoΓArnieΓCecilieΓLaetitiaΓWimΓAlbertΓRicardoΓGermanΓLuis AlfonsoΓMaria HelenaΓMiguel NicolaΓPaulaΓCecileΓXimenaΓFabienneΓChristophe OrsyΓJean LucΓJennyΓMarthaΓJaimeΓPascaleΓJoëlΓChristophe CourbonΓMarleneΓ...

Fernando DE LA ROSA  
GrenobleΓDécembre 1996.

# Table des matières

---

<b>Introduction</b>	<b>1</b>
<b>1 Etat de l'art</b>	<b>5</b>
1.1 Le traitement explicite des incertitudes . . . . .	6
1.1.1 Approches ensemblistes . . . . .	7
1.1.2 Approches probabilistes . . . . .	14
1.2 Le traitement implicite des incertitudes . . . . .	17
1.3 La complexité algorithmique . . . . .	19
1.4 Conclusion . . . . .	20
<b>2 Les incertitudes géométriques dans la planification robotique</b>	<b>23</b>
2.1 Introduction . . . . .	23
2.2 Robot et obstacles dans l'espace des configurations . . . . .	24
2.3 Modélisation des incertitudes géométriques . . . . .	26
2.3.1 L'incertitude de position . . . . .	27
2.3.2 L'incertitude de contrôle . . . . .	28
2.3.3 L'incertitude d'orientation . . . . .	30
2.3.4 Représentation des incertitudes dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$ . . . . .	32
2.4 "Visibilité" et "atteignabilité" des surfaces . . . . .	34
2.4.1 "Visibilité" d'une surface plane dans $\mathbb{R}^2$ . . . . .	35
2.4.2 "Atteignabilité" d'une surface plane dans $\mathbb{R}^2$ . . . . .	36
2.4.3 "Atteignabilité" d'une surface dans $\mathbb{R}^2 \times [0, 2\pi]$ . . . . .	37
2.5 Conclusion . . . . .	41
<b>3 Formalisation du problème traité</b>	<b>43</b>
3.1 Définitions et notations de base . . . . .	43
3.2 Définition du problème de planification traité . . . . .	45
3.2.1 Génération de plans . . . . .	46
3.3 Le sous-espace $\mathcal{C}_{contact}$ de l'espace d'exploration . . . . .	48
3.3.1 Les contacts simples . . . . .	48
3.3.2 Les contacts multiples . . . . .	50
3.4 Capacités du système de contrôle du robot . . . . .	51

3.4.1	Spécification des commandes de contrôle . . . . .	52
3.5	Conclusion . . . . .	57
<b>4</b>	<b>Principes de base de notre approche de planification</b>	<b>59</b>
4.1	Composantes de l'approche de planification . . . . .	59
4.1.1	La méthode de potentiel fictif . . . . .	60
4.1.2	L'algorithme Fil d'Ariane . . . . .	62
4.2	Principe d'intégration dans l'approche de planification . . . . .	67
4.2.1	Notre contribution . . . . .	68
4.3	Un exemple de l'approche de planification . . . . .	68
4.4	Conclusion . . . . .	72
<b>5</b>	<b>Mise en œuvre de l'approche de planification</b>	<b>73</b>
5.1	L'espace d'exploration . . . . .	73
5.1.1	La construction des $\mathcal{C}$ -obstacles . . . . .	74
5.2	La fonction de potentiel fictif . . . . .	78
5.2.1	Le principe . . . . .	78
5.2.2	Comportement du potentiel fictif dans $\mathcal{C}_{libre}$ . . . . .	79
5.2.3	Passages de $\mathcal{C}_{libre}$ vers $\mathcal{C}_{contact}$ . . . . .	82
5.2.4	Comportement du potentiel fictif dans $\mathcal{C}_{contact}$ . . . . .	85
5.2.5	Algorithme de la fonction de potentiel fictif . . . . .	95
5.3	La fonction d'exploration . . . . .	96
5.3.1	Le principe . . . . .	96
5.3.2	Concept de balise . . . . .	96
5.3.3	Le sous-espace de contact adjacent et son exploration . . . . .	97
5.3.4	Le sous-espace de contact visible et son exploration . . . . .	102
5.4	L'approche de planification . . . . .	109
5.4.1	Les composantes de l'approche de planification . . . . .	109
5.4.2	Représentation de l'exploration de l'espace $\mathcal{C}_{valide}$ . . . . .	110
5.4.3	Modèle réparti de l'approche de planification . . . . .	111
5.4.4	Un algorithme séquentiel de l'approche . . . . .	112
5.4.5	Discussion sur l'approche proposée . . . . .	114
5.5	Transformation des chemins en plans . . . . .	114
5.6	Conclusion . . . . .	116
<b>6</b>	<b>Validations de l'approche</b>	<b>119</b>
6.1	Planification pour une pièce manipulée . . . . .	120
6.1.1	Problème 1. Attraction sans blocage (simulation) . . . . .	121
6.1.2	Problème 2. Insertion d'un goujon dans une concavité (simulation)	124
6.1.3	Problème 3. Passage étroit (simulation) . . . . .	133
6.1.4	Problème 4. Navigation dans un environnement encombré (simulation) . . . . .	135
6.1.5	Problème 5. Priorité sur l'exploration visible (simulation) . . . . .	139

---

6.1.6	Problème 6. Limitation de l'approche . . . . .	144
6.1.7	Conclusion . . . . .	145
6.2	Planification pour un robot mobile . . . . .	146
6.2.1	Modèle du robot . . . . .	146
6.2.2	Systèmes sensoriels du robot . . . . .	147
6.2.3	Mobilité du robot . . . . .	148
6.2.4	Primitives de mouvement du robot . . . . .	149
6.2.5	Contraintes supplémentaires imposées sur le robot . . . . .	150
6.2.6	Résultats de la planification des tâches . . . . .	154
6.2.7	Conclusion . . . . .	174
6.3	Conclusion . . . . .	174
<b>7</b>	<b>Conclusion et perspectives</b>	<b>175</b>
7.1	Conclusion . . . . .	175
7.2	Perspectives . . . . .	176
7.2.1	Améliorations . . . . .	176
7.2.2	Extensions . . . . .	176
<b>A</b>	<b>Algorithmes</b>	<b>179</b>
A.1	Algorithme de la fonction de potentiel fictif . . . . .	179
A.2	Un algorithme séquentiel de notre approche de planification . . . . .	183
A.3	Algorithme de construction d'un obstacle virtuel . . . . .	186
	<b>Table des figures</b>	<b>187</b>
	<b>Bibliographie</b>	<b>191</b>





# Dictionnaire

---

Nous définissons ci-dessous quelques concepts de base qui permettront au lecteur de mieux comprendre la partie introductive de ce mémoire. Des définitions formelles sont données dans §3.1.

**Degré de liberté (d.d.l.) d'un robot :** un d.d.l. d'un robot correspond à chacune des dimension dans l'espace du robot dans laquelle celui-ci est capable de réaliser un déplacement en translation ou en rotation. Un d.d.l. est toujours spécifié par une valeur numérique réelle par rapport à un référentiel: un d.d.l. en translation est spécifiée par une valeur  $x \in \mathbb{R}$  et un d.d.l. en rotation par une valeur  $\theta \in [0, 2\pi]$ .

**Configuration d'un robot :** une configuration d'un robot représente la position/orientation d'un repère fixe au robot par rapport au repère de son environnement. Une configuration est exprimée comme un n-uplet (point dans un espace  $N$ -dimensionnel) dont les coordonnées sont fonction des d.d.l. du robot. A partir d'une configuration on connaît la position/orientation du robot dans son environnement. Dans le cas d'un bras manipulateur une configuration est constituée des valeurs de chacune de ses articulations. Dans le cas d'un robot mobile circulaire évoluant sur un plan une configuration est constituée des coordonnées du point centre du robot dans le plan.

**Tâche robotique :** une tâche robotique dans le cadre de ce rapport consiste à spécifier: (1) une configuration initiale et une configuration but pour le robot; (2) la géométrie et les contraintes liées au robot (e.g. contraintes sur la cinématique la dynamique la précision des déplacements exécutés l'interaction avec l'environnement etc.) et (3) la géométrie la configuration et les contraintes liées aux obstacles présents dans l'environnement où le robot se déplace (e.g. type d'obstacle statique ou dynamique contraintes sur leur rigidité physique etc.).

**Chemin pour un robot :** un chemin est défini comme une séquence continue de configurations pour un robot depuis une configuration initiale jusqu'à une configuration finale.

**Plan pour un robot :** un plan est défini comme un ensemble discret d'actions (commandes) à exécuter avec un robot depuis une configuration initiale jusqu'à une configuration finale.

**Problème élémentaire de planification de mouvement en robotique :** étant donnée une tâche robotique pour un robot la planification de mouvement consiste à générer un chemin ou un plan pour le robot qui l'amène de la configuration initiale vers la configuration finale en respectant les contraintes définies par la tâche.

**Robot "volant" :** un robot volant est un objet qui n'est pas soumis à des contraintes cinématiques sur ses déplacements c'est-à-dire que le robot peut se déplacer librement sur chacun de ses d.d.l..

# Introduction

---

## Problématique abordée

Dans le domaine de la robotique la planification de mouvement est un sujet très étudié par les roboticiens et ceci peut se constater au travers du vaste ensemble de développements réalisés à ce jour. Les premiers travaux ont traité le problème de la planification de mouvements de transfert<sup>1</sup> dont les tolérances<sup>2</sup> acceptées sont considérées comme élevées par rapport aux imprécisions propres du robot et de l'environnement. Sous cette hypothèse il n'était donc pas nécessaire de prendre en compte les imprécisions. Ces imprécisions appelées en robotique les *incertitudes* correspondent aux variations entre la représentation de l'espace où les actions robotiques sont planifiées et l'espace réel où elles sont exécutées. Ensuite la planification de mouvement en robotique a continué son évolution vers de nouvelles tâches (par exemple de saisie de manipulation d'assemblage d'approche etc.) dont les contraintes sont plus complexes. Parmi ces tâches on distingue un sous-ensemble qui est sensible aux effets des incertitudes : une tâche dans ce sous-ensemble ne pourra pas être accomplie malgré une solution nominale de planification satisfaisante. Les besoins de résolution de ces tâches ont donc demandé la recherche de techniques de planification devant prendre en compte les incertitudes géométriques. En particulier un sujet de recherche d'intérêt est celui de la planification de mouvement dont les incertitudes sur le positionnement (position/orientation) et sur le déplacement (contrôle) du robot pourraient faire échouer l'exécution d'actions nominales correctes. Ces incertitudes sont appelées *incertitudes géométriques* car elles portent sur des données géométriques (position/orientation/vecteur de déplacement). Dans ce contexte le nouveau problème de planification qui se pose consiste à définir des solutions "robustes" pour le robot vis-à-vis des incertitudes géométriques. Ces solutions garantissent au robot d'atteindre son but malgré les effets des incertitudes géométriques.

---

1. Un mouvement de transfert permet à un robot d'effectuer de longs déplacements dans des régions peu encombrées de son environnement.

2. Une tolérance est l'écart acceptable sur certaines grandeurs (dimensions, angles, vitesses, etc.) dû aux défauts de fabrication mécanique.

## Principales contributions de notre travail de recherche

Les travaux présentés dans ce mémoire s'inscrivent dans le cadre de la *planification de mouvement pour un robot sous des contraintes d'incertitudes géométriques* et ont abouti à la définition et au développement d'une nouvelle approche de planification automatique. Les contributions principales de notre approche portent sur :

- la modélisation des incertitudes géométriques ainsi que sur la définition de leur évolution selon les situations rencontrées en planification;
- la définition de stratégies de réduction des incertitudes géométriques adaptées aux situations rencontrées en planification;
- l'application de stratégies de planification adaptées pour évaluer incrémentalement les incertitudes géométriques;
- la prise en compte à la fois des translations et des rotations dans les stratégies de planification;
- l'intégration des stratégies précédentes dans une approche de planification de mouvement.

La validation de l'approche a été faite à deux niveaux. Tout d'abord nous avons eu recours à des simulations faites par ordinateur ce qui nous a permis de considérer un ensemble plus vaste de tâches de planification. Par la suite nous avons fait des expérimentations sur la plate-forme du robot mobile Hilare II du LAAS-CNRS pour un ensemble contraint de tâches de planification. Les résultats obtenus nous encouragent à penser que des adaptations de l'approche pourraient apporter de bonnes solutions à un ensemble de problèmes réels de planification de mouvement où les stratégies mises en œuvre actuellement ne sont guère adéquates.

## Organisation du mémoire

Ce mémoire se compose de six chapitres.

Le chapitre 1 présente les principaux travaux concernant la gestion des incertitudes géométriques dans la planification de mouvements de robots. Nous mettons l'accent sur les stratégies qui considèrent une modélisation explicite des incertitudes afin d'avoir un point de référence proche de notre approche.

Dans nos travaux nous nous intéressons aux incertitudes géométriques qui affectent directement le robot et dont le comportement est prévisible et continu. Dans le chapitre 2 les incertitudes de base seront présentées et des modèles explicites pour déterminer leurs évolutions seront proposés. Des modèles géométriques plus complets seront

construits en intégrant les modèles de base pour combiner leurs différents effets.

Le chapitre 3 pose d'une manière formelle le problème de planification de mouvement traité et aborde la faisabilité des solutions possibles.

Le chapitre 4 présente les principes de base sur lesquels notre approche de planification s'appuie pour mieux faire comprendre sa philosophie et son application. L'approche est née des besoins identifiés dans des travaux réalisés en planification de mouvements fins (robotique d'assemblage) où il est nécessaire de prendre en compte les incertitudes. Ces besoins ont guidé différents travaux qui ont débouché sur l'approche de planification proposée dans ce mémoire.

Le chapitre 5 précise l'espace d'exploration ainsi que les composantes faisant partie de l'approche de résolution. L'analyse de l'incertitude géométrique est prise en compte dans l'approche et est effectuée à l'aide des modèles géométriques proposés. L'intégration des différentes composantes est explicitée dans un algorithme.

Le chapitre 6 illustre l'application de l'approche proposée afin de résoudre le problème de planification de mouvement présent dans deux types de tâches robotiques : d'une part des tâches pour une pièce polygonale manipulée sur un plan et d'autre part des tâches pour un robot mobile. Les résultats montrés ont été obtenus à partir de simulations faites par ordinateur mais également à partir d'expériences réelles avec le robot mobile Hilare II du LAAS-CNRS.



# Chapitre 1

## Etat de l'art

---

Depuis plusieurs années des travaux de recherche sont menés dans le domaine de la planification de mouvement en robotique. Les approches de résolution issues de ces travaux peuvent être regroupées en deux grandes catégories :

- les *approches globales* qui cherchent à trouver une solution dans une représentation complète de l'espace de résolution du problème posé. L'avantage de disposer d'une telle représentation est que s'il existe une solution au problème celle-ci sera trouvée. De ce fait ces méthodes sont dites *complètes*. Cependant l'inconvénient de ces méthodes réside dans la complexité algorithmique pour construire et exploiter les représentations proposées. En général ces approches restent efficaces pour des problèmes dont l'espace de résolution a un nombre réduit (2 ou 3) de dimensions.
- les *approches locales* qui ne considèrent qu'une représentation partielle et incrémentale de l'espace de résolution. Les critères guidant la recherche des solutions sont aussi de nature locale. On obtient des approches efficaces et applicables même pour des problèmes dont l'espace de résolution a un grand nombre de dimensions. Cependant la plupart de ces approches accepte des minima locaux. Pour pallier ce problème des stratégies de traitement de minima locaux sont nécessaires. Malheureusement dans certains cas certaines approches ne garantissent pas la génération d'une solution malgré son existence. De telles approches sont dites *incomplètes*.

Afin de situer notre travail parmi ces travaux nous nous sommes concentré sur les approches qui prennent en compte les contraintes d'incertitude. Nous présentons donc un ensemble significatif d'approches dans lequel les incertitudes sont considérées de plusieurs façons. Le lecteur désireux d'acquérir une vision plus générale sur la problématique de planification de mouvement en robotique peut se rapporter à l'ouvrage de Latombe [Lat91].



Dans les travaux de recherche développés auparavant sur la planification sous des contraintes d'incertitude nous distinguons deux grands sous-ensembles. Un premier sous-ensemble considère les incertitudes de manière explicite. Dans ce cas l'évaluation des incertitudes est estimée par différentes représentations et son résultat permet de prévoir le succès ou l'échec d'un déplacement. Ainsi la validité des déplacements nominaux du robot est conditionnée par la validité dans l'espace réel des représentations d'incertitude choisies. Un deuxième sous-ensemble de travaux considère les incertitudes de manière implicite. Dans ce cas les actions sur le robot permettent elles-mêmes de maintenir des incertitudes faibles ou nulles (par exemple en définissant des contacts).

Dans la suite nous introduisons les principaux travaux qui prennent en compte les incertitudes aussi bien de façon explicite que de façon implicite. Ensuite nous présentons les principaux résultats concernant l'analyse de la complexité algorithmique liée au problème de la planification sous des contraintes d'incertitudes géométriques. Une synthèse des travaux ainsi qu'une bibliographie de référence dans le domaine de la planification sous des contraintes d'incertitudes géométriques sont présentées par Goldberg *et al.* [GMR92].

Dans la plupart des approches qui suivent l'espace de résolution est *l'espace des configurations* introduit par Lozano-Pérez [Loz83]. De manière simple une *configuration* du robot est la représentation de la position/orientation d'un repère fixe au robot par rapport au repère de son environnement. Une configuration est exprimée comme un  $n$ -uplet (point dans un espace  $N$ -dimensionnel) dont les coordonnées sont fonction des degrés de liberté (d.d.l.) du robot. A partir d'une configuration on connaît donc la position/orientation du robot dans son environnement. L'ensemble des configurations d'un robot définit un espace  $N$ -dimensionnel appelé *l'espace des configurations*. Dans cet espace la représentation du robot et des obstacles est différente de celle qui leur sont associées dans l'espace physique où ils existent. D'une part le robot est réduit à un point (*configuration*) et d'autre part chaque obstacle dans l'environnement suit un grossissement avec la géométrie réelle du robot selon chacun de ses degrés de liberté. Chaque obstacle  $N$ -dimensionnel ainsi obtenu est appelé un *C-obstacle*. Dans l'espace des configurations le problème classique de planification de mouvement est re-formulé comme le problème pour un robot ponctuel entouré de C-obstacles. Ces concepts seront définis d'une manière plus précise dans le chapitre 2.

## 1.1 Le traitement explicite des incertitudes

Dans l'ensemble de travaux qui considèrent explicitement les incertitudes on distingue deux points de vue qui découlent de la nature de leur représentation : *l'ensembliste* et le *probabiliste*. Du point de vue ensembliste l'incertitude est représentée par l'ensemble des valeurs possibles que l'on associe à une mesure estimée. Dans ce cas on peut considérer toutes les valeurs dans un ensemble comme équiprobables ce qui veut

dire que l'analyse de l'incertitude prend toujours en compte le pire cas. Du point de vue probabiliste une mesure estimée est garantie avec une probabilité qui caractérise la confiance que l'on l'apporte ce qui laisse une probabilité d'admettre des valeurs inattendues qui pourraient par la suite mettre en échec une solution trouvée. Nous allons classer les différentes approches selon la nature de la représentation mise en œuvre.

### 1.1.1 Approches ensemblistes

Les premiers travaux dans lesquels une analyse des effets de l'incertitude est considérée dans la planification ont été présentés par Taylor [Tay76]. Il propose un mécanisme de propagation d'incertitudes de position dans un plan d'actions. La position relative entre deux objets est représentée par une transformation géométrique générique dépendant de paramètres qui représentent les degrés de liberté non parfaitement connus. Cependant on considère que chaque paramètre peut prendre une valeur dans un intervalle connu. Des contraintes sur les valeurs des paramètres peuvent être définies par des équations ou inéquations. Le mécanisme de traitement d'incertitudes permet de propager des valeurs numériques dans les contraintes où les paramètres apparaissent. Une amélioration de cette approche a été proposée par Brooks [Bro82]. Celle-ci consiste à appliquer un mécanisme de propagation de contraintes symboliques d'incertitude. Cette propagation est faite vers l'avant pour calculer les limites sur les erreurs et vers l'arrière pour restreindre les valeurs initiales de certains paramètres. Ce mécanisme de propagation a été conçu pour être intégré dans le système de programmation automatique en robotique TWAIN. Cependant ce système n'a pas réellement été implanté.

La première approche formelle de synthèse automatique de mouvements compliants (i.e. mouvements en contact) pour la résolution d'un sous-ensemble de tâches robotiques est présentée par Lozano-Pérez *et al.* [LMT84]. L'approche considère un robot ponctuel (pièce manipulée) dans l'espace des configurations  $\mathbb{R}^2$  composé de C-obstacles polygonaux. Un ensemble de modèles géométriques est utilisé dans cette approche :

1. l'incertitude liée à une position nominale est représentée par un disque autour de celle-ci;
2. l'incertitude de contrôle liée à un déplacement de translation (spécifié par un vecteur vitesse) est représentée par un cône placé sur une position initiale dont l'axe suit la direction du déplacement et l'angle correspond à l'imprécision lors de son exécution;
3. les forces de réaction exercées sur un point de contact suivent la loi de Coulomb et sont contenues dans un cône dont l'axe suit la normale extérieure à la surface de contact et l'angle est fonction du coefficient statique de frottement propre à la surface de contact; et
4. les déplacements compliants du robot ponctuel sont assimilables au comportement d'un amortisseur (modèle "generalized damper").

L'approche consiste à appliquer une méthode de chaînage arrière depuis une région but  $R_f$  vers une région de départ  $R_0$  du robot. A chaque pas  $i$  du chaînage on suppose connu l'ensemble de positions (région)  $B_i$  où le robot se trouve réellement à cause des incertitudes. On détermine donc l'ensemble de positions de départ (région)  $P_i$  et l'ensemble de vitesses  $V_i$  garantissant que le robot ponctuel atteigne  $B_i$  malgré d'une part les incertitudes de position et de contrôle et d'autre part le frottement de contact. La région obtenue  $P_i$  est appelée la *pré-image* de  $B_i$  selon les vitesses  $V_i$  (figure 1.1).

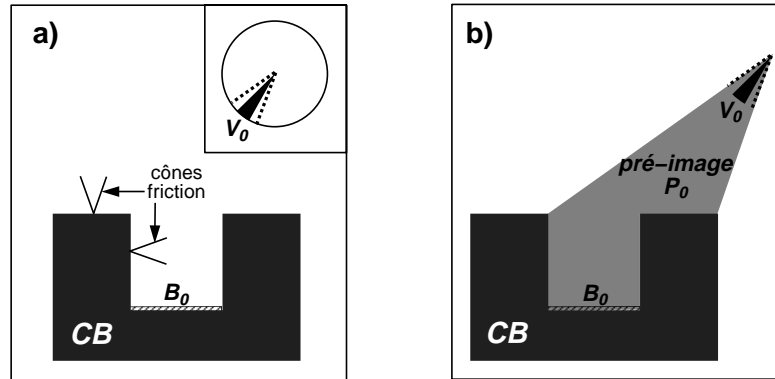


FIG. 1.1 – a) Problème d'insertion dont le but est d'atteindre la région but  $B_0$  malgré l'incertitude de contrôle et le frottement de contact. Un ensemble de vitesses  $V_0$  satisfaisant les contraintes est montré en haut à droite. b) Définition de la pré-image  $P_0$  correspondant à la région  $B_0$  selon les vitesses  $V_0$ . [Lat91]

De plus une condition de terminaison  $C_i$  est nécessaire pour arrêter l'application des vitesses dans  $V_i$  depuis  $P_i$  lorsque le robot atteint  $B_i$ . Le chaînage arrière est lancé avec  $i$  égal à zéro et  $B_i$  égal à  $R_f$ . Le processus de chaînage arrière aboutit soit à un succès soit à un échec. Le succès a lieu lorsqu'une *pré-image*  $P_i$  contient la région  $R_0$  (figure 1.2). Dans ce cas la stratégie solution est constituée par la séquence inverse de couples  $\{(V_i, C_i)\}$  trouvée depuis  $B_0$  jusqu'à la *pré-image*  $P_i$ . Pendant l'exécution d'une stratégie la satisfaction des conditions  $C_i$  détermine le changement de la vitesse à suivre. D'autre part un échec est détecté lorsque les ensembles potentiels de vitesses  $V_i$  deviennent tous nuls sans avoir atteint la région  $R_0$ . Bien que l'approche possède un principe de résolution formelle elle n'a cependant jamais été implantée pour des raisons de complexité algorithmique (§1.3). De plus l'algorithme de base pour la construction de *pré-images* n'est pas explicité. Cette approche a été par la suite à l'origine de plusieurs méthodes de résolution de cas spécifiques.

En raison de la complexité algorithmique de l'approche précédente Erdmann [Erd86] a défini une approche basée sur un modèle géométrique simple de pré-image appelé "*backprojection*" (projection en arrière) noté  $BP_i$ . Ce modèle garantit aussi l'"atteignabilité" des régions  $B_i$  recherchées. Par contre les conditions de terminaison  $C_i$  permettant la

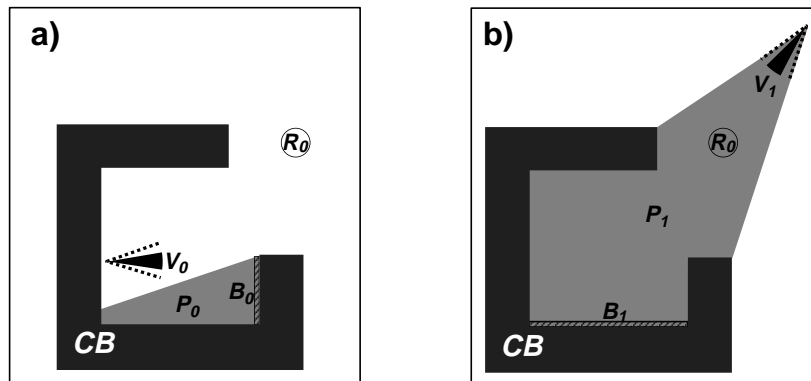


FIG. 1.2 – *Chânage arrière de pré-images depuis la région but  $R_f = B_0$  vers la région de départ  $R_0$ .  $R_0$  est atteinte après la définition de deux pré-images :  $P_0$  selon les vitesses  $V_0$  (a) et  $P_1$  selon les vitesses  $V_1$  (b). [Lat91]*

“reconnaissabilité” des régions  $B_i$  ne tiennent compte que des estimations sur les vitesses et les positions du robot. L’aspect temporel d’exécution d’une vitesse n’est pas considéré comme c’est le cas dans l’approche [LMT84]. Cela permet le calcul d’un ensemble de pré-images simplifiées. En ce qui concerne l’approche proprement dite celle-ci suppose un espace des configurations  $\mathbb{R}^2$  composé de C-obstacles polygonaux et son principe de résolution est de type chânage arrière. Bien que la complexité algorithmique de l’approche ne soit pas démontrée formellement un algorithme explicite de calcul des “backprojections” est donné. A notre connaissance deux méthodes basées sur cette approche ont été développées et validées en simulation par Latombe *et al.* [LLS91]. Dans les deux méthodes on suppose un robot ponctuel qui se déplace en translation dans un espace des configurations  $\mathbb{R}^2$  où des C-obstacles statiques et polygonaux sont définis. La première méthode proposée originellement par Donald [Don88b] construit des “backprojections” pour des segments d’arêtes de contact où le déplacement du robot peut être bloqué. La deuxième méthode construit des “backprojections” pour des régions sous-but. Un inconvénient de ces approches est la complexité de la définition des conditions de terminaison  $C_i$  des vitesses  $V_i$  suivies par le robot. Ces conditions ont un effet important sur la construction des “backprojections”. L’analyse théorique sur la construction des conditions  $C_i$  est présentée dans [SL91].

Egalement Erdmann a proposé dans [Erd86] le modèle “forward projection” (projection en avant) noté  $FP_i$  qui est issu du modèle “backprojection”. La “forward projection”  $FP_i(\vec{v})$  d’un ensemble de positions  $B_i$  selon la vitesse  $\vec{v}$  correspond à l’ensemble des positions que le robot peut atteindre depuis  $B_i$  suivant  $\vec{v}$  en tenant compte du frottement de contact et de l’incertitude de contrôle sur  $\vec{v}$ . Ce nouveau modèle est la base des approches de planification sous des contraintes d’incertitudes qui raisonnent vers l’avant.

Donald [Don86] est le premier à considérer les incertitudes d’orientation sur l’en-

vironnement. Les incertitudes de position et de contrôle sur le robot sont aussi considérées. Le cas général est présenté pour un robot à  $n$  d.d.l. en translation et  $k$  d.d.l. sur l'orientation de l'environnement. La stratégie de résolution proposée est basée sur la construction de “*backprojections*” [Erd86] dans l'espace généralisé des configurations  $\mathbb{R}^n \times S^k$ . Cependant aucune implantation opérationnelle n'a été réalisée. Ultérieurement Donald [Don88a] a analysé la complexité algorithmique des problèmes de planification des mouvements pour un robot ponctuel sous des contraintes d'incertitude de position et de contrôle. L'orientation n'est pas considérée. La complexité est définie pour des problèmes posés dans l'espace des configurations  $\mathbb{R}^2$  composé de C-obstacles polygonaux. Les résultats présentés dans §1.3 dérivent directement de l'analyse faite par Donald [Don86].

Une approche intégrant une analyse sur l'incertitude d'orientation est proposée par Brost [Bro91]. Celle-ci traite le problème de planification pour un sous-ensemble de tâches de manipulation sous des contraintes d'incertitudes. Les tâches d'intérêt sont définies par une position/orientation but où deux objets polygonaux sont en contact stable et par une direction de mouvement à suivre par un des objets appelé “l'objet manipulé”. Le problème à résoudre consiste à trouver un ensemble initial de positions/orientations pour l'objet manipulé garantissant que le suivi du mouvement demandé l'amènera à la situation but malgré : (1) les incertitudes en position/orientation sur les objets (2) l'incertitude sur la direction du mouvement et (3) les frottements de contact (l'incertitude sur la forme géométrique des objets n'étant alors pas prise en compte). L'approche est appliquée dans l'espace des configurations  $\mathbb{R}^2 \times S^1$ . Dans cet espace une configuration (position/orientation) de l'objet manipulé devient un point et l'objet de contact devient un C-obstacle tridimensionnel. L'approche construit un ensemble de configurations (libres et/ou de contact) de départ pour l'objet manipulé qui garantissent d'atteindre la configuration but en suivant la direction de mouvement demandé. Cette première construction tient compte des aspects géométrique cinématique et statique de la tâche et aussi de l'incertitude sur la direction du mouvement. Ensuite cet ensemble est transformé dans le système de référence du robot qui exécutera la tâche. Ce nouvel ensemble est réduit en considérant les incertitudes sur la position des objets. L'ensemble ainsi obtenu est la solution du problème posé. Des résultats expérimentaux variés ont pu être obtenus grâce à la mise au point d'une implantation sur ordinateur. Ces résultats portent sur trois sous-ensembles d'expériences : (1) la planification de mouvements linéaires de poussé (“pushing”) (2) la planification de mouvements d'assemblage et (3) le placement d'objets soumis à la pesanteur. Cependant l'application de cette approche suppose d'une part un environnement composé seulement de deux objets polygonaux placés sur un plan et d'autre part l'analyse de l'application d'un seul déplacement sur l'objet manipulé.

Une approche originale de planification pour un robot ponctuel dans l'espace libre<sup>1</sup>

---

1. Espace libre : sous-espace où le robot n'est ni en contact ni en collision avec les obstacles.

sous des contraintes d’incertitude de position et de contrôle est proposée par Lazanas et Latombe [LL92]. Le robot évolue dans un plan en se déplaçant en translation. L’environnement est composé d’obstacles à éviter et de balises (“landmarks”) à l’intérieur desquelles le robot peut réduire complètement son incertitude. Les obstacles et les balises et la région de départ du robot sont décrits comme une composition de disques. D’autre part le but à atteindre est une région de l’espace libre qui intersecte au moins une des balises dans l’environnement. La stratégie proposée est basée sur l’approche de Lozano-Pérez *et al.* [LMT84] et engendre des chemins composés de vitesses (directions) qui guident le robot dans l’espace libre pouvant traverser des balises (figure 1.3).

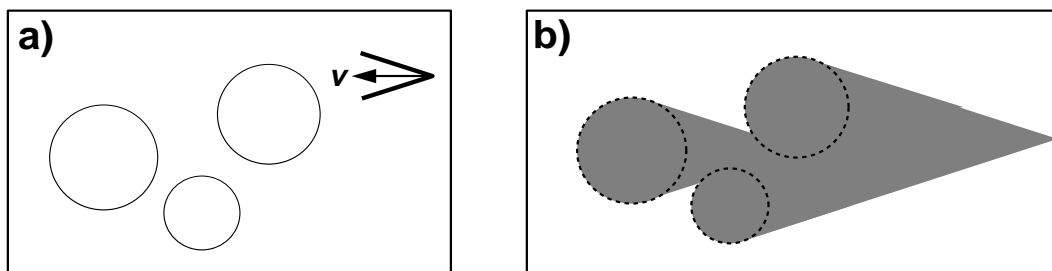


FIG. 1.3 – a) Région but (composée de disques) à atteindre en tenant compte de l’incertitude sur la vitesse  $\vec{v}$ . b) Construction de la pré-image associée à la vitesse  $\vec{v}$  et à son incertitude.[LL92]

Alami et Siméon [AS94] [SA94] proposent une approche de planification d’actions pour un robot ponctuel soumis à des contraintes d’incertitude de position et de contrôle dans un espace parsemé d’obstacles polygonaux. Un plan d’actions est composé de primitives de contrôle basées sur des informations perceptives. Ces primitives tiennent compte de l’accumulation des erreurs sur l’estimation de la position du robot le long du déplacement. L’approche exploite la propriété de réduction de l’incertitude lorsque le robot est en contact avec une arête d’un obstacle. Le robot peut chercher un contact depuis une position libre ou une position en contact avec un autre obstacle. Un modèle géométrique de visibilité de type “*forward projection*” est défini pour déterminer l’ensemble des directions (translations) permettant d’atteindre malgré les incertitudes une arête de contact depuis une position de départ libre ou en contact (figure 1.4).

L’approche construit un graphe d’exploration dont le nœud initial est défini avec la position et l’incertitude initiales du robot. Un nœud dans le graphe correspond à une position atteinte (avec une incertitude estimée) par le robot sur une arête de contact. Deux nœuds adjacents dans le graphe sont liés par un arc étiqueté avec la distance euclidienne et la primitive de contrôle entre leurs positions. Les arcs sont orientés depuis la position de départ vers la position d’arrivée. L’exploration est guidée par l’algorithme  $A^*$  appliqué sur le graphe. Le critère de coût entre deux nœuds adjacents correspond à leur distance euclidienne. A chaque pas de l’exploration on explore le “meilleur” nœud dans le graphe en définissant des nœuds adjacents pour des positions discrètes sur des arêtes atteignables depuis sa position. Depuis la position de chaque nouveau

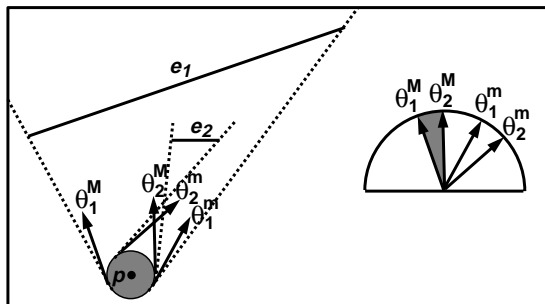


FIG. 1.4 – Construction du modèle de visibilité de l’arête  $e_1$  depuis la position  $p$  incertaine. Le secteur circulaire du cercle unitaire défini entre les angles  $\theta_2^M$  et  $\theta_1^M$  contient l’ensemble des directions “sûres” pour atteindre  $e_1$  malgré les incertitudes de position et de contrôle. [AS94]

nœud (y compris le nœud initial) l’“atteignabilité” de la position but est évaluée. Si celle-ci est atteinte avec une estimation d’incertitude en dessous de la valeur maximale acceptée un nœud but est défini avec la position but et son incertitude estimée. Un plan solution est composé de l’ensemble des primitives de contrôle de la branche du graphe reliant le nœud initial au nœud but. Les résultats obtenus lors des simulations faites par ordinateur ont validé cette approche.

Bouilly *et al.* [BAS95] considèrent une problématique similaire à celle traitée par [AS94] mais en supposant un robot circulaire. L’approche proposée engendre des plans d’actions composés de primitives de contrôle pré-définies. L’approche est basée sur un potentiel numérique défini sur une grille de l’espace des configurations  $\mathbb{R}^2$ . L’estimation de l’incertitude d’une case de la grille est propagée par le potentiel numérique aux cases voisines. L’algorithme maintient une liste des cases à propager ordonnées selon la distance parcourue depuis la configuration initiale. Cette liste est initialisée avec la case correspondant à la configuration initiale. A chaque pas d’itération du potentiel numérique la case de la liste ayant la valeur de distance la plus petite est propagée en estimant l’incertitude des cases voisines. Parmi celles-ci on ajoute à la liste de propagation celles qui ont vu réduire leur estimation d’incertitude. D’autre part le potentiel numérique accepte des contacts avec les arêtes des C-obstacles sous la *condition de visibilité*. Cette condition garantit un contact malgré les incertitudes de position et de contrôle. Un contact permet une réduction de l’incertitude de position. Deux stratégies de re-localisation du robot sont proposées : la première dans l’espace de contact au moment de la détection d’un sommet d’un obstacle et la deuxième dans l’espace libre à l’intérieur des régions appelées “landmarks” (figure 1.5). Une re-localisation permet de réduire l’incertitude liée à la case dans la grille correspondant à la position du robot. Le potentiel numérique trouve un plan valide lorsque la case correspondant à la configuration but est atteinte sous une estimation d’incertitude acceptable. Un échec se produit lorsque la liste de propagation est vide. Cette approche présente l’avantage de ne considérer les contacts que lorsque l’évolution de l’incertitude dans l’espace libre

ne permet pas au robot d'atteindre son but avec la précision souhaitée. En revanche l'en raison de la géométrie circulaire du robot ni l'orientation du robot ni son incertitude associée ne sont prises en compte par l'approche. Des résultats expérimentaux ont été obtenus avec un robot mobile.

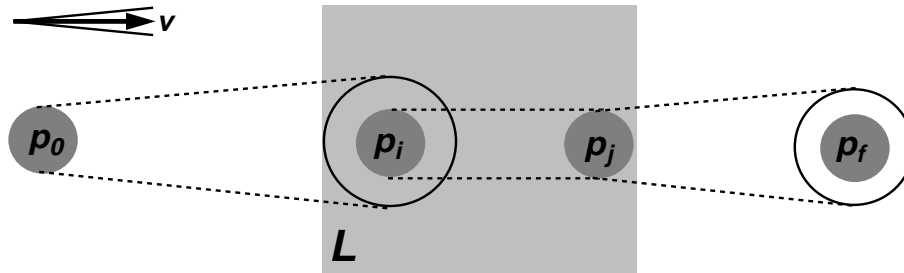


FIG. 1.5 – Re-localisation à l'intérieur d'un "landmark"  $L$  lorsque le robot circulaire se déplace suivant la vitesse  $\vec{v}$  de la position  $p_0$  à la position  $p_f$ . Les cercles autour du robot délimitent la frontière estimée où le robot peut se trouver en raison de l'incertitude de contrôle. [BAS95]

Les données capteurs fournissent des informations sur l'environnement qui permettent de réduire les incertitudes du robot mais ces données sont aussi entachées d'incertitude. Page et Sanderson [PS95] ont proposé une approche de planification tenant compte de l'incertitude des données capteurs et de celle de contrôle. Les capteurs initialement considérés sont de type caméra et proximétrique. Un capteur disponible par le robot est représenté par: (1) l'ensemble des configurations où des acquisitions pourraient être réalisées (2) un champ d'incertitude décrivant la précision absolue et (3) une incertitude incrémentale en déplacement décrivant la précision relative. L'incertitude des données capteurs représente les imprécisions entre la position réelle du robot et sa position estimée. L'incertitude de contrôle représente les imprécisions entre la position estimée et sa position commandée. L'incertitude de position du robot est le résultat de la somme de l'incertitude des données capteurs et de celle de contrôle. D'autre part une intégration des deux approches de planification complémentaires est proposée pour planifier des chemins pour un robot circulaire dans le plan. Chaque approche est définie dans l'espace des configurations  $\mathbb{R}^2$  et intègre les représentations des incertitudes traitées. Un chemin nominal généré est accompagné par un ensemble d'intervalles dans lesquels des acquisitions capteurs sont nécessaires. Dans cette approche les contacts ne sont pas considérés.

En résumé la plupart des approches de type ensembliste présente deux points communs: le premier le robot est considéré comme un point dans un environnement supposé connu et statique et le deuxième la construction des modèles géométriques qui capturent l'évolution des incertitudes sur le robot. L'hypothèse d'un robot ponctuel suppose la re-formulation du problème de planification dans l'espace des configurations.



La dimension de cet espace est fonction des degrés de liberté du robot considéré. Dans cet espace le robot réel est transformé en un point ce qui est un avantage. En revanche les algorithmes qui l'explorent ainsi que la représentation des obstacles sont beaucoup plus complexes. En ce qui concerne les modèles géométriques d'incertitude leur objectif consiste à prédire les positions potentiellement atteignables par le robot depuis une position incertaine lors d'un déplacement également incertain. Afin d'élargir les positions atteignables par le robot les surfaces des obstacles sont utilisées pour le robot comme guides géométriques ce qui permet aussi de réduire les incertitudes relatives entre le robot et les obstacles. La complexité de ce type d'approches dépend donc de la dimension de l'espace des configurations à explorer du principe d'exploration de cet espace et de la complexité des modèles géométriques appliqués.

### 1.1.2 Approches probabilistes

Une approche intégrant différemment les incertitudes appelée *vérification/correction de programme* est présentée par Puget [Pug89]. Les incertitudes considérées portent sur la position/orientation relatives entre deux objets tridimensionnels dans l'environnement et sur l'effet des actions réalisées par un robot manipulateur. Une erreur  $\epsilon$  de position/orientation est représentée par un vecteur de  $\mathbb{R}^6$  (trois coordonnées pour l'erreur de position et trois coordonnées pour l'erreur d'orientation) (figure 1.6).

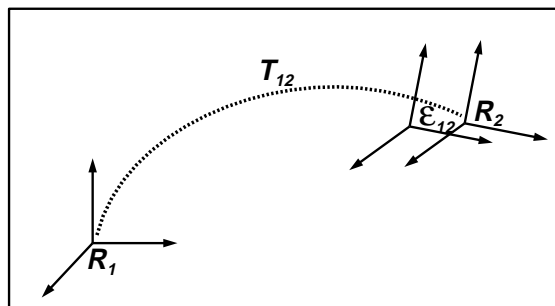


FIG. 1.6 – La position nominale du repère  $R_2$  par rapport au repère  $R_1$  est représentée par la transformation géométrique  $T_{12}$ . La différence estimée entre la position nominale et la position réelle est représentée par la transformation géométrique  $\epsilon_{12}$ . [Pug89]

Cette représentation considère  $\epsilon$  comme une variable aléatoire réelle de dimension six. L'incertitude sur  $\epsilon$  est donnée sous la forme d'une loi de probabilité gaussienne centrée dans  $\mathbb{R}^6$  celle-ci est représentée par une matrice de covariance de dimension  $6 \times 6$ . Pour chaque action dans un programme la vérification consiste à décider si l'état initial des objets intervenant dans l'action est satisfaisant vis-à-vis des incertitudes pour exécuter celle-ci. Cette vérification est faite grâce à un mécanisme de propagation en avant des incertitudes. Dans le cas positif la nouvelle position/orientation des objets est déterminée ainsi que leurs incertitudes. Ensuite l'action suivante dans le programme est vérifiée. Dans le cas négatif une correction du programme est faite en appliquant un mécanisme de propagation en arrière des contraintes d'incertitudes ceci dans le but

d'ajouter des actions locales pour réduire d'avantage l'incertitude. Or le choix d'une action correctrice et du lieu d'insertion de celle-ci dans le plan d'actions n'est pas un problème facile et peut donc être considéré comme l'inconvénient majeur de cette approche. Cependant des mesures capteurs peuvent aussi être ajoutées afin d'améliorer l'estimation de la position/orientation des objets dans l'environnement. Ces mesures sont entachées d'incertitude et intégrées au modèle de l'environnement à l'aide du filtre de Kalman [AF88] [Ram88].

Dans une problématique semblable à celle traitée dans [Pug89] Mazon [Maz90] propose un module de vérification d'incertitudes dans le système de planification SPARA. Ce module est basé sur une propagation symbolique et probabiliste effectuée vers l'avant des incertitudes liées à la position/orientation relatives entre deux objets tridimensionnels sur lesquels une action va être réalisée. La représentation d'une erreur  $\epsilon$  de position/orientation est représentée par un vecteur de  $\mathbb{R}^6$  et l'incertitude associée par une matrice de covariance de dimension  $6 \times 6$ . Les actions considérées pour un bras manipulateur muni d'une pince sont de type "prendre" et "poser". Une action est maintenue dans un plan engendré si l'ensemble des positions/orientations estimé des objets intervenant dans l'action appartient au domaine admissible de celle-ci. Dans le cas contraire une autre action doit être considérée. Un plan d'actions validé par ce module est un plan correct vis-à-vis des incertitudes de position/orientation sur les objets manipulés. Des mesures capteurs peuvent être utilisées afin de mener à bien une action locale cependant celles-ci ne sont pas intégrées au modèle de l'environnement. L'application de cette approche suppose d'une part une "bonne" estimation des incertitudes initiales par rapport aux mesures capteurs et d'autre part un robot disposant d'une "bonne" précision avec des erreurs non cumulatives.

Dans le domaine de la robotique mobile l'incertitude sur la localisation (position/orientation) du robot est inhérente et ne peut être en aucun cas négligée. Cordewener et Meizel [CM93] ont proposé une approche d'estimation de la localisation pour un robot de type voiture<sup>2</sup> navigant dans un environnement connu. Le robot est modélisé par un rectangle et l'environnement par un ensemble de cartes bidimensionnelles composées de segments-obstacles. La problématique traitée consiste à faire suivre au robot un chemin nominal sans collision planifié auparavant. En raison de l'accumulation des incertitudes sur la localisation du robot et des imprécisions sur l'environnement le suivi d'un chemin nominal peut échouer. L'approche proposée consiste à estimer périodiquement la localisation du robot et son incertitude associée à l'aide des acquisitions odométriques<sup>3</sup>. Des mesures extéroceptives (e.g. mesures des capteurs à ultrasons) peuvent être intégrées pour préciser cette estimation car l'environnement est supposé connu. La mise à jour de l'estimation est effectuée à l'aide du filtre de Kalman. L'in-

---

2. Un robot de type voiture est soumis à une contrainte cinématique appelée *non-holonomie* qui contraint les orientations possibles du robot lors d'un déplacement; comme lorsque l'on conduit une voiture.

3. Odomètre: instrument servant à mesurer un trajet parcouru par une voiture.

certitude est caractérisée par une matrice de variance-covariance. Pour une probabilité donnée l'a matérialisation de l'incertitude correspond à une ellipse pour la position et un secteur angulaire pour l'orientation (figure 1.7). Lorsqu'une collision est prévisible

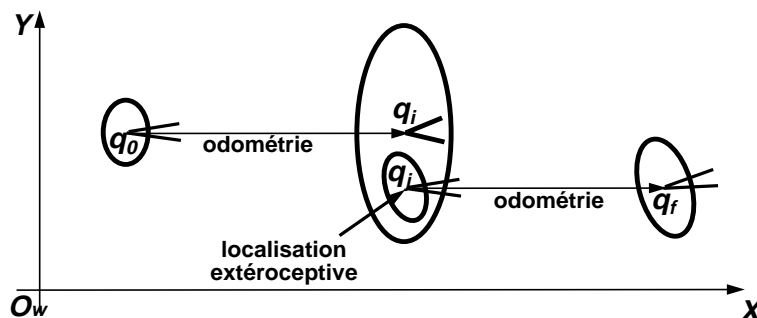


FIG. 1.7 – Evolution de l'incertitude le long du chemin entre les configurations  $q_0$  et  $q_f$  en appliquant un modèle probabiliste. L'incertitude de position est matérialisée par une ellipse autour de la configuration nominale alors que celle d'orientation est matérialisée par un secteur circulaire.[Cor95]

des mesures supplémentaires sur l'obstacle potentiel de collision sont prises pour tenter de réduire les incertitudes et engendrer ainsi une distance de collision plus fiable. La nouvelle estimation sur la localisation tiendra mieux compte de l'encombrement de l'environnement et permettra d'adapter en ligne la vitesse du robot. Des simulations par ordinateur ont validé cette approche. Les travaux concernant la localisation du robot ont été ensuite complétés et validés en expérimentation par Cordewener [Cor95]. Les expériences montrent l'estimation de la localisation statique et dynamique du robot.

Une approche probabiliste basée sur les informations récupérées par le robot le long des déplacements est présentée par Barraquand et Ferbach [BF95]. Cette approche génère des plans d'actions simples pour un robot circulaire qui se déplace en translation dans un plan sous des contraintes d'incertitude de position et de contrôle et entouré d'obstacles polygonaux. Des régions dans l'espace libre appelées "landmarks" permettent au robot de se re-localiser et donc de réduire son incertitude de position. L'espace réel du robot et son espace des configurations correspondant sont représentés dans une grille rectangulaire. Les commandes de contrôle permettent au robot d'avancer vers une des cases voisines dans la grille. L'incertitude de contrôle est modélisée par une perturbation aléatoire sur le contrôle l'incertitude qui dépend de la position réelle du robot et de la commande de contrôle à suivre. Une probabilité est affectée à chacune des cases sans collision pouvant être atteinte par le robot après le suivi d'une commande de contrôle. A chaque position atteinte par le robot (y compris la position initiale) on affecte une récompense. L'objectif consiste donc à maximiser la valeur accumulée de récompense le long du chemin du robot. Depuis chaque position atteinte on peut aussi obtenir des informations sur l'environnement qui dépendront de la position et des acquisitions effectuées par le(s) système(s) sensoriel(s) du robot. Donc la connaissance

de l'environnement retenue par le robot augmente au fur et à mesure que celui-ci se déplace. L'état de connaissance de l'environnement après chaque déplacement du robot est représenté par *un vecteur d'information*. Etant donné un nombre fixe de déplacements du robot l'ensemble des vecteurs d'information définit *l'espace d'information* du robot. L'approche proposée applique le paradigme de la programmation dynamique stochastique dans une approximation de l'espace d'information. Le principe du paradigme est basé sur une stratégie de chaînage arrière partant du vecteur d'information acquis à la position but. A chaque pas on cherche une commande de contrôle qui maximise la récompense attendue selon les informations récupérées. Un plan engendré par l'approche est composé de  $N$  commandes de contrôle ayant la probabilité la plus élevée pour atteindre la position but depuis la position initiale. Les inconvénients les plus importants de cette approche résident dans sa complexité algorithmique et dans l'espace mémoire nécessaire ce qui réduit son applicabilité pratique.

## 1.2 Le traitement implicite des incertitudes

Parmi les tâches robotiques on trouve un sous-ensemble où les incertitudes peuvent affecter le résultat final de la tâche par exemple dans les tâches d'assemblage et de navigation pour un robot mobile. Cependant certaines méthodes de résolution s'avèrent valides pour des tâches de ce type sans inclure explicitement une analyse des incertitudes. En effet le raisonnement (principalement de type géométrique) appliqué dans ces approches permet d'obtenir des actions "robustes" vis-à-vis des incertitudes. Un ensemble de tâches classiques où les actions par elles mêmes permettent de réduire les incertitudes est celui composé des tâches d'assemblage. En effet la résolution d'une telle tâche consiste à trouver une séquence de situations de contacts stables qui conduisent un objet manipulé vers sa situation d'assemblage. Ainsi un contact devient un guide géométrique pour déterminer les actions suivantes en gardant des estimations d'incertitude faibles. De nombreuses approches de résolution dans ce domaine ont été proposées. Dans le cadre de notre travail ces approches sont intéressantes car leur principe de résolution utilise une analyse explicite des contacts entre les obstacles et un objet manipulé afin de guider ce dernier tout en maintenant des incertitudes faibles. Ces approches mettent en évidence l'importance des contacts entre les objets si l'on veut obtenir des actions "robustes".

Dans la suite nous faisons référence à quelques travaux qui traitent du problème de la planification de tâches d'assemblage. Nous ne les présentons pas en détail car il nous semble que cela dépasse le contexte de notre travail. Des travaux plus pertinents sur ce domaine sont ceux de Laugier [Lau87] Sidobre [Sid91] et Nájera [Naj95].

Laugier [Lau89] considère le problème de l'assemblage de deux pièces polyédriques dont une est appelée la pièce manipulée. L'approche proposée engendre des mouvements fins en exploitant une représentation explicite de l'espace des contacts. Cette approche

applique une méthode chaînage arrière (“backchaining”) afin de réduire l’espace d’exploration. A chaque pas une analyse sur la situation de contact courante de la pièce manipulée est faite pour déterminer les déplacements qui rompent progressivement ce contact. Le passage progressif entre situations de contact permet une réduction progressive des incertitudes de position et d’orientation de la pièce manipulée. L’approche est composée de deux phases :

1. une phase d’analyse dont le but est de construire un graphe des solutions du problème de démontage de la pièce manipulée et
2. une phase de recherche dont le but consiste à choisir le meilleur chemin inversé dans le graphe pour résoudre le problème d’assemblage original. Les critères pour choisir ce chemin sont le nombre et la robustesse (faisabilité) des déplacements à réaliser.

Giraud et Sidobre [GS92] considèrent le problème de démontage de deux objets polyédriques dont un est appelé l’objet manipulé. Une approche pour résoudre ce problème est proposée. Elle est basée sur une méthode de “rétraction” de l’espace des configurations. La frontière de l’espace des contacts est représentée par un graphe de contacts construit incrémentalement. Un nœud dans ce graphe correspond à un ensemble connexe de configurations dont la nature du contact est la même. Un sous-ensemble de contacts est accepté selon une condition de non-collision évaluée sur des contacts de base. Une transition entre deux nœuds du graphe est représentée par un arc si le passage est possible par l’exécution d’un déplacement simple en translation ou en rotation sur l’objet manipulé. Un coût est affecté à chaque arc selon une heuristique locale qui tient compte de la distance à la configuration but de démontage des considérations physiques et des incertitudes. L’approche est composée de deux étapes. Dans une première phase les configurations initiale (objets assemblés) et but de l’objet manipulé sont “projetées” sur deux configurations atteignables à l’intérieur des ensembles de contact valides. Ensuite la deuxième étape consiste à construire incrémentalement le graphe d’exploration depuis la configuration initiale projetée jusqu’à l’obtention de la configuration but projetée. A chaque pas de la construction le chemin le moins cher dans le graphe est exploré.

Dans les approches qui étudient la planification des tâches d’assemblage la clef du raisonnement géométrique pour la résolution d’un problème consiste à déterminer les degrés de liberté locaux de la pièce manipulée pour évaluer les possibles progressions de celle-ci. Mattikalli et Khosla [MK92] ont proposé une représentation géométrique des translations et rotations interdites entre deux objets polygonaux ou polyédriques en contact. Cette représentation est basée sur une analyse des degrés de liberté perdus à cause de la nature du contact. Les représentations obtenues peuvent être intégrées dans des systèmes de planification raisonnant sur le problème de démontage d’objets.

## 1.3 La complexité algorithmique

Pour évaluer la complexité d'un algorithme<sup>4</sup> on étudie son comportement (temps d'exécution et place mémoire) en fonction de ses données d'entrée. Si  $n$  caractérise la taille des données d'entrée d'un problème (e.g. le nombre d'arêtes d'un environnement polygonal) alors un algorithme qui le résout en temps  $f(n)$  prouve l'existence d'une borne supérieure en  $\mathcal{O}(f(n))$ . Un algorithme est dit *polynomial* si sa complexité croît polynomialement en fonction de  $n$  c'est-à-dire qu'il résout les problèmes en temps  $\mathcal{O}(f(n))$  avec  $f$  un polynôme en fonction de  $n$ . Un algorithme non polynomial est dit *exponentiel*.

On distingue quatre classes de problèmes par ordre de difficulté croissante :

1. la classe **P** : les problèmes pour lesquels on connaît au moins un algorithme polynomial;
2. la classe **NP** : les problèmes qui peuvent être résolus en temps polynomial sur une machine non déterministe. Une telle machine fait toujours le bon choix face à une alternative. On ne sait pas si ces problèmes sont ou non polynomiaux sur une machine déterministe;
3. la classe **PEspace** : les problèmes qui peuvent être résolus avec une mémoire polynomiale par un algorithme non déterministe;
4. la classe **NExp** : les problèmes qui peuvent être résolus en temps exponentiel sur une machine non déterministe.

Un problème est *dur* pour une classe si tous les problèmes de la classe peuvent s'y ramener par une transformation polynomiale.

Les principaux résultats concernant l'analyse de la complexité algorithmique du principe de résolution de chaînage arrière appliqué au problème de planification de mouvement posé en [LMT84] sont présentés ci-dessous. Nous rappelons que ce problème considère un robot ponctuel dans un espace parsemé d'obstacles polygonaux (dans le cas bidimensionnel) ou polyédriques (dans le cas tridimensionnel) et contraint par les incertitudes de position et de contrôle et par le frottement de contact. Les résultats de complexité sont les suivants :

- dans l'espace  $\mathbb{R}^3$  le problème posé en [LMT84] dont la solution n'est composée que d'une vitesse (i.e. une commande) appartient à la classe "NP-dur" [CR87].
- dans l'espace  $\mathbb{R}^3$  le problème posé en [LMT84] dont la solution est composée de multiples vitesses (i.e. multiples commandes) appartient à la classe "PEspace-

---

4. Un ouvrage de référence sur la complexité algorithmique est celui de Hopcroft et Ullman [HU79].

dur” [Nat86ΓNat88]. De plusΓdans le cas où le nombre de faces dans l'espace valide<sup>5</sup> n'est pas fixeΓle problème appartient à la classe “NExp-dur” [CR87].

- dans l'espace  $\mathbb{R}^2$ Γla borne supérieure au problème posé en [LMT84] dont la solution n'est composée que d'une vitesse (i.e. une commande) est  $\mathcal{O}(n^4 \log n)$  avec  $n$  le nombre d'arêtes composant l'environnement [Don88a].
- dans l'espace  $\mathbb{R}^2$ Γla borne supérieure au problème posé en [LMT84] dont la solution est composée de multiples vitesses (i.e. multiples commandes) est  $n^{r^{\mathcal{O}(1)}}$  avec  $n$  le nombre d'arêtes composant l'environnement et  $r$  le nombre de vitesses dans la solution [Don88a].

A notre avisΓune source des complexités présentées ci-dessus provient de la puissance du modèle géométrique de *pré-image* [LMT84]. Cela est dû au fait que ce modèle contient l'ensemble des configurations de départ pour le robot lui permettant d'atteindre une région butΓmalgré les incertitudes et le frottement de contactΓen suivant une vitesse simple. La construction d'un tel modèle est complexe. D'autre partΓune *pré-image* peut être composée de nombreuses sous-régions pouvant devenir par la suite des sous-buts pour le robot. L'identification ainsi que l'analyse des différentes sous-régions font augmenter la complexité du principe de résolution. Un modèle géométrique plus simple permettrait une construction et une analyse moins complexes des régions atteignables par le robot au détriment du nombre de vitesses à suivre par le robot. En faitΓdes modèles plus simples permettraient le développement et la mise en œuvre des implantations sur ordinateur plus performantes.

## 1.4 Conclusion

La prise en compte des incertitudes dans la planification robotique est devenue de plus en plus importante. Ceci s'explique par le fait que les tâches réalisées par les robots sont de plus en plus complexes et queΓpour aboutir au résultat souhaitéΓles incertitudes ne peuvent pas être négligées. CependantΓl'intégration de l'effet des incertitudes dans le processus de planification fait augmenter énormément la complexité des problèmes traités etΓévidemmentΓcelles de leurs approches de résolution. Pour faire face à la complexité algorithmique du problème de planification de mouvements en robotique sous des contraintes d'incertitudes géométriquesΓles méthodes de résolution proposées considèrent des hypothèses simplificatrices sur certains aspects du problème (e.g. la nature des incertitudes prises en compteΓla géométrie et/ou la mobilité du robotΓla géométrie et/ou le nombre d'obstacles dans l'environnementΓ...). Ces simplifications conduisent à des principes de résolution avec des limitations pour la résolution des problèmes réels. Il est donc nécessaireΓd'une part de relâcher certaines hypothèses afin

---

5. Espace valide : sous-espace où le robot est en contact ou non avec les obstacles, c'est-à-dire que les collisions sont exclues.

---

d'obtenir des approches plus réalistes et d'autre part de réviser le principe de résolution afin d'obtenir une complexité algorithmique moins importante.

L'intégration des incertitudes dans le processus de planification ne peut se faire que si l'on dispose d'un moyen qui permette de connaître l'état réel du robot dans son environnement réel pendant l'exécution d'une tâche. Ce moyen est constitué par les informations perceptives acquises par le robot grâce à ses systèmes sensoriels. Ces informations pour leur part sont fortement liées à la nature de la tâche à accomplir.

Dans la suite de ce mémoire nous analysons une stratégie de planification automatique de mouvement que nous proposons pour générer de chemins "robustes" vis-à-vis des incertitudes de position d'orientation et de contrôle pour un robot polygonal convexe entouré d'obstacles polygonaux (convexes ou concaves). La contrainte de frottement de contact n'est pas considérée initialement mais peut être intégrée (si besoin il y a) sans affecter vraiment la complexité de la stratégie.





# Chapitre 2

## Les incertitudes géométriques dans la planification robotique

---

### 2.1 Introduction

Les imprécisions entre le monde réel dans lequel le robot évolue et son modèle nominal dans lequel la planification est faite seront appelées *les incertitudes*. Dans certaines tâches robotiques (e.g. tâches d'assemblage, navigation d'un robot mobile) l'évolution du robot dans le monde réel peut causer une évolution importante des incertitudes liées à celui-ci. Si cette évolution dépasse les tolérances acceptées pour la tâche à accomplir, les déplacements nominaux corrects planifiés pourraient ne plus être valables, ce qui met en cause le succès de toute la tâche. C'est pourquoi il est nécessaire d'intégrer une analyse sur l'incertitude pendant la planification. Cette analyse doit garantir des plans qui atteignent leur but si le comportement de l'incertitude durant l'exécution reste semblable à celui présumé lors de la planification. Un tel plan sera appelé un *plan robuste* et les mouvements qui le composent seront appelés des *mouvements robustes*.

Parmi les incertitudes qui peuvent perturber l'exécution d'une tâche, nous nous intéressons à un sous-ensemble portant sur des données géométriques (position, orientation, vecteur de déplacement) propres du robot. Pour cette raison, les incertitudes dans ce sous-ensemble seront appelées les *incertitudes géométriques*. Dans ce chapitre, les incertitudes géométriques considérées ainsi que leurs représentations dans l'approche proposée sont présentées. Les différents modèles d'incertitude sont définis dans *l'espace des configurations*. Ce dernier a été introduit dans le domaine de la planification de mouvement par Lozano-Pérez [Loz83]. Par souci de clarté, les définitions de robot et d'obstacle dans l'espace des configurations sont présentées ci-dessous, puis les modèles d'incertitudes géométriques sont explicités.

## 2.2 Robot et obstacles dans l'espace des configurations

En anticipant sur le problème de planification de mouvement qui nous intéresse plus particulièrement, la définition et la représentation du robot et des obstacles considérés sont données dans l'espace réel ainsi que dans l'espace des configurations. Ceci permettra de mieux appréhender la modélisation proposée des incertitudes géométriques.

### Définition 2.1 - Polygone simple

Un *polygone simple*  $P \subset \mathbb{R}^2$  est défini comme une séquence ordonnée de sommets  $\{s_0^P, \dots, s_{k-1}^P\}$  de  $\mathbb{R}^2$  avec  $k$  supérieur à 2. L'arête partant du sommet  $s_i^P$  est notée  $\vec{E}_i^P$  et son vecteur normal extérieur est noté  $\vec{n}_i^P$ . Nous supposons que l'ordre de la séquence correspond à un parcours des sommets avec "matière à gauche" (figure 2.1).

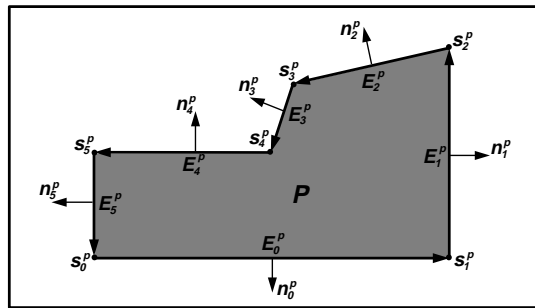


FIG. 2.1 – Définition d'un polygone simple  $P$ .

### Définition 2.2 - Espace de travail

L'espace réel où le robot évolue est appelé *l'espace de travail*  $\mathcal{W}$ . Un repère cartésien fixe  $\mathcal{F}_\mathcal{W}$  dont  $\mathcal{O}_\mathcal{W}$  est l'origine est attaché à  $\mathcal{W}$ . Le sous-ensemble de problèmes de planification que nous considérons se situe dans un espace  $\mathcal{W}$  plan représenté dans  $\mathbb{R}^2$ . Dans  $\mathcal{W}$  peuvent se trouver un robot et un ensemble d'obstacles.

### Définition 2.3 - Robot

Le robot (noté  $\mathcal{A}$ ) est un objet mobile et rigide.  $\mathcal{A}$  est défini par sa géométrie et ses degrés de liberté (d.d.l.). Un repère cartésien fixe  $\mathcal{F}_\mathcal{A}$  dont  $\mathcal{O}_\mathcal{A}$  est l'origine est attaché au robot  $\mathcal{A}$ . Dans  $\mathcal{F}_\mathcal{A}$  la géométrie de  $\mathcal{A}$  est définie comme un polygone simple et convexe. Le robot  $\mathcal{A}$  possède deux d.d.l. en translation et un d.d.l. en rotation. Nous supposons qu'il n'y a pas de contraintes cinématiques sur les mouvements du robot. Ceci veut dire que nous appliquons la cinématique du point au robot  $\mathcal{A}$  :  $\mathcal{A}$  peut se déplacer en translation suivant toute direction en  $\mathcal{W}$  et peut tourner autour d'un axe de rotation perpendiculaire à  $\mathcal{W}$ . De ce fait on dit que  $\mathcal{A}$  est un *robot holonome*. En raison de la *propriété d'holonomie* les déplacements en translation ne sont pas contraints par l'orientation du robot donc le robot  $\mathcal{A}$  peut se déplacer librement sur chacun de ses

d.d.l..

### Définition 2.4 - Obstacle

Un obstacle  $\mathcal{B}_i$  est un objet statique et rigide.  $\mathcal{B}_i$  est défini par sa géométrie et sa position/orientation dans  $\mathcal{W}$ . Un repère cartésien fixe  $\mathcal{F}_i$  dont  $\mathcal{O}_i$  est l'origine est attaché à  $\mathcal{B}_i$ . Dans  $\mathcal{F}_i$  la géométrie de  $\mathcal{B}_i$  est définie comme un polygone simple convexe ou concave. La frontière de  $\mathcal{B}_i$  est référencée comme  $\partial\mathcal{B}_i$ . La position de  $\mathcal{B}_i$  dans  $\mathcal{W}$  est donnée par la position  $(x_i, y_i)$  de l'origine  $\mathcal{O}_i$  dans  $\mathcal{F}_i$  et son orientation par l'angle  $\theta_i$  dans le sens trigonométrique entre l'axe canonique  $X$  de  $\mathcal{F}_i$  et l'axe canonique  $X$  de  $\mathcal{F}_W$ . La position/orientation de  $\mathcal{B}_i$  dans  $\mathcal{W}$  est représentée par le triplet  $(x_i, y_i, \theta_i)$  dans  $\mathbb{R}^2 \times [0, 2\pi]$ .

### Définition 2.5 - Configuration

Une *configuration*  $q$  de  $\mathcal{A}$  est une représentation de la position/orientation de  $\mathcal{A}$  dans  $\mathcal{W}$ . La position de  $\mathcal{A}$  dans  $\mathcal{W}$  est donnée par la position  $(x_A, y_A)$  de l'origine  $\mathcal{O}_A$  dans  $\mathcal{F}_W$  et son orientation par l'angle  $\theta_A$  dans le sens trigonométrique entre l'axe canonique  $X$  de  $\mathcal{F}_A$  et l'axe canonique  $X$  de  $\mathcal{F}_W$ . La position/orientation de  $\mathcal{A}$  dans  $\mathcal{W}$  est représentée par le triplet  $q = (x_A, y_A, \theta_A)$  dans  $\mathbb{R}^2 \times [0, 2\pi]$ .  $q$  détermine d'une manière unique le placement de  $\mathcal{A}$  dans  $\mathcal{W}$ . Le sous-ensemble de  $\mathcal{W}$  occupé par  $\mathcal{A}$  placé à la configuration  $q$  est référencé par  $\mathcal{A}(q)$ .

