

République Algérienne Démocratique et Populaire
Ministère de L'Enseignement Supérieur et de la A Recherche Scientifique

Université Mouloud Mammeri De Tizi-Ouzou



Faculté De Génie Electrique Et D'informatique
DEPARTEMENT D'AUTOMATIQUE

**Mémoire de Fin d'Etude
de MASTER ACADEMIQUE**
Spécialité : **Commande des systèmes**

Présenté par
Souaad LAMARA MOHAMED

Mémoire dirigé par **Mohamed ALI BEY**

Thème

**Génération de trajectoire pour un Robot
Manipulateur Mobile pour tâche
généralisée point à point : Extension de
l'approche des profils aléatoires de
trajectoires.**

Mémoire soutenu publiquement le 11 juillet 2017 devant le jury composé de :

M Ahmed MAIDI
Professeur, UMMTO, Président

M Amar HAMACHE
Maître de conférences, UMMTO, Examineur

M Samir BENBELKACEM
Docteur de recherche, CDTA, Examineur

REMERCIEMENTS

Je tiens à remercier dieu tout puissant de m' avoir donné le courage et la volonté de mener à terme ce présent travail.

Je remercie énormément Mr ALI Bey d' avoir accepté de m'encadrer, pour son aide, son soutien et surtout ses précieux conseils.

*Aussi, je tiens à remercier :
Les membres du jury qui nous ont fait l'insigne honneur
d'examiner le travail.*

Mes remerciements à tous les enseignants et à toute personne ayant contribué de près ou de loin à la réalisation de ce projet.

DEDICACES

*Je dédie ce modeste travail :
A la mémoire de mon Père
A ma chère Mère
A toute ma famille
Avec toute mon affection
Pour tout l'amour qu'ils m'on prodigué*

Sommaire

Introduction Générale.....	1
Chapitre I : Généralités sur la Robotique	
I.1. Introduction.....	3
I.2 Définition d'un robot.....	3
I.3. Composition d'un robot.....	3
I.3.1 Les capteurs.....	4
I.3.2 Les actionneurs.....	4
I.3.3 Unité de traitement (contrôleur).....	5
I.4. Les principaux types de robots.....	5
I.4.1 Les robots mobiles.....	5
I.4.1.1 Types de roues.....	5
I.4.1.2 Holonomie.....	7
I.4.1.3 Types de plateformes mobiles à roues.....	7
I.5. Robots manipulateurs.....	9
I.5.1. Types d'articulations (liaisons).....	10
I.6. Description des Manipulateurs Mobiles.....	11
I.7. Modes de fonctionnement d'un robot.....	13
I.7.1 Mode télé-opérés.	13
I.7.2 Mode autonomes.....	13
I.8. Conclusion.....	14
Chapitre II : Modélisation des manipulateurs mobiles.	
II.1. Introduction.....	15
II.2. Notions fondamentales.....	15
II.2.1. Organe terminal.....	15
II.2.2. Espace articulaire, espace opérationnel.....	15
II.2.3. Degrés de Libertés ddl.....	16
II.2.4. Redondances.....	16
II.3. Non Holonomie.....	17
II.4 Description géométrique.....	17
II.5. Définition des modèles.....	21
II.5.1. Modèle Géométrique Direct.....	21
II.5.2. Modèle Géométrique Inverse.....	21
II.5.3. Modèle Différentiel Direct (MDD) (cinématique).....	21
II.5.4. Modèle Différentiel Inverse (MDI) (cinématique).....	22
II.6. Modélisation des bras manipulateurs.....	22
II.7. Modélisation d'une plateforme mobile à roues.....	23
II.8. Modélisation d'un manipulateur mobile à roues.....	24
II.9. Conclusion.....	25
Chapitre III : Position du problème	
III.1. Introduction.....	26

