



HAL
open science

Manipulateurs paralleles, 5eme partie: determination de l'espace de travail a orientation constante

Jean-Pierre Merlet

► **To cite this version:**

Jean-Pierre Merlet. Manipulateurs paralleles, 5eme partie: determination de l'espace de travail a orientation constante. [Rapport de recherche] RR-1645, INRIA. 1992, pp.79. inria-00077146

HAL Id: inria-00077146

<https://inria.hal.science/inria-00077146>

Submitted on 29 May 2006

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

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

**Manipulateurs parallèles, 5eme partie : Détermination
de l'espace de travail à orientation constante**

**Parallel manipulators, 5th Part: Determination of the
workspace for a constant orientation**

Jean-Pierre MERLET

Résumé

Une phase importante de la conception d'un robot parallèle est la détermination de son espace de travail. Pour un manipulateur parallèle à 6 degrés de liberté celui-ci est borné en raison des limitations sur les variables articulaires (longueur des segments) mais aussi par les limitations sur le débattement des articulations ainsi que par les collisions entre segments. Cette phase est délicate car, contrairement au robot série, il n'est pas possible de découpler orientation et translation ce qui impose théoriquement une représentation en dimension 6. Usuellement on simplifie le problème en supposant que l'orientation de l'organe terminal est constante et en cherchant des coupes planes de l'espace de travail en fixant un degré de liberté en translation. Pour la détermination de ces coupes la méthode la plus répandue consiste à discrétiser les deux degrés de liberté restants. Une telle approche conduit à des temps de calcul importants et à des résultats importants en taille.

Nous proposons dans ce rapport une modélisation géométrique simple des débattements limites des articulations. A partir de ce modèle nous montrons qu'il est alors possible d'obtenir de manière exacte la géométrie de la frontière de l'espace de travail en prenant en compte les limitations sur les variables articulaires, les débattements des articulations ainsi que les problèmes d'intersection entre segments. Cette méthode est illustrée sur différents exemples.

Abstract

The workspace of a 6 d.o.f. parallel manipulator is bounded because of the limitations on the articular coordinates (links lengths) but also because of the mechanical limits on the joints and links interference. Usually the boundary of the workspace is determined by fixing four of the six d.o.f. to get planar cross-sections. Then a discretisation method is used which yield to high computation time and, sometimes, to errors as the workspace may be complex. We propose in this report a geometrical method to determine exactly the boundary of the workspace due to the limitations on the articular coordinates and mechanical limits on the articulations. Furthermore this method enables to determine in the resulting zone the sub-zones for which the distance between the links is always greater than a fixed limit. This method is illustrated on various examples.

Chapitre 1

Introduction

1.1 La structure mécanique d'un robot parallèle

Considérons un robot parallèle à 6 degrés de liberté tel que celui représenté sur la figure 1.1. Il est constitué d'une base supposée fixe et d'un plateau mobile,

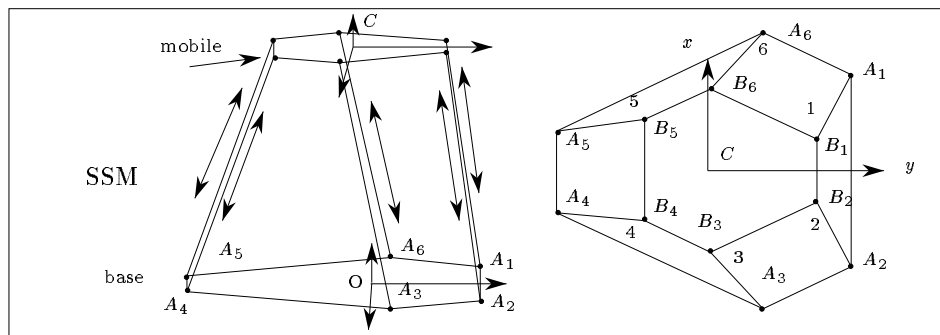


Figure 1.1: Un robot parallèle à 6 degrés de liberté.

reliés par 6 segments. Une extrémité de ces segments est reliée à la base par un joint de Cardan, l'autre extrémité étant reliée au plateau mobile par un rotule. Dans chacun des segments un actionneur linéaire permet de modifier la longueur du segment. Par un contrôle approprié de ces longueurs il est alors possible de placer un point du plateau mobile dans une position donnée puis de fixer l'orientation de ce plateau : on contrôle donc les 6 degrés de liberté du plateau mobile. On obtient ainsi un manipulateur dont l'architecture mécanique est très différente de celle des robots série.

Cette architecture est connue depuis longtemps et certains problèmes théoriques qu'elle pose ont été étudiés dès les années 1800 [3]. Leur réalisation pratique est cependant plus récente puisque le premier prototype semble pouvoir être attribué à Gough [19] en 1949.

Cette architecture a connue un regain d'intérêt dans les années 60 pour la réalisation de plate-forme pour les simulateurs de vol [19,2]. En effet une des caractéristiques intéressantes de ce mécanisme est sa capacité de manipulation de charges lourdes. En effet le poids de la charge est sensiblement réparti sur les segments et exerce principalement une contrainte de traction-compression. Ceci est à opposer au cas des robots série où les premiers actionneurs sont au contraire soumis à des couples importants et où les segments sont soumis principalement à des contraintes de flexion beaucoup plus nuisibles.

L'utilisation de cette structure en tant que robot est encore plus récente puisqu'elle remonte à 1979 [11], avec son apparition dans une cellule d'assemblage. Ici l'intérêt supplémentaire qui apparaît est la précision de positionnement. En effet les erreurs de mesure des capteurs permettant d'estimer les longueurs des segments n'entraînent pas, à l'opposé de ce qui se passe pour les robots série, une erreur importante sur la localisation de l'organe terminal.

De plus cette architecture est intrinsèquement très rigide (un projet de robot minier utilisant cette propriété est d'ailleurs en développement [1]) mais peut être adaptée pour avoir une certaine "souplesse" mécanique sans perte d'information sur la position de l'organe terminal. En plaçant des capteurs de force dans les segments on peut même mesurer les forces et moments s'exerçant sur le plateau mobile, c'est-à-dire disposer d'un manipulateur qui est en même temps un capteur de forces à 6 composantes. Ces caractéristiques en font donc un instrument qui a été souvent utilisé dans des cellules d'assemblage [1,4,15,17,13]. L'étude des robots parallèles est d'ailleurs en plein développement [5,7,8,9,12,18,20] et certains problèmes théoriques ont connu une avancée importante ces dernières années (en particulier les problèmes de cinématique, de dynamique et de configurations singulières). Toutes ces études devraient permettre à terme la construction d'un système de conception de robots parallèles.

Cependant une caractéristique pourtant importante n'a reçu, d'un point de vue théorique, qu'une faible attention jusqu'à maintenant : la détermination de l'espace de travail.

1.2 Complexité de représentation de l'espace de travail

Le calcul de l'espace de travail est évidemment très important dans une phase de conception. Le fait que les travaux dans ce domaine sont peu nombreux se justifie cependant par la complexité du problème. En effet, si l'on considère

un robot série classique la représentation de l'espace de travail est généralement basée sur l'illustration en dimension 3 de l'espace atteignable par le centre de son poignet (caractérisant les translations) et par celui atteignable par l'extrémité de l'organe terminal (ce qui caractérise les orientations), ces deux zones étant découplées. Ce n'est malheureusement pas le cas pour les robots parallèles: la zone atteignable par le centre du plateau mobile est dépendante de l'orientation de ce plateau. La dimension de l'espace dans lequel on doit faire la représentation est donc 6 ce qui est peu favorable à une représentation graphique.

Heureusement la majeure partie des applications de ce type de robot permet de simplifier ce problème : en effet le robot est le plus souvent utilisé soit avec une orientation constante, soit en maintenant fixe un des points du plateau mobile. On ne s'intéresse donc qu'à une restriction de l'espace total dans un espace de dimension 3. Usuellement on simplifie encore le problème en fixant un des degrés de liberté restants de façon à n'avoir à déterminer que des sections planes.

Nous nous intéressons ici au cas où l'on fixe l'orientation et un des paramètres de translation. On cherche donc à déterminer l'ensemble des positions atteignables par un point fixé du plateau mobile lorsqu'il se déplace dans un plan connu.

Les contraintes qui limitent l'évolution du robot sont de trois types:

- les limitations sur les coordonnées articulaires : les segments ont une longueur minimum et une longueur maximum.
- les butées mécaniques sur les articulations passives : joint de Cardan et rotule.
- les collisions entre segments.

D'un point de vue analytique les équations décrivant ces contraintes sont complexes, de type inégalités, et la détermination des bornes par manipulation de ces équations semi-algébriques semble difficile, même avec nos hypothèses simplificatrices.

1.3 Notation

Avant d'exposer une méthode pour le calcul de l'espace de travail nous allons introduire quelques notations. On définit tout d'abord deux repères, l'un fixe et lié à la base, appelé *référentiel absolu* et l'autre attaché au plateau mobile et appelé *référentiel mobile*.

Les notations suivantes seront employées dans ce rapport:

- \wedge : symbole du produit vectoriel.

6 Espace de travail

- A_i : centre de l'articulation du segment i , proche de la base.
- $x_{a_i}, y_{a_i}, z_{a_i}$: coordonnées de A_i dans le référentiel absolu.
- B_i : centre de l'articulation du segment i , proche de l'organe terminal.
- $x_{b_i}, y_{b_i}, z_{b_i}$: coordonnées de B_i dans le référentiel mobile.
- $x_{B_i}, y_{B_i}, z_{B_i}$: coordonnées de B_i dans le référentiel absolu.
- O : centre du référentiel absolu. En général ce point sera le barycentre des points d'articulations proches de la base.
- x, y, z : axes du référentiel absolu lié à la base. L'axe y sera l'axe de symétrie de la base si elle en possède un.
- C : centre du référentiel mobile, sert en général à repérer la position de l'organe terminal. En général on choisira le barycentre des points d'articulation liés au mobile.
- x_c, y_c, z_c : coordonnées de C dans le repère absolu.
- x_r, y_r, z_r : axes du référentiel lié au plateau mobile. L'axe y_r sera l'axe de symétrie du mobile s'il en possède un.
- ψ, θ, ϕ : angles d'Euler permettant de décrire l'orientation de l'organe terminal. Ces angles sont définis de la manière suivante : partant du repère de référence on tourne d'un angle ψ autour de l'axe z , puis d'un angle θ autour du nouvel axe x et enfin d'un angle ϕ autour du nouvel axe z pour obtenir le repère mobile.
- R : matrice de rotation permettant le passage du référentiel mobile au référentiel absolu.
- ρ_i : longueur du segment relié à l'organe terminal de la chaîne i .

Pour alléger la présentation on omettra de faire figurer les indices lorsque le contexte ne sera pas ambigu.

Chapitre 2

L'espace de travail limité par les variables articulaires

2.1 Utilisation des méthodes de discrétisation

La méthode la plus répandue pour déterminer la frontière de l'espace atteignable consiste à procéder par discrétisation. C'est cette méthode que nous avons décrite dans un précédent rapport [14]: nous nous étions intéressé au déplacement en translation dans le plan $x - y$ du centre de notre prototype en utilisant un système de coordonnées polaire pour ces variables. Le rayon polaire était progressivement augmenté et à chaque pas les longueurs des segments étaient calculées jusqu'à ce que l'on trouve l'ensemble des positions limites où au moins l'une des longueurs se trouve en dehors des bornes permises.

Cette méthode a aussi été employée par la suite en incorporant de plus une vérification des débattements des articulations [10] ou bien avec vérification de l'ensemble des contraintes [1]. Cette approche numérique est cependant lourde en temps de calcul et nécessite une implantation soigneuse car nous verrons que la coupe de l'espace de travail est définie non seulement par une enveloppe extérieure mais peut comporter aussi des "trous" qui peuvent échapper à la détection et même conduire à des erreurs, la frontière de ces trous étant confondue avec l'enveloppe extérieure. Si cette méthode est effectivement utilisable sur des manipulateurs de petite dimension elle ne paraît pas satisfaisante pour des mécanismes comme les simulateurs de vol. De plus une représentation graphique obtenue à partir de cette méthode est peu satisfaisante car elle ne fournit aucune information sur la géométrie de la frontière de l'espace de travail. Pour la même raison le stockage des résultats nécessite une place mémoire importante.

2.2 Méthode géométrique pour les contraintes articulaires

Une approche complètement différente a été proposée par C.Gosselin [6] lors de son séjour à l'INRIA. Dans cette approche on ne prend en compte que les limitations sur les longueurs des segments et l'on cherche à déterminer la frontière de l'espace de travail lorsque le point C du plateau mobile se déplace dans un plan connu (pour simplifier l'exposé nous supposons que ce plan est horizontal, sans perte de généralité). On va montrer tout d'abord comment on peut déduire des limitations articulaires sur chaque segment une description géométrique des limitations sur le domaine décrit par C .

2.2.1 Contraintes d'évolution sur C

Si l'on considère la géométrie du robot parallèle il apparaît que les contraintes articulaires imposent des contraintes sur la position des points B_i . En effet le point B de chacun des segments doit être compris entre les deux sphères de centre A et de rayon ρ_{min}, ρ_{max} respectivement longueur minimum et maximum du segment (figure 2.1). Ces sphères seront respectivement appelées *sphère in-*

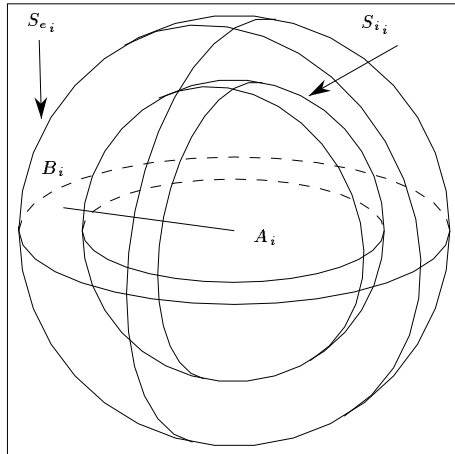


Figure 2.1: Les contraintes articulaires imposent au point B_i d'être dans la zone comprise entre deux sphères de centre A_i et de rayon égale au longueur minimum et maximum des segments.

terne (S_{i_i}) et *sphère externe* (S_{e_i}). Si l'on suppose que le point C évolue dans un plan il en sera de même pour les points B_i de chacun des segments, l'orientation étant supposée constante. La définition d'un plan d'évolution \mathcal{P} pour C impose donc un plan d'évolution \mathcal{P}_i pour chacun des points B_i .

Supposons maintenant que la longueur du segment soit maximum. Le point B_i doit alors appartenir à l'intersection de la sphère externe et du plan \mathcal{P}_i . Si cette intersection est vide le point B_i ne pourra pas évoluer dans le plan et par conséquent l'espace de travail est vide. Dans le cas contraire l'intersection est un cercle, le *cercle externe de B_i* noté $C_{e_{B_i}}$. Le point B_i doit alors se trouver à l'intérieur ou sur la frontière de ce cercle.

Si l'on suppose maintenant que la longueur du segment est minimum un raisonnement analogue nous permet d'affirmer que si l'intersection de la sphère interne et du plan \mathcal{P}_i est non vide alors le point B_i doit être à l'extérieur ou sur la frontière d'un cercle, le *cercle interne de B_i* noté $C_{i_{B_i}}$ (figure 2.2). Notons que les cercles interne et externe ont le même centre (la projection de A_i dans le plan) et que l'on peut facilement calculer leurs rayons en fonction des rayons des sphères et de la position de \mathcal{P} . Ainsi si l'on suppose que le plan \mathcal{P} est est

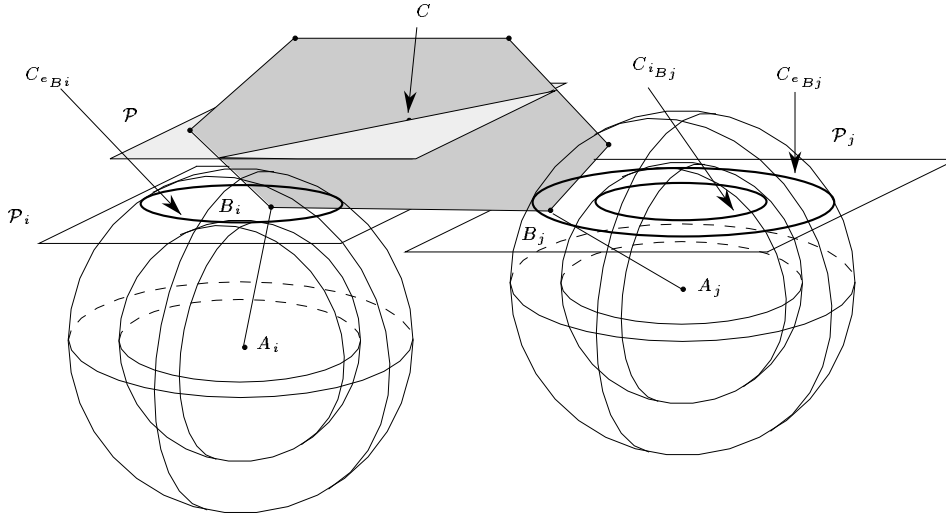


Figure 2.2: Les contraintes articulaires imposent que lorsque C se déplace dans un plan \mathcal{P} les points B_i soient à l'intérieur des cercles $C_{e_{B_i}}$ et à l'extérieur des cercles $C_{i_{B_i}}$ obtenus par intersection des sphères avec le plan \mathcal{P}_i .

un plan horizontal on peut écrire :

$$\mathbf{AB} = \mathbf{AO} + \mathbf{OB} \quad (2.1)$$

En calculant la norme des vecteurs précédents on obtient:

$$(x_b - x_a)^2 + (y_b - y_a)^2 = \rho^2 - (z_b - z_a)^2 \quad (2.2)$$

Le terme de droite de cette équation est complètement défini par la définition du plan d'évolution de C ainsi que l'orientation du plateau mobile. En conséquence

le point B évolue dans le plan \mathcal{P}_i en restant entre des cercles dont le centre a comme coordonnées (x_a, y_a) et comme rayon R_e , pour le cercle externe :

$$\begin{aligned} R_e^2 &= \rho_{max}^2 - (z_b - z_a)^2 & \text{si } |\rho_{max}| > z_b - z_a \\ R_e &= 0 & \text{sinon} \end{aligned}$$

Pour le cercle interne le rayon R_i est défini par:

$$\begin{aligned} R_i^2 &= \rho_{min}^2 - (z_b - z_a)^2 & \text{si } |\rho_{min}| > z_b - z_a \\ R_i &= 0 & \text{sinon} \end{aligned}$$

Notons maintenant que si C est sur la frontière de l'espace de travail due aux limitations sur les variables articulaires alors au moins un des B_i se trouve soit sur son cercle externe soit sur son cercle interne. Supposons maintenant que l'un des points B_i décrit l'un des cercles externe ou interne. Il est alors clair que le point C décrit un cercle qui sera de même rayon que le cercle décrit par B_i mais dont le centre s'obtient à partir de celui du cercle décrit par B_i par simple translation de vecteur $\mathbf{B}_i\mathbf{C}$. En conséquence la contrainte sur le point B_i se transpose pour le point C : les contraintes articulaires sur le segment i imposent au point C d'être compris entre deux cercles appelés *cercle externe* C_{e_i} et *cercle interne* C_{i_i} .

Chaque segment impose donc des zones possibles d'évolution de C et l'espace de travail est l'intersection de ces zones, donc l'intersection des 6 couronnes annulaires définies par les cercles internes et externes. On peut donc en déduire que l'espace de travail dans le plan sera délimité par des arcs de cercle. La figure 2.3 montre deux cas possibles (nous avons limité le nombre de couronnes à 3 pour simplifier le dessin et en faciliter la compréhension).

2.2.2 Algorithme d'intersection des zones permises pour C

Pour trouver l'intersection des zones annulaires et, par conséquent, les arcs de cercle, le principe proposé par Gosselin est le suivant:

- on calcule tout d'abord toutes les intersections entre tous les couples de cercle. On peut trouver 0, 1, ou 2 points d'intersection.
- Pour un cercle donné on ordonne les points d'intersection, ce qui conduit à une liste ordonnée d'arcs de cercle sur le cercle considéré. Si l'on n'a pas trouvé de point d'intersection la liste est réduite au cercle lui-même.
- Pour tous les arcs de la liste d'un cercle donné on teste si l'arc est à l'intérieur de tous les cercles extérieurs et à l'extérieur de tous les cercles intérieurs des 5 autres couronnes annulaires. Pour cela il suffit de prendre un point sur l'arc (usuellement le milieu) et de vérifier la condition ci-dessus. Il est facile de montrer que si ce point satisfait la contrainte alors

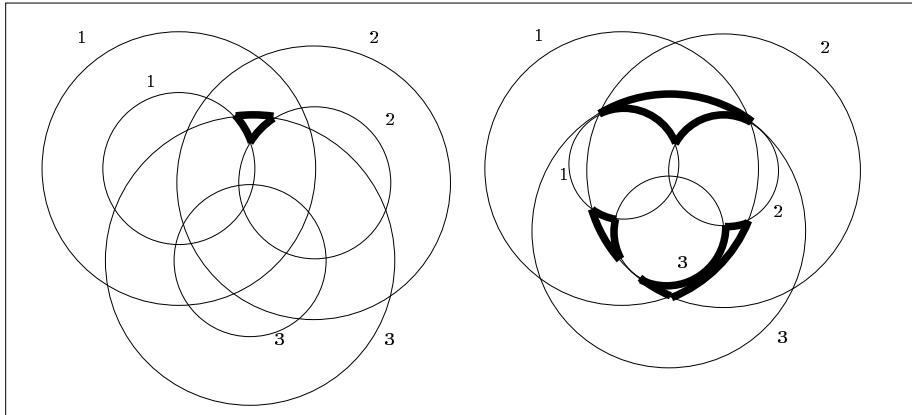


Figure 2.3: En traits épais la frontière de l'espace de travail dans un plan, délimitée par l'intersection de 3 couronnes annulaires définies par les cercles internes et externes.

tout l'arc la satisfait aussi. Si l'arc vérifie la condition alors il fait partie de la frontière de l'espace de travail et on le place dans la liste résultat. S'il ne la vérifie pas alors il est rejeté et on passe à l'arc suivant dans la liste.

Un exemple de traitement est montré en figure 2.4 Remarquons que cet al-

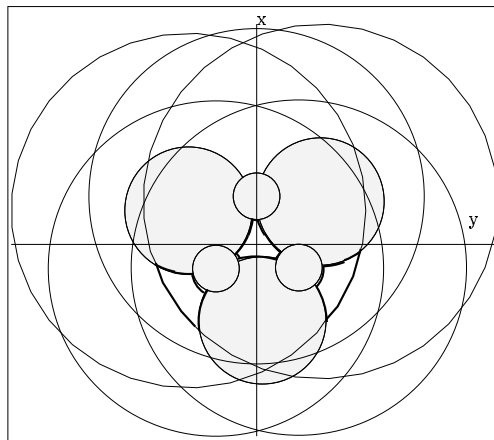


Figure 2.4: Un exemple de calcul d'intersection. En grisé les cercles intérieurs, en traits épais la frontière de l'espace de travail. On peut remarquer que l'on obtient 3 zones distinctes.

gorithme ne fournit pas une liste ordonnée des arcs de cercles. Il serait donc nécessaire de procéder à une étape supplémentaire pour obtenir la frontière de l'espace de travail. Par contre un algorithme simple permet de calculer la surface de l'espace de travail à partir de la liste résultat.

La méthode exposée présente de gros avantages par rapport à la discrétisation en particulier une grande rapidité, la possibilité de calculer rapidement la surface de l'espace de travail ainsi qu'une grande facilité de stockage des résultats (une liste réduite d'arcs de cercle). Toutefois elle ne prend en compte qu'un des trois facteurs limitant l'espace de travail, les contraintes sur les variables articulaires.

Chapitre 3

L'espace de travail limité par les articulations

Dans ce chapitre nous allons proposer une méthode permettant la prise en compte des limites mécaniques sur les articulations.