### Définition 2.6 - Espace des configurations

L'espace de toutes les configurations possibles de  $\mathcal{A}$  est appelé *l'espace des configurations*  $\mathcal{C}$ . Dans  $\mathcal{C}$  le robot  $\mathcal{A}$  se ramène à une configuration (un point)  $q$  et de ce fait on parle du robot ponctuel  $q_A$ . Dans notre cas  $\mathcal{C} = \mathbb{R}^2 \times [0, 2\pi]$ . Un triplet  $(x, y, \theta)$  représente la position/orientation de  $\mathcal{A}$  en  $\mathcal{C}$ . Les coordonnées  $x$  et  $y$  donnent la position de  $\mathcal{O}_A$  dans  $\mathcal{W}$  et la coordonnée  $\theta$  correspond à l'orientation de  $\mathcal{F}_A$  par rapport à  $\mathcal{F}_W$ . Les axes de  $\mathcal{C}$  seront référencés comme  $X, Y$  et  $\Theta$ . Un déplacement de  $q_A$  dans le plan  $XY$  correspond à une translation de  $\mathcal{A}$  dans  $\mathcal{W}$  et un déplacement de  $q_A$  parallèle à l'axe  $\Theta$  correspond à une rotation de  $\mathcal{A}$  dans  $\mathcal{W}$  autour d'un axe perpendiculaire à  $\mathcal{W}$  et placé sur le point  $\mathcal{O}_A$  de  $\mathcal{A}$ .

### Définition 2.7 - Obstacle dans l'espace des configurations

Supposons que le robot  $\mathcal{A}$  garde une orientation  $\theta$ . Un obstacle  $\mathcal{B}_i$  dans  $\mathcal{W}$  devient un *C-obstacle*  $\mathcal{CB}_{i,\theta}$  dans  $\mathcal{C}$  défini comme :

$$\mathcal{CB}_{i,\theta} = \{q = (x, y, \theta) \in \mathcal{C} \mid \mathcal{A}(q) \cap \mathcal{B}_i \neq \emptyset, \theta \text{ fixe}\} \quad (2.1)$$

Pour  $\mathcal{A}$  et  $\mathcal{B}_i$  polygonaux  $\mathcal{CB}_{i,\theta}$  définit un *C-obstacle* polygonal et sa frontière est référencée comme  $\partial\mathcal{CB}_{i,\theta}$ .  $\partial\mathcal{CB}_{i,\theta}$  correspond au parcours du point  $\mathcal{O}_A$  de  $\mathcal{A}$  autour de  $\mathcal{B}_i$  (figure 2.2).

Supposons maintenant que le robot  $\mathcal{A}$  puisse tourner autour d'un axe perpendiculaire à  $\mathcal{W}$  passant par  $\mathcal{O}_A$ . Un obstacle  $\mathcal{B}_i$  dans  $\mathcal{W}$  devient un *C-obstacle*  $\mathcal{CB}_i$  dans  $\mathcal{C}$  défini

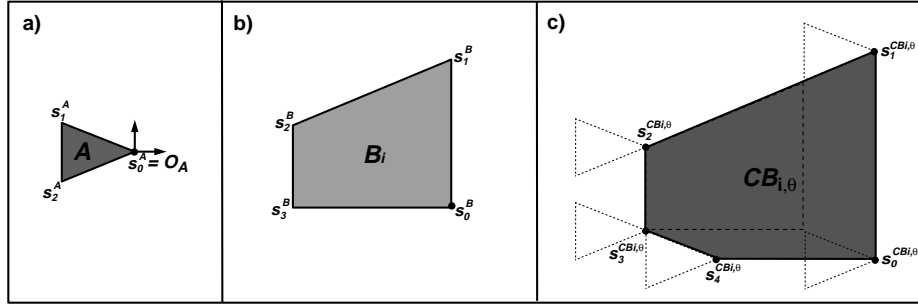


FIG. 2.2 – a) Le robot convexe  $A$  fixe d'orientation  $\theta$ . b) Un obstacle convexe  $B_i$ . c) Le  $C$ -obstacle  $CB_{i,\theta}$  construit comme le parcours du point de référence  $O_A$  de  $A$  autour de l'obstacle  $B_i$ .

comme la composition des tranches  $CB_{i,\theta}$  pour chaque angle  $\theta \in [0, 2\pi]$  (figure 2.3) :

$$CB_i = \bigcup_{\theta \in [0, 2\pi]} CB_{i,\theta} \quad (2.2)$$

*Remarque :* par construction une tranche  $CB_{i,\theta}$  de la surface du  $C$ -obstacle  $CB_i$  est un polygone simple.

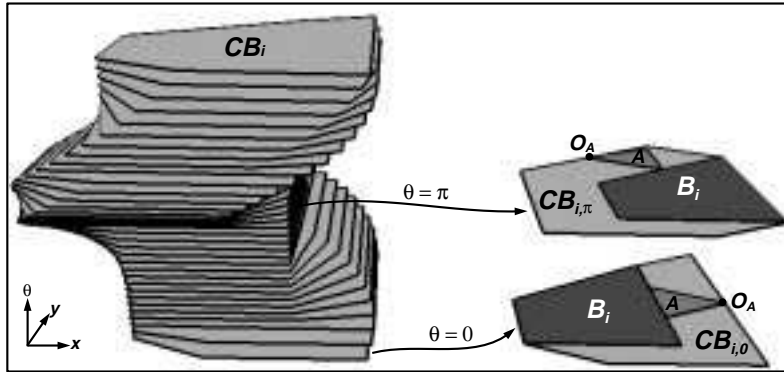


FIG. 2.3 – Construction discrète du  $C$ -obstacle  $CB_i$  dans  $\mathbb{R}^2 \times [0, 2\pi]$  à partir de la composition de tranches  $CB_{i,\theta}$  dans  $\mathbb{R}^2$  pour un ensemble discret d'angles  $\theta$  dans l'intervalle  $[0, 2\pi]$ .

## 2.3 Modélisation des incertitudes géométriques

Les incertitudes géométriques de base considérées ici sont liées d'une part à la localisation (i.e. position et orientation) du robot et d'autre part à la direction d'un déplacement (i.e. suivi d'un vecteur vitesse). Nous supposons que la localisation et le suivi d'un vecteur vitesse pour le robot seront déterminés à l'aide de capteurs dont

les valeurs peuvent être bornées. Puisque la nature des mesures pour la position et l'orientation sont différentes les *incertitudes de position* seront considérées séparément des *incertitudes d'orientation*. De plus les incertitudes sur les vitesses seront appelées les *incertitudes de contrôle*.

### 2.3.1 L'incertitude de position

L'incertitude de position correspond aux écarts entre la position nominale déterminée par les capteurs et la position réelle où le robot se trouve. Cette incertitude est liée à chaque degré de liberté (d.d.l.) en translation du robot. Le modèle d'incertitude de position dépendra donc du nombre de d.d.l. en translation du robot. Pour un robot à 1 d.d.l. en translation si la position nominale est  $x \in \mathbb{R}$  la position réelle  $x'$  se trouvera à l'intérieur de l'intervalle  $[x - \rho, x + \rho] \subset \mathbb{R}$  le long de l'axe de mouvement du robot dont  $\rho \in \mathbb{R}^+$  correspond à l'écart maximal estimé. Dans ce cas l'incertitude peut être donc modélisée par  $x \pm \rho$  et l'axe  $X$  de déplacement (figure 2.4a).

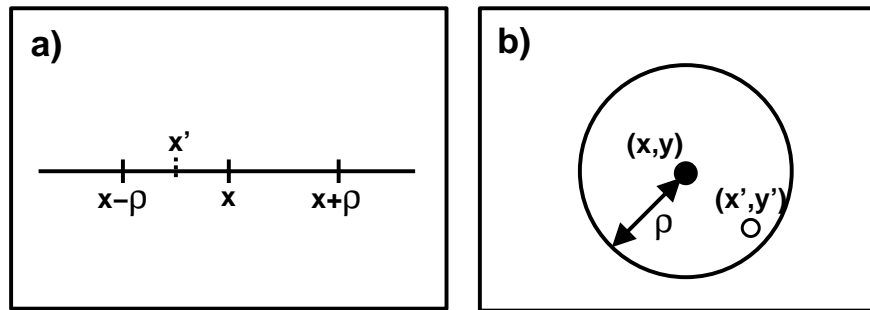


FIG. 2.4 – a) L'intervalle d'incertitude de position dans l'espace  $\mathbb{R}$ . b) Le disque d'incertitude de position dans l'espace  $\mathbb{R}^2$ .

Pour un robot à 2 d.d.l. en translation (i.e. espace d'évolution  $\mathbb{R}^2$ ) nous supposons une incertitude homogène  $\rho$  dans chaque direction de mouvement à partir d'une position nominale  $(x, y)$  ceci engendre un disque d'incertitude centré en  $(x, y)$  de rayon  $\rho$ . La position réelle sera donc un point  $(x', y')$  à l'intérieur du disque  $D(x, y) \subset \mathbb{R}^2$  défini par :

$$D(x, y) = \{(x', y') \mid (x' - x)^2 + (y' - y)^2 \leq \rho^2\} \quad (2.3)$$

Dans ce cas l'incertitude sera modélisée par  $(x, y)$  et  $\rho$  (figure 2.4b).

Pour un robot à  $n$  d.d.l. en translation l'incertitude autour d'une position nominale  $(x_1, \dots, x_n) \in \mathbb{R}^n$  sera une "boule" dans  $\mathbb{R}^n$  de rayon  $\rho$ . Cette "boule" peut être modélisée par la position  $(x_1, \dots, x_n)$  et le rayon  $\rho$ .

Grâce aux capacités sensorielles d'un robot celui-ci peut aussi se repérer par rapport à la position connue des amers<sup>1</sup> dans l'espace (e.g. des obstacles connus). De cette manière l'incertitude liée à une position nominale peut être réduite selon une ou plusieurs directions de translation : si le robot arrive à déterminer sa position sur une direction précise celui-ci réduit son incertitude de position d'une dimension. Par exemple dans le cas d'un robot à 2 d.d.l. en translation le disque d'incertitude associé à une position dans l'espace libre devient un intervalle lorsque le robot est contraint par rapport à une direction précise (e.g. contact avec une surface plane d'un obstacle) (figure 2.5a). Si le robot peut compléter ce repérage par rapport à une deuxième direction (e.g. détection d'une extrémité d'une surface plane) l'intervalle d'incertitude devient un intervalle nul (figure 2.5b). A ce moment-là le robot est complètement re-localisé par rapport à la position d'un obstacle connu dans l'environnement.

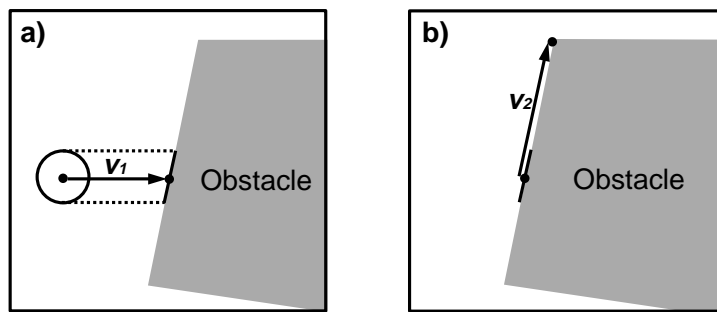


FIG. 2.5 – a) Réduction de l'incertitude de position après le déplacement suivant la direction  $\vec{v}_1$  et la rencontre d'une arête d'un obstacle connu. b) Réduction complète de l'incertitude de position après le déplacement suivant la direction  $\vec{v}_2$  et la rencontre d'un sommet d'une arête d'un obstacle connu.

## Evolution de l'incertitude de position

L'incertitude de position liée à un robot évolue lorsque celui-ci exécute des déplacements en translation. Nous supposons qu'un déplacement en translation est spécifié par un vecteur vitesse  $\vec{v}$ . Les sources possibles d'imprécision sont donc l'amplitude et la direction du vecteur  $\vec{v}$ . L'effet sur l'incertitude de position causé par le suivi d'une vitesse  $\vec{v}$  est le sujet d'étude de l'incertitude de contrôle.

### 2.3.2 L'incertitude de contrôle

L'incertitude de contrôle fait référence aux imprécisions qui surviennent lors de l'exécution d'un déplacement de translation : il se peut qu'ayant commandé au robot une vitesse nominale pour faire passer le robot d'une position initiale  $p_a$  à une position intermédiaire  $p_b$  le robot suive une vitesse effective différente. Bien évidemment l'

1. Amer : élément physique, fixe et visible servant de point de repère pour la navigation.

la conséquence d'une telle déviation est le grossissement de l'ensemble des positions potentielles où le robot peut se trouver après l'exécution de la vitesse effective. Nous analysons maintenant l'effet des écarts de direction et d'amplitude d'un vecteur vitesse  $\vec{v}$  sur la position du robot. Soient  $p_0$  la position initiale du robot dans l'espace libre et  $\vec{v}$  la vitesse commandée au robot. Nous supposons que la direction de la vitesse effective  $\vec{v}'$  suivie par le robot est à l'intérieur d'un cône droit dont le sommet est la position  $p_0$  et l'axe est la vitesse commandée  $\vec{v}$ . La déviation entre toute vitesse effective  $\vec{v}'$  et la vitesse nominale  $\vec{v}$  est bornée par un angle  $\eta$  avec  $0 \leq \eta < \pi/2$ . Ainsi défini l'angle du cône d'incertitude de contrôle est  $2 * \eta$  (figure 2.6a). Nous supposons aussi que la déviation maximale  $\eta$  reste constante pour un système de contrôle. Pour un système précis on a  $\eta = 0$  alors que plus le système est imprécis plus  $\eta$  est grand.

D'autre part nous supposons aussi une incertitude sur l'amplitude  $\|\vec{v}\|$ . Dans le cas idéal l'évolution de cette incertitude est considérée de type linéaire selon une constante de proportionnalité  $k_t \in \mathbb{R}^+$  propre au robot. Le meilleur cas se présente lorsque le robot exécute parfaitement les translations ( $k_t = 0$  cas d'un robot de manipulation) car l'incertitude sur l'amplitude est considérée comme nulle pour tout vecteur  $\vec{v}$ . Néanmoins pour un robot qui exécute les translations avec des imprécisions (cas d'un robot mobile) nous avons  $k_t > 0$  et donc plus  $k_t$  est grande plus l'incertitude sur l'amplitude augmente. Les valeurs d'amplitude potentielles pour une vitesse effective  $\vec{v}'$  sont contenues dans l'intervalle  $[\|\vec{v}\| * (1 - k_t), \|\vec{v}\| * (1 + k_t)]$ .

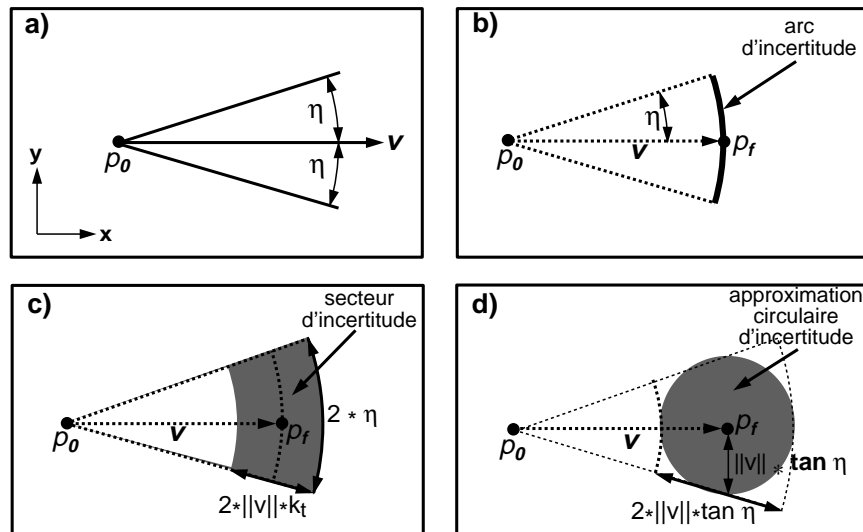


FIG. 2.6 – a) Cône d'incertitude de contrôle en  $\mathbb{R}^2$  autour de la vitesse  $\vec{v}$ .  
 b) Arc d'incertitude de position causé par la déviation angulaire de  $\vec{v}$  (l'amplitude  $\|\vec{v}\|$  du déplacement est alors supposée parfaitement connue).  
 c) Secteur de couronne circulaire d'incertitude de position causé par les imprécisions sur la direction de  $\vec{v}$  et l'amplitude  $\|\vec{v}\|$  du déplacement.  
 d) Approximation circulaire de l'incertitude de position due au suivi de  $\vec{v}$ .



## Effet de l'incertitude en contrôle sur l'incertitude de position

En raison de l'incertitude de contrôle le robot va cumuler des incertitudes en position pendant tout son déplacement déterminé par un vecteur vitesse. Soient  $p_0$  une position dans l'espace libre avec un disque d'incertitude de rayon  $\rho_0$  et  $\vec{v}$  la vitesse commandée au robot.  $\eta$  la déviation maximale sur la vitesse et  $k_t$  la constante de proportionnalité déterminant l'imprécision sur l'amplitude des vitesses. Si l'on ne considère que la déviation angulaire  $\eta$  lors du suivi de la vitesse  $\vec{v}$  l'incrément de l'incertitude de position du robot à la position finale  $p_f$  est défini par un arc de cercle d'angle  $2 * \eta$  (figure 2.6b).

Si l'on tient compte de l'incertitude sur l'amplitude de la vitesse  $\vec{v}$  l'estimation de son erreur (par défaut et par excès) est de  $\|\vec{v}\| * k_t$ . Cette estimation prise en compte sur chaque vitesse effective possible  $\vec{v}'$  donne un secteur de couronne circulaire d'incertitude dont l'angle est  $2 * \eta$  limité par les rayons  $\|\vec{v}\| * (1 - k_t)$  et  $\|\vec{v}\| * (1 + k_t)$ . Le centre du secteur est la position nominale  $p_f$  (figure 2.6c). Cette estimation dépend à la fois de  $\eta$  et de  $k_t$ . Par souci de simplicité nous allons supposer la valeur de  $k_t$  égale à  $\tan(\eta)$  ce qui relie les imprécisions sur la direction et l'amplitude d'une vitesse. De plus ceci nous permet d'approcher le secteur de couronne circulaire d'incertitude par un disque d'incertitude de rayon  $\|\vec{v}\| * \tan(\eta)$  (figure 2.6d)<sup>2</sup>. L'approximation ainsi obtenue rend plus simple l'évaluation de l'effet de l'incertitude de contrôle sur l'incertitude de position. L'estimation de la nouvelle incertitude de position sur  $p_f$  après le suivi de  $\vec{v}$  sans avoir détecté un contact est donc un disque de rayon :

$$\rho_f(\|\vec{v}\|) = \rho_0 + \|\vec{v}\| * \tan(\eta) \quad (2.4)$$

Dans le cas où le système de contrôle suit parfaitement les vitesses commandées ( $\eta = 0$ ) l'incertitude en position reste constante pour toutes les vitesses  $\vec{v}$ . Dans le cas contraire ( $\eta > 0$ ) étant donné que  $\eta$  est fixe l'incrément de l'incertitude est proportionnel à la distance parcourue suivant la vitesse commandée. En d'autres termes plus le robot va loin suivant le vecteur vitesse plus l'incertitude sur la position finale du robot est grande.

### 2.3.3 L'incertitude d'orientation

L'incertitude d'orientation prend en compte les imprécisions sur les d.d.l. en rotation du robot. Pour un robot "volant" à 1 d.d.l. en rotation le robot ne pourra tourner qu'autour d'un seul axe. Nous supposons cet axe fixe sur le robot. Pour une orientation nominale  $\theta \in [0, 2\pi]$  du robot l'incertitude associée sera une valeur  $\psi \in \mathbb{R}^+$ . Dans ce cas l'orientation réelle du robot sera un angle  $\theta' \in [\theta - \psi \pmod{2\pi}, \theta + \psi \pmod{2\pi}] \subset [0, 2\pi]$ . Dorénavant la fonction  $\text{mod } 2\pi$  sera appliquée au résultat de toute addition

2. Une approximation alternative consiste à prendre le cercle englobant le secteur de couronne circulaire, mais celui-ci peut donner lieu à une sur-estimation importante de l'incertitude de position.

et/ou soustraction entre angles. Une généralisation d'un modèle d'incertitude en orientation pour un robot à plusieurs d.d.l. en rotation n'est pas possible comme dans le cas d'un robot à plusieurs d.d.l. en translation. En effet l'hypothèse d'augmentation homogène sur chaque d.d.l. en rotation n'est plus valable car les relations entre les différents axes de rotation (e.g. axes parallèles, axes colinéaires, axes perpendiculaires sur un même plan, etc.) entraînent plutôt un effet cumulatif des incertitudes. De plus, la géométrie des voisinages d'incertitude dépendra des relations géométriques et de la position des axes de rotation sur le robot.

Les incertitudes associées aux orientations du robot peuvent aussi être réduites si le robot arrive à s'aligner sur une orientation connue dans l'environnement. Si c'est le cas, le voisinage d'incertitude est réduit d'une dimension. Par exemple, pour un robot à 1 d.d.l. en rotation avec une orientation nominale  $\theta$ , nous avons vu que le robot aura une orientation effective  $\theta' \in [\theta - \psi, \theta + \psi] \subset [0, 2\pi]$ . Si le robot se positionne parallèlement à une face d'un obstacle, l'incertitude d'orientation deviendra un intervalle nul (figure 2.7).

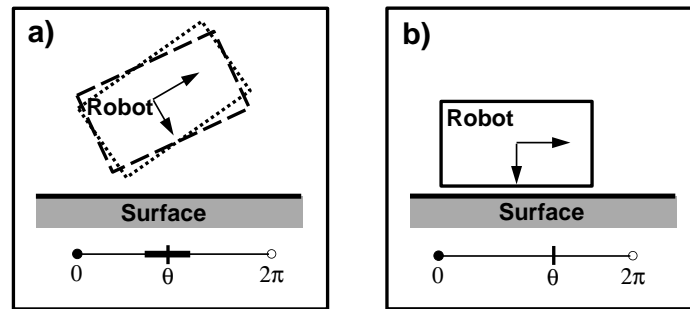


FIG. 2.7 – a) Robot rectangulaire orienté selon deux angles choisis dans l'intervalle d'incertitude d'orientation. b) Réduction complète de l'incertitude d'orientation après la mise en parallèle de surfaces.

### Evolution de l'incertitude d'orientation

Le cas analysé correspond au cas d'un robot  $\mathcal{A}$  holonome à 2 d.d.l. en translation et 1 d.d.l. en rotation. Pour ce type de robot, les déplacements en translation ne causent pas d'effet sur l'orientation du robot, donc l'incertitude de contrôle  $\eta$  liée à un vecteur vitesse nominale  $\vec{v}$  n'a aucun effet sur l'incertitude d'orientation. L'incertitude d'orientation sera causée par les imprécisions propres du robot lors de l'exécution des rotations. L'analyse de l'évolution de cette incertitude est présentée ci-dessous.

Soient  $\theta_0 \in [0, 2\pi]$  une orientation initiale du robot dans l'espace libre,  $\psi_0 \in [0, \pi]$  la valeur d'incertitude initiale associée et  $\delta\theta \in [-\pi, \pi]$  l'angle de la rotation à effectuer. Dans le cas idéal, l'évolution de l'incertitude d'orientation est considérée de type linéaire selon une constante de proportionnalité  $k_r \in \mathbb{R}^+$  propre au robot. Ainsi, si le robot

effectue une rotation d'angle  $\delta\theta$  autour de son axe de rotation sans détecter de contacts  $\Gamma$  à la valeur de l'orientation finale  $\theta_f = \theta_0 + \delta\theta$  sera associée une incertitude qui augmente linéairement selon  $\delta\theta$ . La nouvelle valeur d'incertitude est donnée par :

$$\psi_f(\delta\theta) = \psi_0 + |\delta\theta| * k_r \quad (2.5)$$

où  $k_r$  est une constante qui détermine la précision des rotations du robot. La nouvelle représentation de l'incertitude d'orientation est donc un intervalle centré en  $\theta_f$  avec une amplitude de  $2 * \psi_f(\delta\theta)$ . Dans le meilleur des cas  $\Gamma$  le robot exécute parfaitement les rotations  $\Gamma$  donc  $k_r = 0$  et l'incertitude reste toujours constante. Néanmoins  $\Gamma$  pour un robot qui tourne avec des imprécisions  $\Gamma$  nous avons  $k_r > 0$  et l'incertitude s'accumule selon l'équation 2.5.

### 2.3.4 Représentation des incertitudes dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$

Du fait que les mesures exprimant une position et une rotation sont de natures différentes  $\Gamma$  la représentation d'une position/orientation (configuration) pour un robot ne tiendra pas compte de manière homogène des différentes composantes qui constituent sa définition. La représentation de l'incertitude pour une configuration doit donc  $\Gamma$  elle aussi  $\Gamma$  intégrer des modèles de nature différente. Nous proposons par la suite une représentation de l'incertitude propre à une configuration pour un robot "volant" à 2 d.d.l. en translation et 1 d.d.l. en rotation évoluant dans un plan. Une configuration pour un tel robot est définie comme un triplet  $(x, y, \theta)$  de l'espace  $\mathbb{R}^2 \times [0, 2\pi]$  dont  $(x, y)$  (resp.  $\theta$ ) représente la position (resp. l'orientation) du robot par rapport au référentiel du plan. Soit  $q_0 = (x_0, y_0, \theta_0)$  la configuration initiale (sans contacts) du robot dans l'espace des configurations  $\mathbb{R}^2 \times [0, 2\pi]$ . D'après les modèles d'incertitude présentés précédemment  $\Gamma$  nous avons en position un disque autour de  $(x_0, y_0)$  de rayon  $\rho_0 \in \mathbb{R}^+$  et en orientation  $\Gamma$  un intervalle centré en  $\theta_0$  d'amplitude  $2\psi_0 \in [0, 2\pi]$ . Le modèle d'intégration d'incertitude tiendra compte  $\Gamma$  d'une part de l'estimation de la position  $\Gamma$  et  $\Gamma$  d'autre part  $\Gamma$  de l'estimation de l'orientation. Ainsi  $\Gamma$  l'intégration des incertitudes de position et d'orientation dans l'espace des configurations  $\mathbb{R}^2 \times [0, 2\pi]$  donne lieu à un cylindre droit  $C(q_0, \rho_0, \psi_0)$  centré en  $q_0$  dont la base  $\Gamma$  parallèle au plan  $XY$   $\Gamma$  est le disque de rayon  $\rho_0$  et la hauteur  $\Gamma$  parallèle à l'axe  $\Theta$   $\Gamma$  est l'intervalle d'amplitude  $2 * \psi_0$  (figure 2.8a) :

$$C(q_0, \rho_0, \psi_0) = \{(x', y', \theta') \mid (x' - x_0)^2 + (y' - y_0)^2 \leq \rho_0^2, \theta_0 - \psi_0 \leq \theta' \leq \theta_0 + \psi_0\} \quad (2.6)$$

Si la configuration courante du robot est précisée à l'aide d'informations perceptives sur un d.d.l.  $\Gamma$  son voisinage d'incertitude sera réduit d'une dimension  $\Gamma$  et ceci aura un effet sur sa représentation. Ainsi pour le cas représenté sur la figure 2.8a  $\Gamma$  une fois que le robot est aligné selon une orientation connue  $\Gamma$  la hauteur du cylindre est annulée et donc le cylindre d'incertitude devient le disque d'incertitude de position. En revanche  $\Gamma$  si le robot arrive au contact avec une surface d'un obstacle connu  $\Gamma$  ceci permet de préciser sa position selon une direction. Dans ce cas  $\Gamma$  le nouveau voisinage d'incertitude

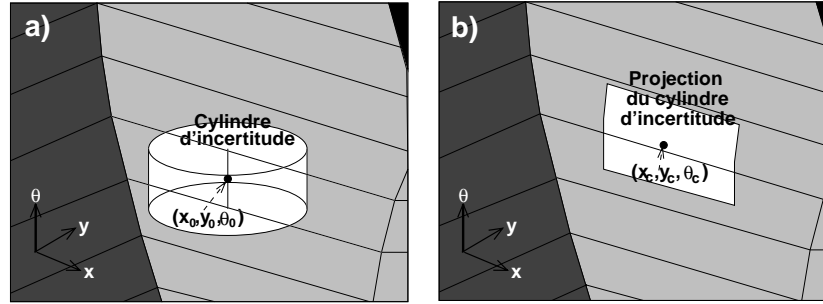


FIG. 2.8 – a) *Cylindre d’incertitude de position/orientation dans l’espace des configurations  $\mathbb{R}^2 \times [0, 2\pi]$  pour une configuration nominale du robot ne touchant pas aux obstacles.* b) *Projection du cylindre d’incertitude sur une surface dans l’espace  $\mathbb{R}^2 \times [0, 2\pi]$  pour une configuration nominale en contact avec un obstacle.*

sera la projection du cylindre sur la surface de contact en  $\mathbb{R}^2 \times [0, 2\pi]$  (figure 2.8b).

Puisque nous supposons un environnement connu et sans incertitudes, un mécanisme pour réduire les incertitudes liées à la localisation du robot consiste à re-localiser le robot par rapport aux surfaces des obstacles. Cela n’est possible que si le robot est capable de s’approcher de la surface d’un obstacle malgré les incertitudes jusqu’à une distance “satisfaisante” pour la re-localisation. Cette distance quant à elle est fonction de la tâche robotique à accomplir. Ainsi, pour une tâche d’assemblage, une pièce manipulée peut se re-localiser lorsque celle-ci s’approche d’un objet dans son environnement jusqu’à une distance nulle. Cette distance implique la définition des contacts physiques entre la pièce manipulée et les objets. Par contre, dans le cas de la navigation d’un robot mobile, le robot ne peut pas s’approcher d’un obstacle en-dessous d’une distance de sécurité  $d_s$  (dont la valeur ne peut pas être nulle) afin d’une part d’éviter d’endommager le robot et d’autre part d’assurer en tout moment sa mobilité. Une distance de proximité satisfaisante pour une tâche détermine donc un “contact” entre le robot et la surface de l’objet “touché”.

Afin de permettre au robot de s’approcher de façon sûre de la surface d’un obstacle, nous proposons un modèle géométrique qui tient compte de l’évolution des incertitudes. Ce modèle est basé sur l’évaluation de deux critères concernant les surfaces des obstacles : la “visibilité” et l’“atteignabilité”. Dans la suite, nous présentons formellement ces concepts ainsi que le modèle géométrique qui les intègre. Ce dernier est utilisé pour permettre au robot d’arriver au “contact” avec une surface d’un obstacle malgré les incertitudes.

## 2.4 “Visibilité” et “atteignabilité” des surfaces

Les critères de “visibilité” et d’“atteignabilité” servent à déterminer des régions “sûres” sur la surface d’un obstacle pouvant être approchées avec un robot ponctuel depuis une configuration initiale. En tenant compte des contraintes d’incertitudes géométriques. Le rapprochement d’une surface va jusqu’au “contact” selon le type de robot et de la tâche à effectuer. Un “contact” avec une surface permet au robot de réduire son voisinage d’incertitude de position d’une dimension. Le robot peut ensuite tenter de s’aligner avec la surface pour réduire son incertitude d’orientation. Un modèle géométrique appelé *projection en avant* est défini pour garantir l’“atteignabilité” des surfaces “visibles” malgré l’évolution des incertitudes. Par souci de simplicité le modèle initial est défini pour un robot ponctuel dans l’espace des configurations  $\mathbb{R}^2$ . Par la suite le modèle est étendu au cas d’un robot ponctuel dans l’espace des configurations  $\mathbb{R}^2 \times [0, 2\pi]$ .

Le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_{i,\theta}$  (resp.  $\mathcal{CB}_i$ ) dans l’espace des configurations  $\mathbb{R}^2$  (resp.  $\mathbb{R}^2 \times [0, 2\pi]$ ) associé à l’obstacle réel  $\mathcal{B}_i$  est défini dans §2.2 pour un robot  $\mathcal{A}$  polygonal convexe et un obstacle  $\mathcal{B}_i$  polygonal (convexe ou concave). La construction des surfaces respectives est expliquée dans §5.1.1. Nous rappelons ici les géométries des surfaces de  $\mathcal{CB}_{i,\theta}$  et  $\mathcal{CB}_i$  qui sont également illustrées avec l’exemple sur la figure 2.9. Dans l’espace  $\mathbb{R}^2$   $\mathcal{CB}_{i,\theta}$  correspond au grossissement de l’obstacle réel  $\mathcal{B}_i$  avec la géométrie du robot  $\mathcal{A}$  fixe d’orientation  $\theta$ . La surface de  $\mathcal{CB}_{i,\theta}$  résultante est un polygone. Dans l’espace  $\mathbb{R}^2 \times [0, 2\pi]$   $\mathcal{CB}_i$  correspond au grossissement de l’obstacle réel  $\mathcal{B}_i$  avec la géométrie du robot  $\mathcal{A}$  orienté selon chaque angle  $\theta \in [0, 2\pi]$ . La surface de  $\mathcal{CB}_i$  résulte de l’empilement des polygones  $\mathcal{CB}_{i,\theta}$  résultant du grossissement pour chaque orientation  $\theta$ .

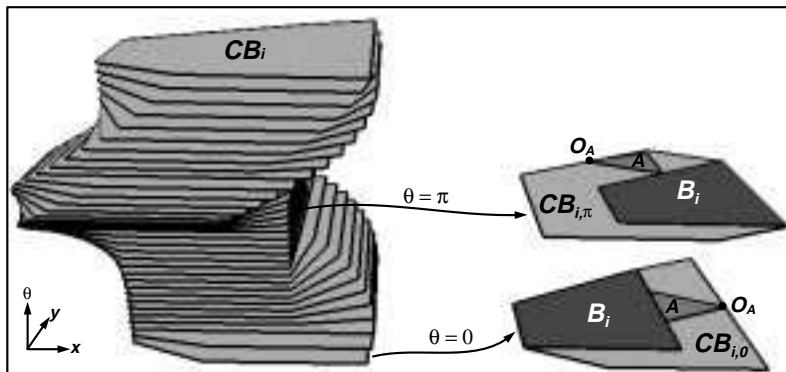


FIG. 2.9 – Construction discrète de la surface du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  dans  $\mathbb{R}^2 \times [0, 2\pi]$  à partir des polygones  $\mathcal{CB}_{i,\theta}$  résultant du grossissement de l’obstacle réel  $\mathcal{B}_i$  avec la géométrie du robot  $\mathcal{A}$  orienté selon un ensemble discret d’angles  $\theta$  dans  $[0, 2\pi]$ .

### 2.4.1 “Visibilité” d’une surface plane dans $\mathbb{R}^2$

Une surface plane  $S \subset \mathbb{R}^2$  peut être définie par l’équation :

$$S(x_1, x_2) = A_1 * x_1 + A_2 * x_2 + C = 0 \quad (2.7)$$

avec  $A_1, A_2, C \in \mathbb{R}$ . La surface  $S$  divise l’espace  $\mathbb{R}^2$  en trois sous-espaces disjoints  $S^+$ ,  $S$  et  $S^-$  (figure 2.10).  $S^+$  est appelé le *sous-espace extérieur* de  $S$  et  $S^-$  est appelé le *sous-espace intérieur* de  $S$ . Les équations respectives sont :

$$\begin{aligned} S^+(x_1, x_2) &= A_1 * x_1 + A_2 * x_2 + C < 0; \\ S^-(x_1, x_2) &= A_1 * x_1 + A_2 * x_2 + C > 0 \end{aligned} \quad (2.8)$$

Le vecteur unitaire normal à  $S$  qui pointe vers  $S^+$  est appelé le *vecteur normal extérieur* de  $S$  et est noté  $\vec{n}_S^+$ . Le *vecteur normal intérieur* de  $S$  est défini comme  $\vec{n}_S^- = -\vec{n}_S^+$ .

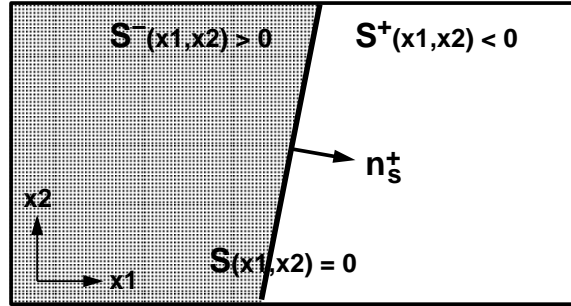


FIG. 2.10 – Définition des sous-espaces  $S^+$ ,  $S$  et  $S^-$  dans  $\mathbb{R}^2$ .

#### Définition 2.8 - Surface “visible”

Soient  $S \in \mathbb{R}^2$  une surface plane bornée<sup>3</sup> à atteindre  $p \in \mathbb{R}^2$  la position initiale du robot et  $\vec{v} \in \mathbb{R}^2$  un vecteur vitesse unitaire. La surface  $S$  est dite “visible” à partir de  $p$  selon la vitesse  $\vec{v}$  (noté “ $visible(S, p, \vec{v})$ ”) si et seulement si :

$$visible(S, p, \vec{v}) = (S^+(p) < 0) \wedge (\vec{v} \cdot \vec{n}_S^+ < 0) \quad (2.9)$$

où  $\wedge$  est le et logique et  $\cdot$  est le produit scalaire de deux vecteurs. Ce critère de “visibilité” impose d’une part que le point  $p$  soit extérieur à la surface  $S$  et d’autre part que l’angle entre les vecteurs  $\vec{v}$  et  $\vec{n}_S^+$  soit supérieur à  $\pi/2$ . En d’autres termes ce critère assure que le robot va suivre un “bon” vecteur vitesse  $\vec{v}$  depuis  $p$  afin de s’approcher de la surface  $S$  sans pour autant garantir le contact avec  $S$ . Ce critère ne tient pas compte de possibles collisions pendant le déplacement ni des incertitudes qui peuvent empêcher le contact avec  $S$ . Cependant il sert à écarter des surfaces qui ne pourront pas être atteintes à partir de  $p$  suivant  $\vec{v}$ . Finalement le critère de “visibilité” d’une surface  $S$  est nécessaire mais il n’est pas suffisant pour que  $S$  soit “atteignable” par le robot.

3. Dans  $\mathbb{R}^2$ , la surface d’intérêt  $S$  correspond à une arête  $E_i^{CB_i, \theta}$  d’un  $\mathcal{C}$ -obstacle  $CB_{i, \theta}$ .

### 2.4.2 “Atteignabilité” d’une surface plane dans $\mathbb{R}^2$

Soient  $S \subset \mathbb{R}^2$  une surface plane bornée à atteindre  $p_0 = (x_0, y_0) \in S^+$  la position initiale du robot avec  $\rho_0$  le rayon du disque d’incertitude associé  $\vec{v} \in \mathbb{R}^2$  un vecteur vitesse unitaire et  $2 * \eta$  l’angle du cône d’incertitude de contrôle.

#### Définition 2.9 - Surface “atteignable”

Une surface  $S$  est dite “atteignable” à partir de  $p_0$  selon la vitesse  $\vec{v}$  si toute vitesse effective  $\vec{v}'$  (prenant en compte l’incertitude de contrôle  $\eta$  sur  $\vec{v}$ ) appliquée depuis tout point potentiel  $p'$  (prenant en compte l’incertitude de position  $\rho_0$  sur  $p_0$ ) amène le point  $p'$  au contact avec la surface  $S$ . Cette définition ne prend pas en compte de possibles obstructions avec des surfaces autres que la surface  $S$  visée sur le chemin déterminé par chaque vitesse  $\vec{v}'$ .

L’“atteignabilité” de  $S$  depuis  $p_0$  est donc définie par l’ensemble de vitesses  $\{\vec{v}\}$  dans le cercle unitaire permettant d’atteindre  $S$  malgré d’une part l’incertitude de position<sup>4</sup> initiale  $\rho_0$  et d’autre part l’incertitude de contrôle  $\eta$ . Cet ensemble noté  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$  définit un secteur circulaire<sup>5</sup> dont l’angle dépend de la longueur de  $S$  du rayon d’incertitude  $\rho_0$  et de l’angle  $\eta$  et de la distance entre  $S$  et  $p_0$ . Une vitesse  $\vec{v}$  dans  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$  est définie comme  $\vec{v} = (v_x, v_y)$  où  $(v_x, v_y)$  spécifie la direction de translation et  $\|(v_x, v_y)\| = 1$ .

Par définition  $p_0$  et chaque vitesse  $\vec{v}$  dans  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$  satisfont le critère “visible( $S, p_0, \vec{v}$ )”.

#### Définition 2.10 - Vitesse sûre

Une vitesse  $\vec{v}$  dans  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$  est dite *vitesse sûre*.

#### Définition 2.11 - Région sûre

La projection du secteur circulaire  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$  placé à la position  $p_0$  sur la surface  $S$  définit la *région sûre*  $R$  en  $S$ . Les positions dans  $R$  peuvent servir d’attracteurs pour attirer le robot depuis  $p_0$  vers un contact “robuste” avec  $S$ .

Un modèle géométrique appelé “*projection en avant*” est proposé pour calculer l’ensemble  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$ . Le modèle proposé s’inspire du modèle “*forward projection*” proposé par Erdmann [Erd86].

#### 2.4.2.1 Calcul de l’“atteignabilité” dans l’espace $\mathbb{R}^2$

Dans l’espace  $\mathbb{R}^2$  l’algorithme de construction du modèle “projection en avant” pour un robot ponctuel est (figure 2.11) :

4. Dans l’espace  $\mathbb{R}^2$ , l’incertitude d’orientation est considérée constante.

5. Pour une surface  $S$  qui n’est pas “atteignable”,  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta) = \emptyset$ .

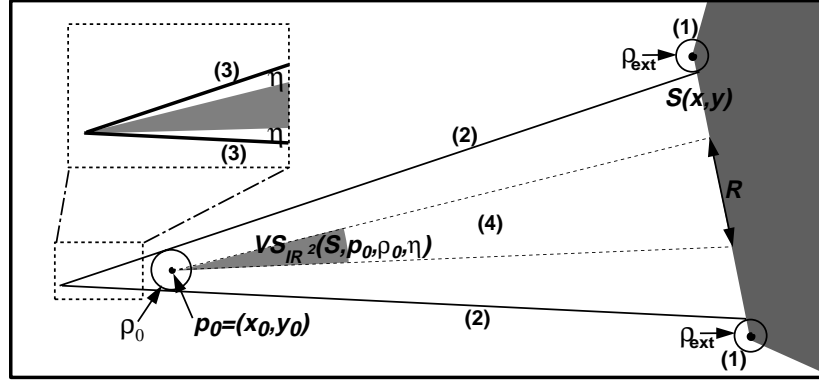


FIG. 2.11 – Calcul de l’“atteignabilité” dans l’espace des configurations  $\mathbb{R}^2$ .

1. Réduire les extrémités de  $S$  d’une distance  $\rho_{ext}$  afin d’éviter des “contacts” avec celles-ci. La distance  $\rho_{ext}$  peut être une valeur constante définie en fonction de la taille du robot réel. Si la longueur de la surface réduite est inférieure ou égale à  $2 * \rho_0$  (le diamètre du disque d’incertitude de position en  $p_0$ )  $\Gamma S$  ne peut pas être atteinte sous la contrainte d’incertitude de position. Dans ce cas  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta) = \emptyset$ .
2. Définir deux segments de droite partant des nouvelles extrémités de  $S$  et tangents au disque d’incertitude centré en  $p_0$  jusqu’à leur intersection. Les segments tangents choisis sont ceux qui définissent un cône contenant le disque d’incertitude centré en  $p_0$ . De plus  $\Gamma$  ce cône contient l’ensemble des vitesses  $\{\vec{v}\}$  qui permettent d’amener le robot sur la surface  $S$  en l’absence d’incertitude de contrôle (i.e.  $\eta = 0$ ). Si l’angle du cône est inférieur ou égal à  $2 * \eta \Gamma S$  ne peut pas être atteinte sous la contrainte d’incertitude de contrôle  $\Gamma$  dans ce cas  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta) = \emptyset$ .
3. Réduire le cône de vitesses  $\{\vec{v}\}$  de chaque côté d’un angle égal à  $\eta$ . Le cône intérieur obtenu contient l’ensemble des vitesses  $\{\vec{v}\}$  qui garantissent au robot un contact avec  $S$  sous l’incertitude de position et de contrôle. Ce nouveau cône contient l’ensemble de vitesses  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$ .
4. Définir la *région sûre*  $R$  de  $S$  qui peut être atteinte sous les contraintes d’incertitude de position et de contrôle en plaçant le sommet du secteur circulaire  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$  à la position  $p_0$  et en projetant ce dernier sur la surface  $S$ .

Par construction  $\Gamma$  le suivi d’une vitesse  $\vec{v}$  dans  $VS_{\mathbb{R}^2}(S, p_0, \rho_0, \eta)$  garantit au robot d’atteindre la surface  $S$  visée depuis  $p_0$  malgré l’augmentation de son incertitude de position  $\rho_0$  le long du déplacement  $\Gamma$  et en absence d’autres obstructions.

### 2.4.3 “Atteignabilité” d’une surface dans $\mathbb{R}^2 \times [0, 2\pi]$

Dans ce cas  $\Gamma$  l’“atteignabilité” d’une surface tient aussi compte de la contrainte d’incertitude d’orientation.



Soient

- $S \subset \mathbb{R}^2 \times [0, 2\pi]$  la surface bornée à atteindre<sup>6</sup>;
- $q_0 = (x_0, y_0, \theta_0) \in S^+$  la configuration de départ du robot avec un cylindre d'incertitude associé dont les paramètres sont  $\rho_0$  et  $\psi_0$ ;
- $p_0 = (x_0, y_0)$  la position de départ du robot;
- $2 * \eta$  l'angle du cône d'incertitude de contrôle;
- $k_r$  la constante de proportionnalité pour les imprécisions en rotation du robot;
- $\theta_f \in [0, 2\pi]$  l'orientation souhaitée du robot au moment du contact avec la surface  $S$ ;
- $\delta\theta = \theta_f - \theta_0$ ;
- $\psi_f = \psi_0 + |\delta\theta| * k_r$  l'incertitude d'orientation associée à  $\theta_f$  (équation 2.5). L'intervalle d'incertitude d'orientation centré sur  $\theta_f$  le long de l'axe  $\Theta$  de l'espace  $\mathbb{R}^2 \times [0, 2\pi]$  est  $[\theta_f - \psi_f, \theta_f + \psi_f]$ .

Dans l'espace  $\mathbb{R}^2 \times [0, 2\pi]$  le modèle “*projection en avant*” calcule l'ensemble des vitesses  $\{\vec{v}\}$  permettant au robot d'atteindre la surface  $S$  avec l'orientation  $\theta_f$  (à  $\psi_f$ -près) depuis la configuration  $q_0$  malgré l'évolution des incertitudes initiales  $\rho_0$  et  $\psi_0$  causée par l'incertitude de contrôle  $\eta$  et l'incertitude d'orientation  $k_r$ . Cet ensemble noté  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  définit un secteur circulaire<sup>7</sup>. Les nouveaux paramètres  $k_r$  et  $\theta_f$  apparaissent en raison de la prise en compte de l'incertitude d'orientation. Une vitesse  $\vec{v}$  dans  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  est définie comme  $\vec{v} = (v_x, v_y, v_\theta)$  où  $(v_x, v_y)$  spécifie la direction de translation avec  $\|(v_x, v_y)\| = 1$  et  $v_\theta$  spécifie l'incrément en rotation par unité de translation. Le déplacement en translation et celui en rotation déterminés par une vitesse  $\vec{v}$  sont indépendants et effectués simultanément. Le modèle “*projection en avant*” d'un cylindre d'incertitude dans  $\mathbb{R}^2 \times [0, 2\pi]$  est construit à l'aide du modèle “*projection en avant*” d'un disque d'incertitude dans  $\mathbb{R}^2$ .

### 2.4.3.1 Calcul de l’“atteignabilité” dans l’espace $\mathbb{R}^2 \times [0, 2\pi]$

Dans l'espace  $\mathbb{R}^2 \times [0, 2\pi]$  l'algorithme de construction du modèle “*projection en avant*” pour un robot ponctuel est :

1. Construire le cylindre d'incertitude associé à  $q_i = (x_0, y_0, \theta_f)$ . En ce qui concerne l'incertitude de position celle-ci n'a pas changé donc la base du cylindre garde le

6. Dans  $\mathbb{R}^2 \times [0, 2\pi]$ , la construction de la surface  $S$  d'un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  définit un polygone  $\mathcal{CB}_{i,\theta}$  sur chaque plan  $\Theta = \theta$ .

7. Pour une surface  $S$  qui n'est pas “atteignable”,  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f) = \emptyset$ .

rayon  $\rho_0$ . Par contre la hauteur du cylindre devient l'intervalle  $[\theta_f - \psi_f, \theta_f + \psi_f]$  (figure 2.12).

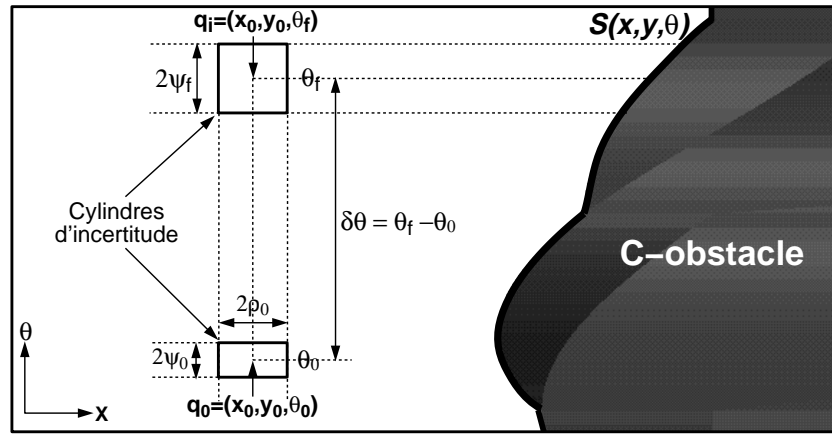


FIG. 2.12 – Construction du cylindre d’incertitude correspondant à la configuration  $q_i = (x_0, y_0, \theta_f)$  : la base est le disque d’incertitude de position de rayon  $\rho_0$  et la hauteur est l’intervalle d’incertitude d’orientation d’amplitude  $2 * \psi_f$  (vue sur le plan  $X\Theta$ ).

2. Pour chaque orientation  $\theta' \in [\theta_f - \psi_f, \theta_f + \psi_f]$  :

- déterminer l’ensemble de positions  $S_{\theta'} \subset \mathbb{R}^2$  de  $S$  dont la coordonnée  $\theta$  est égale à  $\theta'$ . Selon l’hypothèse sur la géométrie de  $S$  la frontière de  $S_{\theta'}$  (notée  $\partial S_{\theta'}$ ) définit une ligne polygonale.
- calculer l’“atteignabilité” de  $S_{\theta'}$  depuis  $p_0$  malgré l’augmentation de l’incertitude de position  $\rho_0$  causée par l’incertitude de contrôle  $\eta$ . Cela revient à calculer le secteur circulaire de vitesses  $VS_{\mathbb{R}^2}(S_{\theta'}, p_0, \rho_0, \eta)$  (figure 2.13).

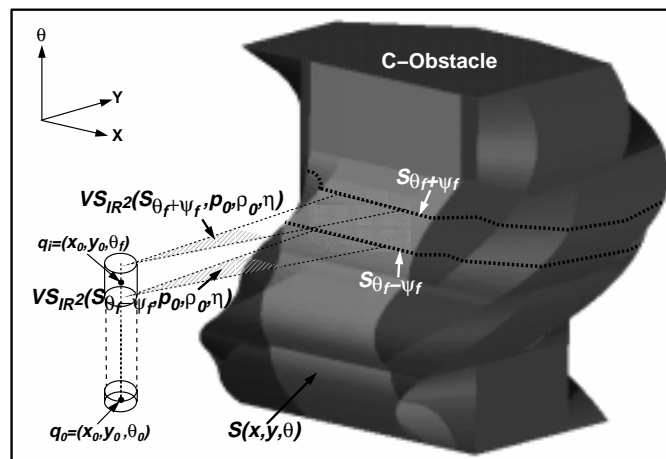


FIG. 2.13 – Calcul de l’“atteignabilité” des segments  $S_{\theta_f + \psi_f}$  et  $S_{\theta_f - \psi_f}$  (vue dans  $XY\Theta$ ).

3. Calculer la composante de translation  $(v_x, v_y)$  des vitesses  $\vec{v}$  dans le secteur circulaire  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  comme l'intersection des secteurs circulaires  $VS_{\mathbb{R}^2}(S_{\theta'}, p_0, \rho_0, \eta)$  (figure 2.14) :

$$VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)_{v_x, v_y} = \bigcap_{\theta' = \theta_f - \psi_f}^{\theta_f + \psi_f} VS_{\mathbb{R}^2}(S_{\theta'}, p_0, \rho_0, \eta) \quad (2.10)$$

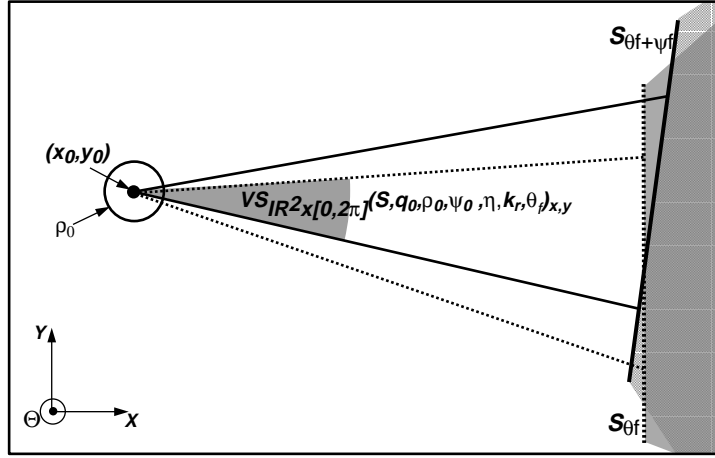


FIG. 2.14 – Vitesses communes en translation atteignant les segments  $S_{\theta_f}$  et  $S_{\theta_f + \psi_f}$  et tenant compte de l'augmentation de l'incertitude de position  $\rho_0$  due à l'incertitude de contrôle  $\eta$  (vue sur le plan  $XY$ ).

4. Définir la *région sûre*  $R$  de  $S$  qui peut être atteinte sous les contraintes d'incertitude de position  $\Gamma$  d'orientation et de contrôle en plaçant le sommet du secteur circulaire  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  à la configuration  $q_i$  et en projetant ce dernier sur la surface  $S$  (figure 2.15). Dans cette étape  $\Gamma$  la composante  $v_\theta$  des vitesses  $\vec{v}$  dans le secteur circulaire projeté est considérée comme nulle. A chaque vitesse  $\vec{v}$  dans le secteur circulaire  $\Gamma$  on peut lui associer la configuration  $q'$   $\Gamma$  sur la région  $R$   $\Gamma$  étant l'image de la configuration  $q_i$  suivant  $\vec{v}$ .
5. Calculer la valeur normalisée de la composante  $v_\theta$  pour chaque vitesse  $\vec{v}$  dans le secteur circulaire  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$ .  $v_\theta$  est définie comme  $\delta\theta/\delta p$  où  $\delta p$  est la distance euclidienne entre les positions de  $q_0$  et de  $q'$  ( $q'$  la configuration associée à la vitesse précédente  $\vec{v}$ )  $\Gamma$  et  $\delta\theta$  est la distance angulaire entre les orientations de  $q_0$  et de  $q'$  (par construction  $\Gamma q'_\theta = \theta_f$ ).

Par construction  $\Gamma$  le suivi d'une vitesse  $\vec{v}$  dans  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  garantit au robot d'atteindre la surface  $S$  visée avec l'orientation  $\theta_f$  (à  $\psi_f$ -près) depuis  $q_0$  malgré l'augmentation de ses incertitudes de position  $\rho_0$  et d'orientation  $\psi_0$  le long du déplacement  $\Gamma$  et en absence d'autres obstructions.

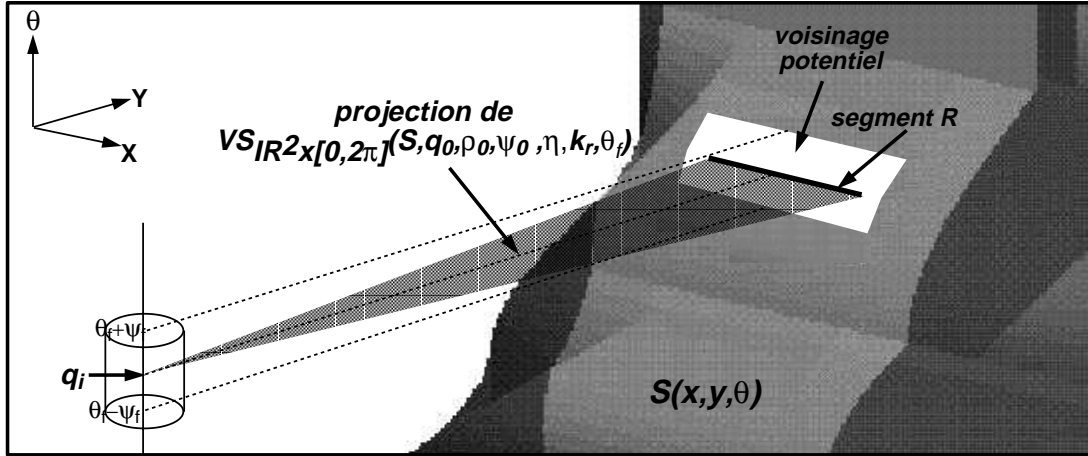


FIG. 2.15 – Le segment  $R$  de  $S$  résulte de la projection du secteur circulaire  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  sur la surface  $S$ . Les configurations potentielles de contact pour l'ensemble de vitesses dans le secteur circulaire définissent un voisinage autour de la région  $R$  à l'intérieur de  $S$  (vue dans l'espace  $XY\Theta$ ).

La construction du secteur circulaire  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  est complexe et sa détermination n'est possible que si une représentation continue de la surface  $S$  est disponible. Par souci de simplicité la définition d'un tel secteur circulaire est simplifiée en ne considérant qu'un nombre discret d'orientations (e.g.  $\theta_f - \psi_f, \theta_f, \theta_f + \psi_f$ ) lors du calcul des secteurs circulaires de vitesses sûres dans  $\mathbb{R}^2$  (simplification des étapes 2 et 3 de l'algorithme). La région sûre  $R$  résultante sera approchée en projetant le secteur circulaire  $VS_{\mathbb{R}^2 \times [0, 2\pi]}(S, q_0, \rho_0, \psi_0, \eta, k_r, \theta_f)$  obtenu sur la surface  $S$ .

## 2.5 Conclusion

Pour des tâches robotiques où les incertitudes liées au robot évoluent de façon importante et en permanence l'exécution des plans nominaux corrects générés par un planificateur automatique pourrait échouer. Dans ce cas il est nécessaire d'intégrer lors de la phase de planification une analyse sur l'évaluation et l'évolution des incertitudes afin d'engendrer des *plans robustes* qui aboutissent en succès malgré les effets des incertitudes. Nous avons présenté dans ce chapitre des modèles explicites et de nature ensembliste qui concernent les incertitudes de position d'orientation et de contrôle affectant le robot. Les modèles de base existaient déjà et ont été utilisés dans plusieurs approches de planification proposées auparavant. Notre contribution au niveau de la modélisation des incertitudes peut être résumée en :

- la représentation des incertitudes de position et d'orientation dans l'espace des configurations  $\mathbb{R}^2 \times [0, 2\pi]$  liées au robot dans un modèle cylindrique. Cette représentation rend explicite la distinction entre les mesures de position et celles d'orientation.

- la définition d’une méthode de calcul appelée “*projection en avant*” dont le principe se trouve dans le modèle “forward projection” d’Erdmann [Erd86]. Cette méthode détermine des *vitesses sûres* permettant au robot de s’approcher de la surface des obstacles  $\Gamma$  en tenant compte de l’évolution des incertitudes  $\Gamma$  afin de se re-localiser par rapport à celles-ci  $\Gamma$  et donc de réduire ses incertitudes. La méthode a été explicitée pour un robot ponctuel évoluant dans les espaces  $\mathbb{R}^2$  et  $\mathbb{R}^2 \times [0, 2\pi]$ .
- l’intégration de l’ensemble des modèles présentés dans une nouvelle approche de planification automatique de tâches robotiques. Cette intégration est détaillée dans le chapitre 5. Les validations de la mise en œuvre des modèles sont illustrées dans le chapitre 6. La nature ensembliste de ces modèles détermine le type d’approche proposée par rapport aux travaux existants dans le même domaine (chapitre 1).

Dans la suite  $\Gamma$  nous posons formellement le problème de planification de mouvement traité dans ce mémoire.

# Chapitre 3

## Formalisation du problème traité

---

Comme nous l'avons énoncé précédemment il est nécessaire dans certaines tâches robotiques de prendre en compte les incertitudes pour garantir l'exécution correcte des solutions nominales générées en planification. De ce fait nous nous sommes intéressés au problème de la planification de mouvement d'un robot sous des contraintes d'incertitudes géométriques.

Comme nous avons pu le constater dans l'état de l'art peu de travaux évaluent explicitement les incertitudes pour engendrer des chemins (ou des plans) robustes. Ceci est dû à une complexité trop importante liée d'une part aux problèmes de planification (complexité d'ordre exponentiel) et d'autre part à la nouvelle dimension apportée par l'incertitude. On trouve cependant des travaux intéressants basés sur des hypothèses simplificatrices quant à la géométrie le nombre de degrés de liberté et la cinématique du robot; à la géométrie et le nombre d'obstacles dans l'environnement; aux contraintes d'incertitudes prises en compte; etc.

Ce chapitre présente le problème de planification robotique abordé. D'abord les définitions et notations de base nécessaires pour la formulation du problème sont données. Ensuite le problème robotique traité est posé formellement. Finalement des aspects supplémentaires liés à la faisabilité des solutions sont abordés. Les principes de base de l'approche proposée pour résoudre le problème posé sont donnés dans le chapitre 4 une description de sa mise en œuvre est détaillée dans le chapitre 5 et son application est illustrée dans le chapitre 6.

### 3.1 Définitions et notations de base

La définition formelle du problème traité est basée sur un ensemble de concepts propres au domaine de la robotique. La définition de ces concepts ainsi que la notation pour les référencer est présentée ci-dessous.

Soit  $\mathcal{S}$  une région dans un espace topologique  $\mathcal{X}$  nous référençons la frontière de  $\mathcal{S}$  comme  $\partial\mathcal{S}$  l'intérieur de  $\mathcal{S}$  comme  $int(\mathcal{S})$  et le complément de  $\mathcal{S}$  par rapport à  $\mathcal{X}$  comme  $\mathcal{X} \setminus \mathcal{S}$ .

En ce qui concerne les définitions de l'*espace de travail* (noté  $\mathcal{W}$ ) du *robot* (noté  $\mathcal{A}$ ) d'un *obstacle* (noté  $\mathcal{B}_i$ ) d'une *configuration* (notée  $q$ ) de l'*espace des configurations* (noté  $\mathcal{C}$ ) d'un *C-obstacle* (noté  $\mathcal{CB}_i$ ) le lecteur pourra se référer à §2.2. Nous rappelons les hypothèses suivantes :

- l'*espace de travail*  $\mathcal{W}$  est l'espace plan  $\mathbb{R}^2$ ;
- le *robot*  $\mathcal{A}$  est de type *holonome* caractérisé par une géométrie polygonale convexe et par trois d.d.l. (deux d.d.l. en translation et un d.d.l. en rotation);
- les obstacles  $\{\mathcal{B}_i\}$  possèdent une géométrie polygonale (convexe ou concave);
- l'espace des configurations  $\mathcal{C}$  est représenté par l'espace  $\mathbb{R}^2 \times [0, 2\pi]$ .

### Définition 3.1 - Chemin dans l'espace des configurations

Un *chemin* entre deux configurations  $q_0$  et  $q_f$  de  $\mathcal{C}$  est une transformation continue  $\tau : [0, 1] \rightarrow \mathcal{C}$  avec  $\tau(0) = q_0$  et  $\tau(1) = q_f$ .

### Définition 3.2 - Espace de contact et chemin en contact

L'*espace de contact*  $\mathcal{C}_{contact}$  est le sous-ensemble des configurations de  $\mathcal{A}$  qui touchent un(ou plusieurs) obstacle(s)  $\mathcal{B}_i$  dans  $\mathcal{W}$  sans le(s) pénétrer :

$$\mathcal{C}_{contact} = \{q \in \mathcal{C} \mid \mathcal{A}(q) \cap \bigcup_{i=1}^n \mathcal{B}_i \neq \emptyset \text{ et } int(\mathcal{A}(q)) \cap \bigcup_{i=1}^n int(\mathcal{B}_i) = \emptyset\} \quad (3.1)$$

Un *chemin en contact*  $\tau$  est un chemin dont l'image de la transformation  $\tau$  appartient à  $\mathcal{C}_{contact}$ .

### Définition 3.3 - Espace libre, chemin libre et chemin quasi-libre

L'*espace libre*  $\mathcal{C}_{libre}$  est le sous-ensemble des configurations de  $\mathcal{A}$  qui ne sont en contact avec aucun obstacle  $\mathcal{B}_i$  dans  $\mathcal{W}$  :

$$\mathcal{C}_{libre} = \mathcal{C} \setminus \bigcup_{i=1}^n \mathcal{CB}_i \quad (3.2)$$

où  $n$  est le nombre d'obstacles dans  $\mathcal{W}$ .

Un *chemin libre*  $\tau$  est un chemin dont l'image de la transformation  $\tau$  appartient à  $\mathcal{C}_{libre}$ .

Un *chemin quasi-libre*  $\tau$  est un chemin libre dont au moins une extrémité (i.e.  $\tau(0)$  ou  $\tau(1)$ ) appartient à  $\mathcal{C}_{contact}$ .

**Définition 3.4 - Espace valide et chemin valide**

L'espace valide  $\mathcal{C}_{valide}$  est le sous-ensemble des configurations de  $\mathcal{A}$  qui ne pénètrent pas les obstacles  $\mathcal{B}_i$  dans  $\mathcal{W}$  :

$$\mathcal{C}_{valide} = \mathcal{C}_{libre} \cup \mathcal{C}_{contact} \quad (3.3)$$

Un *chemin valide*  $\tau$  est un chemin dont l'image de la transformation  $\tau$  appartient à  $\mathcal{C}_{valide}$ .

**Définition 3.5 - Les incertitudes géométriques**

(voir chapitre 2).

**Définition 3.6 - Chemin robuste**

Un *chemin robuste*  $\tau$  est un chemin dont la configuration but  $q_f$  est atteinte malgré les effets de l'incertitude tout au long de celui-ci. Dans notre cas la robustesse est considérée vis-à-vis des incertitudes de position d'orientation et de contrôle sur le robot  $\mathcal{A}$  dont l'évaluation est faite en appliquant les modèles d'incertitude présentés dans le chapitre 2.

**Définition 3.7 - Plan**

Un *plan*  $\Sigma$  est une séquence finie d'actions que le robot  $\mathcal{A}$  peut interpréter puis exécuter.

## 3.2 Définition du problème de planification traité

Soient :

- $\{\mathcal{B}_1, \dots, \mathcal{B}_n\}$  un ensemble d'obstacles dont on connaît la géométrie (sans incertitude) et la position/orientation (sans incertitude) dans  $\mathcal{W}$ ;
- $\mathcal{A}$  un robot dont on connaît la géométrie (sans incertitude) dans  $\mathcal{W}$ ;
- $(q_0, \mathcal{R}_0)$  le couple définissant respectivement la configuration nominale initiale du robot  $\mathcal{A}$  et son voisinage d'incertitude.  $q_0$  et  $\mathcal{R}_0$  doivent être définis dans le même sous-espace de configurations  $\mathcal{C}_{libre}$  ou  $\mathcal{C}_{contact}$ . De plus la géométrie du voisinage  $\mathcal{R}_0$  est modélisée à l'aide des modèles d'incertitude de position et d'orientation présentés dans le chapitre 2;
- $(q_f, \mathcal{R}_f)$  le couple définissant respectivement la configuration but du robot  $\mathcal{A}$  à atteindre et son voisinage d'incertitude.  $q_f$  et  $\mathcal{R}_f$  doivent être définis dans le même sous-espace de configurations  $\mathcal{C}_{libre}$  ou  $\mathcal{C}_{contact}$ . La contrainte sur la géométrie de  $\mathcal{R}_0$  est aussi valable pour la géométrie de  $\mathcal{R}_f$ ;
- $\eta$  l'angle de déviation maximal sur les déplacements en translation du robot  $\mathcal{A}$ ;



- $k_t$  la constante de proportionnalité sur les imprécisions des amplitudes des déplacements en translation<sup>1</sup>;
- $k_r$  la constante de proportionnalité sur les imprécisions en rotation du robot  $\mathcal{A}$ .