Sommaire

III.2. Définitions.....	26
III.2.1. Trajectoire et Chemin.....	26
III.2.2 Types de tâches pouvant être assigné à un Manipulateur Mobile.....	27
A) Tâches Généralisées Point-à-Point (GPP).....	27
B) Tâches Opérationnelles Point-à-Point (OPP).....	28
C) Tâches à Chemin Opérationnel Imposé (COI).....	29
III.3. Formulation mathématique du problème.....	29
III.3.1 Contraintes.....	29
III.3.1.1 Contraintes liées au système mécanique.....	29
III.3.1.2. Contraintes liées à la tâche.....	31
III.3.1.3. Contraintes liées à l'environnement.....	32
III.4. Conclusion.....	32
Chapitre IV : Approche des profils aléatoires de trajectoire (RPA)	
IV.1. Introduction.....	33
IV.2. Approche des profils aléatoires de trajectoire (RPA).....	33
IV.2.1. Profil de trajectoire.....	33
IV.2.1.1 Décomposition du profil de trajectoire.....	34
IV.2.2 Modélisation paramétrique d'un profil.....	34
IV.2.2.1 Modélisation paramétrique du chemin et du mouvement.....	35
IV.2.3. Méthodologie pour le calcul de la fonction objective.....	35
IV.2.4. Détermination de la trajectoire optimale.....	36
IV.2.5 Récapitulatif.....	37
IV.3 Adaptation de RPA aux tâches GPP.....	39
IV.4 Conclusion.....	42
Chapitre V : Mise en œuvre : Simulations	
V.1 Introduction.....	43
V.2 Description du simulateur.....	43
V.3 Résultats et description des expérimentations.....	44
V.3.1 Tâche généralisée point à point (GPP).....	44
V.3.1.1 Les démarches du programme.....	44
V.3.1.2 Résultats de simulation.....	45
V.4 Conclusion.....	48
Conclusion Générale.....	50
Annexe A : Techniques stochastiques d'optimisation.....	52
Annexe B : Interpolations, courbes de bézier , β-spline et spline cubique.....	59
Annexe C : Caractéristiques numériques des tâches traités	65
Références Bibliographiques	

Liste des figures

Fig. I.1:	Exemple de Capteurs.....	4
Fig. I.2:	Exemples d'actionneurs.....	4
Fig. I.3:	Différents types de roues en robotique mobile.....	6
Fig. I.4:	Roue troncosphérique, roue suédoise.....	6
Fig. I.5:	Exemple de plate-forme différentielle. Pioneer 2 DX de la société Mobile Robots. Urban Robot de la société iRobot.....	7
Fig. I.6:	Plateformes omnidirectionnelles.....	8
Fig. I.7:	Exemple de plate-forme non holonome de type Ackerman.....	9
Fig. I.8:	Exemples de robots à pattes. Hexapode de AAI Canada, Aibo de Sony	9
Fig. I.9:	Robot à chaîne ouverte simple.....	10
Fig. I.10:	Types de liaisons : (a) liaison rotoïde (b) liaison prismatique.....	11
Fig. I.11:	Le robot humanoïde Qrio.....	11
Fig. I.12:	Deux manipulateurs mobiles sous-marins.....	12
Fig. I.13:	Manipulateur mobile du CDTA.....	13
Fig. II.1:	Exemple explicatif pour la définition d'un degré de liberté : (a) bras à un degré de liberté, (b) bras à deux degrés de libertés	16
Fig. II.2:	Numérotation des corps et des articulations dans une structure arborescente.....	18
Fig. II.3:	Paramètres géométriques pour un corps à plus de deux articulations	20
Fig. II.4:	Paramétrisation d'un bras manipulateur à 2 degrés de liberté.....	23
Fig. II.5:	Paramétrisation d'une plateforme mobile de type différentielle.....	24
Fig. II.6:	Représentation d'un manipulateur mobile plan.....	25
Fig. III.1:	Tâche généralisée point-à-point.....	28
Fig. III.2:	Tâche opérationnelle point-à-point.....	28
Fig. III.3:	Tâche à chemin opérationnel imposé.....	29
Fig. IV.1:	Profil de la trajectoire généralisée selon la méthode RPA.....	34
Fig. IV.2:	Modèles spline du profil de chemin et du profil du mouvement sur ce chemin.....	35
Fig. IV.3:	Portée des calculs effectués par RPA.....	37
Fig. IV.4:	L'organigramme de la recherche d'une trajectoire optimale selon la méthode RPA.....	38
Fig. IV.5:	Représentation discrète de la fonction mouvement.....	39
Fig. IV.6:	Représentation discrète du chemin des paramètres de la plateforme X_p, Y_p et du chemin de θ_p déduit.....	40
Fig. IV.7:	Représentation discrète du chemin de la plateforme.....	40
Fig. IV.8:	Représentation discrète du chemin généralisé du bras pour articulation.....	40
Fig. IV.9:	L'organigramme de l'adaptation de RPA aux tâches GPP.....	41