3.1 Limitations mécaniques sur les articulations de base

Ayant affaire au monde réel il est clair que la nature de ces limitations mécaniques peut être quelconque. Notre but est donc de trouver une modélisation ouverte de ces contraintes, pour pouvoir considérer le plus grand nombre possible d'articulations réelles, mais aussi géométrique pour pouvoir trouver la topologie de la frontière de l'espace de travail.

Notons que nous avons à modéliser des contraintes sur des articulations de type joint de Cardan ou rotule. Si l'on suppose que chaque segment est soit une ligne ou un corps de révolution la rotation du segment autour de son axe n'intervient donc pas. Dans un premier temps nous allons examiner comment on peut modéliser les contraintes sur les articulations liées à la base.

3.1.1 Contraintes sur les articulations de la base

Au vue de la remarque précédente on peut considérer que les limites mécaniques peuvent être définies par une surface délimitant un volume à l'intérieur duquel doit se trouver le segment du manipulateur. L'articulation étant un joint de Cardan le segment ne peut que tourner autour du centre de l'articulation. Le volume a donc un point singulier, le centre de l'articulation. Si l'on suppose par

exemple que le volume est limité par des facettes planes toutes les facettes se réduisent à un point au point singulier.

3.1.1.1 Pyramide et cône

Dans notre système nous allons supposer que l'on peut modéliser la surface englobant le volume par un ensemble de facettes planes ou par un cône. Dans le premier cas on parlera de *pyramide* et l'opérateur aura à définir ces pyramides selon les contraintes de l'articulation utilisée. Pour cela on définit un repère $(A_i, \mathbf{x}_r, \mathbf{y}_r, \mathbf{z}_r)$ lié à la pyramide, de centre le centre A_i de l'articulation (figure 3.1). Chacune des n facettes est alors définie complètement par le vecteur normal \mathbf{n}_i à la facette. Ce vecteur est défini par les deux angles d'Euler ψ, θ et nous avons:

$$\mathbf{n}_i = \begin{pmatrix} \sin(\psi) \sin(\theta) \\ -\cos(\psi) \sin(\theta) \\ \cos(\theta) \end{pmatrix} \quad (3.1)$$

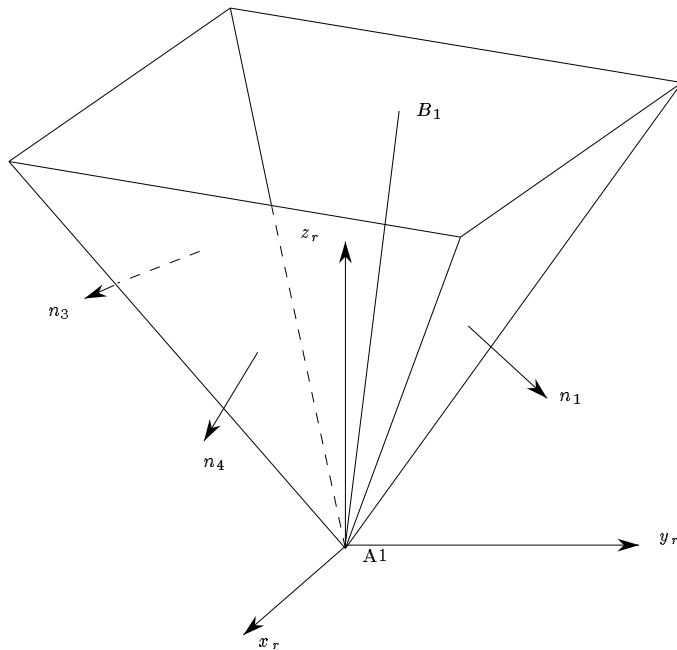


Figure 3.1: La définition d'une pyramide. Ici une pyramide à 4 facettes. Chaque facette i est complètement définie par le vecteur normal à la facette \mathbf{n}_i , lui même défini dans un repère $A_1, \mathbf{x}_r, \mathbf{y}_r, \mathbf{z}_r$. Pour définir complètement la pyramide il suffit alors de définir l'orientation du repère de la pyramide par rapport au référentiel absolu par 3 angles d'Euler. Le segment $A_1 B_1$ doit alors être contenu à l'intérieur de la pyramide.

3.1.2 Conséquence pour la position des points B_i

L'introduction des pyramides et des cônes comme modèle pour les contraintes sur les articulations de la base va nous permettre de trouver facilement la frontière de la zone permise pour les points B_i , du point de vue des articulations, qui est l'intersection de notre modèle de contrainte avec le plan \mathcal{P}_i d'évolution du point B_i . La nature de cette intersection est donc :

- un polygone si le modèle est une pyramide
- un cercle, une ellipse ou une hyperbole si le modèle est un cône.

3.2 Prise en compte des contraintes articulaires et des contraintes sur les articulations de base

3.2.1 Contraintes sur les points B_i

La zone permise pour les points B_i est clairement l'intersection des zones permises dues aux contraintes articulaires et celles dues aux contraintes sur les articulations de base. Nous supposons que le modèle utilisé pour les articulations de la base est du type pyramide. En conséquence la frontière de la zone permise pour B_i , lorsque les deux contraintes sont prises en compte, va être l'intersection des cercles internes et externes pour B_i et du polygone. Nous verrons par la suite comment calculer cette intersection. Mais déjà il est clair que cette intersection va conduire à une ou des régions dont les frontières seront soit un polygone, soit un cercle, soit une entité géométrique composée d'une succession de segments et d'arcs de cercle. Nous regrouperons sous le nom de *polygonarc* l'ensemble de ces possibilités.

Nous appellerons *zone permise* le résultat du calcul de l'intersection. Une zone permise est constituée d'une ou plusieurs *zones* dont les frontières sont des *polygonarcs*. Chacune de ces zones peut aussi avoir une frontière interne. Dans le cas d'intersections des zones permises pour B_i cette zone interne, si elle existe, est unique et constituée d'un cercle. La figure 3.2 montre en 3D les contraintes imposées par les conditions sur les longueurs et celles sur les articulations de base et la figure 3.3 présente un exemple typique de contraintes pour les B_i .

3.2.2 Zone permise pour C

De chaque zone permise pour B_i nous avons déjà vu que l'on déduisait par une simple translation de vecteur $\mathbf{B}_i\mathbf{C}$ une zone permise pour C , ZP_{C_i} , qui est une zone permise où les contraintes sur le segment i sont satisfaites. La zone permise pour C où toutes les contraintes sur les segments sont satisfaites est alors l'intersection des zones permises ZP_{C_i} . Comme dans le cas précédent on

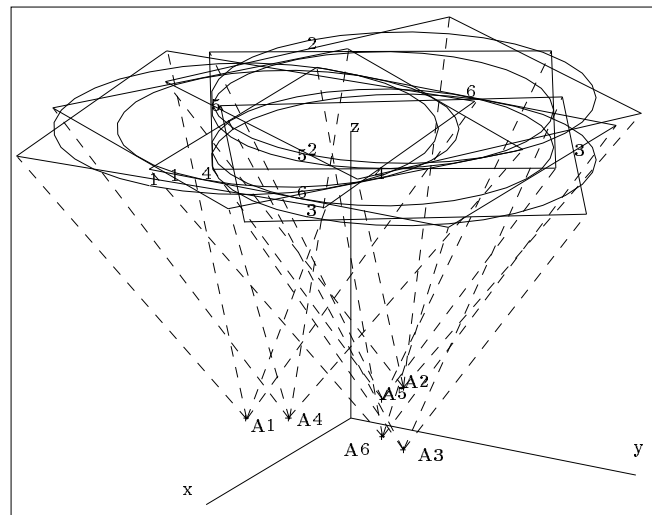


Figure 3.2: Vue 3D de l'ensemble des zones permises pour les points B_i . Les traits en pointillé délimitent les facettes des pyramides pour les articulations de base.

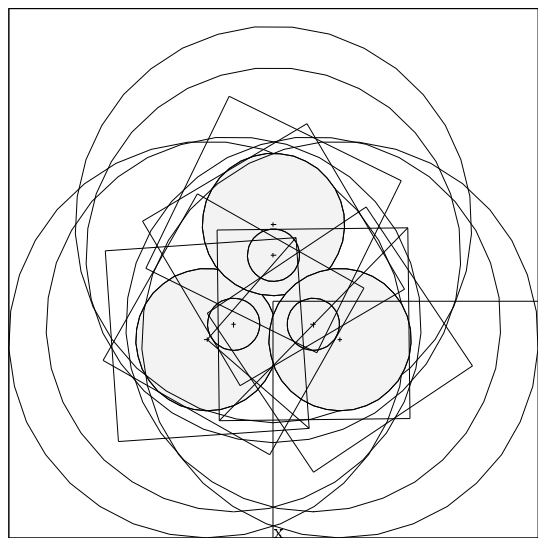


Figure 3.3: Exemples de zones permises pour les B . Les cercles correspondent aux cercles externes, les cercles grisés aux cercles internes et les carrés aux zones dues aux contraintes sur les articulations de base.

retrouve donc une intersection mais dans un cadre plus général puisque l'on doit procéder à l'intersection de 6 zones permises dont les zones ont une frontière externe constituée d'un polygonarc (et non plus simplement d'un cercle ou d'un polygone) avec éventuellement des polygonarcs internes. On doit donc rechercher un algorithme général d'intersection de zones permises dont les zones ont une frontière externe constituée d'un polygonarc avec une ou plusieurs frontières internes elles aussi constituées de polygonarcs.

3.2.3 Algorithme d'intersection des zones permises

Le problème du calcul efficace de l'intersection de polygones est un problème classique de la Géométrie Algorithmique, pour lequel de nombreuses méthodes ont été proposées [16]. Le problème que nous avons à traiter est cependant un peu particulier dans le sens où nos frontières sont des polygonarcs (parfois appelés aussi des *polygones généralisés*) avec de plus des frontières internes. Le nombre limité de zones permises à traiter va cependant nous permettre l'emploi de techniques "brutales" en terme de complexité. Nous espérons ici que la simplicité de la méthode permettra globalement d'obtenir une efficacité au moins égale à celle de méthodes plus sophistiquées, certe de complexité théorique moindre, mais dont les constantes ne les rendent en fait plus efficaces que pour un nombre d'objets plus important que celui de notre problème.

Notre algorithme d'intersection est basé sur l'utilisation de procédures multiples que nous allons maintenant exposer. La première de ces procédures est celle qui permet de déterminer l'appartenance d'un point à l'intérieur d'une zone permise.

3.2.3.1 Procédure `est-dans-zone-permise`

La procédure `est-dans-zone-permise(X, ZP)` est une procédure booléenne qui retourne `VRAI` si le point X est à l'intérieur de la zone permise ZP . Pour ce faire nous utilisons tout d'abord une procédure `est-sur-frontiere(X, ZP)` qui retourne `VRAI` si le point X est sur une frontière d'une des zones de la zone permise ZP . Si pour le point X cette procédure retourne `VRAI` alors la procédure `est-dans-zone-permise` va retourner `VRAI`. Dans le cas contraire nous construisons une ligne L issue de X de vecteur directeur quelconque \mathbf{u} avec la procédure `construit-ligne(X)` et nous comptons le nombre d'intersection de la ligne L avec les frontières des zones de la zone permise en utilisant la procédure `nombre-intersection(L, ZP)`. Si ce nombre est impair alors X est à l'intérieur de la zone permise. Une précaution est à prendre toutefois en évitant de compter deux fois un point d'intersection s'il est confondu avec un des sommets d'une des frontières des zones. Pour éviter ce problème on peut soit marquer les sommets lorsqu'ils sont rencontrés pour la première fois et ne pas compter les sommets marqués ou bien modifier le vecteur \mathbf{u} pour éviter que la

ligne L passe par un sommet. La procédure `est-dans-zone-permise` écrite en pseudo-code est décrite en Procédure 1.

```

procedure est-dans-zone-permise( $X, ZP$ )
Zone Permise  $ZP$ ; Point  $X$ ;
debut
  Ligne  $L$ ; Entier  $n$ ;
  si (est-sur-frontiere( $X, ZP$ )=VRAI) retourne VRAI;
   $L$ =construit-ligne( $X$ );
   $n$ =nombre-intersection( $L, ZP$ );
  si  $n$  pair retourne FAUX;
  sinon retourne VRAI;
fin

```

Procédure 1: `est-dans-zone-permise`

3.2.3.2 Procédure trouve-point-marquant

Le calcul de l'intersection des zones permises va nous conduire à une zone permise. Les frontières de cette zone permise peuvent être de deux natures différentes :

- un polygonarc d'une des zones permises fournies à l'entrée. Ce sera le cas si une frontière externe d'une des zones permises est complètement à l'intérieur de toutes les autres zones permises ou, pour une frontière interne, si elle contient toutes les frontières internes des autres zones permises.
- un polygonarc dont les sommets sont soit des sommets d'un polygonarc d'une des zones permises d'entrée ou l'intersection de deux arêtes de deux polygonarcs.

Nous appellerons *point marquant* tout point qui est soit un sommet d'un des polygonarcs d'entrée, soit l'intersection de deux polygonarcs, et qui appartient à l'ensemble des zones permises d'entrée. Il est alors clair que tout point marquant est un sommet de la frontière de la zone permise résultat du calcul de l'intersection. La procédure `trouve-point-marquant` a pour objectif la détermination de ces points marquants: elle stocke dans une structure particulière les coordonnées du point marquant ainsi que les deux arêtes qui ont conduit à ce point. Une arête est identifiée par le numéro de la zone permise, le numéro de la zone, le numéro de l'arête du polygonarc sur lequel se trouve ce point ainsi qu'un entier qui est à zéro si le polygonarc est une frontière externe ou qui donne le numéro de la frontière interne si ce polygonarc est une frontière interne. Ainsi (x,y) , $(1,2,2,0)$, $(1,2,3,0)$ va indiquer que le point marquant de coordonnées (x,y) est le sommet de la zone 2 de la zone permise 1 compris entre

les arêtes 2 et 3 de la frontière externe de cette zone permise. Par contre (x,y) , $(1,2,5,0)$, $(2,1,3,4)$ indique que le point marquant est le résultat de l'intersection de l'arête 5 de la frontière externe de la zone 2 de la zone permise 1 et de l'arête 3 de la frontière interne 4 de la zone 1 de la zone permise 2. Nous appellerons cette structure de donnée une structure **POINT-MARQUANT**. La procédure **trouve-point-marquant** va retourner le nombre de point marquant trouvé. Cette procédure utilise les procédures **sommet-zone-permise** qui renvoie tous les sommets d'une zone permise dans une structure de type **POINT-MARQUANT** et **intersection-zone-permise** qui renvoie tous les points intersection de deux zones permises dans une structure de type **POINT-MARQUANT**. On fera de plus appel à la procédure **coordonnee** qui renvoie les coordonnées d'un point marquant et **est-dans-toute-zone-permise** qui teste si un point appartient à un ensemble de zones permises. La procédure **trouve-point-marquant** écrite en pseudo-code est décrite en Procédure 2.

3.2.3.3 Procédure ordonne-point-marquant

Supposons dans un premier temps que nous ayons effectivement trouvé des points marquants. Ils définissent les sommets de la zone résultat que l'on doit maintenant chaîner, c'est-à-dire déterminer leur ordre et trouver quel type d'arête (circulaire ou droite) relie deux points marquants voisins. C'est le but de la procédure **ordonne-point-marquant** qui prend en entrée la liste de points marquants trouvée à l'aide de la procédure **trouve-point-marquant** et donne en sortie une zone permise. Pour ce faire on va utiliser une procédure auxiliaire **successeur-sur-arete** qui détermine le successeur dans la liste des points marquants d'un point marquant donné T_i sur une arête donnée ainsi que l'arête (dite *arête de liaison*) qui va joindre ces deux points dans la zone permise résultat.

Deux cas vont être distingués selon la nature de l'arête. Si l'arête est une droite, non verticale, on va chercher dans la liste, si il existe, le point marquant T_l le plus proche de T_i , dont l'abscisse est inférieure à celle de T_i est telle que le milieu du segment $T_i T_l$ appartient à l'ensemble des zones. Le point T_l sera appelé *point à gauche* de T_i . On cherchera de même un *point à droite* T_r répondant aux mêmes contraintes mais dont l'abscisse est supérieure à celle de T_i . Pour une arête verticale on considérera non pas les abscisses mais les ordonnées. En général pour une arête droite il n'existera qu'un point à droite ou un point à gauche qui sera le successeur de T_i et qui créera pour la zone résultat une arête liant T_i à ce point par un segment de droite. Certaines exceptions sont toutefois possibles auquel cas on prendra le point le plus proche de T_i (figure 3.4).

Pour une arête circulaire la procédure est identique mais la notion de point à gauche ou à droite est obtenue lorsque l'on passe du point T_i à son successeur en tournant dans le sens trigonométrique ou non (il y a donc potentiellement

```

procedure trouve-point-marquant(pm, ZP)
structure POINT-MARQUANT pm[];
Zone Permise ZP[6];
debut
Entier nb, i, j, k;
structure POINT-MARQUANT pmn[ ];
  nb=0;
  pour i =1,6
    debut
      n=sommet-zone-permise(ZP[i],pmn);
      for j=1,n
        debut
          X = coordonnee(pmn[j]);
          si (est-dans-toute-zone-permise(X, ZP)= VRAI) alors
            debut
              pm[nb]= pmn[j];
              nb=nb+1;
            fin
          fin
        fin
      fin
    pour i =1,5
      debut
        pour j =i+1,6
          debut
            n=intersection-zone-permise(ZP[i],ZP[j],pmn);
            for k=1,n
              debut
                X = coordonnee(pmn[k]);
                si (est-dans-toute-zone-permise(X, ZP)= VRAI) alors
                  debut
                    pm[nb]= pmn[k];
                    nb=nb+1;
                  fin
                fin
              fin
            fin
          fin
        fin
      fin
    retourne (nb);
  fin

```

Procedure 2: trouve-point-marquant

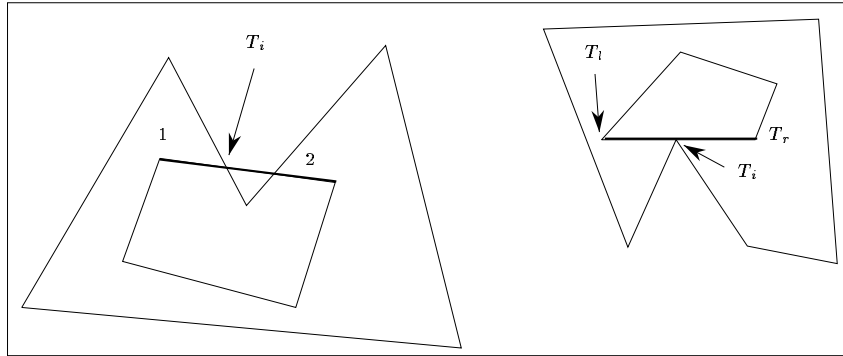


Figure 3.4: Successeur d'un point marquant T_i sur une arête droite (en trait épais). A gauche une exception où il existe simultanément un point à gauche et un point à droite.

toujours un point à droite et un à gauche) et l'on ne teste l'appartenance du milieu de l'arc qu'après avoir déterminé les points à gauche et à droite. En conséquence les points à droite et à gauche peuvent avoir les mêmes coordonnées mais l'arc utilisé pour relier le point final au point T_i différera selon que le point sera à gauche ou à droite (figure 3.5). La procédure **successeur-sur-arete**

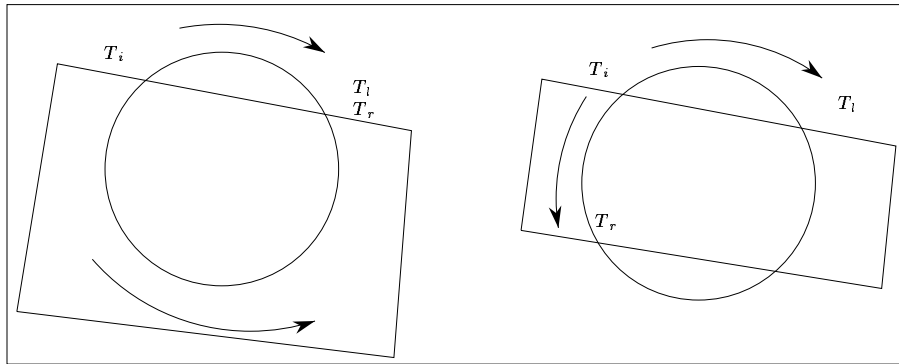


Figure 3.5: Successeur d'un point marquant T_i sur une arête circulaire (en trait épais). A gauche les points à gauche et à droite sont identiques mais le point à gauche sera rejeté car le milieu de l'arc n'appartient pas aux zones. A droite il y a deux points à gauche et à droite distincts mais le point à gauche sera rejeté

écrite en pseudo-code est décrite en Procédure 3. A partir de cette procédure il va être maintenant possible de décrire la procédure **ordonne-point-marquant** qui crée une zone permise à partir des points marquants (on suppose toujours que la liste des points marquants n'est pas vide). A chaque point marquant on associe un booléen qui indiquera si le point a été utilisé.

```

procedure successeur-sur-arete( $T_i, A, pm, nb\_pm, ZP$ )
structure POINT-MARQUANT  $pm[ ]$ ; Entier  $nb\_pm$ ;
Arete  $A$ ; Point  $X$ ; Zone Permise  $ZP[ ]$ ;
debut
Entier  $i$ ; Flottant  $dist\_gauche, dist\_droite$ ;

 $dist\_gauche = dist\_droite = \text{BIG}$ ;
 $T_l = T_r = 0$ ;
pour  $i = 1, nb\_pm$ 
  debut
  si ( $pm[i] = T_i$  ou Arete ( $pm[i] \neq A$ )) continue;
  si ( $A = \text{Droite}$ )
    debut
     $X = \text{milieu}(T_i, pm[i])$ ;
    si (est-dans-toute-zone-permise( $X, ZP$ ) = FAUX) continue;
    si ( $pm[i]$  a-gauche  $T_i$  et distance( $pm[i], T_i$ )  $\leq dist\_gauche$ )
       $dist\_gauche = \text{distance}; T_l = pm[i]$ ;
    si ( $pm[i]$  a-droite  $T_i$  et distance( $pm[i], T_i$ )  $\leq dist\_droite$ )
       $dist\_droite = \text{distance}; T_r = pm[i]$ ;
    fin
  si ( $A = \text{Circulaire}$ )
    debut
    si ( $pm[i]$  a-gauche  $T_i$  et distance( $pm[i], T_i$ )  $\leq dist\_gauche$ )
       $dist\_gauche = \text{distance}; T_l = pm[i]$ ;
    si ( $pm[i]$  a-droite  $T_i$  et distance( $pm[i], T_i$ )  $\leq dist\_droite$ )
       $dist\_droite = \text{distance}; T_r = pm[i]$ ;
    fin
  fin
si ( $A = \text{Circulaire}$ )
  debut
  si (est-dans-toute-zone-permise ( $\text{milieu}(T_i, T_l), ZP$ ) = FAUX)
     $T_l = 0$ ;
  si (est-dans-toute-zone-permise ( $\text{milieu}(T_i, T_r), ZP$ ) = FAUX)
     $T_r = 0$ ;
  fin
 $T = \text{plus-proche}(T_r, T_l)$ ;
Arete  $A_n = \text{arete}(T_i, T)$ ;
retourne ( $A_n, T$ );
fin

```

Procedure 3: successeur-sur-arete

Initialement on prend un point marquant arbitraire de la liste, que l'on appelle *point initial* et on choisit de manière arbitraire une de ces deux arêtes. On cherche alors son successeur sur cette arête. Ce successeur est alors marqué comme traité et l'arête de liaison déterminée est ajoutée à la zone courante. Ce successeur devient alors le point courant, l'arête courante devient l'arête du point courant qui n'a pas servi à sa détermination et on recommence le processus.