LE PROBLÈME POSÉ consiste à trouver un chemin  $\tau$  valide et robuste pour le robot ponctuel  $q_{\mathcal{A}}$  partant de  $q_0$  avec une incertitude  $\mathcal{R}_0$  et devant aller en  $q_f$  avec une incertitude inférieure ou égale à celle spécifiée par  $\mathcal{R}_f$ . Ainsi le problème de planification des mouvements du robot bi-dimensionnel  $\mathcal{A}$  dans  $\mathbb{R}^2$  est transformé en un problème équivalent pour le robot ponctuel  $q_{\mathcal{A}}$  dans l'espace  $\mathbb{R}^2 \times [0, 2\pi]$ .

### 3.2.1 Génération de plans

Le but final de la résolution d'un problème de planification consiste à engendrer un plan pour le robot  $\mathcal{A}$ . Afin de pouvoir engendrer un tel plan nous allons ajouter des contraintes supplémentaires au problème posé. Dans la suite les concepts liés aux nouvelles contraintes sont définis avant d'exprimer ces dernières.

#### Définition 3.8 - Capteur de contact "virtuel"

Un *capteur de contact "virtuel"*  $CV_{\mathcal{A}}$  correspond à une région continue sur la surface physique du robot  $\mathcal{A}$  où une mesure de "contact" est demandée. La région est définie soit comme un sommet soit comme une face de la surface du robot. La spécification d'un capteur de contact "virtuel" est de la forme  $CV_{\mathcal{A}}(Id)$  avec  $Id$  l'identificateur correspondant de la région sur la géométrie du robot  $\mathcal{A}$ .

#### Définition 3.9 - Mesure de "contact"

Une *mesure de "contact"*  $M_{contact}$  est une grandeur physique pouvant être acquise par un capteur de contact "virtuel". La mesure de "contact" est fonction de la tâche robotique à accomplir : (1) pour une tâche d'assemblage la mesure de "contact" correspond à une force physique  $\vec{f}_{réaction}$  exercée par les obstacles sur la pièce manipulée et (2) pour une tâche de navigation d'un robot mobile la mesure de "contact" correspond à une distance  $d_{contact}$  entre la surface du robot et la surface des obstacles.

#### Définition 3.10 - Commande de contrôle

Une *commande de contrôle* est une action qui peut être commandée au robot  $\mathcal{A}$  et spécifiée par un quadruplet  $\sigma_i = (Id_i, Cl_i, PM_i, CA_i)$  où  $Id_i$  est l'identificateur de la commande,  $Cl_i$  est la condition initiale à valider avant de lancer la commande,  $PM_i$  est l'ensemble des paramètres définissant le mouvement en translation et/ou en rotation à effectuer et  $CA_i$  est la condition à valider pour arrêter la commande. Une condition initiale  $Cl_i$  est définie sur les conditions de "contact" du robot  $\mathcal{A}$  à sa configuration courante  $q^*$ . Ces dernières sont évaluées à partir de mesures de "contact" acquises par des capteurs de contact virtuels. Pour sa part une condition d'arrêt  $CA_i$  peut être

1. Selon une hypothèse faite dans §2.3.2, nous supposons  $k_t = \tan(\eta)$ .

définie sur les conditions de “contact” du robot  $\mathcal{A}$  mais également sur le déplacement relatif en translation/rotation effectué par le robot  $\mathcal{A}$  depuis le début de la commande. Un mouvement en translation/rotation doit s’exécuter simultanément. La sémantique de  $\sigma_i = (\text{Id}_i, \text{Cl}_i, \text{PM}_i, \text{CA}_i)$  est :

COMMANDE  $\text{Id}_i(\text{Cl}_i; \text{PM}_i; \text{CA}_i)$

SI la condition  $\text{Cl}_i$  est vérifiée à la configuration courante  $q^*$  du robot

ALORS suivre le mouvement spécifié par les paramètres  $\text{PM}_i$

JUSQU’À que la condition  $\text{CA}_i$  soit vérifiée pour la configuration courante  $q^*$  du robot.

Nous avons spécifié un ensemble générique de commandes de contrôle que le robot  $\mathcal{A}$  peut exécuter (section 3.4.1).

### Définition 3.11 - Plan robuste

Un *plan robuste*  $\Sigma$  est une séquence de commandes de contrôle  $\{\sigma_1, \dots, \sigma_m\}$  avec  $m \geq 1$  permettant de mener le robot  $\mathcal{A}$  d’une configuration potentielle  $q'_0$  (celle-ci à l’intérieur du voisinage d’incertitude  $\mathcal{R}_0$  associé à la configuration initiale  $q_0$ ) à une configuration potentielle  $q'_f$  (celle-ci à l’intérieur du voisinage d’incertitude  $\mathcal{R}_f$  associé à la configuration but  $q_f$ ) malgré les effets des incertitudes.

### Contraintes supplémentaires au problème posé

Les contraintes pour garantir une transformation d’un chemin  $\tau$  valide et robuste en un plan robuste  $\Sigma$  sont :

1. tout chemin  $\tau$  planifié doit être défini comme une suite de sous-chemins  $\tau = \{\tau_1, \dots, \tau_m\}$  satisfaisant :

$$\tau_1(0) = q_0, \quad (3.4)$$

$$\tau_m(1) = q_f, \quad (3.5)$$

$$\tau_i(1) = \tau_{i+1}(0), \quad i \in [1, \dots, m-1] \quad (3.6)$$

Les conditions (3.4) et (3.5) garantissent que le problème posé est résolu et la condition (3.6) garantit la continuité entre les sous-chemins consécutifs;

2. chaque sous-chemin  $\tau_i$  ne peut être que de type : *libre en contact* ou *quasi-libre* (choix exclusif);
3. chaque sous-chemin  $\tau_i$  doit être robuste;
4. le parcours de chaque sous-chemin  $\tau_i$  doit pouvoir être accompli par une commande de contrôle  $\sigma_i$  du robot.

Puisqu'une commande primitive de contrôle a été spécifiée comme un quadruplet  $\sigma_i = (\text{Id}_i, \text{Cl}_i, \text{PM}_i, \text{CA}_i)$  cette contrainte impose que :

- (a) la condition initiale  $\text{Cl}_i$  soit vérifiée à la configuration initiale  $\tau_i(0)$ .
- (b) le mouvement déterminé par les paramètres  $\text{PM}_i$  suive le chemin défini par  $\tau_i$ .
- (c) la condition d'arrêt  $\text{CA}_i$  ne soit vérifiée qu'à la configuration finale  $\tau_i(1)$ .

Un plan robuste  $\Sigma$  sera ainsi composé d'une suite de commandes de contrôle  $\{\sigma_1, \dots, \sigma_m\}$  dont chaque commande  $\sigma_i$  est chargée de garantir le déplacement du robot  $\mathcal{A}$  le long du sous-chemin  $\tau_i$ . L'exécution correcte du plan  $\Sigma$  est garantie à condition que les incertitudes réelles suivent l'évolution présumée dans les sous-chemins  $\tau_i$  respectifs.

### 3.3 Le sous-espace $\mathcal{C}_{\text{contact}}$ de l'espace d'exploration

La planification de chemins valides implique la possibilité d'explorer l'espace libre  $\mathcal{C}_{\text{libre}}$  mais aussi l'espace de contact  $\mathcal{C}_{\text{contact}}$ . L'exploration de  $\mathcal{C}_{\text{contact}}$  permet d'une part de réduire l'incertitude de position et d'orientation du robot pendant son déplacement et d'autre part de considérer des chemins écartés par les approches d'évitement d'obstacles. L'incertitude de position et/ou d'orientation du robot après avoir effectué un mouvement atteignant un contact ou un mouvement contraint à garder un contact sera plus faible que celle engendrée par l'exécution d'un mouvement libre de contacts. Cependant l'exploration de  $\mathcal{C}_{\text{contact}}$  n'est possible que si le robot est équipé d'un système de contrôle fiable pour détecter et gérer les contacts.

En ce qui concerne l'exploration de  $\mathcal{C}_{\text{libre}}$  celle-ci reste nécessaire pour permettre au robot le passage entre des composantes non connexes de  $\mathcal{C}_{\text{contact}}$  et pour engendrer des mouvements moins contraints en translation et en rotation.

L'espace  $\mathcal{C}_{\text{contact}}$  peut devenir très complexe même dans l'espace des configurations engendré par un robot  $\mathcal{A}$  polygonal entouré d'obstacles polygonaux. La classe de contacts possibles peut être définie à partir de deux sous-classes : les contacts simples et les contacts multiples. Nous ne considérons qu'un sous-ensemble de contacts de chaque sous-classe. Dans la suite nous définissons les deux sous-classes de contacts et précisons les sous-ensembles à retenir dans l'exploration de  $\mathcal{C}_{\text{contact}}$ .

#### 3.3.1 Les contacts simples

Un *contact simple* est défini comme une relation de "contact" entre un élément géométrique<sup>2</sup> du robot  $\mathcal{A}$  et un élément géométrique d'un obstacle  $\mathcal{B}_i$ . Les contacts simples possibles entre  $\mathcal{A}$  et  $\mathcal{B}_i$  seront donc du type :

---

2. Un élément géométrique correspond soit à un sommet, soit à une arête.

**Contact arête $_{\mathcal{A}}$ -sommet $_{\mathcal{B}_i}$**  : le contact est défini entre une arête  $\vec{E}_j^{\mathcal{A}}$  de  $\mathcal{A}$  et un sommet  $s_k^{\mathcal{B}_i}$  de  $\mathcal{B}_i$  (figure 3.1a). Ce type de contact satisfait :

$$\begin{aligned} \exists \lambda \in ]0, 1[ \mid s_k^{\mathcal{B}_i} &= s_j^{\mathcal{A}} + \lambda \vec{E}_j^{\mathcal{A}}, \\ \vec{n}_j^{\mathcal{A}} \cdot -\vec{E}_{k-1}^{\mathcal{B}_i} &> 0, \\ \vec{n}_j^{\mathcal{A}} \cdot \vec{E}_k^{\mathcal{B}_i} &> 0 \end{aligned}$$

**Contact sommet $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$**  : le contact est défini entre un sommet  $s_j^{\mathcal{A}}$  de  $\mathcal{A}$  et une arête  $\vec{E}_k^{\mathcal{B}_i}$  de  $\mathcal{B}_i$  (figure 3.1b). Ce type de contact satisfait :

$$\begin{aligned} \exists \lambda \in ]0, 1[ \mid s_j^{\mathcal{A}} &= s_k^{\mathcal{B}_i} + \lambda \vec{E}_k^{\mathcal{B}_i}, \\ \vec{n}_k^{\mathcal{B}_i} \cdot -\vec{E}_{j-1}^{\mathcal{A}} &> 0, \\ \vec{n}_k^{\mathcal{B}_i} \cdot \vec{E}_j^{\mathcal{A}} &> 0 \end{aligned}$$

**Contact arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$**  : le contact est défini entre une arête  $\vec{E}_j^{\mathcal{A}}$  de  $\mathcal{A}$  et une arête  $\vec{E}_k^{\mathcal{B}_i}$  de  $\mathcal{B}_i$  (figure 3.1c). Ce type de contact satisfait :

$$\begin{aligned} \exists \lambda_1, \lambda_2 \in ]0, 1[, \lambda_1 \neq \lambda_2 \mid \{ \forall \lambda \in [\lambda_1, \lambda_2], \exists \lambda' \in ]0, 1[ \mid s_j^{\mathcal{A}} + \lambda \vec{E}_j^{\mathcal{A}} &= s_k^{\mathcal{B}_i} + \lambda' \vec{E}_k^{\mathcal{B}_i} \}, \\ \vec{n}_j^{\mathcal{A}} \cdot \vec{n}_k^{\mathcal{B}_i} &= -1 \end{aligned}$$

**Contact sommet $_{\mathcal{A}}$ -sommet $_{\mathcal{B}_i}$**  : le contact est défini entre un sommet  $s_j^{\mathcal{A}}$  de  $\mathcal{A}$  et un sommet  $s_k^{\mathcal{B}_i}$  de  $\mathcal{B}_i$ . Ce type de contact satisfait :

$$s_j^{\mathcal{A}} = s_k^{\mathcal{B}_i}$$

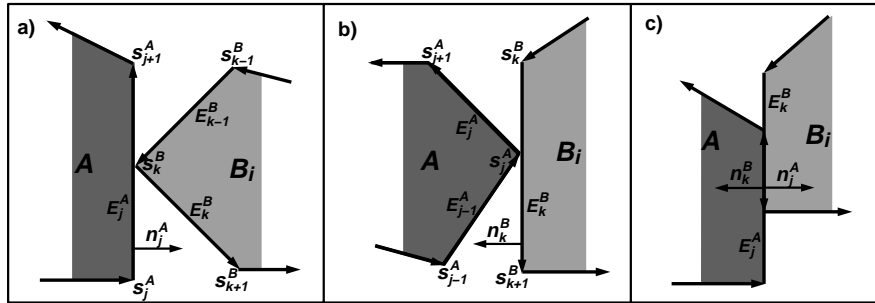


FIG. 3.1 – Définition des contacts simples entre une pièce manipulée  $\mathcal{A}$  et un obstacle  $\mathcal{B}_i$  : a) contact simple de type arête $_{\mathcal{A}}$ -sommet $_{\mathcal{B}_i}$ ; b) contact simple de type sommet $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$ ; c) contact simple de type arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$ .

De la sous-classe des *contacts simples* le type de contact sommet $_{\mathcal{A}}$ -sommet $_{\mathcal{B}_i}$  est particulier car d'une part son "atteignabilité" réelle ne peut être assurée sans guidage mais d'autre part le robot peut recaler sa position par rapport à la position connue d'un sommet d'un obstacle. Du point de vue de l'incertitude cette dernière

caractéristique rend très attractive la prise en compte de ce type de contact dans la planification. En raison de cela ce sous-ensemble de contacts sera pris en compte dans notre approche de planification. En ce qui concerne les autres types de contact *simples* leur “atteignabilité” réelle demande moins d’efforts sur le guidage du robot tout en apportant également une réduction de l’incertitude du robot. En résumé l’ensemble de contacts simples est donc considéré dans le processus de planification.

### 3.3.2 Les contacts multiples

Un *contact multiple* est défini comme une interaction multiple entre le robot  $\mathcal{A}$  et les obstacles. Une telle interaction peut avoir lieu entre :

1. un élément géométrique de  $\mathcal{A}$  et deux ou plusieurs éléments géométriques des obstacles;
2. deux ou plusieurs éléments géométriques de  $\mathcal{A}$  et un élément géométrique des obstacles;
3. deux ou plusieurs éléments géométriques de  $\mathcal{A}$  et deux ou plusieurs éléments géométriques des obstacles.

Parmi les contacts multiples nous ne considérons que les contacts doubles dans l’exploration de  $\mathcal{C}_{contact}$ . Un *contact double* peut être défini soit comme deux contacts simples entre  $\mathcal{A}$  et deux obstacles différents  $\mathcal{B}_i$  et  $\mathcal{B}_j$  (figure 3.2a) soit comme deux contacts simples entre le robot  $\mathcal{A}$  et un obstacle concave  $\mathcal{B}_i$  (figure 3.2b). Dans la pratique un contact double peut être atteint à partir d’un premier contact simple. Ensuite le robot  $\mathcal{A}$  continue à se déplacer en gardant le premier contact jusqu’à détecter le deuxième contact.

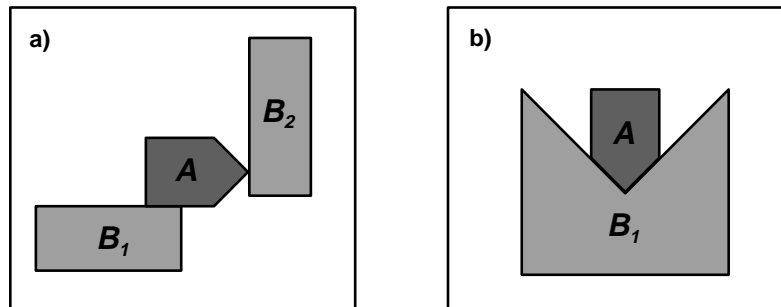


FIG. 3.2 – Illustration des contacts doubles entre une pièce manipulée  $\mathcal{A}$  et les obstacles  $\mathcal{B}_1$  et  $\mathcal{B}_2$  : a) contact double de type (arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_1}$ )-(sommet $_{\mathcal{A}}$ -arête $_{\mathcal{B}_2}$ ), et b) contact double de type (arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_1}$ )-(arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_1}$ ).

### 3.4 Capacités du système de contrôle du robot

Le fait de planifier des mouvements dans l'espace de contact  $\mathcal{C}_{contact}$  impose au système de contrôle du robot  $\mathcal{A}$  la capacité de pouvoir effectuer des déplacements en maintenant des “contacts”. Cette capacité s'appuie sur l'analyse des mesures de “contact” acquises par des *capteurs de contact virtuels* activés par le robot. Ce qui permet de déterminer les directions et les amplitudes des actions de contrôle sur le robot pour continuer le déplacement en “contact”. Cette capacité dans un système de contrôle est appelée la *compliance* et un mouvement qui l'exploite est appelé *mouvement compliant* [Mas81] (figure 3.3a).

La planification des mouvements dans l'espace libre  $\mathcal{C}_{libre}$  impose au système de contrôle du robot  $\mathcal{A}$  la capacité d'effectuer des mouvements qui ne prennent pas en compte les interactions avec les obstacles. Un tel mouvement est appelé *mouvement libre* (figure 3.3b).

Une troisième capacité demandée au système de contrôle du robot  $\mathcal{A}$  consiste à pouvoir garantir la détection des “contacts” en “souplesse” lors des passages de l'espace libre  $\mathcal{C}_{libre}$  vers l'espace de contact  $\mathcal{C}_{contact}$ . Dans ce cas le système de contrôle doit exécuter des mouvements dans l'espace libre  $\mathcal{C}_{libre}$  pouvant déclencher un mode de détection de “contacts” lorsque l'estimation de la localisation du robot  $\mathcal{A}$  est considérée proche des obstacles. Un mouvement de ce type est appelé *mouvement gardé* [WG75] (figure 3.3c).

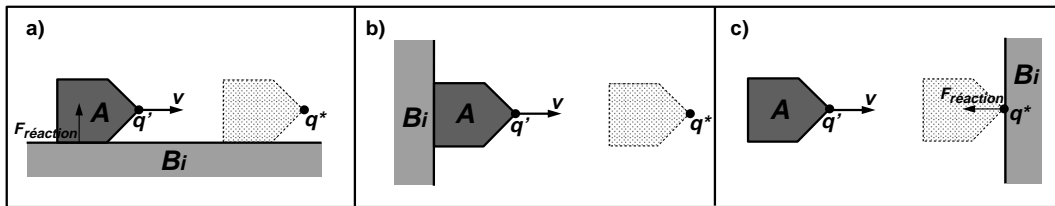


FIG. 3.3 – Illustration des mouvements compliant, libre et gardé pour une pièce manipulée  $\mathcal{A}$ . Dans ce cas, la mesure de contact correspond à une force physique exercée sur la pièce  $\mathcal{A}$  par les obstacles : a) mouvement compliant suivant la vitesse  $\vec{v}$  en maintenant le contact avec la surface de l'obstacle  $\mathcal{B}_i$ ; b) mouvement libre suivant la vitesse  $\vec{v}$ ; c) mouvement gardé suivant la vitesse  $\vec{v}$  et atteignant la surface de l'obstacle  $\mathcal{B}_i$ .

Pour le sous-ensemble des problèmes de planification traités le système de contrôle du robot  $\mathcal{A}$  doit fournir des commandes basées sur les types de mouvements précédents. Les commandes de type *compliant* et *gardé* seront conditionnées par les mesures de “contact” acquises par le robot ainsi que par les amplitudes des mouvements (translations/rotations) effectués alors que les commandes de type *libre* ne seront conditionnés que par les amplitudes des mouvements.

### 3.4.1 Spécification des commandes de contrôle

Un ensemble (nécessaire) de commandes de haut niveau  $\Gamma$  basé sur les types de mouvements présentés ci-dessus est spécifié pour garantir la transformation des chemins  $\tau$  planifiés en plans  $\Sigma$  exécutables. Nous donnerons pour chaque commande de contrôle  $\sigma_i$  sa spécification de type  $(Id_i | Cl_i | PM_i | CA_i)$  ainsi que l'image du sous-chemin  $\tau_i$  dans l'espace valide  $\mathcal{C}_{valide}$  correspondant à une exécution correcte de la commande associée.

#### Définition 3.12 - Vecteur homogène en translation/rotation

Un *vecteur homogène*  $\vec{v} = (\delta x, \delta y, \delta \theta)$  définit l'incrément en rotation  $\delta \theta$  pendant un incrément unitaire en translation selon la direction  $(\delta x, \delta y)$  avec  $\|(\delta x, \delta y)\| = 1$ . Le déplacement en translation/rotation se fait simultanément le long du vecteur  $\vec{v}$ .

#### Définition 3.13 - Axe de rotation $\mathcal{R}_A$

L'*axe de rotation*  $\mathcal{R}_A$  correspond à l'axe perpendiculaire au référentiel  $\mathcal{F}_A$  attaché au point  $\mathcal{O}_A$  de  $\mathcal{A}$ .

*Remarque :* dans un souci de clarté les figures ci-dessous concernant les différentes commandes de contrôle ne montrent que les configurations initiale et finale du robot  $\Gamma$  dans l'espace de travail  $\mathcal{W}\Gamma$  avant et après l'exécution des commandes. Pour chaque commande de contrôle  $\Gamma$  la transition entre ces deux configurations est définie par ses paramètres de mouvement.

#### 1. Commande-libre-en-distance (figure 3.4) :

- Condition initiale **Cl** : configuration  $q' \in \mathcal{C}_{valide}$ .
- Paramètres du mouvement **PM** :  
 direction en translation/rotation : vecteur homogène  $\vec{v} = (\delta x, \delta y, \delta \theta) \in \Gamma$   
 position de l'axe de rotation :  $\mathcal{R}_A \in \Gamma$   
 distance en translation :  $d > 0 \in \Gamma$   
 angle de rotation :  $\theta$ .
- Condition d'arrêt **CA** : atteinte de la configuration  $q^*$ <sup>3</sup> telle que  $q^* \in \mathcal{C}_{libre}$ <sup>4</sup>  $\Gamma$   
 $\|(x^*, y^*) - (x', y')\| = d$  et  $|\theta^* - \theta'| = |\theta|$ .
- Sous-chemin  $\tau_i$  associé : sous-chemin engendré par un vecteur homogène  $\vec{v}$  atteignant la configuration but  $q_f$  lorsque celle-ci appartient à  $\mathcal{C}_{libre}$ .

---

3.  $q^*$  représente la configuration courante.

4. Dans le cas d'une pièce manipulée  $\mathcal{A}$ ,  $q^* \in \mathcal{C}_{libre} - \sum \vec{f}_{réaction}$  sur  $\mathcal{A} = 0$ , avec  $\vec{f}_{réaction}$  une mesure de "contact"  $M_{contact}$  acquise à la configuration  $q^*$  par  $\mathcal{A}$ . Dans le cas d'un robot mobile  $\mathcal{A}$ ,  $q^* \in \mathcal{C}_{libre} - \min\{d_{contact}\}$  de  $\mathcal{A} > d_{seuil}$ , avec  $d_{contact}$  une mesure de "contact"  $M_{contact}$  acquise à la configuration  $q^*$  par  $\mathcal{A}$ , et  $d_{seuil} > 0$  une distance en-dessous de laquelle on considère que  $\mathcal{A}$  est en contact avec les obstacles  $\mathcal{B}_i$ .

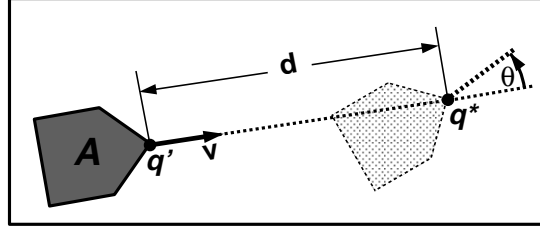


FIG. 3.4 – Illustration d’une commande libre suivant le vecteur homogène  $\vec{v}$  le long d’une distance  $d$  et un angle  $\theta$ .

## 2. Commande-gardée (figure 3.5) :

- Condition initiale Cl : configuration  $q' \in \mathcal{C}_{valide}$ .
- Paramètres du mouvement PM :
  - direction en translation/rotation : vecteur homogène  $\vec{v} = (\delta x, \delta y, \delta \theta)\Gamma$
  - position de l’axe de rotation :  $\mathcal{R}_A\Gamma$
  - distance en translation :  $d > 0\Gamma$
  - angle de rotation :  $\theta\Gamma$
  - Capteur de contact “virtuel” à activer :  $CV_A(Id)\Gamma$
  - Seuil de la mesure de “contact” sur  $\mathcal{A}$  à détecter :
    - $\vec{f}_{seuil}$  sur  $CV_A(Id)$  (cas d’une pièce manipulée) $\Gamma$ ou
    - $d_{seuil}$  sur  $CV_A(Id)$  (cas d’un robot mobile).
- Condition d’arrêt CA : atteinte de la configuration  $q^*$  telle que
  - $\vec{f}_{réaction} \geq \vec{f}_{seuil}$  (cas d’une pièce manipulée) $\Gamma$ ou
  - $d_{contact} \leq d_{seuil}$  (cas d’un robot mobile).

(–  $q^*$  définit un “contact” entre le robot  $\mathcal{A}$  et un obstacle  $\mathcal{B}_i$  où l’élément géométrique  $Id$  de  $\mathcal{A}$  intervient.)
- Sous-chemin  $\tau_i$  associé : sous-chemin engendré par un vecteur homogène  $\vec{v}$  dans  $\mathcal{C}_{libre}$  atteignant un obstacle  $\mathcal{B}_i$ .

## 3. Commande-gardée-jusqu’à-alignement (figure 3.6) :

- Condition initiale Cl : configuration  $q' \in \mathcal{C}_{valide}$ .
- Paramètres du mouvement PM :
  - sens de la rotation : positif/négatif $\Gamma$
  - position de l’axe de rotation :  $(x, y)$  “axe fixe” $\Gamma$
  - angle de rotation :  $\theta\Gamma$
  - Capteur de contact “virtuel” à activer :  $CV_A(Id)\Gamma$
  - Seuil de la mesure de “contact” sur  $\mathcal{A}$  à détecter :
    - $\vec{f}_{seuil}$  sur  $CV_A(Id)$  (cas d’une pièce manipulée) $\Gamma$ ou



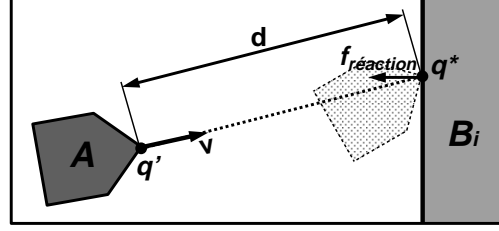


FIG. 3.5 – Illustration d’une commande gardée suivant le vecteur homogène  $\vec{v}$  le long d’une distance  $d$  et un angle  $\theta$ . Le contact détecté sur le sommet  $Id = s_j^A$  induit une force de réaction  $\vec{f}_{réaction}$  sur la pièce manipulée  $\mathcal{A}$ .

- $d_{seuil}$  sur  $CV_{\mathcal{A}}(Id)$  (cas d’un robot mobile).
- Condition d’arrêt CA : atteinte de la configuration  $q^*$  telle que  $|\theta^* - \theta'| = |\theta|$  et
  - $\vec{f}_{réaction} \geq \vec{f}_{seuil}$  (cas d’une pièce manipulée) $\Gamma$ ou
  - $d_{contact} \leq d_{seuil}$  (cas d’un robot mobile).
- Sous-chemin  $\tau_i$  associé : sous-chemin dans  $\mathcal{C}_{libre}$  ou  $\mathcal{C}_{contact}$  engendré par une rotation pure $\Gamma$ autour d’un axe fixe $\Gamma$ atteignant un obstacle  $\mathcal{B}_i$ .

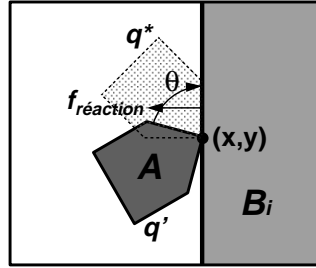


FIG. 3.6 – Illustration d’une commande gardée en rotation effectuée autour d’un axe perpendiculaire en  $(x, y)$  selon un angle  $\theta$  négatif. Le contact détecté sur la face  $Id = \vec{E}_j^A$  induit une force de réaction  $\vec{f}_{réaction}$  sur la pièce manipulée  $\mathcal{A}$ .

#### 4. Commande-compliante-en-distance (figure 3.7) :

- Condition initiale CI : configuration  $q' \in \mathcal{C}_{contact}$  ayant un contact simple.
- Paramètres du mouvement PM :
  - direction en translation/rotation : vecteur homogène  $\vec{v} = (\delta x, \delta y, \delta \theta)\Gamma$
  - distance en translation :  $d > 0\Gamma$
  - Capteur de contact “virtuel” à activer :  $CV_{\mathcal{A}}(Id)\Gamma$
  - mesure de “contact” sur  $\mathcal{A}$  à maintenir :

- $\vec{f}_{seuil}$  sur  $CV_{\mathcal{A}}(Id)$  (cas d’une pièce manipulée) $\Gamma$ ou

- $d_{seuil}$  sur  $CV_{\mathcal{A}}(Id)$  (cas d'un robot mobile).
- Condition d'arrêt CA : atteinte de la configuration  $q^*$  telle que  $\|(x^*, y^*) - (x', y')\| = d$  et
  - $\vec{f}_{réaction} \geq \vec{f}_{seuil}$  (cas d'une pièce manipulée)  $\Gamma$ ou
  - $d_{contact} \leq d_{seuil}$  (cas d'un robot mobile).
- Sous-chemin  $\tau_i$  associé : sous-chemin dans  $\mathcal{C}_{contact}$  qui maintient le “contact” avec une arête d'un obstacle  $\mathcal{B}_i$  sur une distance définie.

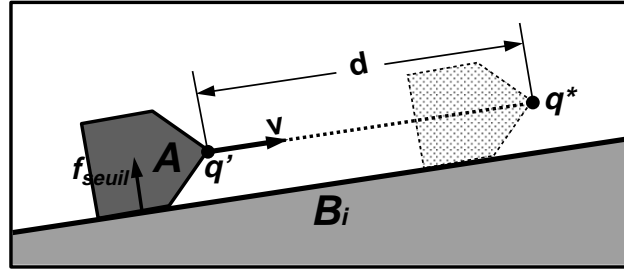


FIG. 3.7 – Illustration d'une commande compliant, maintenant la force  $\vec{f}_{seuil}$  sur la face  $Id = \vec{E}_j^A$ , suivant le vecteur homogène  $\vec{v}$  sur une distance  $d$  (cas d'une pièce manipulée).

### 5. Commande-compliant-jusqu'à-lâcher-contact (figure 3.8) :

- Condition initiale Cl : configuration  $q' \in \mathcal{C}_{contact}$  ayant un contact simple.
- Paramètres du mouvement PM :
  - direction en translation/rotation : vecteur homogène  $\vec{v} = (\delta x, \delta y, \delta \theta) \Gamma$
  - distance en translation :  $d > 0 \Gamma$
  - Capteur de contact “virtuel” à activer :  $CV_{\mathcal{A}}(Id) \Gamma$
  - mesure de “contact” sur  $\mathcal{A}$  à maintenir :
    - $\vec{f}_{seuil}$  sur  $CV_{\mathcal{A}}(Id)$  (cas d'une pièce manipulée)  $\Gamma$ ou
    - $d_{seuil}$  sur  $CV_{\mathcal{A}}(Id)$  (cas d'un robot mobile).
- Condition d'arrêt CA : atteinte de la configuration  $q^*$  telle que  $q^* \in \mathcal{C}_{libre}$ .
- Sous-chemin  $\tau_i$  associé : sous-chemin dans  $\mathcal{C}_{contact}$  qui maintient le “contact” avec une arête d'un obstacle  $\mathcal{B}_i$  jusqu'à détecter son extrémité.

### 6. Commande-compliant-jusqu'au-contact (figure 3.9) :

- Condition initiale Cl : configuration  $q' \in \mathcal{C}_{contact}$  ayant un contact simple.

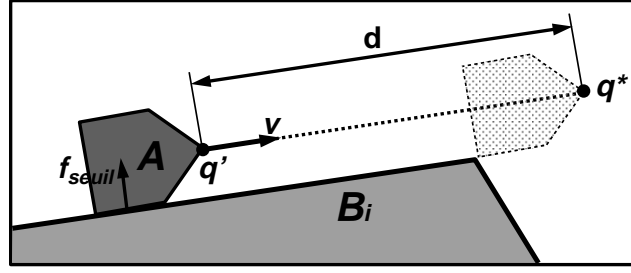


FIG. 3.8 – Illustration d’une commande compliant, maintenant la force  $\vec{f}_{seuil}$  sur la face  $Id = \vec{E}_j^A$ , suivant le vecteur homogène  $\vec{v}$  sur une distance  $d$ . Le contact est lâché à la fin du déplacement (cas d’une pièce manipulée).

– Paramètres du mouvement PM :

direction en translation/rotation : vecteur homogène  $\vec{v} = (\delta x, \delta y, \delta \theta)\Gamma$

distance en translation :  $d > 0\Gamma$

Capteurs de contact “virtuels” à activer :  $CV_{\mathcal{A}}(Id^1)\Gamma CV_{\mathcal{A}}(Id^2)\Gamma$

mesure de “contact” sur  $\mathcal{A}$  à maintenir :

- $\vec{f}'_{seuil}$  sur  $CV_{\mathcal{A}}(Id^1)$  (cas d’une pièce manipulée) $\Gamma$ ou
- $d'_{seuil}$  sur  $CV_{\mathcal{A}}(Id^1)$  (cas d’un robot mobile).

Seuil de la mesure de “contact” sur  $\mathcal{A}$  à détecter :

1.  $\vec{f}''_{seuil}$  sur  $CV_{\mathcal{A}}(Id^2)$  (cas d’une pièce manipulée) $\Gamma$ ou
2.  $d''_{seuil}$  sur  $CV_{\mathcal{A}}(Id^2)$  (cas d’un robot mobile).

– Condition d’arrêt CA : atteinte de la configuration  $q^*$  telle que

- $\vec{f}^1_{réaction} \geq \vec{f}'_{seuil}$  et  $\vec{f}^2_{réaction} \geq \vec{f}''_{seuil}$  (cas d’une pièce manipulée) $\Gamma$ ou
- $d^1_{contact} \leq d'_{seuil}$  et  $d^2_{contact} \leq d''_{seuil}$  (cas d’un robot mobile).

(–  $q^*$  ayant un contact double).

– Sous-chemin  $\tau_i$  associé : sous-chemin dans  $\mathcal{C}_{contact}$  qui maintient un “contact” avec l’arête d’un obstacle  $\mathcal{B}_i$  jusqu’à détecter un nouveau “contact”.

### 7. Commande-compliant-multiple-jusqu’à-lâcher-contact (figure 3.10) :

– Condition initiale Cl : configuration  $q' \in \mathcal{C}_{contact}$  ayant un contact double.

– Paramètres du mouvement PM :

sens de la rotation : positif/négatif $\Gamma$

nombre de configurations de contrôle en contact double :  $n^c\Gamma$

séquence de configurations de contrôle en contact double :  $\{q_1, \dots, q_{n^c}\}$ .

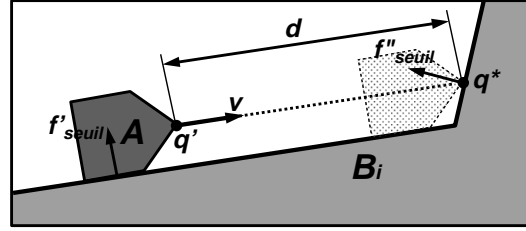


FIG. 3.9 – Illustration d’une commande souple, maintenant la force  $\vec{f}'_{seuil}$  sur la face  $Id^1 = \vec{E}_j^A$ , suivant le vecteur homogène  $\vec{v}$  sur une distance  $d$ . Le nouveau contact détecté sur le sommet  $Id^2 = s_k^A$  induit une force de réaction  $\vec{f}''_{seuil}$  sur  $\mathcal{A}$  (cas d’une pièce manipulée).

- Condition d’arrêt CA: atteinte de la configuration  $q^*$  telle que  $q^* \in \mathcal{C}_{contact}$  définissant un contact simple.
- Sous-chemin  $\tau_i$  associé: sous-chemin dans  $\mathcal{C}_{contact}$  qui maintient un double contact en rotation jusqu’à lâcher un des contacts.

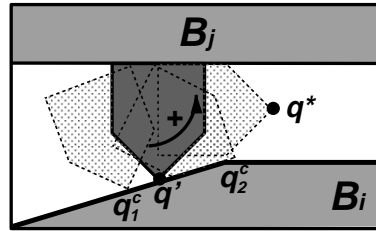


FIG. 3.10 – Illustration d’une commande souple suivant la séquence de configurations en contact double  $\{q_1, q_2\}$  selon le sens positif en rotation jusqu’à lâcher un contact (cas d’une pièce manipulée).

## 3.5 Conclusion

Les définitions et notations de base ainsi que la formulation du problème de planification traité ont été présentées. Le problème posé a été exprimé en un problème équivalent dans l’espace des configurations  $\mathbb{R}^2 \times [0, 2\pi]$  ce qui fait que le robot  $\mathcal{A}$  devient un robot ponctuel  $q_A$  entouré d’obstacles tridimensionnels appelés  $\mathcal{C}$ -obstacles. La re-formulation du problème permet d’une part de réduire la complexité du problème de planification et d’autre part d’intégrer d’une manière plus aisée le raisonnement concernant les incertitudes sur le robot. Certains aspects liés aux capacités réelles d’un robot ont été aussi traités afin de générer des solutions faisables. L’aspect exécution contraint la planification à fournir des plans robustes pour le robot à partir des chemins géométriques robustes. L’aspect évolution des incertitudes du robot dans l’espace

libre contraint la planification à considérer l'exploration de l'espace de contact afin de les réduire grâce à l'interaction du robot avec les obstacles. De plus, cette exploration permet d'engendrer des solutions qui ne sont pas considérées par les stratégies d'évitement d'obstacles. Cependant, l'exploration de l'espace de contact fait augmenter la complexité du problème de planification et ne peut être exploitée que si le robot dispose d'un système de contrôle fiable pour détecter et gérer les contacts.

# Chapitre 4

## Principes de base de notre approche de planification

---

L'objectif de ce chapitre est de présenter les principes de base de l'approche de planification que nous proposons pour résoudre le problème traité dans le chapitre 3 afin de mieux faire comprendre sa philosophie et son application. Cette approche est née des besoins identifiés dans des travaux réalisés en planification de mouvements fins (robotique d'assemblage). Ces besoins sont principalement :

- de garantir l'exécution des mouvements planifiés en présence d'incertitude;
- d'utiliser les contacts entre le robot et les obstacles pour guider la recherche des solutions;
- d'explorer efficacement l'espace de recherche en temps de planification.

Ces besoins peuvent être trouvés dans d'autres domaines de planification de tâches robotiques et en particulier dans la robotique mobile. Bien entendu dans ce cas les "contacts" doivent être définis à proximité des obstacles. Les besoins identifiés ont guidé différents travaux qui ont débouché sur l'approche de planification proposée dans ce document.

Dans la suite les méthodes que nous avons retenues pour composer l'approche proposée seront présentées. Ensuite le principe d'intégration des méthodes dans l'approche sera expliqué et un exemple illustrera son application. L'approche de résolution sera détaillée dans le chapitre 5 et des tâches de validation seront illustrées dans le chapitre 6.

### 4.1 Composantes de l'approche de planification

Notre approche de planification intègre deux méthodes de planification indépendantes et de nature locale. De plus les deux méthodes à l'origine utilisent un principe

de résolution d'évitement d'obstacles. La validation de ces méthodes dans plusieurs applications ont permis de constater leurs avantages et leurs inconvénients. Pour chacune des méthodes nous présentons son principe et signalons les adaptations de base vis-à-vis des besoins identifiés précédemment pour son intégration dans notre approche.

### 4.1.1 La méthode de potentiel fictif

Dans cette méthode de planification le robot est considéré comme une particule sous l'influence d'une fonction de potentiel fictif. De ce fait la méthode est appliquée dans l'espace des configurations  $\mathcal{C}$  puisque le robot y est réduit à un point. Dorénavant nous considérons un robot ponctuel. La fonction de potentiel classique est composée de deux potentiels de base définis dans l'espace libre  $\mathcal{C}_{libre}$  (figure 4.1) : un *potentiel attractif* qui attire le robot vers le but à atteindre et un *potentiel répulsif* qui repousse la robot des obstacles [Kha86]. Le potentiel attractif ne tient pas compte des obstacles alors que le potentiel répulsif pour sa part ne tient pas compte du but à atteindre. L'approche de planification consiste donc à évaluer itérativement la fonction de potentiel pour tenter d'atteindre la configuration but  $q_f$  depuis une configuration initiale  $q_0$ . A chaque itération un déplacement incrémental dans  $\mathcal{C}_{libre}$  est déterminé par le gradient induit par la fonction de potentiel sur la configuration courante. Ainsi le robot se dirige vers la configuration but (où la fonction de potentiel évalue à sa valeur minimale) en évitant les obstacles (où la fonction de potentiel évalue à des valeurs très élevées). Bien que son principal avantage soit son efficacité en temps de calcul l'approche reste sensible aux minima locaux engendrés par des configurations de blocage ou d'oscillation. Ces situations de blocage doivent donc être traitées par des stratégies qui explorent l'espace autour afin de repositionner le robot là où la fonction de potentiel puisse continuer la recherche du but.

Les méthodes basées sur ce principe sont de type *local* et leur définition dépend de la fonction de potentiel et de la stratégie de traitement des minima locaux appliquées. En ce qui concerne la définition du potentiel il existe un compromis entre sa complexité et le nombre de minima locaux qu'elle admet. Dans le cas idéal on voudrait toujours disposer d'un potentiel qui n'ait qu'un minimum correspondant au but à atteindre (minimum global). Selon la complexité du problème de planification (la dimension de l'espace de recherche; la géométrie et les degrés de liberté du robot; la géométrie la distribution et le nombre d'obstacles; etc.) le potentiel sera plus ou moins complexe à définir. Barraquand et Latombe [BL89] proposent une fonction de potentiel qui sert à illustrer le compromis entre la complexité du problème de planification et la complexité du potentiel. La fonction proposée est numérique et définie sur une discrétisation uniforme de l'espace d'évolution du robot. Dans le cas d'un robot ponctuel la définition d'une telle fonction n'admet qu'un seul minimum à la position but du robot. Cependant dans le cas d'un robot différent d'un point la fonction de potentiel est composée de potentiels numériques définis sur des points de contrôle du robot et cette composition peut admettre des minima locaux.

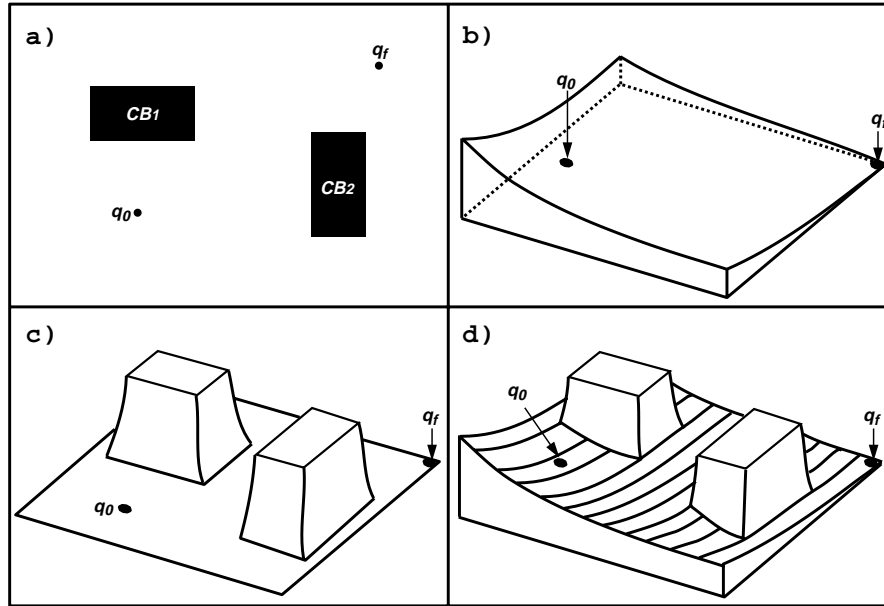


FIG. 4.1 – a) Un problème de planification dans l'espace des configurations  $\mathbb{R}^2$ .  
 b) Représentation d'un potentiel attractif qui attire le robot vers le but.  
 c) Représentation d'un potentiel répulsif qui repousse le robot des obstacles.  
 d) Représentation d'un champ de potentiel fictif qui intègre un potentiel attractif et un potentiel répulsif.[Lat91]

En ce qui concerne les stratégies pour sortir des minima locaux elles s'appuient sur la construction de graphes à partir des minima locaux identifiés par le potentiel. Un tel graphe permet d'évaluer l'état courant d'exploration de l'espace de recherche pour déterminer la meilleure configuration de déblocage et ainsi pouvoir continuer la recherche du but.

Comme nous l'avons déjà dit le principal avantage de cette approche est en général son efficacité en temps : des solutions sont trouvées rapidement. Mais si l'approche est efficace elle ne garantit l'obtention de chemins optimaux. Par exemple il est possible qu'avant de trouver un chemin solution le potentiel tombe plusieurs fois sur un même minimum local. Un deuxième avantage est son vaste domaine d'application en planification robotique. En particulier cette approche est intéressante dans deux grands ensembles de problèmes où d'autres approches ne sont pas applicables ou dégèrent par rapport à leur complexité algorithmique : d'une part des problèmes dont la dimension de l'espace de recherche est élevée ; d'autre part des problèmes où seule une connaissance incomplète de l'environnement du robot est disponible.

L'inconvénient de l'approche est en général son incomplétude : il se peut qu'en certains cas un chemin ne soit pas trouvé même s'il existe.



#### 4.1.1.1 Notre contribution

Les contributions de la fonction de potentiel proposée concernent trois aspects :

1. L'ESPACE DE RECHERCHE : contrairement à l'approche classique les obstacles ne repoussent pas le robot ponctuel et donc celui-ci peut arriver au contact sans entrer en collision avec les obstacles. Afin de détecter les contacts des techniques efficaces et fiables sont nécessaires dans l'approche qui intègre le potentiel proposé. Au moment du contact une nouvelle force est engendrée par la fonction de potentiel à cause de l'interaction entre le robot et la surface en contact. Cette force s'appelle la *force de cohésion* et sa direction suit la normale intérieure à la surface de contact au point de contact. Grâce à cette force le robot peut garder le contact avec une surface en effectuant des mouvements compliants qui réduisent l'incertitude relative entre le robot et les obstacles. La relation entre la force attractive et la force de cohésion sur une configuration de contact détermine si le robot maintient ou lâche le contact. Ainsi dans l'espace  $\mathcal{C}_{libre}$  le robot se dirige directement vers son but et lorsque celui-ci est contraint par un obstacle un contact est défini avec cet obstacle. Ensuite le robot continue son déplacement dans l'espace  $\mathcal{C}_{contact}$  en suivant la surface de l'obstacle en contact. La direction du déplacement est celle qui rapproche le robot de son but. Le robot évolue en contact soit jusqu'au point où le suivi de la surface de l'obstacle ne le rapproche plus de son but (passage dans l'espace  $\mathcal{C}_{libre}$ ) (figure 4.2a) soit jusqu'à détecter un blocage (situation de minimum local);
2. LES INCERTITUDES GÉOMÉTRIQUES LIÉES AU ROBOT : ces incertitudes sont évaluées au fur et à mesure que le robot se déplace dans l'espace de recherche (figure 4.2b). De ce fait la fonction engendre des chemins robustes vis-à-vis des incertitudes de position d'orientation et de contrôle sur le robot. L'évaluation explicite des incertitudes est faite à l'aide des modèles géométriques d'incertitude présentés dans le chapitre 2;
3. LE TRAITEMENT DES MINIMA LOCAUX : ce problème est traité par la technique d'exploration appliquée dans la méthode de planification présentée dans la suite.

L'avantage et l'inconvénient principaux de ce type de fonctions c'est-à-dire l'efficacité et l'incomplétude sont aussi présents dans le potentiel fictif proposé. Les validations (illustrées dans le chapitre 6) pour des instantiations du problème posé dans le chapitre 3 montrent que des solutions sont trouvées pour des situations variées.

#### 4.1.2 L'algorithme Fil d'Ariane

L'algorithme Fil d'Ariane est une approche de planification de plans (pour un robot) qui exploite efficacement l'espace de commandes pour explorer l'espace des configurations  $\mathcal{C}$  [Ahu94]. Un *plan* dans cette approche est défini comme un ensemble de commandes pour les actionneurs du robot. Un *plan valide* entre deux configurations  $q_a$

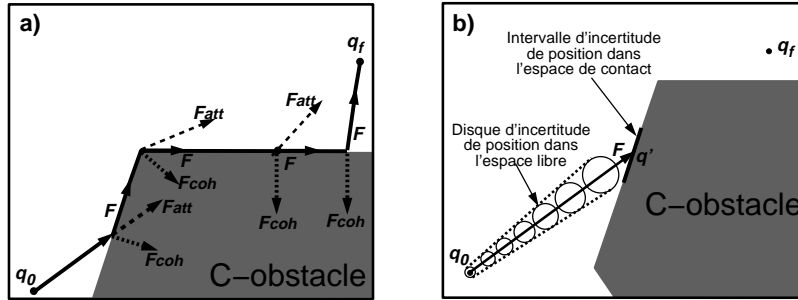


FIG. 4.2 – a) Le potentiel attractif attire le robot de  $q_0$  vers  $q_f$  en appliquant une force  $\vec{F}_{att}$  sur la configuration courante. Dans l'espace libre  $\mathcal{C}_{libre}$ , cette force guide le robot directement vers la configuration but. Dans l'espace de contact  $\mathcal{C}_{contact}$ , cette force est composée avec une force de cohésion  $\vec{F}_{coh}$  qui a tendance à maintenir le contact. La force  $\vec{F}$  résultante de la combinaison des forces  $\vec{F}_{att}$  et  $\vec{F}_{coh}$  guide le robot soit vers  $\mathcal{C}_{libre}$ , soit vers  $\mathcal{C}_{contact}$ .

b) Evaluation explicite et incrémentale des incertitudes.

et  $q_b$  est un plan libre de collisions  $\Gamma$  donc dans l'espace  $\mathcal{C}_{libre}$   $\Gamma$  reliant  $q_a$  et  $q_b$  par un chemin continu. Le principe de cette approche consiste à construire incrémentalement une approximation discrète de l'espace  $\mathcal{C}_{libre}$  au fur et à mesure qu'un plan valide est cherché entre une configuration initiale  $q_0$  et une configuration finale  $q_f$ <sup>1</sup>. L'originalité de cette méthode réside dans l'adaptation automatique de la résolution d'exploration à la complexité du problème posé. Ainsi  $\Gamma$  pour les problèmes où le robot peut exécuter de longs déplacements  $\Gamma$  la résolution d'exploration sera grande; par contre  $\Gamma$  pour les problèmes où l'espace  $\mathcal{C}_{libre}$  est fortement contraint par les obstacles (portes et couloirs étroits)  $\Gamma$  le robot est forcé de se déplacer en petits déplacements et donc la résolution d'exploration sera petite. En conclusion  $\Gamma$  la résolution s'adapte automatiquement selon les composantes de  $\mathcal{C}_{libre}$  que le robot doit visiter pour trouver une solution au problème posé.

L'algorithme Fil d'Ariane est composé de deux fonctions indépendantes : EXPLORE et SEARCH.

#### 4.1.2.1 La fonction EXPLORE

L'objectif d'EXPLORE est de construire une approximation discrète de  $\mathcal{C}_{libre}$  accessible depuis  $q_0$ . Cette approximation est représentée par un graphe d'exploration. L'idée de base consiste à placer des balises équitablement réparties dans  $\mathcal{C}_{libre}$ . Pour chaque balise placée  $l_i$   $\Gamma$  il existe un plan valide et connu depuis  $q_0$  qui l'atteint.  $q_0$  est considérée comme la première balise (i.e.  $l_0 = q_0$ ). Ensuite  $\Gamma$  chaque nouvelle balise  $l_i$  ( $i \geq 1$ ) est placée le plus loin possible des balises précédemment définies (i.e.  $l_0, \dots, l_{i-1}$ ). Chaque nouvelle balise définit un nœud du graphe d'exploration; ce dernier est lié au nœud de

1.  $q_0$  et  $q_f$  sont supposées dans l'espace  $\mathcal{C}_{libre}$ .

la balise qui a été la configuration de départ pour atteindre cette nouvelle balise (figure 4.3).

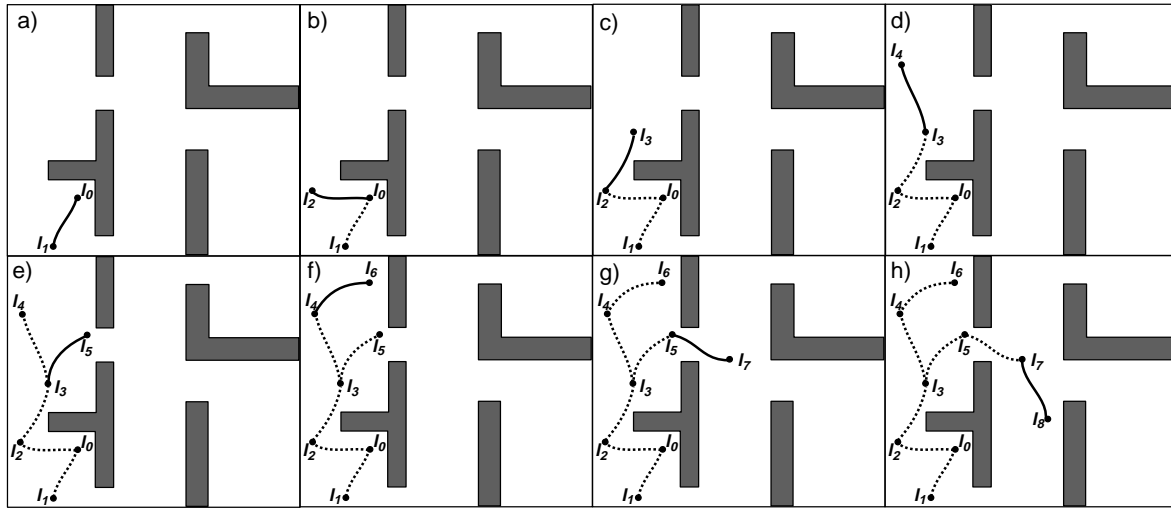


FIG. 4.3 – Construction du graphe de balises avec la fonction EXPLORE du Fil d'Ariane.

Le placement des balises est indépendant de la configuration but du problème posé. Dans [Ahu94] il est formellement démontré que pour toute configuration  $q'$  dans  $\mathcal{C}_{libre}$  accessible depuis  $q_0$  EXPLORE peut placer une balise  $l_i$  à une distance inférieure ou égale à une résolution arbitraire  $\varepsilon$ . Cette propriété est appelée *l'approximation à  $\varepsilon$ -près de l'espace de recherche accessible*. Si  $q_f$  est une configuration accessible depuis  $q_0$  on peut affirmer que la complexité d'un problème est inversement proportionnelle à la plus grande résolution constante  $\varepsilon^*$  qui définit une séquence de balises  $\{l_0, \dots, l_i\}$  dont  $l_i$  est à une distance de  $q_f$  inférieure ou égale à  $\varepsilon^*$ . Autrement dit plus  $\varepsilon^*$  est grand moins le problème est difficile (i.e. plus EXPLORE trouvera rapidement une solution).

Soient  $\varepsilon \in \mathbb{R}^+$  une résolution donnée et  $q_0$  une configuration initiale. La fonction EXPLORE peut être exprimée comme un problème d'optimisation :

$$\text{EXPLORE}(L_i, \varepsilon) = \begin{cases} \text{Maximiser } d(L_i, \hat{q}) \\ \hat{q} \in X \end{cases} \quad (4.1)$$

avec  $i > 0$  l'indice de la nouvelle balise à définir  $L_i = (l_0, \dots, l_{i-1})$  l'ensemble des balises définies auparavant  $l_0$  la balise initiale définie par  $q_0 \in X \subset \mathcal{C}_{libre}$  le sous-ensemble des configurations défini comme l'union des "boules" de rayon  $\varepsilon$  autour de chaque balise dans  $L_i$  et  $d(L_i, \hat{q})$  la distance de l'ensemble  $L_i$  à la configuration  $\hat{q}$ . A chaque pas  $i$  d'évaluation d'EXPLORE la configuration  $\hat{q}$  choisie est celle qui maximise la distance à l'ensemble connu  $L_i$ . Ensuite on peut générer un *plan* de commandes depuis la plus proche balise dans  $L_i$  et arrivant à  $\hat{q}$ . Ce plan n'est pas nécessairement un *plan valide*. Dans le cas où c'est un plan valide  $\hat{q}$  devient la nouvelle balise  $l_i$  dans  $L_i$ . Dans le cas contraire l'application d'une stratégie de "rebondissement" sur les obstacles permet

de transformer un plan *non-valide* (avec collisions) en un *plan valide*. La configuration atteinte  $\hat{q}'$  par ce nouveau plan devient la nouvelle balise  $l_i$  dans  $L_i$ .

#### 4.1.2.2 La fonction SEARCH

Cette fonction est définie comme un planificateur local de plans qui cherche à atteindre la configuration finale  $q_f$  à partir de chaque balise  $l_i$  définie par EXPLORE. La fonction SEARCH peut être exprimée comme un problème d'optimisation :

$$\text{SEARCH}(l_i, q_f) = \begin{cases} \text{Minimiser } d'(\gamma(l_i), q_f) \\ \gamma(l_i) \in \Gamma(l_i) \end{cases} \quad (4.2)$$

avec  $\gamma(l_i)$  un *plan valide* partant de  $l_i$   $\Gamma(l_i)$  l'ensemble de plans d'origine  $l_i$  dans l'espace des commandes  $\Gamma$  et  $d'(\gamma(l_i), q_f)$  la distance de la nouvelle extrémité du plan  $\gamma(l_i)$  à la configuration  $q_f$ . Lorsqu'un plan  $\gamma(l_i)$  atteint la configuration  $q_f$  on obtient  $d'(\gamma(l_i), q_f) = 0$ . Dans ce cas le *plan valide* de  $q_0$  à  $q_f$  est reconstruit à partir du graphe d'exploration comme la composition des *plans valides* liés aux nœuds de la branche du graphe de  $l_0$  à  $l_i$  (engendré par EXPLORE)  $\Gamma$  suivi par le *plan valide* de  $l_i$  à  $q_f$  (engendré par SEARCH).

#### 4.1.2.3 L'intégration d'EXPLORE et SEARCH

Si l'on dispose d'une fonction SEARCH qui garantit la résolution d'un problème de planification de mouvement dans un disque de rayon  $\varepsilon$  et si l'on tient compte de la propriété d'approximation à  $\varepsilon$ -près liée à EXPLORE  $\Gamma$  SEARCH en coopération avec EXPLORE définissent une approche de planification *complète*<sup>2</sup> pour la résolution  $\varepsilon$ .

Un algorithme simplifié du Fil d'Ariane en fonction de  $q_0, q_f$  et  $\varepsilon$  est :

DÉBUT.

$l_0 = q_0, i = 1, L_i = \{l_{i-1}\}, \varepsilon_i = d(L_i, q_f)$ .

##### 1. EXÉCUTION DE SEARCH.

- Résoudre le problème d'optimisation  $\text{SEARCH}(l_{i-1}, q_f)$  en obtenant comme solution le plan valide  $\gamma(l_{i-1})$  et la distance  $d'(\gamma(l_{i-1}), q_f)$ ;
- Si  $d'(\gamma(l_{i-1}), q_f)$  est égal à zéro (i.e.  $q_f$  est atteinte) alors retourner SUCCÈS et le plan composé du *plan valide* de  $l_0$  à  $l_{i-1}$  suivi du *plan valide*  $\gamma(l_{i-1})$ . Sinon  $\Gamma$  si  $\varepsilon_i$  est inférieur à  $\varepsilon$  alors retourner ÉCHEC.

##### 2. EXÉCUTION D'EXPLORE.

- Résoudre le problème d'optimisation  $\text{EXPLORE}(L_i, \varepsilon_i)$  en obtenant comme solution la configuration  $\hat{q}$  et la distance  $d(L_i, \hat{q})$ ;

---

2. Une approche de planification est dite *complète* si une solution peut être trouvée pour un problème lorsque celle-ci existe.

- Trouver  $l_i^*$  dans  $L_i$  qui satisfait la distance  $d(L_i, \hat{q})$ ;
- Générer un *plan*  $P$  depuis  $l_i^*$  et arrivant à  $\hat{q}$ ;
- Si  $P$  est un *plan valide* alors  $l_i = \hat{q}$ .  
Sinon l'appliquer la stratégie de "rebondissement" sur le plan  $P$  afin d'obtenir un *plan valide*  $P'$  avec  $\hat{q}'$  la nouvelle configuration extrême;  $l_i = \hat{q}'$ ;
- $\varepsilon_{i+1} = d(L_i, l_i)/2$ ;
- $L_{i+1} = L_i \cup \{l_i\}$ ;
- $i = i + 1$ ;
- Retourner en 1.

#### 4.1.2.4 Notre contribution

Dans notre approche de planification la fonction EXPLORE du Fil d'Ariane construit incrémentalement un graphe d'exploration de l'espace  $\mathcal{C}_{contact}$  à partir des minima locaux de la fonction de potentiel fictif proposée dans §4.1.1.1 afin de re-positionner le robot sur des configurations lointaines de ceux-ci et ainsi permettre au potentiel de continuer la recherche de la configuration but. Un minimum local est toujours détecté dans l'espace  $\mathcal{C}_{contact}$  car faute d'un potentiel répulsif autour des obstacles le robot ne peut être bloqué qu'en deux cas : le premier à la détection de la configuration but qui n'est pas un minimum local et le deuxième dans une concavité formée par les obstacles. Pour cette raison EXPLORE est adaptée à l'exploration de l'espace  $\mathcal{C}_{contact}$ . Toute situation de minimum local de la fonction de potentiel déclenche l'activation d'EXPLORE qui considère un minimum local comme une balise  $l_i$ . Pour sa part chaque évaluation d'EXPLORE génère une nouvelle balise qui sert comme configuration de départ pour la fonction de potentiel fictif. Une *balise*  $l_i$  dans notre approche est accompagnée par l'estimation de ses incertitudes de position  $\rho_{l_i}$  et d'orientation  $\psi_{l_i}$ . Pour chaque balise  $l_i$  deux voisinages dans l'espace  $\mathcal{C}_{contact}$  sont définis :

1. un *voisinage adjacent* autour de  $l_i$  sur l'obstacle (les obstacles) en contact;
2. un *voisinage visible* sur de nouveaux obstacles potentiellement "atteignables" depuis  $l_i$ . Un obstacle est potentiellement "atteignable" s'il existe au moins une direction de déplacement depuis  $l_i$  permettant au robot de l'atteindre malgré les incertitudes de position d'orientation et de contrôle (voir §2.4.2 et §2.4.3). Ce voisinage peut cependant être indéfini.

Deux stratégies d'exploration d'une balise découlent des voisinages définis (figure 4.4) : la première en contournant les obstacles en contact et la deuxième en cherchant des contacts avec d'autres obstacles.

EXPLORE est donc fonction du type des voisinages à explorer : *adjacent* ou *visible*. Sous chaque mode d'exploration EXPLORE détermine dans les voisinages respectifs

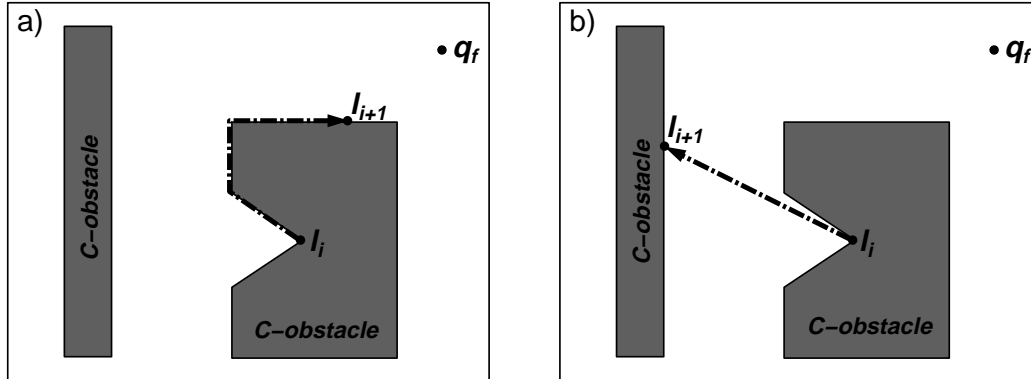


FIG. 4.4 – Deux stratégies d’exploration permettent de contourner une balise  $l_i$  : la première (a) contourne l’obstacle (les obstacles) en contact et la deuxième (b) cherche à définir un contact avec d’autres obstacles.

des balises existantes  $L_i$  la configuration de contact la plus “éloignée” de celles-ci pour re-positionner le robot. Ceci correspond à la phase d’optimisation  $\text{EXPLORE}(L_i, \varepsilon)$  de l’algorithme original. Cette configuration devient un sous-but à atteindre avec le robot depuis une balise précédente. La configuration de contact atteinte par la recherche d’un sous-but définit une nouvelle balise qui deviendra la nouvelle configuration de départ pour la fonction de potentiel.

Les avantages de la fonction  $\text{EXPLORE}$  proposée sont principalement :

- la génération des déplacements plus robustes car les incertitudes sont réduites au contact;
- l’adaptation automatique de la résolution d’exploration à la complexité des situations de minima locaux trouvées.

Par contre la mise en œuvre d’une telle fonction suppose des mécanismes fiables pour gérer les contacts avec le robot.

## 4.2 Principe d’intégration dans l’approche de planification

Le principe d’intégration de l’approche proposée est celui de l’algorithme Fil d’Ariane qui prend d’une part une fonction  $\text{SEARCH}$  de type potentiel fictif adaptée à l’espace  $\mathcal{C}_{valide}$  et présentée dans §4.1.1.1; et d’autre part une fonction  $\text{EXPLORE}$  adaptée à l’espace  $\mathcal{C}_{contact}$  et présentée dans §4.1.2.4.

Les nouvelles fonctions  $\text{SEARCH}$  et  $\text{EXPLORE}$  coopèrent de la façon suivante : d’une part les minima locaux de  $\text{SEARCH}$  deviennent des nouvelles *balises* pour  $\text{EXPLORE}$

et d'autre part les balises générées par EXPLORE deviennent des nouvelles *configurations de départ* pour SEARCH. Ces fonctions tiennent compte explicitement des incertitudes de position d'orientation et de contrôle liées au robot au fur et à mesure de son déplacement afin de générer des déplacements robustes.

La combinaison "SEARCH - EXPLORE" est appliquée soit jusqu'à arriver à la configuration but (succès) soit jusqu'à obtenir une discrétisation assez fine de l'espace  $\mathcal{C}_{contact}$  accessible depuis  $q_0$  sans avoir trouvé de solution (échec).

Le résultat de l'approche pour un problème résolu est un *chemin valide et robuste* depuis  $q_0$  jusqu'à  $q_f$ . Tout chemin valide et robuste est transformé ensuite en un *plan robuste* à partir de l'ensemble de commandes de contrôle exécutables par le robot et présenté dans §3.4.1.

### 4.2.1 Notre contribution

Notre contribution au principe de résolution du Fil d'Ariane est :

1. la génération des chemins *valides* et *robustes* (vis-à-vis des incertitudes géométriques sur le robot);
2. la définition des échecs de la fonction SEARCH (minima locaux) comme *balises* pour la fonction EXPLORE;
3. la définition d'une fonction EXPLORE sous deux formes indépendantes : le suivi d'une surface en contact (*exploration adjacente*) et le passage vers d'autres obstacles (*exploration visible*).

## 4.3 Un exemple de l'approche de planification

L'objectif de cet exemple est d'illustrer et de commenter de manière sommaire l'application de notre approche pour la résolution d'un problème. Nous espérons que cet exemple permettra au lecteur d'avoir une idée plus précise de l'approche proposée avant de passer à l'explication de sa mise en œuvre. Plusieurs exemples de validation de l'approche sont présentés dans le chapitre 6.