Liste des figures

Fig. V.1:	Vue de la forme principale.....	43
Fig. V.2:	Résultat obtenu pour une solution initiale.....	45
Fig. V.3:	Résultat obtenu pour une première solution initiale.....	46
Fig. V.4:	Résultat obtenu pour une seconde solution initiale.....	46
Fig. V.5:	Solution optimale.....	47
Fig. V.6:	Solution optimale avec $N_p=4$	48

Introduction Générale.

Introduction Générale

Depuis quelques années, un intérêt croissant est porté au sein de la communauté robotique au développement des plateformes mobiles et des manipulateurs mobiles à roues. Cette classe de robots est caractérisée par une conception mécanique relativement simple, un système de déplacement qui s'adapte à plusieurs types d'environnement et des capacités de locomotion et de manipulation qui peuvent être utilisées de façon combinée ou alternée selon les besoins. Grâce à leurs potentialités, ces systèmes sont devenus l'outil indispensable à l'exécution de tâches complexes, souvent répétitives et contraignantes, réalisées dans de vastes volumes de travail. Les champs d'applications sont naturellement très diversifiés : aide aux handicapés, opérations agricoles, opérations d'inspection et de contrôle, intervention dans des sites contaminés, manipulation de matériaux radioactifs ou toxiques, exécution de missions militaires, exploration, opérations de surveillance.

De nombreux travaux de recherche dans le domaine de la robotique mobile ont été réalisés dans le but d'apporter des solutions à la planification de trajectoires. Diverses méthodes ont été développées dans ce sens. On peut citer les méthodes basées sur le Principe de Maximum de Pontryagin (PMP) [12], les méthodes basées sur la technique dite du plan de phase [5], la méthode des champs de potentiel [6], ainsi que d'autres méthodes plus spécialisées qui, elles, ciblent des cas particuliers. Ces méthodes peuvent donner des solutions satisfaisantes; certaines peuvent même garantir une solution optimale (par exemple, les méthodes basées sur le PMP). Malheureusement, elles ne sont pas toujours applicables. Elles exigent, en général, des développements analytiques préalables qui sont spécifiques au problème traité. De ce fait, la mise en œuvre peut s'avérer pénible ou même irréalisable, surtout lorsque le nombre de degrés de libertés du système augmente.

En conséquence, il existe un réel besoin d'une méthode suffisamment générale pour être applicable à une large catégorie de problèmes et dont la mise en œuvre ne demande pas des efforts exorbitants de développement selon la complexité du problème à traiter. Une telle méthode sera certainement utile en pratique même si, en définitive, elle ne fournirait que des solutions sous optimales.

Introduction Générale

L'approche des profils aléatoires de trajectoires, proposée par Chettibi et Lehtihet [2], tente justement de répondre à ce type de besoin mais dans le contexte plus restreint des bras manipulateurs à base fixe. Cette approche, ci-après appelée RPA (*Random-profile Approach*), n'est pas directement applicable à la robotique mobile, principalement en raison de la difficulté de prise en compte des contraintes non holonomes. Toutefois, elle se distingue par quelques propriétés intéressantes [2]: la versatilité, l'efficacité et la simplicité de mise en œuvre.

Les résultats que nous allons présenter indiquent, d'une part, que RPA s'adapte remarquablement bien à la robotique mobile. L'extension que nous proposons permet effectivement la planification de trajectoires pour manipulateurs mobiles exécutant tâches généralisées point-à-point sous divers types de contraintes (contraintes cinématiques, contraintes non holonomes...). D'autre part, grâce à certaines modifications fondamentales que nous avons apportées à la méthode, nous aboutissons non pas seulement à une extension de l'approche initiale mais à sa généralisation. En effet, par un simple ajustement de quelques paramètres de fonctionnement.