Si le point courant est le point initial (figure 3.6) il est marqué et la zone courante est alors close. On ouvre une nouvelle zone, que l'on remplit à partir des points marquants non traités. Nous utilisons donc dans la procédure

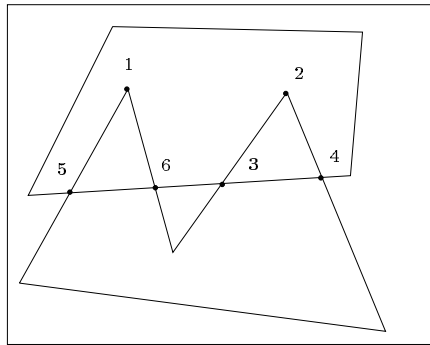


Figure 3.6: Ici nous avons 6 points marquants notés de 1 à 6. Lors de la mise en ordre les points 1,5,6,1 sont reliés. Comme on est revenu au point initial on a créé une zone. Le processus d'ordonnement est alors réitéré à partir des points 2,3,4 qui vont créer une nouvelle zone.

ordonne-point-marquant des procédures auxiliaires :

- `est-marque(P)` qui retourne VRAI si le point marquant P a été traité
- `marque(P)` qui indique que le point marquant P a été traité
- `autre-arete(P,A)` qui retourne l'arête du point marquant P différente de l'arête A
- `ajoute-arc(A,ZP)` qui ajoute l'arête A à la zone courante de la zone permise ZP

Notons aussi que la procédure `successeur-sur-arete` est légèrement modifiée pour ne considérer comme successeur que des points marquants non traités. Enfin il faut remarquer que le point initial lors de la création d'une zone n'est marqué que lorsque la zone est bouclée. La procédure `ordonne-point-marquant` écrite en pseudo-code est décrite en Procédure 4.

```

procedure ordonne-point-marquant(pm,nb_pm,ZP,Resultat)
POINT-MARQUANT pm [ ];Entier nb_pm;
Zone Permise ZP [ ],Resultat;Arete A
debut
Entier i, m, init;
POINT-MARQUANT Ti;
Arete An;
Resultat.nombre_de_zone=0;
new_zone:
pour i = 1,nb_pm
  debut
  si (est-marque(pm[i]))continue;
  A=Arete (pm[i])(0);
  m=i;init = i;
  Ti = pm[i];
  goto suite_zone;
  fin
/* Tous les points sont marques */
retourne ;
suite_zone:
/*On est revenu au point initial */
si (i != init et Ti=pm[init])
  debut
  marque(pm[init]);
  Resultat.nombre_de_zone=Resultat.nombre_de_zone+1;
  goto new_zone;
  fin
/*On recherche le successeur non marque */
(pm[j],An)=successeur-sur-arete(Ti,A,pm,nb_pm,ZP);
si (i!=init)marque(pm[i])
Ti=pm[j];
i = j;
A=autre-arete(Ti,A);
ajoute-arete(An,Resultat);
goto suite_zone;
fin

```

Procédure 4: ordonne-point-marquant

3.2.3.4 Procédure zone-est-seule

Dans les sections précédentes nous avons supposé que le nombre de point marquant était non nul. Il est cependant deux cas où cette hypothèse ne va pas être vérifiée : lorsque l'intersection des zones permises va être une des zones permises circulaires (figure 3.7) ou lorsque l'une des zones est circulaire et en dehors d'au moins une des autres, auquel cas l'intersection est vide. Le cas à traiter est donc celui où la frontière extérieure de chacune des zones des zones permises se trouve complètement à l'extérieur de la zone circulaire. La procé-

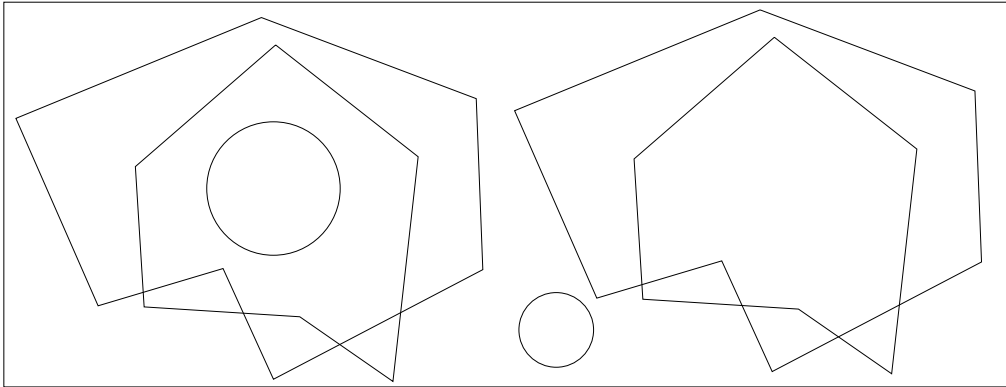


Figure 3.7: Deux cas où il n'y a pas de point marquant. La zone intersection se réduit au cercle ou à l'ensemble vide. Dans chaque cas la frontière extérieure des zones est complètement à l'extérieur de la zone circulaire.

ure **zone-est-seule** est chargée de gérer ces deux cas. Cette procédure prend en entrée l'ensemble des zones permises, retourne **VRAI** si une zone résultat est établie et place le résultat dans un de ces arguments. Dans un premier temps cette procédure détermine dans l'ensemble des zones permises le cercle de rayon minimal C_{min} non interne. Elle vérifie ensuite que toutes les zones des autres zones permises sont complètement à l'extérieur du cercle C_{min} . Si ce n'est pas le cas il n'est pas nécessaire de continuer puisque l'on ne se trouve plus dans un cas d'exception. Ensuite on teste si le centre du cercle C_{min} est à l'intérieur d'au moins une zone de toutes les zones permises et si le cercle C_{min} est à l'extérieur des cercles internes si il y en a. Si tel est le cas alors l'intersection se réduit au cercle C_{min} sinon l'intersection est vide. Cette procédure fait appel à différentes procédures auxiliaires:

- **trouve-cercle-minimal(Zone Permise)** qui retourne $(n, Cercle)$ si dans l'ensemble des zones permises il y a au moins un cercle, n étant le numéro de la zone, et **FAUX** si il n'y a pas de cercle.
- **centre-cercle(Cercle)** qui retourne le point centre du cercle.

- `zone-est-exterieur-cercle(Zone Permise ,Cercle)` qui retourne VRAI si toutes les zones de **Zone Permise** sont complètement à l'extérieur du cercle **Cercle** .
- `point-interieur-zone(Zone Permise ,Point)` qui retourne 1 si le point est à l'intérieur de la zone **Zone Permise** , 0 sinon.
- `zone-interne(Zone Permise)` qui renvoie la liste des zones internes de la zone **Zone Permise** .
- `cercle-interne-zone(Cercle ,Zone)` qui renvoie VRAI si le cercle **Cercle** est à l'intérieur de la zone **Zone** .

La procédure `zone-est-seule` écrite en pseudo-code est décrite en Procédure 5.

3.2.3.5 Procédure `zone-est-interne`

Avec les procédures précédentes nous pouvons déterminer l'ensemble des polygonarcs définissant des frontières de la zone résultat. Il reste à déterminer si un polygonarc donné constitue une frontière externe ou interne (c'est-à-dire si elle délimite une zone dont l'intérieur est interdit jusqu'à la frontière externe suivante). La figure 3.8 présente ainsi une région avec deux frontières externes et une frontière interne. La procédure `zone-est-interne` a justement pour but de déterminer la nature d'une zone. Les zones que nous avons à traiter n'ont clairement pas d'intersection. En conséquence pour tester si une zone est interne on prend un point quelconque de sa frontière et l'on détermine le nombre de zones contenant ce point: si ce nombre est pair la frontière est externe, sinon elle est interne. La procédure `zone-est-interne` met à jour la structure de données des zones pour indiquer si la frontière est interne ou externe et va utiliser une procédure auxiliaire `renvoie-point` qui renvoie un point de la frontière d'une zone. La procédure `zone-est-interne` écrite en pseudo-code est décrite en Procédure 6.

3.2.4 Algorithme général d'intersection

A partir des procédures décrites dans les sections précédentes on peut maintenant exposer la procédure `inter-zone` permettant le calcul de l'intersection d'un ensemble de zones permises. Cette procédure, écrite en pseudo-code, est décrite en Procédure 7. Une fois la zone *Resultat* déterminée il est possible de tester l'appartenance d'un point à cette zone à l'aide de la procédure `est-dans-zone-permise`, légèrement modifiée pour prendre en compte les zones internes et externes.

```

procedure zone-est-seule(ZP,nb,Resultat)
Zone Permise ZP[ ],Resultat;
Entier nb;
debut
Zone Permise ZI[ ];
Entier i, m, n;
Cercle Cercle;
Point X;

si (trouve-cercle-minimal(ZP)!= (n,Cercle))retourne FAUX;
X=centre-cercle(Cercle);
m=0;
pour i=1, i= nb
  debut
    si (zone-est-exterieur-cercle(ZP[i],Cercle)=FAUX)retourne FAUX;
    ZI=zone-interne(ZP[i]);
    si (cercle-interne-zone(Cercle,ZI)=VRAI)retourne FAUX;
    m=m+point-interieur-zone(ZP[i],X);
  fin
si (m=nb)
  debut
    Resultat=ZP[n];
    retourne VRAI;
  fin
sinon
  debut
    Resultat=∅;
    retourne VRAI;
  fin
fin

```

Procédure 5: zone-est-seule

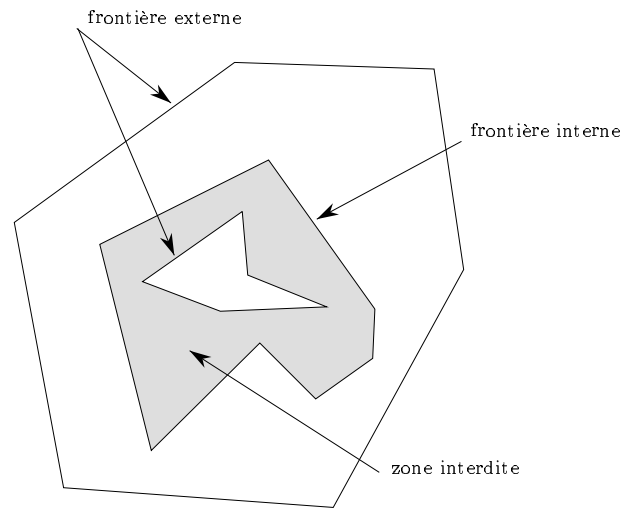


Figure 3.8: Une zone permise avec deux frontières externes et une frontière interne. La zone interdite est en grisée.

```

procedure zone-est-interne(ZP)
Zone Permise ZP;
debut
Entier i, j, k, m;
Point X;

pour j=1, ZP.nombre-de-zone
  debut
    m=0;
    X=renvoit-point(ZP.zone[j]);
    pour k=1, ZP.nombre-de-zone
      debut
        si (k=j) continue;
        m=m+point-interieur-zone(ZP[i].zone[k],X);
      fin
    si (m pair) ZP.zone[j].interne=1;
    sinon ZP.zone[j].interne=0;
  fin
fin

```

Procédure 6: zone-est-interne

```

procedure inter-zone(ZP,nb,Resultat)
/* intersection de nb zones permises */
Zone Permise ZP[ ], Resultat;
Entier nb;
debut

Entier nb_pm, j, k;
structure POINT-MARQUANT pm;

si (zone-est-seule(ZP,nb,Resultat)=VRAI) retourne ;
trouve-point-marquant(pm,ZP);
ordonne-point-marquant(pm,nb_pm,ZP,Resultat);
zone-est-interne(Resultat);
fin

```

Procédure 7: *inter-zone*

3.3 Prise en compte des contraintes sur les articulations du plateau mobile

3.3.1 Définition des contraintes

L'algorithme d'intersection des zones permises présenté dans la section précédente nous permet de déterminer l'espace de travail en prenant en compte les contraintes articulaires et les contraintes sur les articulations de la base.

Pour ce qui concerne les contraintes sur les articulations du mobile on peut clairement adopter le même modèle que celui utilisé pour les articulations de la base. On peut ainsi définir une pyramide P_i de sommet B_i telle que si les contraintes sur l'articulation sont satisfaites le point A_i se trouve à l'intérieur de la pyramide (figure 3.9). On va supposer que la pyramide est définie dans le repère B_i, x_i, y_i, z_i tel que

$$x_i = x_r \quad y_i = -y_r \quad z_i = -z_r$$

On peut imposer à la définition de P_i que seul son angle de rotation ψ_i autour de l'axe z_i puisse être non nul puisque chaque facette peut être décrite dans le repère de la pyramide. On peut alors définir une pyramide équivalente à P_i , P'_i , de sommet A_i tel que si A_i est à l'intérieur de P_i alors B_i est à l'intérieur de P'_i . Pour définir la pyramide équivalente il suffit de garder la même définition des facettes que la pyramide P_i mais poser que son repère de définition est obtenu à partir du repère absolu à l'aide des angles d'Euler $\psi_i + \psi, \theta, \phi$ où ψ, θ, ϕ sont les angles définissant l'orientation du plateau mobile (figure 3.10). Il faut noter que le modèle pyramide ne permet pas de décrire un certain nombre d'articulations rotule comme par exemple celle constituée d'un joint de Cardan dont la base

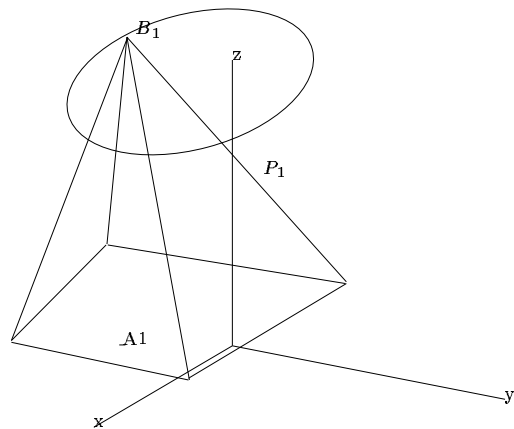


Figure 3.9: Définition de la pyramide caractérisant les contraintes sur l'articulation du plateau mobile. Le point A_1 se trouve à l'intérieur de la pyramide si les contraintes sont satisfaites.

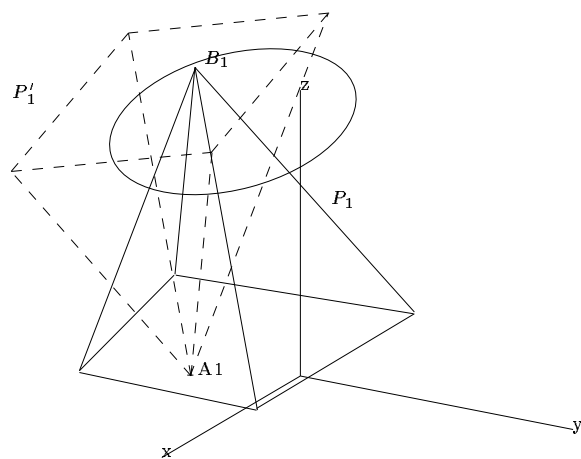


Figure 3.10: Définition de la pyramide équivalente caractérisant les contraintes sur l'articulation du plateau mobile. A partir de P_1 on peut construire P'_1 tel que si les contraintes sur l'articulation du plateau mobile sont satisfaites alors le point B_1 se trouve à l'intérieur de P'_1 .

est montée sur une articulation rotoïde.

3.4 Prise en compte des contraintes articulaires et des contraintes sur l'ensemble des articulations

A partir de la pyramide équivalente on voit qu'il est possible de calculer l'espace de travail en prenant en compte l'ensemble des contraintes articulaires et des contraintes sur les articulations. En effet la zone permise pour B_i est alors définie par l'intersection des couronnes annulaires obtenues à partir des contraintes articulaires, du polygone obtenu par intersection de la pyramide de base et du plan où se déplace B_i et enfin du polygone obtenu par intersection de ce même plan avec la pyramide équivalente de l'articulation du plateau mobile. Cette intersection est calculée par notre algorithme d'intersection des zones permises. Par simple translation de vecteur $\mathbf{B}_i\mathbf{C}$ on obtient alors une zone permise pour le point C qui définit la position du point C où les contraintes sur le segment i sont satisfaites. En faisant de même pour l'ensemble des segments on obtient ainsi 6 zones permises et l'espace de travail est obtenu comme intersection de ces zones permises, là aussi calculable par notre algorithme d'intersection des zones permises.

En conclusion de ce chapitre nous disposons d'une méthode qui permet de calculer efficacement la frontière de l'espace de travail en prenant en compte l'ensemble des contraintes sur les variables articulaires ainsi que les limitations mécaniques sur les articulations. Des exemples de résultat sont exposés dans le dernier chapitre de ce rapport. Il reste maintenant à traiter le dernier facteur limitatif de l'espace de travail, les possibilités d'intersection entre segments.

Chapitre 4

L'espace de travail limité par les intersections entre segments

4.1 Introduction

Les résultats présentés dans le chapitre précédent permettent d'établir la zone de travail en prenant en compte les contraintes articulaires et les contraintes sur les articulations. Le dernier facteur limitatif pour l'espace de travail est le risque d'intersection entre les segments.

4.2 Prise en compte des intersections entre segments

4.2.1 Notion de distance entre segment

Notre but est ici de rechercher les lieux de l'espace de travail précédemment déterminé tels que la distance minimum entre tous les points des couples de segment soit égale à une constante d non nulle appelée *distance de sécurité*, dont on supposera, pour des raisons évidentes qu'elle est inférieure au minimum des distances entre les points A_i, B_i des couples de segment considérés. La distance de sécurité a une valeur qui est associée au couple choisi de segments. Par la suite nous considérons le couple particulier de segment 1, 2 pour illustrer notre propos. Une telle approche permettra de traiter le cas des intersections entre segments de type cylindrique en imposant comme distance de sécurité pour la recherche de l'intersection entre deux segments la somme des rayons de chacun des segments. Par la suite pour simplifier on dira que tous les segments sont à la distance de sécurité d pour exprimer que la distance entre tous les couples de segments est au moins égale à leur distance de sécurité.

La distance entre les segments est définie de la manière suivante :

- si les points des droites associées aux segments appartenant à leur perpendiculaire commune sont sur les segments la distance est la distance entre les droites.
- sinon soient les projections des points A_1, B_1 sur la droite 2 et les projections des points A_2, B_2 sur la droite 1. On considère seulement les points dont la projection est sur le segment associé à la droite. Si l'ensemble de ces points est non vide la distance d est alors le minimum des distances de ces points au droite, sinon la distance d est définie comme le minimum des distances entre les points A_1, B_1, A_2, B_2 .

Nous allons démontrer que la distance ainsi définie correspond bien au minimum des distances entre tout couple de points appartenant aux segments 1 et 2.

La démonstration est triviale dans le cas où des points de la perpendiculaire commune aux deux droites appartiennent aux 2 segments. La distance est alors la distance entre les deux droites qui est bien un minorant de la distance entre tout couple de points de deux segments.

Nous supposons donc maintenant qu'au moins un des points appartenant à la droite et à la perpendiculaire commune n'appartient pas aux segments.

Nous allons démontrer le lemme suivant:

Lemme 1 : *Soit un point M du segment 1 dont la projection sur la droite 2 n'appartient pas au segment 2. Le minimum de la distance de M à un point du segment 2 est soit la norme de \mathbf{MA}_2 ou la norme de \mathbf{MB}_2 .*

Pour démontrer ce résultat il suffit de constater que la distance de M à un point Q de la droite 2 est une fonction monotone de la position de Q sur la droite 2. Comme par hypothèse le minimum de cette fonction est hors du segment 2 l'extremum de distance avec l'ensemble des points du segment est obtenu pour un des deux points extrémité du segment 2 \square

Lemme 2: *Le minimum des distances entre tous points de deux segments A_1B_1, A_2B_2 tels qu'aucune des projections des points du segment 1 (2) sur la droite associée au segment 2 (1) n'appartienne au segment 2 (1) est obtenu pour un des couples de point extrême sur les segments.*

Supposons qu'il existe un point M sur le segment 2, différent de A_2, B_2 tel que la distance de M à un point Q du segment 1 soit minimum sur l'ensemble des points du segment 2. Par application du lemme 1 le point Q est soit A_1 soit B_1 . On applique le lemme 1 sur le point Q pour montrer que la distance de M à Q est supérieure à la distance de Q à A_2 ou B_2 . Donc M ne peut être que A_2 ou B_2 \square

On va noter $P_{A_i}^j$ la projection de A_i sur la droite j et $P_{B_i}^j$ la projection de B_i sur la droite j et d_{12} le minimum des distances entre les points A_1, B_1, A_2, B_2 .

Par application directe du lemme 2 si aucun des $P_{A_1}^2, P_{A_2}^1, P_{B_1}^2, P_{B_2}^1$ n'appartient aux segments alors la distance entre les segments est d_{12} .

Comme par hypothèse on a supposé que la distance de sécurité était inférieure à d_{12} on va donc supposer maintenant qu'au moins un des $P_{A_1}^2, P_{A_2}^1, P_{B_1}^2, P_{B_2}^1$ appartient au segment.

Nous nous proposons de montrer maintenant que le minimum de distance entre les points des segments est alors soit d_{12} soit la distance d'un des points A_i, B_i à sa projection sur le segment opposé. Pour cela nous utilisons le lemme suivant:

Lemme 3: *Si les deux points extrêmes d'un segment ont leurs projections qui appartiennent toutes les deux au segment opposé alors le minimum de distance entre les points des deux segments est obtenu pour un des points extrêmes.*

Si les deux points extrêmes d'un segment ont leurs projections qui appartiennent toutes les deux au segment opposé alors tout point du segment a sa projection aussi sur le segment opposé. La distance entre les points étant monotone le minimum sera obtenu pour un point extrême \square

On va donc supposer maintenant qu'un des points extrêmes du segment a une projection hors du segment opposé. Soit A_2 tel que $P_{A_2}^1$ soit dans le segment A_1B_1 . Soit M le point du segment 2 tel que sur le segment A_2M tout point se projete sur le segment 1. Si M est B_2 on applique le lemme 3. On suppose donc M compris entre A_2 et B_2 . Sa projection sur le segment 1 est donc soit A_1 soit B_1 . Sans perte de généralité supposons que ce soit B_1 .

Si la projection $P_{B_1}^2$ de B_1 sur la droite 2 appartient au segment 2 alors la distance de M à B_1 est supérieure à la distance de $P_{B_1}^2$ à B_1 . Pour tout point S du segment MB_2 (donc dont la projection est en dehors de A_1B_1) la distance minimum au segment 1 est la norme de \mathbf{SB}_1 (par application du lemme 1), en conséquence supérieure à la distance de $P_{B_1}^2$ à B_1 . Comme la distance est monotone sur le segment A_2M le minimum de distance sera donc soit la distance de B_1 à sa projection, soit la distance de A_2 à sa projection (figure 4.1).

Supposons maintenant que la projection $P_{B_1}^2$ de B_1 sur la droite 2 n'appartienne pas au segment 2. Sur le segment MB_2 la distance minimum est réalisée soit pour le couple M, B_1 soit pour le couple B_1, B_2 par application du lemme 1. Sur le segment A_2M le minimum de distance est réalisé soit pour A_2 soit pour M . Considérons le triangle $MB_1P_{B_1}^2$. Par construction le point B_2 se trouve sur le côté de ce triangle opposé à B_1 , entre M et $P_{B_1}^2$ (figure 4.2). La ligne $B_1P_{B_1}^2$ étant perpendiculaire à la ligne 2 la ligne MB_1 constitue l'hypoténuse du triangle et par conséquent la distance de B_2 à B_1 est inférieure à celle de M à B_1 .



Figure 4.1: Illustration du minimum de distance lorsque les projections de deux points, un de chaque segment, appartiennent aux segments opposés.

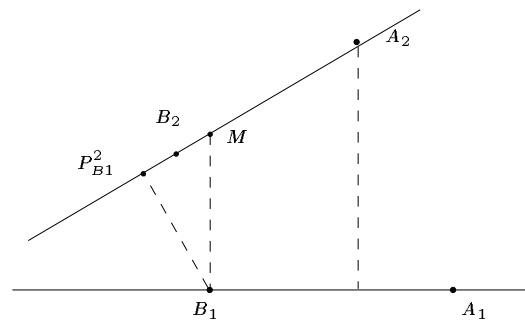


Figure 4.2: Illustration de la distance minimum lorsque la projection d'un point du segment appartient au segment opposé.