### Définition du problème à résoudre

L'instantiation du problème posé dans §3.2 que nous utiliserons comme exemple d'illustration est définie par (figure 4.5a) :

- les obstacles polygonaux  $\{\mathcal{B}_1, \dots, \mathcal{B}_6\}$  dans  $\mathcal{W}$ ;
- le robot  $\mathcal{A}$  triangulaire;

- la configuration initiale  $q_0 = (x_0, y_0, \theta_0)$  et son voisinage d'incertitude  $\mathcal{R}_0$  défini par le disque centré en  $(x_0, y_0)$  de rayon  $\rho_0$ . L'incertitude d'orientation est supposée nulle (i.e.  $\psi_0 = 0$ )<sup>3</sup>;
- la configuration but à atteindre  $q_f = (x_f, y_f, \theta_f)$  et son voisinage d'incertitude  $\mathcal{R}_f$  défini par le disque centré en  $(x_f, y_f)$  de rayon  $\rho_f$ . L'incertitude d'orientation est supposée nulle (i.e.  $\psi_f = 0$ )<sup>4</sup>;
- $\eta = 4$  deg. La déviation maximale pour une translation; et
- $k_r = 0$  la constante de proportionnalité sur les imprécisions en rotation.

Le problème dans l'espace de travail  $\mathcal{W}$  est illustré dans la figure 4.5a. Le problème à résoudre consiste à trouver un chemin  $\tau$  valide et robuste pour le robot ponctuel  $q_A$  partant de  $q_0$  avec une incertitude  $\mathcal{R}_0$  et arrivant à  $q_f$  avec une incertitude inférieure ou égale à  $\mathcal{R}_f$  (figure 4.5b). Les  $\mathcal{C}$ -obstacles  $\{\mathcal{CB}_1, \dots, \mathcal{CB}_6\}$  dans  $\mathcal{C}$  correspondent au grossissement des obstacles dans  $\mathcal{W}$  avec la géométrie du robot  $\mathcal{A}$ . Dans un souci de clarté les figures illustrant l'exemple seront montrées dans l'espace de travail  $\mathcal{W}$ .

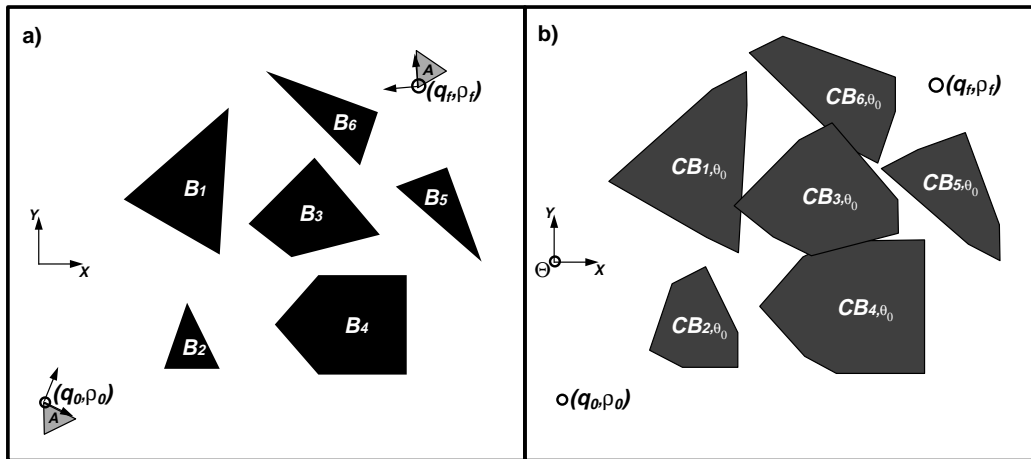


FIG. 4.5 – a) Définition du problème dans l'espace de travail  $\mathcal{W}$ .

b) Re-formulation du problème dans l'espace des configurations  $\mathcal{C}$  (grossissement des obstacles selon l'orientation initiale  $\theta_0$  du robot)(vue sur le plan  $XY$ ).

## Commentaires sur le principe de résolution

DÉBUT : la configuration  $q_0$  est considérée comme la première balise dans l'espace de recherche.

3.  $q_0$  et  $\mathcal{R}_0$  appartiennent à l'espace  $\mathcal{C}_{libre}$ .

4.  $q_f$  et  $\mathcal{R}_f$  appartiennent à l'espace  $\mathcal{C}_{libre}$ .



EXÉCUTION DE SEARCH : SEARCH est appliquée pour tenter de trouver un chemin direct vers  $q_f$  (figure 4.6a). Le long du chemin généré par SEARCH nous pouvons constater les faits suivants par rapport aux mouvements et aux incertitudes :

1. le robot se déplace simultanément en translation et en rotation dans l'espace libre. Un changement sur l'orientation  $\theta$  du robot implique un changement sur les  $\mathcal{C}$ -obstacles  $\mathcal{CB}_{i,\theta}$  considérés;
2. le disque d'incertitude de position augmente dans l'espace libre. Ce disque est toujours associé au point de référence du robot;
3. l'obstacle  $\mathcal{B}_2$  est atteint et le chemin qui conduit au contact est corrigé pour garantir un mouvement robuste;
4. l'incertitude d'orientation est réduite en contact en appliquant une stratégie d'alignement entre une face du robot et une face d'un obstacle;
5. le disque d'incertitude de position dans l'espace libre devient un intervalle au moment du contact. Cet intervalle est associé également au point de référence du robot; cependant dans la figure nous le montrons comme un disque dans un souci de clarté;
6. SEARCH peut continuer la recherche du but en glissant sur la surface de  $\mathcal{B}_2$ ;
7. l'intervalle d'incertitude de position augmente dans l'espace de contact jusqu'à la détection d'un sommet d'un obstacle où il devient un intervalle nul;
8. SEARCH conduit le robot au minimum local  $q_{min1}$  entre les obstacles  $\mathcal{B}_3$  et  $\mathcal{B}_4$ .

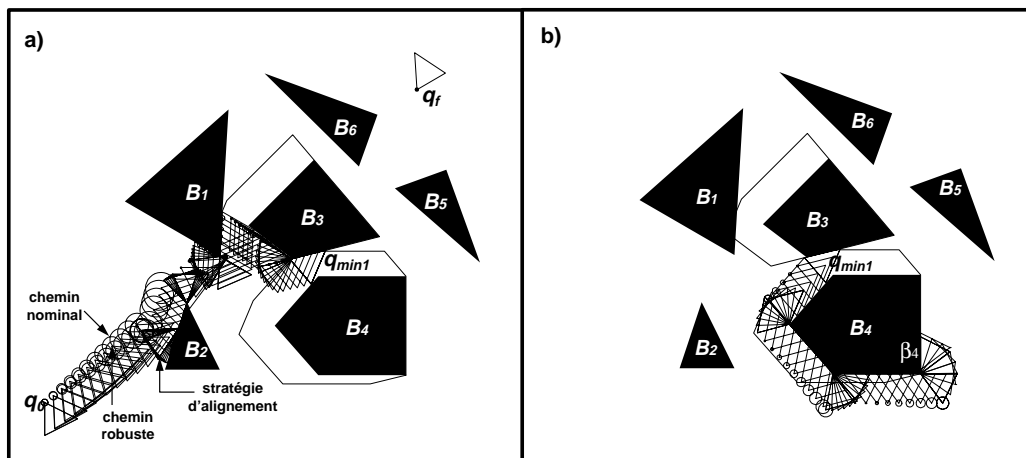


FIG. 4.6 – a) Trace du robot lors de la première tentative de recherche du but avec la fonction SEARCH. b) Trace du robot lors de l'exécution d'EXPLORE en contournant l'obstacle  $\mathcal{B}_4$ .

EXÉCUTION D'EXPLORE :  $q_{min1}$  devient une nouvelle balise dans l'espace de recherche. Afin de sortir du minimum local la fonction EXPLORE place des configurations sous-but sur les surfaces des obstacles qui bloquent le robot ( $\mathcal{B}_3$  et  $\mathcal{B}_4$ ). Ensuite l'EXPLORE réalise une évaluation des sous-but pour choisir le plus éloigné de la balise courante et ainsi avoir plus de possibilités d'y échapper. Le sous-but choisi a été étiqueté  $\beta_4$ .

Une stratégie de contournement de  $\mathcal{B}_4$  est exécutée depuis  $q_{min1}$  pour vérifier si  $\beta_4$  peut être atteint. Cette stratégie suit le chemin le plus court vers le sous-but prenant en compte les incertitudes sur le robot.  $\beta_4$  est atteint et devient une nouvelle balise dans l'espace de recherche (figure 4.6b).

EXÉCUTION DE SEARCH : la fonction SEARCH tente à nouveau d'approcher le but  $q_f$  depuis la nouvelle balise  $\beta_4$ . Cette fois-ci la SEARCH atteint  $q_f$  avec une précision satisfaisante (figure 4.7a). Le chemin généré est donc retenu comme solution au problème posé.

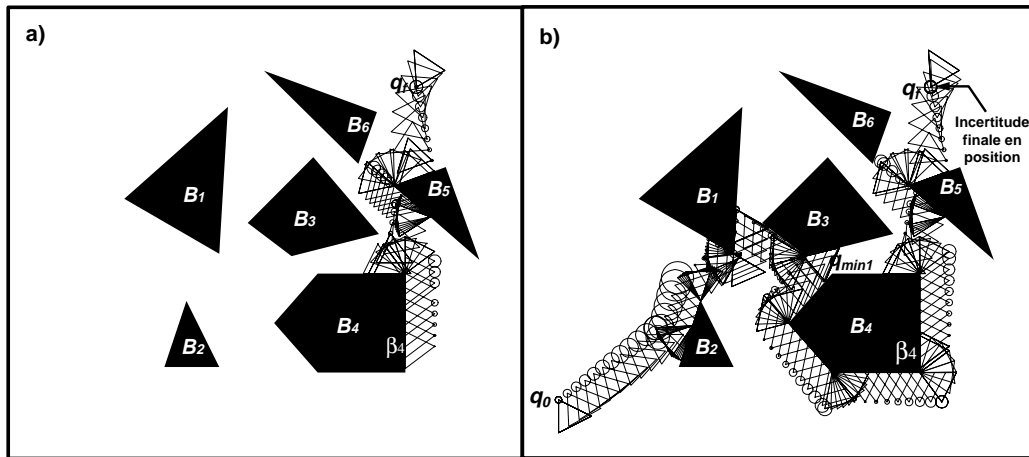


FIG. 4.7 – a) Trace du robot lors de la deuxième tentative de recherche du but avec la fonction SEARCH. b) Chemin valide et robuste défini comme la composition des sous-chemins trouvés par les fonctions SEARCH et EXPLORE.

Le chemin solution est construit comme la composition des chemins de  $q_0$  à  $q_{min1}$  de  $q_{min1}$  à  $\beta_4$  et de  $\beta_4$  à  $q_f$  (figure 4.7b). Puisque le chemin solution est continu et constitué de chemins valides et robustes il est également valide et robuste.

GÉNÉRATION D'UN PLAN ROBUSTE : le dernier pas de l'approche consiste à transformer le chemin solution en un plan robuste à l'aide d'un ensemble de commandes exécutables par le robot.

## 4.4 Conclusion

Les principes de base de l'approche de planification proposée pour résoudre le problème posé dans §3.2 ont été présentés. En principe cette approche satisfait les besoins énoncés au début du chapitre de la manière suivante :

- la prise en compte explicite des incertitudes géométriques pendant la génération des mouvements permet d'engendrer des chemins valides et robustes pour un problème de planification de mouvement. Dans l'espace  $\mathcal{C}_{libre}$  l'estimation des incertitudes sert à garantir des contacts robustes (ceux-ci réduisent l'incertitude). Dans l'espace  $\mathcal{C}_{contact}$  les incertitudes peuvent être réduites davantage selon les situations de contact rencontrées (extrémité d'une face d'un obstacle, l'alignement du robot selon une face d'un obstacle);
- l'inclusion de l'espace  $\mathcal{C}_{contact}$  dans l'espace de recherche permet de générer des déplacements plus robustes car les incertitudes sont réduites au contact. Néanmoins l'exploration dans l'espace  $\mathcal{C}_{libre}$  reste nécessaire pour :
  1. permettre le passage entre des obstacles disjoints;
  2. engendrer des mouvements moins contraints en translation et en rotation;
  3. atteindre la configuration but d'un problème de planification lorsque celle-ci est définie dans l'espace  $\mathcal{C}_{libre}$ ;
- l'intégration de deux fonctions locales (une fonction SEARCH de type potentiel fictif et une fonction EXPLORE incrémentale) donne une approche efficace pour un vaste ensemble de problèmes de planification. La fonction EXPLORE traite les minima locaux de la fonction SEARCH en adaptant la résolution d'exploration à la complexité des minima rencontrés. Au fur et à mesure qu'une solution est recherchée un graphe d'exploration de l'espace  $\mathcal{C}_{contact}$  est construit par EXPLORE. Cependant le caractère local des stratégies appliquées ne garantit pas de solutions optimales.

Après avoir donné les lignes directrices de notre approche nous allons tout d'abord approfondir sur la méthode de résolution (chapitre 5) et ensuite illustrer la validation de son application sur deux types de tâches robotiques (chapitre 6).

# Chapitre 5

## Mise en œuvre de l’approche de planification

---

Le problème consistant à planifier un chemin robuste pour un robot polygonal  $\mathcal{A}$  dans l’espace bidimensionnel  $\mathcal{W}$  parsemé d’obstacles polygonaux  $\Gamma$  a été transformé en un problème équivalent pour un robot ponctuel  $q_{\mathcal{A}}$  évoluant dans l’espace des configurations  $\mathcal{C} = \mathbb{R}^2 \times [0, 2\pi]$  encombré de  $\mathcal{C}$ -obstacles tridimensionnels.

Dans la suite les principaux aspects concernant notre approche de planification de chemins dans l’espace  $\mathcal{C}$  sous des contraintes d’incertitudes géométriques de position  $\Gamma$  d’orientation et de contrôle sont détaillés. Tout d’abord l’espace d’exploration utilisé par l’approche est précisé. En ce qui concerne les  $\mathcal{C}$ -obstacles une méthode de construction incrémentale d’un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  est présentée. Ensuite les composantes de base de l’approche sont expliquées. Ces composantes sont :

- une fonction de potentiel fictif adaptée à la navigation dans l’espace valide  $\mathcal{C}_{valide}$  ;
- une fonction d’exploration de l’espace de contact  $\mathcal{C}_{contact}$  pour sortir des minima locaux de la fonction précédente.

Suit l’intégration des différentes composantes de l’approche modélisée par un système réparti et explicitée avec un algorithme séquentiel. Finalement la transformation des chemins générés en plans est abordée.

### 5.1 L’espace d’exploration

L’approche de planification proposée explore l’espace  $\mathcal{C}_{valide} \subset \mathbb{R}^2 \times [0, 2\pi]$  (équation 3.3). Une configuration  $q$  du robot ponctuel  $q_{\mathcal{A}}$  dans  $\mathcal{C}_{valide}$  appartient soit à l’espace  $\mathcal{C}_{libre}$  (équation 3.2) soit à l’espace  $\mathcal{C}_{contact}$  (équation 3.1). S’il n’y a pas d’obstacles dans l’espace de travail  $\mathcal{W}$  du robot  $\mathcal{A}$   $\mathcal{C}_{libre}$  devient l’espace d’exploration ( $\mathcal{C}_{valide} = \mathcal{C}_{libre}$ ). Dans le cas contraire les obstacles dans l’espace  $\mathcal{W}$  deviennent des  $\mathcal{C}$ -obstacles dans

l'espace  $\mathcal{C}$ . Le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  dans  $\mathcal{C}\Gamma$  correspondant à l'obstacle  $\mathcal{B}_i$  dans  $\mathcal{W}\Gamma$  est composé des configurations  $q$  de  $\mathcal{A}$  dont  $\mathcal{A}(q)$  touche ou intersecte l'obstacle  $\mathcal{B}_i$  (équations 2.1 et 2.2). Les configurations définissant la frontière de  $\mathcal{CB}_i\Gamma$  notée  $\partial\mathcal{CB}_i\Gamma$  correspondent aux configurations  $q$  dont  $\mathcal{A}(q)$  touche l'obstacle  $\mathcal{B}_i$  sans le pénétrer. L'ensemble des configurations de  $\partial\mathcal{CB}_i$  fait part de  $\mathcal{C}_{contact}$ .

### 5.1.1 La construction des $\mathcal{C}$ -obstacles

D'après l'équation 2.2  $\Gamma$  un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  dans  $\mathcal{C}$  est défini comme l'union des  $\mathcal{C}$ -obstacles  $\mathcal{CB}_{i,\theta}$  pour chaque  $\theta \in [0, 2\pi]$  où  $\mathcal{CB}_{i,\theta}$  correspond au grossissement de l'obstacle  $\mathcal{B}_i$  avec le robot  $\mathcal{A}$  fixe à l'orientation  $\theta$  (figure 2.3). Le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_{i,\theta}$  correspond donc à la tranche du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  sur le plan  $\Theta = \theta$  de l'espace  $\mathcal{C} = \mathbb{R}^2 \times [0, 2\pi]$ . Une construction continue d'un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  n'est en général pas possible. Pour cette raison  $\Gamma$  une représentation discrète de  $\mathcal{CB}_i$  sera construite incrémentalement à partir des tranches  $\mathcal{CB}_{i,\theta}$ . En effet  $\Gamma$  les tranches  $\mathcal{CB}_{i,\theta}$  seront construites au fur et à mesure que le robot ponctuel  $q_{\mathcal{A}}$  évolue dans  $\mathcal{C}_{valide}$  pour des orientations déterminées soit par la fonction de potentiel  $\Gamma$  soit par la stratégie de traitement des minima locaux.

Le contour  $\partial\mathcal{CB}_{i,\theta}$  peut être construit en appliquant un algorithme proposé originalement par Lozano-Pérez [Loz83] et basé sur la différence de Minkowski. Dans la suite  $\Gamma$  l'algorithme de construction du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_{i,\theta}$  est expliqué dans le cas d'un obstacle  $\mathcal{B}_i$  convexe et d'un robot  $\mathcal{A}$  convexe. Ensuite  $\Gamma$  l'algorithme est étendu pour le cas des obstacles concaves.

#### 5.1.1.1 Construction du $\mathcal{C}$ -obstacle $\mathcal{CB}_{i,\theta}$ avec $\mathcal{B}_i$ convexe et $\mathcal{A}$ convexe

Soit un obstacle  $\mathcal{B}_i$  convexe et le robot  $\mathcal{A}$  convexe fixe à l'orientation  $\theta$ .  $\partial\mathcal{CB}_{i,\theta}$  est construit en appliquant l'algorithme suivant [Loz83  $\Gamma$  Lat91] (figure 5.1) :

1. Placer les vecteurs normaux extérieurs  $\vec{n}_k^{\mathcal{B}_i}$  de  $\mathcal{B}_i$  dans le cercle unitaire  $C^1$ .
2. Placer les vecteurs normaux intérieurs  $-\vec{n}_j^{\mathcal{A}}$  de  $\mathcal{A}$  dans le cercle unitaire  $C^1$ .
3. Parcourir les vecteurs dans le cercle  $C^1$  dans le sens trigonométrique en partant du vecteur ayant l'angle positif le plus petit par rapport au vecteur unitaire canonique  $\vec{x}$ . Pour chaque vecteur  $\vec{n}$  dans  $C^1$  :
  - si  $\vec{n}$  appartient à  $\mathcal{A}$  donc  $\vec{n} = -\vec{n}_j^{\mathcal{A}}$ . Supposons que le vecteur  $\vec{n}$  soit différent à tout vecteur  $\vec{n}_k^{\mathcal{B}_i}$   $\Gamma$  c'est-à-dire que  $\vec{n}$  est placé dans  $C^1$  entre deux vecteurs  $\vec{n}_{k-1}^{\mathcal{B}_i}$  et  $\vec{n}_k^{\mathcal{B}_i}$ . Ceci va définir une arête de  $\partial\mathcal{CB}_{i,\theta}$  entre les sommets définis par  $s_k^{\mathcal{B}_i} - s_j^{\mathcal{A}}$  et  $s_k^{\mathcal{B}_i} - s_{j+1}^{\mathcal{A}}$ . Le long de cette arête un contact simple de type arête- $\mathcal{A}$ -sommet- $\mathcal{B}_i$  est défini.
  - si  $\vec{n}$  appartient à  $\mathcal{B}_i$   $\Gamma$  donc  $\vec{n} = \vec{n}_k^{\mathcal{B}_i}$ . Supposons que le vecteur  $\vec{n}$  soit différent à tout vecteur  $\vec{n}_j^{\mathcal{A}}$   $\Gamma$  c'est-à-dire que  $\vec{n}$  est placé dans  $C^1$  entre deux vecteurs

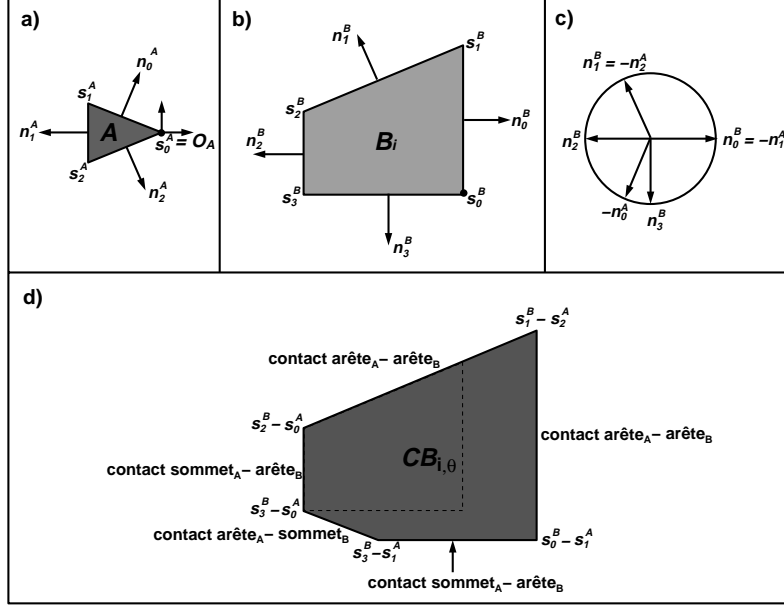


FIG. 5.1 – a) Le robot  $\mathcal{A}$  convexe fixe d'orientation  $\theta$ .  
 b) Un obstacle  $\mathcal{B}_i$  convexe.  
 c) Les vecteurs normaux extérieurs de  $\mathcal{B}_i$  et intérieurs de  $\mathcal{A}$  dans le cercle unitaire  $C^1$ .  
 d) La tranche  $\mathcal{CB}_{i,\theta}$  du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  construite comme le parcours du point de référence  $\mathcal{O}_A$  du robot  $\mathcal{A}$  autour de l'obstacle  $\mathcal{B}_i$ .

$\vec{n}_{j-1}^A$  et  $\vec{n}_j^A$ . Ceci va définir une arête de  $\partial\mathcal{CB}_{i,\theta}$  entre les sommets définis par  $s_k^{\mathcal{B}_i} - s_j^{\mathcal{A}}$  et  $s_{k+1}^{\mathcal{B}_i} - s_j^{\mathcal{A}}$ . Le long de cette arête un contact simple de type sommet $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$  est défini.

- il ne reste que le cas où deux vecteurs  $-\vec{n}_j^A$  et  $\vec{n}_k^{\mathcal{B}_i}$  sont égaux. Ceci va définir une arête de  $\partial\mathcal{CB}_{i,\theta}$  entre les sommets définis par  $s_k^{\mathcal{B}_i} - s_j^{\mathcal{A}}$  et  $s_{k+1}^{\mathcal{B}_i} - s_{j+1}^{\mathcal{A}}$ . Le long de cette arête un contact simple de type arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$  est défini.

Le contour  $\partial\mathcal{CB}_{i,\theta}$  résultat est un polygone simple et convexe. Le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_{i,\theta}$  est défini par le contour  $\partial\mathcal{CB}_{i,\theta}$  placé sur le plan  $\Theta = \theta$  de l'espace  $\mathbb{R}^2 \times [0, 2\pi]$ .

### 5.1.1.2 Construction du $\mathcal{C}$ -obstacle $\mathcal{CB}_{i,\theta}$ avec $\mathcal{B}_i$ concave et $\mathcal{A}$ convexe

Dans notre cas l'un obstacle  $\mathcal{B}_i$  concave est défini comme une composition de polygones convexes disjoints  $\mathcal{B}_i = \{b_1, \dots, b_n\}$  où chaque composante  $b_i$  est en contact avec au moins une autre composante  $b_j$ . Le contact est défini entre les contours de  $b_i$  et  $b_j$  (i.e.  $b_i \cap b_j \neq \emptyset$  et  $\text{int}(b_i) \cap \text{int}(b_j) = \emptyset$ ).

La construction de  $\partial\mathcal{CB}_{i,\theta}$  est obtenue comme l'union des polygones  $\partial\mathcal{C}b_{i,\theta}$  convexes qui résultent du grossissement de chaque composante  $b_i$  convexe avec le robot  $\mathcal{A}$  convexe fixe à l'orientation  $\theta$  (figure 5.2). Le contour  $\partial\mathcal{CB}_{i,\theta}$  résultat est un polygone simple.

Les sommets en  $\partial\mathcal{CB}_{i,\theta}$  qui résultent de l'intersection entre deux composantes  $\partial\mathcal{C}b_{i,\theta}$  et  $\partial\mathcal{C}b_{j,\theta}$  définissent un *contact double* entre  $\mathcal{A}\Gamma b_i$  et  $b_j$ . De la même manière les segments d'arête communs à  $\partial\mathcal{C}b_{i,\theta}$  et  $\partial\mathcal{C}b_{j,\theta}$  définissent un *contact double* entre  $\mathcal{A}\Gamma b_i$  et  $b_j$ . Le *C-obstacle*  $\mathcal{CB}_{i,\theta}$  est défini par le contour  $\partial\mathcal{CB}_{i,\theta}$  placé sur le plan  $\Theta = \theta$  de l'espace  $\mathbb{R}^2 \times [0, 2\pi]$ .

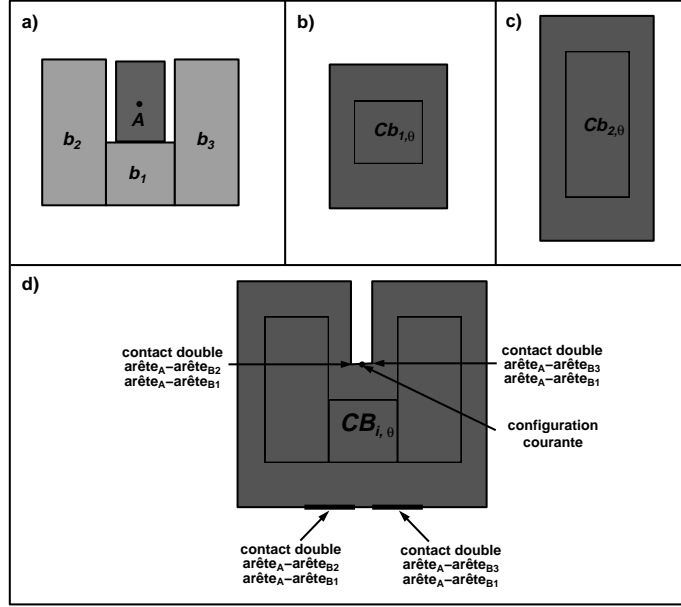


FIG. 5.2 – a) Un obstacle concave défini par  $b_1, b_2$  et  $b_3$  et le robot  $\mathcal{A}$  convexe fixe à l'orientation  $\theta$ .

b) Géométrie du *C-obstacle*  $\mathcal{C}b_{1,\theta}$  définie autour de  $b_1$ .

c) Géométrie du *C-obstacle*  $\mathcal{C}b_{2,\theta}$  définie autour de  $b_2$ . La géométrie du *C-obstacle*  $\mathcal{C}b_{3,\theta}$  est similaire à celle de  $\mathcal{C}b_{2,\theta}$ .

d) Le *C-obstacle*  $\mathcal{CB}_{i,\theta}$  construit comme l'union des contours  $\partial\mathcal{C}b_{1,\theta}, \partial\mathcal{C}b_{2,\theta}$  et  $\partial\mathcal{C}b_{3,\theta}$ .

### 5.1.1.3 Information d'“alignement” dans les *C-obstacles* $\mathcal{CB}_{i,\theta}$

L'algorithme de construction du contour  $\partial\mathcal{CB}_{i,\theta}$  associe à chacune des arêtes un type de contact en accord avec l'interaction du robot  $\mathcal{A}$  et de l'obstacle  $\mathcal{B}_i$  qu'elle représente. Selon les géométries du robot et des obstacles considérées un contact peut être simple ou double (§3.3). Les types de contacts simples traités sont : arête $_{\mathcal{A}}$ -sommet $_{\mathcal{B}_i}$ , sommet $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$  et arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$ . Puisque les contacts de type arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$  sont réducteurs d'incertitude d'orientation on va inclure dans chaque arête  $\vec{E}_j^{\mathcal{CB}_{i,\theta}}$  de  $\partial\mathcal{CB}_{i,\theta}$  représentant un contact arête $_{\mathcal{A}}$ -sommet $_{\mathcal{B}_i}$  ou sommet $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$  les angles positif et négatif ( $\varphi_j^+$  et  $\varphi_j^-$  respectivement) ainsi que le point de rotation  $\varphi_j^{\circ}$  sur l'obstacle  $\mathcal{B}_i$  afin de transformer ce contact en un contact arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$ . L'angle  $\varphi_j^+$  détermine une rotation du robot autour de  $\varphi_j^{\circ}$  dans le sens trigonométrique tandis que l'angle  $\varphi_j^-$  détermine une rotation du robot autour de  $\varphi_j^{\circ}$  dans le sens trigonométrique inverse.

Ainsi lorsque le robot ponctuel  $q_A$  atteint une arête de  $\partial\mathcal{CB}_{i,\theta}$  correspondant à un contact simple de type arête $_A$ -sommet $_{B_i}$  ou sommet $_A$ -arête $_{B_i}$ , une stratégie de rotation pourra être appliquée pour passer à un contact de type arête $_A$ -arête $_{B_i}$ . L'incertitude d'orientation sera donc annulée. Les paramètres  $\varphi_l^+ \Gamma \varphi_l^-$  et  $\varphi_l^\circ$  associés à l'arête  $\vec{E}_l^{\mathcal{CB}_{i,\theta}}$  de  $\partial\mathcal{CB}_{i,\theta}$  peuvent être déterminés de la manière suivante :

- **cas de contact arête $_A$ -sommet $_{B_i}$**  : supposons que le contact soit défini entre une arête  $\vec{E}_j^A$  de  $\mathcal{A}$  et un sommet  $s_k^{B_i}$  de  $\mathcal{B}_i$  (figure 5.3a) :

$$\begin{aligned}\varphi_l^+ &= \arccos(-\vec{n}_j^A \cdot \vec{n}_k^{B_i}), \\ \varphi_l^- &= -\arccos(-\vec{n}_j^A \cdot \vec{n}_{k-1}^{B_i}), \\ \varphi_l^\circ &= s_k^{B_i}\end{aligned}\quad (5.1)$$

- **cas de contact sommet $_A$ -arête $_{B_i}$**  : supposons que le contact soit défini entre un sommet  $s_j^A$  de  $\mathcal{A}$  et une arête  $\vec{E}_k^{B_i}$  de  $\mathcal{B}_i$  (figure 5.3b) :

$$\begin{aligned}\varphi_l^+ &= \arccos(-\vec{n}_{j-1}^A \cdot \vec{n}_k^{B_i}), \\ \varphi_l^- &= -\arccos(-\vec{n}_j^A \cdot \vec{n}_k^{B_i}), \\ \varphi_l^\circ &= s_j^A\end{aligned}\quad (5.2)$$

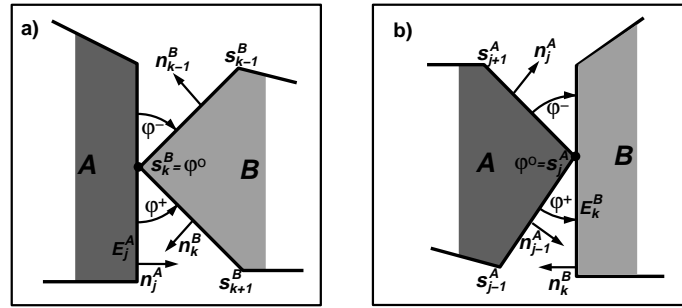


FIG. 5.3 – a) Paramètres d’“alignement” pour un contact simple de type arête $_A$ -sommet $_{B_i}$ . b) Paramètres d’“alignement” pour un contact simple de type sommet $_A$ -arête $_{B_i}$ .

En ce qui concerne les contacts doubles  $\Gamma$  nous avons vu qu’ils sont représentés soit comme un sommet d’une concavité d’un contour  $\partial\mathcal{CB}_{i,\theta}$  soit comme un intervalle à l’intérieur d’une arête d’un contour  $\partial\mathcal{CB}_{i,\theta}$  (figure 5.2d). Dans le premier cas  $\Gamma$  l’incertitude de position est annulée tandis que dans le deuxième cas  $\Gamma$  c’est l’incertitude d’orientation qui est annulée. Dans aucun cas  $\Gamma$  la stratégie de rotation ne sera appliquée car le contact pour lui-même est réducteur d’incertitude.



## 5.2 La fonction de potentiel fictif

### 5.2.1 Le principe

Comme nous l'avons expliqué dans §4.1.1 une méthode de potentiel fictif engendre des déplacements incrémentaux suivant le gradient induit par une fonction de potentiel sur la configuration courante du robot ponctuel (particule)  $q_A$ . Dans le cas classique il suffit de définir une fonction continue et dérivable dans l'espace  $\mathcal{C}_{libre}$  avec (1) un minimum global à la configuration but et (2) des surfaces asymptotiques autour des surfaces des  $\mathcal{C}$ -obstacles (figure 4.1). En appliquant cette méthode le robot s'approche incrémentalement du but en restant dans l'espace  $\mathcal{C}_{libre}$ .

La fonction de potentiel fictif dans notre cas est basée sur un critère de distance entre la configuration courante  $q_c = (x_c, y_c, \theta_c)$  et la configuration but  $q_f = (x_f, y_f, \theta_f)$ . Puisque la métrique des coordonnées dans une configuration n'est pas la même nous avons décidé de définir une distance  $d$ - $\theta$  composée par la distance euclidienne entre les positions et la distance angulaire entre les orientations (équation 5.5).

Soient  $p_c = (x_c, y_c)$  et  $p_f = (x_f, y_f)$  dans  $\mathbb{R}^2$  les positions courante et but respectivement la fonction  $\delta d(p_c, p_f)$  retourne la distance euclidienne entre  $p_c$  et  $p_f$  i.e. :

$$\begin{aligned} \delta d : \mathbb{R}^2 \times \mathbb{R}^2 &\rightarrow \mathbb{R}^+ \\ (p_c, p_f) &\rightarrow \sqrt{(x_f - x_c)^2 + (y_f - y_c)^2} \end{aligned} \quad (5.3)$$

Cette distance sera notée  $\|p_f - p_c\|$ .

Soient  $\theta_c$  et  $\theta_f$  dans  $[0, 2\pi]$  les orientations courante et but respectivement la fonction  $\delta\theta(\theta_c, \theta_f)$  retourne la distance angulaire entre  $\theta_c$  et  $\theta_f$  i.e. :

$$\begin{aligned} \delta\theta : [0, 2\pi] \times [0, 2\pi] &\rightarrow [-\pi, \pi] \\ (\theta_c, \theta_f) &\rightarrow \begin{cases} \theta_f - \theta_c & \text{si } |\theta_f - \theta_c| \leq \pi, \\ \theta_f - \theta_c - 2\pi & \text{si } \theta_f - \theta_c > \pi, \\ \theta_f - \theta_c + 2\pi & \text{si } \theta_f - \theta_c < -\pi \end{cases} \end{aligned} \quad (5.4)$$

La distance  $d$ - $\theta$  entre les configurations courante  $q_c$  et but  $q_f$  est donc exprimée par :

$$\begin{aligned} d\text{-}\theta : \mathcal{C} \times \mathcal{C} &\rightarrow \mathbb{R}^+ \times [0, \pi] \\ (q_c, q_f) &\rightarrow (\delta d(p_c, p_f), |\delta\theta(\theta_c, \theta_f)|) \end{aligned} \quad (5.5)$$

Par définition le minimum de  $\delta d(p_c, p_f)$  est atteint lorsque  $p_c$  devient  $p_f$  et celui de  $|\delta\theta(\theta_c, \theta_f)|$  est atteint lorsque  $\theta_c$  devient  $\theta_f$ . Donc la distance  $d$ - $\theta(q_c, q_f)$  atteint son minimum (la valeur zéro) lorsque  $q_c$  devient  $q_f$ . Ceci montre que la distance  $d$ - $\theta$  n'admet qu'un seul minimum à la configuration but  $q_f$ . Le critère de distance  $d$ - $\theta$  sera donc suffisant pour faire naviguer le robot dans l'espace  $\mathcal{C}_{libre}$ . Cependant ce critère

doit être modifié pour le faire naviguer dans l'espace  $\mathcal{C}_{contact}$ .

Dans la suite le comportement de la fonction de potentiel est présenté pour chaque espace d'évolution du robot ponctuel  $q_A$ . Tout d'abord la navigation dans l'espace  $\mathcal{C}_{libre}$  est expliquée. Ensuite le cas des transitions de l'espace  $\mathcal{C}_{libre}$  vers l'espace  $\mathcal{C}_{contact}$  est considéré. Puis la navigation dans l'espace  $\mathcal{C}_{contact}$  est traitée. Finalement un algorithme intégrant les différents comportements de la fonction de potentiel est donné.

## 5.2.2 Comportement du potentiel fictif dans $\mathcal{C}_{libre}$

### 5.2.2.1 Définition

#### Définition 5.1 - Configuration libre-sûre

Soit la configuration  $q_j \in \mathcal{C}_{libre}$  dont le cylindre d'incertitude autour de celle-ci est défini par  $C(q_j, \rho_j, \psi_j)$  (cf. équation 2.6). La configuration  $q_j$  est une *configuration libre-sûre* si le cylindre d'incertitude de position/orientation  $C(q_j, \rho_j, \psi_j)$  n'intersecte pas les  $\mathcal{C}_{obstacles}$  donc  $C(q_j, \rho_j, \psi_j) \subset \mathcal{C}_{libre}$ .

### 5.2.2.2 Déplacements en translation

Le déplacement en translation du robot ponctuel  $q_A$  est défini par un vecteur  $\vec{\Delta}q_t$  permettant de réduire la distance euclidienne entre la position courante  $p_c$  et la position but  $p_f$ . La longueur de ce vecteur est constante et de valeur  $\epsilon_t$  sauf si  $p_c$  se trouve à une distance inférieure de  $\epsilon_t$  de  $p_f$ .  $\vec{\Delta}q_t$  est défini de la manière suivante :

$$\begin{aligned} \vec{\Delta}q_t : \mathbb{R}^2 \times \mathbb{R}^2 &\rightarrow \mathbb{R}^2 \\ (p_c, p_f) &\rightarrow \begin{cases} \epsilon_t * \frac{p_f - p_c}{\|p_f - p_c\|} & \text{si } \|p_f - p_c\| \geq \epsilon_t \\ p_f - p_c & \text{si } \|p_f - p_c\| < \epsilon_t \end{cases} \end{aligned} \quad (5.6)$$

La constante  $\epsilon_t$  sera appelée le *pas de discrétisation en translation*.

### 5.2.2.3 Déplacements en rotation

L'approche du robot ponctuel  $q_A$  vers la configuration but se fait de manière homogène en translation et en rotation. Pour cela un *pas de discrétisation en rotation*  $\epsilon_r$  est défini en fonction d'une part de la distance en translation de la position but et d'autre part du pas de discrétisation en translation :

$$\begin{aligned} \epsilon_r : \mathbb{R}^2 \times \mathbb{R}^2 &\rightarrow [0, 1] \\ (p_c, p_f) &\rightarrow \begin{cases} \frac{\epsilon_t}{\|p_f - p_c\|} & \text{si } \|p_f - p_c\| \geq \epsilon_t \\ 1 & \text{si } \|p_f - p_c\| < \epsilon_t \end{cases} \end{aligned} \quad (5.7)$$

La valeur de  $\epsilon_r(p_c, p_f)$  garantit que le robot va tourner selon un déplacement angulaire  $\Delta q_\theta$  uniformément tout au long de son déplacement entre les positions  $p_c$  et

$p_f$ . Le déplacement angulaire  $\Delta q_\theta \Gamma$  pour sa part  $\Gamma$  réduit la distance angulaire entre l'orientation courante  $\theta_c$  et l'orientation but  $\theta_f$  :

$$\begin{aligned} \Delta q_\theta &: \mathcal{C} \times \mathcal{C} \rightarrow [-\pi, \pi] \\ (q_c, q_f) &\rightarrow \epsilon_r(p_c, p_f) * \delta\theta(\theta_c, \theta_f) \end{aligned} \quad (5.8)$$

#### 5.2.2.4 Composition d'un déplacement en translation/rotation

Dans  $\mathcal{C}_{libre} \Gamma$  la fonction de potentiel est évaluée itérativement sur la configuration courante  $q_c$  du robot ponctuel  $q_A$ . A chaque itération  $\Gamma$  un déplacement  $\vec{\Delta}q$  est induit pour permettre à  $q_A$  d'approcher la configuration courante  $q_c$  de la configuration but  $q_f$ . Cette approche se fait de façon homogène en translation et en rotation. Le déplacement  $\vec{\Delta}q$  est défini de la manière suivante :

$$\begin{aligned} \vec{\Delta}q &: \mathcal{C} \times \mathcal{C} \rightarrow \mathbb{R}^2 \times [-\pi, \pi] \\ (q_c, q_f) &\rightarrow (\vec{\Delta}q_t(p_c, p_f), \Delta q_\theta(q_c, q_f)) \end{aligned} \quad (5.9)$$

Soit  $\delta d_0$  (resp.  $\delta\theta_0$ ) la distance euclidienne en position (resp. la distance angulaire en orientation) entre  $q_c$  et  $q_f$ . La figure 5.4 représente la relation induite par le déplacement  $\vec{\Delta}q(q_c, q_f)$  entre les distances  $\delta d$  et  $|\delta\theta|$ .

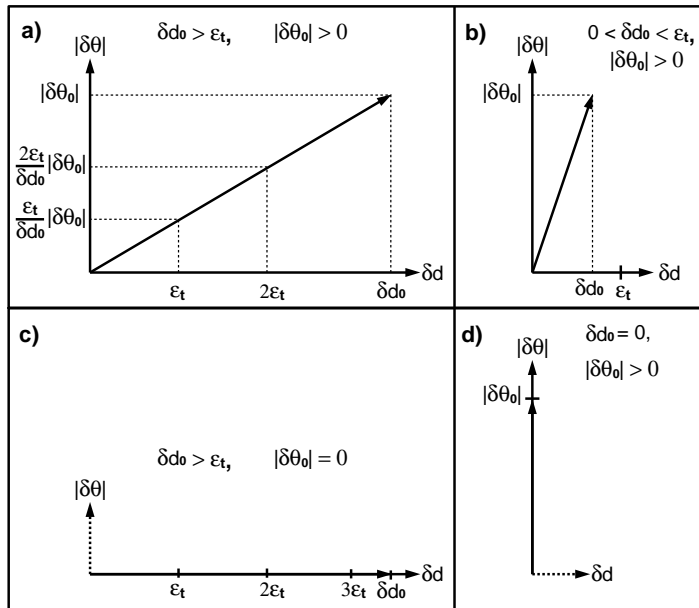


FIG. 5.4 – a) et b) La distance euclidienne  $\delta d$  et la distance angulaire  $|\delta\theta|$  sont réduites linéairement selon la pente  $m = \frac{|\delta\theta_0|}{\delta d_0}$ . c) La distance euclidienne  $\delta d$  est réduite de façon constante lorsque la distance angulaire  $|\delta\theta|$  est nulle (i.e.  $m = 0$ ). d) La distance angulaire  $|\delta\theta|$  est réduite en un seul pas lorsque la distance euclidienne  $\delta d$  est nulle. (i.e.  $m = +\infty$ ).

Soit  $\vec{\Delta}q(q_c, q_f) = (\delta_x, \delta_y, \delta_\theta)$  le déplacement induit sur la configuration  $q_c$ . La configuration courante  $q'_c$  après avoir effectué le déplacement  $\vec{\Delta}q$  est définie de la manière suivante :

$$\begin{aligned} \hat{\dagger} & : \mathcal{C} \times (\mathbb{R}^2 \times [-\pi, \pi]) \rightarrow \mathcal{C} \\ & (q_c, \vec{\Delta}q) \rightarrow q'_c \end{aligned} \quad (5.10)$$

où

$$\begin{aligned} x'_c &= x_c + \delta_x; \\ y'_c &= y_c + \delta_y; \\ \theta'_c &= \begin{cases} \theta_c + \delta_\theta & \text{si } 0 \leq \theta_c + \delta_\theta < 2\pi, \\ \theta_c + \delta_\theta - 2\pi & \text{si } \theta_c + \delta_\theta \geq 2\pi, \\ \theta_c + \delta_\theta + 2\pi & \text{si } \theta_c + \delta_\theta < 0 \end{cases} \end{aligned}$$

Par souci de simplicité on note  $q'_c = q_c + \vec{\Delta}q$ .

### 5.2.2.5 Déplacement en translation/rotation sous des contraintes d'incertitudes géométriques

Soit  $q_c$  une configuration libre-sûre dont les valeurs d'incertitude de position et d'orientation sont données par  $\rho_c$  et  $\psi_c$  respectivement. Le déplacement  $\vec{\Delta}q(q_c, q_f)$  induit par la fonction de potentiel amène le robot ponctuel  $q_A$  vers la configuration  $q'_c$  (cf. équation 5.10).

Supposons  $q'_c \in \mathcal{C}_{libre}$ . La contrainte d'incertitude en contrôle  $\eta$  (§2.3.2) et celle en orientation  $k_r$  (§2.3.3) imposées sur les déplacements en translation et en rotation ont comme effet l'augmentation de l'incertitude de position et d'orientation liées à la configuration  $q'_c$ . Les incréments sur les incertitudes sont exprimés comme :

$$\begin{aligned} \delta\rho &= \|\vec{\Delta}q_t(p_c, p_f)\| * \tan(\eta) \quad (\text{cf. équation 2.4}), \\ \delta\psi &= |\Delta q_\theta(q_c, q_f)| * k_r \quad (\text{cf. équation 2.5}) \end{aligned} \quad (5.11)$$

Les nouvelles valeurs d'incertitude de position et d'orientation sur la configuration courante  $q'_c$  seront donc :

$$\begin{aligned} \rho_{c'} &= \rho_c + \delta\rho, \\ \psi_{c'} &= \psi_c + \delta\psi \end{aligned} \quad (5.12)$$

Ces paramètres définissent le cylindre d'incertitude  $C(q'_c, \rho_{c'}, \psi_{c'})$  autour de  $q'_c$ . La configuration  $q'_c$  sera retenue par le potentiel si d'une part le robot ponctuel  $q_A$  ne trouve pas de collisions le long du déplacement  $\vec{\Delta}q(q_c, q_f)$  avec les  $\mathcal{C}$ -obstacles et si d'autre part la configuration  $q'_c$  est une configuration libre-sûre. Dans le cas contraire un passage de l'espace  $\mathcal{C}_{libre}$  vers l'espace  $\mathcal{C}_{contact}$  doit être évalué.

En résumé dans l'espace  $\mathcal{C}_{libre}$  la fonction de potentiel fictif engendre un segment de droite composé de sous-segments dont les extrémités sont des configurations libres-sûres. L'extrémité initiale du segment peut être soit une configuration libre-sûre soit

une configuration de contact. La fonction de potentiel cherche à prolonger ce segment jusqu'à la configuration but  $q_f$  (figure 5.5a). Cependant il se peut qu'avant d'y arriver un contact soit détecté (figure 5.5b).

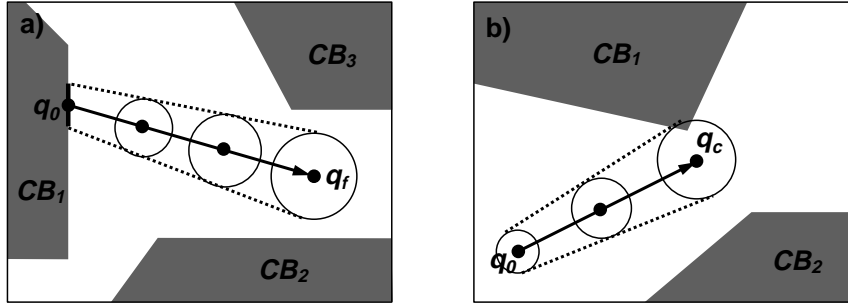


FIG. 5.5 – a) Séquence de déplacements dans  $\mathcal{C}_{libre}$  arrivant à une configuration  $q_f$  libre-sûre. b) Séquence de déplacements dans  $\mathcal{C}_{libre}$  touchant au  $\mathcal{C}$ -obstacle  $CB_1$ .

### 5.2.3 Passages de $\mathcal{C}_{libre}$ vers $\mathcal{C}_{contact}$

Lorsque le robot ponctuel  $q_A$  navigue dans l'espace  $\mathcal{C}_{libre}$  deux tests de collisions sont appliqués pour vérifier les intersections potentielles avec les  $\mathcal{C}$ -obstacles : tout d'abord un test le long du déplacement nominal  $\bar{\Delta}q$  engendré par la fonction de potentiel puis un test avec l'enveloppe d'incertitude liée à la configuration atteinte  $q'_c$  après avoir effectué le déplacement. La procédure de traitement des collisions peut être : soit tenter de corriger le déplacement nominal pour que lui et son enveloppe d'incertitude n'intersectent pas les  $\mathcal{C}$ -obstacles soit permettre le contact et ensuite continuer les déplacements dans l'espace  $\mathcal{C}_{contact}$ .

Les avantages des contacts sont que d'une part les incertitudes de position et/ou d'orientation liées au robot peuvent être réduites et d'autre part les déplacements suivis par le robot approchent mieux les déplacements engendrés par la fonction de potentiel. Ainsi le robot ponctuel  $q_A$  s'approche de la configuration but en réduisant ses incertitudes géométriques.

Les inconvénients des contacts sont que d'une part le robot perd certains degrés de liberté et d'autre part l'intégrité physique du robot est fortement compromise. Ces inconvénients peuvent disparaître si l'on dispose de (1) un système de contrôle fiable et performant sur un ensemble contraint de déplacements en contact et (2) des stratégies appropriées de raisonnement sur les contacts.

Notre fonction de potentiel permet que le robot ponctuel  $q_A$  entre en contact avec les  $\mathcal{C}$ -obstacles sous la condition d'"atteignabilité".

### 5.2.3.1 Condition d’“atteignabilité” d’un contact

Un déplacement  $\vec{\Delta}q$  dans l’espace  $\mathcal{C}_{libre}$  guidant le robot ponctuel  $q_A$  vers un contact potentiel avec un (ou plusieurs)  $\mathcal{C}\text{-obstacle}(s)$  n’est retenu par la fonction de potentiel que si la configuration nominale de contact  $q_c$  est à l’intérieur de la “région sûre”  $R$  d’une surface visible  $S$  touchée. La région  $R$  d’une surface  $S$  d’un  $\mathcal{C}\text{-obstacle}$   $\mathcal{CB}_i$  est déterminée à l’aide des modèles *projection en avant* définis dans §2.4.2. Les constructions de tels modèles évaluent l’“atteignabilité” d’une surface de contact visible depuis la dernière configuration “atteignable” dans l’espace  $\mathcal{C}_{contact}$ . La configuration initiale  $q_0$  d’un problème de planification est aussi considérée comme une configuration “atteignable”.

Si l’on ne considère que des déplacements dont l’incertitude d’orientation est nulle il suffit d’utiliser le modèle bidimensionnel de *projection en avant* (§2.4.2.1). L’enveloppe d’incertitude considérée par ce modèle correspond au disque d’incertitude de position. Mais si l’estimation de l’incertitude d’orientation n’est pas nulle le long des déplacements le modèle tridimensionnel de *projection en avant* doit donc être utilisé (§2.4.3). Ce modèle considère comme enveloppe d’incertitude le cylindre d’incertitude en position/orientation. Dans les deux cas si la surface visible  $S$  est “atteignable” la région  $R$  des configurations sûres de  $S$  définit un segment dont la coordonnée  $\theta$  est constante (figures 2.11 et 2.15). Mais dans le cas où  $S$  est “non-atteignable” la région  $R$  est indéfinie.

Une analyse basée sur l’appartenance de la configuration nominale de contact  $q_c$  dans la “région sûre”  $R$  calculée pour la surface de contact  $S$  détermine si  $q_c$  est retenue par la fonction de potentiel. Ainsi :

- cas 1. Région  $R$  définie et  $q_c$  appartenant à  $R$ .  
Dans ce cas la fonction de potentiel accepte la configuration  $q_c$  et son comportement change pour naviguer dans l’espace  $\mathcal{C}_{contact}$  (figure 5.6a).
- cas 2. Région  $R$  définie et  $q_c$  n’appartenant pas à  $R$ .  
Dans ce cas la fonction de potentiel change temporairement de configuration but pour tenter d’atteindre une configuration  $q'_c$  à l’intérieur de la région  $R$ . Cette nouvelle configuration but  $q'_c$  entraîne une nouvelle génération de déplacements depuis la dernière configuration “atteignable” dans  $\mathcal{C}_{contact}$  ( $q_0$  est considérée aussi comme “atteignable”). La recherche de ce nouveau but peut aboutir à un succès ou à un échec. Dans les deux cas la configuration but originale  $q_f$  est récupérée. Si la configuration de contact  $q'_c$  est atteinte alors celle-ci est acceptée par la fonction de potentiel et le comportement de la fonction change pour naviguer dans l’espace  $\mathcal{C}_{contact}$  (figure 5.6b). Dans le cas contraire la dernière configuration “atteignable” est considérée comme un minimum local pour la fonction de potentiel.
- cas 3. Région  $R$  indéfinie.  
La surface  $S$  n’est pas “atteignable” donc la configuration  $q_c$  n’est pas “attei-

nable” non plus (figure 5.6c). La dernière configuration “atteignable” est donc considérée comme un minimum local pour la fonction de potentiel.

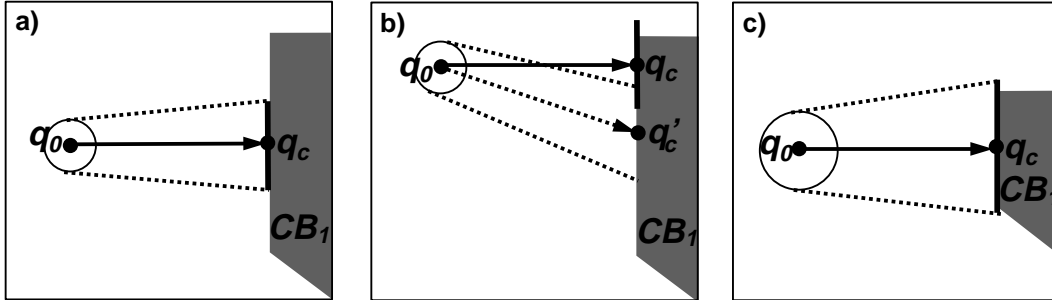


FIG. 5.6 – a) Configuration nominale de contact  $q_c$  satisfaisant la condition d’“atteignabilité” depuis  $q_0$ . b) Correction de la configuration nominale de contact  $q_c$  pour garantir la condition d’“atteignabilité” depuis  $q_0$ . c) Surface de contact “non-atteignable” depuis  $q_0$ .

L’enveloppe d’incertitude de position/orientation liée à une configuration de contact sûre  $q_c$  résulte de la projection de son enveloppe d’incertitude estimée dans l’espace  $\mathcal{C}_{libre}$  (e.g. un cylindre  $C(q_c, \rho_c, \psi_c)$ ) sur la surface de contact  $S$  (figure 5.7).

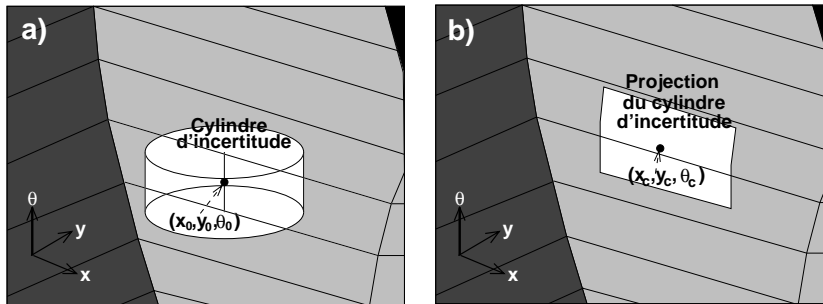


FIG. 5.7 – a) Cylindre d’incertitude de position/orientation dans l’espace des configurations  $\mathbb{R}^2 \times [0, 2\pi]$  pour une configuration nominale du robot dans l’espace  $\mathcal{C}_{libre}$ . b) Projection du cylindre d’incertitude sur la surface d’un C-obstacle dans l’espace  $\mathbb{R}^2 \times [0, 2\pi]$  pour une configuration nominale en contact avec un obstacle.

Par souci de simplicité la projection obtenue est approchée par un rectangle centré en  $q_c$  dont la base (parallèle au plan  $XY$ ) est  $2\rho_c$  et la hauteur (parallèle à l’axe  $\Theta$ ) est  $2\psi_c$ . L’enveloppe d’incertitude est donc réduite d’une dimension sur la position. La simplification faite sur la projection n’est pas très contraignante car comme il est expliqué dans la suite un déplacement d’“alignement” entre une surface du robot et la surface de contact  $S$  est appliqué après avoir atteint une configuration de contact sûre.

### 5.2.3.2 “Alignement” robot-obstacle en contact

Un déplacement de rotation  $\Delta_\theta^p$  est appliqué lorsqu’une configuration de contact  $q_c$  a validé un passage de  $\mathcal{C}_{libre}$  à  $\mathcal{C}_{contact}$  afin d’une part d’annuler l’incertitude d’orientation et d’autre part de rendre plus facile par la suite les déplacements en contact. Ce déplacement cherche à aligner une face du robot  $\mathcal{A}$  avec une face d’un obstacle  $\mathcal{B}_i$  en gardant le contact défini entre eux c’est-à-dire que l’on cherche à définir un contact simple de type arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$ . La définition d’un déplacement  $\Delta_\theta^p(q_c, p, \theta)$  implique la détermination du point de rotation (i.e. placement de l’axe de rotation)  $p$  et de l’angle de rotation  $\theta$ . Pour cela les paramètres d’“alignement” ( $\varphi_j^+$ ,  $\varphi_j^-$  et  $\varphi_j^0$  (cf. §5.1.1.3)) de la arête de contact  $\vec{E}_j^{\mathcal{CB}_i}$  courante (associée à la configuration  $q_c$ ) seront utilisés. Le point  $p$  prend la valeur du paramètre  $\varphi_j^0$  et l’angle  $\theta$  prend soit la valeur  $\varphi_j^+$  soit la valeur  $\varphi_j^-$ . Après avoir effectué la rotation  $\Delta_\theta^p(q_c, p, \theta)$  une nouvelle configuration de contact  $q'_c$  est définie. Cette configuration détermine un contact de type arête $_{\mathcal{A}}$ -arête $_{\mathcal{B}_i}$  si aucune collision n’est détectée avec d’autres obstacles (figure 5.8). En cas de collision un contact double est défini entre le robot  $\mathcal{A}$  l’obstacle initial de contact  $\mathcal{B}_i$  et le nouveau obstacle bloquant la rotation.

Après avoir effectué le déplacement de rotation  $\Delta_\theta^p(q_c, p, \theta)$  l’incertitude d’orientation estimée pour la nouvelle configuration de contact  $q'_c$  est nulle (i.e.  $\psi_{c'} = 0$ ) et l’incertitude de position reste inchangée.

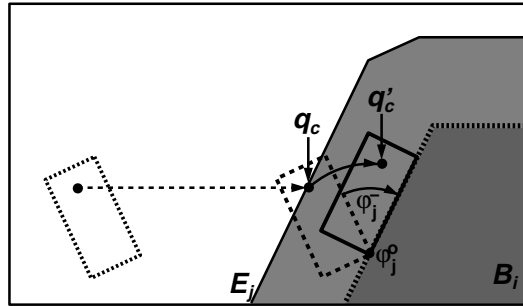


FIG. 5.8 – Déplacement d’“alignement”  $\Delta_\theta^p$  sur la configuration de contact  $q_c$  autour du point  $p = \varphi_j^0$  un angle  $\theta = \varphi_j^-$ .

### 5.2.4 Comportement du potentiel fictif dans $\mathcal{C}_{contact}$

Un contact valide entre le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  et le robot ponctuel  $q_A$  placé à la configuration nominale  $q_c = (x, y, \theta)$  a lieu sur la tranche  $\mathcal{CB}_{i,\theta}$  plus précisément sur le contour polygonal  $\partial\mathcal{CB}_{i,\theta}$ . La configuration  $q_c$  peut se trouver ainsi soit à l’intérieur soit à l’extrémité d’une arête de  $\partial\mathcal{CB}_{i,\theta}$ . Avant de détailler la génération des déplacements dans ces cas quelques définitions de base sont données. Il faut noter qu’un contact multiple à la configuration  $q_c$  définit une tranche commune aux  $\mathcal{C}$ -obstacles touchés dont la frontière peut être construite à partir de leurs tranches  $\mathcal{CB}_{i,\theta}$  respectives.



### 5.2.4.1 Définitions

#### Définition 5.2 - Sens d'un déplacement le long d'une arête orientée

Le sens d'un déplacement  $\vec{v}$  le long de une arête orientée  $\vec{E}_i^P$  est *positif* si  $\vec{v}$  suit la direction de  $\vec{E}_i^P$ . Dans le cas contraire le sens est *négatif*.

#### Définition 5.3 - Décomposition orthogonale d'un vecteur bidimensionnel

Soient  $\vec{v}, \vec{w} \in \mathbb{R}^2$  et  $\alpha \in [-\pi, +\pi]$  l'angle entre les deux vecteurs. Le vecteur  $\vec{v}$  peut être décomposé par rapport au vecteur  $\vec{w}$  en deux composantes orthogonales : une composante colinéaire  $\vec{v}^{\leftrightarrow}$  et une composante perpendiculaire  $\vec{v}^\perp$ . La décomposition garantit  $\vec{v} = \vec{v}^{\leftrightarrow} + \vec{v}^\perp$  et  $\vec{v}^{\leftrightarrow} \cdot \vec{v}^\perp = 0$  où  $+$  (resp.  $\cdot$ ) est l'opérateur d'addition (resp. le produit scalaire) de deux vecteurs.

#### Définition 5.4 - Cône de rupture de contact

Soient  $\vec{E}_i^P$  une arête orientée appartenant au polygone  $P$  dont le vecteur normal extérieur est  $\vec{n}_i^P$   $p$  une position sur l'arête  $\vec{E}_i^P$   $\vec{v}$  un vitesse dans le cercle unitaire  $C^1$  et  $\varrho \in [0, \pi/2]$  un angle.

Supposons que  $p$  soit placée entre les extrémités de  $\vec{E}_i^P$  c'est-à-dire que  $p$  n'est pas un sommet de celle-ci. L'ensemble de vitesses  $\vec{v}$  dans le cercle unitaire  $C^1$  qui sont à  $\varrho$ -près du vecteur normal extérieur  $\vec{n}_i^P$  définit un cône d'angle  $2\varrho$  dont l'axe correspond à  $\vec{n}_i^P$ . Une vitesse dans ce cône appliquée sur la position  $p$  permet d'éloigner  $p$  de l'arête  $\vec{E}_i^P$  vers son extérieur. On peut dire donc que  $p$  lâche le contact avec l'arête  $\vec{E}_i^P$ . De ce fait un tel cône est appelé *cône de rupture de contact* et  $\varrho$  *angle de rupture de contact* (figure 5.9).

Supposons maintenant que  $p$  soit une extrémité de  $\vec{E}_i^P$  donc  $p$  est une extrémité commune avec une deuxième arête  $\vec{E}_j^P$ . Le *cône de rupture de contact* dans ce cas est défini par l'ensemble de vitesses dans le cercle unitaire  $C^1$  permettant d'éloigner  $p$  de  $\vec{E}_i^P$  et de  $\vec{E}_j^P$  vers leur extérieur en même temps. Cet ensemble définit un cône dont les côtés correspondent aux côtés les plus écartés des *cônes de rupture de contact* de  $\vec{E}_i^P$  et  $\vec{E}_j^P$  (figure 5.9).

*Remarque :* la construction du *cône de rupture de contact* est semblable à celle du *cône de friction* défini par la loi de Coulomb [Lat91] mais leur fonctionnalité est opposée. En effet une vitesse  $\vec{v}$  dans le *cône de rupture de contact* appliquée sur une position  $p$  permet à cette dernière de s'éloigner de l'arête de contact tandis qu'une vitesse  $\vec{v}$  dans le *cône de friction* permet de garantir que l'application de la vitesse  $-\vec{v}$  sur  $p$  n'entraînera pas de glissement en contact.

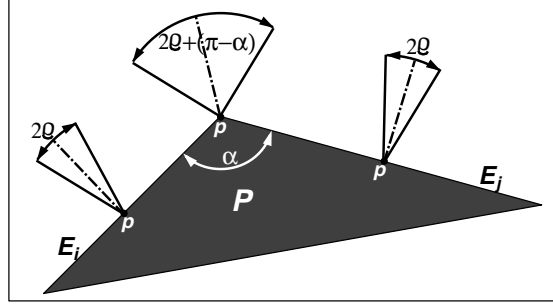


FIG. 5.9 – Définition des cônes de rupture de contact selon l'angle  $\varrho$  sur les arêtes  $\vec{E}_i^P$ ,  $\vec{E}_j^P$  et leur sommet commun. [Lat91]

#### 5.2.4.2 Configuration $q_c$ à l'intérieur d'une arête de $\mathcal{CB}_{i,\theta}$

Soit  $q_c \in \mathcal{C}_{contact}$  la configuration courante du robot ponctuel  $q_A$  définissant un contact simple. Cette situation correspond au placement du robot ponctuel  $q_A$  à l'intérieur d'une arête  $\vec{E}_j^{\mathcal{CB}_{i,\theta}} \subset \partial\mathcal{CB}_{i,\theta}$ . Si  $q_f$  est la configuration but à atteindre le meilleur déplacement incrémental pour s'approcher de  $q_f$  est  $\vec{\Delta}q(q_c, q_f)$ . Cependant ce déplacement doit être transformé en un déplacement  $\vec{\Delta}q'(q_c, q_f)$  qui tient compte de la contrainte de contact. Cette transformation est définie en fonction du déplacement  $\vec{\Delta}q(q_c, q_f)$  et du vecteur normal extérieur  $\vec{n}_j^{\mathcal{CB}_{i,\theta}}$  de l'arête de contact  $\vec{E}_j^{\mathcal{CB}_{i,\theta}}$ .

A cause des contraintes de faisabilité des mouvements et de sécurité sur le robot la fonction de potentiel ne va considérer que les déplacements en translation le long des surfaces des obstacles. Ceci revient à déterminer la composante  $\vec{\Delta}q_t$  d'un déplacement  $\vec{\Delta}q$  du robot et supposer la composante  $\Delta q_\theta$  nulle. Du point de vue géométrique la composante  $\vec{\Delta}q_t$  sur la position courante  $p_c$  donne lieu à trois cas possibles par rapport à l'arête de contact  $\vec{E}_j^{\mathcal{CB}_{i,\theta}}$  :

1.  $\vec{\Delta}q_t$  rentre dans l'arête de contact  $\Gamma$
2.  $\vec{\Delta}q_t$  pointe vers l'extérieur de l'arête de contact  $\Gamma$
3.  $\vec{\Delta}q_t$  suit l'arête de contact.

La fonction de potentiel doit donc considérer la génération du déplacement  $\vec{\Delta}q'_t$  à partir de  $\vec{\Delta}q_t$  pour ces cas. Cela est expliquée par la suite. En ce qui concerne la composante  $\Delta q'_\theta$  elle sera supposée nulle. En bref le potentiel engendrera le déplacement  $\vec{\Delta}q'(q_c, q_f) = (\vec{\Delta}q'_t(p_c, p_f), 0)$ .

Soient  $\vec{u}(\vec{v})$  le vecteur unitaire suivant la direction du vecteur  $\vec{v}$  et  $\vec{\Delta}q_t^{\leftrightarrow}$  et  $\vec{\Delta}q_t^\perp$  les vecteurs résultant de la décomposition orthogonale de  $\vec{\Delta}q_t$  par rapport à  $\vec{E}_j^{\mathcal{CB}_{i,\theta}}$ .