Notre mémoire se subdivisera donc en cinq chapitres : nous exposerons dans le premier les différentes caractéristiques propres à chacun des sous systèmes bras manipulateur et plateforme mobile composant le manipulateur, ainsi qu' une description des manipulateurs mobiles, ensuite, Le deuxième chapitre porte essentiellement sur la modélisation des manipulateurs mobiles, puis le troisième chapitre est consacrée, d'une part, à la formulation mathématique du problème de génération de trajectoires et, d'autre part, à la définition des différentes tâches , le quatrième chapitre sera consacré à la présentation du fond de notre travail , qui porte essentiellement sur l'application de RPA à la génération de trajectoires, en phase d'exécution d'une tâche généralisée point-à-point . Enfin, le dernier chapitre sera consacré aux résultats et simulations de notre travail.

Chapitre I: Généralités sur la robotique.

I.1 Introduction

Les manipulateurs mobiles sont constitués d'un ou plusieurs bras manipulateurs embarqués sur un robot assurant la mobilité. Pour notre part, nous nous sommes intéressés au cas où il n'y a qu'un seul bras manipulateur embarqué sur une plateforme mobile à roues. La spécificité de ce type de systèmes est le fait que les bras ont des caractéristiques particulières, Il en est de même pour les robots mobiles qui se distinguent par leurs propres éléments. De ce fait, nous avons trouvé nécessaire de décrire les deux sous systèmes, ainsi que le système manipulateur mobile.

Ce chapitre a pour but de donner des notions essentielles sur notre système, pour faciliter la compréhension de certains critères. Une première partie sera consacrée à des définitions générales liées à la robotique. Vu que notre robot comporte le sous systèmes mobile à roue, nous décrirons alors certaines notions liées à la robotique mobile, à savoir les roue variées utilisées pour ce genre de systèmes, et les types de véhicules mobiles les plus courants. Nous consacrerons une troisième partie aux bras manipulateurs, qui seront décrits relativement à leurs constituants articulaires. Enfin, la présentation des systèmes manipulateurs mobiles.

I.2 Définition d'un robot

Le Petit Larousse définit un robot comme étant un appareil automatique capable de manipuler des objets, ou d'exécuter des opérations selon un programme fixe ou modifiable. L'Association Française de Normalisation (A.F.N.O.R.) définit également un robot comme étant un système mécanique de type manipulateur commandé en position, reprogrammable, polyvalent (i.e., à usages multiples), à plusieurs degrés de liberté, capable de manipuler des matériaux, des pièces, des outils et des dispositifs spécialisés, au cours de mouvements variables et programmés pour l'exécution d'une variété de tâches. Il a souvent l'apparence d'un, ou plusieurs, bras se terminant par un poignet).

I.3 Composition d'un robot

Un robot est composé de plusieurs parties, capteurs, actionneurs et une unité de traitement (contrôleur), le tout lié sur une base mécanique.

I.3.1 Les Capteurs

Il existe de nombreux types de capteurs, du simple interrupteur, au capteur de distance en passant par les capteurs de température. Ils permettent d'avoir des informations sur le fonctionnement interne du robot (proprioceptives), ainsi que sur l'état de l'environnement (extéroceptifs).



Fig I.1 : Exemple de Capteurs.

I.3.2 Les actionneur

Les actionneurs permettent au robot de réagir suivant un ordre donné. Il existe beaucoup de types d'actionneurs tels que les moteurs électriques par exemple.



Fig. I.2 : Exemples d'actionneurs.

I.3.3 Unité de traitement (contrôleur)

L'unité de traitement permet de traiter les données venant des capteurs, et de commander les actionneurs en conséquence.

Après avoir exposé les différentes définitions relatives aux robots en général, nous allons dans ce qui suit nous intéresser spécialement aux deux parties composant les manipulateurs mobiles, à savoir, les structures articulées (bras manipulateurs), et les plateformes mobiles.