En conséquence nous venons de démontrer que la distance telle que nous l'avons définie correspond bien à la distance minimum entre tout couple de points des segments. Comme nous avons supposé que la distance de sécurité est toujours inférieure à d_{12} il s'agit donc maintenant de rechercher les zones de l'espace de travail où cette distance correspond soit à la distance entre les droites, soit à la distance entre les points extrêmes des segments et leur projection lorsque celle-ci appartient au segment opposé.

4.2.2 Lieux limites pour les droites, conique de sécurité

Considérons les deux droites associées à des segments notés arbitrairement 1 et 2 et soit l la distance entre ces droites. On a :

$$l^2 = \frac{(\mathbf{A}_1 \mathbf{A}_2 \cdot (\mathbf{A}_1 \mathbf{B}_1 \wedge \mathbf{A}_2 \mathbf{B}_2))^2}{\|\mathbf{A}_1 \mathbf{B}_1 \wedge \mathbf{A}_2 \mathbf{B}_2\|^2} \quad (4.1)$$

Si l'on pose que l doit être égale à la distance de sécurité d cette équation définit une contrainte sur les inconnues x_c, y_c et l'on peut montrer que cette contrainte s'écrit sous la forme:

$$a_1 x_c^2 + a_2 y_c^2 + a_3 x_c y_c + a_4 x_c + a_5 y_c + a_6 = 0 \quad (4.2)$$

En conséquence lorsque les droites associées aux segments 1 et 2 se trouve à la distance de sécurité le centre du plateau mobile se trouve sur une conique que nous appellerons *conique de sécurité* \mathcal{C}_l définie par l'équation précédente. La figure 4.3 en présente un exemple. Avec quelques manipulations sur les repères on peut déterminer la nature de la conique de sécurité à l'aide d'une quantité \mathcal{D} qui s'écrit:

$$\mathcal{D} = l^2 A^2 (\|\mathbf{A}_1 \mathbf{A}_2 \wedge \mathbf{B}_1 \mathbf{B}_2\|^2 - l^2 \|\mathbf{A}_1 \mathbf{A}_2 + \mathbf{B}_1 \mathbf{B}_2\|^2) \quad (4.3)$$

où A est une quantité qui ne s'annule que si il existe deux plans parallèles contenant les lignes passant par A_1, A_2 et B_1, B_2 . Si \mathcal{D} est strictement négatif la conique sera une ellipse, si \mathcal{D} est strictement positif la conique sera une hyperbole (ou deux droites concourantes) et si \mathcal{D} est nulle la conique sera une parabole (ou deux droites parallèles).

4.2.3 Premier cas: la conique de sécurité coupe \mathcal{W}

C'est évidemment une condition nécessaire pour que cette conique découpe l'espace de travail en zone où la distance entre les segments est strictement supérieure ou strictement inférieure à la distance de sécurité. Mais ce n'est pas suffisant car il faut de surcroît que deux points de la perpendiculaire commune appartiennent aux segments. Il faut donc déterminer les portions de la conique de sécurité où c'est effectivement le cas. Pour cela il est nécessaire d'introduire une autre conique la *conique des segments*.

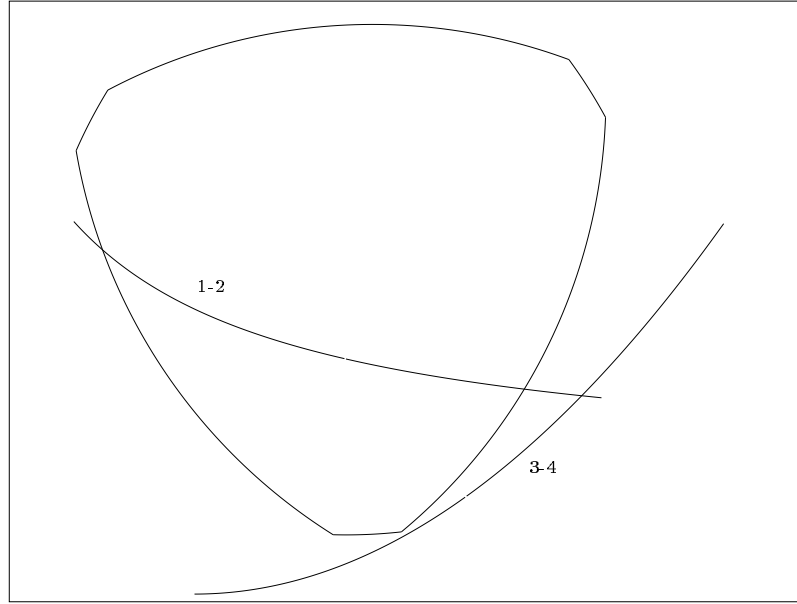


Figure 4.3: Deux exemples de conique de sécurité. Lorsque les droites associées aux segments 1-2, 3-4 se trouve à une distance fixée l'une de l'autre alors le point C se déplace sur une conique (ici une hyperbole pour 1-2 et une parabole pour 3-4).

4.2.3.1 Conique des segments

Pour trouver les portions de la conique de sécurité où des points de la perpendiculaire commune appartiennent aux segments nous allons tout d'abord définir un nouveau repère. Nous plaçons l'origine au point A_1 et nous choisissons les axes de manière à ce que les coordonnées de A_2 dans ce repère soit $(x_{a_2}, 0, z_{a_2})$. Les déplacements du plateau mobile sont repérés par $(x_{b_1}, y_{b_1}, z_{b_1})$ ce dernier terme restant constant. Soit M_1 le point de la droite 1 et M_2 le point de la droite 2 tels que la distance entre ces points soit égale à la distance de sécurité. On peut écrire:

$$\mathbf{A}_1\mathbf{M}_1 = (1 - \lambda)\mathbf{A}_1\mathbf{B}_1 \quad (4.4)$$

$$\mathbf{A}_2\mathbf{M}_2 = (1 - \alpha)\mathbf{A}_2\mathbf{B}_2 \quad (4.5)$$

Ces deux relations expriment que les points M_1, M_2 se trouvent sur les droites 1 et 2. Si les points appartiennent aux segments alors α, λ seront compris entre 0 et 1. Exprimons maintenant que les points M_1, M_2 sont les points définissant la distance entre les droites et que cette distance est la distance de sécurité d . Pour cela on va écrire que le vecteur $\mathbf{M}_1\mathbf{M}_2$ est perpendiculaire aux vecteurs $\mathbf{A}_1\mathbf{B}_1, \mathbf{A}_2\mathbf{B}_2$:

$$\mathbf{M}_1\mathbf{M}_2 \cdot \mathbf{A}_1\mathbf{B}_1 = 0 \quad (4.6)$$

$$\mathbf{M}_1\mathbf{M}_2.\mathbf{A}_2\mathbf{B}_2 = 0 \quad (4.7)$$

$$\mathbf{M}_1\mathbf{M}_2.\mathbf{M}_1\mathbf{M}_2 = d^2 \quad (4.8)$$

Ces trois équations sont quadratiques en terme de x_{b_1}, y_{b_1} . Toutefois l'équation obtenue par différence des équations (4.6), (4.7) est linéaire en ces termes. Il en sera de même si l'on procède à la différence entre l'équation (4.6) et l'équation (4.8) divisée par $\lambda - \alpha$. Nous allons donc tout d'abord faire l'hypothèse que $\lambda - \alpha$ est non nul. On obtient alors deux équations linéaires dont le déterminant ne s'annule que si $x_{a_2} = 0$ et $\lambda = \alpha$, cas que l'on peut donc écarter. Après résolution on reporte les valeurs de x_{b_1}, y_{b_1} dans l'équation (4.7) qui devient alors une équation en λ, α . Cette équation s'écrit sous la forme:

$$b_1\lambda^2 + b_2\alpha^2 + b_3\lambda\alpha + b_4\lambda + b_5\alpha + b_6 = 0 \quad (4.9)$$

C'est donc une conique que nous appellerons *conique des segments*, \mathcal{S}_s . Cette conique définit l'ensemble des positions des points M_1, M_2 telles que la distance entre les droites 1 et 2 soit la distance de sécurité. Mais nous ne sommes intéressé qu'au valeur de λ, α comprise entre 0 et 1, qui correspondent au cas où la distance de sécurité est obtenue pour des points appartenant aux segments A_1B_1, A_2B_2 . Cette contrainte définit dans le plan λ, α un rectangle \mathcal{R} et l'on peut distinguer deux cas selon la position de ce rectangle par rapport à la conique des segments (figure 4.4):

- \mathcal{R} n'est jamais coupé par la conique des segments. Quelle que soit la position de C la distance entre les droites sera soit toujours supérieure, soit toujours inférieure à la distance de sécurité. On prend n'importe quel point de l'espace de travail pour trouver dans quelle alternative on se trouve.
- \mathcal{R} a des intersections avec la conique des segments. Les parties de la conique des segments à l'intérieur de \mathcal{R} vont définir des portions de la conique de sécurité où la distance entre les segments est la distance de sécurité.

Le premier cas est donc trivial à traiter puisque pour un point de la conique de sécurité la distance entre les segments n'est jamais la distance entre les droites (nous traiterons ce cas dans une section ultérieure). Dans le second cas la restriction de la conique des segments contenue dans \mathcal{R} permet de définir une restriction sur les x_{b_1}, y_{b_1} qui correspond à une restriction sur la conique de sécurité définissant le lieu de l'espace de travail où la distance de sécurité est obtenue pour les segments. Nous reviendrons plus en détail sur ce passage. Nous avons en effet encore un cas particulier à traiter, celui où $\lambda = \alpha$. Dans ce cas particulier les équations (4.6, 4.7) deviennent linéaire en x_{b_1}, y_{b_1} mais ne sont plus indépendantes car leur différence ne dépend que de α ce qui définit une contrainte \mathcal{S} . L'équation (4.8) est indépendante de ces termes et est quadratique en α . Cette équation additionnée à $(\alpha)\mathcal{S}$ est linéaire en α . On trouve alors:

$$\alpha = \frac{d^2 - \|\mathbf{B}_1\mathbf{B}_2\|^2}{\mathbf{A}_1\mathbf{A}_2.\mathbf{B}_1\mathbf{B}_2 - \|\mathbf{B}_1\mathbf{B}_2\|^2} \quad (4.10)$$

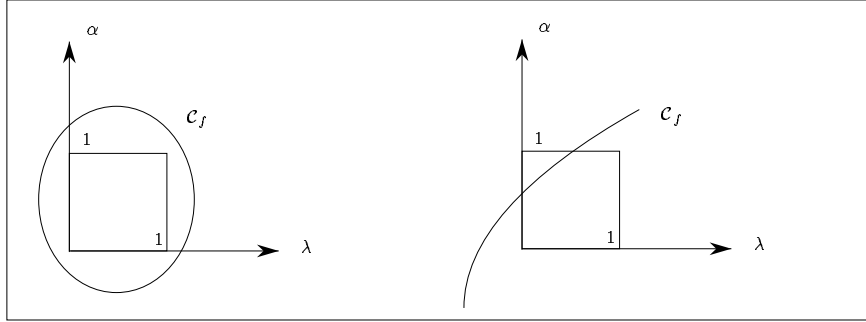


Figure 4.4: Dans le plan λ, α l'intersection de la conique des segments \mathcal{S}_s avec le rectangle défini par λ, α compris entre 0 et 1 permet de déterminer si les segments peuvent être ou non à la distance de sécurité. Si il n'y a pas d'intersection alors quelle que soit la position les segments sont toujours à une distance soit supérieure soit inférieure à la distance de sécurité.

Si le dénominateur de cette équation est nulle les deux segments sont parallèles, la distance entre les segments est donc identique à la distance entre les droites. On peut noter que α est supérieur à zéro si $\mathbf{A}_1 \mathbf{A}_2 \cdot \mathbf{B}_1 \mathbf{B}_2 < \|\mathbf{B}_1 \mathbf{B}_2\|^2$ et inférieur à 1 si $d^2 > \mathbf{A}_1 \mathbf{A}_2 \cdot \mathbf{B}_1 \mathbf{B}_2$. Avec cette valeur de α l'équation (4.8) se décompose alors en deux parties:

$$d^2 = \|\mathbf{B}_1 \mathbf{B}_2\|^2 \quad (4.11)$$

$$d^2 = \frac{\|\mathbf{A}_1 \mathbf{A}_2 \wedge \mathbf{B}_1 \mathbf{B}_2\|^2}{\|\mathbf{A}_1 \mathbf{A}_2 - \mathbf{B}_1 \mathbf{B}_2\|^2} \quad (4.12)$$

Dans le premier cas la distance de sécurité est égale à la distance entre les points B_1, B_2 ce que nous avons exclu. Si la deuxième équation est satisfaite alors le point C se déplace sur une droite qui constituera un cas dégénéré de la conique de sécurité. Nous appellerons *limite critique* de d^2 la valeur définie par cette deuxième équation.

4.2.3.2 Points et courbes critiques de la conique de sécurité

Supposons tout d'abord que la distance de sécurité soit différente de sa limite critique. L'intersection de la conique des segments avec \mathcal{R} permet de définir sur la conique de sécurité des points particuliers que nous appellerons des *points critiques*. Pour tous les points de la conique de sécurité compris entre deux points critiques on peut affirmer que soit la distance de sécurité est toujours obtenue pour des points appartenant aux segments ou, au contraire, jamais pour des points appartenant tous les deux aux segments. N'importe quel point pris entre les deux points critiques permet de décider entre ces deux choix. On appellera *courbe critique* toute portion de la conique de sécurité à l'intérieur de

l'espace de travail et telle que la distance de sécurité est obtenue pour des points appartenant aux segments.

Si la distance de sécurité est égale à la limite critique de d^2 on peut alors calculer la valeur de α qui est constante. Si cette valeur n'est pas dans l'intervalle $0, 1$ alors la distance de sécurité n'est jamais obtenue pour des points appartenant aux segments. Par contre si α est dans l'intervalle tous les points de la conique de sécurité sont tels que la distance entre les segments est d .

4.2.3.3 Détermination des courbes critiques

Supposons que nous ayons déterminé l'ensemble des points d'intersection et des points critiques de la conique de sécurité. Nous appellerons *points de séparation* tout point qui est soit critique soit un point d'intersection. Ces points de séparation vont découper la conique de sécurité en portion et certaines de ces portions vont définir les courbes critiques. Pour établir ces courbes on considère toute les portions des coniques limitées par deux points de séparation, que nous appellerons *points extrêmes*, et qui ne contiennent pas d'autre points de séparation que ces deux points, (figure 4.5). On a alors plusieurs cas:

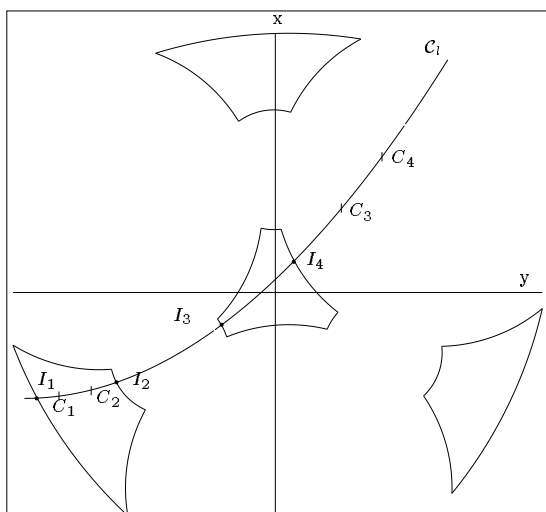


Figure 4.5: Exemple de séparation de la conique de sécurité en courbe critique. Les points critiques et les points d'intersection déterminent des portions de la conique dont on peut déterminer si elles sont des courbes critiques.

- les deux points de séparation sont des points critiques: par construction ils sont soit complètement à l'intérieur d'une zone de l'espace de travail (C_1, C_2) ou à l'opposé tous les deux sont à l'extérieur de l'espace de travail (C_3, C_4) et tous les deux du même côté de l'espace de travail. On teste

l'appartenance d'un des points à l'espace de travail pour décider dans quel cas l'on se trouve. Si les points sont à l'extérieur la portion ne peut pas être une courbe critique et l'on passe à la portion suivante. S'ils sont à l'intérieur on prend leur milieu sur la conique de sécurité. Si la distance entre les segments pour ce point est la distance entre les droites alors cette portion constitue une courbe critique, sinon on passe à la portion suivante.

- un point de séparation est un point critique et l'autre un point d'intersection. Si le point critique est à l'extérieur de l'espace de travail (I_4, C_3) la portion ne peut pas être une courbe critique car elle est complètement en dehors de l'espace de travail. Si il est à l'intérieur (I_1, C_1) on considère le milieu de la portion et si la distance entre les segments pour ce point est la distance entre les droites alors cette portion constitue une courbe critique, sinon on passe à la portion suivante.
- les deux points sont des points d'intersection (I_3, I_4): là aussi on teste leur milieu pour déterminer si la portion est une courbe critique. Ce sera le cas si ce milieu appartient à l'espace de travail et que la distance entre les segments pour ce point est égale à la distance entre les droites.

4.2.3.4 Utilisation des courbes critiques

Le cas où il n'existe pas de courbe critique pour la conique de sécurité sera étudié dans la section suivante puisqu'il sera équivalent au cas où il n'y a pas d'intersection de la conique de sécurité avec l'espace de travail. On va donc supposer que l'on a déterminé au moins une courbe critique. Deux cas se présentent : la courbe critique découpe l'espace de travail en deux zones distinctes (les points extrêmes de la courbe sont des points d'intersection) ou elle ne le découpe pas (au moins un des points extrêmes est un point critique).

Dans le premier cas (figure 4.6) une des deux zones est telle que la distance entre les deux droites est toujours supérieure à la distance de sécurité et, à fortiori la distance entre les deux segments le sera aussi. Par contre dans la zone où la distance entre les droites est inférieure à la distance de sécurité on ne peut pas statuer puisqu'il peut exister dans son intérieur des points où la distance entre les segments est obtenue pour des points extrêmes des segments avec une distance supérieure à la distance entre les droites. On peut simplement dire qu'il va exister un voisinage de la courbe critique où la distance entre les segments est inférieure à la distance de sécurité.

Dans le cas où la courbe critique ne découpe pas l'espace de travail en zone on peut simple dire qu'il va exister deux zones admettant la courbe critique comme élément de frontière où la distance entre segment sera respectivement supérieure et inférieure à la distance de sécurité (figure 4.7). Pour statuer dans les cas indécis il est nécessaire de s'occuper des cas où la distance entre les

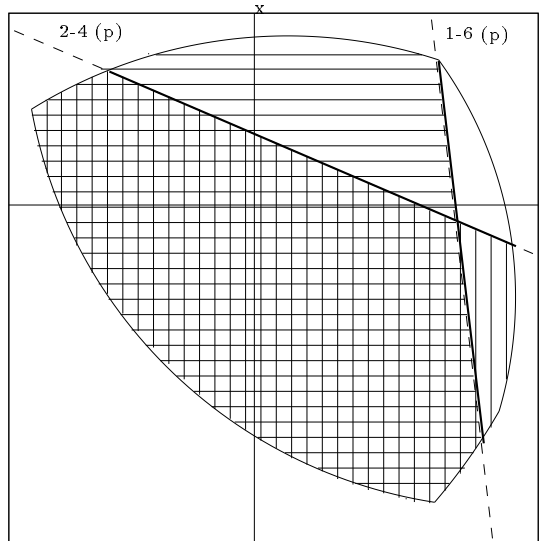


Figure 4.6: Exemple de séparation de la conique de sécurité en courbe critique. Les deux coniques (en pointillé) sont des paraboles dont les points critiques sont de part et d'autre de l'espace de travail. Les portions entre les deux points d'intersections sont des courbes critiques (en trait épais) qui sépare l'espace en deux zones. Dans les zones hachurées la distance entre les droites est toujours supérieure à la distance de sécurité et par conséquent la distance entre les segments aussi. Par contre dans la zone non hachurée on ne peut pas statuer.

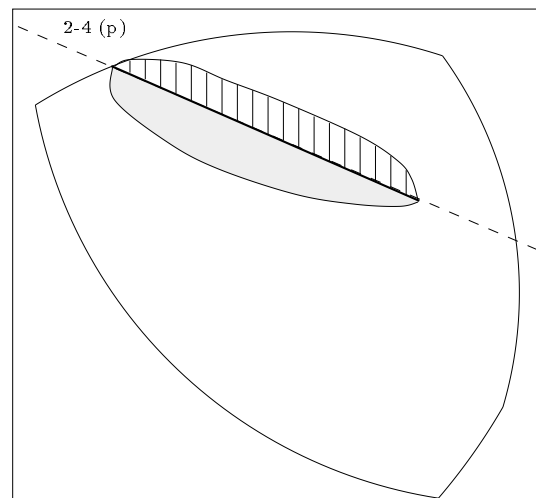


Figure 4.7: La courbe critique ne découpe pas l'espace de travail en zone. On peut simplement dire qu'il va exister deux zones admettant la courbe critique comme élément de frontière tels que la distance entre segment est supérieure à la distance de sécurité (en hachuré) ou inférieure à celle-ci (en grisées).

segments est obtenue pour des points extrêmes des segments. Comme nous aurons à le faire dans le cas où la conique de sécurité ne coupe pas l'espace de travail nous allons maintenant étudier ce cas.

4.2.4 Deuxième cas: la conique de sécurité ne coupe pas \mathcal{W}

On considère le cas où la conique de sécurité ne coupe pas l'espace de travail (par exemple pour les segments 3-4 dans l'exemple de la figure 4.3). Dans cette situation seul deux cas peuvent se présenter pour les points de l'espace de travail:

- la distance entre droites est toujours supérieure à la distance de sécurité
- la distance entre droites est toujours inférieure à la distance de sécurité.

Pour déterminer dans quel cas l'on se trouve il suffit de prendre n'importe quel point de l'espace de travail, de calculer la distance entre les droites et de voir si cette distance est supérieure ou inférieure à la distance de sécurité. Si l'on se trouve dans le premier cas la distance entre les segments sera toujours supérieure à la distance de sécurité. Par contre on ne peut pas conclure pour le premier cas car la distance entre les segments est obtenue pour un des points extrêmes d'un segment et est toujours supérieure à la distance entre les droites. On va introduire un autre concept qui nous permettra de lever l'ambiguïté.

4.2.4.1 Conique des points

Pour chacun des points A_1, A_2, B_1, B_2 on peut écrire une relation sur les coordonnées x_c, y_c qui indique que la distance du point à la droite est la distance de sécurité. Ainsi pour exprimer que le point A_1 se trouve à la distance d de la droite 2 on écrit:

$$d = \frac{\|\mathbf{A}_1 \mathbf{A}_2 \wedge \mathbf{A}_2 \mathbf{B}_2\|}{\|\mathbf{A}_2 \mathbf{B}_2\|} \quad (4.13)$$

Après développement de cette équation on obtient alors que le point C se trouve sur une conique que l'on va appeler la *conique des points*. On parlera ainsi de conique des A_1 , conique des A_2 etc..soit 4 coniques.

Si aucune de ces 4 coniques ne coupe l'espace de travail on peut déjà conclure que pour tout point de l'espace de travail la distance entre segment sera placée toujours de la même manière par rapport à la distance de sécurité, c'est-à-dire toujours inférieure ou toujours supérieure. Pour déterminer dans quel cas on se trouve il suffit de prendre un point particulier de l'espace de travail et de calculer la distance entre les segments.

Supposons maintenant qu'au moins une des coniques des points coupe la zone de travail. Nous allons supposer que c'est la conique de A_1 . Par analogie avec la conique de sécurité nous appellerons point critique les points de la conique

des points où la projection de A_1 sur la droite est un des points extrêmes du segment 2 c'est-à-dire A_2 ou B_2 . Si M est la projection de A_1 sur 2 on a :

$$\mathbf{A}_2\mathbf{M} = \frac{\mathbf{A}_2\mathbf{A}_1 \cdot \mathbf{A}_2\mathbf{B}_2}{\|\mathbf{A}_2\mathbf{B}_2\|^2} \mathbf{A}_2\mathbf{B}_2 \quad (4.14)$$

Pour obtenir la position des points critiques il suffit de poser dans l'équation précédente $M = A_2$ ou $M = B_2$ pour obtenir une équation dans les inconnues x_c, y_c . Si $M = A_2$ on obtient une équation linéaire dans les inconnues et l'on recherche donc l'intersection d'une droite avec la conique des points. Si $M = B_2$ l'équation amène à une conique et l'on va donc rechercher l'intersection de cette conique avec la conique des points. Il y aura donc un maximum de 4 points critiques.