**Cas 1.  $\vec{\Delta}q_t$  rentre dans l'arête de contact  $\vec{E}_j^{CB_i,\theta}$** 

Ce cas vérifie la condition  $\vec{\Delta}q_t \cdot \vec{n}_j^{CB_i,\theta} < 0$  où  $\cdot$  est le produit scalaire de deux vecteurs. Cette situation se présente lorsque la position but  $p_f$  se trouve derrière l'arête de contact  $\vec{E}_j^{CB_i,\theta}$  (figure 5.10). Dans le cas général  $\Gamma$  une telle situation détermine un *mouvement compliant*  $\vec{\Delta}q'_t$  vers un sommet  $s_j^{CB_i,\theta}$  de l'arête  $\vec{E}_j^{CB_i,\theta}$ . Le sommet choisi est celui dont le déplacement depuis la position  $p_c$  approche le plus le déplacement  $\vec{\Delta}q_t$ . Le déplacement  $\vec{\Delta}q'_t$  est défini de la manière suivante :

$$\vec{\Delta}q'_t = \begin{cases} \vec{\Delta}q_t^{\leftrightarrow} & \text{si } \vec{u}(\vec{\Delta}q_t) \cdot \vec{n}_j^{CB_i,\theta} \neq -1 \quad (\text{cas vecteurs non colinéaires}), \\ \vec{0} & \text{sinon} \quad (\text{cas vecteurs colinéaires}) \end{cases} \quad (5.13)$$

La figure 5.10a illustre le premier cas. Dans le deuxième cas (vecteurs colinéaires)  $\Gamma$  un déplacement compliant approchant le robot ponctuel  $q_A$  de la position but  $p_f$  ne peut pas être généré. La configuration courante  $q_c$  devient donc un minimum local pour la fonction de potentiel (figure 5.10b).

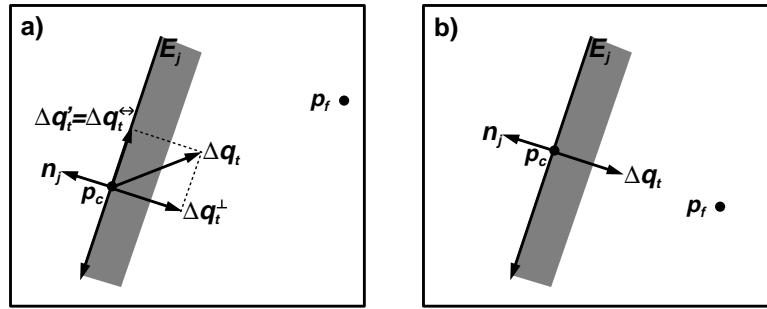


FIG. 5.10 – a) Définition du mouvement compliant  $\vec{\Delta}q'_t$  le long de l'arête de contact  $\vec{E}_j^{CB_i,\theta}$  pour approcher la position but  $p_f$ . b) Situation de minimum local pour la fonction de potentiel fictif.

En raison de l'erreur sur la précision de  $\vec{u}(\vec{\Delta}q_t) \cdot \vec{n}_j^{CB_i,\theta}$   $\Gamma$  une situation d'oscillation (mouvement de va-et-vient) peut se présenter autour d'un minimum local.

**Cas 2.  $\vec{\Delta}q_t$  pointe vers l'extérieur de l'arête de contact  $\vec{E}_j^{CB_i,\theta}$** 

Ce cas vérifie la condition  $\vec{\Delta}q_t \cdot \vec{n}_j^{CB_i,\theta} > 0$  où  $\cdot$  est le produit scalaire de deux vecteurs. Cette situation se présente lorsque le déplacement  $\vec{\Delta}q_t$  permet de lâcher le contact pour approcher  $p_c$  de  $p_f$ . Pour éviter l'augmentation de l'incertitude de position et d'orientation  $\Gamma$  un contact ne sera lâché que si  $\vec{\Delta}q_t$  et  $\vec{n}_j^{CB_i,\theta}$  sont "proches". Ceci garantit que le robot va garder le contact le plus possible pour  $\Gamma$  après  $\Gamma$ 's'éloigner de l'obstacle suivant une direction "proche" de celle de  $\vec{n}_j^{CB_i,\theta}$  (figure 5.11).

Soit  $\varrho \in [0, \pi/2]$  l'angle de rupture de contact permettant de lâcher le contact présent à la position nominale  $p_c$  sur l'arête de contact  $\vec{E}_j^{cB_i, \theta}$ . Le déplacement  $\vec{\Delta}q'_t$  est défini de la manière suivante :

$$\vec{\Delta}q'_t = \begin{cases} \vec{\Delta}q_t & \text{si } \vec{u}(\vec{\Delta}q_t) \cdot \vec{n}_j^{cB_i} \geq \cos(\varrho) \quad (\text{cas vecteurs "proches"}) \\ \vec{\Delta}q_t^{\leftrightarrow} & \text{sinon} \quad (\text{cas vecteurs "éloignés"}) \end{cases} \quad (5.14)$$

Dans le premier cas le vecteur  $\vec{\Delta}q_t$  appartient au cône de rupture de contact défini pour l'arête  $\vec{E}_j^{cB_i, \theta}$  donc le contact présent en  $p_c$  sera lâché en suivant le déplacement original  $\vec{\Delta}q_t$  (figure 5.11a). Le critère appliqué est fonction de la valeur de  $\varrho$ . Ainsi si  $\varrho = 0$  le potentiel garde le robot en contact au maximum. Par contre si  $\varrho = \pi/2$  le critère sera toujours valable grâce à la satisfaction de la condition initiale du cas étudié. Dans le deuxième cas un déplacement compliant est généré (figure 5.11b).

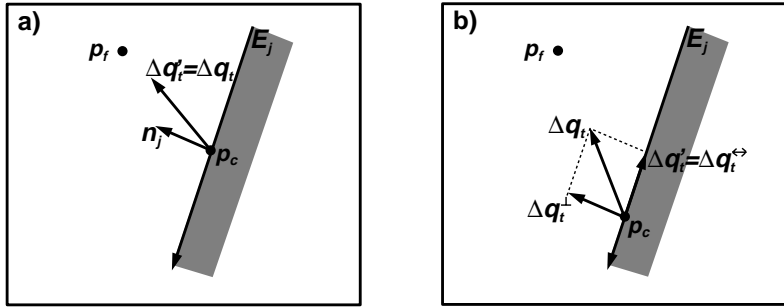


FIG. 5.11 – a) Mouvement  $\vec{\Delta}q'_t$  retenu pour approcher la position but  $p_f$  en lâchant le contact. b) Définition du mouvement compliant  $\vec{\Delta}q'_t$  le long de l'arête de contact  $\vec{E}_j^{cB_i, \theta}$  pour approcher la position but  $p_f$ .

L'erreur sur la précision de  $\vec{u}(\vec{\Delta}q_t) \cdot \vec{n}_j^{cB_i}$  et la longueur de l'intervalle  $[\cos(\varrho), 1]$  peuvent déterminer une situation d'oscillation autour d'une configuration de contact qui n'est pas du tout un minimum local. La détection d'une telle situation force la fonction de potentiel à lâcher le contact suivant le déplacement  $\vec{\Delta}q'_t = \vec{\Delta}q_t$ .

### Cas 3. $\vec{\Delta}q_t$ suit l'arête de contact $\vec{E}_j^{cB_i, \theta}$

Ce cas vérifie la condition  $\vec{\Delta}q_t \cdot \vec{n}_j^{cB_i, \theta} = 0$  où  $\cdot$  est le produit scalaire de deux vecteurs. Cette situation se présente lorsque le déplacement  $\vec{\Delta}q_t$  est colinéaire à l'arête de contact  $\vec{E}_j^{cB_i, \theta}$ . La décomposition orthogonale de  $\vec{\Delta}q_t$  obtenue donne une composante colinéaire  $\vec{\Delta}q_t^{\parallel} = \vec{\Delta}q_t$  et une composante perpendiculaire  $\vec{\Delta}q_t^{\perp}$  nulle. Le potentiel fictif génère donc le déplacement compliant  $\vec{\Delta}q'_t = \vec{\Delta}q_t$ . Ainsi le robot ponctuel  $q_A$  va se déplacer en gardant le contact (figure 5.12).

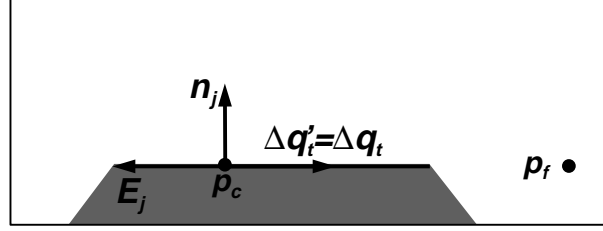


FIG. 5.12 – Déplacement compliant  $\vec{\Delta}q'_t$  approchant la position but  $p_f$ .

### Sens d'un déplacement compliant $\vec{\Delta}q'_t$

Soit  $w_z$  la composante  $z$  du produit vectoriel  $\vec{n}_j^{\mathcal{CB}_i,\theta} \wedge \vec{\Delta}q'_t$  entre les vecteurs  $\vec{n}_j^{\mathcal{CB}_i,\theta}$  et  $\vec{\Delta}q'_t$  dans  $\mathbb{R}^2$ .

Le sens d'un déplacement compliant  $\vec{\Delta}q'_t$  généré par rapport à l'arête (orientée) de contact  $\vec{E}_j^{\mathcal{CB}_i,\theta}$  est *positif* si  $w_z > 0$  et *negatif* dans le cas contraire.

### Traitement de l'incertitude

On ne considère que l'effet de l'incertitude sur la position du robot ponctuel  $q_A$  puisque l'on a supposé que le robot ne tourne pas tant qu'il est en contact avec un obstacle. Dans l'espace  $\mathcal{C}_{contact}$  la représentation de l'incertitude de position est modélisée par un intervalle centré sur la position nominale et étendu le long de l'arête de contact (§2.3.1). Cet intervalle augmente au fur et à mesure que le robot ponctuel  $q_A$  suit un mouvement compliant et l'incrément est exprimé comme :

$$\delta\rho = \|\vec{\Delta}q'_t(p_c, p_f)\| * k_t \quad (5.15)$$

avec  $k_t = \tan(\eta)$  selon une hypothèse faite dans §2.3.2. De plus sous cette hypothèse l'estimation de l'incrément de l'incertitude de position dans l'espace  $\mathcal{C}_{libre}$  (équation 5.11) et celle de l'incrément dans l'espace  $\mathcal{C}_{contact}$  (équation 5.15) deviennent semblables. Les incertitudes de position et d'orientation liées à la nouvelle configuration courante  $q'_c$  seront donc :

$$\begin{aligned} \rho_{c'} &= \rho_c + \delta\rho, \\ \psi_{c'} &= \psi_c \end{aligned} \quad (5.16)$$

Cependant il y a deux cas spéciaux qui doivent être soulignés. Le premier correspond au moment où le robot  $q_A$  lâche le contact et passe donc dans l'espace  $\mathcal{C}_{libre}$ . Pour rester cohérent avec la modélisation de l'incertitude de position un disque d'incertitude de position doit accompagner la configuration  $q'_c$ . Par souci de simplicité le rayon de ce disque sera  $\rho_{c'}$ . Le deuxième cas se présente lorsque le long d'un mouvement compliant le robot ponctuel  $q_A$  arrive à un sommet de l'arête de contact suivie. Dans cette situation le mouvement est arrêté et le robot recale sa position par rapport à la position du sommet détecté. L'incertitude de position de la configuration courante  $q'_c$  devient donc nulle (i.e.  $\rho_{c'} = 0$ ).

### Condition d'arrêt d'un déplacement compliant $\vec{\Delta}q'_t$

Si la fonction de potentiel engendre un déplacement compliant  $\vec{\Delta}q'_t$  une évaluation supplémentaire doit être faite pour garantir que la condition de contact est maintenue le long du déplacement (y compris dans le sous-intervalle d'incertitude de position autour de la configuration nominale finale  $q'_c$ ). Si tout le déplacement peut se réaliser à l'intérieur d'une même arête de contact  $\Gamma$  le déplacement initial  $\vec{\Delta}q'_t$  est retenu par le potentiel. Dans le cas contraire  $\Gamma$  le déplacement  $\vec{\Delta}q'_t$  dépasse les limites de l'arête de contact  $\vec{E}_j^{\mathcal{CB}_{i,\theta}}$ . Pour garantir la condition de contact  $\Gamma$  le déplacement  $\vec{\Delta}q'_t$  est raccourci jusqu'à la détection d'un sommet (extrémité)  $s_j^{\mathcal{CB}_{i,\theta}}$  de l'arête  $\vec{E}_j^{\mathcal{CB}_{i,\theta}}$ . La "reconnaissabilité" d'un sommet est possible grâce au changement des conditions de contact. Ce changement ne va apparaître que si le sommet  $s_j^{\mathcal{CB}_{i,\theta}}$  correspond à une convexité ou une concavité locale du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_{i,\theta}$ . Dans le cas d'une convexité  $\Gamma$  l'arrivée à un sommet  $s_j^{\mathcal{CB}_{i,\theta}}$  peut être reconnue par un "saut" (un fort changement) des valeurs récupérées par les capteurs qui suivent le contact (en l'occurrence  $\Gamma$  le passage entre deux arêtes adjacentes)  $\Gamma$  et il peut se produire même une annulation complète de ces valeurs lors de la perte totale du contact. Dans le cas d'une concavité  $\Gamma$  le sommet  $s_j^{\mathcal{CB}_{i,\theta}}$  est reconnu lorsqu'un nouveau contact est détecté le long de la direction du déplacement du robot.

#### 5.2.4.3 Configuration $q_c$ à l'extrémité d'une arête de $\mathcal{CB}_{i,\theta}$

Pendant le suivi en contact d'une arête  $\vec{E}_j^{\mathcal{CB}_{i,\theta}} \subset \partial\mathcal{CB}_{i,\theta}$   $\Gamma$  le robot ponctuel  $q_A$  peut arriver à un sommet  $s_j^{\mathcal{CB}_{i,\theta}}$  en exécutant un déplacement compliant  $\vec{\Delta}q'_t$ . Un tel sommet peut être soit convexe  $\Gamma$  soit concave. Une convexité correspond à un *contact simple* de type sommet $_A$ -sommet $_B$ , tandis qu'une concavité correspond à un *contact double*. Afin garantir la continuité de la fonction de potentiel  $\Gamma$  son comportement doit être défini lorsqu'un sommet est détecté. La contrainte de sécurité du robot doit être prise en compte car localement le robot est très proche d'au moins un obstacle. De plus  $\Gamma$  pour ne pas tomber dans une situation d'oscillation autour d'un sommet  $s_j^{\mathcal{CB}_{i,\theta}}$   $\Gamma$  le prochain déplacement en contact avec le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  devra suivre le sens (positif ou négatif) du déplacement  $\vec{\Delta}q'_t$  arrivant sur  $s_j^{\mathcal{CB}_{i,\theta}}$ . Cette contrainte est aussi valable pour les déplacements en rotation qui cherchent à définir un contact avec  $\mathcal{CB}_i$ .

#### Concavité locale

Si le sommet détecté correspond à une concavité locale  $\Gamma$  la fonction de potentiel détermine un déplacement compliant  $\vec{\Delta}q''_t$  depuis la configuration courante  $q_c$  le long de l'arête adjacente  $\vec{E}_k^{\mathcal{CB}_{i,\theta}}$ . Le robot garde donc son orientation.

#### Convexité locale

Dans le cas où le sommet détecté correspond à une convexité locale  $\Gamma$  le *cône de rupture de contact* sur le sommet  $s_j^{\mathcal{CB}_{i,\theta}}$  permet de décider si le robot ponctuel  $q_A$  conti-

nue la recherche du but dans l'espace  $\mathcal{C}_{libre}$  ou si le contact est maintenu avec l'arête adjacente  $\vec{E}_k^{\mathcal{C}\mathcal{B}_i,\theta}$ .

Un passage du robot ponctuel  $q_A$  vers l'espace  $\mathcal{C}_{libre}$  est possible si la direction  $\vec{\Delta}q_t$  sur  $q_c$  est à l'intérieur du *cône de rupture de contact* sur le même sommet. Le déplacement suivi est celui défini par  $\vec{\Delta}q_t$ . L'analyse d'incertitude dans l'espace  $\mathcal{C}_{libre}$  est appliquée en sachant que l'incertitude de position en  $q_c$  est annulée (i.e.  $\rho_c = 0$ ).

Par contre si la navigation doit continuer dans l'espace  $\mathcal{C}_{contact}$  le sommet  $s_j^{\mathcal{C}\mathcal{B}_i,\theta}$  est contourné en appliquant deux déplacements consécutifs : un déplacement en rotation  $\Delta_\theta^p$  et un déplacement en translation  $\vec{\Delta}q_t''$ . La génération de ces déplacements est expliquée par la suite.

### Déplacement en rotation $\Delta_\theta^p$

Ce déplacement cherche à définir un contact simple de type arête $_A$ -arête $_{B_i}$  d'une manière semblable au déplacement d'"alignement" appliqué dans §5.2.3.2. Nous ne précisons donc que la détermination des paramètres  $p$  et  $\theta$  nécessaires pour  $\Delta_\theta^p$ .

Soient :

- $\vec{E}_j^{\mathcal{C}\mathcal{B}_i,\theta}$  la dernière arête de contact suivie avant d'arriver au sommet  $s_j^{\mathcal{C}\mathcal{B}_i,\theta} \Gamma$
- $s$  le sens (+ (positif) ou – (négatif)) du déplacement le long de  $\vec{E}_j^{\mathcal{C}\mathcal{B}_i,\theta} \Gamma$
- $\vec{E}_k^{\mathcal{C}\mathcal{B}_i,\theta}$  l'arête adjacente de contact selon le sens  $s \Gamma$
- $\varphi_j^+$ ,  $\varphi_j^-$  et  $\varphi_j^\circ$  les paramètres d'"alignement" associés à la arête  $\vec{E}_j^{\mathcal{C}\mathcal{B}_i,\theta} \Gamma$
- $\varphi_k^+$ ,  $\varphi_k^-$  et  $\varphi_k^\circ$  les paramètres d'"alignement" associés à la arête adjacente  $\vec{E}_k^{\mathcal{C}\mathcal{B}_i,\theta} \Gamma$
- $\alpha$  ( $0 < \alpha < \pi$ ) l'angle à l'intérieur de l'obstacle  $\mathcal{B}_i$  entre les arêtes  $\vec{E}_j^{\mathcal{B}_i}$  et  $\vec{E}_k^{\mathcal{B}_i}$  dont  $\varphi_k^\circ$  est le sommet commun.

En ce qui concerne au paramètre  $p$  il prend la valeur du paramètre d'"alignement"  $\varphi_k^\circ$ .

En ce qui concerne au paramètre  $\theta$  l'expression qui détermine sa valeur absolue  $|\theta|$  dépend du type de contact suivi le long de l'arête  $\vec{E}_j^{\mathcal{C}\mathcal{B}_i,\theta}$  :

$$|\theta| = \begin{cases} \pi - \alpha, & \text{contact suivi arête}_A\text{-arête}_{B_i} \quad (\text{cas 1}) \\ |\varphi_j^\circ| + \pi - \alpha, & \text{contact suivi sommet}_A\text{-arête}_{B_i} \quad (\text{cas 2}) \\ |\varphi_j^\circ|, & \text{contact suivi arête}_A\text{-sommet}_{B_i} \quad (\text{cas 3}) \end{cases} \quad (5.17)$$

Le premier cas  $\Gamma$ cas général  $\Gamma$ est illustré sur la figure 5.13a. Le deuxième et troisième cas se présentent moins souvent car le potentiel dans l'espace  $\mathcal{C}_{contact}$  cherche si possible  $\Gamma$

à maintenir des contacts de type arête<sub>A</sub>-arête<sub>B<sub>i</sub></sub>. Ils sont illustrés sur les figures 5.13b et 5.13c respectivement.

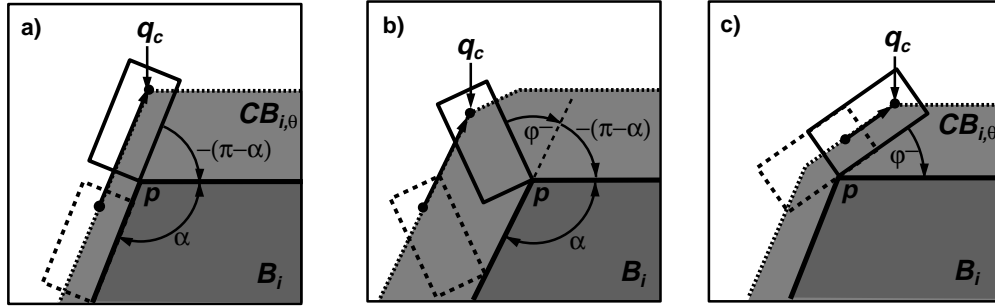


FIG. 5.13 – a) Illustration de l'angle de rotation  $\theta$  pour passer à un contact arête<sub>A</sub>-arête<sub>B<sub>i</sub></sub> après le suivi d'un contact arête<sub>A</sub>-arête<sub>B<sub>i</sub></sub>.  
 b) Illustration de l'angle de rotation  $\theta$  pour atteindre un contact arête<sub>A</sub>-arête<sub>B<sub>i</sub></sub> après le suivi d'un contact sommet<sub>A</sub>-arête<sub>B<sub>i</sub></sub>.  
 c) Illustration de l'angle de rotation  $\theta$  pour atteindre un contact arête<sub>A</sub>-arête<sub>B<sub>i</sub></sub> après le suivi d'un contact arête<sub>A</sub>-sommet<sub>B<sub>i</sub></sub>.

La configuration après avoir effectué le déplacement de rotation<sup>1</sup>  $\Delta_\theta^p(q_c, p, s|\theta)$  est notée  $q'_c$ .

### Déplacement en translation $\vec{\Delta}q''_t$

La configuration  $q'_c$  atteinte par le déplacement en rotation  $\Delta_\theta^p(q_c, p, s|\theta)$  correspond à un nouveau sommet  $s_k^{\mathcal{CB}_{i,\theta'_c}}$  du contour  $\partial\mathcal{CB}_{i,\theta'_c}$ . Afin d'éloigner le robot ponctuel  $q_A$  du sommet  $\Gamma$  un deuxième déplacement en translation  $\vec{\Delta}q''_t$  est défini par la fonction de potentiel. Le robot garde donc son orientation  $\theta'_c$ . Ce nouveau déplacement est un déplacement compliant si le long de la nouvelle arête de contact  $\vec{E}_k^{\mathcal{CB}_{i,\theta'_c}}$  le robot ponctuel  $q_A$  peut s'approcher de la position but  $p_f\Gamma$  sinon un déplacement libre est défini pour approcher directement  $p_f$ . Ceci dépend à la fois de la direction  $\vec{D}_k^{\mathcal{CB}_{i,\theta'_c}}$  à suivre de l'arête  $\vec{E}_k^{\mathcal{CB}_{i,\theta'_c}}$  et aussi de la direction d'attraction  $\vec{\Delta}q_t$  depuis  $p'_c$  vers la position but  $p_f$ . La direction  $\vec{D}_k^{\mathcal{CB}_{i,\theta'_c}}$  doit garder le sens  $s$  du déplacement  $\Delta_\theta^p$  précédent. Ainsi  $\vec{D}_k^{\mathcal{CB}_{i,\theta'_c}} = \vec{E}_k^{\mathcal{CB}_{i,\theta'_c}}$  si le sens  $s$  est positif  $\Gamma$  sinon  $\vec{D}_k^{\mathcal{CB}_{i,\theta'_c}} = -\vec{E}_k^{\mathcal{CB}_{i,\theta'_c}}$ .

Soient :

- $\vec{u}(\vec{v})$  le vecteur unitaire suivant la direction du vecteur  $\vec{v}\Gamma$
- $\varrho$  l'angle de rupture de contact permettant de lâcher le contact présent à la position nominale  $p'_c$  sur la nouvelle arête de contact  $\vec{E}_k^{\mathcal{CB}_{i,\theta'_c}}$ .

1. La rotation est effectuée dans le sens  $s$  (+ ou -) du dernier déplacement en contact.



–  $\vec{\Delta}q_t^{\leftrightarrow}$  la composante colinéaire du déplacement  $\vec{\Delta}q_t$  le long de  $\vec{D}_k^{\mathcal{C}B_i, \theta'_c}$ .

Le déplacement  $\vec{\Delta}q_t''$  est défini de la manière suivante :

$$\vec{\Delta}q_t'' = \begin{cases} \vec{\Delta}q_t^{\leftrightarrow}, & \text{si } (\vec{\Delta}q_t \cdot \vec{D}_k^{\mathcal{C}B_i, \theta'_c} > 0) \text{ et } (\vec{u}(\vec{\Delta}q_t) \cdot \vec{n}_k^{\mathcal{C}B_i, \theta'_c} < \cos(\varrho)) \\ \vec{\Delta}q_t, & \text{sinon} \end{cases} \quad (5.18)$$

Le premier cas correspond à la génération d'un *déplacement compliant* (figure 5.14a) et le deuxième cas génère un *déplacement libre* (figure 5.14b).

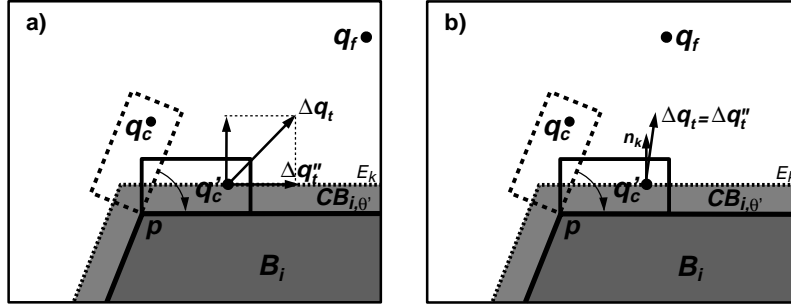


FIG. 5.14 – Suite des déplacements de rotation  $\Delta_\theta^p(q_c, p, s|\theta|)$  et de translation  $\vec{\Delta}q_t''(p'_c, p_f)$  pour la situation illustrée sur la figure 5.13a. Après avoir exécuté le déplacement  $\Delta_\theta^p(q_c, p, -|\theta|)$ , l'approche vers le but  $q_f$  continue suivant le déplacement  $\vec{\Delta}q_t''$ . Ce déplacement est compliant pour le cas illustré dans a), et libre pour le cas b).

### Traitement de l'incertitude

L'estimation de l'incertitude géométrique doit être faite d'une part pour le déplacement en rotation  $\Delta_\theta^p(q_c, p, s|\theta|)$  et d'autre part pour le déplacement en translation  $\vec{\Delta}q_t''(p'_c, p_f)$ . L'état initial de la configuration  $q_c$  est caractérisé par une incertitude de position nulle<sup>2</sup> ( $\rho_c = 0$ ) et une incertitude d'orientation  $\psi_c \geq 0$ . La valeur de  $\psi_c$  dépend du contact suivi pour arriver à la configuration  $q_c$ . Le déplacement en rotation  $\Delta_\theta^p$  permet d'annuler l'incertitude d'orientation même si la rotation a été bloquée par autre obstacle c'est-à-dire que  $\psi_c = 0$ . Ensuite le déplacement compliant  $\vec{\Delta}q_t''$  n'affecte que l'incertitude de position de la nouvelle configuration  $q_c''$ . Cette incertitude est modélisée par un intervalle centré sur  $p_c''$  le long de l'arête de contact (§2.3.1). Les estimations sur l'incertitude de position/orientation sur  $q_c''$  sont données par :

$$\begin{aligned} \rho_{c''} &= \|\vec{\Delta}q_t''(p'_c, p_f)\| * k_t \quad (\text{cf. équation 5.15}), \\ \psi_{c''} &= 0 \end{aligned} \quad (5.19)$$

avec  $k_t = \tan(\eta)$  selon une hypothèse faite dans §2.3.2.

<sup>2</sup>. La position du robot a pu être précisée par rapport au sommet (d'un obstacle) détecté à la configuration  $q_c$ .

### 5.2.5 Algorithme de la fonction de potentiel fictif

Un algorithme simplifié est présenté ci-dessous. Cependant un algorithme plus détaillé est explicité en annexe §A.1.

ENTRÉES :

- $q_0$  la configuration de départ et  $(\rho_0, \psi_0)$  les incertitudes de position et d'orientation associées;
- $q_f$  la configuration but et  $(\rho_f, \psi_f)$  les incertitudes maximales de position et d'orientation acceptées;
- $\eta$  et  $k_r$  les paramètres d'incertitude propres au robot  $\mathcal{A}$ ;
- $\epsilon_t$  et  $\varrho$  les paramètres propres à la fonction de potentiel fictif.

SORTIES :

- $q_c$  la dernière configuration atteignable par le robot et  $(\rho_c, \psi_c)$  les incertitudes de position et d'orientation associées;
- $\tau_{pot}$  le chemin *valide et robuste* reliant la configuration  $q_0$  à la configuration  $q_c$ ;
- La condition d'arrêt à la configuration  $q_c$  :
  - Valeur **SUCCÈS** si la configuration  $q_f$  est atteinte depuis la configuration  $q_0$  avec des estimations d'incertitude de position et d'orientation satisfaisantes.
  - Valeur **ÉCHEC** dans le cas contraire.

ALGORITHME :

INITIALISATION :  $q_c = q_0 \Gamma \rho_c = \rho_0 \Gamma \psi_c = \psi_0$ .

1. Tant-que  $(q_c \neq q_f)$  et  $\neg$  minimum-local( $q_c$ )
  - Déterminer le déplacement incrémental  $\vec{\Delta}q(q_c, q_f) \Gamma$
  - Si  $q_c \in \mathcal{C}_{libre}$  Alors
    - Si Voisinage-Incertitude( $q_c, \rho_c, \psi_c$ )  $\subset \mathcal{C}_{libre}$  Alors
      - Traiter le cas : comportement dans  $\mathcal{C}_{libre}$  depuis  $q_c$  avec  $\vec{\Delta}q(q_c, q_f)$
    - Sinon
      - Traiter le cas : passage de  $\mathcal{C}_{libre}$  vers  $\mathcal{C}_{contact}$  depuis  $q_c$  avec  $\vec{\Delta}q(q_c, q_f)$
    - Sinon
      - Traiter le cas : comportement dans  $\mathcal{C}_{contact}$  depuis  $q_c$  avec  $\vec{\Delta}q(q_c, q_f)$
  - Fin-Tant-que
2. Construire le chemin  $\tau_{pot}$  entre  $q_0$  et  $q_c \Gamma$
3. Si  $q_c = q_f$  et  $\rho_c \leq \rho_f$  et  $\psi_c \leq \psi_f$  Alors retourner **SUCCÈS**. Sinon  $\Gamma$  retourner **ÉCHEC**.

## 5.3 La fonction d’exploration

### 5.3.1 Le principe

La fonction de potentiel fictif proposée dans l’espace  $\mathcal{C}_{valide}$  peut tomber dans des configurations de minimum local définies dans l’espace  $\mathcal{C}_{contact}$ . Comme nous l’avons expliqué dans §4.1.2.4 une adaptation de la fonction EXPLORE de l’algorithme Fil d’Ariane est appliquée pour chercher à s’échapper des minima locaux. Plus précisément le but de la fonction EXPLORE est de choisir et d’amener le robot ponctuel  $q_A$  vers une configuration sous-but lui permettant de sortir d’un minimum local. Le choix d’une configuration sous-but se fait parmi l’ensemble des configurations appartenant à une discrétisation d’un voisinage potentiellement “atteignable” depuis un minimum local. Le voisinage correspond à un sous-espace de contact autour d’un minimum local. Selon la situation de blocage définie par un minimum local et la taille du voisinage considéré autour de celui-ci il se peut que la configuration sous-but choisie ne permette pas au robot  $q_A$  d’échapper au minimum en retombant donc dans le même minimum local. Cependant une configuration sous-but permet à la fonction EXPLORE d’élargir le sous-espace de recherche propre au minimum local traité. Ainsi les configurations sous-buts sont définies de plus en plus loin d’un minimum local. La fonction EXPLORE s’adapte donc à la complexité de la situation de minimum local traité.

Lorsqu’une configuration sous-but est choisie un planificateur local est appliqué pour tenter de l’atteindre depuis un minimum local traité ou une configuration sous-but précédemment atteinte. Le planificateur local est propre à la fonction EXPLORE et peut aussi tomber dans des configurations de minimum local. La configuration atteinte par le planificateur (i.e. le sous-but choisi ou un minimum local) devient la nouvelle configuration de départ pour la fonction de potentiel.

Dans la suite le concept de balise sur lequel est basé l’exploration de l’espace de recherche autour d’un minimum local est défini. Ensuite deux stratégies d’exploration de l’espace de recherche chacune appliquée dans un voisinage spécifique autour d’une “balise” sont expliquées. Un algorithme simplifié est présenté pour chaque stratégie d’exploration.

### 5.3.2 Concept de balise

L’application de la fonction EXPLORE originale est basée sur la définition des balises. Dans notre approche une *balise* correspond à une configuration dans l’espace  $\mathcal{C}_{contact}$  qui a pu être atteinte depuis la configuration initiale  $q_0$  (e.g. un minimum local ou une configuration sous-but) tenant compte des contraintes d’incertitude de position d’orientation et de contrôle sur le robot ponctuel  $q_A$ . Une configuration de contact  $q = (x, y, \theta)$  atteinte par  $q_A$  est définie soit comme une configuration sur le contour d’une tranche  $\mathcal{CB}_{i,\theta}$  d’un *C-obstacle*  $\mathcal{CB}_i$  ( $q \in \partial\mathcal{CB}_{i,\theta}$ ) soit comme une configuration

d'intersection entre les contours de deux tranches  $\mathcal{CB}_{i,\theta}$  et  $\mathcal{CB}_{j,\theta}$  des  $\mathcal{C}$ -obstacles  $\mathcal{CB}_i$  et  $\mathcal{CB}_j$  ( $q \in \partial\mathcal{CB}_{i,\theta} \cap \partial\mathcal{CB}_{j,\theta}$ ). Dans les deux cas la configuration de contact  $q$  définit une balise  $l_{i,\theta}^k$  dont les incertitudes de position et d'orientation sont  $\rho_{l_{i,\theta}^k}$  et  $\psi_{l_{i,\theta}^k}$  respectivement.

### Cas spécial : la configuration initiale $q_0$

L'aspect "atteignable" dans la définition de balise considère que le robot ponctuel  $q_A$  peut atteindre une configuration  $q$  depuis la configuration initiale  $q_0$  malgré les effets des incertitudes. Puisque l'on suppose que le robot part de la configuration  $q_0$  on peut considérer que  $q_0$  est "atteignable" donc  $q_0$  peut devenir une balise. Cependant selon la définition de la configuration initiale  $q_0$  cette dernière appartient à  $\mathcal{C}_{valide}$  ce qui admet la possibilité que  $q_0$  soit dans l'espace  $\mathcal{C}_{libre}$ . Si c'est le cas l'acceptation de la configuration  $q_0$  comme balise sera en contradiction avec la condition d'appartenance à l'espace  $\mathcal{C}_{contact}$ . Pour cette raison si  $q_0$  est dans  $\mathcal{C}_{libre}$  celle-ci est considérée comme un cas spécial de balise. Cette balise est notée  $l_{0,\theta}^0$  et ne peut être liée à aucun  $\mathcal{C}$ -obstacle dans l'espace des configurations  $\mathcal{C}$ .

### 5.3.3 Le sous-espace de contact adjacent et son exploration

Un premier voisinage défini autour d'une balise est appelé le *sous-espace de contact adjacent*. Le but de l'exploration de ce sous-espace est de positionner le robot ponctuel  $q_A$  dans une configuration de contact qui puisse être atteinte depuis une balise par l'application des déplacements compliants. Ceci permet au robot de se déplacer en gardant une incertitude de position faible. La configuration de contact atteinte après l'exploration adjacente devient d'une part une nouvelle balise et d'autre part une nouvelle configuration de départ pour la fonction de potentiel fictif.

Dans la suite la définition de *sous-espace de contact adjacent* est donnée à plusieurs niveaux : balise  $l_{i,\theta}^k$   $\mathcal{C}$ -obstacle (tranche)  $\mathcal{CB}_{i,\theta}$  et  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ . L'exploration des différents sous-espaces est aussi détaillée. Finalement un algorithme simplifié résume l'application de l'exploration adjacente sur un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ .

#### 5.3.3.1 Sous-espace de contact adjacent autour d'une balise $l_{i,\theta}^k$

Soit  $l_{i,\theta}^k$  une balise. Si  $l_{i,\theta}^k$  n'appartient qu'à une tranche  $\mathcal{CB}_{i,\theta}$  le *sous-espace de contact adjacent*  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$  autour de  $l_{i,\theta}^k$  est défini par une ligne polygonale composée de deux demi-lignes polygonales définies chacune depuis la balise  $l_{i,\theta}^k$ . Chaque demi-ligne suit un sens différent du contour  $\partial\mathcal{CB}_{i,\theta}$ . La longueur de chaque demi-ligne est définie comme un pourcentage du périmètre du contour  $\partial\mathcal{CB}_{i,\theta}$  borné à 50% (figure 5.15a). On a donc  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k) \subseteq \partial\mathcal{CB}_{i,\theta}$ .

Par contre  $\Gamma$  si  $l_{i,\theta}^k$  correspond à une intersection entre deux tranches  $\mathcal{CB}_{i,\theta}$  et  $\mathcal{CB}_{j,\theta}$  le sous-espace de contact adjacent  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$  autour de  $l_{i,\theta}^k$  est défini par deux demi-lignes polygonales  $\Gamma$  dont l'origine est  $l_{i,\theta}^k$  l'une suivant le côté atteignable du contour  $\partial\mathcal{CB}_i$  et l'autre selon le côté atteignable du contour  $\partial\mathcal{CB}_j$  (figure 5.15b). La longueur de chaque demi-ligne est aussi bornée à 50% du contour atteignable commun. On a donc  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k) \subseteq \partial((\mathcal{CB}_{i,\theta} \cup \mathcal{CB}_{j,\theta}) - (\mathcal{CB}_{i,\theta} \cap \mathcal{CB}_{j,\theta}))$ .

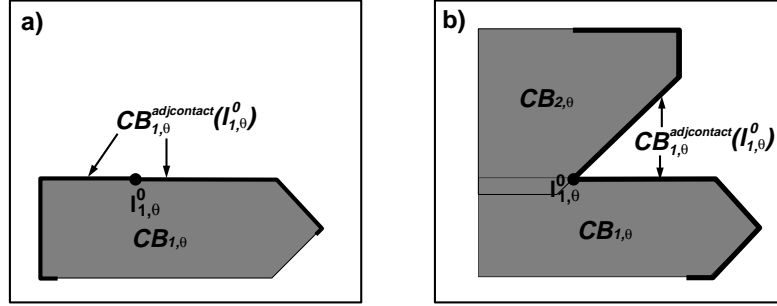


FIG. 5.15 – a) Sous-espace de contact adjacent autour de la balise initiale  $l_{1,\theta}^0$  sur le contour  $\partial\mathcal{CB}_{1,\theta}$ . b) Sous-espace de contact adjacent autour de la balise  $l_{1,\theta}^0$  commune aux contours  $\partial\mathcal{CB}_{1,\theta}$  et  $\partial\mathcal{CB}_{2,\theta}$ .

### Discretisation du sous-espace $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$

Le sous-espace continu  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$  peut être discrétisé par un ensemble discret de configurations (attracteurs)  $AAT_{i,\theta}^k$ . Cet ensemble est défini comme l'union des discrétisations de chacune des arêtes appartenant au sous-espace  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$ . La discrétisation d'une arête contient un nombre fixe d'attracteurs définis de manière aléatoire dans un intervalle inclus dans celle-ci. Cet intervalle écarte un voisinage autour de chaque extrême de l'arête pour éviter de définir des attracteurs très proches à ceux-ci. Un attracteur dans l'ensemble  $AAT_{i,\theta}^k$  est noté  $a_{i,\theta}^{k,m}$ . On associe à chaque attracteur  $a_{i,\theta}^{k,m}$  la balise  $l_{i,\theta}^k$  qui l'a généré.

#### 5.3.3.2 Sous-espace de contact adjacent autour de $\mathcal{CB}_{i,\theta}$

Soit  $\mathcal{L}_{i,\theta}$  l'ensemble des balises définies sur le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  dont l'angle est  $\theta$ . Le sous-espace de contact adjacent  $\mathcal{CB}_{i,\theta}^{adjcontact}$  autour d'une tranche  $\mathcal{CB}_{i,\theta}$  du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  est défini comme l'union des sous-espaces de contact adjacent des balises  $l_{i,\theta}^k$  dans  $\mathcal{L}_{i,\theta}$  ainsi :

$$\mathcal{CB}_{i,\theta}^{adjcontact} = \bigcup_{l_{i,\theta}^k \in \mathcal{L}_{i,\theta}} \mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k) \quad (5.20)$$

On a donc  $\mathcal{CB}_{i,\theta}^{adjcontact} \subseteq \partial\mathcal{CB}_{i,\theta}$ . Dans le cas où une balise  $l_{i,\theta}^k$  est définie comme l'intersection de deux tranches  $\mathcal{CB}_{i,\theta}$  et  $\mathcal{CB}_{j,\theta}$  de  $\mathcal{C}$ -obstacles différents  $\Gamma$  les sous-espaces

de contact adjacent  $\mathcal{CB}_{i,\theta}^{adjcontact}$  et  $\mathcal{CB}_{j,\theta}^{adjcontact}$  deviennent un sous-espace de contact adjacent commun. Ce sous-espace commun est défini sur le contour atteignable commun aux deux tranches  $\Gamma\partial((\mathcal{CB}_{i,\theta} \cup \mathcal{CB}_{j,\theta}) - (\mathcal{CB}_{i,\theta} \cap \mathcal{CB}_{j,\theta}))$ . De plus toute référence postérieure au contour de la tranche  $\mathcal{CB}_{i,\theta}$  ou  $\mathcal{CB}_{j,\theta}$  fait référence au nouveau contour atteignable. La définition d'un contour commun force la redéfinition des sous-espaces de contact adjacent des balises dans  $\mathcal{L}_{i,\theta}$  et  $\mathcal{L}_{j,\theta}$ . Chaque sous-espace  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$  et  $\mathcal{CB}_{j,\theta}^{adjcontact}(l_{j,\theta}^k)$  doit être redéfini dans le contour commun. Lorsque ces sous-espaces sont redéfinis le sous-espace de contact adjacent commun est défini comme :

$$\mathcal{CB}_{i,\theta}^{adjcontact} = \mathcal{CB}_{j,\theta}^{adjcontact} = \bigcup_{l_{i,\theta}^k \in \mathcal{L}_{i,\theta}} \mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k) \cup \bigcup_{l_{j,\theta}^k \in \mathcal{L}_{j,\theta}} \mathcal{CB}_{j,\theta}^{adjcontact}(l_{j,\theta}^k) \quad (5.21)$$

### Discrétisation du sous-espace $\mathcal{CB}_{i,\theta}^{adjcontact}$

L'union des discrétisations  $AAT_{i,\theta}^k$  des sous-espaces  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$  correspondants aux balises  $l_{i,\theta}^k$  dans l'ensemble  $\mathcal{L}_{i,\theta}$  définit une discrétisation  $AAT_{i,\theta}$  du sous-espace continu  $\mathcal{CB}_{i,\theta}^{adjcontact}$  (figure 5.16a) :

$$AAT_{i,\theta} = \bigcup_{\{k \mid l_{i,\theta}^k \in \mathcal{L}_{i,\theta}\}} AAT_{i,\theta}^k \quad (5.22)$$

### Exploration du sous-espace $\mathcal{CB}_{i,\theta}^{adjcontact}$

Le sous-espace de contact continu  $\mathcal{CB}_{i,\theta}^{adjcontact}$  peut être exploré incrémentalement depuis une balise initiale  $l_{i,\theta}^0$ . L'exploration se fait à partir d'une discrétisation  $AAT_{i,\theta}$  de l'espace  $\mathcal{CB}_{i,\theta}^{adjcontact}$ . A chaque pas  $k$  de l'exploration une nouvelle balise  $l_{i,\theta}^k$  est définie. La définition d'une balise  $l_{i,\theta}^k$  passe par deux étapes dans l'exploration. D'abord une étape de définition d'un sous-but  $\beta_{i,\theta}^k \in \mathcal{CB}_{i,\theta}^{adjcontact}$  à atteindre depuis une balise  $l_{i,\theta}^j$  précédemment définie. Le choix du sous-but est local à la tranche  $\mathcal{CB}_{i,\theta}$  du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ . Pour cette raison il se peut que le sous-but  $\beta_{i,\theta}^k$  choisi ne soit pas atteint depuis la balise  $l_{i,\theta}^j$ . Il est donc nécessaire de vérifier l'"atteignabilité" du sous-but choisi. Pour cela une deuxième étape dans l'exploration est considérée. Elle consiste à approcher le sous-but  $\beta_{i,\theta}^k$  depuis la balise  $l_{i,\theta}^j$  en suivant le plus court chemin le long de la ligne polygonale  $\mathcal{CB}_{i,\theta}^{adjcontact}$ . La configuration atteinte après l'approche dans le cas de succès ou d'échec devient la nouvelle balise  $l_{i,\theta}^k$  dans le sous-espace  $\mathcal{CB}_{i,\theta}^{adjcontact}$ .

### Choix d'un sous-but $\beta_{i,\theta}^k$ dans le sous-espace $\mathcal{CB}_{i,\theta}^{adjcontact}$

Initialement  $\Gamma\mathcal{L}_{i,\theta} = \{l_{i,\theta}^0\} \Gamma \mathcal{CB}_{i,\theta}^{adjcontact} = \mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^0)$  et  $AAT_{i,\theta} = AAT_{i,\theta}^0$ . Le premier sous-but  $\beta_{i,\theta}^1$  correspond à l'attracteur  $a_{i,\theta}^{0,m} \in AAT_{i,\theta}$  le plus éloigné de la balise  $l_{i,\theta}^0$  (figure 5.16a). La distance entre la balise  $l_{i,\theta}^0$  et un attracteur  $a_{i,\theta}^{0,m}$  correspond

à la distance géodésique<sup>3</sup> entre les deux configurations le long de la ligne polygonale  $\mathcal{CB}_{i,\theta}^{adjcontact}$ . Cette distance est notée :  $\|l_{i,\theta}^0 - a_{i,\theta}^{0,m}\|_{g\acute{e}od\acute{e}sique}$ . Le choix du premier sous-but  $\beta_{i,\theta}^1$  passe donc par la résolution du problème d'optimisation :

$$\begin{cases} \text{choisir } \beta_{i,\theta}^1 \text{ comme l'attracteur } a_{i,\theta}^{0,m} \text{ maximisant :} \\ \max_{a_{i,\theta}^{0,m} \in AAT_{i,\theta}^0} \|l_{i,\theta}^0 - a_{i,\theta}^{0,m}\|_{g\acute{e}od\acute{e}sique} \end{cases} \quad (5.23)$$

Le problème d'approche ainsi posé consiste donc à atteindre le sous-but  $\beta_{i,\theta}^1$  depuis  $l_{i,\theta}^0$  en suivant le plus court chemin le long de la ligne polygonale  $\mathcal{CB}_{i,\theta}^{adjcontact}$ .

Dans le cas général on va déterminer le sous-but  $\beta_{i,\theta}^k$  dans le sous-espace  $\mathcal{CB}_{i,\theta}^{adjcontact}$  lorsqu'un ensemble de balises  $\mathcal{L}_{i,\theta} = \{l_{i,\theta}^0, \dots, l_{i,\theta}^{k-1}\}$  a été défini dans le sous-espace. Le sous-but  $\beta_{i,\theta}^k$  correspond à l'attracteur  $a_{i,\theta}^{j,n} \in AAT_{i,\theta}$  le plus éloigné de l'ensemble de balises  $\mathcal{L}_{i,\theta}$ . Intuitivement le sous-but  $\beta_{i,\theta}^k$  correspond à une configuration à l'intérieur de la région moins explorée du sous-espace  $\mathcal{CB}_{i,\theta}^{adjcontact}$ . Le choix de  $\beta_{i,\theta}^k$  passe donc par la résolution du problème d'optimisation :

$$\begin{cases} \text{choisir } \beta_{i,\theta}^k \text{ comme l'attracteur } a_{i,\theta}^{j,n} \text{ maximisant :} \\ \max_{a_{i,\theta}^{j,n} \in AAT_{i,\theta}} \min_{l_{i,\theta}^h \in \mathcal{L}_{i,\theta}} \|l_{i,\theta}^h - a_{i,\theta}^{j,n}\|_{g\acute{e}od\acute{e}sique} \end{cases} \quad (5.24)$$

L'attracteur  $a_{i,\theta}^{j,n}$  choisi comme sous-but  $\beta_{i,\theta}^k$  appartient au sous-ensemble d'attracteurs  $AAT_{i,\theta}^j$  défini par la balise  $l_{i,\theta}^j$ . Le nouveau problème d'approche posé consiste donc à atteindre le sous-but  $\beta_{i,\theta}^k$  depuis  $l_{i,\theta}^j$  en suivant le plus court chemin le long de la ligne polygonale  $\mathcal{CB}_{i,\theta}^{adjcontact}$ . Il faut noter que le sous-but  $\beta_{i,\theta}^k$  et la balise  $l_{i,\theta}^j$  appartiennent à la même tranche  $\mathcal{CB}_{i,\theta}$  du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ .

#### Approche d'un sous-but $\beta_{i,\theta}^k$ dans le sous-espace $\mathcal{CB}_{i,\theta}^{adjcontact}$

La stratégie d'approche d'un sous-but  $\beta_{i,\theta}^k$  ( $k > 0$ ) depuis une balise  $l_{i,\theta}^j$  ( $0 \leq j \leq k - 1$ ) déjà atteinte est une stratégie de suivi du plus court chemin le long de la ligne polygonale  $\mathcal{CB}_{i,\theta}^{adjcontact}$ . Cette stratégie est basée sur l'application des déplacements compliants sur l'obstacle  $\mathcal{B}_i$  gardant l'orientation  $\theta$  du robot  $\mathcal{A}$ . L'incertitude de position est prise en compte le long des déplacements effectués et sa valeur peut être estimée à l'aide de l'équation 5.15. Nous rappelons que l'incertitude le long d'une arête en contact est un intervalle centré sur une configuration nominale et de longueur  $2\rho$  dont  $\rho$  est l'imprécision de chaque côté de l'arête. De plus l'intervalle est annulé lorsqu'un sommet est détecté. La configuration atteinte après le suivi devient la nouvelle balise  $l_{i,\theta}^k$  (figure 5.16b). En cas de succès  $l_{i,\theta}^k = \beta_{i,\theta}^k$ . Sinon  $l_{i,\theta}^k$  correspond à une configuration d'intersection entre deux tranches  $\mathcal{CB}_{i,\theta}$  et  $\mathcal{CB}_{j,\theta}$  des  $\mathcal{C}$ -obstacles  $\mathcal{CB}_i$  et  $\mathcal{CB}_j$ .

3. La distance géodésique entre deux points  $p$  et  $q$  appartenant à une surface connexe  $S$  est définie comme la longueur du plus court chemin dans  $S$  reliant  $p$  et  $q$ .

Pour chaque nouvelle balise  $l_{i,\theta}^k$  l'ensemble de contact adjacent  $\mathcal{CB}_{i,\theta}^{adjcontact}(l_{i,\theta}^k)$  et l'ensemble de la discrétisation  $AAT_{i,\theta}^k$  sont définis. De plus l'ensemble de balises  $\mathcal{L}_{i,\theta}$  et le sous-espace de contact adjacent  $\mathcal{CB}_{i,\theta}^{adjcontact}$  et l'ensemble  $AAT_{i,\theta}$  sont mis à jour.

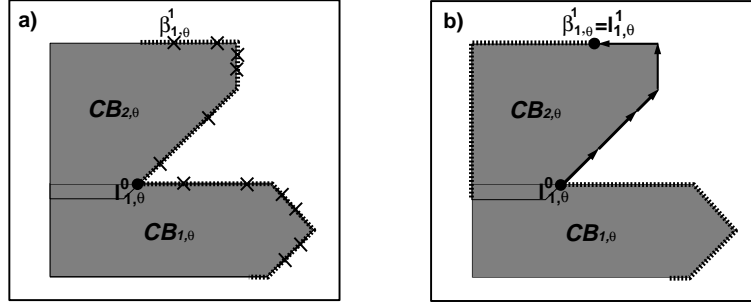


FIG. 5.16 – a) Ensemble d'attracteurs adjacents appartenant à la discrétisation du sous-espace initial  $\mathcal{CB}_{1,\theta}^{adjcontact}$ . L'attracteur choisi comme sous-but adjacent  $\beta_{1,\theta}^1$  est montré. b) Définition de la nouvelle balise  $l_{1,\theta}^1$  suite à l'approche adjacente du sous-but  $\beta_{1,\theta}^1$  depuis la balise  $l_{1,\theta}^0$ . Le sous-espace  $\mathcal{CB}_{1,\theta}^{adjcontact}$  est mis à jour.

### 5.3.3.3 Sous-espace de contact adjacent autour d'un $\mathcal{C}$ -obstacle $\mathcal{CB}_i$

Soit  $\mathcal{L}_i$  l'ensemble des balises  $l_{i,\theta}^k$  sur le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ . Le sous-espace de contact adjacent  $\mathcal{CB}_i^{adjcontact}$  associé au  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  est défini par l'ensemble des sous-espaces  $\mathcal{CB}_{i,\theta}^{adjcontact}$  engendrés par les sous-ensembles  $\mathcal{L}_{i,\theta}$  des balises ayant la même orientation  $\theta$  ainsi :

$$\mathcal{CB}_i^{adjcontact} = \bigcup_{\{\mathcal{L}_{i,\theta} \mid \mathcal{L}_{i,\theta} \subseteq \mathcal{L}_i\}} \mathcal{CB}_{i,\theta}^{adjcontact} \quad (5.25)$$

#### Exploration du sous-espace $\mathcal{CB}_i^{adjcontact}$

Dans l'ensemble des balises  $\mathcal{L}_i$  il peut exister des sous-ensembles des balises  $\mathcal{L}_{i,\theta}$  ayant des valeurs  $\theta$  différentes. L'exploration du sous-espace continu à morceaux  $\mathcal{CB}_i^{adjcontact}$  peut se faire incrémentalement. A chaque pas  $k$  de l'exploration le sous-espace de contact adjacent  $\mathcal{CB}_{i,\theta}^{adjcontact}$  définissant l'attracteur  $\beta_{i,\theta}^k$  le plus éloigné de ses propres balises est exploré.

#### Algorithme de la stratégie d'exploration du sous-espace $\mathcal{CB}_i^{adjcontact}$

ENTRÉES :

- $\mathcal{L}_i = \mathcal{L}_{i,\theta_1} \cup \dots \cup \mathcal{L}_{i,\theta_f} \cup \dots \cup \mathcal{L}_{i,\theta_n}$  la liste des balises (non vide) sur le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  dont  $\mathcal{L}_{i,\theta_f}$  est le sous-ensemble des balises ayant l'angle  $\theta_f$ ;
- $k_t$  la constante de proportionnalité sur les imprécisions des magnitudes des déplacements en translation;



- $\epsilon_t$  le pas de discrétisation en translation.

SORTIES :

- $l_{i,\theta_f}^k$  la nouvelle balise définie dans  $\mathcal{CB}_i$ ;
- $(\rho_{l_{i,\theta_f}^k}, \psi_{l_{i,\theta_f}^k})$  l'incertitude de position et d'orientation sur la nouvelle balise  $l_{i,\theta_f}^k$ ;
- $l_{i,\theta_f}^j$  la balise explorée dans  $\mathcal{L}_i$ ;
- $\tau_{adj}$  le chemin *valide* et *robuste* reliant les balises  $l_{i,\theta_f}^j$  et  $l_{i,\theta_f}^k$ ;
- $\mathcal{L}_i$  la nouvelle liste des balises sur le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ .

ALGORITHME :

1. Pour chaque sous-ensemble de balises  $\mathcal{L}_{i,\theta_e}$  ( $1 \leq e \leq n$ ) dans  $\mathcal{L}_i$  déterminer dans l'ensemble  $AAT_{i,\theta_e}$  la distance géodésique  $g_{i,\theta_e}$  du meilleur attracteur adjacent  $\beta_{i,\theta_e}^k$  (§5.3.3.2).
2. Déterminer la plus grande distance  $g_{i,\theta_f}$  de l'ensemble  $\{g_{i,\theta_1}, \dots, g_{i,\theta_n}\}$ .
3. Déterminer l'attracteur adjacent  $\beta_{i,\theta_f}^k$  et la balise  $l_{i,\theta_f}^j$  qui ont engendré la distance  $g_{i,\theta_f}$ .
4. Appliquer la stratégie d'approche dans l'espace de contact adjacent  $\mathcal{CB}_{i,\theta_f}^{adjcontact}$  depuis la balise  $l_{i,\theta_f}^j$  vers le sous-but  $\beta_{i,\theta_f}^k$ . La configuration atteinte par l'approche devient la nouvelle balise  $l_{i,\theta_f}^k$ . Cette balise est accompagnée des estimations d'incertitude de position  $\rho_{l_{i,\theta_f}^k}$  et d'orientation  $\psi_{l_{i,\theta_f}^k}$ . Le chemin  $\tau_{adj}$  reliant les balises  $l_{i,\theta_f}^j$  et  $l_{i,\theta_f}^k$  est retourné.
5. Mettre à jour  $\mathcal{L}_i$  (i.e.  $\mathcal{L}_i = \mathcal{L}_{i,\theta_1} \cup \dots \cup \mathcal{L}_{i,\theta_f} \cup \{l_{i,\theta_f}^k\} \cup \dots \cup \mathcal{L}_{i,\theta_n}$ ).

### 5.3.4 Le sous-espace de contact visible et son exploration

Un deuxième voisinage  $\Gamma$  appelé *le sous-espace de contact visible*  $\Gamma$  est associé à une balise. Le but de l'exploration de ce sous-espace est de positionner le robot ponctuel  $q_A$  dans une configuration de contact atteignable depuis une balise par l'application des déplacements dans l'espace  $\mathcal{C}_{libre}$ . Ceci permet au robot  $q_A$  d'arriver au contact avec des surfaces des  $\mathcal{C}$ -obstacles qui ne sont atteignables depuis une balise que si  $q_A$  navigue dans l'espace  $\mathcal{C}_{libre}$ . L'"atteignabilité" de ces surfaces est garantie sous les contraintes d'incertitude de position et de contrôle. Chaque configuration de contact ainsi atteinte devient  $\Gamma$  d'une part  $\Gamma$  une nouvelle balise  $\Gamma$  et  $\Gamma$  d'autre part  $\Gamma$  une nouvelle configuration de départ pour la fonction de potentiel fictif.

Tout d'abord les aspects de "visibilité" et "atteignabilité" depuis une balise sont traités. Ensuite le *sous-espace de contact visible* est défini depuis une balise vers un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_{i,\theta}$  et vers l'ensemble des  $\mathcal{C}$ -obstacles  $\mathcal{CB}_{i,\theta}$ . Finalement l'approche d'exploration du *sous-espace de contact visible* sur l'ensemble des  $\mathcal{C}$ -obstacles  $\mathcal{CB}_i$  est détaillée et explicitée avec un algorithme simplifié.

#### 5.3.4.1 "Visibilité" et "atteignabilité" depuis une balise $l_{i,\theta}^k$

Soient :

- $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$  une arête orientée du contour  $\mathcal{CB}_{j,\theta}$ ;
- $S(x, y) = 0$  l'équation de la ligne droite contenant l'arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$ ;
- $S^+(x, y)$  le sous-espace extérieur de  $S(x, y)$  par rapport à l'arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$ ;
- $\vec{n}_m^{\mathcal{CB}_{j,\theta}}$  le vecteur normal extérieur de l'arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$ ;
- $\vec{v}$  une vitesse en translation dans le cercle unitaire  $C^1$ .

D'après le critère de "visibilité" (§2.4.1) l'arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$  est "visible" depuis la balise (configuration)  $l_{i,\theta}^k = (x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k}, \theta)$  suivant la vitesse  $\vec{v}$  si les deux conditions ci-dessous sont satisfaites :

$$\begin{cases} S^+(x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k}) < 0; & \text{balise extérieur par rapport à l'arête} \\ \vec{v} \cdot \vec{n}_m^{\mathcal{CB}_{j,\theta}} < 0; & \text{vitesse d'approche vers la ligne } S(x, y) = 0 \end{cases} \quad (5.26)$$

La "visibilité" garantit que la vitesse  $\vec{v}$  permet au robot ponctuel  $q_A$  placé sur la balise  $l_{i,\theta}^k$  de s'approcher de la ligne droite  $S(x, y) = 0$  sans pour autant garantir le contact avec l'arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$ . Les obstructions potentielles le long du déplacement  $\vec{v}$  entre la balise  $l_{i,\theta}^k$  et la ligne  $S(x, y) = 0$  ne sont pas prises en compte.

En ce qui concerne les vitesses en translation  $\vec{v}$  pouvant être suivies depuis une balise  $l_{i,\theta}^k$  placée sur  $\mathcal{CB}_{i,\theta}$  celles-ci sont contenues dans le *cône des translations libres* propre à la balise  $l_{i,\theta}^k$  [Lau89]. Ce cône noté  $TL_{\mathbb{R}^2}(x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k})$  est défini comme l'ensemble des vecteurs dans le cercle unitaire  $C^1$  permettant de lâcher le contact présent sur la balise  $l_{i,\theta}^k$  et donc de passer dans l'espace  $\mathcal{C}_{libre}$ . La figure 5.17 illustre ce cône pour différents placements d'un point sur un polygone simple.

D'autre part l'ensemble des vitesses pouvant atteindre l'arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$  depuis la balise  $l_{i,\theta}^k$  sous les contraintes d'incertitude de position  $\rho_{l_{i,\theta}^k}$  et de contrôle  $\eta$  appelé le *cône des directions sûres* et noté  $VS_{\mathbb{R}^2}(\vec{E}_m^{\mathcal{CB}_{j,\theta}}, (x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k}), \rho_{l_{i,\theta}^k}, \eta) \subset C^1$  est déterminé à l'aide du modèle *projection en avant* dans l'espace  $\mathbb{R}^2$  (§2.4.2.1).

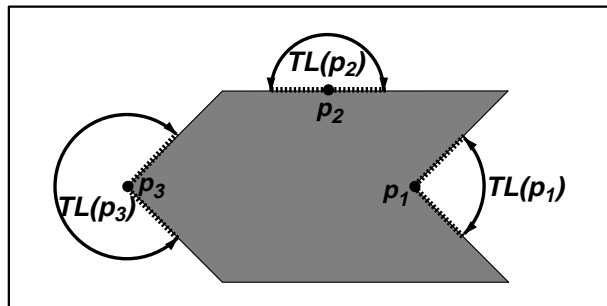


FIG. 5.17 – Cônes des translations libres  $TL_{\mathbb{R}^2}(p)$  définis pour: un sommet d’une concavité ( $p = p_1$ ), un point à l’intérieur d’une arête ( $p = p_2$ ) et un sommet d’une convexité ( $p = p_3$ ).

Le critère d’“atteignabilité” d’une arête visible  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$  depuis la balise  $l_{i,\theta}^k$  restreint les directions dans le cône  $TL_{\mathbb{R}^2}(x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k})$  au sous-ensemble des directions permettant d’atteindre l’arête sous les contraintes d’incertitude de position  $\rho_{l_{i,\theta}^k}$  et de contrôle  $\eta$ . En d’autres termes le cône effectif des vitesses pouvant atteindre l’arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$  résulte de l’intersection des cônes  $TL_{\mathbb{R}^2}(x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k})$  et  $VS_{\mathbb{R}^2}(\vec{E}_m^{\mathcal{CB}_{j,\theta}}, (x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k}), \rho_{l_{i,\theta}^k}, \eta)$ . Si le cône d’intersection n’est pas nul l’intervalle atteignable  $R$  de l’arête  $\vec{E}_m^{\mathcal{CB}_{j,\theta}}$  est obtenu en plaçant le sommet du cône résultat à la position  $(x_{l_{i,\theta}^k}, y_{l_{i,\theta}^k})$  et en projetant le cône sur l’arête (figure 5.18). La définition des cônes  $TL_{\mathbb{R}^2}$  et  $VS_{\mathbb{R}^2}$  est aussi possible pour une configuration dans l’espace  $\mathcal{C}_{libre}$  En l’occurrence la balise spéciale  $l_{0,\theta_0}^0$ .

#### 5.3.4.2 Sous-espace de contact visible depuis une balise $l_{i,\theta}^k$ sur $\mathcal{CB}_{j,\theta}$

Le sous-espace de contact visible  $\mathcal{CB}_{j,\theta}^{viscontact}(l_{i,\theta}^k)$  depuis la balise  $l_{i,\theta}^k$  est défini comme l’ensemble d’intervalles atteignables des arêtes visibles appartenant à la tranche  $\mathcal{CB}_{j,\theta}$  du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_j$  (figure 5.18)<sup>4</sup>.

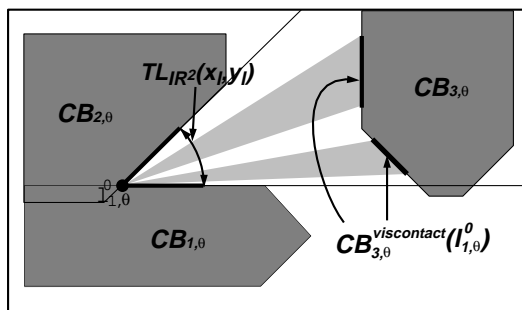


FIG. 5.18 – Sous-espace de contact visible depuis la balise initiale  $l_{1,\theta}^0$  sur le contour  $\partial\mathcal{CB}_{3,\theta}$ .

4. Le sous-espace de contact visible  $\mathcal{CB}_{j,\theta_0}^{viscontact}(l_{0,\theta_0}^0)$  peut être aussi défini.

### Discrétisation du sous-espace $\mathcal{CB}_{j,\theta}^{viscontact}(l_{i,\theta}^k)$

Le sous-espace continu à morceaux  $\mathcal{CB}_{j,\theta}^{viscontact}(l_{i,\theta}^k)$  peut être discrétisé par un ensemble discret de configurations (attracteurs)  $VAT_{j,\theta}^k$ . Cet ensemble est défini comme l'union des discrétisations de chacun des intervalles appartenant au sous-espace  $\mathcal{CB}_{j,\theta}^{viscontact}(l_{i,\theta}^k)$ . La discrétisation d'un intervalle contient un nombre fixe d'attracteurs définis de manière aléatoire à l'intérieur de celui-ci. Un attracteur dans l'ensemble  $VAT_{j,\theta}^k$  est noté  $v_{j,\theta}^{k,m}$ . On associe à chaque attracteur  $v_{j,\theta}^{k,m}$  la balise  $l_{i,\theta}^k$  qui l'a généré.