I.4 Les principaux types de robots

I.4.1 Les Robots mobiles

Bien souvent, quand on parle de robotique mobile, on sous entend robots mobiles à roues. Ce sont en effet les systèmes les plus étudiés, parce qu'ils sont plus simples à réaliser que les autres types de robots mobiles, ce qui permet d'en venir plus rapidement à l'étude de leur navigation. Ce type de robots est notamment très souvent utilisé pour l'étude des systèmes autonomes. Vient ensuite la robotique mobile à pattes, avec notamment la robotique humanoïde, mais également des robots avec un nombre de pattes plus élevés qui offrent de bonnes propriétés pour la locomotion en milieu difficile (milieux forestiers et agricoles). La stabilité des mouvements de ce type de robots est en particulier un thème de recherche important [3]. Il existe également de nombreux autres types de robots mobiles (robots marins [7], sous marins, drones volants, micro et nano robots). (Nous n'allons nous intéresser qu'aux robots mobiles à roues). L'étude de ce type de robots se fait généralement dans des thématiques spécifiques avec des problèmes particuliers à l'application visée. La section suivante présente la classification des différents types de roue.

I.4.1.1 Types de roues

La mobilité des robots mobiles étudiée dépend grandement du type de roues utilisées. Dans [10] a présenté une classification des différents types de roues et configurations rencontrées en robotique mobile. Les différents types de roues que l'on rencontre sont :

- la roue fixe : cette roue n'autorise qu'un déplacement dans la direction de son plan médian, l'orientation n'est pas modifiable,
- la roue centrée orientable : elle possède un axe d'orientation en plus de l'axe de rotation, et cet axe d'orientation passe par le centre de la roue,
- la roue décentrée orientable ou roue folle : son axe d'orientation ne passe pas par le centre de la roue (c'est le cas par exemple des roues des chaises de bureau) (voir Fig.I.3).

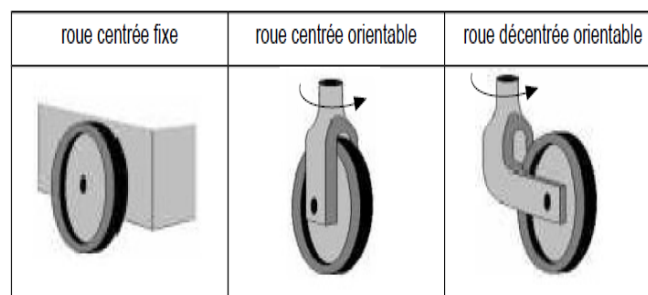


Fig. I.3 :Différents types de roues en robotique mobile.

En plus de ces roues classiques (Fig. I.3), d'autres roues ont été développées pour accroître la mobilité du robot[10]. Elles permettent d'augmenter les capacités de déplacement dans toutes les directions du plan (Fig. I.4) Cependant, elles ne sont commandables que dans certaines de ces directions. Dans cette catégorie nous trouvons notamment :

- les roues suédoises - les roues tronco-sphériques.

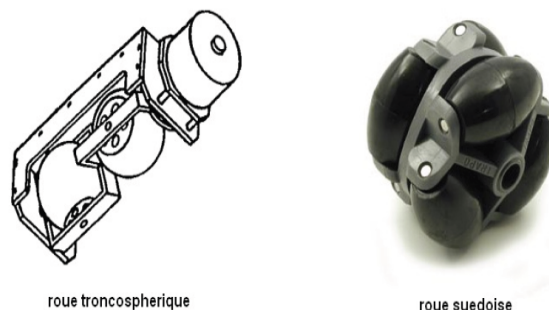


Fig. I.4 : Roue troncospherique,roue suedoise.

En associant les différents types de roues selon une structure mécanique donnée, le robot mobile disposera de plus ou moins de mobilité. Le nombre, le type et la disposition des roues engendrera ou non la contrainte de non holonomie du robot. Cette caractéristique va être explicitée dans ce qui suit.