Pour toute position de C sur la conique des points, comprise entre deux points critiques, la projection de A_1 va toujours se trouver soit entre A_2, B_2 soit à l'extérieur du segment.

Comme pour la conique de sécurité on a donc des points de séparation sur les coniques des points qui sont soit des points critiques soit des points d'intersection. Les coniques des points vont donc se trouver découpées en portions délimitées par les points de séparation. Après calcul des points de séparation on détermine l'ensemble des portions de cette conique et l'on rejete les portions qui ne sont pas dans l'espace de travail, ce que l'on détermine en testant l'appartenance du milieu de la portion à l'espace de travail.

Dans les portions restantes on élimine celles qui correspondent au cas où la projection de A_1 n'est pas sur le segment A_2, B_2 (là aussi on teste le point milieu de la portion). Après ces éliminations on obtient des portions qui ne peuvent pas être traversées par le point C puisqu'en ces points la distance entre les segments est au plus égale à la distance de sécurité. Ce sont donc les courbes critiques associées aux conique des points.

Ce sont ces courbes critiques qui vont compléter les courbes critiques obtenues pour la conique de sécurité pour nous permettre de statuer sur les cas ambigus. Des exemples de coniques des points et de courbes critiques associées sont présentées en figure 4.8. Dans le cas en haut à gauche il n'y a pas de courbe critique. Comme la distance entre segment pour un point aléatoire est supérieure à la distance de sécurité on en déduit que dans tout l'espace de travail la distance entre les segments est supérieure à la distance de sécurité. En haut à droite il existe deux courbes critiques et les zones S_1, S_2 sont les zones où la distance de sécurité n'est pas respecté. En bas il existe un ensemble de courbes critiques.

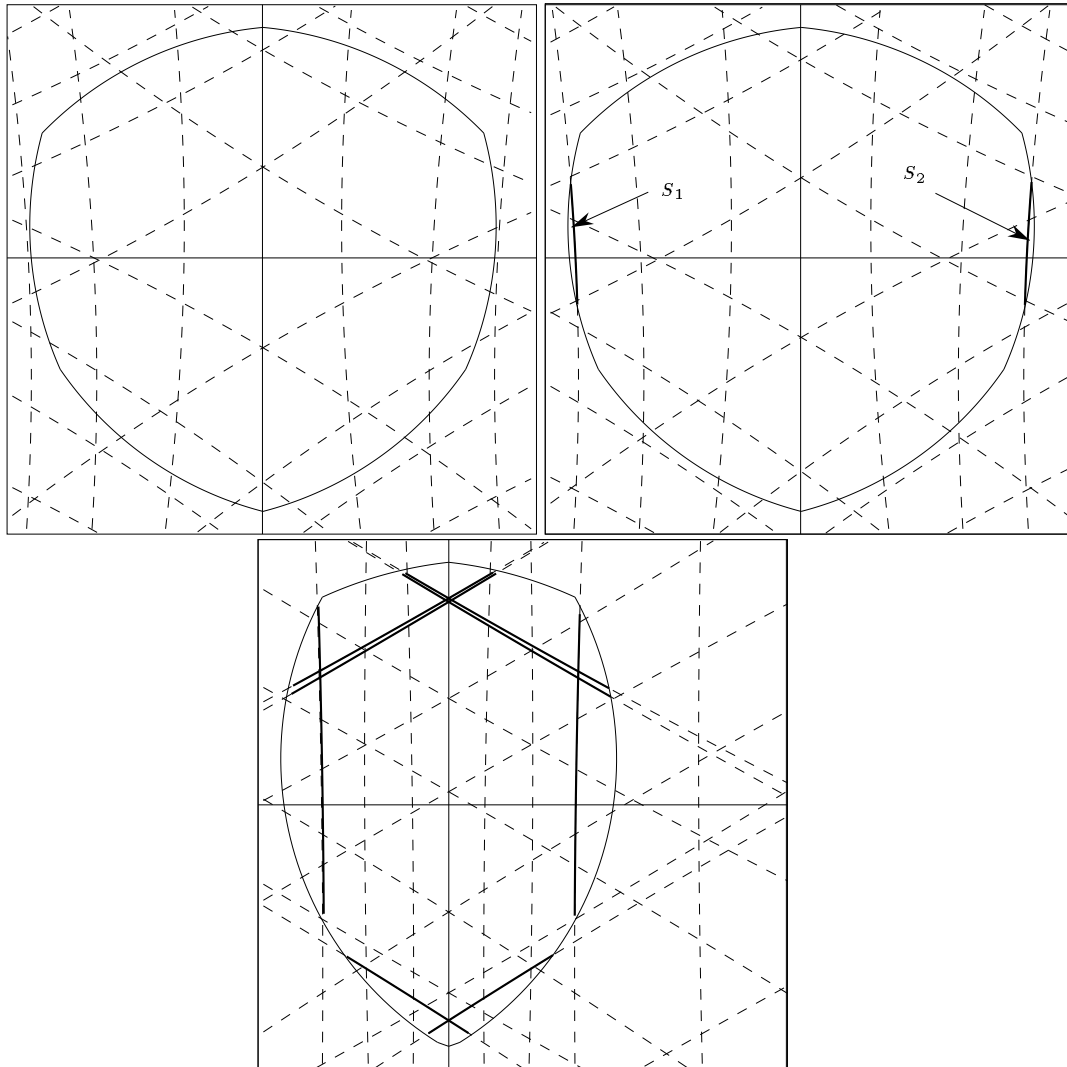


Figure 4.8: Exemples de conique des points et de courbes critiques associées (en trait épais). En haut à gauche pas de courbe critique, à droite 2 courbes critiques et en bas un ensemble de courbes critiques.

4.2.5 Partage de l'espace de travail

4.2.5.1 Zone délimitée par les courbes critiques

On dispose maintenant des courbes critiques dans l'espace de travail et l'on a mémorisé la nature de leurs points extrêmes, point critique ou point d'intersection. Nous allons montrer que l'ensemble des courbes critiques permet soit de définir une zone où la distance entre les segments sera toujours supérieure à la distance de sécurité soit amène à conclure qu'il n'existe pas une telle zone.

La première question que l'on va se poser est s'il est possible de trouver une courbe critique dont une extrémité n'est pas un point d'intersection et n'appartient pas à une autre courbe critique.

Pour cela on démontre tout d'abord le lemme suivant:

Lemme 4: *Toute extrémité d'une courbe critique associée à la conique de sécurité qui n'est pas un point d'intersection appartient à une courbe critique associée à une conique des points*

L'extrémité d'une courbe critique associée à la conique de sécurité correspond à la configuration où un point de la perpendiculaire commune est un point extrémité d'un des deux segments (sans perte de généralité A_2), l'autre point M appartenant au segment opposé (figure 4.9). Si la distance de M à A_2 est la

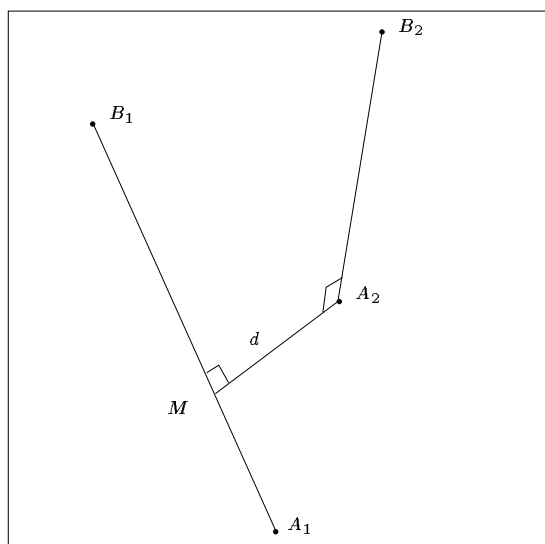


Figure 4.9: La configuration correspondant à la limite d'une courbe critique de la conique de sécurité.

distance de sécurité d il est alors clair que le point C appartient à la conique

des A_2 □

On peut aussi montrer le résultat suivant:

Lemme 5: *Toute extrémité d'une courbe critique associée à une conique des points est un point d'intersection*

Un point critique d'une courbe critique associée à une conique des points correspond au cas où la projection d'une extrémité d'un segment (supposons B_2) correspond à une extrémité du segment opposé (supposons B_1), les autres éléments définissant la distance étant supérieure à la distance de B_1 à B_2 . La distance de sécurité étant supposé inférieure strictement à d_{12} ce cas ne peut donc pas se produire. Par conséquent les points extrêmes d'une telle courbe critique ne peuvent être que des points d'intersection □

On peut maintenant énoncer le résultat suivant:

Lemme 6: *L'ensemble des courbes critiques permet de construire une zone fermée (éventuellement vide) où la distance entre les segments est toujours supérieure à la distance de sécurité*

Par application directe des lemmes 4 et 5 on peut dire que les courbes critiques définissent des zones fermées de l'espace de travail où la distance entre les segments est toujours supérieure ou inférieure à la distance de sécurité. Par intersection des zones où la distance est supérieure on obtiendra une zone fermée, éventuellement vide, telle que la distance entre l'ensemble des segments sera supérieure à la distance de sécurité.

4.2.5.2 Détermination de la zone de sécurité

On va appeler *zone de sécurité* \mathcal{S}_W la partie de l'espace de travail \mathcal{W} où la distance de sécurité est respectée pour l'ensemble des segments. Cette zone, si elle existe, va avoir une frontière constituée de portions de courbe critique et d'arêtes de l'espace de travail.

Dans un premier temps nous allons déterminer les arêtes de \mathcal{S}_W qui sont des portions de courbes critiques. Pour cela on va calculer la liste des points \mathcal{L} qui sont soit des points extrêmes d'une courbe critique soit l'intersection de deux courbes critiques et tels que la distance entre tous les segments est au moins égale à la distance de sécurité. Pour chacun de ces points on mémorise la courbe critique (pour les points extrêmes) ou les deux courbes critiques (pour les intersections) auquel il appartient. On mémorise aussi si le point est un point de la frontière de \mathcal{W} ou s'il est à l'intérieur de \mathcal{W} (auquel cas on dira que c'est un *point interne*).

Pour créer cette liste on considère tout d'abord chaque point extrême des courbes critiques (qui est forcément dans l'espace de travail) et l'on vérifie si pour ce point la contrainte sur les distances de sécurité est vérifiée pour tous les couples de segments auquel cas on mémorise le point. Ensuite on calcule les intersections des coniques qui ont des courbes critiques. Pour chacun des points d'intersection on regarde s'il est à l'intérieur de l'espace travail, si la contrainte sur les distances de sécurité est vérifiée pour tous les couples de segments et s'il appartient aux mêmes branches des coniques que les courbes critiques.

Chaque point de la liste \mathcal{L} ainsi obtenue est alors un sommet de \mathcal{S}_W . Si la liste est vide \mathcal{S}_W n'aura pas d'arête constituée de portion de courbe critique. Dans le cas contraire il faut chaîner les points.

Pour cela on prend un point arbitraire P_1 dans la liste, et l'on cherche quel est le point P_2 de la liste, le plus proche du point initial et qui est sur la même courbe critique C_1 (dans les cas d'intersection on choisit de manière arbitraire une des deux courbes critiques). Si le point trouvé est un point d'intersection la partie de la courbe critique entre les deux points constitue une arête de la zone \mathcal{S}_W et les points sont marqués comme traité (le point initial bénéficie d'un marquage spécial). Sinon on cherche le point non marqué le plus proche de P_2 qui est sur la courbe critique où n'est pas P_1 . On recommence alors le processus jusqu'à ce que l'on trouve un point d'intersection ou que l'on retrouve le point P_1 . On réitère le processus jusqu'à ce que tous les points soient marqués comme traités. Ainsi sur la figure 4.10, à gauche, si le point initial est T_1 on va trouver comme point suivant I_2 . La portion de 1-6 entre T_1 et I_2 est alors une arête de \mathcal{S}_W et le point I_2 est marqué. Le prochain point non marqué est I_1 dont le suivant est T_1 , qui est alors marqué. Il n'existe à ce stade plus de point non marqué donc la portion de 2-3 entre I_1 et T_1 est stockée comme arête de \mathcal{S}_W et le processus s'arrête. Si le point initial est I_1 le point suivant est T_1 , la portion de 2-3 entre I_1 et T_1 est stockée comme arête de \mathcal{S}_W . Le point T_1 est marqué et son suivant est I_2 , point d'intersection, donc la portion de 1-6 de T_1 à I_2 est stockée et le processus s'arrête.

Dans l'exemple de droite on va passer de T_1 à T_3 (stockage de la portion de 1-6 entre T_1, T_3), de T_3 à T_2 (stockage de la portion de 5-6 entre T_3, T_2), de T_2 à T_1 (stockage de la portion de 2-3 entre T_2, T_1). On est alors revenu au point initial, tous les points sont marqués et la procédure s'arrête.

Il ne reste maintenant plus qu'à déterminer les arêtes ou portion d'arête de \mathcal{W} qui sont des arêtes de \mathcal{S}_W . Pour cela on procède en deux étapes en déterminant tout d'abord s'il existe des arêtes complètes constituant une arête de \mathcal{S}_W . On parcourt la liste des arêtes de \mathcal{W} et l'on regarde si dans la liste \mathcal{L} il existe un point qui appartient à cette arête. Si aucun point n'y appartient on vérifie si le milieu de l'arête vérifie la distance de sécurité pour tous les segments. Si c'est le cas l'arête est une arête de \mathcal{S}_W . Ainsi sur la figure 4.10 l'arête $A_1 A_2$ est découverte comme arête de \mathcal{S}_W .

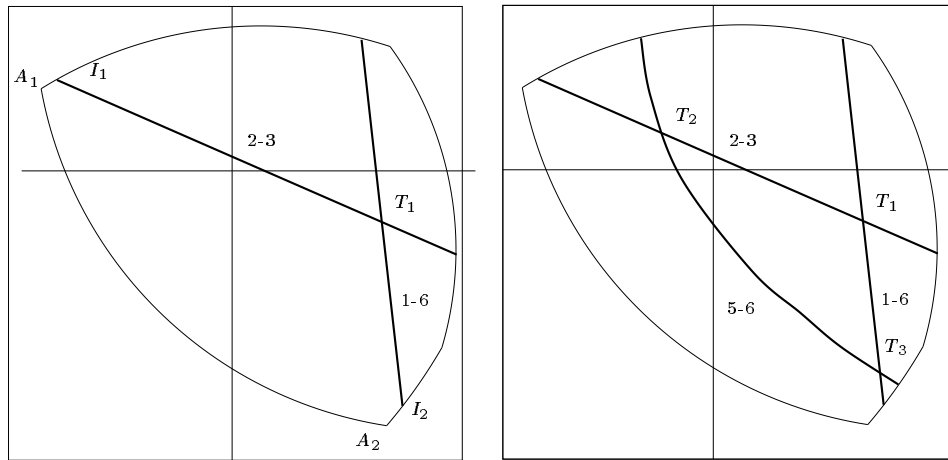


Figure 4.10: Comment on détermine les arêtes de \mathcal{S}_W qui sont des portions de courbe critique.

Si pour une arête il existe des points d'intersection on constitue une liste dont le premier élément est le point de départ de l'arête, les membres suivants les points d'intersections ordonnés et le dernier élément le point final de l'arête. On considère ensuite chaque partie de l'arête comprise entre deux éléments de la liste, on détermine son milieu et l'on teste si pour ce point la distance entre les segments est supérieure à la distance de sécurité. Si c'est le cas cette partie de l'arête fait partie de la frontière de \mathcal{S}_W . Ainsi sur la figure 4.10, à gauche on va trouver les arêtes I_1A_1 , A_2I_2 .

4.2.6 Exemples

La figure 4.11 présente quelques cas de détermination de la zone \mathcal{S}_W sans contrainte sur les articulations alors que la figure 4.12 présente des résultats ou on a pris en compte des contraintes sur les articulations de base. On peut constater que l'on peut obtenir des zones complètement internes à l'espace de travail.

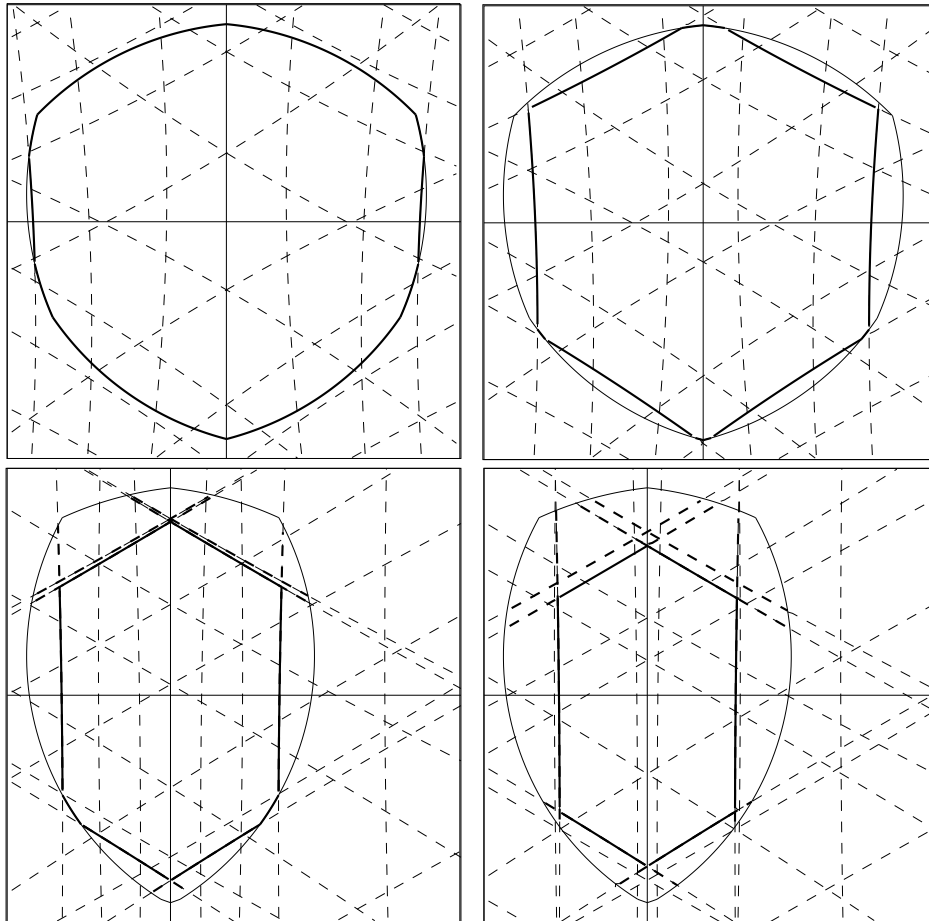


Figure 4.11: Exemple de détermination de zone S_W . Les courbes critiques sont en pointillé épais et la zone de sécurité en trait épais.

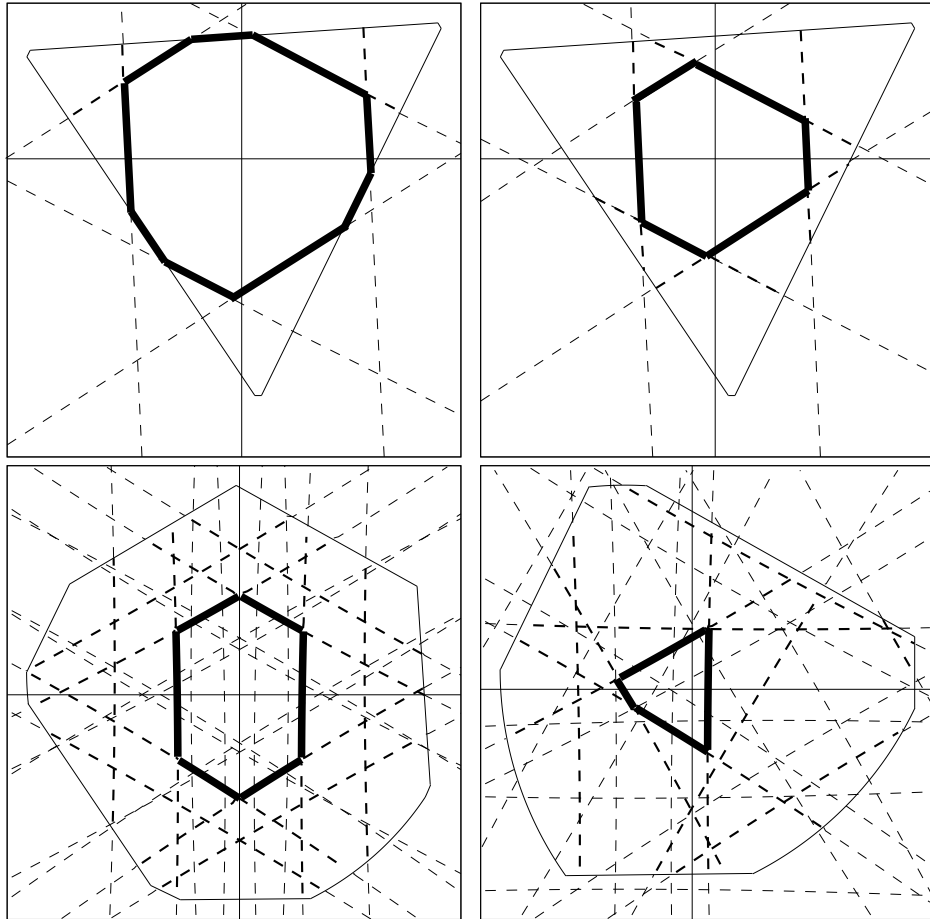


Figure 4.12: Exemple de détermination de zone S_W . Les courbes critiques sont en pointillé épais et la zone de sécurité en trait épais.

4.3 Cas particulier, $d = 0$

Dans cette section nous supposons que la distance de sécurité est nulle c'est-à-dire que l'on recherche les intersections entre segments.