#### 5.3.4.3 Sous-espace de contact visible depuis une balise $l_{i,\theta}^k$ sur l'ensemble de $\mathcal{CB}_{j,\theta}$

Le sous-espace de contact visible  $\mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k)$  depuis la balise  $l_{i,\theta}^k$  est défini comme l'union des sous-espaces de contact visible  $\mathcal{CB}_{j,\theta}^{viscontact}(l_{i,\theta}^k)$  sur chaque  $\mathcal{C}$ -obstacle  $\mathcal{CB}_j$  dans l'espace  $\mathcal{C}$  :<sup>5</sup>

$$\mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k) = \bigcup_{\{j \mid \mathcal{CB}_j \in \mathcal{C}\}} \mathcal{CB}_{j,\theta}^{viscontact}(l_{i,\theta}^k) \quad (5.27)$$

### Discrétisation du sous-espace $\mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k)$

La discrétisation  $VAT_{\theta}^k$  du sous-espace continu à morceaux  $\mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k)$  est définie comme l'union des discrétisations  $VAT_{j,\theta}^k$  de chaque sous-espace  $\mathcal{CB}_{j,\theta}^{viscontact}(l_{i,\theta}^k)$  dans l'espace  $\mathcal{C}$  :

$$VAT_{\theta}^k = \bigcup_{\{j \mid \mathcal{CB}_j \in \mathcal{C}\}} VAT_{j,\theta}^k \quad (5.28)$$

#### 5.3.4.4 Sous-espace de contact visible sur l'ensemble de $\mathcal{C}$ -obstacles $\mathcal{CB}_i$

Soit  $\mathcal{L}$  l'ensemble des balises  $l_{i,\theta}^k$  dans l'espace  $\mathcal{C}$  (y compris la balise spéciale  $l_{0,\theta_0}^0$ ). Le sous-espace de contact visible  $\mathcal{CB}^{viscontact}$  associé aux  $\mathcal{C}$ -obstacles  $\mathcal{CB}_i$  est défini comme l'union des sous-espaces de contact visible  $\mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k)$  de chaque balise  $l_{i,\theta}^k$  dans  $\mathcal{L}$  ainsi :

$$\mathcal{CB}^{viscontact} = \bigcup_{l_{i,\theta}^k \in \mathcal{L}} \mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k) \quad (5.29)$$

### Discrétisation du sous-espace $\mathcal{CB}^{viscontact}$

La discrétisation  $VAT$  du sous-espace continu à morceaux  $\mathcal{CB}^{viscontact}$  est définie comme l'union des discrétisations  $VAT_{\theta}^k$  des sous-espaces  $\mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k)$  engendrés par chaque balise  $l_{i,\theta}^k$  dans  $\mathcal{L}$  (figure 5.19) :

$$VAT = \bigcup_{l_{i,\theta}^k \in \mathcal{L}} VAT_{\theta}^k \quad (5.30)$$

---

5. Le sous-espace de contact visible  $\mathcal{CB}_{\theta_0}^{viscontact}(l_{0,\theta_0}^0)$  peut être aussi défini.

### Exploration du sous-espace $\mathcal{CB}^{viscontact}$

Le sous-espace de contact continu à morceaux  $\mathcal{CB}^{viscontact}$  peut être exploré incrémentalement depuis la balise spéciale  $l_{0,\theta_0}^0$  ou une balise initiale  $l_{i,\theta}^0$ . L'exploration se fait à partir d'une discrétisation  $VAT$  de l'espace  $\mathcal{CB}^{viscontact}$ . A chaque pas  $k$  de l'exploration une nouvelle balise  $l_{j,\theta}^k$  est définie. La définition d'une balise  $l_{j,\theta}^k$  passe par deux étapes dans l'exploration. D'abord une étape de définition d'un sous-but  $\nu_{j,\theta}^k \in \mathcal{CB}_{\theta}^{viscontact}(l_{h,\theta}^g)$  à atteindre depuis une balise  $l_{h,\theta}^g$  déjà définie. Le choix du sous-but  $\nu_{j,\theta}^k$  se fait parmi un ensemble d'attracteurs visibles dans le sous-espace  $\mathcal{CB}^{viscontact}$  dont la définition ne tient pas compte des obstructions possibles le long de la direction d'approche depuis la balise  $l_{h,\theta}^g$ . Pour cette raison il se peut que le sous-but  $\nu_{j,\theta}^k$  choisi ne soit pas atteint depuis la balise  $l_{h,\theta}^g$ . Il est donc nécessaire de vérifier l'"atteignabilité" du sous-but choisi. Pour cela une deuxième étape dans l'exploration est considérée. Elle consiste à approcher le sous-but  $\nu_{j,\theta}^k$  en ligne droite en partant de la balise  $l_{h,\theta}^g$ . Cette approche suit le plus court chemin entre une balise et un sous-but. La configuration atteinte après l'approche dans le cas de succès ou d'échec devient la nouvelle balise  $l_{j,\theta}^k$  dans le sous-espace  $\mathcal{CB}^{viscontact}$ .

### Choix d'un sous-but $\nu_{j,\theta}^k$ dans le sous-espace $\mathcal{CB}^{viscontact}$

Soit  $q_0 = (x_0, y_0, \theta_0)$  la configuration initiale du robot ponctuel  $q_A$ . Initialement si  $q_0 \in \mathcal{C}_{libre}$  alors  $\mathcal{L} = \{l_{0,\theta_0}^0\}$  (cas spécial §5.3.2). Sinon  $q_0 \in \mathcal{C}_{contact}$  et  $\mathcal{L} = \{l_{i,\theta_0}^0\}$ . Dorénavant la balise initiale sera  $l_{i,\theta_0}^0$  dont  $i$  prend la valeur 0 si  $q_0$  appartient à  $\mathcal{C}_{libre}$  et l'identificateur de l'obstacle en contact dans le cas contraire. On a donc  $\mathcal{CB}^{viscontact} = \mathcal{CB}_{\theta_0}^{viscontact}(l_{i,\theta_0}^0)$  et  $VAT = VAT_{\theta_0}^0$ . Le premier sous-but  $\nu_{j,\theta_0}^1$  correspond à l'attracteur  $v_{j,\theta_0}^{0,m} \in VAT$  le plus éloigné de la balise  $l_{i,\theta_0}^0$  (figure 5.19). La distance entre la balise  $l_{i,\theta_0}^0$  et un attracteur  $v_{j,\theta_0}^{0,m}$  correspond à la distance euclidienne entre les positions des deux configurations. Cette distance est notée :  $\|l_{i,\theta_0}^0 - v_{j,\theta_0}^{0,m}\|$ . Le choix du premier sous-but  $\nu_{j,\theta_0}^1$  passe donc par la résolution du problème d'optimisation :

$$\begin{cases} \text{choisir } \nu_{j,\theta_0}^1 \text{ comme l'attracteur } v_{j,\theta_0}^{0,m} \text{ maximisant :} \\ \max_{v_{j,\theta_0}^{0,m} \in VAT_{\theta_0}^0} \|l_{i,\theta_0}^0 - v_{j,\theta_0}^{0,m}\| \end{cases} \quad (5.31)$$

Le problème d'approche ainsi posé consiste donc à atteindre le sous-but  $\nu_{j,\theta_0}^1$  depuis  $l_{i,\theta_0}^0$  en suivant le segment de droite les reliant.

Dans le cas général on va déterminer le sous-but  $\nu_{j,\theta}^k$  dans le sous-espace  $\mathcal{CB}^{viscontact}$  lorsqu'un ensemble de balises  $\mathcal{L} = \{l_{i,\theta_0}^0, \dots, l_{n,\theta_{k-1}}^{k-1}\}$  a été défini dans le sous-espace. Le sous-but  $\nu_{j,\theta}^k$  correspond à l'attracteur  $v_{j,\theta}^{g,m} \in VAT$  le plus éloigné de l'ensemble de balises  $\mathcal{L}$ . Intuitivement le sous-but  $\nu_{j,\theta}^k$  correspond à une configuration à l'intérieur de la région moins explorée du sous-espace  $\mathcal{CB}^{viscontact}$ . Le choix de  $\nu_{j,\theta}^k$  passe donc par la

résolution du problème d'optimisation :

$$\begin{cases} \text{choisir } \nu_{j,\theta}^k \text{ comme l'attracteur } v_{j,\theta}^{g,m} \text{ maximisant :} \\ \max_{v_{j,\theta}^{g,m} \in VAT} \min_{l_{i,\theta}^n \in \mathcal{L}} \|l_{i,\theta}^n - v_{j,\theta}^{g,m}\| \end{cases} \quad (5.32)$$

L'attracteur  $v_{j,\theta}^{g,m}$  choisi comme sous-but  $\nu_{j,\theta}^k$  appartient au sous-ensemble d'attracteurs  $VAT_{j,\theta}^g$  défini par la balise  $l_{h,\theta}^g$ . Le nouveau problème d'approche posé consiste donc à atteindre le sous-but  $\nu_{j,\theta}^k$  depuis  $l_{h,\theta}^g$  en suivant le segment de droite les reliant. Il faut noter que le sous-but  $\nu_{j,\theta}^k$  et la balise  $l_{h,\theta}^g$  appartiennent à des tranches de  $\mathcal{C}$ -obstacles différents (resp.  $\mathcal{CB}_{j,\theta}$  et  $\mathcal{CB}_{h,\theta}$ ) ayant la même orientation  $\theta$ .

#### Approche d'un sous-but $\nu_{j,\theta}^k$ dans le sous-espace $\mathcal{CB}^{viscontact}$

La stratégie d'approche d'un sous-but  $\nu_{j,\theta}^k$  ( $k > 0$ ) depuis une balise  $l_{h,\theta}^g$  ( $0 \leq h \leq n, 0 \leq g \leq k-1$ ) suit une direction en translation qui lâche le contact présent sur la balise  $l_{h,\theta}^g$ . Le robot ponctuel  $q_A$  passe ainsi dans l'espace  $\mathcal{C}_{libre}$ . L'incertitude de contrôle est prise en compte le long du déplacement en translation ce qui fait augmenter l'incertitude de position. La valeur d'incertitude peut être estimée à l'aide de l'équation 2.4. Nous rappelons que l'incertitude de position dans l'espace  $\mathcal{C}_{libre}$  est un disque centré sur une configuration nominale et de rayon  $\rho$ . De plus le disque devient un intervalle lorsqu'une arête est détectée. Par définition un sous-but choisi est un attracteur visible et atteignable sous les contraintes d'incertitude de position et de contrôle depuis la balise de départ. Cependant la définition des attracteurs visibles ne prend pas en compte les collisions potentielles le long du chemin d'approche. S'il n'y a pas d'obstructions le long du segment de droite entre  $l_{h,\theta}^g$  et  $\nu_{j,\theta}^k$  le robot ponctuel  $q_A$  s'arrête au contact lorsqu'il atteint le sous-but. La nouvelle balise est donc définie comme  $l_{j,\theta}^k = \nu_{j,\theta}^k$  (figure 5.19). En cas d'obstructions la stratégie d'approche suit le comportement de la fonction de potentiel fictif qui tente de s'approcher du sous-but en glissant en contact sur le(s)  $\mathcal{C}$ -obstacle(s) interposés. Si le sous-but  $\nu_{j,\theta}^k$  n'est pas atteint la nouvelle balise  $l_{i,\theta}^k$  correspond à une configuration de minimum local pour la stratégie d'approche partant de la balise  $l_{h,\theta}^g$  vers le sous-but  $\nu_{j,\theta}^k$ .

Pour chaque nouvelle balise  $l_{j,\theta}^k$  (resp.  $l_{i,\theta}^k$ ) le sous-ensemble de contact visible  $\mathcal{CB}_{\theta}^{viscontact}(l_{j,\theta}^k)$  (resp.  $\mathcal{CB}_{\theta}^{viscontact}(l_{i,\theta}^k)$ ) et l'ensemble de la discrétisation  $VAT_{\theta}^k$  sont déterminés. De plus l'ensemble de balises  $\mathcal{L}$  le sous-espace de contact visible  $\mathcal{CB}^{viscontact}$  et l'ensemble  $VAT$  sont mis à jour.

#### Algorithme de la stratégie d'exploration du sous-espace $\mathcal{CB}^{viscontact}$

ENTRÉES :

- $\mathcal{L} = \{l_{i,\theta_0}^0\} \cup \mathcal{L}_1 \cup \dots \cup \mathcal{L}_j \cup \dots \cup \mathcal{L}_n$  la liste des balises dans l'espace  $\mathcal{C}$  où  $\mathcal{L}_j$  est la liste des balises sur le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_j$ ;
- $\eta$  l'incertitude de contrôle sur les déplacements en translation;

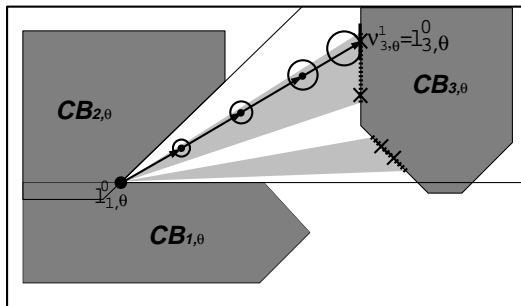


FIG. 5.19 – Ensemble d'attracteurs visibles appartenant à la discrétisation du sous-espace initial  $\mathcal{CB}_{3,\theta}^{viscontact}(l_{1,\theta}^0)$ . L'attracteur choisi comme sous-but visible  $\nu_{3,\theta}^1$  est montré. Cet attracteur devient la balise  $l_{3,\theta}^0$  suite à l'approche visible depuis la balise  $l_{1,\theta}^0$ .

- $\epsilon_t$  le pas de discrétisation en translation.

#### SORTIES :

- $l_{j,\theta_f}^k$  la nouvelle balise définie dans  $\mathcal{CB}_j$ ;
- $(\rho_{l_{j,\theta_f}^k}, \psi_{l_{j,\theta_f}^k})$  l'incertitude de position et d'orientation sur la nouvelle balise  $l_{j,\theta_f}^k$ ;
- $l_{h,\theta_f}^g$  la balise explorée dans  $\mathcal{L}$ ;
- $\tau_{vis}$  le chemin *valide* et *robuste* reliant les balises  $l_{h,\theta_f}^g$  et  $l_{j,\theta_f}^k$ ;
- $\mathcal{L}$  la nouvelle liste des balises dans l'espace  $\mathcal{C}$ .

#### ALGORITHME :

1. Pour le sous-ensemble de balises  $\mathcal{L}$  déterminer dans l'ensemble  $VAT$  le meilleur attracteur visible  $\nu_{j,\theta_f}^k$  (§5.3.4.4).
2. Déterminer la balise  $l_{h,\theta_f}^g$  dans  $\mathcal{L}$  qui a engendré l'attracteur visible  $\nu_{j,\theta_f}^k$ .
3. Appliquer la stratégie d'approche dans l'espace  $\mathcal{C}_{valide}$  depuis la balise  $l_{h,\theta_f}^g$  vers le sous-but  $\nu_{j,\theta_f}^k$ . La configuration atteinte par l'approche devient la nouvelle balise  $l_{j,\theta_f}^k$ . Cette balise est accompagnée des estimations d'incertitude de position  $\rho_{l_{j,\theta_f}^k}$  et d'orientation  $\psi_{l_{j,\theta_f}^k}$ . Le chemin  $\tau_{vis}$  reliant les balises  $l_{h,\theta_f}^g$  et  $l_{j,\theta_f}^k$  est retourné.
4. Mettre à jour  $\mathcal{L}$  (i.e.  $\mathcal{L} = \{l_{i,\theta_0}^0\} \cup \mathcal{L}_1 \cup \dots \cup \mathcal{L}_j \cup \{l_{j,\theta_f}^k\} \cup \dots \cup \mathcal{L}_n$ ).

## 5.4 L'approche de planification

L'approche de planification que nous proposons pour résoudre le problème énoncé résulte de l'application de la fonction de potentiel fictif adaptée dans l'espace  $\mathcal{C}_{valide}$  combinée avec les stratégies d'exploration de l'espace  $\mathcal{C}_{contact}$  pour s'échapper des minima locaux.

Dans la suite l'intégration de la fonction de potentiel avec les stratégies d'exploration est traitée ce qui donne lieu à l'approche proposée. Initialement la façon d'appliquer chaque composante dans l'approche est résumée. Ensuite la représentation de l'espace exploré pendant l'application de l'approche est expliquée. Puis l'ensemble de l'approche est montré à l'aide d'un modèle basé sur la coopération de tâches. Finalement un algorithme séquentiel simplifié de l'implantation développée de l'approche est donné. Cette implantation nous a permis de valider l'approche pour des instantiations différentes du problème posé. Les principaux résultats sont montrés dans le chapitre 6.

### 5.4.1 Les composantes de l'approche de planification

#### 5.4.1.1 Fonction de potentiel fictif

Le but de cette fonction est d'atteindre la configuration but  $q_f$  dans  $\mathcal{C}_{valide}$  depuis une configuration de départ  $q$  dans  $\mathcal{C}_{valide}$ . Premièrement cette fonction est appliquée depuis la configuration initiale  $q_0$  du problème de planification à résoudre. Deuxièmement elle est appliquée depuis chaque balise  $l_{i,\theta}^k$  définie par les stratégies d'exploration. L'application de cette fonction peut aboutir soit à un succès soit à un échec. Le succès garantit que la configuration but a été atteinte sous les contraintes d'incertitudes géométriques imposées par le problème. L'échec peut se présenter sous un des deux cas de figure suivants. Le premier correspond à l'atteinte de la configuration but sous des estimations d'incertitude qui sont au-dessus des valeurs maximales acceptées par le problème. Le deuxième cas correspond à la détection d'un minimum local de la fonction de potentiel qui bloque le robot ponctuel  $q_A$  sur le chemin vers la configuration but  $q_f$ . Un minimum local correspond toujours à une configuration de contact. Il peut être défini soit sur une surface d'un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  soit entre deux  $\mathcal{C}$ -obstacles  $\mathcal{CB}_i$  et  $\mathcal{CB}_j$ . Tout minimum local définit une balise sur son(ses)  $\mathcal{C}$ -obstacle(s) en contact. Pour leur part les balises sont des configurations de référence qui guident les stratégies d'exploration de l'espace  $\mathcal{C}_{contact}$ .

#### 5.4.1.2 Stratégie d'exploration de l'espace de contact adjacent

Le but de cette stratégie est d'explorer de manière discrète le sous-espace de contact continu correspondant à un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ . Cette stratégie peut être déclenchée sur chaque  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  où une balise initiale  $l_{i,\theta}^0$  est définie. A chaque pas d'itération de l'exploration une nouvelle balise est définie sur le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ . Celle-ci devient une nouvelle configuration de départ pour la fonction de potentiel fictif.



### 5.4.1.3 Stratégie d'exploration de l'espace de contact visible

Le but de cette stratégie est d'atteindre des sous-espaces de l'espace de contact  $\mathcal{C}_{contact}$  qui ne sont pas atteignables par l'exploration de l'espace de contact adjacent. Les sous-espaces d'intérêt correspondent aux surfaces atteignables  $\Gamma$  depuis des balises déjà définies  $\Gamma$  des  $\mathcal{C}$ -obstacles  $\mathcal{CB}_j$  qui ne sont pas encore explorés par l'exploration adjacente. Cette stratégie est déclenchée après la définition de la première balise dans l'espace valide  $\mathcal{C}_{valide}$ . Puisque la configuration initiale  $q_0$  est considérée comme une balise (§5.3.2)  $\Gamma$  cette stratégie peut être déclenchée depuis le début de l'approche. A chaque pas d'itération de l'exploration  $\Gamma$  une nouvelle balise est définie sur un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_j$  (si c'est possible). Celle-ci devient une nouvelle configuration de départ pour la fonction de potentiel fictif.

## 5.4.2 Représentation de l'exploration de l'espace $\mathcal{C}_{valide}$

L'exploration de l'espace  $\mathcal{C}_{valide}$  pendant l'application de l'approche de planification est représentée par un arbre orienté et étiqueté  $\mathcal{T}$  construit au fur et à mesure de l'exploration même.

### 5.4.2.1 Les nœuds de l'arbre $\mathcal{T}$

Les nœuds de l'arbre  $\mathcal{T}$  correspondent aux balises qu'elles soient définies par la fonction de potentiel fictif ou par les stratégies d'exploration de l'espace  $\mathcal{C}_{contact}$ .

Le nœud initial de l'arbre  $\mathcal{T}$  - sa racine - correspond à la balise  $l_{i,\theta_0}^0$  définie par la configuration initiale  $q_0$  (§5.3.2).

L'application de la fonction de potentiel fictif depuis une balise de départ  $l_{h,\theta}^g$  vers la configuration but  $q_f$  peut aboutir soit à un succès  $\Gamma$  soit à un échec. En cas de succès  $\Gamma$  le nœud dans  $\mathcal{T}$  correspondant à  $l_{h,\theta}^g$  est marqué comme un nœud de succès. Par contre  $\Gamma$  en cas d'échec  $\Gamma$  la dernière configuration de contact atteignable  $q_{dern-att}$  définit une nouvelle balise sur les  $\mathcal{C}$ -obstacle(s) en contact  $\Gamma$  et le nœud respectif est créé dans l'arbre  $\mathcal{T}$ .

La balise résultat  $l_{j,\theta}^k$  de l'approche d'un sous-but adjacent  $\beta_{i,\theta}^k$  ou visible  $\nu_{j,\theta}^k$  en appliquant la stratégie d'exploration appropriée depuis une balise de départ  $l_{h,\theta}^g$  définit aussi un nœud dans l'arbre  $\mathcal{T}$ .

### 5.4.2.2 Les arcs de l'arbre $\mathcal{T}$

Toute nouvelle balise  $l_{j,\theta}^k$   $\Gamma$  mise à part la balise initiale  $l_{i,\theta_0}^0$   $\Gamma$  a été atteinte par l'application soit de la fonction de potentiel  $\Gamma$  soit d'une stratégie d'exploration depuis une balise de départ  $l_{h,\theta}^g$ . Les nœuds respectifs de ces balises dans l'arbre  $\mathcal{T}$  sont liés par deux arcs orientés : un arc positif depuis la balise de départ vers la balise d'arrivée  $\Gamma$  et un arc négatif dans le sens inverse. De plus  $\Gamma$  l'arc positif est étiqueté avec le chemin

*valide* et *robuste* reliant les balises. Le type d'approche appliquée pour passer du départ à l'arrivée et la configuration but cherchée par cette approche.

Les arcs positifs permettent de reconstruire le chemin suivi par le robot ponctuel  $q_A$  depuis la balise initiale  $l_{i,\theta_0}^0$  vers toute balise définie dans l'espace  $\mathcal{C}_{valide}$ . Les arcs négatifs permettent de revenir en arrière depuis une balise définie dans l'arbre  $\mathcal{T}$  vers la balise initiale  $l_{i,\theta_0}^0$ .

### 5.4.3 Modèle réparti de l'approche de planification

L'intégration des composantes dans l'approche de planification est expliquée à l'aide d'un modèle réparti de tâches qui permet de présenter les composantes et leur interaction indépendamment de l'implantation sur ordinateur. L'interaction des tâches exige la définition d'un protocole de communication de messages entre elles. Ce modèle peut donner lieu à un premier algorithme parallèle de notre approche de planification.

Le modèle est composé d'un ensemble de tâches, chacune spécialisée sur la fonctionnalité d'une composante de l'approche. Ainsi, une tâche  $\mathcal{FPF}$  exécute la fonction de potentiel fictif, une tâche  $\mathcal{EA}_i$  exécute l'exploration de l'espace de contact adjacent dans le  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  et une tâche  $\mathcal{EV}$  exécute l'exploration de l'espace de contact visible pour l'ensemble de  $\mathcal{C}$ -obstacles (figure 5.20). Dans la suite, une description par tâche en incluant ses interactions avec les autres tâches est présentée.

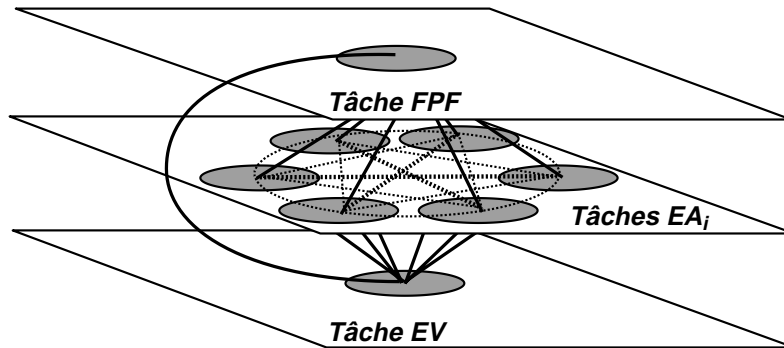


FIG. 5.20 – Modèle réparti correspondant à la sous-classe d'instantiations du problème posé dont l'espace  $\mathcal{C}$  est composé de six  $\mathcal{C}$ -obstacles. Dans le modèle toutes les tâches peuvent communiquer entre elles. Une liaison entre deux tâches d'une même couche est montrée avec un segment en pointillé, et entre deux tâches de couches différentes avec un segment continu.

Du point de vue séquentiel, l'ordre d'application des stratégies d'exploration et la façon de les alterner donnent lieu à des algorithmes séquentiels différents de l'approche.

### 5.4.3.1 La tâche $\mathcal{FPF}$

Cette tâche est dédiée à l'application de la fonction de potentiel fictif dans l'espace valide  $\mathcal{C}_{valide}$  depuis une balise de départ jusqu'à la configuration but  $q_f$ . Les balises de départ sont reçues des différentes tâches  $\mathcal{EA}_i$  et  $\mathcal{EV}$ . Un résultat de succès permet de reconstruire une solution du problème posé depuis la balise initiale  $l_{i,\theta_0}^0$ . Par contre une configuration de minimum local est communiquée à la(aux) tâche(s)  $\mathcal{EA}_i$  liée(s) au(x)  $\mathcal{C}$ -obstacle(s) en contact. Elle est aussi communiquée à la tâche  $\mathcal{EV}$ .

### 5.4.3.2 Une tâche $\mathcal{EA}_i$

Une telle tâche est dédiée à l'exploration de l'espace de contact adjacent du  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$ . Chaque nouvelle balise définie est communiquée aux tâches  $\mathcal{FPF}$  et  $\mathcal{EV}$ . De plus si une balise est partagée avec un deuxième  $\mathcal{C}$ -obstacle  $\mathcal{CB}_j$  elle est communiquée à la tâche  $\mathcal{EA}_j$ .

### 5.4.3.3 La tâche $\mathcal{EV}$

Cette tâche est dédiée à l'exploration de l'espace de contact visible sur l'ensemble des  $\mathcal{C}$ -obstacles dans l'espace  $\mathcal{C}$ . Chaque nouvelle balise définie est communiquée d'abord à la tâche  $\mathcal{FPF}$  et puis à la(aux) tâche(s)  $\mathcal{EA}_i$  liée(s) au(x)  $\mathcal{C}$ -obstacle(s)  $\mathcal{CB}_i$  en contact.

### 5.4.3.4 Déclenchement du modèle réparti

Le modèle réparti de l'approche est déclenché par la communication de la balise initiale  $l_{i,\theta_0}^0$  aux tâches  $\mathcal{FPF}$  et  $\mathcal{EV}$ . De plus si  $q_0$  est dans  $\mathcal{C}_{contact}$  la balise initiale doit être communiquée à la(aux) tâche(s)  $\mathcal{EA}_i$  liée(s) au(x)  $\mathcal{C}$ -obstacle(s)  $\mathcal{CB}_i$  en contact.

## 5.4.4 Un algorithme séquentiel de l'approche

Des différents algorithmes séquentiels possibles de notre approche de planification nous avons mis en œuvre un algorithme qui donne la préférence à l'exploration de l'espace de contact adjacent plutôt qu'à l'exploration de l'espace de contact visible. Ceci est justifié par le fait que les incertitudes géométriques restent davantage réduites pendant l'exploration de l'espace de contact adjacent. L'algorithme est composé de trois fonctions dont chacune implémente une des composantes de base de l'approche. Les contraintes d'incertitudes géométriques de position d'orientation et de contrôle sur le robot ponctuel  $q_A$  sont prises en compte dans chaque fonction de l'algorithme. Un algorithme simplifié est présenté ci-dessous. Cependant un algorithme plus détaillé est explicité en annexe §A.2.

ENTRÉES :

- $q_0$  la configuration initiale et  $(\rho_0, \psi_0)$  les incertitudes de position et d'orientation associées;

- $q_f$  la configuration but  $\Gamma$  et  $(\rho_f, \psi_f)$  les incertitudes maximales de position et d'orientation acceptées;
- $\eta, k_t$  et  $k_r$  les paramètres d'incertitude propres au robot  $\mathcal{A}$ ;
- $\epsilon_t$  et  $\varrho$  les paramètres propres à la fonction de potentiel fictif.

SORTIES :

- Valeur **SUCCÈS** si la configuration  $q_f$  est atteinte depuis la configuration  $q_0$  en respectant les estimations d'incertitude maximales  $\rho_f$  et  $\psi_f$ .
- Valeur **ECHEC** dans le cas contraire.

ALGORITHME :

1. Initialiser les listes  $\mathcal{L}$  et  $\mathcal{L}_i\Gamma$  et de l'arbre d'exploration  $\mathcal{T}$  avec la balise initiale  $l_{i,\theta_0}^0$ .
2. Chercher à atteindre la configuration but  $q_f$  depuis la balise initiale  $l_{i,\theta_0}^0$  avec la **Fonction de Potentiel Fictif**.  
En cas de **SUCCÈS** retourner **SUCCÈS**. Sinon  $\Gamma$  définir une nouvelle balise avec la configuration de blocage  $q_c$  et ses estimations d'incertitude  $(\rho_c \mathbb{F}\psi_c)$ .
3. Pour chaque  $\mathcal{C}$ -obstacle  $\mathcal{CB}_i$  où il y a au moins une balise définie :
  - (a) Appliquer l'**Exploration de l'espace de contact adjacent** sur  $\mathcal{CB}_i$  afin de définir une nouvelle balise  $l_{i,\theta_f}^k$  avec ses estimations d'incertitude  $(\rho_{l_{i,\theta_f}^k} \mathbb{F}\psi_{l_{i,\theta_f}^k})$ .
  - (b) Chercher à atteindre la configuration but  $q_f$  depuis la balise  $l_{i,\theta_f}^k$  avec la **Fonction de Potentiel Fictif**.  
En cas de **SUCCÈS** retourner **SUCCÈS**. Sinon  $\Gamma$  définir une nouvelle balise avec la configuration de blocage  $q_c$  et ses estimations d'incertitude  $(\rho_c \mathbb{F}\psi_c)$ .  
Itérer la boucle en 3.
4. (a) Appliquer l'**Exploration de l'espace de contact visible** sur l'ensemble de  $\mathcal{C}$ -obstacles  $\{\mathcal{CB}_i\}$  afin de définir une nouvelle balise  $l_{j,\theta_f}^k$  avec ses estimations d'incertitude  $(\rho_{l_{j,\theta_f}^k} \mathbb{F}\psi_{l_{j,\theta_f}^k})$ .
- (b) Chercher à atteindre la configuration but  $q_f$  depuis la balise  $l_{j,\theta_f}^k$  avec la **Fonction de Potentiel Fictif**.  
En cas de **SUCCÈS** retourner **SUCCÈS**. Sinon  $\Gamma$  définir une nouvelle balise avec la configuration de blocage  $q_c$  et ses estimations d'incertitude  $(\rho_c \mathbb{F}\psi_c)$ .  
Aller au pas 3.

### 5.4.5 Discussion sur l'approche proposée

Tout d'abord l'analyse du problème abordé nous a permis de voir de manière plus claire l'importance des incertitudes dans certains domaines de la planification en robotique tels que la planification de mouvements d'assemblage et la navigation d'un robot mobile. Un résultat issu de cette analyse correspond à l'approche proposée.

L'approche proposée quant à elle intègre deux méthodes locales : une méthode de navigation de type potentiel fictif et une méthode d'exploration incrémentale. Une adaptation de chaque méthode est proposée pour exploiter au mieux leurs avantages en faveur de la résolution du problème posé. L'intégration donne lieu à une *approche globale* possédant des avantages mais aussi des inconvénients. Les avantages que nous considérons les plus importants sont : **l'efficacité** et **le caractère incrémental** du principe de résolution. L'efficacité permet de trouver en général des solutions rapidement alors que le caractère incrémental facilite l'intégration d'une analyse explicite d'incertitude validant en permanence la robustesse des mouvements. D'ailleurs le type d'analyse géométrique appliquée sur l'incertitude donne lieu à une *approche ensembliste*.

Bien que l'approche possède des avantages intéressants il est aussi judicieux de remarquer ses inconvénients. Le premier correspond à une exploration systématique de l'espace de contact. En effet l'espace libre est surtout utilisé pour permettre au robot le passage entre composantes non connexes de l'espace de contact. Cela nous permet d'identifier un premier compromis entre la robustesse des mouvements et le degré de mobilité du robot. Notre approche favorise l'aspect robustesse. Le deuxième inconvénient correspond à une exploitation des contacts qui s'avère insuffisante dans certains cas en l'occurrence le contournement des sommets des obstacles. Cela nous permet d'identifier un deuxième compromis entre la complexité de l'analyse des contacts et la sous-classe de situations de blocage traitables. Notre approche propose une première analyse géométrique contraignant les orientations au contact. Hélas les inconvénients vont en détriment de la *complétude* de l'approche; ainsi une solution pourrait ne pas être trouvée même si celle-ci existe.

## 5.5 Transformation des chemins en plans

Une solution à une instantiation du problème posé dans §3.2 s'appuyant sur l'approche proposée est un *chemin valide et robuste*  $\tau$ . Ce chemin appartient par définition à l'espace  $\mathcal{C}_{valide}$  c'est-à-dire que  $\tau$  est défini dans l'espace libre  $\mathcal{C}_{libre}$  et/ou l'espace de contact  $\mathcal{C}_{contact}$ .

On peut définir une décomposition du chemin  $\tau$  telle que  $\tau$  soit une suite de sous-chemins  $\{\tau_1, \dots, \tau_m\}$  avec  $m$  supérieur ou égal à 1 et chaque sous-chemin  $\tau_i$  étant de type *libre*, *quasi-libre* ou *en contact*. Cette décomposition est définie de la manière

suivante :

- si  $\tau$  est un *chemin libre* ou *en contact* alors  $m = 1$  et  $\tau_1 = \tau$ ;
- dans le cas contraire  $\tau$  est décomposé en une suite  $\{\tau_1, \dots, \tau_m\}$  avec  $m$  supérieur ou égal à 1 chaque sous-chemin  $\tau_i$  étant de type *quasi-libre* ou *en contact* deux chemins consécutifs  $\tau_i$  et  $\tau_{i+1}$  ayant un type différent.  
En d'autres termes  $\tau$  définit une suite alternée de sous-chemins *quasi-libres* et *en contact*.

On peut définir maintenant une décomposition d'un sous-chemin  $\tau_i$  telle que  $\tau_i$  soit une suite de sous-chemins  $\{\tau_{i,1}, \dots, \tau_{i,p}\}$  avec  $p$  supérieur ou égal à 1 chaque sous-chemin  $\tau_{i,j}$  étant une translation pure suivant le vecteur  $\vec{v} = (\delta x, \delta y, 0)$  une rotation pure d'angle  $\delta\theta$  autour d'un axe fixe ou une translation/rotation homogène suivant le vecteur  $\vec{v} = (\delta x, \delta y, \delta\theta)$  (choix exclusif).

A partir des décompositions précédentes le chemin  $\tau$  est redéfini comme une suite de chemins élémentaires  $\{\tau_{i,j}\}$  où chaque  $\tau_{i,j}$  peut être généré par une des primitives de contrôle  $\sigma_i$  définies dans §3.4.1. Ainsi :

- un sous-chemin  $\tau_{i,j}$  *libre* est généré par la primitive COMMANDE-LIBRE-EN-DISTANCE;
- un sous-chemin  $\tau_{i,j}$  *quasi-libre* est généré par la primitive COMMANDE-LIBRE-EN-DISTANCE si la configuration extrémité  $\tau_{i,j}(1)$  appartient à l'espace  $\mathcal{C}_{libre}$ . Dans le cas contraire si  $\tau_{i,j}$  correspond à une rotation pure alors  $\tau_{i,j}$  est généré par la primitive COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT sinon  $\tau_{i,j}$  est généré par la primitive COMMANDE-GARDÉE;
- un sous-chemin  $\tau_{i,j}$  *en contact* dont l'extrémité  $\tau_{i,j}(1)$  n'est pas un sommet d'un  $\mathcal{C}$ -obstacle est généré par la primitive COMMANDE-COMPLIANTE-EN-DISTANCE. Dans le cas contraire si le sommet correspond à une convexité  $\tau_{i,j}$  est généré par la primitive COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT. Sinon le sommet correspond à une concavité et  $\tau_{i,j}$  est généré par la primitive COMMANDE-COMPLIANTE-JUSQU'AU-CONTACT.

En résumé le chemin valide et robuste  $\tau = \{\tau_{i,j}\}$  pour le robot ponctuel  $q_{\mathcal{A}}$  est transformé en un plan robuste  $\Sigma$  défini comme une suite de commandes de contrôle  $\{\sigma_{i,j}\}$  pour le robot  $\mathcal{A}$  dont chaque commande  $\sigma_{i,j}$  est chargée de garantir le déplacement le long du sous-chemin élémentaire  $\tau_{i,j}$ .

Dans la pratique les configurations qui déterminent le début et/ou la fin des sous-chemins  $\tau_{i,j}$  sont identifiées lors de la planification. Ces configurations sont liées aux événements :

- départ de la configuration initiale  $q_0$ ;

- arrivée à la configuration but  $q_f$ ;
- passage de l'espace  $\mathcal{C}_{libre}$  à l'espace  $\mathcal{C}_{contact}$  (ou vice-versa);
- accomplissement d'un déplacement d'alignement;
- détection d'un sommet d'un  $\mathcal{C}$ -obstacle  $\mathcal{CB}_{i,\theta}$ ;
- détection d'un minimum local.

## 5.6 Conclusion

Dans ce chapitre nous avons développé les principaux aspects caractérisant notre approche à savoir : l'espace d'exploration, la fonction de navigation du robot, la stratégie de traitement de minima locaux de la fonction de navigation et l'intégration des composantes dans une approche de planification automatique. Les conclusions sur chaque aspect sont résumées ci-dessous.

- L'ESPACE D'EXPLORATION :  
l'espace d'exploration est composé de l'espace  $\mathcal{C}_{libre}$  et de l'espace  $\mathcal{C}_{contact}$ . A la différence des approches d'évitement d'obstacles, l'espace  $\mathcal{C}_{contact}$  est pris en compte pour permettre au robot d'aller au contact des obstacles et ainsi de se repositionner et/ou de se réorienter par rapport à eux. Les incertitudes de position et d'orientation sur le robot peuvent donc être réduites. L'espace  $\mathcal{C}_{contact}$  est lié à la représentation des  $\mathcal{C}$ -obstacles dans l'espace des configurations  $\mathcal{C}$ . L'approche utilise une représentation discrète des  $\mathcal{C}$ -obstacles construite incrémentalement pendant la navigation du robot.
- LA FONCTION DE NAVIGATION :  
une fonction de potentiel fictif est proposée dans l'espace des configurations  $\mathcal{C} = \mathbb{R}^2 \times [0, 2\pi]$  pour faire naviguer le robot ponctuel  $q_A$  aussi bien dans l'espace  $\mathcal{C}_{libre}$  que dans l'espace  $\mathcal{C}_{contact}$ . Les passages du robot de l'espace  $\mathcal{C}_{libre}$  vers l'espace  $\mathcal{C}_{contact}$  sont possibles sous la condition d'"atteignabilité". Cette condition est évaluée à l'aide des modèles géométriques de *projection en avant* proposés. D'autre part, les contraintes d'incertitudes géométriques de position, d'orientation et de contrôle sur le robot sont prises en compte explicitement le long des déplacements. L'effet de ces contraintes sur le robot ponctuel  $q_A$  est modélisé par une enveloppe géométrique évoluant selon l'espace de navigation. Ainsi, le robot  $q_A$  avec son enveloppe d'incertitude peut devenir un point, un disque ou un cylindre dans l'espace  $\mathcal{C}_{libre}$ ; et un point, un intervalle ou une partie connexe d'une surface d'un  $\mathcal{C}$ -obstacle dans l'espace  $\mathcal{C}_{contact}$ .
- LE TRAITEMENT DE MINIMA LOCAUX :  
la fonction de navigation ayant une nature locale, le robot ponctuel peut se voir dirigé vers des minima locaux, ceux-ci étant définis dans l'espace  $\mathcal{C}_{contact}$ . Autour

d'une configuration de minimum local  $\Gamma$  deux sous-espaces (voisinages) de contact peuvent être définis : un *sous-espace adjacent* et un *sous-espace visible*. Chaque sous-espace peut être exploré de manière incrémentale par une stratégie d'exploration appliquée sur une discrétisation de celui-ci. Le principe de chaque stratégie d'exploration est celui de la fonction **EXPLORE** de l'algorithme Fil d'Ariane (de nature locale) adapté à l'espace  $\mathcal{C}_{contact}$ . Le but de ces stratégies consiste à éloigner le robot des configurations correspondant à des minima locaux tout en gardant des incertitudes faibles. Les avantages des telles stratégies sont  $\Gamma$  d'une part  $\Gamma$  l'adaptation automatique de la résolution d'exploration à la complexité de chaque situation de minimum local détectée  $\Gamma$  et  $\Gamma$  d'autre part  $\Gamma$  l'élargissement incrémental des espaces d'exploration adjacent et visible.

– L'INTÉGRATION :

l'intégration de la fonction de potentiel fictif proposée avec les stratégies d'exploration donne lieu à une approche de planification de chemins robustes vis-à-vis des incertitudes géométriques sur le robot. L'approche obtenue est à la fois *globale* et *incrémentale*  $\Gamma$  ce qui la rend  $\Gamma$  en général  $\Gamma$  efficace.

Cependant  $\Gamma$  la méthode n'est pas *complète*. Effectivement  $\Gamma$  dans certains cas  $\Gamma$  une solution ne peut pas être générée même si elle existe. Ceci est dû premièrement  $\Gamma$  à une exploration systématique de l'espace de contact et  $\Gamma$  deuxièmement  $\Gamma$  à une stratégie d'exploitation de contacts qui n'est pas suffisante pour le traitement de minima locaux. Des améliorations de l'approche concernant ces deux aspects sont discutées dans les perspectives §7.2.





# Chapitre 6

## Validations de l'approche

---

L'approche de planification telle qu'elle est présentée permet de planifier des tâches pour un objet physique pouvant être modélisé par un objet polygonal. Les capacités de mobilité dont l'objet physique doit disposer sont d'une part la translation sur un plan et d'autre part la rotation autour de n'importe quel axe perpendiculaire au plan de déplacement. L'objet physique satisfaisant les capacités de mobilité correspond à un *objet holonome* ayant 2 d.d.l. en translation et 1 d.d.l. en rotation et n'ayant pas de contraintes sur le placement de l'axe de rotation. Deux cas de figure où l'approche s'avère applicable sont : d'une part la résolution des tâches pour une pièce polygonale manipulée sur un plan et d'autre part la résolution des tâches pour un robot mobile.

L'approche originale est applicable pour résoudre les tâches du premier type où les contacts entre la pièce manipulée et les obstacles sont possibles. Nous avons donc développé et mis au point une implantation opérationnelle en langage C pouvant tourner aussi bien sur des machines Silicon Graphics (sous la librairie graphique GL) que sur des machines SUN (sous la librairie graphique VOpenGL). Faute d'un système de contrôle fiable pour gérer les contacts sur les bras manipulateurs dont nous disposons dans notre équipe robotique la validation de ce type de tâches n'a pas pu être faite qu'au moyen de simulations sur ordinateur.

Des adaptations de la première implantation ont été nécessaires pour la rendre opérationnelle pour la résolution de tâches pour un robot mobile. De plus nous sommes parvenu à obtenir de premiers résultats expérimentaux. Pour ceci nous avons intégré l'implantation dans la plate-forme du robot mobile Hilare II du LAAS<sup>1</sup>-CNRS<sup>2</sup>.

Dans la suite l'application de l'approche pour la planification du mouvement d'une pièce manipulée est illustrée avec plusieurs exemples. Ensuite dans une deuxième partie les adaptations de l'approche originale pour traiter le problème de la planification

---

1. LAAS : Laboratoire d'Automatique et d'Analyse des Systèmes (Toulouse).  
2. CNRS : Centre National de la Recherche Scientifique.

propre à un robot mobile sont précisées et la résolution de différents problèmes dans ce cas est montrée dont une a été validée avec une expérience réelle.

## 6.1 Planification pour une pièce manipulée

Un cas de figure où l'on trouve l'objet physique sujet de la planification dans l'approche proposée est celui de la manipulation d'une pièce rigide dont les objets dans l'environnement ainsi que la pièce elle-même possèdent une géométrie  $2D\frac{1}{2}$ . Les tâches auxquelles nous nous intéressons sont celles où la pièce peut interagir avec les obstacles c'est-à-dire que des contacts entre la pièce et les obstacles sont possibles. Dans les exemples qui suivent la planification des tâches pour une pièce manipulée  $\mathcal{A}$  qui interagit avec les obstacles  $\{\mathcal{B}_i\}$  dans l'environnement  $\mathcal{W}$  est illustrée.

### Remarques

- Les commandes de mouvement composant les plans générés correspondent aux primitives de base présentées dans la section 3.4.1.
- En ce qui concerne les distances dans les commandes de mouvement elles seront mesurées en centimètres. Pour avoir une idée sur la grandeur des distances et des obstacles dans l'environnement la taille de la pièce manipulée sera donnée dans chaque exemple.  
En ce qui concerne les angles ils seront mesurés en degrés selon le sens trigonométrique.
- Par défaut les paramètres des commandes de mouvement seront exprimés dans le système de référence  $\mathcal{F}_\mathcal{W}$  de l'environnement. Cependant certains paramètres pourront être exprimés dans le système de référence  $\mathcal{F}_\mathcal{A}$  de la pièce et dans ce cas le sous-index  $\mathcal{A}$  est ajouté aux paramètres concernés.
- Dans les figures illustrant les exemples un petit triangle est placé au point de référence  $\mathcal{O}_\mathcal{A}$  du repère local  $\mathcal{F}_\mathcal{A}$  de la pièce  $\mathcal{A}$ . De plus le sommet le plus allongé de ce triangle pointe vers l'axe  $X$  positif du repère  $\mathcal{F}_\mathcal{A}$ .
- A propos de l'incertitude de position celle-ci est montrée comme un disque autour du point de référence  $\mathcal{O}_\mathcal{A}$  même en situation de contact ce qui permet de mieux apprécier son évolution. Nous rappelons que l'incertitude de position est représentée par un disque dans l'espace libre et comme un intervalle le long de la surface de contact dans l'espace de contact.
- En ce qui concerne l'incertitude d'orientation celle-ci ne sera pas évaluée explicitement pendant la planification. Cependant l'alignement de la pièce nécessaire pour faciliter les mouvements en contact est appliquée ce qui permet en plus d'annuler l'incertitude d'orientation.

- Une solution générée par le planificateur est illustrée avec deux figures :
  1. la première figure montre la trace de la pièce manipulée  $\mathcal{A}$  dans son environnement le long du chemin généré;
  2. la deuxième figure montre les configurations initiale et finale nominales de la pièce  $\mathcal{A}$  avant et après l'exécution de chaque commande dans le plan de commandes généré. Pour chaque commande un trait de liaison montre le chemin nominal qui suit la pièce  $\mathcal{A}$  entre ses deux configurations extrêmes.
- Les temps CPU utilisés pour déterminer la solution des exemples présentés ont été mesurés pour une implantation opérationnelle de l'algorithme séquentiel de l'approche lancé sur une machine SGI de type INDY. Pour chaque exemple des temps correspondant à cinq exécutions du planificateur générant une solution semblable à celle illustrée ont été pris. A partir de ces mesures trois temps sont présentés : (1) le temps moyen des cinq mesures ( $t_{moyen}$ ) (2) le temps minimum ( $t_{min}$ ) (3) le temps maximum ( $t_{max}$ ). Ces temps sont donnés en millisecondes (ms).

### 6.1.1 Problème 1. Attraction sans blocage (simulation)

#### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.1). La taille de la pièce  $\mathcal{A}$  est  $130 \times 80 \text{ cm}^2$ .

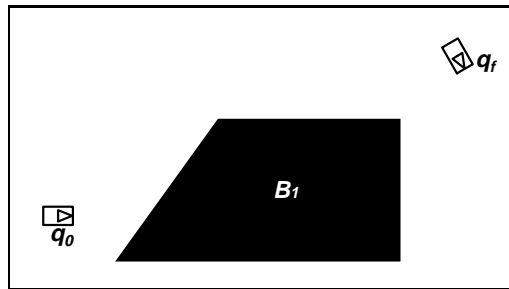


FIG. 6.1 – Définition de la tâche à résoudre.

#### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 10 cm.
- $\eta$  l'incertitude de contrôle : 2.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 25 cm.
- $\varrho$  l'angle de rupture de contact : 45 deg.

## Illustrations :

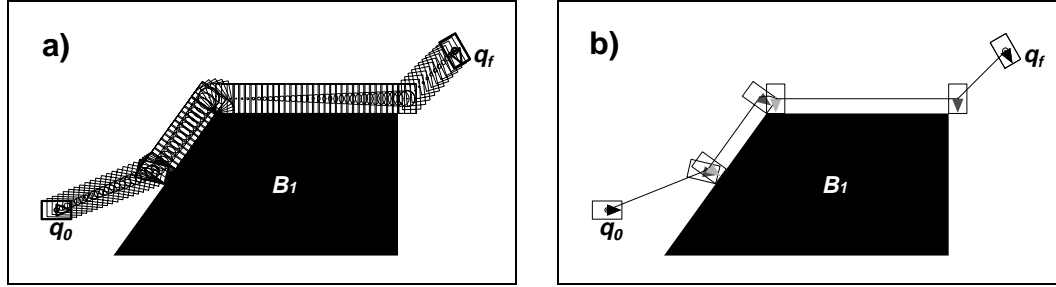


FIG. 6.2 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.

Plan robuste à appliquer sur la pièce  $\mathcal{A}$

1. COMMANDE-GARDÉE  
( $\vec{v}_{\mathcal{A}} = (0.928, 0.371, -0.296)$ ,  $\text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}$ ,  $d = 455.6$ ,  $\theta = -13.52$ ,  $CV_{\mathcal{A}}(s_1^{\mathcal{A}})$ ,  
 $\vec{f}_{\text{seuil}\mathcal{A}} = (-0.925, 0.378)$ )
2. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
(sens = négatif,  $\text{axe}_{\mathcal{A}} = (65.0, -40.0)$ ,  $\theta = -22.22$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{\text{seuil}\mathcal{A}} = (-1.0, 0.0)$ )
3. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
( $\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0)$ ,  $d = 381.8$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{\text{seuil}\mathcal{A}} = (-1.0, 0.0)$ )
4. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
(sens = négatif,  $\text{axe}_{\mathcal{A}} = (64.99, -40.00)$ ,  $\theta = -54.24$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{\text{seuil}\mathcal{A}} = (-1.0, 0.0)$ )
5. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
( $\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0)$ ,  $d = 800.0$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{\text{seuil}\mathcal{A}} = (-1.0, 0.0)$ )
6. COMMANDE-LIBRE-EN-DISTANCE  
( $\vec{v}_{\mathcal{A}} = (-0.707, 0.707, 1.01)$ ,  $\text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}$ ,  $d = 296.9$ ,  $\theta = 29.99$ )

## Commentaires :

La solution proposée joint les configurations  $q_0$  et  $q_f$  en contournant en contact l'obstacle  $B_1$ . Le mouvement d'alignement est appliqué lors de la détection du premier contact ainsi qu'au passage entre surfaces adjacentes (figure 6.2).

## Evaluation de la solution :

- Nombre d'obstacles convexes définissant l'environnement : 1
- Nombre de commandes dans le plan : 6
- Temps de calcul du chemin : 94 ms ( $t_{moyen}$ )  $\Gamma$  80 ms ( $t_{min}$ )  $\Gamma$  100 ms ( $t_{max}$ )

- Estimation de la distance parcourue par la pièce : 1934.3 cm.
- Estimation de l'incertitude finale de position : 12.9 cm.

### 6.1.2 Problème 2. Insertion d'un goujon dans une concavité (simulation)

Le but de cet exemple est d'illustrer l'effet de l'incertitude de position sur l'accomplissement d'une tâche élémentaire. Pour ceci nous allons présenter la résolution d'un même problème de base sous des contraintes différentes. La taille de la pièce est  $80 \times 50 \text{ cm}^2$  et celle de la concavité est  $80 \times 60 \text{ cm}^2$ .

#### Problème 2a. Insertion élémentaire sans incertitude

##### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.3).

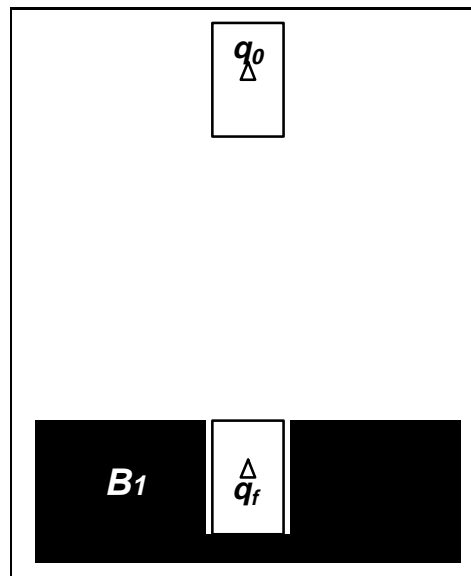


FIG. 6.3 – Définition de la tâche à résoudre.

##### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 3 cm.
- $\eta$  l'incertitude de contrôle : 0 deg. (contrôle idéal).
- $\epsilon_t$  le pas de discrétisation en translation : 10 cm.
- $\varrho$  l'angle de rupture de contact : 5 deg.

## Illustrations :

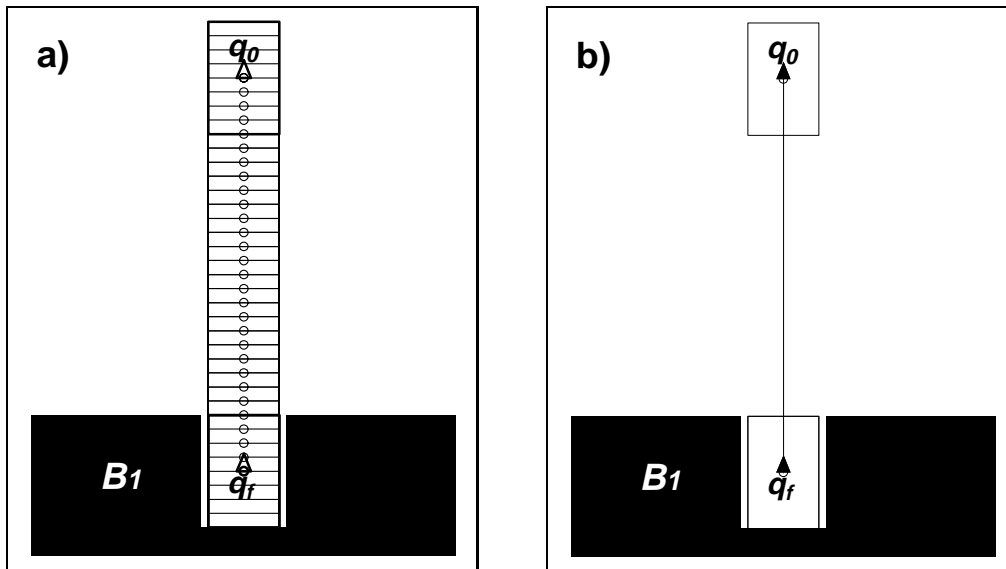


FIG. 6.4 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.

Plan robuste à appliquer sur la pièce  $\mathcal{A}$

1. COMMANDE-GARDÉE

$(\vec{v}_{\mathcal{A}} = (-1.0, 0.0, 0.0), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 280, \theta = 0, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$

## Commentaires :

L'incertitude initiale de position reste constante pendant le mouvement et dans ce cas sa valeur ne cause pas de collisions ce qui permet d'admettre la solution nominale générée.

## Evaluation de la solution :

- Nombre d'obstacles convexes définissant l'environnement : 3
- Nombre de commandes dans le plan : 1
- Temps de calcul du chemin : 46 ms ( $t_{moyen}$ )  $\Gamma$  40 ms ( $t_{min}$ )  $\Gamma$  50 ms ( $t_{max}$ )
- Estimation de la distance parcourue par la pièce : 280 cm.
- Estimation de l'incertitude finale de position : 3 cm.



## Problème 2b. Insertion élémentaire avec incertitude

### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.3).

### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 3 cm.
- $\eta$  l'incertitude de contrôle : 2.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 10 cm.
- $\varrho$  l'angle de rupture de contact : 5 deg.

### Illustrations :

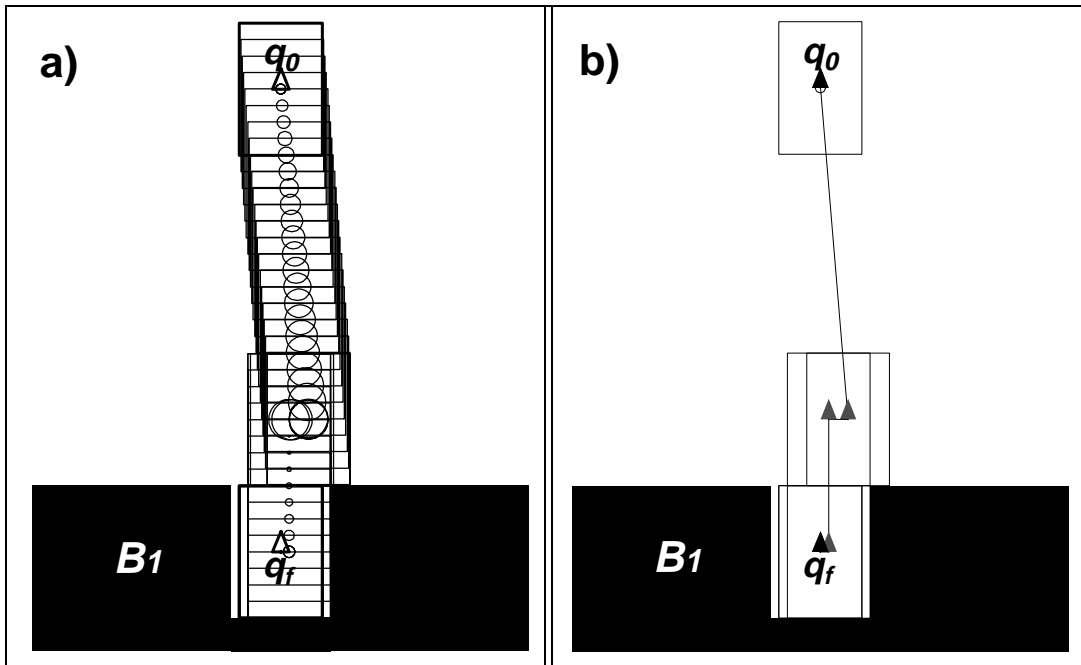


FIG. 6.5 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.

Plan robuste à appliquer sur la pièce  $\mathcal{A}$

1. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (-0.99, -0.08, 0.0), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 200.7, \theta = 0, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
2. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0), d = 11.76, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
3. COMMANDE-COMPLIANTE-JUSQU'AU-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (-1.0, 0.0, 0.0), d = 80.0, CV_{\mathcal{A}}(E_4^{\mathcal{A}}), CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}'_{seuil\mathcal{A}} = (0.0, 1.0), \vec{f}''_{seuil\mathcal{A}} = (1.0, 0.0))$
4. COMMANDE-COMPLIANTE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0), d = 5.0, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$

**Commentaires :**

L'effet cumulatif de l'incertitude de position ne permet pas de garantir une insertion directe. La manœuvre d'insertion proprement dite ne peut s'effectuer qu'une fois détecté un contact robuste avec la surface supérieure entourant la concavité. Ensuite les contacts sont utilisés pour guider la pièce vers l'intérieur puis jusqu'au but.

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 3
- Nombre de commandes dans le plan : 4
- Temps de calcul du chemin : 42 ms ( $t_{moyen}$ )  $\Gamma$  40 ms ( $t_{min}$ )  $\Gamma$  50 ms ( $t_{max}$ )
- Estimation de la distance parcourue par la pièce : 297.4 cm.
- Estimation de l'incertitude finale de position : 0.21 cm.

## Problème 2c. Insertion élémentaire avec incertitude

### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.6).

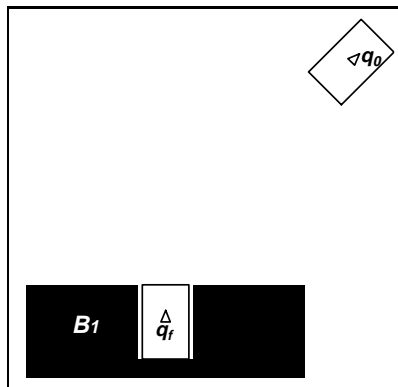


FIG. 6.6 – Définition de la tâche à résoudre.

### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 3 cm.
- $\eta$  l'incertitude de contrôle : 2.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 10 cm.
- $\varrho$  l'angle de rupture de contact : 5 deg.

### Illustrations :

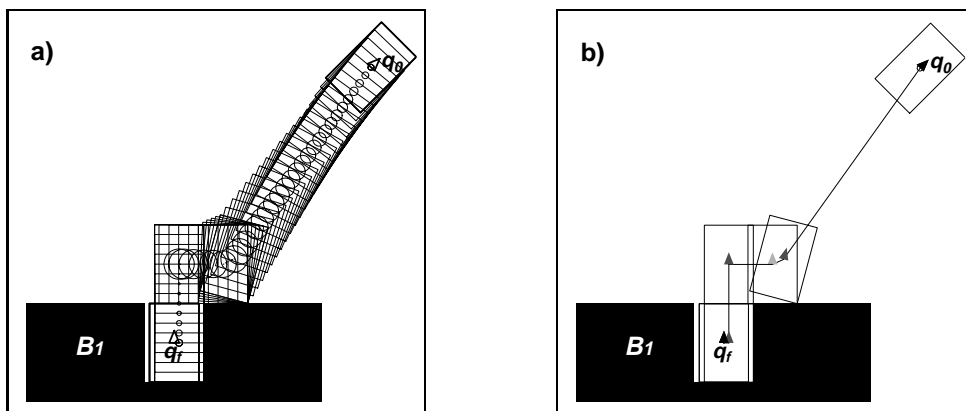


FIG. 6.7 – (a) Chemin et (b) plan robuste générés par le planificateur.

Plan robuste à appliquer sur la pièce  $\mathcal{A}$

1. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (-0.986, -0.164, 0.125), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 239.5, \theta = 30.07, CV_{\mathcal{A}}(s_4^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.96, 0.25))$
2. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
(sens = positif,  $\text{axe}_{\mathcal{A}} = (-40.0, -25.0), \theta = 14.92, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0)$ )
3. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0), d = 44.63, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
4. COMMANDE-COMPLIANTE-JUSQU'AU-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (-1.0, 0.0, 0.0), d = 80.0, CV_{\mathcal{A}}(E_4^{\mathcal{A}}), CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}'_{seuil\mathcal{A}} = (0.0, 1.0), \vec{f}''_{seuil\mathcal{A}} = (1.0, 0.0))$
5. COMMANDE-COMPLIANTE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0), d = 5.0, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$

**Commentaires :**

Comme dans l'exemple précédent l'insertion s'effectue après la détection d'un contact robuste avec la surface supérieure entourant la concavité suivie d'une manœuvre d'alignement autour du sommet de contact de la pièce.

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 3
- Nombre de commandes dans le plan : 5
- Temps de calcul du chemin : 330 ms ( $t_{moyen}$ )  $\Gamma$  320 ms ( $t_{min}$ )  $\Gamma$  350 ms ( $t_{max}$ )
- Estimation de la distance parcourue par la pièce : 369.1 cm.
- Estimation de l'incertitude finale de position : 0.21 cm.

## Problème 2d. Insertion élémentaire avec incertitude

### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.8).

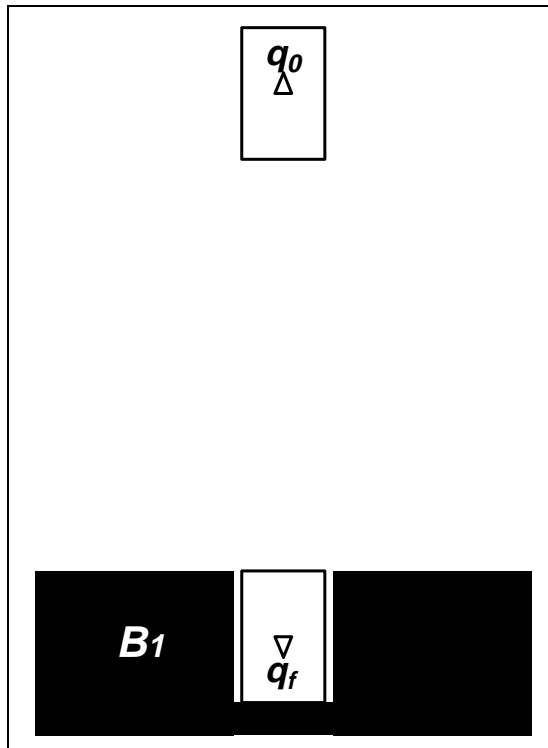


FIG. 6.8 – Définition de la tâche à résoudre.

### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 3 cm.
- $\eta$  l'incertitude de contrôle : 2.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 10 cm.
- $\varrho$  l'angle de rupture de contact : 5 deg.

Illustrations :

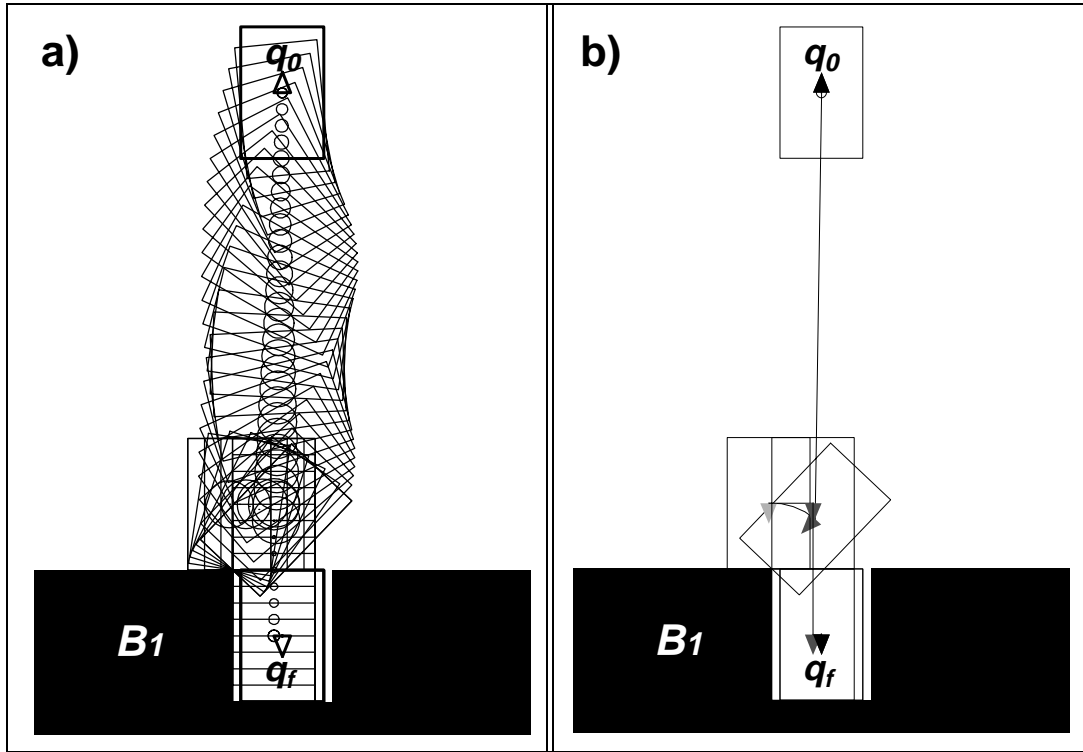


FIG. 6.9 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.

Plan robuste à appliquer sur la pièce  $\mathcal{A}$

1. COMMANDE-GARDÉE  
( $\vec{v}_{\mathcal{A}} = (-0.99, 0.01, 0.525)$ ),  $\text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}$ ,  $d = 259.73$ ,  $\theta = 136.36$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  
 $\vec{f}_{seuil\mathcal{A}} = (-1.0, 0.0)$ )
2. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
(sens = positif,  $\text{axe}_{\mathcal{A}} = (39.99, 1.94)$ ),  $\theta = 43.63$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{seuil\mathcal{A}} = (-1.0, 0.0)$ )
3. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
( $\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0)$ ),  $d = 26.94$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{seuil\mathcal{A}} = (-1.0, 0.0)$ )
4. COMMANDE-COMPLIANTE-JUSQU'AU-CONTACT  
( $\vec{v}_{\mathcal{A}} = (1.0, 0.0, 0.0)$ ),  $d = 80.0$ ,  $CV_{\mathcal{A}}(E_4^{\mathcal{A}})$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{seuil\mathcal{A}} = (0.0, 1.0)$ ,  
 $\vec{f}'_{seuil\mathcal{A}} = (-1.0, 0.0)$ )
5. COMMANDE-COMPLIANTE-EN-DISTANCE  
( $\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0)$ ),  $d = 5.0$ ,  $CV_{\mathcal{A}}(E_1^{\mathcal{A}})$ ,  $\vec{f}_{seuil\mathcal{A}} = (-1.0, 0.0)$ )

**Commentaires :**

La manœuvre d'alignement après la détection d'un contact robuste ne s'effectue pas autour d'un sommet de la pièce  $\mathcal{A}$  mais autour d'un sommet de l'obstacle  $\mathcal{B}_1$ .

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 3
- Nombre de commandes dans le plan : 5
- Temps de calcul du chemin : 342 ms ( $t_{moyen}$ )  $\Gamma$  330 ms ( $t_{min}$ )  $\Gamma$  350 ms ( $t_{max}$ )
- Estimation de la distance parcourue par la pièce : 371.64 cm.
- Estimation de l'incertitude finale de position : 0.21 cm.

### 6.1.3 Problème 3. Passage étroit (simulation)

#### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.10). La taille de la pièce  $\mathcal{A}$  est  $40 \times 20 \text{ cm}^2$  et la hauteur du passage entre les obstacles  $\mathcal{B}_1$  et  $\mathcal{B}_2$  est 30 cm.