4.3.1 Intersection entre les droites

La première condition pour que les segments se coupent et évidemment que les droites associées à ces segments se coupent. Il s'agit d'exprimer la condition de cette intersection en fonction des deux inconnus x_c, y_c . On peut représenter ces droites par leurs vecteurs de Plücker P , un vecteur de dimension 6 qui s'écrit:

$$P = \begin{pmatrix} \mathbf{A}_i \mathbf{B}_i \\ \mathbf{O} \mathbf{A}_i \wedge \mathbf{O} \mathbf{B}_i \end{pmatrix} \quad (4.15)$$

Remarquons alors que l'on a :

$$\mathbf{A}_i \mathbf{B}_i = \mathbf{A}_i \mathbf{O} + \mathbf{O} \mathbf{C} + \mathbf{C} \mathbf{B}_i \quad (4.16)$$

Dans le membre de droite de cette équation le premier et le troisième terme sont constant alors que les inconnus interviennent de manière linéaire dans le deuxième membre. Ainsi pour le segment 1 nous avons:

$$\mathbf{A}_1 \mathbf{B}_1 = \begin{pmatrix} x_c + x_{B_1} - x_{a_1} \\ y_c + y_{B_1} - y_{a_1} \\ z_c + z_{B_1} - z_{a_1} \end{pmatrix} \quad (4.17)$$

Pour le deuxième vecteur composante du vecteur de Plücker remarquons que:

$$\mathbf{O} \mathbf{B}_i = \mathbf{O} \mathbf{C} + \mathbf{C} \mathbf{B}_i \quad (4.18)$$

Là aussi les inconnues interviennent de manière linéaire dans les composantes. Pour le segment 1 nous avons:

$$\mathbf{O} \mathbf{A}_1 \wedge \mathbf{O} \mathbf{B}_1 = \begin{pmatrix} y_{a_1} (z_c + z_{B_1}) - z_{a_1} (y_c + y_{B_1}) \\ z_{a_1} (x_c + x_{B_1}) - x_{a_1} (z_c + z_{B_1}) \\ x_{a_1} (y_c + y_{B_1}) - y_{a_1} (x_c + x_{B_1}) \end{pmatrix} \quad (4.19)$$

Une caractéristique intéressante des vecteurs de Plücker est que la condition d'intersection de deux droites s'écrit de manière simple. Ainsi pour les droites associées aux segments 1 et 2 cette condition s'écrit:

$$\mathbf{A}_1 \mathbf{B}_1 \cdot (\mathbf{O} \mathbf{A}_2 \wedge \mathbf{O} \mathbf{B}_2) + \mathbf{A}_2 \mathbf{B}_2 \cdot (\mathbf{O} \mathbf{A}_1 \wedge \mathbf{O} \mathbf{B}_1) = 0 \quad (4.20)$$

Après développement de cette équation on trouve alors que la condition d'intersection s'écrit sous la forme :

$$u x_c + v y_c + w_1 z_c + w_2 = 0 \quad (4.21)$$

avec:

$$\begin{aligned}
 u &= y_{a_1} z_{b_1} + y_{a_2} z_{b_2} - z_{a_2} y_{b_2} + y_{b_1} z_{a_2} - z_{b_1} y_{a_2} - \\
 &\quad z_{a_1} y_{b_1} + y_{b_2} z_{a_1} - z_{b_2} y_{a_1} \\
 v &= z_{a_2} x_{b_2} - x_{b_1} z_{a_2} - x_{a_2} z_{b_2} + z_{b_1} x_{a_2} - x_{b_2} z_{a_1} + \\
 &\quad z_{a_1} x_{b_1} - x_{a_1} z_{b_1} + z_{b_2} x_{a_1} \\
 w_1 &= x_{b_2} y_{a_1} - y_{b_2} x_{a_1} + x_{b_1} y_{a_2} - y_{b_1} x_{a_2} + x_{a_2} y_{b_2} - \\
 &\quad y_{a_2} x_{b_2} + x_{a_1} y_{b_1} - y_{a_1} x_{b_1} \\
 w_2 &= x_{b_1} y_{a_2} z_{b_2} - z_{b_1} y_{a_2} x_{b_2} - x_{b_1} z_{a_2} y_{b_2} - x_{a_1} y_{a_2} z_{b_2} + x_{a_1} z_{a_2} y_{b_2} + \\
 &\quad y_{b_1} z_{a_2} x_{b_2} - y_{b_1} x_{a_2} z_{b_2} - y_{a_1} z_{a_2} x_{b_2} + y_{a_1} x_{a_2} z_{b_2} + z_{b_1} x_{a_2} y_{b_2} - \\
 &\quad z_{a_1} x_{a_2} y_{b_2} + z_{a_1} y_{a_2} x_{b_2} + x_{b_2} y_{a_1} z_{b_1} - x_{b_2} z_{a_1} y_{b_1} - x_{a_2} y_{a_1} z_{b_1} + \\
 &\quad x_{a_2} z_{a_1} y_{b_1} + y_{b_2} z_{a_1} x_{b_1} - y_{b_2} x_{a_1} z_{b_1} - y_{a_2} z_{a_1} x_{b_1} + y_{a_2} x_{a_1} z_{b_1} + \\
 &\quad z_{b_2} x_{a_1} y_{b_1} - z_{b_2} y_{a_1} x_{b_1} - z_{a_2} x_{a_1} y_{b_1} + z_{a_2} y_{a_1} x_{b_1}
 \end{aligned}$$

qui traduit que le lieu des points (x_c, y_c) où les droites associées aux segment se coupent est une droite D_C . Si cette droite ne coupe pas la zone permise pour C calculée dans les phases précédentes alors les segments ne se coupent pas. Toutefois si cette droite coupe la zone permise cela ne signifie pas que les segments se coupent, il faut de plus que l'intersection se trouve entre A_1, B_1 et A_2, B_2 .

4.3.2 Intersection entre les segments

Pour étudier cette contrainte supplémentaire on peut travailler en projection sur le plan de base. Nous faisons ici l'hypothèse que le point C a été choisi de manière à ce qu'aucun triplet C, B_i, B_j ne soit dans le même plan vertical. Dans ce cas les projections des points $C, B_1, B_2, C_p, B_{1p}, B_{2p}$ constituent un triangle qui se translate dans le plan de projection. Lorsque les droites associées aux segments se coupent le point C_p se trouve sur la droite D_C . De plus si les segments se coupent alors leurs projections vont aussi se couper (figure 4.13).

Pour déterminer s'il y a intersection des deux segments $A_1 B_1, A_2 B_2$ nous allons d'abord définir les deux droites D_{B_1}, D_{B_2} qui sont les droites décrites par les points B_{1p}, B_{2p} lorsque le point C_p se déplace sur la droite D_C . Ces deux droites sont parallèles à la droite D_C . Soient les demi-plans P_1, P_2 limités par les droites D_{B_1}, D_{B_2} et contenant les points A_1, A_2 (figure 4.14). Remarquons que les segments $A_1 B_1, A_2 B_2$ sont complètement contenus dans les demi-plan P_1, P_2 . En conséquence si ces segments se coupent leur point commun appartient à l'intersection de P_1, P_2 . En conséquence si cette intersection est vide les segments ne peuvent donc pas se couper (figure 4.14). Supposons maintenant que cette intersection soit non-vide. Il est facile d'établir que les segments se coupent si une des deux familles de condition suivante est vérifiée :

$$\mathbf{B}_1 \mathbf{A}_1 \wedge \mathbf{B}_1 \mathbf{B}_2 > 0$$

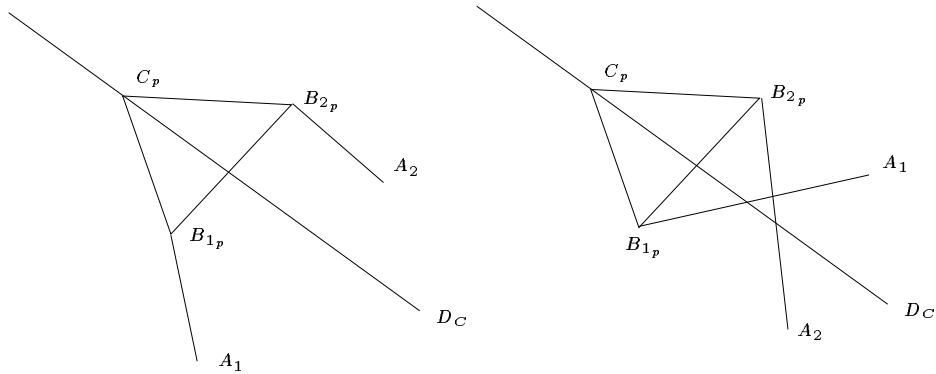


Figure 4.13: En projection dans le plan de base les points C, B_1, B_2 constituent un triangle dont l'orientation est constante. Si les droites associées aux segments se coupent le point C se déplace sur une droite D_C . Si de plus les droites se coupent entre les points A_1, B_1 et A_2, B_2 alors en projection les segments A_1B_1, A_2B_2 vont aussi se couper.

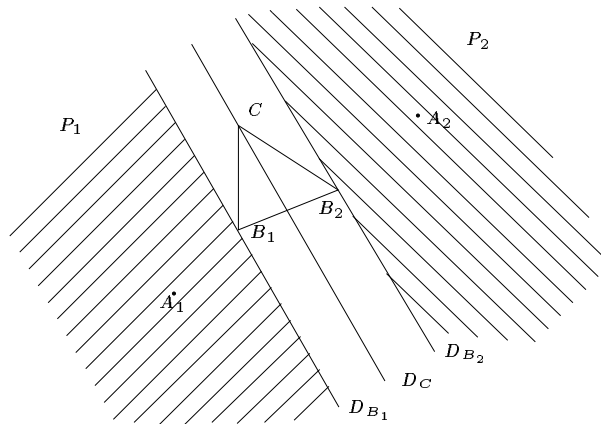


Figure 4.14: Les demi-plans P_1, P_2 limités par les droites D_{B_1}, D_{B_2} et contenant les points A_1, A_2 . Les demi-plans P_1, P_2 ont une intersection vide : les segments A_1B_1, A_2B_2 ne peuvent pas se couper.

$$\begin{aligned}
\mathbf{B}_2\mathbf{A}_2 \wedge \mathbf{B}_2\mathbf{B}_1 &< 0 \\
\mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{A}_1\mathbf{B}_1 &< 0 \\
\mathbf{A}_2\mathbf{A}_1 \wedge \mathbf{A}_2\mathbf{B}_2 &> 0
\end{aligned} \tag{4.22}$$

ou

$$\begin{aligned}
\mathbf{B}_1\mathbf{A}_1 \wedge \mathbf{B}_1\mathbf{B}_2 &< 0 \\
\mathbf{B}_2\mathbf{A}_2 \wedge \mathbf{B}_2\mathbf{B}_1 &> 0 \\
\mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{A}_1\mathbf{B}_1 &> 0 \\
\mathbf{A}_2\mathbf{A}_1 \wedge \mathbf{A}_2\mathbf{B}_2 &< 0
\end{aligned} \tag{4.23}$$

Notons C_1, C_2, C_3, C_4 les quatre quantités intervenant dans ces équations. Nous allons faire l'hypothèse que les vecteurs $\mathbf{B}_1\mathbf{B}_2, \mathbf{A}_1\mathbf{A}_2$ ne sont pas parallèles au vecteur \mathbf{u} , vecteur directeur des droites D_{B_1}, D_{B_2} . Pour C_1 il est alors clair que la droite passant par A_1 , de vecteur directeur $\mathbf{B}_1\mathbf{B}_2$ découpe la droite D_{B_1} en deux parties (de part et d'autre du point d'intersection U_1) telles que la quantité C_1 change de signe selon l'appartenance de B_1 à l'une ou l'autre de ces parties. De même pour C_2 la droite passant par A_2 de vecteur directeur $\mathbf{B}_1\mathbf{B}_2$ découpe la droite D_{B_1} en deux parties (de part et d'autre du point d'intersection U_2) telles que la quantité C_2 change de signe selon l'appartenance de B_1 à l'une ou l'autre de ces parties.

La droite D_{B_1} se trouve donc découpée en trois tronçons: une partie allant de l'infini à U_1 , la partie entre U_1, U_2 et la partie allant de U_2 à l'infini (figure 4.15). On va alors montrer que les seules parties où les quantités C_1, C_2 ont des signes opposés (condition nécessaire pour que les segments se coupent) ne peuvent être que les parties s'étendant à l'infini. Pour cela nous allons considérer la quantité C_2 et l'exprimer en fonction de C_1 :

$$C_2 = \mathbf{B}_2\mathbf{A}_2 \wedge \mathbf{B}_2\mathbf{B}_1 = -\mathbf{B}_1\mathbf{A}_1 \wedge \mathbf{B}_1\mathbf{B}_2 - \mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{B}_2\mathbf{B}_1 = -C_1 - \mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{B}_2\mathbf{B}_1 \tag{4.24}$$

Rappelons que sur les tronçons $]U_1, \infty],]U_2, \infty]$ les quantités C_1, C_2 ont des signes constants. Lorsque le point B_1 se déplace vers l'infini la quantité $\mathbf{B}_1\mathbf{A}_1 \wedge \mathbf{B}_1\mathbf{B}_2$ tend vers l'infini alors que la quantité $\mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{B}_2\mathbf{B}_1$ reste constante. En conséquence à l'infini les quantités C_1, C_2 sont bien de signe opposé. Ces quantités ayant un signe constant sur les tronçons considérés nous avons donc démontré notre résultat.

Considérons maintenant les quantités C_3, C_4 qui doivent être de signe opposé si les segments se coupent. La droite passant par A_1, A_2 découpe la droite D_{B_1} en deux parties (de part et d'autre du point d'intersection U_3) telles que les signes de C_3 sont opposés selon l'appartenance de B_1 à l'une ou l'autre des parties. Cette même droite découpe la droite D_{B_2} en deux parties (de part et d'autre du point d'intersection V_4) où les signes de C_4 sont opposés. En translatant V_4

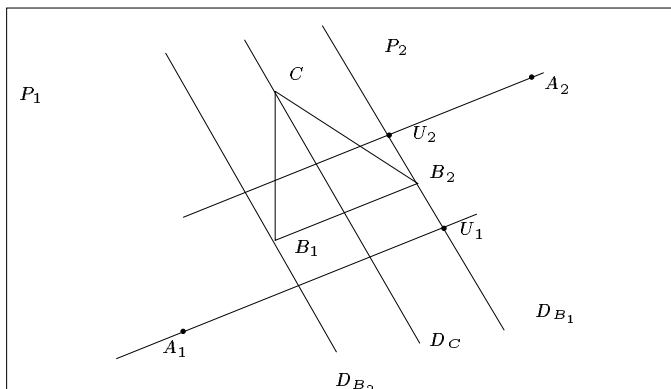


Figure 4.15: La droite D_{B_1} est séparée en trois tronçons par les points U_1, U_2 . Les tronçons où les quantités C_1, C_2 sont de signe opposé sont ceux qui s'étendent à l'infini.

du vecteur $\mathbf{B}_1\mathbf{B}_2$ on obtient alors un point U_4 découpant la droite D_{B_1} en deux parties telles que le signe de C_4 change selon l'appartenance de B_1 à l'une ou l'autre des parties (figure 4.16).

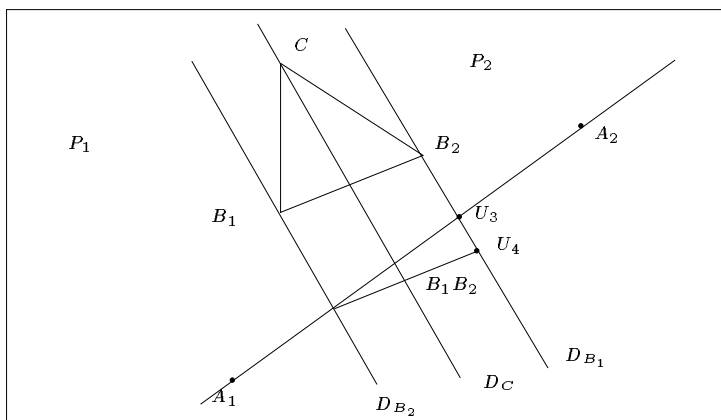


Figure 4.16: La droite passant par A_1, A_2 coupe D_{B_1} en U_3 et D_{B_2} en U_4 . En translatant U_4 de $\mathbf{B}_1\mathbf{B}_2$ on obtient le point U_4 sur D_{B_1} . La droite D_{B_1} est alors séparée en trois et les signes de C_3, C_4 sont opposés sur les tronçons s'étendant à l'infini.

Comme précédemment la droite D_{B_1} se trouve donc découpée en trois parties. Nous allons montrer maintenant que C_3, C_4 ne peuvent être de signe opposé que sur les tronçons s'étendant à l'infini. Pour cela considérons la quantité C_4 exprimé en fonction de C_3 :

$$C_4 = \mathbf{A}_2\mathbf{A}_1 \wedge \mathbf{A}_2\mathbf{B}_2 = -\mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{A}_1\mathbf{B}_1 - \mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{B}_1\mathbf{B}_2 = -C_3 - \mathbf{A}_1\mathbf{A}_2 \wedge \mathbf{B}_1\mathbf{B}_2 \quad (4.25)$$

Par le même raisonnement que précédemment nous voyons qu'à l'infini C_3, C_4 sont de signe opposé et en conséquence les parties de D_{B_1} où ces quantités sont de signe opposé sont bien les tronçons issus de U_3, U_4 et s'étendant à l'infini. La figure 4.17 présente deux cas typiques de situation pour les points U_1, U_2, U_3, U_4 .

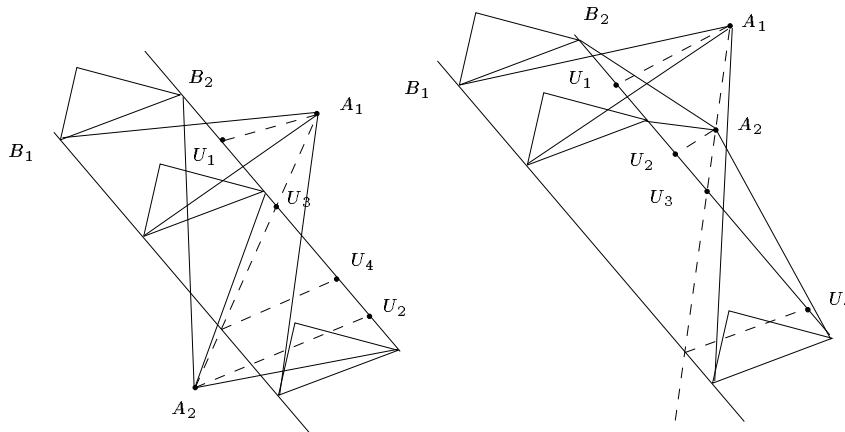


Figure 4.17: Cas typique de situation pour les points U_1, U_2, U_3, U_4 .

Nous disposons maintenant sur la droite D_{B_1} de 4 points U_1, U_2, U_3, U_4 et il suffit de prendre les 2 points extrêmes entre ces 4 points et de considérer les tronçons s'étendant de ces points à l'infini pour obtenir les parties de D_{B_1} telles que l'appartenance de B_1 à ces tronçons garantisse l'intersection entre segment. Par une simple translation de vecteur $\mathbf{B}_1\mathbf{C}$ on obtient alors les deux parties de D_C telle l'appartenance de C à l'un des deux tronçons garantit l'intersection des segments considérés.

4.3.3 Notion de segments interdits

Soit D_1, D_2 les deux tronçons de D_C déterminés dans la partie précédente. L'intersection de ces tronçons avec l'espace de travail déterminé par les contraintes articulaires et les contraintes sur les articulations peut être de trois types (figure 4.18):

- aucun des tronçons ne coupe la zone permise.
- un ou deux tronçons coupe complètement une des zones de la zone permise.
- un ou deux tronçons coupe partiellement une des zones de la zone permise.

Dans le premier cas aucun des segments ne peut se couper dans la zone permise. Dans le second le tronçon sépare la zone en deux parties et il est impossible de

franchir le segment constituant la limite entre les deux parties: on parlera alors de *segment interdit*. Seule la zone compatible avec le montage initial du manipulateur est accessible par translation. Dans le troisième cas on aura toujours un segment interdit mais qui ne découpe pas la zone. Si le point de départ et le but d'une trajectoire sont de part et d'autre du segment interdit il sera nécessaire de faire un "détour" mais tout point reste accessible.

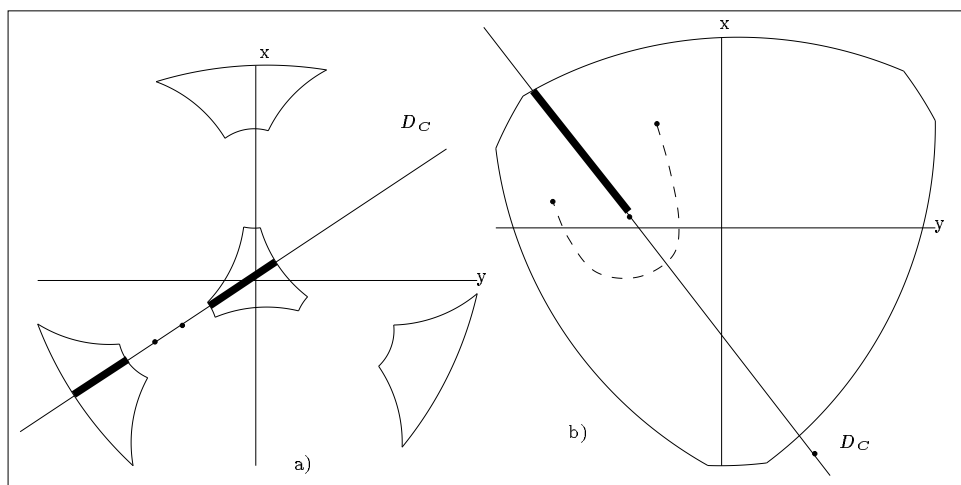


Figure 4.18: Les tronçons de D_C définissant l'intersection entre deux segments peuvent soit découper complètement une zone de l'espace de travail (a) ou simplement créer un segment interdit (b), indiqué en trait épais. Dans le premier cas on ne peut pas passer d'une partition à l'autre alors que dans le second il sera toujours possible d'aller d'un point à un autre au prix, parfois, d'un détour (indiqué en pointillé).

4.3.4 Cas particuliers

Trois cas particuliers sont possibles selon que les vecteurs $\mathbf{B}_1\mathbf{B}_2, \mathbf{A}_1\mathbf{A}_2$ sont parallèles au vecteur \mathbf{u} , vecteur directeur des droites D_{B_1}, D_{B_2} . Si ces deux vecteurs ne sont pas simultanément parallèles à \mathbf{u} on ne dispose alors que de deux points parmi les U_i mais le raisonnement reste identique. Dans le cas où les deux vecteurs sont simultanément parallèles à \mathbf{u} il est facile de constater que les segments sont toujours ou jamais sécants quelque soit la position de C sur sa droite.

4.3.5 Exemples

La figure 4.19 présente quelques cas de calcul de segment interdit. On y voit que l'on peut avoir un ou plusieurs segments interdits qui coupent ou non complètement l'espace de travail. Ainsi en 1) nous avons un segment interdit qui coupe

la zone de travail en deux zones. En 2), 3) nous avons un ou deux segments interdits qui ne créent aucune zone alors qu'en 4) nous avons deux segments interdits qui découpent l'espace de travail en trois zones.

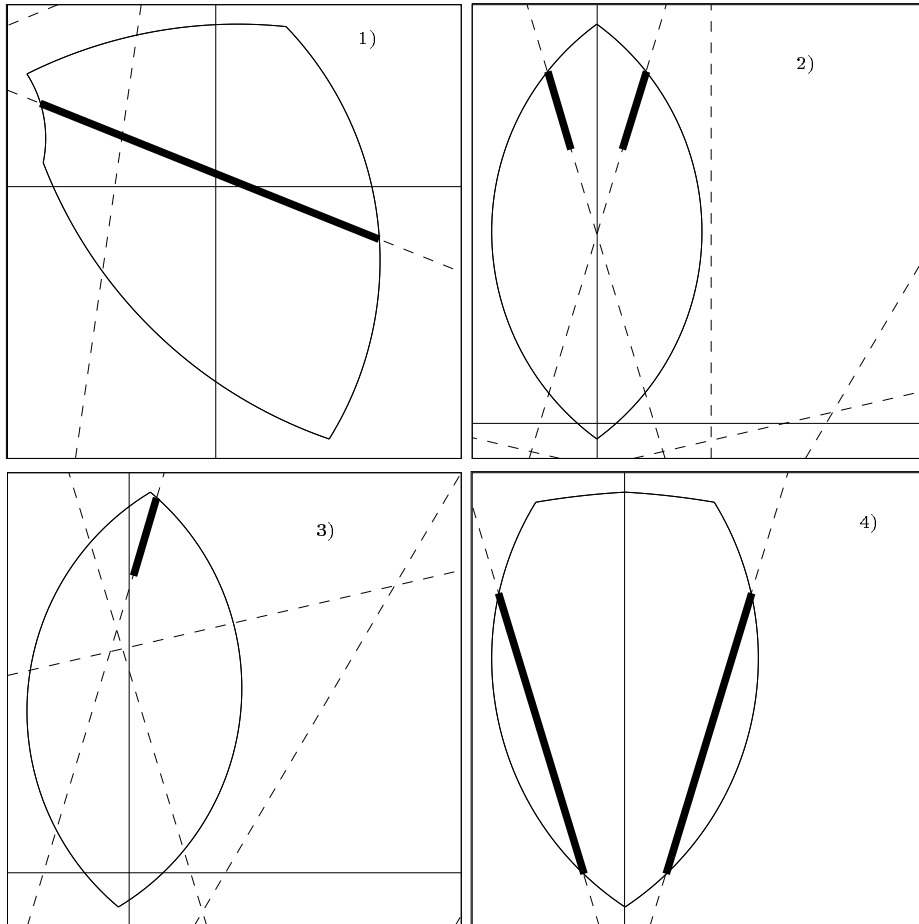


Figure 4.19: Exemple de calcul de segments interdits (en trait épais).