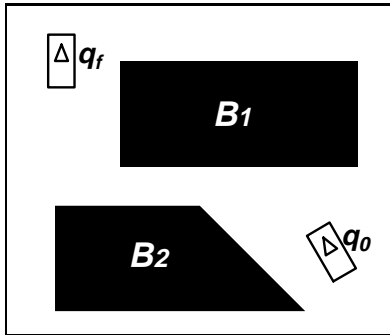


FIG. 6.10 – Définition de la tâche à résoudre.

#### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 3 cm.
- $\eta$  l'incertitude de contrôle : 1.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 10 cm.
- $\varrho$  l'angle de rupture de contact : 10 deg.

#### Illustrations :

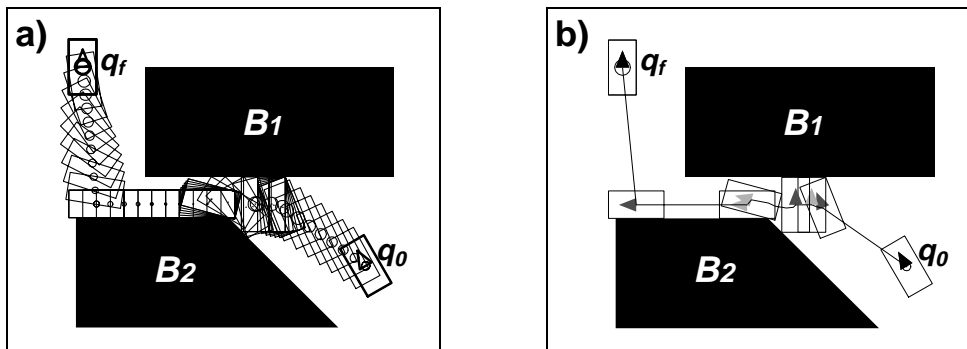


FIG. 6.11 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.



Plan robuste à appliquer sur la pièce  $\mathcal{A}$

1. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (0.90, 0.41, -0.11), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 73.98, \theta = -8.36, CV_{\mathcal{A}}(s_1^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-0.92, 0.36))$
2. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
 $(\text{sens} = \text{néгатif}, \text{axe}_{\mathcal{A}} = (20.0, -9.99), \theta = -21.63, CV_{\mathcal{A}}(E_1^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-1.0, 0.0))$
3. COMMANDE-COMPLIANTE-JUSQU'AU-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.0, 1.0, 0.0), d = 11.52, CV_{\mathcal{A}}(E_1^{\mathcal{A}}), CV_{\mathcal{A}}(s_3^{\mathcal{A}}), \vec{f}'_{seuil\mathcal{A}} = (-1.0, 0.0), \vec{f}''_{seuil\mathcal{A}} = (0.75, -0.66))$
4. COMMANDE-COMPLIANTE-MULTIPLE-JUSQU'À-LÂCHER-CONTACT  
 $(\text{sens} = \text{positif}, n^c = 6,$   
 $\text{config. contrôle} = \{(330.0, 200.0, 90.0), (328.29, 198.09, 104.99),$   
 $(323.30, 197.67, 119.99), (315.35, 198.78, 134.99), (310.35, 201.33, 149.99),$   
 $(298.11, 203.53, 164.99)\})$
5. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
 $(\text{sens} = \text{positif}, \text{axe}_{\mathcal{A}} = (-14.98, 10.0), \theta = 15, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.0, -1.0))$
6. COMMANDE-COMPLIANTE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (1.0, 0.0, 0.0), d = 80.01, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.0, -1.0))$
7. COMMANDE-LIBRE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (0.099, -0.995, -0.895), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 100.49, \theta = -90)$

**Commentaires :**

L'application de deux commandes compliantes permet de débloquent la pièce  $\mathcal{A}$  coincée entre deux obstacles disjoints à cause d'une mauvaise orientation; ainsi des passages étroits entre deux obstacles peuvent être franchis. La première commande (COMMANDE-COMPLIANTE-MULTIPLE-JUSQU'À-LÂCHER-CONTACT) consiste à faire tourner la pièce en gardant un double contact avec les obstacles pour parvenir quasiment à un alignement où un contact peut être lâché. Ensuite la deuxième commande (COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT) continue la rotation en gardant un contact jusqu'à alignement. A la fin de cette commande la pièce n'est en contact qu'avec un obstacle et son orientation lui permet de franchir sans contraintes le passage.

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 2
- Nombre de commandes dans le plan : 7
- Temps de calcul du chemin : 352 ms ( $t_{moyen}$ )  $\Gamma$  340 ms ( $t_{min}$ )  $\Gamma$  370 ms ( $t_{max}$ )
- Estimation de la distance parcourue par la pièce : 266 cm.
- Estimation de l'incertitude finale de position : 5.64 cm.

### 6.1.4 Problème 4. Navigation dans un environnement encombré (simulation)

Le but de cet exemple est d'introduire une première solution pour un robot mobile navigant à l'intérieur d'un environnement encombré. Des solutions adaptées pour un robot mobile réel seront illustrées dans la section 6.2.

#### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.12). La taille du robot  $\mathcal{A}$  est  $130 \times 80 \text{ cm}^2$ .

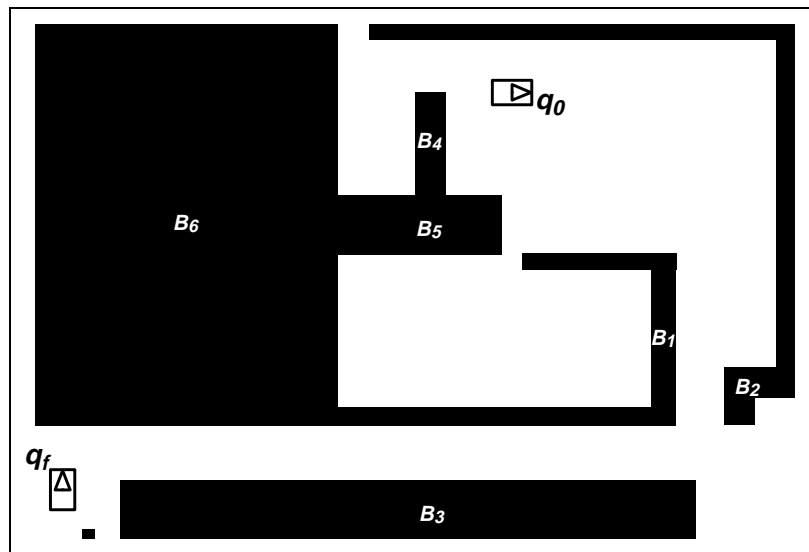


FIG. 6.12 – Définition de la tâche à résoudre.

#### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 2.5 cm.
- $\eta$  l'incertitude de contrôle : 2.0 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 25 cm.
- $\varrho$  l'angle de rupture de contact : 85 deg.

Illustrations :

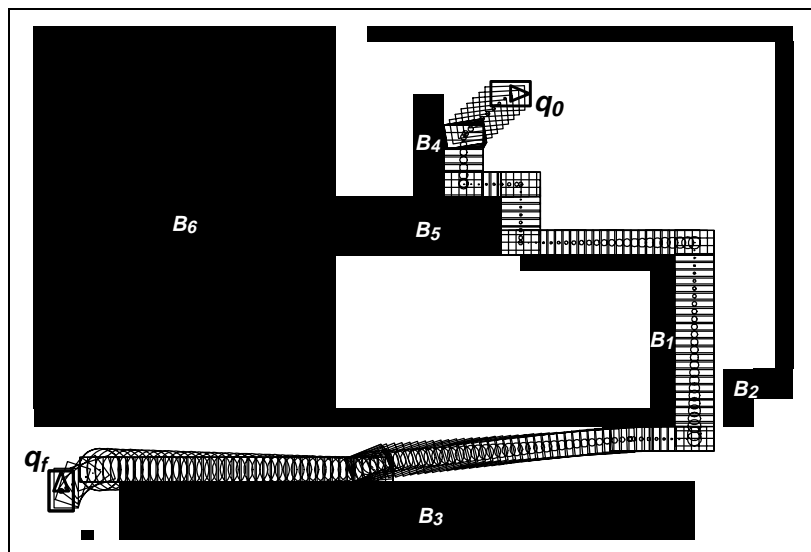


FIG. 6.13 – Chemin robuste généré par le planificateur.

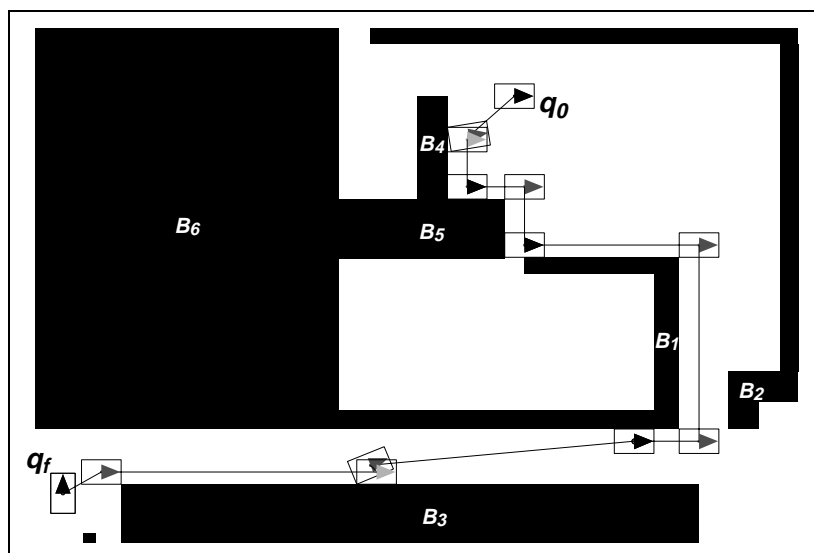


FIG. 6.14 – Plan robuste généré par le planificateur.

Plan robuste pour le robot  $\mathcal{A}$

1. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (-0.75, -0.65, 0.45), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 198.9, \theta = 9.13, CV_{\mathcal{A}}(s_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.98, -0.15))$
2. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
 $(\text{sens} = \text{négatif}, \text{axe}_{\mathcal{A}} = (-65.0, 40.0), \theta = -9.137, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
3. COMMANDE-COMPLIANTE-JUSQU'AU-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.0, -1.0, 0.0), d = 154.8, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), CV_{\mathcal{A}}(E_4^{\mathcal{A}}), \vec{f}'_{seuil\mathcal{A}} = (1.0, 0.0), \vec{f}''_{seuil\mathcal{A}} = (0.0, 1.0))$
4. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (1.0, 0.0, 0.0), d = 188.09, CV_{\mathcal{A}}(E_4^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.0, 1.0))$
5. COMMANDE-COMPLIANTE-JUSQU'AU-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.0, -1.0, 0.0), d = 190.9, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), CV_{\mathcal{A}}(E_4^{\mathcal{A}}), \vec{f}'_{seuil\mathcal{A}} = (1.0, 0.0), \vec{f}''_{seuil\mathcal{A}} = (0.0, 1.0))$
6. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (1.0, 0.0, 0.0), d = 571.8, CV_{\mathcal{A}}(E_4^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.0, 1.0))$
7. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.0, -1.0, 0.0), d = 641.9, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
8. COMMANDE-COMPLIANTE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (-1.0, 0.0, 0.0), d = 212.7, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.0, -1.0))$
9. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (-0.99, -0.09, 0.25), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 867.8, \theta = 22.11, CV_{\mathcal{A}}(s_4^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.37, 0.92))$
10. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
 $(\text{sens} = \text{négatif}, \text{axe}_{\mathcal{A}} = (-64.99, -39.99), \theta = -22.11, CV_{\mathcal{A}}(E_4^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.0, 1.0))$
11. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (-1.0, 0.0, 0.0), d = 901.8, CV_{\mathcal{A}}(E_4^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.0, 1.0))$
12. COMMANDE-LIBRE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (-0.87, -0.48, 6.24), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 144.1, \theta = 90)$

**Commentaires :**

A la base la solution proposée consiste à contourner les obstacles entraînant une situation de blocage. D'autre part la rotation du robot  $\mathcal{A}$  n'est vraiment effective que pendant l'exécution de la dernière commande à cause du mouvement d'alignement appliqué en contact c'est-à-dire que l'effet de l'incertitude d'orientation est annulé tout au long du chemin.

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 12

- Nombre de commandes dans le plan : 12
- Temps de calcul du chemin : 2739 ms ( $t_{moyen}$ )  $\Gamma$  2566 ms ( $t_{min}$ )  $\Gamma$  2999 ms ( $t_{max}$ )
- Estimation de la distance parcourue par le robot : 4072.79 cm.
- Estimation de l'incertitude finale de position : 5.03 cm.

### 6.1.5 Problème 5. Priorité sur l'exploration visible (simulation)

Cet exemple montre l'effet d'inter-changer l'ordre des stratégies d'exploration adjacente et visible (resp. les pas 3 et 4) dans l'algorithme de planification présenté en §5.4.4. Après avoir détecté un minimum local l'exploration est effectuée d'abord vers des obstacles non visités (exploration visible) et ensuite autour des obstacles déjà atteints (exploration adjacente). Ce changement permet aussi d'apprécier la mise en œuvre de l'exploration visible dans l'algorithme de planification.

De plus cet exemple permet d'illustrer la résolution d'un problème pour une pièce dont la géométrie est autre qu'un rectangle en l'occurrence un triangle.

#### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  et ensuite revenir sur  $q_0$  (figure 6.15). La résolution de ce problème passe par la résolution de deux sous-tâches de planification : la première depuis  $q_0$  jusqu'à  $q_f$  et la deuxième depuis  $q_f$  jusqu'à  $q_0$ . A cause de l'effet cumulatif de l'incertitude de position l'inversion de la solution d'une sous-tâche n'est pas valable comme solution pour l'autre.

Les dimensions de la pièce triangulaire sont 24 cm. de base (parallèle à l'axe  $Y$  du repère local  $\mathcal{F}_A$ ) et 20 cm. de hauteur (parallèle à l'axe  $X$  du repère local  $\mathcal{F}_A$ ).

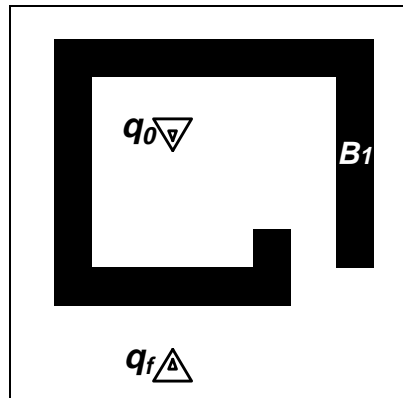


FIG. 6.15 – Définition de la tâche à résoudre.

#### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 3 cm.
- $\eta$  l'incertitude de contrôle : 2.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 6 cm.

–  $\varrho$  l'angle de rupture de contact : 10 deg.

Illustrations :

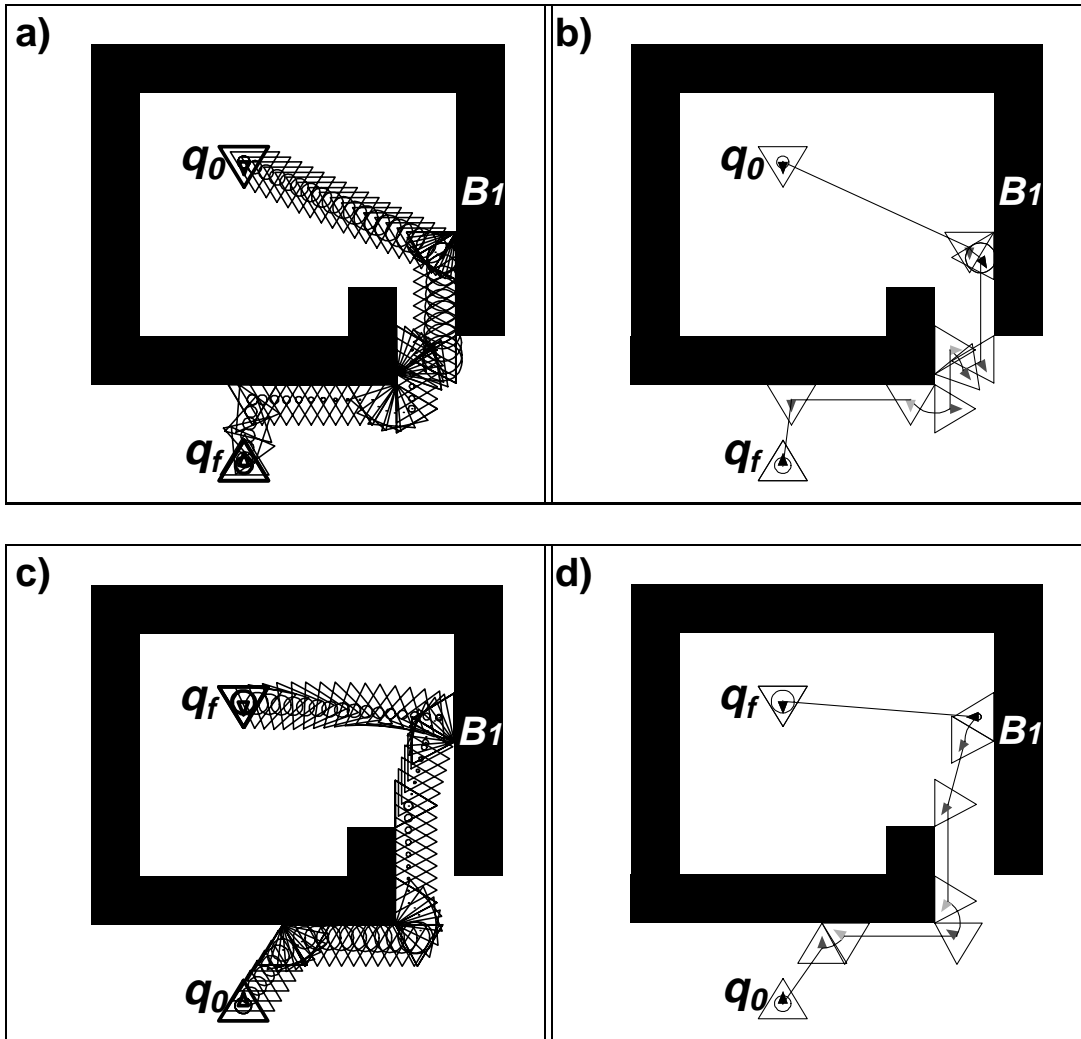


FIG. 6.16 – a) Chemin robuste de la résolution de la sous-tâche initiale.  
 b) Plan robuste de la résolution de la sous-tâche initiale.  
 c) Chemin robuste de la résolution de la sous-tâche finale.  
 d) Plan robuste de la résolution de la sous-tâche finale.

Plan robuste pour la sous-tâche initiale

1. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (0.417, 0.908, 0.0), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 101.24, \theta = 0, CV_{\mathcal{A}}(s_2^{\mathcal{A}}),$   
 $\vec{f}_{seuil\mathcal{A}} = (0.0, -1.0))$
2. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
(sens = positif,  $\text{axe}_{\mathcal{A}} = (-7.5, 12.0), \theta = 30.96, CV_{\mathcal{A}}(E_1^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-0.51, -0.85))$ )
3. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.85, -0.51, 0.0), d = 51.23, CV_{\mathcal{A}}(E_1^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-0.51, -0.85))$
4. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (-0.05, -0.99, 0.88), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 9.65, \theta = 8.57, CV_{\mathcal{A}}(s_3^{\mathcal{A}}),$   
 $\vec{f}_{seuil\mathcal{A}} = (0.63, 0.77))$
5. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
(sens = positif,  $\text{axe}_{\mathcal{A}} = (-7.5, 12.0), \theta = 50.46, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$ )
6. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (0.0, -1.0, 0.0), d = 29.04, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
7. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
(sens = négatif,  $\text{axe}_{\mathcal{A}} = (-7.5, 12.0), \theta = -90, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$ )
8. COMMANDE-COMPLIANTE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (0.0, -1.0, 0.0), d = 58.75, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
9. COMMANDE-LIBRE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (0.99, -0.12, -5.49), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 32.77, \theta = -180)$



Plan robuste pour la sous-tâche finale

1. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (0.81, -0.57, 0.0), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 33.67, \theta = 0, CV_{\mathcal{A}}(s_1^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-1.0, 0.0))$
2. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
 $(\text{sens} = \text{positif}, \text{axe}_{\mathcal{A}} = (12.5, 0.0), \theta = 59.03, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-0.51, 0.85))$
3. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (-0.85, -0.51, 0.0), d = 55.56, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-0.51, 0.85))$
4. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
 $(\text{sens} = \text{positif}, \text{axe}_{\mathcal{A}} = (12.5, 0.0), \theta = 90, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-0.51, 0.85))$
5. COMMANDE-COMPLIANTE-JUSQU'À-LÂCHER-CONTACT  
 $(\vec{v}_{\mathcal{A}} = (-0.85, -0.51, 0.0), d = 48.0, CV_{\mathcal{A}}(E_3^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (-0.51, 0.85))$
6. COMMANDE-GARDÉE  
 $(\vec{v}_{\mathcal{A}} = (-0.96, -0.27, 0.0), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 32.53, \theta = 0, CV_{\mathcal{A}}(s_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (0.51, -0.85))$
7. COMMANDE-GARDÉE-JUSQU'À-ALIGNEMENT  
 $(\text{sens} = \text{négatif}, \text{axe}_{\mathcal{A}} = (-7.5, 12.0), \theta = -59.03, CV_{\mathcal{A}}(E_2^{\mathcal{A}}), \vec{f}_{seuil\mathcal{A}} = (1.0, 0.0))$
8. COMMANDE-LIBRE-EN-DISTANCE  
 $(\vec{v}_{\mathcal{A}} = (0.99, -0.07, 0.92), \text{axe}_{\mathcal{A}} = \mathcal{R}_{\mathcal{A}}, d = 96.79, \theta = 90)$

**Commentaires :**

La configuration nominale finale  $q_f$  de la première sous-tâche devient la configuration initiale  $q_0$  de la deuxième sous-tâche. De même pour son estimation nominale sur l'incertitude de position.

Bien qu'une solution puisse être générée en contournant en contact l'obstacle  $\mathcal{B}_1$  la stratégie d'exploration visible permet dans les deux sous-tâches de générer un chemin plus court vers la configuration but.

**Evaluation de la solution (sous-tâche initiale) :**

- Nombre d'obstacles convexes définissant l'environnement : 5
- Nombre de commandes dans le plan : 9
- Temps de calcul du chemin : 1324 ms ( $t_{moyen}$ ) 1240 ms ( $t_{min}$ ) 1630 ms ( $t_{max}$ )
- Estimation de la distance parcourue par la pièce : 282.68 cm.
- Estimation de l'incertitude finale de position : 3.99 cm.

**Évaluation de la solution (sous-tâche finale) :**

- Nombre d'obstacles convexes définissant l'environnement : 5
- Nombre de commandes dans le plan : 8
- Temps de calcul du chemin : 1570 ms ( $t_{moyen}$ )  $\Gamma$  1560 ms ( $t_{min}$ )  $\Gamma$  1580 ms ( $t_{max}$ )
- Estimation de la distance parcourue par la pièce : 266.55 cm.
- Estimation de l'incertitude finale de position : 5.64 cm.

### 6.1.6 Problème 6. Limitation de l'approche

Cet exemple illustre une limitation de l'approche de planification proposée. Celle-ci est due au fait qu'en cas de détection d'un minimum local (blocage en contact) la stratégie pour sortir d'une telle situation consiste à explorer le sous-espace de contact  $\mathcal{C}_{contact}$  "atteignable" par le robot  $\Gamma$  malgré que l'espace libre  $\mathcal{C}_{libre}$  pourrait aussi permettre de repositionner le robot pour continuer la recherche de la configuration but.

#### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.17). La taille de la pièce  $\mathcal{A}$  est  $60 \times 40 \text{ cm}^2$ .

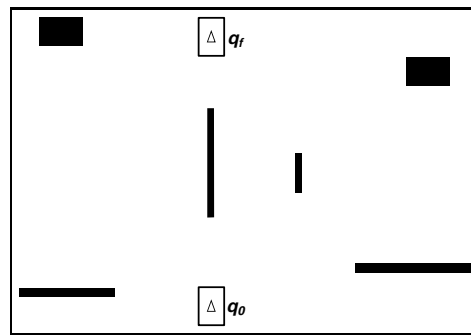


FIG. 6.17 – Définition de la tâche à résoudre.

#### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 5 cm.
- $\eta$  l'incertitude de contrôle : 3 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 10 cm.
- $\varrho$  l'angle de rupture de contact : 30 deg.

#### Commentaires :

A cause d'une taille trop petite de l'arête de contact rencontrée sur le chemin droit entre  $q_0$  et  $q_f$  la configuration nominale de contact  $q_{min}$  n'est pas "atteignable" sous les contraintes d'incertitudes imposées (figure 6.18). La configuration initiale  $q_0$  devient donc un "minimum local" pour la fonction de potentiel. Malheureusement l'analyse d'"atteignabilité" des arêtes des obstacles dans l'environnement depuis la configuration  $q_0$  ne permet pas non plus de définir une configuration de contact "robuste" afin de continuer la recherche de la configuration but  $q_f$ . Une solution ne peut donc pas être générée par l'approche de planification. Cependant l'exploration de l'espace

libre  $\mathcal{C}_{libre}$  permettrait de repositionner le robot de manière à changer les conditions d’“atteignabilité” depuis une configuration correspondant à un minimum local. Après avoir repositionné le robot dans l’espace libre un chemin solution pourrait être trouvé (figure 6.19).

### Illustrations :

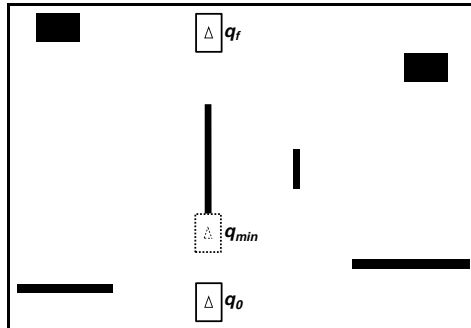


FIG. 6.18 – Configuration correspondant à un minimum local  $q_{min}$  “non-atteignable”, sous les contraintes d’incertitudes imposées, depuis la configuration initiale  $q_0$ .

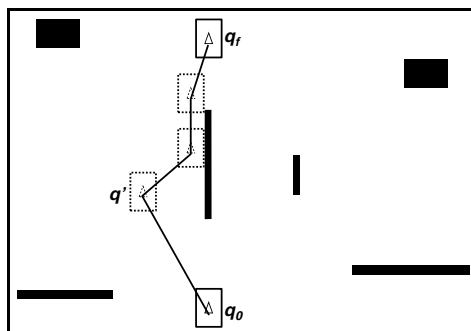


FIG. 6.19 – Un chemin solution de la tâche en explorant l’espace libre.

### 6.1.7 Conclusion

L’approche de planification proposée permet de générer des solutions pour un sous-ensemble de tâches où d’une part les stratégies basées sur l’évitement des obstacles ne sont guère adéquates et d’autre part l’effet de l’incertitude géométrique est critique pour garantir son accomplissement. Les solutions proposées cherchent à gagner des contacts robustes et ensuite à les maintenir le plus possible en évoluant vers la configuration but. Malheureusement des validations en expérimentation n’ont pas pu être réalisées ce qui nous aurait permis d’évaluer la faisabilité des plans générés.

Il est clair que le point d’analyse le plus important qui se répercute directement sur l’approche de planification est lié aux capacités du robot pour détecter et maintenir les contacts. Ces capacités dépendent fortement du support mis à disposition par le système de contrôle du robot.

## 6.2 Planification pour un robot mobile

L'approche originellement proposée peut être appliquée pour planifier des tâches affectées à un robot mobile. L'application pour un tel robot implique des adaptations de l'approche dues au fait que l'on ne peut pas considérer de façon similaire une pièce manipulée et un robot mobile même si la modélisation de l'objet mobile (pièce ou robot) ainsi que de son environnement est représentée dans un espace plan (bidimensionnel).

Tout d'abord le modèle du robot mobile pour lequel les tâches seront planifiées est décrit. Ensuite les systèmes sensoriels embarqués sur le robot et dont l'on dispose pour la détection des "contacts" sont détaillés. Puis les capacités de mobilité du robot ainsi que les primitives de mouvement (commandes de contrôle) que celui-ci peut exécuter sont présentées. Après les contraintes supplémentaires sur le déplacement du robot dues aux primitives de mouvement sont détaillées et leur intégration dans l'approche de planification est expliquée. Finalement des résultats de l'approche obtenus en simulation et en expérimentation sont illustrés.

### 6.2.1 Modèle du robot

Le modèle décrit ici correspond à une modélisation du robot mobile Hilare II du LAAS-CNRS [BFLV92Fer91] (figure 6.20). Dans notre cas nous ne décrivons que les composantes du système robotique les plus pertinentes à prendre en compte pour le planificateur.



FIG. 6.20 – *Le robot mobile Hilare II.*

#### 6.2.1.1 Géométrie

Le robot Hilare II (noté  $\mathcal{A}$ ) est modélisé par un rectangle dont la longueur est de 130 centimètres et la largeur de 80 centimètres. Le point de référence  $\mathcal{O}_{\mathcal{A}}$  du robot est le point milieu du rectangle. Un repère  $\mathcal{F}_{\mathcal{A}}$  en trois axes  $(X, Y, \Theta)$  est fixé au robot en  $\mathcal{O}_{\mathcal{A}}$ . Le repère  $\mathcal{F}_{\mathcal{A}}$  est orienté de façon à ce que la direction de l'axe  $X$  suive la longueur du robot et celle de l'axe  $Y$  la largeur (figure 6.21). L'axe de rotation  $\Theta$  du robot est

perpendiculaire au rectangle et passe par le point  $\mathcal{O}_A$ .

On peut nommer les faces du robot rectangulaire par rapport à  $\mathcal{F}_A$  ainsi :

- Face avant  $FF_A$  : face perpendiculaire à l’axe  $X$  dont l’abscisse est positive.
- Face arrière  $FA_A$  : face perpendiculaire à l’axe  $X$  dont l’abscisse est négative.
- Face droite  $FD_A$  : face perpendiculaire à l’axe  $Y$  dont l’ordonnée est négative.
- Face gauche  $FG_A$  : face perpendiculaire à l’axe  $Y$  dont l’ordonnée est positive.

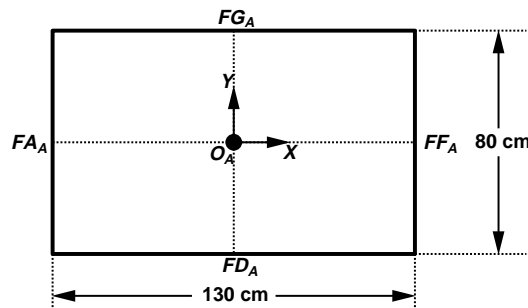


FIG. 6.21 – Modélisation géométrique du robot mobile Hilare II.

### 6.2.1.2 Localisation

La localisation du robot  $\mathcal{A}$  dans un environnement plan  $\mathcal{W}$  correspond à la position/orientation du repère  $\mathcal{F}_A$  du robot par rapport au repère fixe  $\mathcal{F}_W$  de l’environnement. Plus précisément la position  $(x, y)$  du robot est celle du point  $\mathcal{O}_A$  dans le repère  $\mathcal{F}_W$ ; et l’orientation  $\theta$  correspond à l’angle dans le sens trigonométrique entre les demi-axes  $X$  positifs des repères  $\mathcal{F}_W$  et  $\mathcal{F}_A$ .

## 6.2.2 Systèmes sensoriels du robot

Le robot Hilare II est équipé de plusieurs systèmes sensoriels. Nous décrivons ceux qui sont utilisés pour l’estimation de la position/orientation du robot la détection de “contacts” et l’asservissement du mouvement en “contact”.

### 6.2.2.1 L’odométrie

Le robot  $\mathcal{A}$  est doté de deux roues motrices et deux roues odométriques sur lesquelles des codeurs incrémentaux ont été installés. L’intégration des données provenant de ces codeurs permet d’avoir une estimation de la position/orientation de  $\mathcal{A}$  dans l’environnement  $\mathcal{W}$ . Une telle estimation est entachée d’une erreur cumulative.

### 6.2.2.2 Les télémètres ultrasoniques

Le robot  $\mathcal{A}$  dispose d'une ceinture de 32 télémètres ultrasoniques destinés à l'obtention d'une représentation proximétrique de l'environnement immédiat du robot. Ce type de télémètre permet au robot de détecter et d'éviter des obstacles ainsi que de suivre des parois à une distance de sécurité  $d_s$ . La disposition des télémètres ultrasoniques est illustrée dans la figure 6.22. D'après des tests expérimentaux menés au LAAS-CNRS le seuil inférieur  $d_{ultrasons}$  d'approche aux obstacles pour la détection de contacts et suivi des parois est de 15 centimètres. Cette valeur garantit que le récepteur d'un télémètre puisse recevoir l'écho ultrasonore de son émetteur.

### 6.2.2.3 Le télémètre laser

Le robot  $\mathcal{A}$  possède un télémètre laser monté sur une plate-forme orientable en azimut dont le faisceau peut être lui aussi orientable en site au moyen de la rotation d'un miroir. L'application d'un tel télémètre est multiple : dans notre cas le télémètre est utilisé pour détecter un sommet convexe au cours d'un suivi de paroi. De cette manière le robot peut se recalculer par rapport au sommet détecté et son incertitude de position est de ce fait réduite. Ce télémètre se trouve placé sur le demi-axe  $X$  positif du repère  $\mathcal{F}_A$  à une distance de 22 centimètres du point  $\mathcal{O}_A$  (figure 6.22).

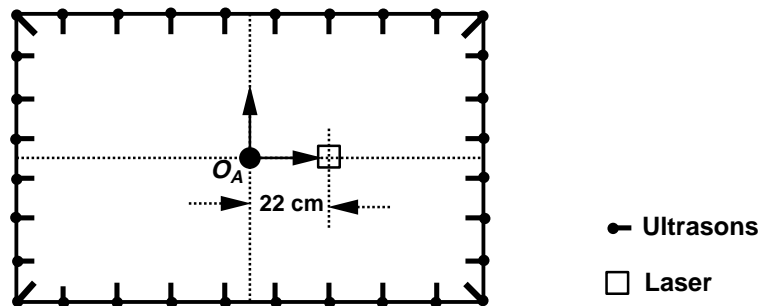


FIG. 6.22 – Placement des télémètres ultrasoniques et laser sur le robot mobile Hilare II.

## 6.2.3 Mobilité du robot

Sous le mode d'application du robot Hilare II que nous utilisons la mobilité du robot est caractérisée par deux degrés de liberté : un degré en translation le long de l'axe  $X$  du repère  $\mathcal{F}_A$  et un degré en rotation autour de l'axe de rotation  $\Theta$  passant par le point  $\mathcal{O}_A$ . Le degré de liberté en translation ne permet que des déplacements vers l'avant ou vers l'arrière du robot en gardant l'orientation courante. La combinaison translation-rotation permet au robot d'atteindre n'importe quelle position dans un plan.

### 6.2.4 Primitives de mouvement du robot

Le système de contrôle disponible sur le robot Hilare III développé et mis au point au LAAS-CNRS permet de commander celui-ci à l'aide des primitives définies ci-dessous :

1. **TURN**( $\delta\theta$ ).  
Rotation autour de l'axe  $\Theta$  du repère  $\mathcal{F}_A$  d'un angle  $\delta\theta$ .
2. **MOVE**(sens  $\Gamma \delta\theta \Gamma l$ ).  
Rotation autour de l'axe  $\Theta$  du repère  $\mathcal{F}_A$  d'un angle  $\delta\theta$  suivi d'une translation selon l'axe  $X$  dans le sens spécifié (avant/arrière) sur une distance  $l$ .
3. **MOVE-UNTIL-CONTACT**(sens  $\Gamma d_{contact} \Gamma l \Gamma \epsilon_l$ ).  
Translation suivant l'axe  $X$  du repère  $\mathcal{F}_A$  dans le sens spécifié (avant/arrière) sur une distance  $l$  ( $\pm \epsilon_l$ ) interrompu lorsque la distance à l'obstacle le plus proche devient  $d_{contact}$  (figure 6.23).

La distance de contact  $d_{contact}$  sera déterminée par les télémètres ultrasoniques et correspond donc à la distance de l'enveloppe rectangulaire du robot à la surface de l'obstacle le plus proche.

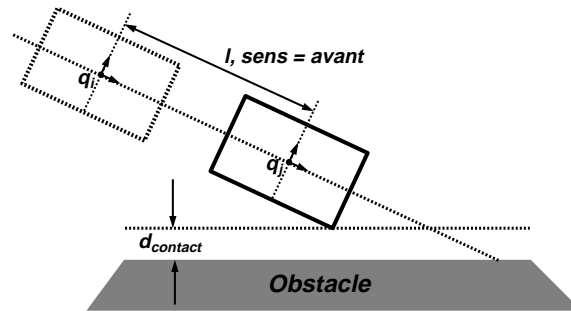


FIG. 6.23 – Illustration de la primitive MOVE-UNTIL-CONTACT depuis  $q_i$  jusqu'à  $q_j$ .

4. **FOLLOW**( $n_{murs} \Gamma d_{contact} \Gamma$ liste-murs).  
Cette primitive permet le suivi de  $n_{murs}$  murs consécutifs en gardant une distance  $d_{contact}$  (figure 6.24). Pour chaque mur il faut spécifier :
  - le côté (droit/gauche) sur lequel le robot doit faire l'asservissement avec les télémètres ultrasoniques;
  - le type de sommet (convexe/concave) à détecter à la fin du suivi;
  - l'angle  $\delta\theta$  entre le mur courant et le suivant;
  - la distance  $l_1$  sans activation de la détection des sommets;
  - la distance  $l_2$  avec activation de la détection des sommets;
  - la condition de terminaison du suivi (distance  $l_1 + l_2$  parcourue et/ou sommet détecté);



- le seuil de distance  $d_{concave}$  utilisé pour la détection des sommets concaves.

Comme nous l'avons dit précédemment les sommets convexes seront détectés à l'aide du télémètre laser et les sommets concaves à l'aide des télémètres ultrasoniques.

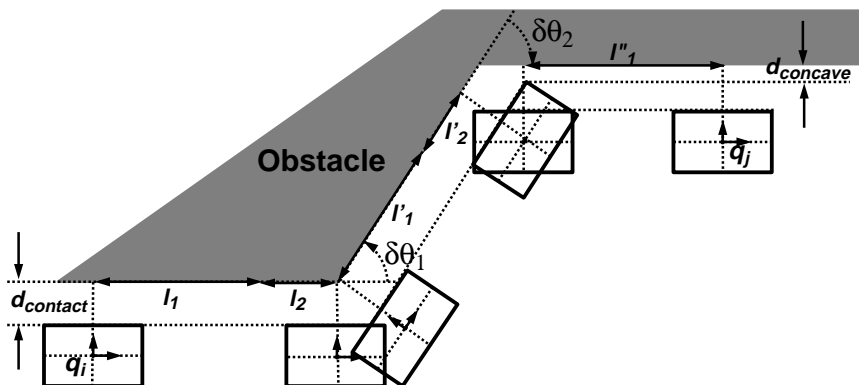


FIG. 6.24 – Illustration de la primitive FOLLOW depuis  $q_i$  jusqu'à  $q_j$ .

#### 5. LOCALIZE( $x, y, \theta, M$ ).

Cette commande renvoie la localisation  $(x, y, \theta)$  du repère  $\mathcal{F}_A$  par rapport au repère  $\mathcal{F}_W$  de l'environnement ainsi que sa matrice de covariance  $M$  associée.

#### 6. PARALLELIZE(côté $\Gamma d \Gamma \delta \theta \Gamma \epsilon_d \Gamma \epsilon_\theta \Gamma S$ ).

Réalise une mise en parallèle du robot le long d'une paroi à une distance  $d$  ( $\pm \epsilon_d$ ). Le côté (droit/gauche) sur lequel le robot doit faire l'asservissement avec les télémètres ultrasoniques ainsi qu'une estimation  $\delta \theta$  ( $\pm \epsilon_\theta$ ) de la déviation angulaire doivent être spécifiés. Un segment  $S$  parallèle à la paroi limite le débattement longitudinal du robot lors de l'exécution de la primitive.  $\epsilon_d$  et  $\epsilon_\theta$  sont les imprécisions respectivement sur  $d$  et  $\delta \theta$ .

Les distances sont mesurées en mètres et les angles en radians dans le sens trigonométrique.

Un plan valable pour le robot Hilare II ( $\mathcal{A}$ ) doit donc être une suite de commandes primitives.

### 6.2.5 Contraintes supplémentaires imposées sur le robot

Les primitives de mouvement disponibles pour le système de contrôle du robot  $\mathcal{A}$  imposent des contraintes supplémentaires sur son déplacement à savoir :

- Contrainte 1 : la détection des "contacts" entre le robot  $\mathcal{A}$  et les obstacles  $\{\mathcal{B}_i\}$  dans l'environnement  $\mathcal{W}$  doit se faire à une distance de contact  $d_{contact}$  au-dessus

du seuil inférieur  $d_{ultrasons}$  défini par les télémètres ultrasoniques (i.e.  $d_{ultrasons} < d_{contact}$ ).  $d_{contact}$  mesure la distance de l'enveloppe rectangulaire du robot à la surface des obstacles.

- Contrainte 2: toute rotation du robot  $\mathcal{A}$  doit être effectuée autour de l'axe de rotation  $\Theta$  passant par le point  $\mathcal{O}_{\mathcal{A}}$ .
- Contrainte 3: il n'est plus possible de se déplacer en translation et en rotation en même temps.
- Contrainte 4: l'asservissement pendant le suivi d'une paroi ne peut se faire que du côté droit  $FD_{\mathcal{A}}$  ou du côté gauche  $FG_{\mathcal{A}}$  du robot  $\mathcal{A}$ . Cependant lors d'un suivi de multiple parois le côté d'asservissement peut changer au moment du changement de paroi.
- Contrainte 5: Le placement du télémètre laser vers l'avant du robot  $\mathcal{A}$  impose lors d'un suivi de paroi un sens de déplacement vers l'avant.

### 6.2.5.1 Intégration des contraintes dans l'approche de planification

#### Contrainte 1: Distance de détection des contacts

Afin de satisfaire une détection de contacts à une distance au-dessus du seuil inférieur  $d_{ultrasons}$  des télémètres ultrasoniques un obstacle virtuel  $\mathcal{V}_i$  est défini autour de chaque obstacle réel  $\mathcal{B}_i$ . Ces obstacles virtuels respecteront une distance de sécurité  $d_s$  dont la borne inférieure est  $d_{ultrasons}$  (i.e.  $d_{ultrasons} < d_s \leq d_{contact}$ ). Les "contacts" seront donc définis entre le robot  $\mathcal{A}$  et les obstacles virtuels  $\{\mathcal{V}_i\}$ . Cependant la valeur de  $d_s$  crée un compromis entre les rotations possibles près des obstacles réels et les intersections entre les obstacles virtuels. Ceci s'explique du fait que plus  $d_s$  est grand plus le robot est loin des obstacles réels mais en même temps plus grande est la probabilité d'une intersection entre les obstacles virtuels. Une intersection entre deux obstacles virtuels signifie que le robot  $\mathcal{A}$  ne peut pas satisfaire en même temps la distance de sécurité  $d_s$  entre leurs obstacles réels associés.

Puisque la planification est faite dans l'espace des configurations  $\mathcal{C}$  les  $\mathcal{C}$ -obstacles  $\{\mathcal{CB}_i\}$  seront construits comme le grossissement des obstacles virtuels  $\{\mathcal{V}_i\}$  avec la géométrie du robot  $\mathcal{A}$ . La distance de sécurité  $d_s$  permettant une rotation quelconque du robot  $\mathcal{A}$  sans risque des collisions avec les obstacles réels  $\{\mathcal{B}_i\}$  correspond à la différence entre le plus petit cercle englobant le robot et le plus grand cercle inscrit dans le robot. Pour le robot  $\mathcal{A}$  rectangulaire considéré on obtient (figure 6.25):

$$d_s = \sqrt{65^2 + 40^2} - 40 \text{ (cm)} \quad (6.1)$$

Ce qui donne  $d_s = 36.32$  (cm) supérieur à  $d_{ultrasons}$  (15cm.). Avec cette valeur les passages entre deux obstacles réels dont la distance est inférieure à  $2d_s + 80$  centimètres (i.e. 152.64 cm) ne seront pas franchissables par le robot. Or dans le cas idéal où le

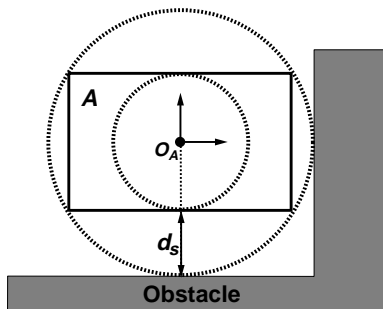


FIG. 6.25 – Seuil inférieur de la distance de sécurité  $d_s$  pour garantir les rotations du robot mobile  $\mathcal{A}$  près des obstacles.

robot est déjà orienté convenablement. La distance limite d'un passage entre deux obstacles  $\Gamma$  pour être franchissable  $\Gamma$  est de  $2d_{ultrasons} + 80$  centimètres (i.e. 110 cm). On peut donc conclure que la distance de sécurité  $d_s$  est bornée inférieurement par  $d_{ultrasons}$  et qu'au delà de 36.32 centimètres les rotations n'entraîneront pas de collisions.

L'algorithme proposé pour la construction d'un obstacle virtuel  $\mathcal{V}_i$  autour de l'obstacle réel  $\mathcal{B}_i$  est explicité en annexe §A.3 (figure 6.26).

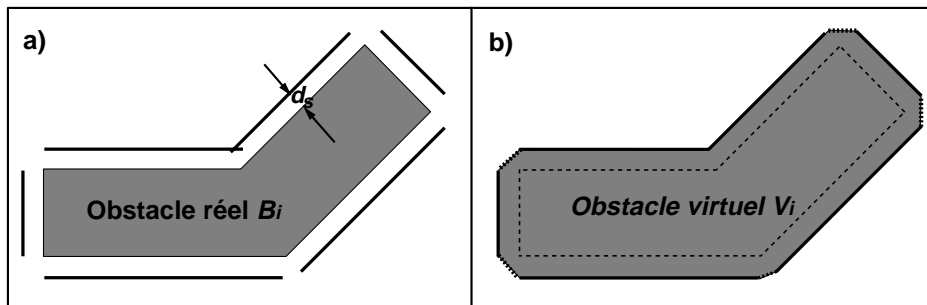


FIG. 6.26 – a) Construction des arêtes de base pour l'obstacle virtuel  $\mathcal{V}_i$ . b) Fermeture du contour définissant l'obstacle virtuel  $\mathcal{V}_i$ .

Cette construction ne fait que déplacer les arêtes réelles ce qui garantira ensuite une détection correcte des contacts à distance par rapport à celles-ci. Bien que la distance  $d_s$  ne soit pas respectée aux alentours des sommets convexes de  $\mathcal{B}_i$  on peut reconnaître cette situation par l'identification des arêtes de liaison dans  $\mathcal{V}_i$ .

### Contrainte 2 : Rotations autour du point $\mathcal{O}_A$

Dans l'espace libre  $\mathcal{C}_{libre}$  il n'y a pas de changement car la rotation se fait autour de l'axe  $\Theta$  passant par  $\mathcal{O}_A$ . Par contre lorsque le robot  $\mathcal{A}$  doit s'aligner avec un obstacle  $\Gamma$  ceci n'est plus possible de se faire autour du sommet de contact contenu dans l'enveloppe rectangulaire de  $\mathcal{A}$  (contrainte 4).

### Contrainte 3 : Translation et rotation simultanées

Dans l'approche originale la translation et la rotation simultanées sont possibles en deux situations : la première lorsque le robot évolue dans l'espace libre  $\mathcal{C}_{libre}$  et la deuxième lorsque le robot effectue un déplacement d'alignement. En ce qui concerne la première situation un chemin *libre* ou *quasi-libre*  $\tau$  (§3.1) entre deux configurations extrêmes  $q_i = (x_i, y_i, \theta_i)$  et  $q_j = (x_j, y_j, \theta_j)$  généré par l'application des translations-rotations simultanées peut être obtenu par une suite composée d'une rotation initiale sur  $(x_i, y_i)$  une translation pure entre  $(x_i, y_i)$  et  $(x_j, y_j)$  et une rotation finale sur  $(x_j, y_j)$  (figure 6.27). En ce qui concerne la deuxième situation elle est liée à la contrainte 4.

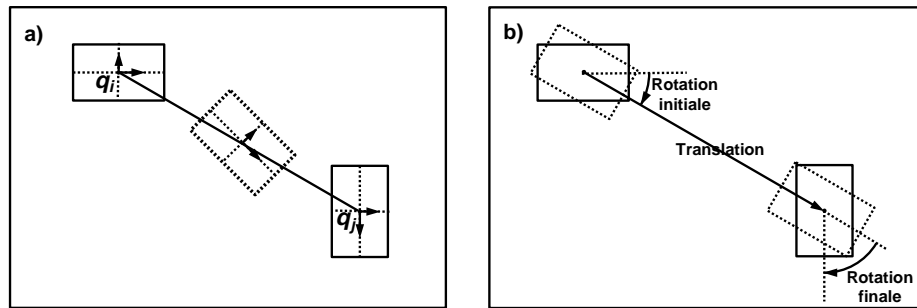


FIG. 6.27 – a) Translation et rotation simultanées depuis  $q_i$  jusqu'à  $q_j$ . b) Décomposition du déplacement en : rotation initiale, translation et rotation finale.

### Contrainte 4 : Côté d'asservissement pour le suivi des parois

Lors d'un suivi de paroi l'asservissement ne peut se faire que du côté droit  $FD_{\mathcal{A}}$  ou du côté gauche  $FG_{\mathcal{A}}$  du robot  $\mathcal{A}$ . Cette contrainte intervient sur le déplacement d'alignement appliqué pour contourner les  $\mathcal{C}$ -obstacles. Ce déplacement est appliqué en deux occasions :

1. Lorsque le robot  $\mathcal{A}$  gagne un contact après avoir évolué dans l'espace libre  $\mathcal{C}_{libre}$ . A cause de la contrainte de rotation (contrainte 2) l'alignement autour du point  $\mathcal{O}_{\mathcal{A}}$  fait lâcher le contact détecté. Pour cette raison l'alignement effectué ne laisse pas le robot tout à fait parallèle à l'arête virtuelle de contact mais penché vers celle-ci de façon à pouvoir regagner le contact dans la suite (figure 6.28a).
2. Lorsque le robot  $\mathcal{A}$  doit contourner des sommets des obstacles réels. Dans ce cas l'alignement avec l'arête suivante est assuré par l'application de la primitive FOLLOW (figure 6.28b).

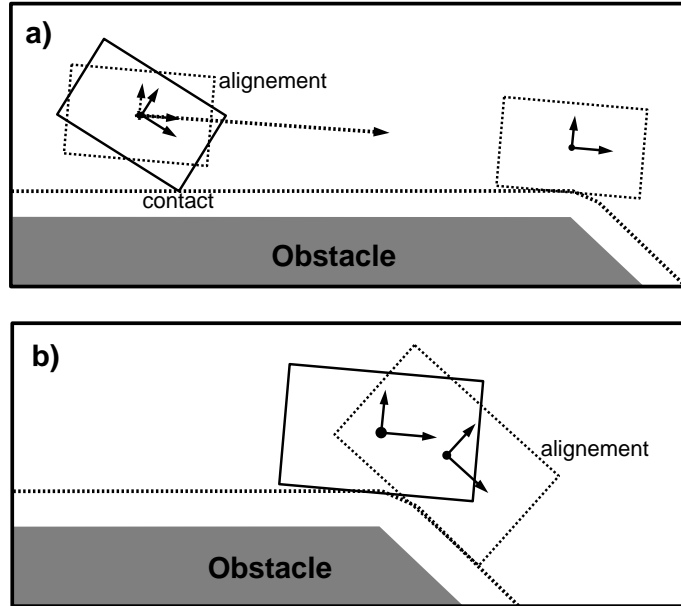


FIG. 6.28 – a) Alignement après la détection d'un contact. b) Alignement autour des sommets réels.

### Contrainte 5 : Suivi de parois vers l'avant

Pour l'approche de planification  $\Gamma$  la stratégie de suivi de parois correspond au contournement des obstacles déclenché en deux occasions :

1. Lorsque la fonction de potentiel fictif est contrainte à glisser en contact pour s'approcher de la configuration but.  
Dans ce cas  $\Gamma$  il est nécessaire de garantir que la configuration initiale de contact soit alignée et orientée de manière convenable pour suivre la direction du potentiel.
2. Lorsque l'exploration adjacente est activée pour approcher un attracteur adjacent depuis une configuration de contact alignée.  
Dans ce cas  $\Gamma$  il est nécessaire de considérer l'orientation de la configuration de départ pour placer les attracteurs adjacents le long du sous-chemin qui peut être exploré dans la direction avant  $FF_{\mathcal{A}}$  du robot.

### 6.2.6 Résultats de la planification des tâches

Dans la suite  $\Gamma$  on va illustrer l'application de l'approche de planification pour résoudre des tâches affectées au robot mobile  $\mathcal{A}$  considéré. Dans la spécification de telles tâches  $\Gamma$  la distance de sécurité  $d_s$  doit être ajoutée.

### Remarques

- Afin de faciliter la compréhension des paramètres des primitives de mouvement les distances seront spécifiées en centimètres et les angles en degrés.
- Dans les figures illustrant les exemples un triangle est placé au point de référence  $\mathcal{O}_A$  du repère local  $\mathcal{F}_A$  et pointe vers l'avant  $FF_A$  du robot  $\mathcal{A}$ .
- A propos de l'incertitude de position celle-ci est montrée comme un disque autour du point de référence  $\mathcal{O}_A$  même en situation de contact ce qui permet de mieux apprécier son évolution. Nous rappelons que l'incertitude de position est représentée par un disque dans l'espace libre et comme un intervalle le long de la surface de contact dans l'espace de contact.
- En ce qui concerne l'incertitude d'orientation celle-ci ne sera pas évaluée explicitement pendant la planification. Cependant en raison de l'alignement de la pièce nécessaire pour asservir les mouvements le long d'une paroi (contrainte 4) l'incertitude d'orientation est dans ces cas annulée.
- Les lignes polygonales autour des obstacles réels  $\{\mathcal{B}_i\}$  correspondent aux obstacles virtuels  $\{\mathcal{V}_i\}$  définis à la distance de sécurité  $d_s$ .
- Une solution générée par le planificateur est illustrée avec trois figures :
  1. la première figure montre la trace du robot dans son environnement le long du chemin généré;
  2. la deuxième figure montre les configurations initiale et finale nominales du robot avant et après l'exécution de chaque commande dans un premier plan de commandes. Pour chaque commande un trait de liaison montre le chemin nominal qui suit le robot mobile  $\mathcal{A}$  entre ses deux configurations extrêmes. Ce premier plan solution est défini à partir de l'ensemble générique des commandes de contrôle présenté dans §3.4.1;
  3. la troisième figure est semblable à la figure précédente mais dans ce cas le plan de commandes initial est transformé en un plan composé de primitives de mouvement propres au robot mobile  $\mathcal{A}$  considéré. L'ensemble de ces primitives est présenté dans §6.2.4. Une différence entre les figures deux et trois peut être remarquée lors d'un changement de paroi. Dans la troisième figure la transition entre deux parois consécutives est assurée de manière implicite par la primitive FOLLOW. Cela explique la discontinuité entre les traits nominaux du suivi de deux parois consécutives.
- Les temps CPU utilisés pour déterminer la solution des exemples présentés ont été mesurés pour une implantation opérationnelle de l'algorithme séquentiel de l'approche lancé sur une machine SGI de type INDY. Pour chaque exemple des temps correspondant à cinq exécutions du planificateur générant une solution

semblable à celle illustrée ont été pris. À partir de ces mesures trois temps sont présentés : (1) le temps moyen des cinq mesures ( $t_{moyen}$ ) (2) le temps minimum ( $t_{min}$ ) (3) le temps maximum ( $t_{max}$ ). Ces temps sont donnés en millisecondes (ms).

### 6.2.6.1 Problème 1. Attraction sans blocage (simulation)

#### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  (figure 6.29).

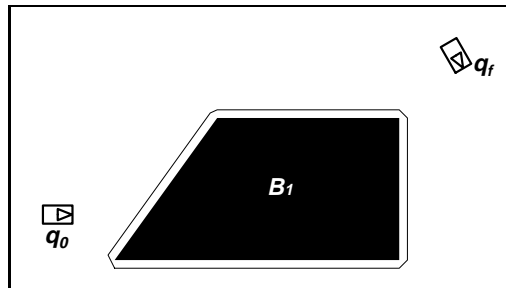


FIG. 6.29 – Définition de la tâche à résoudre.

#### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 10 cm.
- $\eta$  l'incertitude de contrôle : 2.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 25 cm.
- $\varrho$  l'angle de rupture de contact : 45 deg.
- $d_s$  la distance de sécurité : 36.3 cm.

Illustrations :

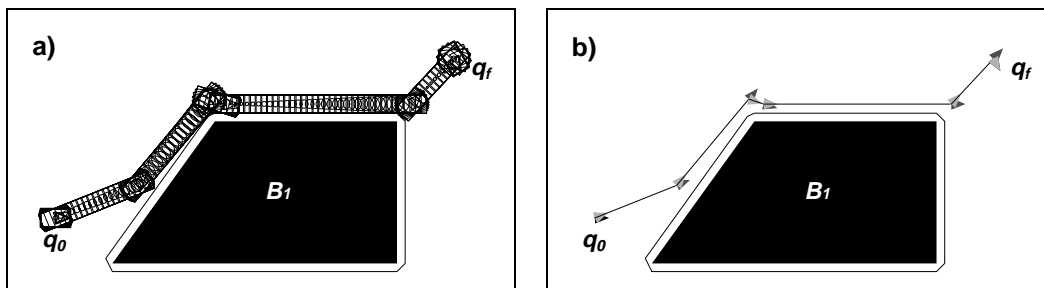


FIG. 6.30 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.

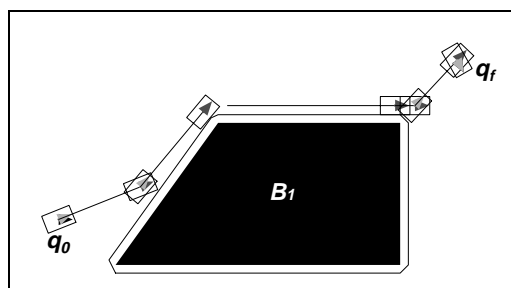


FIG. 6.31 – Transformation du plan généré en un plan exécutable.



Plan exécutable par le robot  $\mathcal{A}$

1. TURN( $\delta\theta = 21.8$ )
2. MOVE-UNTIL-CONTACT(sens = avant,  $d_{contact} = 43.4$ ,  $l = 387.1$ ,  $\epsilon_l = 26.9$ )
3. TURN( $\delta\theta = 28.2$ )
4. FOLLOW( $n_{murs} = 2$ ,  $d_{contact} = 36.3$ )  
Paramètres mur 1 :  
 côté = droit, sommet = convexe,  $\delta\theta = -50.0$ ,  $l_1 = 382.8$ ,  $l_2 = 91.2$ ,  
 détection vertex,  $d_{concave} = -$   
  
Paramètres mur 2 :  
 côté = droit, sommet = convexe,  $\delta\theta = 0$ ,  $l_1 = 704.2$ ,  $l_2 = 64.3$ , détection  
 vertex,  $d_{concave} = -$
5. MOVE(sens = avant,  $\delta\theta = 0$ ,  $l = 87.0$ )
6. TURN( $\delta\theta = 47.0$ )
7. MOVE(sens = avant,  $\delta\theta = 0$ ,  $l = 271.4$ )
8. TURN( $\delta\theta = -107.0$ )

**Commentaires :**

La solution trouvée se limite à suivre le chemin le plus court entre la configuration initiale  $q_0$  et la configuration finale  $q_f$  en contournant l'obstacle  $\mathcal{B}_1$ . On peut remarquer dans la figure 6.31 que le robot  $\mathcal{A}\Gamma$  avant l'exécution de la primitive FOLLOW $\Gamma$  est aligné à une distance supérieure à celle spécifiée par le paramètre  $d_{contact}$ . La correction correspondante est effectuée par le système de contrôle le long de la première paroi.

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 1
- Nombre de commandes dans le plan : 8
- Temps de calcul du chemin : 106 ms ( $t_{moyen}$ )  $\Gamma$  100 ms ( $t_{min}$ )  $\Gamma$  120 ms ( $t_{max}$ )
- Estimation de la distance parcourue par le robot : 1910.25 cm
- Estimation de l'incertitude finale de position : 15.6 cm



Illustrations :

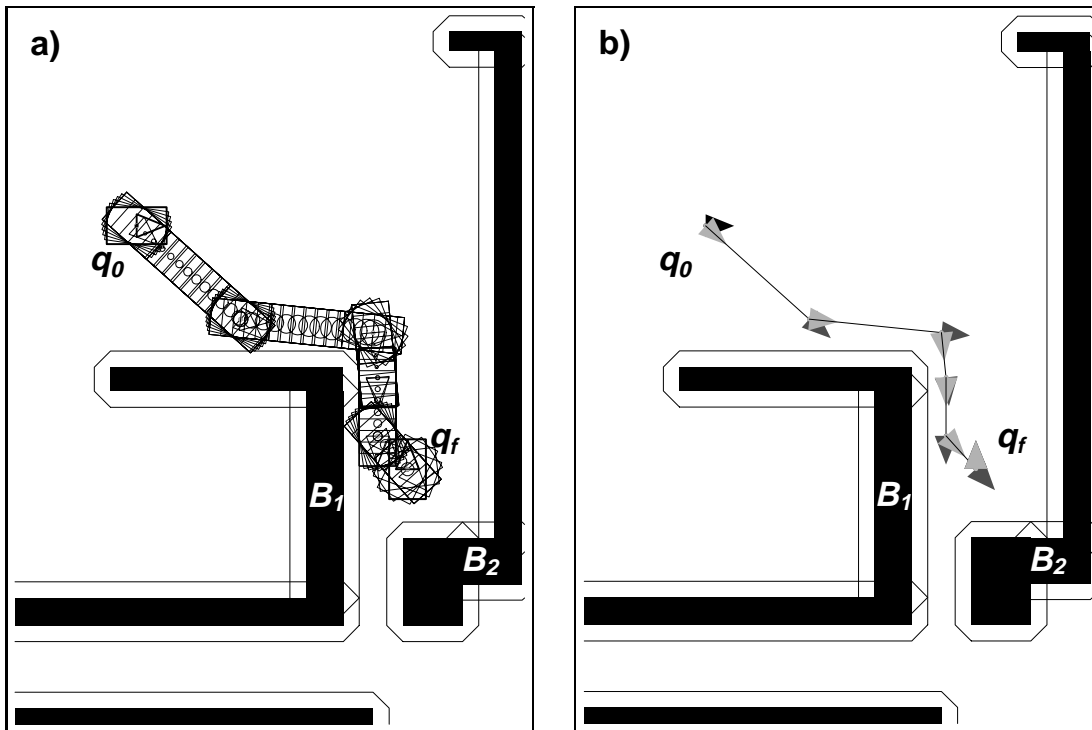


FIG. 6.33 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.

#### Plan exécutable par le robot $\mathcal{A}$

1. TURN( $\delta\theta = -41.9$ )
2. MOVE-UNTIL-CONTACT(sens = avant,  $d_{contact} = 35.0$ ,  $l = 304.4$ ,  $\epsilon_l = 15.7$ )
3. TURN( $\delta\theta = 36.0$ )
4. FOLLOW( $n_{murs} = 2$ ,  $d_{contact} = 35.0$ )  
Paramètres mur 1 :  
 côté = droit, sommet = convexe,  $\delta\theta = -84.1$ ,  $l_1 = 199.3$ ,  $l_2 = 51.2$ ,  
 détection vertex,  $d_{concave} = -$   
  
Paramètres mur 2 :  
 côté = droit, sommet = -,  $\delta\theta = -$ ,  $l_1 = 127.2$ ,  $l_2 = 0$ , distance parcourue,  
 $d_{concave} = -$
5. TURN( $\delta\theta = 42.0$ )
6. MOVE(sens = avant,  $\delta\theta = 0$ ,  $l = 97.0$ )
7. TURN( $\delta\theta = 137.9$ )

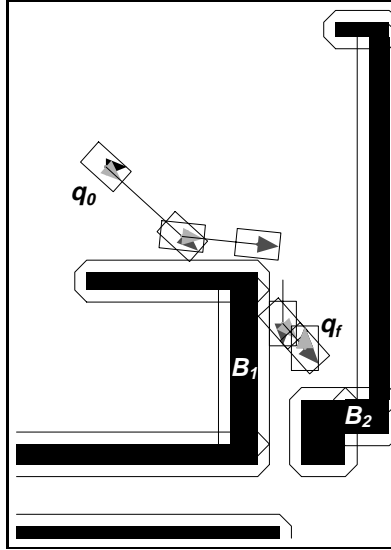


FIG. 6.34 – Transformation du plan généré en un plan exécutable.

### Commentaires :

Dans cet exemple la configuration initiale  $q_0$  du robot  $\mathcal{A}$  est récupérée par un système de vision placé sous le toit du bâtiment où l'on réalise les expériences (salle robotique du LAAS-CNRS). Cependant ce système ne peut fournir la configuration du robot que dans les cas où le robot se trouve à l'intérieur d'une région précise. Dans l'exemple la configuration initiale  $q_0$  est à la portée du système de vision mais ce n'est pas le cas pour la configuration finale  $q_f$ . Une petite incertitude de position a été affectée à la configuration initiale récupérée.

Bien que la distance de sécurité  $d_s$  spécifiée dans la primitive FOLLOW soit plus petite que la borne inférieure garantissant une rotation près des obstacles cette distance est suffisante pour effectuer la rotation qui lâche le contact avant de gagner la configuration finale  $q_f$  c'est-à-dire que l'exécution de l'avant dernière primitive TURN dans le plan est sans risque de collision.

### Evaluation de la solution :

- Nombre d'obstacles convexes définissant l'environnement : 11
- Nombre de commandes dans le plan : 7
- Temps de calcul du chemin : 1070 ms ( $t_{moyen}$ )  $\Gamma$  1060 ms ( $t_{min}$ )  $\Gamma$  1080 ms ( $t_{max}$ )
- Estimation de la distance parcourue par le robot : 753.5 cm
- Estimation de l'incertitude finale de position : 9.7 cm

### Description de la tâche (suite) :

Tâche inverse de la tâche précédente.

### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 9.7 cm.
- $\eta$  l'incertitude de contrôle : 2.5 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 25 cm.
- $\varrho$  l'angle de rupture de contact : 50 deg.
- $d_s$  la distance de sécurité : 35 cm.

### Illustrations :

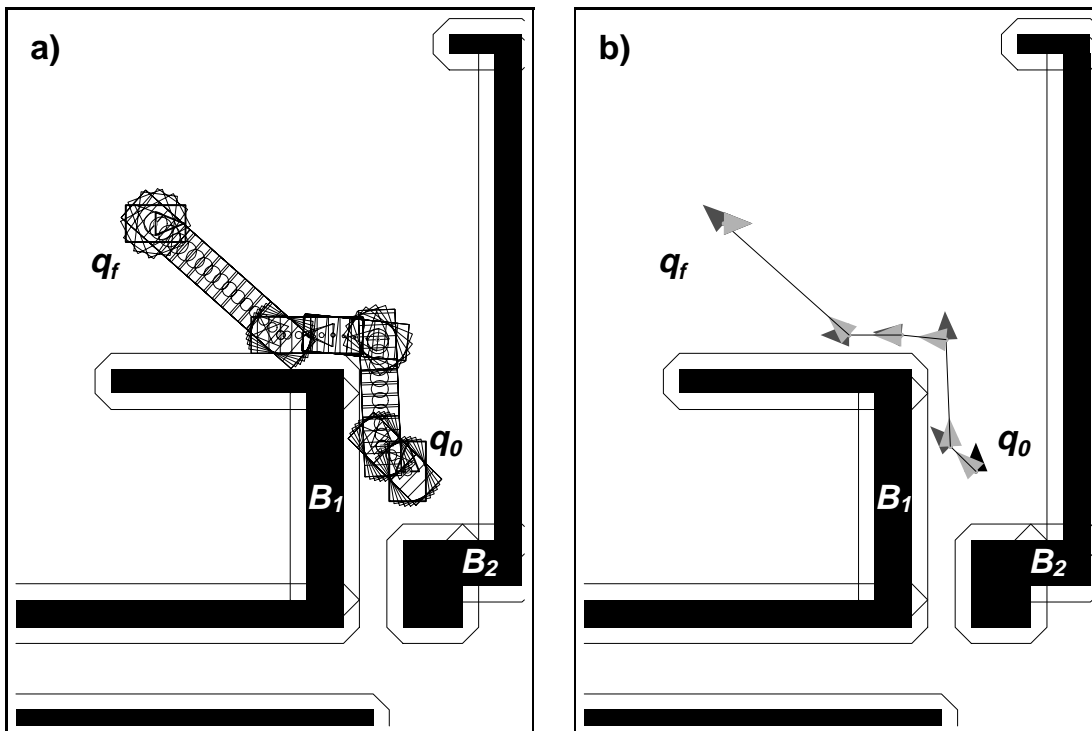


FIG. 6.35 – a) Chemin robuste généré par le planificateur.  
b) Plan robuste généré par le planificateur.