4.4 Surface et volume de l'espace de travail

La description géométrique de la frontière de l'espace de travail que nous obtenons permet, comme le fait remarquer Gosselin [6], d'obtenir par la formule de Green la surface d'une coupe, c'est-à-dire la surface de l'espace atteignable par le manipulateur. A l'aide de plusieurs coupes en altitude il est alors possible de calculer efficacement le volume atteignable.

Chapitre 5

Conclusion

La méthode de calcul géométrique de l'espace de travail que nous avons proposé permet de prendre en compte l'ensemble des contraintes exerçant une influence effective sur l'espace de travail: limitations articulaires, débattements des articulations et intersections entre segments. Elle repose principalement sur des procédures géométriques peu coûteuses en temps de calcul et donne un résultat exact dans la mesure où les éléments physiques intervenant dans le manipulateur entrent dans le cadre de la modélisation proposée: contraintes sur les articulations pouvant être décrites par des pyramides et segments de type cylindrique.

Elle présente cependant l'inconvénient de ne calculer que des coupes de l'espace de travail, ce qui impose une discrétisation sur l'altitude du manipulateur. En raison de la rapidité des calculs cela ne crée pas vraiment de problème pour la grande majorité des prototypes existants qui ont une faible amplitude de déplacement en altitude.

Toutefois il pourrait être concevable de proposer non plus des coupes mais directement le volume par extension de notre méthode. On aurait ainsi à calculer des intersections de sphères, polyèdres et de surfaces d'ordre 2 pour ce qui concerne les distances entre segments, qui, si elles sont plus complexes, sont néanmoins calculables.

Après avoir traité le problème de l'espace de travail à orientation constante nous nous proposons d'entreprendre une démarche du même type lorsque le centre du plateau mobile occupe une position fixe et que son orientation est quelconque. On peut alors représenter l'espace de travail en orientation en illustrant les déplacements possibles d'un point du plateau mobile différent du centre. Une approche géométrique est, là aussi, envisageable.

Remerciements: Le travail présenté dans ce rapport a bénéficié largement de nombreuses discussions avec C. Gosselin, T. Arai et R. Stoughton. Les as-

pects géométriques ont pu être traités grâce à l'aide précieuse des membres du projet PRISME, principalement J-D. Boissonnat, Mariette Yvinec et Jacqueline Duquesne. Mais les autres membres de ce projet ont aussi apporté une contribution par leurs exigences à propos du système qui a permis d'obtenir les dessins de ce rapport dont il m'est déceimment pas possible de remercier l'auteur...

Chapitre 6

Appendice : Exemples d'espace de travail

Une grande partie des idées présentées dans ce rapport ont été élaborées lors d'un séjour au Mechanical Engineering Laboratory de Tsukuba à l'invitation du Dr. Arai. On va donc présenter des résultats pour des manipulateurs construits dans ce laboratoire.

6.1 Micro-manipulateur

Il s'agit d'un manipulateur avec des actionneurs piézo-électriques permettant des variations maxima de longueur des segments de $8 \mu\text{m}$. Ici le seul facteur limitatif de l'espace de travail est les longueurs articulaires, les débattements des articulations de la base et du mobile ne jouant pas de rôle. On conçoit que l'espace de travail d'un tel manipulateur va être très réduit. La position des points d'articulation (en μm) est donnée dans les tableaux suivants. Nous présentons des vues 3D de l'espace de travail de ce manipulateur pour différentes orientations. Pour des raisons de clarté les dimensions en z sont multipliés par 1.5 ou 2 dans les vues en perspective.

	1	2	3	4	5	6
x	13982.5	-7597.22	-6385.29	-6385.29	-7597.22	13982.5
y	699.708	11759.3	-12459.1	12459.1	-11759.3	-699.708
z	0	0	0	0	0	0
x_b	4341.27	-7990.0	3648.74	3648.74	-7990.0	4341.27
y_b	6719.63	399.833	-7119.46	7119.46	-399.833	-6719.63
z_b	20000	20000	20000	20000	20000	20000

Tableau 6.1: Position des points d'articulation de la base et du mobile pour un micro-manipulateur (en μm).

23012.2	23012.2	23012.2	23012.2	23012.2	23012.2
23004.2	23004.2	23004.2	23004.2	23004.2	23004.2

Tableau 6.2: Longueur minima et maxima des segments du micro-manipulateur (en μm).

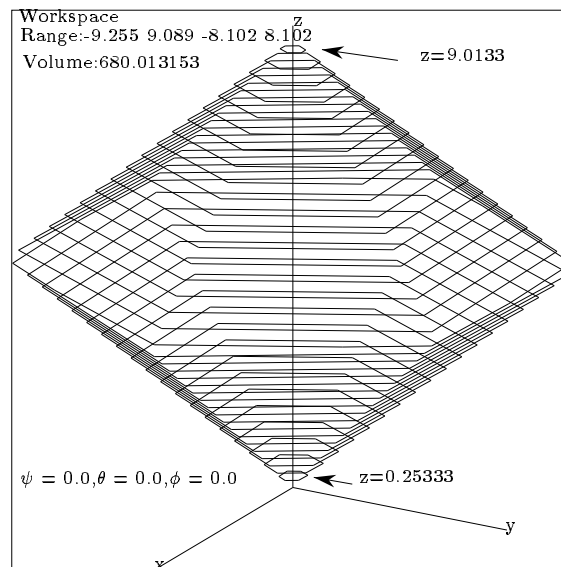


Figure 6.1: Vue perspective de l'espace de travail du micro-manipulateur, orientation $\psi = \theta = \phi = 0$ (dimensions en μm).

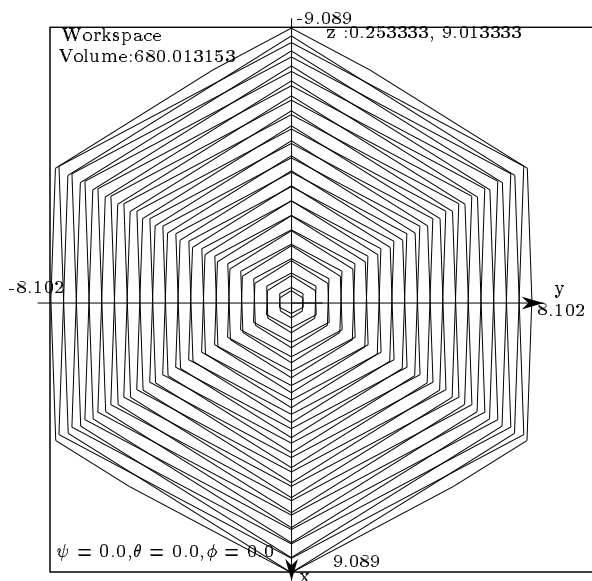


Figure 6.2: Vue de dessus de l'espace de travail du micro-manipulateur, orientation $\psi = \theta = \phi = 0$ (dimensions en μm).

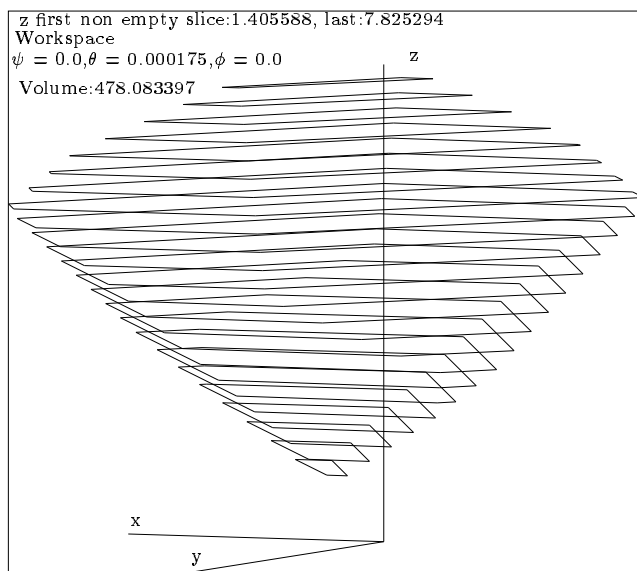


Figure 6.3: Vue perspective de l'espace de travail du micro-manipulateur, orientation $\theta = 0.00175$ degré, $\psi = \phi = 0$ (dimensions en μm).

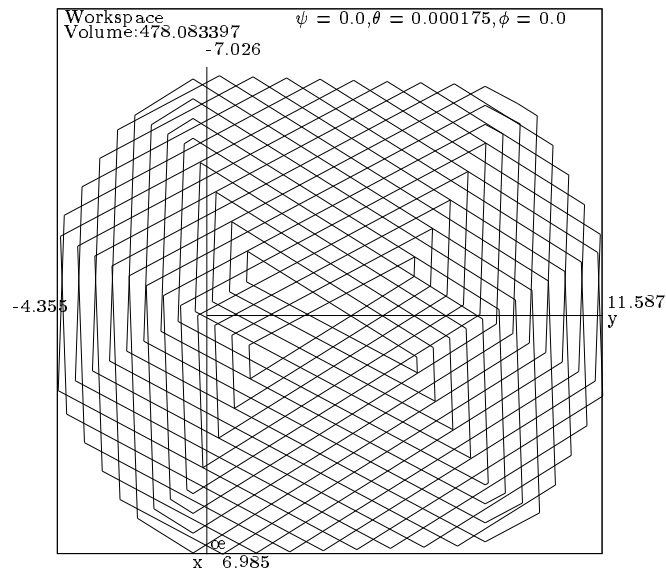


Figure 6.4: Vue de dessus de l'espace de travail du micro-manipulateur, orientation $\theta=0.00175$ degré, $\psi = \phi = 0$ (dimensions en μm).

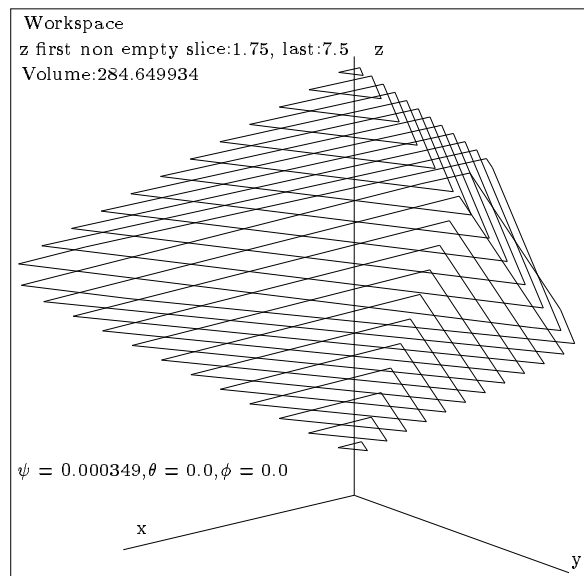


Figure 6.5: Vue perspective de l'espace de travail du micro-manipulateur, orientation $\psi=0.00349$ degré, $\theta = \phi = 0$ (dimensions en μm).

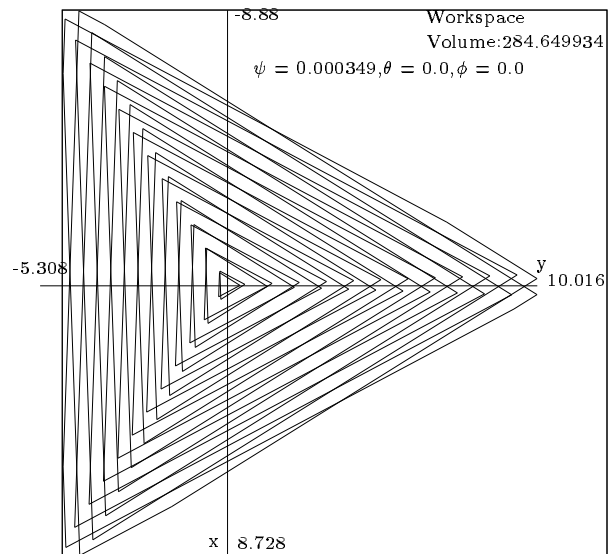


Figure 6.6: Vue de dessus de l'espace de travail du micro-manipulateur, orientation $\psi=0.00349$ degré, $\theta = \phi = 0$ (dimensions en μm).

6.2 Un prototype pour l'assemblage

Dans ce prototype les articulations de base se trouvent en dessous de la base, les segments émergent d'un orifice carré (figure 6.7). On peut donc modéliser les contraintes sur les articulations de la base par des pyramides à 4 faces orientées vers le centre de la base. Nous présentons ici des vues de l'espace de travail avec prise en compte seulement des limitations articulaires, puis, pour les mêmes orientations, avec prise en compte des contraintes sur les articulations de base. On pourra constater que les contraintes sur les articulations influent sensiblement sur l'espace de travail.

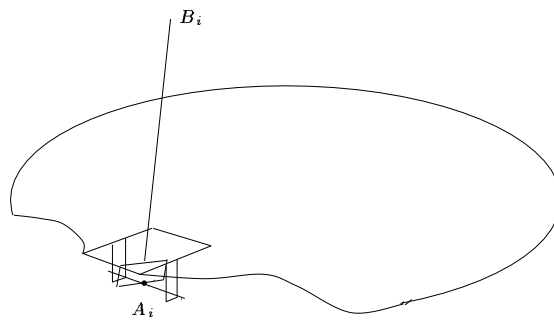


Figure 6.7: Les articulations de base du prototype. Les centres A_i des articulations se trouvent en dessous du plateau de base et les segments sont contraints par l'orifice carré C .

On peut remarquer aussi que les articulations présentent une disposition croisée qui, si elle favorise l'espace de travail en orientation, implique un plus grand risque d'intersection entre segments. Nous verrons en effet que c'est bien le cas et que pour une distance de sécurité de 8 mm l'espace de travail se trouve réduit par les collisions entre segments.

La position des points d'articulation (en mm) est donnée dans les tableaux suivants. Tous les angles sont donnés en degré.

	1	2	3	4	5	6
x_a	112.5	-225	-112.5	67.5	-135	67.5
y_a	-194.856	0	194.856	-116.913	0	116.913
z_a	-25	-25	-25	-25	-25	-25
x_b	95.263	0	-95.263	-69.282	0	69.282
y_b	55	-110	55	-40	80	-40
z_b	-20	-20	-20	-20	-20	-20

Tableau 6.3: Position des points d'articulation de la base et du mobile pour un prototype (en mm).

757	757	757	733	733	733
528	528	528	491	491	491

Tableau 6.4: Longueur minima et maxima des segments du prototype (en mm).

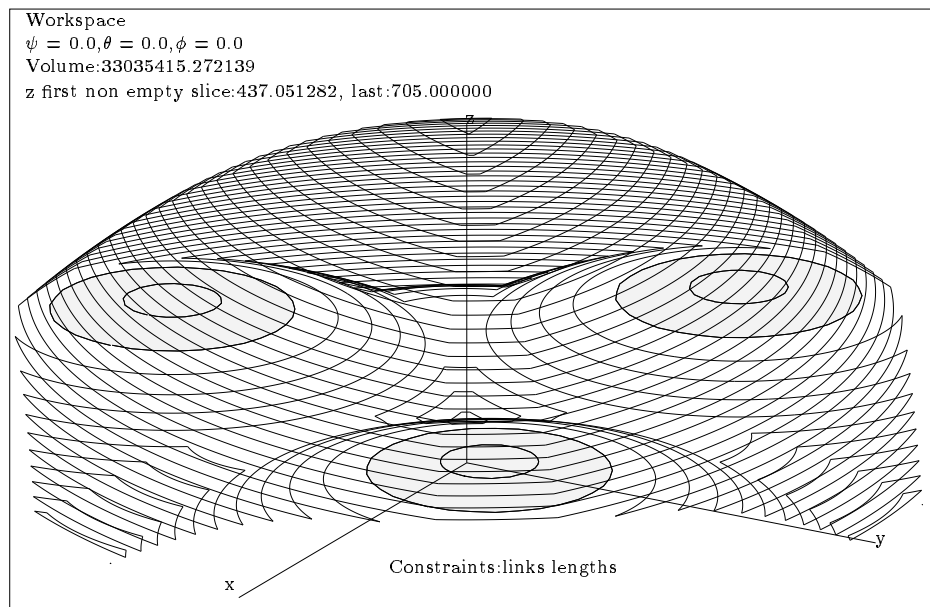


Figure 6.8: Vue perspective de l'espace de travail du prototype, orientation $\psi = \theta = \phi = 0$ (dimensions en mm, les parties grisées sont des zones interdites).

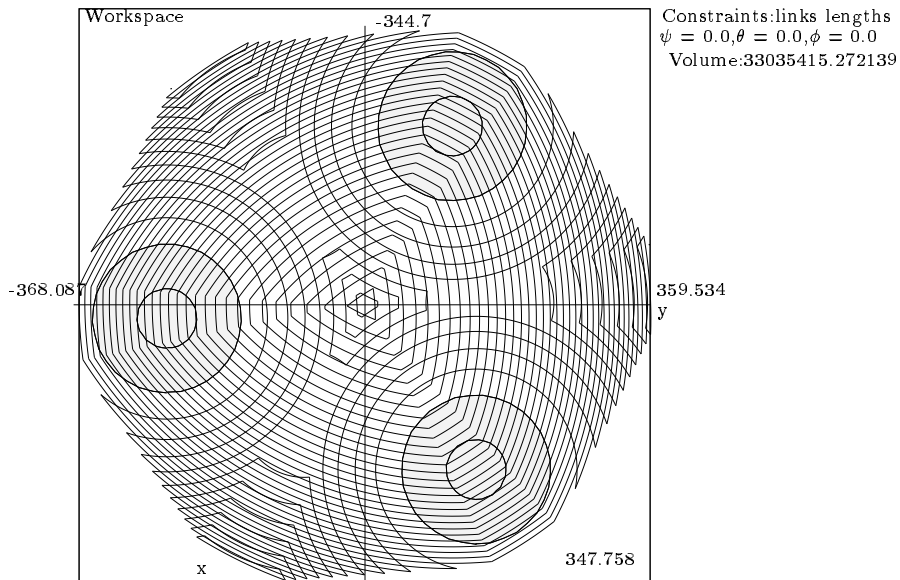


Figure 6.9: Vue de dessus de l'espace de travail du prototype, orientation $\psi = \theta = \phi = 0$ (dimensions en mm, les parties grisées sont des zones interdites).

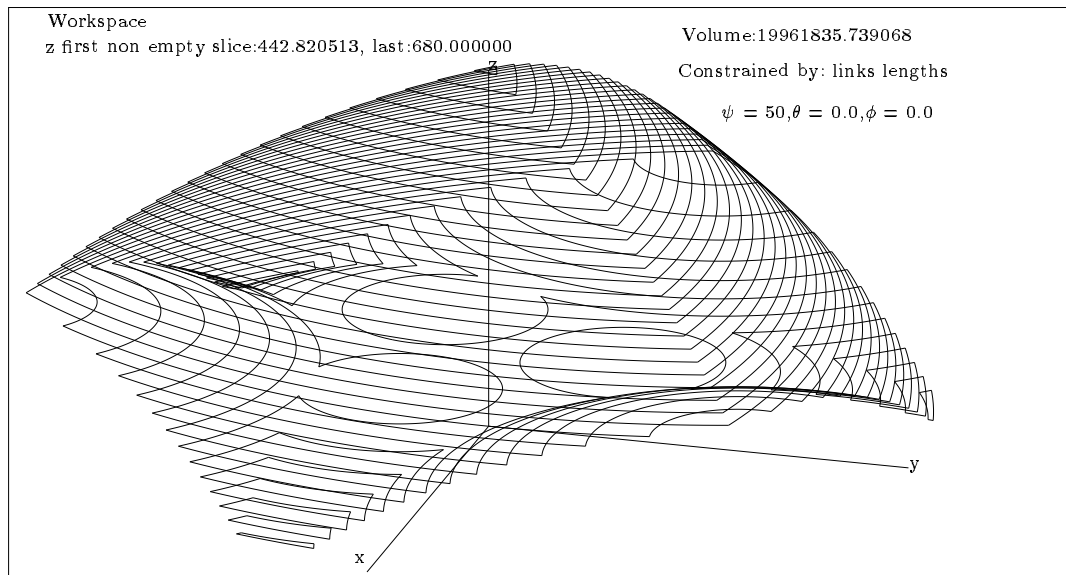


Figure 6.10: Vue perspective de l'espace de travail du prototype, orientation $\psi = 50, \theta = \phi = 0$ (dimensions en mm).

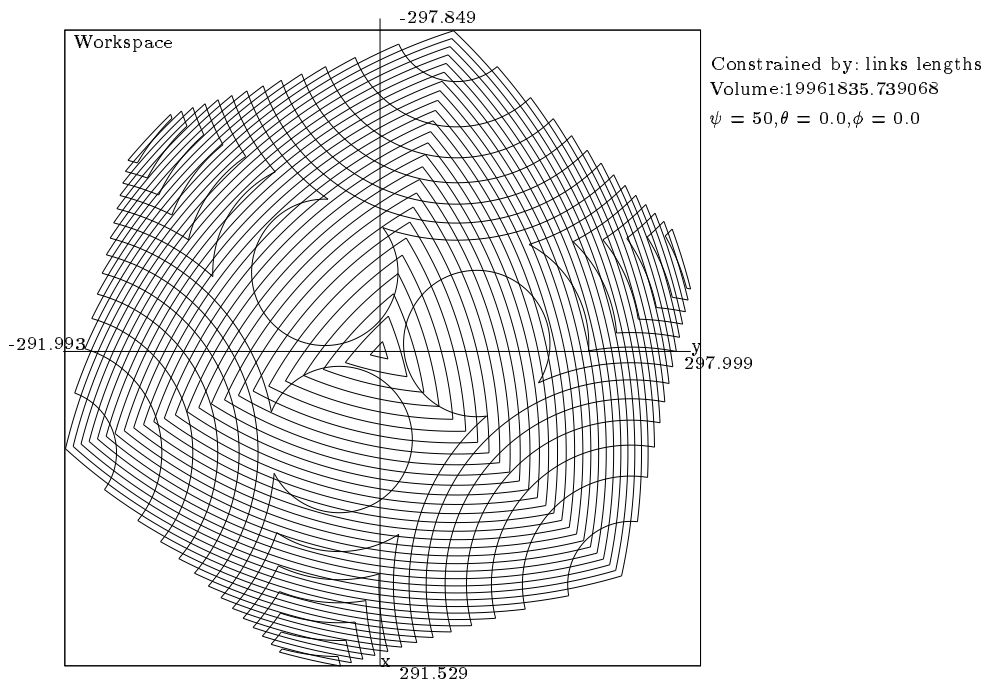


Figure 6.11: Vue de dessus de l'espace de travail du prototype, orientation $\psi = 50, \theta = \phi = 0$ (dimensions en mm).

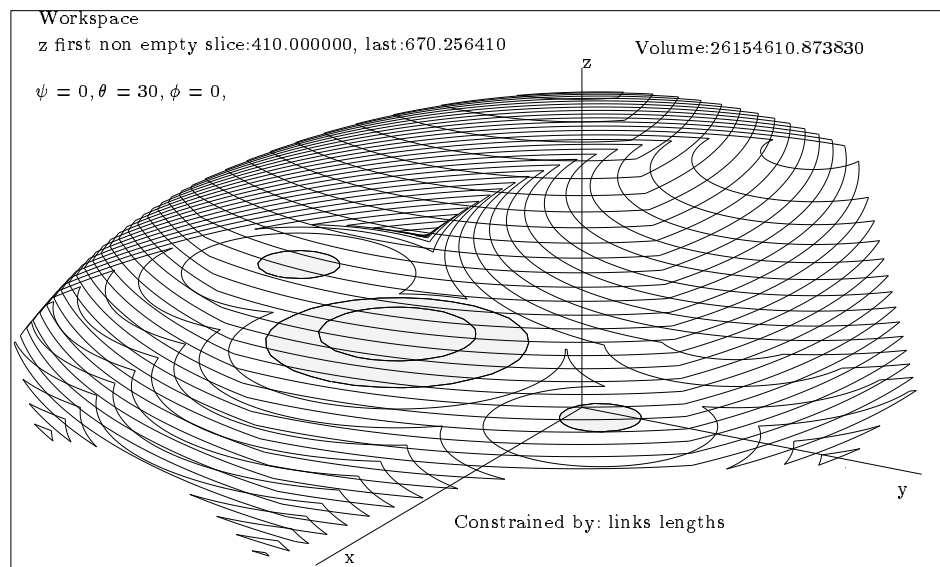


Figure 6.12: Vue perspective de l'espace de travail du prototype, orientation $\psi = 0, \theta = 30, \phi = 0$ (dimensions en mm, les parties grisées sont des zones interdites).