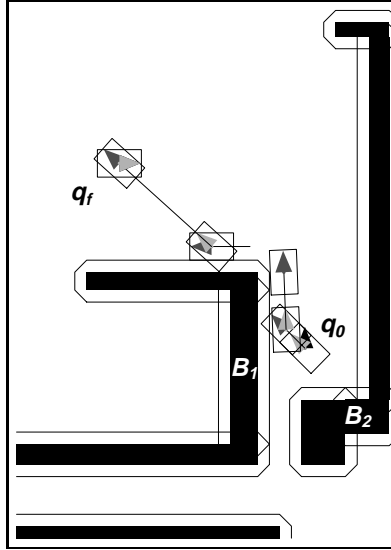


FIG. 6.36 – Transformation du plan généré en un plan exécutable.

Plan exécutable par le robot  $\mathcal{A}$

1. TURN( $\delta\theta = 45.5$ )
2. MOVE-UNTIL-CONTACT(sens = avant,  $d_{contact} = 35.0$ ,  $l = 77.8$ ,  $\epsilon_l = 13.1$ )
3. TURN( $\delta\theta = -43.2$ )
4. FOLLOW( $n_{murs} = 2$ ,  $d_{contact} = 35.0$ )  
Paramètres mur 1 :  
côté = gauche, sommet = convexe,  $\delta\theta = 87.6$ ,  $l_1 = 147.2$ ,  $l_2 = 40.8$ ,  
détection vertex,  $d_{concave} = -$   
  
Paramètres mur 2 :  
côté = gauche, sommet = -,  $\delta\theta = -$ ,  $l_1 = 113.9$ ,  $l_2 = 0$ , distance parcourue,  
 $d_{concave} = -$
5. TURN( $\delta\theta = -41.6$ )
6. MOVE(sens = avant,  $\delta\theta = 0$ ,  $l = 365.7$ )
7. TURN( $\delta\theta = -138.3$ )

**Commentaires :**

La configuration nominale finale  $q_f$  de la sous-tâche précédente devient la configuration initiale  $q_0$  de cette sous-tâche. De même  $\Gamma$  pour son estimation nominale sur l'incertitude de position.

La configuration nominale finale  $q_f$  de cette sous-tâche correspond à la configuration  $q_0$  récupérée pour la sous-tâche précédente.

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 11
- Nombre de commandes dans le plan : 7
- Temps de calcul du chemin : 1234 ms ( $t_{moyen}$ )  $\Gamma$  1230 ms ( $t_{min}$ )  $\Gamma$  1240 ms ( $t_{max}$ )
- Estimation de la distance parcourue par le robot : 725 cm
- Estimation de l'incertitude finale de position : 21.9 cm



FIG. 6.37 – Les images de la colonne de gauche illustrent l'exécution de la sous-tâche initiale avec le robot Hilare II, et celles de la colonne de droite illustrent la sous-tâche finale.



### 6.2.6.3 Problème 3. Navigation dans un environnement encombré (simulation)

#### Description de la tâche :

Planifier un plan robuste depuis  $q_0$  jusqu'à  $q_f$  et ensuite revenir sur  $q_0$  (figure 6.38). Pareil que pour la tâche 2 cette tâche est découpée en deux sous-tâches : une sous-tâche initiale pour amener le robot de  $q_0$  à  $q_f$  et une sous-tâche finale pour le faire revenir sur  $q_0$ .

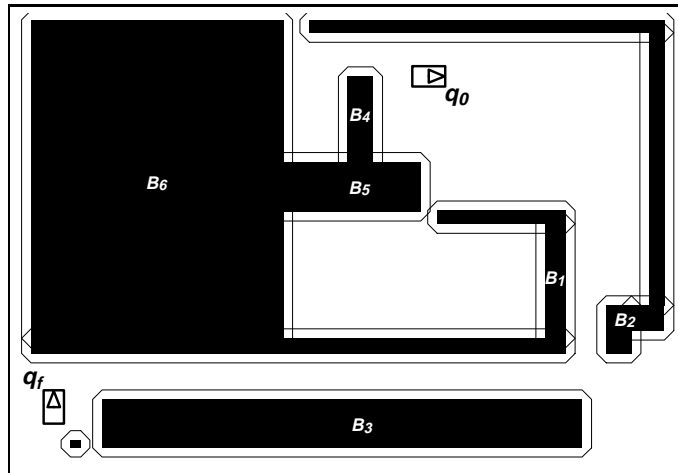


FIG. 6.38 – Définition de la tâche à résoudre.

#### Paramètres de la tâche :

- $\rho_0$  l'incertitude initiale de position : 2.5 cm.
- $\eta$  l'incertitude de contrôle : 2 deg.
- $\epsilon_t$  le pas de discrétisation en translation : 25 cm.
- $\varrho$  l'angle de rupture de contact : 85 deg.
- $d_s$  la distance de sécurité : 37 cm.

Illustrations :

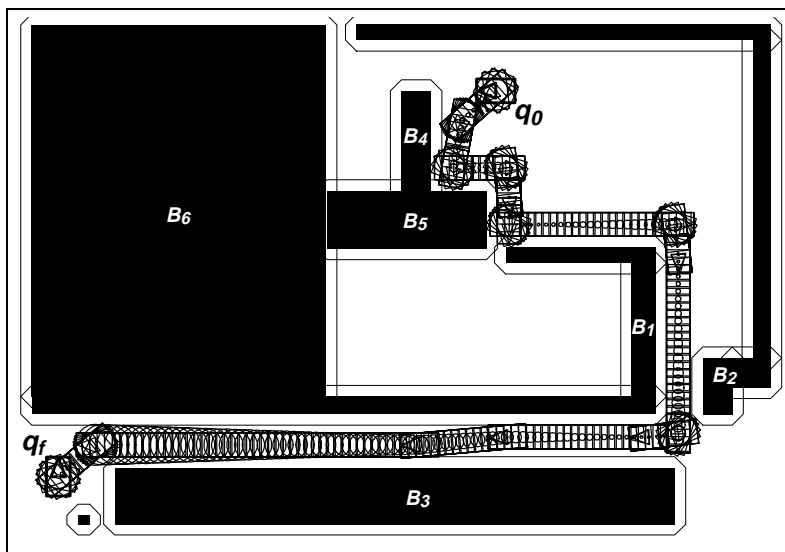


FIG. 6.39 – Chemin robuste généré par le planificateur.

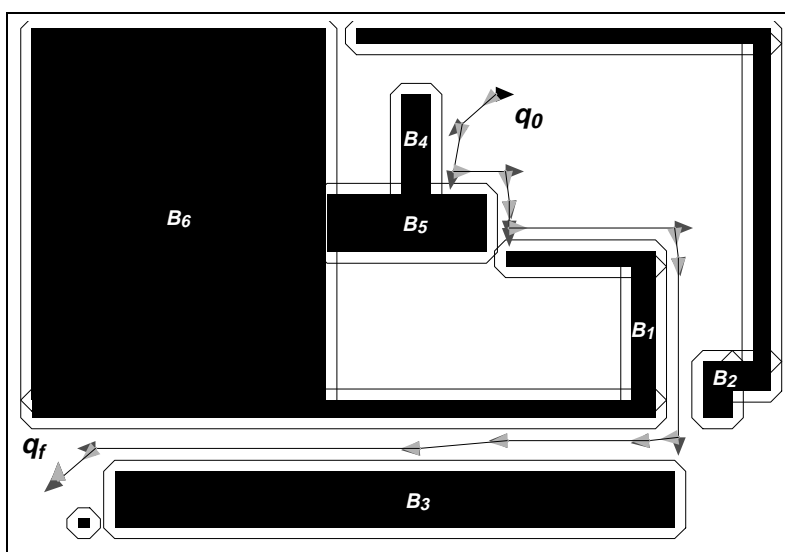


FIG. 6.40 – Plan robuste généré par le planificateur.

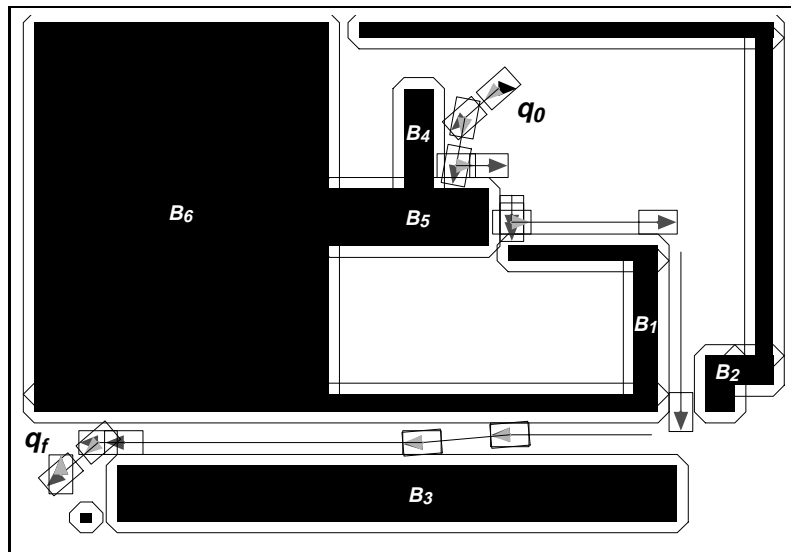


FIG. 6.41 – Transformation du plan généré en un plan exécutable.

Plan exécutable par le robot  $\mathcal{A}$

1. TURN( $\delta\theta = -138.7$ )
2. MOVE-UNTIL-CONTACT(sens = avant,  $d_{contact} = 37.0$ ,  $l = 151.3$ ,  $\epsilon_l = 7.7$ )
3. TURN( $\delta\theta = 38.3$ )
4. FOLLOW( $n_{murs} = 6$ ,  $d_{contact} = 37.0$ )  
Paramètres mur 1 :  
côté = droit, sommet = concave,  $\delta\theta = 91.0$ ,  $l_1 = 162.7$ ,  $l_2 = 10.9$ , détection vertex,  $d_{concave} = 15.0$   
  
Paramètres mur 2 :  
côté = droit, sommet = convexe,  $\delta\theta = -90.0$ ,  $l_1 = 93.75$ ,  $l_2 = 6.7$ , détection vertex,  $d_{concave} = -$   
  
Paramètres mur 3 :  
côté = droit, sommet = concave,  $\delta\theta = 89.9$ ,  $l_1 = 64.0$ ,  $l_2 = 4.6$ , détection vertex,  $d_{concave} = 15.0$   
  
Paramètres mur 4 :  
côté = droit, sommet = convexe,  $\delta\theta = -90.3$ ,  $l_1 = 476.7$ ,  $l_2 = 36.3$ , détection vertex,  $d_{concave} = -$   
  
Paramètres mur 5 :  
côté = droit, sommet = convexe,  $\delta\theta = -89.9$ ,  $l_1 = 520.4$ ,  $l_2 = 37.6$ , détection vertex,  $d_{concave} = -$   
  
Paramètres mur 6 :  
côté = droit, sommet = -,  $\delta\theta = -$ ,  $l_1 = 478.5$ ,  $l_2 = 0$ , distance parcourue,  $d_{concave} = -$
5. TURN( $\delta\theta = 5.0$ )
6. MOVE-UNTIL-CONTACT(sens = avant,  $d_{contact} = 37.0$ ,  $l = 298.0$ ,  $\epsilon_l = 27.1$ )
7. TURN( $\delta\theta = -5.0$ )
8. FOLLOW( $n_{murs} = 1$ ,  $d_{contact} = 37.0$ )  
Paramètres mur 1 :  
côté = gauche, sommet = convexe,  $\delta\theta = -$ ,  $l_1 = 946.5$ ,  $l_2 = 124.6$ , détection vertex,  $d_{concave} = -$
9. MOVE(sens = avant,  $\delta\theta = 0$ ,  $l = 87.0$ )
10. TURN( $\delta\theta = 40.3$ )
11. MOVE(sens = avant,  $\delta\theta = 0$ ,  $l = 165.3$ )
12. TURN( $\delta\theta = -130.3$ )

**Commentaires :**

Dans cet exemple il est nécessaire de spécifier une distance de sécurité  $d_s$  supérieure à la borne inférieure garantissant une rotation près des obstacles à cause des rotations sur les coins concaves lors de l'exécution de la primitive FOLLOW.

La primitive FOLLOW déclenche la détection des sommets concaves les plus près possible des obstacles en tenant compte de la portée des sensors ultrasoniques. De ce fait le paramètre  $d_{concave}$  correspond au seuil  $d_{ultrasons}$ . Seulement dans ce cas la contrainte sur la distance de sécurité  $d_s$  pourra être violée.

Après avoir exécuté la première primitive FOLLOW la suite consiste à avancer le long d'une autre paroi. Pour ceci des déplacements d'approche vers la nouvelle paroi sont nécessaires. De plus le changement de paroi exige un changement du côté d'asservissement.

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 12
- Nombre de commandes dans le plan : 12
- Temps de calcul du chemin : 3628 ms ( $t_{moyen}$ )  $\Gamma$  3040 ms ( $t_{min}$ )  $\Gamma$  4090 ms ( $t_{max}$ )
- Estimation de la distance parcourue par le robot : 3444.1 cm
- Estimation de l'incertitude finale de position : 5.77 cm



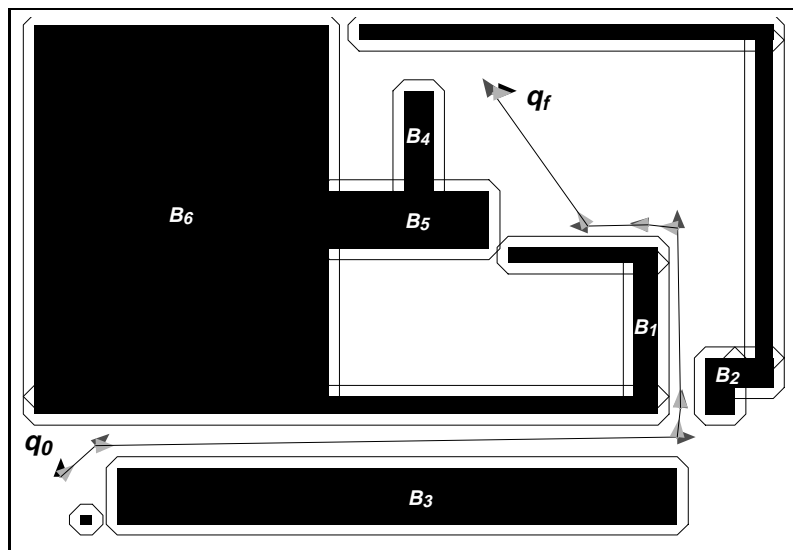


FIG. 6.43 – Plan robuste généré par le planificateur.

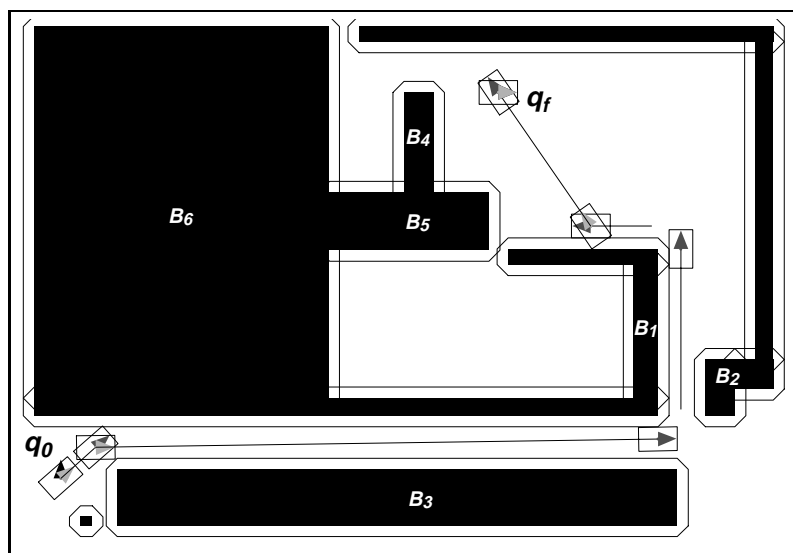


FIG. 6.44 – Transformation du plan généré en un plan exécutable.

Plan exécutable par le robot  $\mathcal{A}$

1. TURN( $\delta\theta = -48.7$ )
2. MOVE-UNTIL-CONTACT(sens = avant,  $d_{contact} = 37.0$ ,  $l = 157.1$ ,  $c_l = 11.1$ )
3. TURN( $\delta\theta = -40.4$ )
4. FOLLOW( $n_{murs} = 3$ ,  $d_{contact} = 24.0$ )
 

Paramètres mur 1 :  
côté = gauche, sommet = convexe,  $\delta\theta = 89.1$ ,  $l_1 = 1824.5$ ,  $l_2 = 155.2$ ,  
détection vertex,  $d_{concave} = -$

Paramètres mur 2 :  
côté = gauche, sommet = convexe,  $\delta\theta = 90.3$ ,  $l_1 = 520.4$ ,  $l_2 = 37.6$ ,  
détection vertex,  $d_{concave} = -$

Paramètres mur 3 :  
côté = gauche, sommet = -,  $\delta\theta = -$ ,  $l_1 = 204.5$ ,  $l_2 = 0$ , distance parcourue,  
 $d_{concave} = -$
5. TURN( $\delta\theta = -55.25$ )
6. MOVE(sens = avant,  $\delta\theta = 0$ ,  $l = 548.8$ )
7. TURN( $\delta\theta = -124.7$ )

**Evaluation de la solution :**

- Nombre d'obstacles convexes définissant l'environnement : 12
- Nombre de commandes dans le plan : 7
- Temps de calcul du chemin : 2554 ms ( $t_{moyen}$ )  $\Gamma$  2470 ms ( $t_{min}$ )  $\Gamma$  2630 ms ( $t_{max}$ )
- Estimation de la distance parcourue par le robot : 3255.3 cm
- Estimation de l'incertitude finale de position : 31.1 cm



### 6.2.7 Conclusion

L'application de l'approche proposée pour un robot mobile a été expliquée et illustrée avec quelques exemples dont un correspondant à une expérience réelle dans un environnement réel. Les premiers résultats nous ont permis d'une part d'intégrer les modules de planification et de contrôle pour un robot réel et d'autre part de valider les aspects de base de notre approche. Nous sommes convaincu que pour disposer d'une validation plus complète et donc d'une implantation opérationnelle plus fiable des expériences supplémentaires seraient nécessaires.

## 6.3 Conclusion

L'application de l'approche de planification de mouvements proposée a été validée pour deux types de tâches robotiques : (1) la manipulation d'une pièce sur un plan et (2) la navigation d'un robot mobile. Pour les tâches à résoudre la géométrie de l'"objet mobile" ainsi que celle des obstacles dans l'environnement est supposée  $2D\frac{1}{2}$ . Dans les deux cas de figure considérés l'évolution de l'incertitude liée à l'"objet mobile" (une pièce manipulée et un robot mobile respectivement) peut causer l'échec d'une tâche nominale satisfaisante. C'est pourquoi les "contacts" entre l'objet mobile et les obstacles sont utilisés comme réducteurs d'incertitude. Les solutions ainsi engendrées par notre planificateur sont "robustes" vis-à-vis des incertitudes sur l'objet mobile. Ces solutions sont faisables à condition que le système qui contrôle l'objet mobile puisse détecter et gérer les "contacts" de façon fiable. La résolution de problèmes variés pour chaque cas de figure illustre la génération des solutions sous des conditions de planification différentes.

# Chapitre 7

## Conclusion et perspectives

---

### 7.1 Conclusion

Dans ce mémoire une nouvelle approche de planification automatique de mouvements sous des contraintes d'incertitudes géométriques est proposée pour un robot holonome (ayant 2 degrés de liberté en translation et 1 degré de liberté en rotation). Les incertitudes considérées portent sur le positionnement (position/orientation) et sur le déplacement du robot. L'originalité de l'approche réside d'une part dans le principe de résolution et d'autre part dans la modélisation des incertitudes et son intégration dans l'approche. Notre approche de planification combine deux méthodes : (1) une recherche par potentiel fictif qui engendre des déplacements incrémentaux aussi bien dans l'espace libre que dans l'espace de contact ; (2) une exploration de l'espace de contact afin de contourner les minima locaux de la méthode précédente. Les "contacts" entre le robot et les obstacles sont utilisés pour réduire l'incertitude de position et d'orientation du robot. L'intégration des deux méthodes donne lieu à une *approche globale* de planification de mouvements.

Les deux méthodes précédentes prennent en considération les incertitudes et garantissent donc la génération de mouvements "robustes" pour le robot. Afin de garantir la "robustesse" d'une solution en présence d'incertitudes nous avons retenu des modèles d'incertitude de type *ensembliste*. Une méthode de calcul d'"atteignabilité" (appelée "*projection en avant*") tenant compte des incertitudes géométriques est proposée pour garantir les passages de l'espace libre vers l'espace de contact.

L'approche proposée engendre donc des chemins "robustes" vis-à-vis des incertitudes géométriques sur le robot. Ces chemins sont ensuite transformés en plans "robustes" composés de commandes de contrôle pouvant être exécutées par le robot. Parmi les commandes de contrôle celles qui considèrent les "contacts" sont basées sur des informations perceptives acquises par le robot lors de l'exécution.

L'approche de planification proposée a été validée pour deux types de tâches robotiques : (1) la manipulation d'une pièce sur un plan; (2) la navigation d'un robot mobile. Dans les deux cas de figure considérés la géométrie de l'"objet mobile" ainsi que celle des obstacles dans l'environnement est supposée  $2D\frac{1}{2}$ . Pour le deuxième cas d'application des résultats expérimentaux ont pu être obtenus grâce à l'intégration du planificateur sur la plate-forme du robot mobile Hilare II du LAAS-CNRS.

## 7.2 Perspectives

Dans cette partie nous distinguons d'une part les améliorations de l'approche originale et d'autre part son extension à d'autres problèmes de planification de mouvement.

### 7.2.1 Améliorations

En ce qui concerne la longueur des solutions engendrées une étape de lissage permettrait l'obtention de chemins plus courts. Cependant cette étape doit respecter l'estimation des incertitudes géométriques des solutions originales.

Comme nous l'avons vu précédemment les petits obstacles peuvent en raison des incertitudes engendrer une "région d'obstruction" pour le robot. Afin de contourner ce problème il est possible d'enrichir la planification : actuellement celle-ci se limite à la recherche de *balises* "au contact"; il suffit alors d'autoriser la recherche de *balises* dans l'espace libre. Il sera de ce fait nécessaire de considérer l'incertitude associée à ces nouvelles *balises*. Un autre avantage de cette amélioration permettra également d'avoir un ensemble plus large des "conditions d'atteignabilité" associées à certains obstacles difficilement "atteignables".

Il est également envisageable de prendre en compte les capacités de contrôle du robot utilisé afin d'enrichir les rotations au contact considérées par la planification.

### 7.2.2 Extensions

Des extensions de l'approche proposée pour la résolution d'autres problèmes de planification de mouvement peuvent être envisagées. Ces extensions sont obtenues en prenant en compte les contraintes dans le problème de planification traité. Des contraintes qui donnent lieu à des extensions intéressantes sont :

- la *contrainte cinématique de non-holonomie*: cette contrainte est propre aux robots mobiles de type voiture. Dans l'approche originale proposée une telle contrainte affecte la mobilité ainsi que l'analyse des incertitudes géométriques sur le robot. Des travaux de recherche prenant en compte les nouvelles contraintes

---

de mobilité et d'incertitude pour un robot rectangulaire *non-holonyme* sont en cours [Mer96].

- la *connaissance partielle de l'environnement*: cette contrainte implique d'une part une phase de planification effectuée en ligne (i.e. lors de l'exécution) et d'autre part une mise à jour incrémentale de l'environnement en intégrant des mesures perceptives sur le modèle courant de celui-ci. Le caractère incrémental de l'approche rend possible l'intégration de mesures perceptives durant la phase de planification. Cependant ces mesures sont entachées d'incertitude; donc l'analyse d'incertitude dans l'approche doit aussi prendre en compte la nouvelle incertitude sur le modèle de l'environnement.



---

# Annexe A

## Algorithmes

---

### A.1 Algorithme de la fonction de potentiel fictif

ENTRÉES :

- $\{\mathcal{B}_1, \dots, \mathcal{B}_n\}$  l'ensemble d'obstacles dans l'espace  $\mathcal{W}$  dont la géométrie ainsi que la position/orientation sont connues (sans incertitude);
- $\mathcal{A}$  le robot dont la géométrie est connue (sans incertitude);
- $q_0$  la configuration de départ. S'il s'agit d'une configuration en contact l'identification des obstacles en contact est spécifiée;
- $(\rho_0, \psi_0)$  l'incertitude de position et d'orientation sur la configuration  $q_0$ ;
- $q_f$  la configuration but. S'il s'agit d'une configuration en contact l'identification des obstacles en contact est spécifiée;
- $(\rho_f, \psi_f)$  les incertitudes maximales de position et d'orientation acceptées sur la configuration  $q_f$ ;
- $\eta$  l'incertitude de contrôle sur les déplacements en translation;
- $k_r$  la constante de proportionnalité sur les imprécisions en rotation;
- $\epsilon_t$  le pas de discrétisation en translation;
- $\varrho$  l'angle de rupture de contact.

SORTIES :

- $q_c$  la dernière configuration atteignable par le robot. S'il s'agit d'une configuration en contact l'identification des obstacles en contact est spécifiée;
- $(\rho_c, \psi_c)$  l'incertitude de position et d'orientation sur la configuration  $q_c$ ;

- $\tau_{pot}$  le chemin *valide et robuste* reliant la configuration  $q_0$  à la configuration  $q_c$ ;
- La condition d'arrêt à la configuration  $q_c$  :
  - Valeur SUCCÈS-EN-DESSOUS si la configuration  $q_f$  est atteinte depuis la configuration  $q_0$  avec des estimations d'incertitude de position et d'orientation en dessous des valeurs maximales  $\rho_f$  et  $\psi_f$ .
  - Valeur SUCCÈS-EN-DESSUS si la configuration  $q_f$  est atteinte depuis la configuration  $q_0$  avec des estimations d'incertitude de position et d'orientation en dessus des valeurs maximales  $\rho_f$  et/ou  $\psi_f$ .
  - Valeur ECHEC-BLOCAGE si la fonction de potentiel fictif tombe dans un minimum local suivant le chemin vers la configuration  $q_f$ . Dans ce cas la configuration de minimum  $q_{min}$  avec ses estimations d'incertitude de position  $\rho_{min}$  et d'orientation  $\psi_{min}$  ainsi que l'identification des obstacles en contact sont retournées.
  - Valeur ECHEC-CONTACT-NON-ATTEIGNABLE si la fonction de potentiel fictif rencontre une surface de contact qui ne satisfait pas la condition d'“atteignabilité” sous les contraintes d'incertitudes géométriques imposées. Dans ce cas la dernière configuration de contact atteignable  $q_{dern-att}$  ainsi que l'identification des obstacles en contact sont retournées.

ALGORITHME :

1. INITIALISATION.

$$\begin{aligned} q_c &= q_0 \Gamma \\ q_{dern-att} &= q_0 \Gamma \\ \rho_c &= \rho_0 \Gamma \\ \psi_c &= \psi_0. \end{aligned}$$

2. TEST DE BUT ATTEINT.

Si Configuration-atteinte( $q_c, q_f$ ) Alors  
 Si Valeurs-d'incertitude-respectées( $\rho_c, \rho_f, \psi_c, \psi_f$ )  
 Alors Retourner(SUCCÈS-EN-DESSOUS).  $\square$   
 Sinon Retourner(SUCCÈS-EN-DESSUS).  $\square$

3. COMPORTEMENT DANS L'ESPACE LIBRE (cf. §5.2.2).

Si  $q_c \in \mathcal{C}_{libre}$  :

Déterminer  $\vec{\Delta}q(q_c, q_f) \Gamma / * \text{déplac. libre} *$   
 Suivre-déplacement( $\vec{\Delta}q(q_c, q_f)$ ) arrêté en  $q'_c \Gamma$   
 Selon la condition d'arrêt :  
 En cas de DÉPLACEMENT-ACCOMPLI :  
 Déterminer  $\delta\rho \Gamma \delta\psi \Gamma \rho_{c'}$  et  $\psi_{c'}$  (équations 5.11 et 5.12)  $\Gamma$

Si  $C(q'_c, \rho_{c'}, \psi_{c'}) \subset \mathcal{C}_{libre}$   
 Alors  $q_c = q'_c$  et Aller à 2.  
 Sinon Aller à 4.

En cas de CONTACT-EN-DÉPLACEMENT :  
 Aller à 4.

#### 4. PASSAGE DE L'ESPACE LIBRE VERS L'ESPACE DE CONTACT (cf. §5.2.3).

Si  $q_c \in \mathcal{C}_{libre}$  et  $(\vec{\Delta}q(q_c, q_f) \cap \mathcal{C}_{contact} \neq \emptyset$  ou  $C(q_c, \rho_c, \psi_c) \not\subset \mathcal{C}_{libre}$ ) :

$q_c = \text{Intersection-nominale}(q_{dern-att}, q_c, S)\Gamma$   
 Si Configuration-atteignable( $q_{dern-att}, q_c, S$ )

Alors

Déterminer  $p$  et  $\theta\Gamma$   
 Suivre-déplacement( $\Delta_\theta^p(q_c, p, \theta)$ ) arrêté en  $q'_c\Gamma$   
 $q_{dern-att} = q'_c$  et Aller à 2.

Sinon

$q'_c = \text{Choisir-configuration-atteignable}(q_{dern-att}, S)\Gamma$   
 Déterminer  $\vec{\Delta}q'_t(p_{dern-att}, p'_c)\Gamma$  /\* déplac. gardé \*/  
 Suivre-déplacement( $\vec{\Delta}q'_t(p_{dern-att}, p'_c)$ ) arrêté en  $q'_c\Gamma$   
 Selon la condition d'arrêt :

En cas de DÉPLACEMENT-ACCOMPLI :

Déterminer  $p$  et  $\theta\Gamma$   
 Suivre-déplacement( $\Delta_\theta^p(q'_c, p, \theta)$ ) arrêté en  $q''_c\Gamma$   
 $q_c = q''_c, q_{dern-att} = q''_c\Gamma$  et Aller à 2.

En cas de DÉTECTION-BLOCAGE :

$q_c = q_{min}, q_{dern-att} = q_{min}\Gamma$   
 Retourner(ÉCHEC-BLOCAGE).  $\square$

En cas de CONTACT-NON-ATTEIGNABLE :

$q_c = q_{dern-att}\Gamma$   
 Retourner(ÉCHEC-CONTACT-NON-ATTEIGNABLE).  $\square$

#### 5a. COMPORTEMENT DANS L'ESPACE DE CONTACT.

Si  $q_c$  est à l'intérieur d'une arête  $\vec{E}_j^{\mathcal{CB}_i, \theta} \subset \mathcal{C}_{contact}$  (cf. §5.2.4.2) :

Déterminer  $\vec{\Delta}q_t(p_c, p_f)\Gamma$

Selon la valeur de  $\vec{\Delta}q_t \cdot \vec{n}_j^{\mathcal{CB}_i, \theta}$  :

En cas de VALEUR NÉGATIF : /\* Cas 1 \*/

Si  $\vec{u}(\vec{\Delta}q_t) \cdot \vec{n}_j^{\mathcal{CB}_i, \theta} \neq -1$   
 Alors  $\vec{\Delta}q'_t = \vec{\Delta}q_t^{\leftrightarrow}$  /\* déplac. compliant \*/.  
 Sinon Retourner(ÉCHEC-BLOCAGE).  $\square$

En cas de VALEUR POSITIF : /\* Cas 2 \*/

Si  $\vec{u}(\vec{\Delta}q_t) \cdot \vec{n}_j^{\mathcal{CB}_i} \geq \cos(\varrho)$



Alors  $\vec{\Delta}q'_t = \vec{\Delta}q_t$  /\* déplac. libre \*/.

Sinon  $\vec{\Delta}q'_t = \vec{\Delta}q_t^{\leftrightarrow}$  /\* déplac. compliant \*/.

En cas de VALEUR NULLE : /\* Cas 3 \*/

$\vec{\Delta}q'_t = \vec{\Delta}q_t^{\leftrightarrow}$  /\* déplac. compliant \*/.

Suivre-déplacement( $\vec{\Delta}q'_t(p_c, p_f)$ ) arrêté en  $q'_c\Gamma$

Selon la condition d'arrêt :

En cas de DÉPLACEMENT-ACCOMPLI :

Déterminer  $\delta\rho\Gamma\rho_{c'}$  et  $\psi_{c'}$  (équations 5.15Γ5.16)Γ

$q_{dern-att} = q'_c\Gamma q_c = q'_c\Gamma$  et Aller à 2.

En cas de DÉTECTION-SOMMET :

$\rho_{c'} = 0\Gamma q_c = q'_c$  et Aller à 5b.

En cas de DÉTECTION-BLOCAGE :

$q_c = q_{min}$  et Retourner(ÉCHEC-BLOCAGE). □

#### 5b. COMPORTEMENT DANS L'ESPACE DE CONTACT (suite).

Si  $q_c$  correspond à un sommet  $s_j^{\mathcal{CB}_{i,\theta}} \in \mathcal{C}_{contact}$  (cf. §5.2.4.3) :

Si Concavité-locale( $s_j^{\mathcal{CB}_{i,\theta}} \Gamma \mathcal{CB}_{i,\theta}$ )

Alors

Déterminer  $\vec{\Delta}q''_t$  le long de l'arête adjacente  $\vec{E}_k^{\mathcal{CB}_{i,\theta}}\Gamma$

Suivre-déplacement( $\vec{\Delta}q''_t$ ) arrêté en  $q_c\Gamma$  et Aller à 2.

Sinon /\* convexité-locale \*/

Déterminer  $\vec{\Delta}q_t(p_c, p_f)$

Si Intérieur-cône-rupture( $\vec{\Delta}q_t\Gamma s_j^{\mathcal{CB}_{i,\theta}} \Gamma \mathcal{CB}_{i,\theta}$ )

Alors

Suivre-déplacement( $\vec{\Delta}q_t(p_c, p_f)$ ) arrêté en  $q_c\Gamma$  et Aller à 2.

Sinon

Déterminer  $p\Gamma s$  et  $|\theta|\Gamma$

Suivre-déplacement( $\Delta_\theta^p(q_c, p, s|\theta)$ ) arrêté en  $q'_c\Gamma$

Déterminer  $\vec{\Delta}q_t(p'_c, p_f)\Gamma$

Si  $\vec{\Delta}q_t(p'_c, p_f) \cdot \vec{D}_k^{\mathcal{CB}_{i,\theta'_c}} > 0$  et  $\vec{u}(\vec{\Delta}q_t(p'_c, p_f)) \cdot \vec{n}_k^{\mathcal{CB}_{i,\theta'_c}} < \cos(\varrho)$

Alors  $\vec{\Delta}q''_t = \vec{\Delta}q_t^{\leftrightarrow}$  /\* déplac. compliant \*/

Sinon  $\vec{\Delta}q''_t = \vec{\Delta}q_t$  /\* déplac. libre \*/

Suivre-déplacement( $\vec{\Delta}q''_t(p'_c, p_f)$ ) arrêté en  $q''_c\Gamma$

Selon la condition d'arrêt :

En cas de DÉPLACEMENT-ACCOMPLI :

Déterminer  $\rho_{c''}$  et  $\psi_{c''}$  (équation 5.19)

$q_c = q''_c\Gamma q_{dern-att} = q''_c\Gamma$  et Aller à 2.

En cas de DÉTECTION-BLOCAGE :

$q_c = q_{min}$  et Retourner(ÉCHEC-BLOCAGE). □

## A.2 Un algorithme séquentiel de notre approche de planification

ENTRÉES :

- $\{\mathcal{B}_1, \dots, \mathcal{B}_n\}$  l'ensemble d'obstacles dans l'espace  $\mathcal{W}$  dont la géométrie ainsi que la position/orientation sont connues (sans incertitude);
- $\mathcal{A}$  le robot dont la géométrie est connue (sans incertitude);
- $q_0$  la configuration initiale. S'il s'agit d'une configuration en contact l'identification des obstacles en contact est spécifiée;
- $(\rho_0, \psi_0)$  l'incertitude de position et d'orientation sur la configuration  $q_0$ ;
- $q_f$  la configuration but. S'il s'agit d'une configuration en contact l'identification des obstacles en contact est spécifiée;
- $(\rho_f, \psi_f)$  les incertitudes maximales de position et d'orientation acceptées sur la configuration  $q_f$ ;
- $\eta$  l'incertitude de contrôle sur les déplacements en translation;
- $k_t$  la constante de proportionnalité sur les imprécisions des magnitudes des déplacements en translation<sup>1</sup>;
- $k_r$  la constante de proportionnalité sur les imprécisions en rotation;
- $\epsilon_t$  le pas de discrétisation en translation;
- $\varrho$  l'angle de rupture de contact.

SORTIES :

- Valeur SUCCÈS si la configuration  $q_f$  est atteinte depuis la configuration  $q_0$  en respectant les estimations d'incertitude maximales  $\rho_f$  et  $\psi_f$ . Un chemin *valide* et *robuste*  $\tau = \{\tau_1, \dots, \tau_m\}$  pour le robot ponctuel  $q_A$  depuis  $q_0$  vers  $q_f$  est aussi fourni.
- Valeur ECHEC dans le cas contraire.

ALGORITHME :

1. Initialiser les listes  $\mathcal{L}$  et  $\mathcal{L}_i$  et de l'arbre d'exploration  $\mathcal{T}$  avec la balise  $l_{i,\theta_0}^0$  correspondante à la configuration initiale  $q_0$  (§5.3.2).

---

1. Selon une hypothèse dans §2.3.2, nous supposons  $k_t = \tan(\eta)$ .

2. Fonction de Potentiel Fictif ( $\{\mathcal{B}_1, \dots, \mathcal{B}_n\} \Gamma \mathcal{A} \Gamma q_0 \Gamma (\rho_0, \psi_0) \Gamma q_f \Gamma (\rho_f, \psi_f) \Gamma \eta \Gamma k_r \Gamma \epsilon_t \Gamma \varrho$ ).  
 Les valeurs retournées correspondent à la dernière configuration atteignable  $q_c$  avec ses estimations d'incertitude ( $\rho_c \mathbb{P} \psi_c$ ) ainsi que le chemin  $\tau_{pot}$  reliant  $q_0$  à  $q_c$ .  
 Selon la condition d'arrêt retournée :

En cas de SUCCÈS-EN-DESSOUS :

Affecter  $\tau = \tau_{pot}$ ;

Retourner SUCCÈS.  $\square$

En cas de ÉCHEC-BLOPAGE ou ÉCHEC-CONTACT-NON-ATTEIGNABLE :

Mettre à jour les listes  $\mathcal{L}$  et  $\mathcal{L}_i$  et l'arbre d'exploration  $\mathcal{T}$  avec

les balises correspondant à  $q_0$  et  $q_c$  plus le chemin  $\tau_{pot}$ ;

Aller au pas 3.

En cas de SUCCÈS-EN-DESSUS :

Aller au pas 3.

3. Pour chaque liste non vide  $\mathcal{L}_i \subset \mathcal{L} \Gamma 1 \leq i \leq n$  :

- (a) Exploration de l'espace de contact adjacent ( $\mathcal{L}_i, k_t, \epsilon_t$ ).

Les valeurs retournées correspondent à la nouvelle balise  $l_{i,\theta_f}^k$  avec ses estimations d'incertitude ( $\rho_{l_{i,\theta_f}^k} \mathbb{P} \psi_{l_{i,\theta_f}^k}$ ) la balise  $l_{i,\theta_f}^j$  explorée dans  $\mathcal{L}_i$  ainsi que le chemin  $\tau_{adj}$  entre les balises.

- (b) Mettre à jour les listes  $\mathcal{L}$  et  $\mathcal{L}_i$  et l'arbre d'exploration  $\mathcal{T}$  avec les balises  $l_{i,\theta_f}^j$  et  $l_{i,\theta_f}^k$  plus le chemin  $\tau_{adj}$ .

- (c) Fonction de Potentiel Fictif ( $\{\mathcal{B}_1, \dots, \mathcal{B}_n\} \Gamma \mathcal{A} \Gamma l_{i,\theta_f}^k \Gamma (\rho_{l_{i,\theta_f}^k}, \psi_{l_{i,\theta_f}^k}) \Gamma q_f \Gamma (\rho_f, \psi_f) \Gamma \eta \Gamma k_r \Gamma \epsilon_t \Gamma \varrho$ ).

Les valeurs retournées correspondent à la dernière configuration atteignable  $q_c$  avec ses estimations d'incertitude ( $\rho_c \mathbb{P} \psi_c$ ) ainsi que le chemin  $\tau_{pot}$  reliant  $l_{i,\theta_f}^k$  à  $q_c$ .

Selon la condition d'arrêt retournée :

En cas de SUCCÈS-EN-DESSOUS :

Affecter  $\tau = \{\tau_{prev}, \tau_{pot}\}$  dont  $\tau_{prev}$  correspond au chemin depuis  $q_0$  jusqu'à  $l_{i,\theta_f}^k$  et celui-ci reconstruit à l'aide de l'arbre d'exploration  $\mathcal{T}$ ;

Retourner SUCCÈS.  $\square$

En cas de ÉCHEC-BLOPAGE ou ÉCHEC-CONTACT-NON-ATTEIGNABLE :

Mettre à jour les listes  $\mathcal{L}$  et  $\mathcal{L}_i$  et l'arbre d'exploration  $\mathcal{T}$

avec la balise  $l_{i,\theta_f}^k$  et celle de  $q_c$  plus le chemin  $\tau_{pot}$ ;

Itérer la boucle en 3.

En cas de SUCCÈS-EN-DESSUS :

Itérer la boucle en 3.

4. (a) Exploration de l'espace de contact visible ( $\mathcal{L}, \eta, \epsilon_t$ ).

Les valeurs retournées correspondent à la nouvelle balise  $l_{j,\theta_f}^k$  avec ses esti-

mations d'incertitude  $(\rho_{l_{j,\theta_f}^k}, \psi_{l_{j,\theta_f}^k})$  la balise explorée  $l_{h,\theta_f}^g$  dans  $\mathcal{L}$  ainsi que le chemin  $\tau_{vis}$  entre les balises.

- (b) Mettre à jour les listes  $\mathcal{L}$  et  $\mathcal{L}_j$  et l'arbre d'exploration  $\mathcal{T}$  avec les balises  $l_{h,\theta_f}^g$  et  $l_{j,\theta_f}^k$  plus le chemin  $\tau_{vis}$ .
- (c) **Fonction de Potentiel Fictif**  $(\{\mathcal{B}_1, \dots, \mathcal{B}_n\} \Gamma \mathcal{A} \Gamma l_{j,\theta_f}^k \Gamma (\rho_{l_{j,\theta_f}^k}, \psi_{l_{j,\theta_f}^k}) \Gamma q_f \Gamma (\rho_f, \psi_f) \Gamma \eta \Gamma k_r \Gamma \epsilon_t \Gamma \varrho)$ .

Les valeurs retournées correspondent à la dernière configuration atteignable  $q_c$  avec ses estimations d'incertitude  $(\rho_c, \psi_c)$  ainsi que le chemin  $\tau_{pot}$  reliant  $l_{j,\theta_f}^k$  à  $q_c$ .

Selon la condition d'arrêt retournée :

En cas de **SUCCÈS-EN-DESSOUS** :

Affecter  $\tau = \{\tau_{prev}, \tau_{pot}\}$  dont  $\tau_{prev}$  correspond au chemin depuis  $q_0$  jusqu'à  $l_{j,\theta_f}^k$  celui-ci reconstruit à l'aide de l'arbre d'exploration  $\mathcal{T}$ ;

Retourner **SUCCÈS**.  $\square$

En cas de **ÉCHEC-BLOPAGE** ou **ÉCHEC-CONTACT-NON-ATTEIGNABLE** :

Mettre à jour les listes  $\mathcal{L}$  et  $\mathcal{L}_i$  et l'arbre d'exploration  $\mathcal{T}$  avec la balise  $l_{j,\theta_f}^k$  et celle de la configuration  $q_c$  plus le chemin  $\tau_{pot}$ ;

Aller au pas 3.

En cas de **SUCCÈS-EN-DESSUS** :

Aller au pas 3.

### A.3 Algorithme de construction d'un obstacle virtuel

ENTRÉES :

- $\mathcal{B}_i$  l'obstacle réel;
- $d_s$  la distance de sécurité.

SORTIES :

- $\mathcal{V}_i$  l'obstacle virtuel.

ALGORITHME :

1. DÉFINITION DES ARÊTES PARALLÈLES.

Pour chaque arête  $\vec{E}_j^{\mathcal{B}_i}$  du contour  $\mathcal{B}_i\Gamma$  définir l'arête parallèle  $\vec{E}_j^{\mathcal{V}_i}$  suivant la direction du vecteur normal extérieur  $\vec{n}_i^{\mathcal{B}_i}$  à une distance  $d_s$ . Donc  $\Gamma$   
 $\|\vec{E}_j^{\mathcal{V}_i}\| = \|\vec{E}_j^{\mathcal{B}_i}\|$ .

2. LIAISON ENTRE LES ARÊTES.

Les arêtes  $\vec{E}_j^{\mathcal{V}_i}$  définies à partir de deux arêtes  $\vec{E}_j^{\mathcal{B}_i}$  formant un sommet convexe seront reliées par une arête de liaison.

En ce qui concerne les sommets concaves de l'obstacle  $\mathcal{B}_i\Gamma$  ils seront redéfinis dans l'obstacle  $\mathcal{V}_i$  comme l'intersection des arêtes parallèles  $\vec{E}_j^{\mathcal{V}_i}$  respectives.

---

# Table des figures

---

1.1	Modèle de <i>pré-image</i> tenant compte de l'incertitude de contrôle. . . . .	8
1.2	Principe de chaînage arrière de <i>pré-images</i> . . . . .	9
1.3	Application du modèle de <i>pré-image</i> pour atteindre des régions circulaires. . . . .	11
1.4	Modèle de visibilité sur une arête tenant compte des incertitudes de position et contrôle. . . . .	12
1.5	Stratégie de re-localisation à l'intérieur d'une "landmark". . . . .	13
1.6	Représentation matricielle de l'erreur de position. . . . .	14
1.7	Matérialisation d'une représentation probabiliste des incertitudes sur la localisation d'un robot mobile. . . . .	16
2.1	Définition d'un polygone simple $P$ . . . . .	24
2.2	Construction d'un $\mathcal{C}$ -obstacle $\mathcal{CB}_{i,\theta}$ dans l'espace des configurations $\mathbb{R}^2$ à partir d'un robot et un obstacle convexes. . . . .	26
2.3	Construction discrète d'un $\mathcal{C}$ -obstacle $\mathcal{CB}_i$ dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$ à partir d'un robot et un obstacle convexes. . . . .	26
2.4	Modèles explicites d'incertitude de position. . . . .	27
2.5	Evolution explicite de l'incertitude de position dans l'espace $\mathbb{R}^2$ . . . . .	28
2.6	L'incertitude de contrôle et son effet sur l'incertitude de position. . . . .	29
2.7	L'incertitude d'orientation dans l'espace $\mathbb{R}^2$ . . . . .	31
2.8	Modèle d'intégration des incertitudes de position et d'orientation. . . . .	33
2.9	Définition de la surface discrète d'un obstacle $\mathcal{CB}_i$ dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$ . . . . .	34
2.10	Définition des sous-espaces $S^+ \Gamma S$ et $S^-$ dans $\mathbb{R}^2$ . . . . .	35
2.11	Calcul de l'"atteignabilité" dans l'espace des configurations $\mathbb{R}^2$ . . . . .	37
2.12	Cylindre d'incertitude à projeter sur une surface d'un $\mathcal{C}$ -obstacle à l'aide du modèle "projection en avant" dans l'espace des configurations $\mathbb{R}^2$ . . . . .	39
2.13	Calcul de l'"atteignabilité" des segments appartenant à la surface d'un $\mathcal{C}$ -obstacle. . . . .	39
2.14	Vitesses communes en translation atteignant une surface d'un $\mathcal{C}$ -obstacle dans l'espace $\mathbb{R}^2 \times [0, 2\pi]$ malgré les incertitudes de position $\Gamma$ d'orientation et de contrôle. . . . .	40

2.15	Région $R$ de $S$ obtenue à l'aide du modèle "projection en avant" dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$ . . . . .	41
3.1	Définition de la sous-classe de contacts simples entre une pièce manipulée et un obstacle. . . . .	49
3.2	Illustration de contacts doubles sur une pièce manipulée. . . . .	50
3.3	Illustration des mouvements compliant libre et gardé pour une pièce manipulée. . . . .	51
3.4	Spécification de la primitive <b>Commande-libre-en-distance</b> . . . . .	53
3.5	Spécification de la primitive <b>Commande-gardée</b> . . . . .	54
3.6	Spécification de la primitive <b>Commande-gardée-jusqu'à-alignement</b> . . . . .	54
3.7	Spécification de la primitive <b>Commande-compliante-en-distance</b> . . . . .	55
3.8	Spécification de la primitive <b>Commande-compliante-jusqu'à-lâcher-contact</b> . . . . .	56
3.9	Spécification de la primitive <b>Commande-compliante-jusqu'au-contact</b> . . . . .	57
3.10	Spécification de la primitive <b>Commande-compliante-multiple-jusqu'à-lâcher-contact</b> . . . . .	57
4.1	Représentation d'une fonction classique de potentiel fictif. . . . .	61
4.2	Caractéristiques d'une fonction de potentiel fictif adaptée à l'exploration de l'espace valide $\mathcal{C}_{valide}$ . . . . .	63
4.3	Construction du graphe de balises avec la fonction <b>EXPLORE</b> du Fil d'Ariane. . . . .	64
4.4	Stratégies d'exploration dans l'espace $\mathcal{C}_{contact}$ définies pour contourner une configuration de minimum local. . . . .	67
4.5	Un exemple de l'approche de planification proposée dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$ . . . . .	69
4.6	Un exemple de l'approche de planification proposée dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$ (suite). . . . .	70
4.7	Un exemple de l'approche de planification proposée dans l'espace des configurations $\mathbb{R}^2 \times [0, 2\pi]$ (fin). . . . .	71
5.1	Algorithme de construction d'un $\mathcal{C}$ -obstacle $\mathcal{CB}_{i,\theta}$ dans l'espace des configurations $\mathbb{R}^2$ à partir d'un robot et un obstacle convexes. . . . .	75
5.2	Construction d'un $\mathcal{C}$ -obstacle $\mathcal{CB}_{i,\theta}$ dans l'espace des configurations $\mathbb{R}^2$ à partir d'un robot convexe et un obstacle concave. . . . .	76
5.3	Paramètres d'"alignement" associés aux contacts simples. . . . .	77
5.4	Relation induite par un déplacement simultané de translation/rotation entre la distance euclidienne et la distance angulaire. . . . .	80
5.5	Comportement du potentiel fictif dans l'espace $\mathcal{C}_{libre}$ . . . . .	82
5.6	Condition d'"atteignabilité" d'un contact. . . . .	84
5.7	Projection du modèle cylindrique d'incertitude de position/orientation sur la surface d'un $\mathcal{C}$ -obstacle. . . . .	84
5.8	Mouvement d'"alignement" entre le robot et un obstacle. . . . .	85
5.9	Le cône de rupture de contact. . . . .	87

5.10	Comportement du potentiel fictif dans l'espace $\mathcal{C}_{contact}$ (cas 1).	88
5.11	Comportement du potentiel fictif dans l'espace $\mathcal{C}_{contact}$ (cas 2).	89
5.12	Comportement du potentiel fictif dans l'espace $\mathcal{C}_{contact}$ (cas 3).	90
5.13	Mouvement d'“alignement” autour des extrémités des surfaces des obstacles.	93
5.14	Mouvement d'“alignement” autour des extrémités des surfaces des obstacles (suite).	94
5.15	Sous-espace de contact adjacent autour d'une balise.	98
5.16	Discrétisation et exploration du sous-espace de contact adjacent.	101
5.17	Illustration du cône des translations libres.	104
5.18	Sous-espace de contact visible depuis une balise.	104
5.19	Discrétisation et exploration du sous-espace de contact visible.	108
5.20	Intégration des composantes de l'approche de planification dans un modèle réparti.	111
6.1	Problème 1 de planification à résoudre pour une pièce manipulée.	121
6.2	Résolution du problème 1 de planification posé.	122
6.3	Problème 2a de planification à résoudre pour une pièce manipulée.	124
6.4	Résolution du problème 2a de planification posé.	125
6.5	Résolution du problème 2b de planification posé.	126
6.6	Problème 2c de planification à résoudre pour une pièce manipulée.	128
6.7	Résolution du problème 2c de planification posé.	128
6.8	Problème 2d de planification à résoudre pour une pièce manipulée.	130
6.9	Résolution du problème 2d de planification posé.	131
6.10	Problème 3 de planification à résoudre pour une pièce manipulée.	133
6.11	Résolution du problème 3 de planification posé.	133
6.12	Problème 4 de planification à résoudre pour une pièce manipulée.	135
6.13	Résolution du problème 4 de planification posé.	136
6.14	Résolution du problème 4 de planification posé (suite).	136
6.15	Problème 5 de planification à résoudre pour une pièce manipulée.	139
6.16	Résolution du problème 5 de planification posé.	140
6.17	Problème 6 de planification à résoudre pour une pièce manipulée.	144
6.18	Minimum local “non-atteignable” pour le problème 6 de planification.	145
6.19	Un chemin solution en explorant l'espace libre.	145
6.20	Le robot mobile Hilare II.	146
6.21	Modélisation géométrique du robot mobile Hilare II.	147
6.22	Placement des télémètres ultrasoniques et laser sur le robot mobile Hilare II.	148
6.23	Spécification de la primitive MOVE-UNTIL-CONTACT pour le robot mobile.	149
6.24	Spécification de la primitive FOLLOW pour le robot mobile.	150
6.25	Seuil inférieur de la distance de sécurité $d_s$ pour garantir les rotations du robot mobile $\mathcal{A}$ près des obstacles.	152
6.26	Définition de l'obstacle virtuel $\mathcal{V}_i$ autour d'un obstacle réel $\mathcal{B}_i$ .	152



6.27	Décomposition des déplacements de translation/rotation simultanées. . .	153
6.28	Mouvement d'alignement effectué avec un robot mobile réel. . . . .	154
6.29	Problème 1 de planification à résoudre pour le robot mobile Hilare II. .	156
6.30	Résolution du problème 1 de planification posé. . . . .	157
6.31	Plan exécutable pour le robot mobile Hilare II résolvant le problème 1.	157
6.32	Problème 2 de planification à résoudre pour le robot mobile Hilare II. .	159
6.33	Résolution de la sous-tâche initiale du problème 2 de planification posé.	160
6.34	Plan exécutable pour le robot mobile Hilare II résolvant la sous-tâche initiale du problème 2. . . . .	161
6.35	Résolution de la sous-tâche finale du problème 2 de planification posé. .	162
6.36	Plan exécutable pour le robot mobile Hilare II résolvant la sous-tâche finale du problème 2. . . . .	163
6.37	Séquence d'images du robot Hilare II exécutant les plans générés pour résoudre le problème 2 posé. . . . .	165
6.38	Problème 3 de planification à résoudre pour le robot mobile Hilare II. .	166
6.39	Résolution de la sous-tâche initiale du problème 3 de planification posé.	167
6.40	Résolution de la sous-tâche initiale du problème 3 de planification posé (suite). . . . .	167
6.41	Plan exécutable pour le robot mobile Hilare II résolvant la sous-tâche initiale du problème 3. . . . .	168
6.42	Résolution de la sous-tâche finale du problème 3 de planification posé. .	171
6.43	Résolution de la sous-tâche finale du problème 3 de planification posé (suite). . . . .	172
6.44	Plan exécutable pour le robot mobile Hilare II résolvant la sous-tâche finale du problème 3. . . . .	172

---

# Bibliographie

---

- [AF88] N. Ayache and O. Faugeras. Maintaining representations of the environment of a mobile robot. Technical Report 789ΓINRIAΓFebruary 1988.
- [Ahu94] J.M. Ahuactzin. *Le Fil d'Ariane: Une Méthode de Planification Générale. Application à la Planification Automatique de Trajectoires*. PhD thesisΓInstitut National Polytechnique de GrenobleΓGrenobleΓFranceΓseptembre 1994.
- [AS94] R. Alami and T. Siméon. Planning robust motion strategies for a mobile robot. In *IEEE International Conference on Robotics and Automation*Γvolume 2Γpages 1312–1318ΓSan DiegoΓCaliforniaΓMay 1994.
- [BAS95] B. BouillyΓR. AlamiΓand T. Siméon. A numerical technique for planning motion strategies for a mobile robot in presence of uncertainties. In *IEEE International Conference on Robotics and Automation*Γvolume 2Γpages 1327–1332ΓNagoyaΓJapanΓMay 1995.
- [BF95] J. Barraquand and P. Ferbach. Motion planning with uncertainty: The information space approach. In *IEEE International Conference on Robotics and Automation*Γpages 1341–1348ΓNagoyaΓJapanΓMay 1995.
- [BFLV92] G. BauzilΓR. Ferraz de CamargoΓC. LemaireΓand G. Vialaret. Robot mobile hilare II : Matériel et bibliothèques associées. Technical Report 92265ΓLaboratoire d'Automatique et d'Analyse des SystèmesΓToulouseΓFranceΓJuly 1992.
- [BL89] J. Barraquand and J.C. Latombe. Robot motion planning: a distributed representation approach. Technical Report STAN-CS-89-1257ΓRobotics LaboratoryΓComputer Science DepartmentΓStanford UniversityΓMay 1989.
- [Bro82] R.A. Brooks. Symbolic error analysis and robot planning. *International Journal of Robotics Research*Γ1(4):29–68Γ1982.
- [Bro91] R.C. Brost. *Analysis and Planning of Planar Manipulation Tasks*. PhD thesisΓCarnegie Mellon UniversityΓPittsburghΓUSAΓJanuary 1991.

- [CM93] L. Cordewener and D. Meizel. Adaptation en ligne de la vitesse d'exécution de tâches en robotique mobile. *Revue d'Intelligence Artificielle RIAI* 7(4):465–480Γ1993.
- [Cor95] L. Cordewener. *Contrôle d'Exécution de Missions de Robots Mobiles à partir de Tâches Robot*. PhD thesisΓUniversité de Technologie de CompiègneΓCompiègneΓFranceΓjanvier 1995.
- [CR87] J.F. Canny and J. Reif. New lower bound techniques for robot motion planning problems. In *IEEE Symposium on Foundations of Computer Science*Γpages 49–60ΓLos AngelesΓ1987.
- [De 92] F. De la Rosa. Planification de mouvements fins en robotique d'assemblage par fonctions potentiels. DEAΓInstitut National Polytechnique de GrenobleΓFranceΓjuin 1992.
- [DLN93] F. De la RosaΓC. LaugierΓand J. Najera. Exploring the contact space to plan robot motions under geometric uncertainty constraints. In *International Workshop on Intelligent Robotic Systems*Γpages 71–80ΓZakopaneΓPolandΓJuly 1993.
- [DLN94] F. De la RosaΓC. LaugierΓand J. Najera. Planning motion strategies in the contact space under geometric uncertainty constraints. *Revue d'Intelligence Artificielle RIAI* 8(3):279–312Γ1994.
- [DLN96] F. De la RosaΓC. LaugierΓand J. Najera. Robust path planning in the plane. *IEEE Transactions on Robotics and Automation* 12(2):347–352ΓApril 1996.
- [DNL94] F. De la RosaΓJ. NajeraΓand C. Laugier. Planning robot motion strategies under geometric uncertainty constraints. In *International Symposium on Intelligent Robotic Systems*Γpages 37–44ΓGrenobleΓFranceΓJuly 1994.
- [DNL95] F. De la RosaΓJ. NajeraΓand C. Laugier. Dealing with uncertainty constraints in motion planning. In *Intelligent Autonomous Systems IAS-4*Γpages 497–504ΓKarlsruheΓGermanyΓMarch 1995.
- [Don86] B.R. Donald. Robot motion planning with uncertainty in the geometric models of the robot and environment: A formal framework for error detection and recovery. In *IEEE International Conference on Robotics and Automation*Γpages –ΓSan FranciscoΓUSAΓApril 1986.
- [Don88a] B.R. Donald. The complexity of planar compliant motion planning under uncertainty. In *ACM Fourth Symposium on Computational Geometry*Γpages 309–318ΓJune 1988.

- [Don88b] B.R. Donald. A geometric approach to error detection and recovery for robot motion planning under uncertainty. *Artificial Intelligence Journal* 37(1-3):223–271 1988.
- [Erd86] M. Erdmann. Using backprojections for fine motion planning with uncertainty. *International Journal of Robotics Research* 5(1):19–45 1986.
- [Fer91] R. Ferraz de Camargo. *Architecture Matérielle et Logicielle pour le Contrôle d'Exécution d'un Robot Mobile Autonome*. PhD thesis 1991, Université Paul Sabatier, Toulouse, France, juillet 1991.
- [GMR92] K.Y. Goldberg, M.T. Mason, and A. Requicha. Geometric uncertainty in motion planning: Summary report and bibliography. Technical Report IRIS-TR-297, University of Southern California, Institute for Robotics and Intelligent Systems, Los Angeles, California, August 1992.
- [GS92] A. Giraud and D. Sidobre. A heuristic motion planner using contact for assembly. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2165–2170, Nice, France, May 1992.
- [HU79] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.
- [Kha86] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research* 5(1):90–98 1986.
- [Lat91] J.C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.
- [Lau87] C. Laugier. *Raisonnement Géométrique et Méthodes de Décision en Robotique. Application à la Programmation Automatique des Robots*. PhD thesis, Institut National Polytechnique de Grenoble, Grenoble, France, décembre 1987.
- [Lau89] C. Laugier. Planning fine motion strategies by reasoning in the contact space. In *IEEE International Conference on Robotics and Automation*, pages 653–659, Scottsdale, Arizona, 1989.
- [LL92] A. Lazanas and J.C. Latombe. Landmark-based robot navigation. Technical Report STAN-CS-92-1428, Stanford University, Department of Computer Science, Stanford, California, May 1992.
- [LLS91] J.C. Latombe, A. Lazanas, and S. Shekhar. Robot motion planning with uncertainty in control and sensing. *Artificial Intelligence* 52:1–47 1991.
- [LMT84] T. Lozano-Pérez, M.T. Mason, and R.H. Taylor. Automatic synthesis of fine motion strategies for robots. *International Journal of Robotics Research* 3(1):3–24 1984.

- [Loz83] T. Lozano-Pérez. Spatial planning: A configuration space approach. *The IEEE Transactions on Computers* 32(2):108–120 1983.
- [Mas81] M.T. Mason. Compliance and force control for computer controlled manipulators. *The IEEE Transactions Systems, Manipulators and Cybernetics* 11(6):418–432 1981.
- [Maz90] I. Mazon. *Raisonnement sur les contraintes géométriques et d'incertitudes pour la planification de tâches de manipulation robotisées*. PhD thesis 1 Université Paul Sabatier 1 Toulouse 1 France 1 mai 1990.
- [Mer96] R. Mermond. Planification de chemins pour un robot non-holonome sous des contraintes d'incertitudes géométriques. DEA 1 Institut National Polytechnique de Grenoble 1 France 1 juin 1996.
- [MK92] R.S. Mattikalli and P.K. Khosla. Motions constraints from contact geometry: Representation and analysis. In *IEEE International Conference on Robotics and Automation* 1 volume 3 1 pages 2178–2185 1 Nice 1 France 1 May 1992.
- [Naj95] J.A. Najera. *Planification et Exécution de Stratégies d'Assemblage en Robotique*. PhD thesis 1 Institut National Polytechnique de Grenoble 1 Grenoble 1 France 1 octobre 1995.
- [Nat86] B.K. Natarajan. The complexity of fine motion planning. Technical Report TR-86-734 1 Cornell University 1 Department of Computer Science 1 Ithaca 1 New York 1 1986.
- [Nat88] B.K. Natarajan. The complexity of fine motion planning. *International Journal of Robotics Research* 7(2):36–42 1988.
- [NDL94] J. Najera 1 F. De la Rosa 1 and C. Laugier. Planning robot motion strategies under geometric uncertainty constraints. In *International Conference on Intelligent Robots and Systems* 1 pages 462–469 1 Munich 1 Germany 1 September 1994.
- [PS95] L.A. Page and A.C. Sanderson. Robot motion planning for sensor-based control with uncertainties. In *IEEE International Conference on Robotics and Automation* 1 pages 1333–1340 1 Nagoya 1 Japan 1 May 1995.
- [Pug89] P. Puget. *Vérification-Correction de Programme pour la Prise en Compte des Incertitudes en Programmation Automatique des Robots*. PhD thesis 1 Institut National Polytechnique de Grenoble 1 Grenoble 1 France 1 février 1989.
- [Ram88] F. Ramparany. *Un Cadre pour la Perception Multisensorielle en Robotique*. PhD thesis 1 Institut National Polytechnique de Grenoble 1 Grenoble 1 France 1 janvier 1988.

- 
- [SA94] T. Siméon and R. Alami. Planning robust motion strategies for a mobile robot in a polygonal world. *Revue d'Intelligence Artificielle RIA* 8(4):383–401 1994.
- [Sid91] D. Sidobre. *Raisonnement Géométrique et Synthèse de Stratégies d'Assemblage en Robotique*. PhD thesis 1 Université Paul Sabatier 1 Toulouse 1 France 1 mars 1991.
- [SL91] S. Shekhar and J.C. Latombe. On goal recognizability in motion planning with uncertainty. In *IEEE International Conference on Robotics and Automation* 1 pages 1728–1733 1 Sacramento 1 California 1 April 1991.
- [Tay76] R.H. Taylor. Synthesis of manipulator control programs from task-level specifications. Technical Report AIM 228 1 Stanford University 1 Department of Computer Science 1 Stanford 1 California 1 July 1976.
- [WG75] P.M. Will and D.D. Grossman. Experimental system for computer controlled mechanical assembly. *The IEEE Transactions on Computers* 24(9):879–888 1975.

## PLANIFICATION DE MOUVEMENTS AVEC PRISE EN COMPTE EXPLICITE DES INCERTITUDES GEOMETRIQUES.

**Résumé :** Une nouvelle approche de planification automatique de mouvements est proposée pour un robot holonome (ayant 2 degrés de liberté en translation et 1 degré de liberté en rotation) sous des contraintes d'incertitudes géométriques. Les incertitudes considérées portent sur le positionnement (position/orientation) et sur le déplacement du robot. L'originalité de l'approche réside d'une part dans le principe de résolution, et, d'autre part, dans la modélisation des incertitudes et son intégration dans l'approche. L'approche est composée de deux fonctions définies dans l'espace des configurations : (1) une fonction de potentiel fictif qui engendre des déplacements incrémentaux aussi bien dans l'espace libre que dans l'espace de contact; (2) une fonction d'exploration qui parcourt de manière discrète l'espace de contact afin de contourner les minima locaux de la fonction précédente. Les "contacts" entre le robot et les obstacles sont utilisés pour réduire l'incertitude sur la position et l'orientation du robot. Une analyse explicite des incertitudes est intégrée dans l'approche proposée en utilisant des modèles géométriques de nature ensembliste. L'approche de planification a été validée pour deux types de tâches robotiques : (1) la planification de mouvements d'un objet "volant" sur un plan, et (2) la navigation d'un robot mobile.

**Mots clés :** planification de mouvements, incertitude géométrique, mouvements fins, mouvements en contact, robotique mobile, robotique d'assemblage.

## MOTION PLANNING UNDER GEOMETRIC UNCERTAINTY CONSTRAINTS.

**Abstract :** A new motion planning approach is proposed for a holonomic robot (with 2 degrees of freedom in translation and 1 in rotation) under geometric uncertainty constraints. The constraints take into account the position, orientation and motion of the robot. The original contribution of the approach is the resolution method and the modelling of uncertainties. The approach consists of two functions defined in the configuration space : (1) an artificial potential function used to generate incremental motions in both the free space and the contact space; (2) an exploration function that analyses the contact space in order to overcome the local minima of the potential function. "Contact" between the robot and the obstacles is used to reduce the position and orientation uncertainty of the robot. An explicit uncertainty analysis using geometric models is integrated into the approach. The approach is validated with two robotic tasks : (1) the motion planning for a free flying object on a plane, and (2) the navigation of a mobile robot.

**Keywords :** motion planning, geometric uncertainty, fine motions, compliant motions, mobile robotics, assembly motion planning.