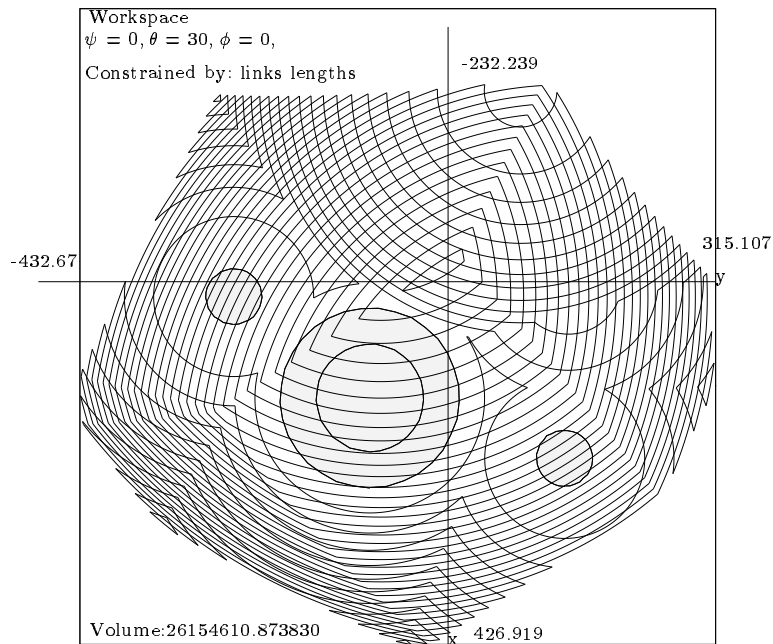


Figure 6.13: Vue de dessus de l'espace de travail du prototype, orientation $\psi = 0, \theta = 30, \phi = 0$ (dimensions en mm, les parties grisées sont des zones interdites).

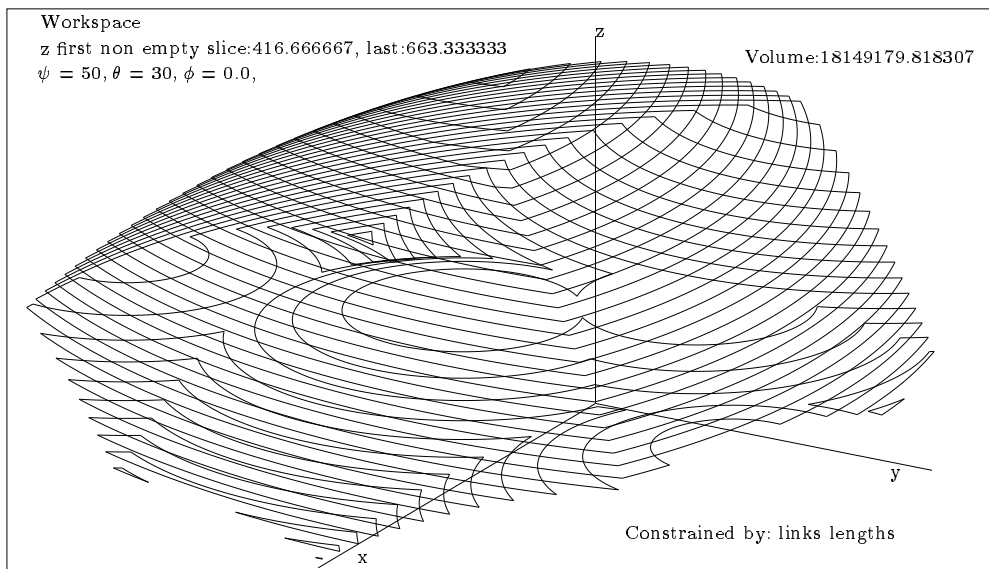


Figure 6.14: Vue perspective de l'espace de travail du prototype, orientation $\psi = 50, \theta = 30, \phi = 0$ (dimensions en mm).

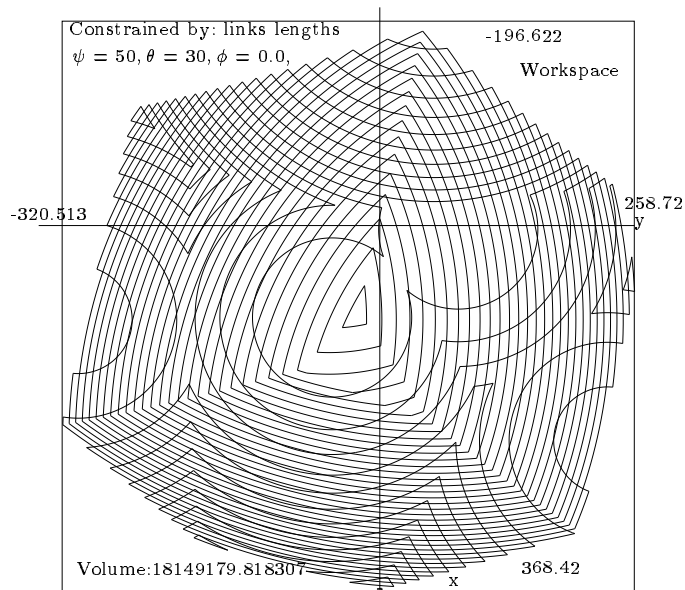


Figure 6.15: Vue de dessus de l'espace de travail du prototype, orientation $\psi = 50, \theta = 30, \phi = 0$ (dimensions en mm).

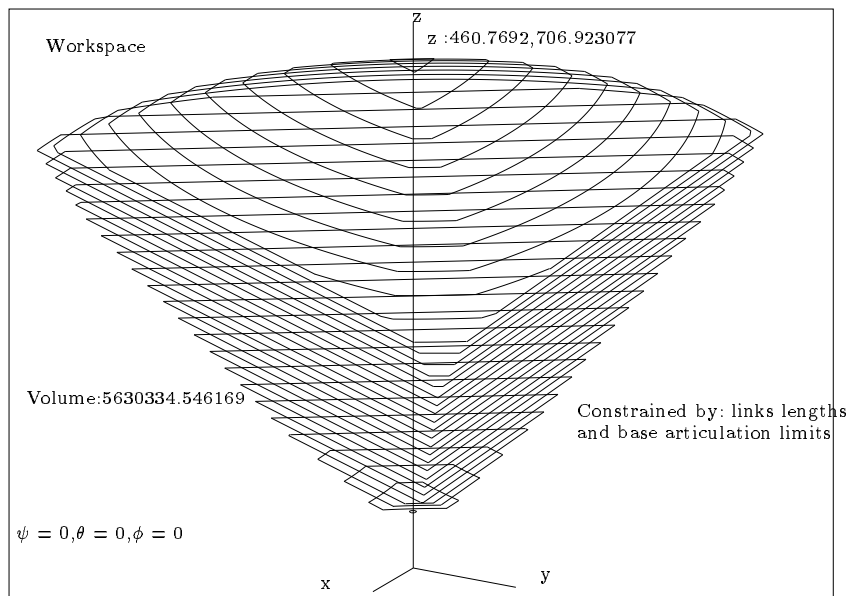


Figure 6.16: Vue perspective de l'espace de travail du prototype, orientation $\psi = \theta = \phi = 0$, contraintes: variables articulaires et 6 pyramides à 4 faces d'angle au centre 30 degré (dimensions en mm).

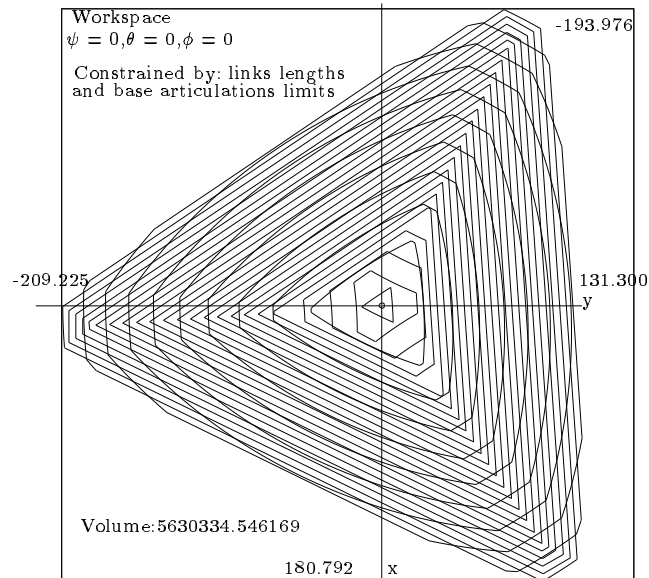


Figure 6.17: Vue de dessus de l'espace de travail du prototype, orientation $\psi = \theta = \phi = 0$, contraintes: variables articulaires et 6 pyramides à 4 faces d'angle au centre 30 degré (dimensions en mm).

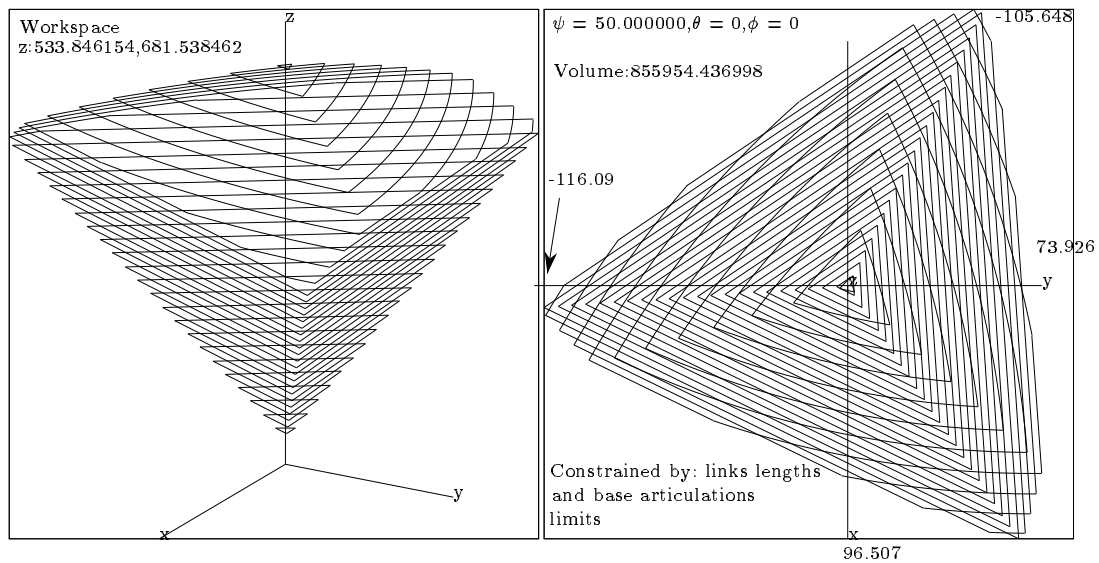


Figure 6.18: Vue perspective et de dessus de l'espace de travail du prototype, orientation $\psi = 50, \theta = \phi = 0$, contraintes: variables articulaires et 6 pyramides à 4 faces d'angle au centre 30 degré (dimensions en mm).

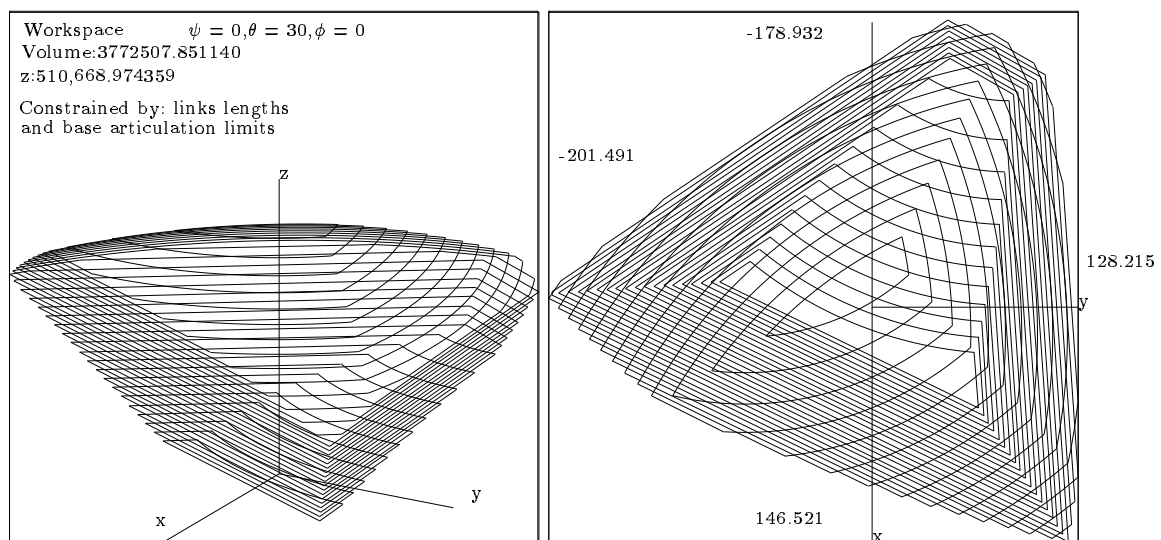


Figure 6.19: Vue perspective et de dessus de l'espace de travail du prototype, orientation $\psi = 0, \theta = 30, \phi = 0$, contraintes: variables articulaires et 6 pyramides à 4 faces d'angle au centre 30 degré (dimensions en mm).

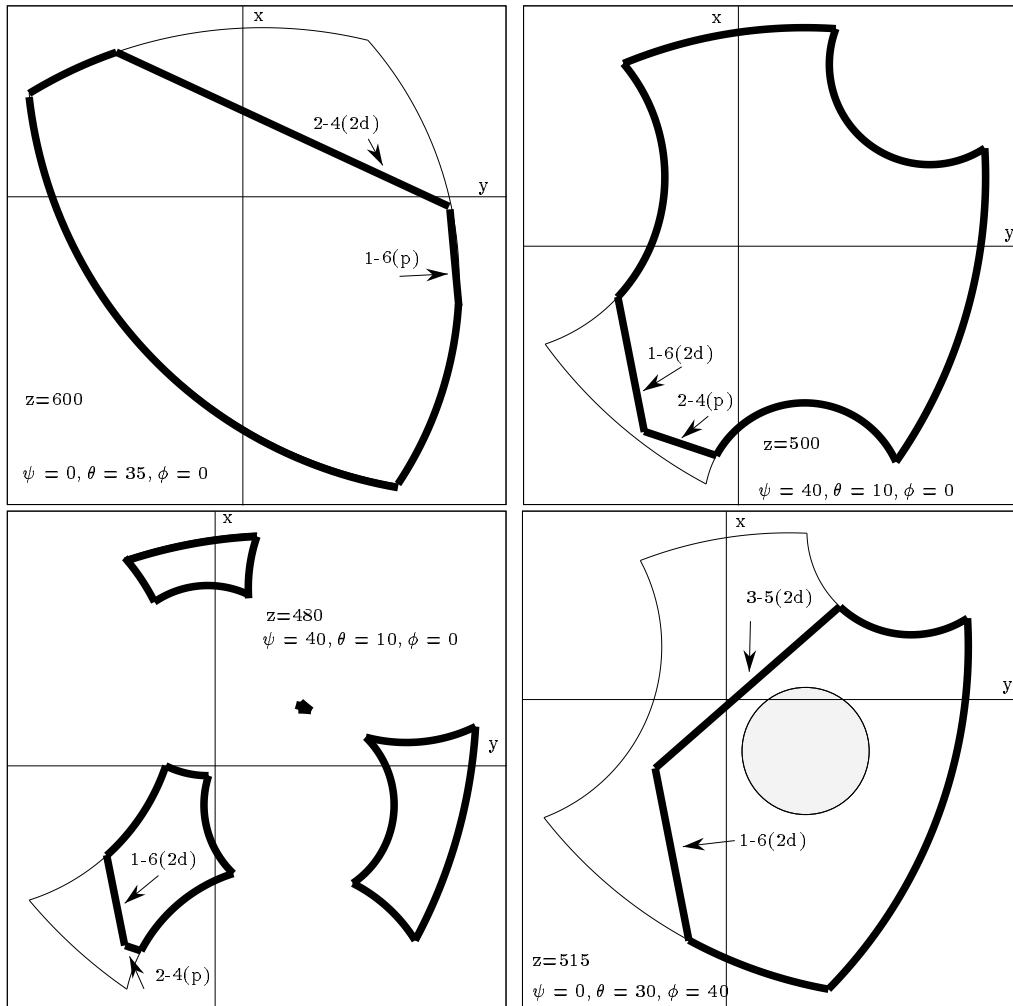


Figure 6.20: Réduction de l'espace de travail en raison des collisions entre segments. La frontière de la zone permise est indiquée en trait épais alors que la zone permise en ne prenant en compte que les limitations articulaires est en traits fins (distance de sécurité: 8mm, dimensions en mm).

Bibliographie

- [1] T. Arai, K. Cleary, et al. Design, analysis and construction of a prototype parallel link manipulator. In *IEEE Int. Workshop on Intelligent Robots and Systems*, 3-6 Juillet 1990.
- [2] M. Baret. Six degrees of freedom large motion system for flight simulators, piloted aircraft environment simulation techniques. In *AGARD Conference Proceeding n°249, Piloted aircraft environment simulation techniques*, pages 22-1/22-7, Bruxelles, 24-27 Avril 1978.
- [3] R. Bricard. Mémoire sur la théorie de l'octaèdre articulé. *Journal de Mathématiques pures et appliquées, Liouville*, tome 3, 1897.
- [4] E.F. Fichter. A Stewart platform based manipulator: general theory and practical construction. *The Int. J. of Robotics Research*, 5(2):157-181, 1986.
- [5] C. Gosselin. *Kinematic analysis optimization and programming of parallel robotic manipulators*. PhD thesis, McGill University, 1988.
- [6] C. Gosselin. Determination of the workspace of 6-dof parallel manipulators. *Trans. of the ASME, J. of Mechanisms Transmissions and Automation in Design*, 1989.
- [7] J-M. Hervé and F. Sparacino. Structural synthesis of parallel robots generating spatial translation. In *ICAR*, pages 808-813, Pise, 19-22 Juin 1991.
- [8] K.H. Hunt. Structural kinematics of in parallel actuated robot arms. *Trans. of the ASME, J. of Mechanisms, Transmissions and Automation in Design*, 105:705-712, Mars 1983.
- [9] H. Inoue, Y. Tsusaka, and T. Fukuizumi. Parallel manipulator. In *Proc. 3^{ème} ISRR*, pages 321-327, Gouvieux, France, 7-11 Octobre 1985.
- [10] K-M Lee and D.K. Shah. Kinematic analysis of a three-degrees-of-freedom in-parallel actuated manipulator. *IEEE J. of Robotics and Automation*, 4(3):354-360, Juin 1988.

- [11] H. Mac Callion and D.T. Pham. The analysis of a six degrees of freedom work station for mechanized assembly. In *Proc. 5^{eme} World Congress on Theory of Machines and Mechanisms*, Montréal, Juillet 1979.
- [12] J-P. Merlet. *Les Robots parallèles*. Hermès, Paris, 1990.
- [13] J-P. Merlet. Force-feedback control of parallel manipulators. In *IEEE Int. Conf. on Robotics and Automation*, Philadelphia, 24-29 Avril 1988.
- [14] J-P. Merlet. Parallel manipulators, Part 1, theory. Research Report 646, INRIA, Mars 1987.
- [15] F. Pierrot. *Robots Pleinement Parallèles Légers : Conception Modélisation et Commande*. PhD thesis, Université Montpellier II, 24 Avril 1991.
- [16] Shamos M.I. Preparata F.P. *Computational geometry, An introduction*. Springer Verlag, 1985.
- [17] C. Reboulet. Hybrid control of a 6 d.o.f. in parallel actuated micro-macro manipulator mounted on a Scara robot. In *3rd ISRAM*, volume 3, pages 293-298, Burnaby, 18-20 Juillet 1990. ASME Press Series.
- [18] C. Reboulet and T. Berthomieu. Dynamic model of a six degree of freedom parallel manipulator. In *ICAR*, Pise, 19-22 Juin 1991.
- [19] D. Stewart. A platform with 6 degrees of freedom. *Proc. of the Institution of mechanical engineers*, 180(Part 1, 15):371-386, 1965-66.
- [20] V.B Zamanov and Z.M. Sotirov. Structures and kinematics of parallel topology manipulating systems. In *Proc. Int. Symp. on Design and Synthesis*, pages 453-458, Tokyo, 11-13 Juillet 1984.

Table des matières

1	Introduction	3
1.1	La structure mécanique d'un robot parallèle	3
1.2	Complexité de représentation de l'espace de travail	4
1.3	Notation	5
2	L'espace de travail limité par les variables articulaires	7
2.1	Utilisation des méthodes de discrétisation	7
2.2	Méthode géométrique pour les contraintes articulaires	8
2.2.1	Contraintes d'évolution sur C	8
2.2.2	Algorithme d'intersection des zones permises pour C	10
3	L'espace de travail limité par les articulations	13
3.1	Limitations mécaniques sur les articulations de base	13
3.1.1	Contraintes sur les articulations de la base	13
3.1.1.1	Pyramide et cône	14
3.1.2	Conséquence pour la position des points B_i	15
3.2	Prise en compte des contraintes articulaires et des contraintes sur les articulations de base	15
3.2.1	Contraintes sur les points B_i	15
3.2.2	Zone permise pour C	15
3.2.3	Algorithme d'intersection des zones permises	17
3.2.3.1	Procédure <code>est-dans-zone-permise</code>	17
3.2.3.2	Procédure <code>trouve-point-marquant</code>	18
3.2.3.3	Procédure <code>ordonne-point-marquant</code>	19
3.2.3.4	Procédure <code>zone-est-seule</code>	25
3.2.3.5	Procédure <code>zone-est-interne</code>	26
3.2.4	Algorithme général d'intersection	26
3.3	Prise en compte des contraintes sur les articulations du plateau mobile	29
3.3.1	Définition des contraintes	29

3.4	Prise en compte des contraintes articulaires et des contraintes sur l'ensemble des articulations	31
4	L'espace de travail limité par les intersections entre segments	32
4.1	Introduction	32
4.2	Prise en compte des intersections entre segments	32
4.2.1	Notion de distance entre segment	32
4.2.2	Lieux limites pour les droites, conique de sécurité	36
4.2.3	Premier cas: la conique de sécurité coupe \mathcal{W}	36
4.2.3.1	Conique des segments	37
4.2.3.2	Points et courbes critiques de la conique de sécurité	39
4.2.3.3	Détermination des courbes critiques	40
4.2.3.4	Utilisation des courbes critiques	41
4.2.4	Deuxième cas: la conique de sécurité ne coupe pas \mathcal{W}	43
4.2.4.1	Conique des points	43
4.2.5	Partage de l'espace de travail	46
4.2.5.1	Zone délimitée par les courbes critiques	46
4.2.5.2	Détermination de la zone de sécurité	47
4.2.6	Exemples	49
4.3	Cas particulier, $d = 0$	52
4.3.1	Intersection entre les droites	52
4.3.2	Intersection entre les segments	53
4.3.3	Notion de segments interdits	57
4.3.4	Cas particuliers	58
4.3.5	Exemples	58
4.4	Surface et volume de l'espace de travail	59
5	Conclusion	60
6	Appendice : Exemples d'espace de travail	62
6.1	Micro-manipulateur	62
6.2	Un prototype pour l'assemblage	67
6.3	Bibliographie	76