I.4.1.2 Holonomie

Un robot holonome est un robot capable à chaque instant de se déplacer dans n'importe quelle direction du plan, sans avoir à effectuer une reconfiguration de ses roues. La contrainte non holonome est la limitation des vitesses admissibles d'un objet, ce qui veut dire que le robot non holonome ne peut effectuer de mouvement instantanément que dans certaines directions. Cette notion sera plus approfondie dans le chapitre 2.

I.4.1.3 Types de plateformes mobiles à roues

- Les plates-formes différentielles

Cette structure classique est constituée de deux roues motrices placées sur le même axe, et d'au moins un appui supplémentaire (généralement une ou deux roues folles) (Figure.1.6). L'avantage de cette structure est qu'elle permet au véhicule de tourner sur place, suivant si les vitesses de rotation des deux roues motrices sont de signe opposées ou pas. Ainsi le robot peut pivoter rapidement, ce qui donne des capacités de déplacement intéressantes[4], Cependant, le déplacement latéral n'étant pas directement réalisable, cette structure n'est pas non plus holonome.

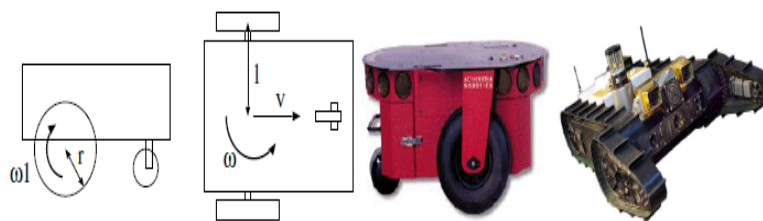


Fig. I.5 : Exemple de plate-forme différentielle. Pioneer 2 DX de la société Mobile Robots. Urban Robot de la société iRobot.

Ce type de plate-forme peut également être utilisé avec des chenilles ce qui fournit une capacité de franchissement de petits obstacles intéressante (Fig.I.6). Ces plates-formes peuvent ainsi être utilisées en extérieur ou dans des décombres.

- **Les plates-formes omnidirectionnelles**

Une plateforme mobile est dite omnidirectionnelle si l'on peut agir indépendamment sur les vitesses : vitesse de translation suivant une direction quelconque dans le plan (x, y) et vitesse de rotation autour de l'axe z [10]. Une plateforme omnidirectionnelle est constituée généralement de trois roues décentrées orientables ou suédoises disposées aux sommets d'un triangle équilatéral (Fig. I.7).



Fig. I.6 : Plate forme omnidirectionnelles.

- **Plateforme mobile de type tricycle (voiture)**

Des plates-formes non holonomes, telles que les voitures, sont également utilisées en robotique mobile (Fig. I.8). C'est plus particulièrement le cas dans le domaine des véhicules intelligents. Ces plates-formes sont toutefois plus difficiles à commander car elles ne peuvent pas tourner sur place et doivent manœuvrer ce qui peut être difficile dans des environnements encombrés.

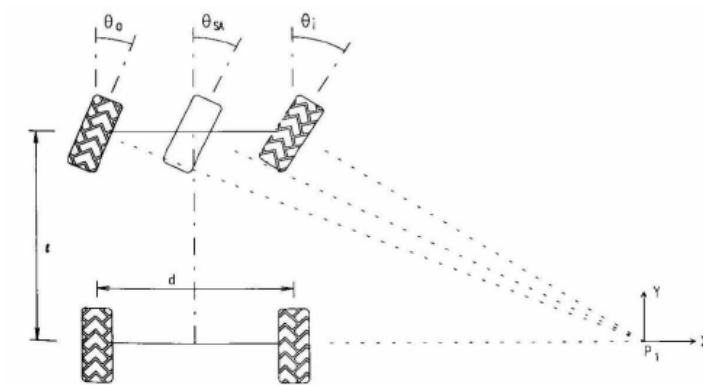


Fig. I.7 : Exemple de plate-forme non holonome de type Ackerman.

Les plates-formes à pattes



Fig. I.8 : Exemples de robots à pattes. Hexapode de AAI Canada, Aibo de Sony.

Des plates-formes à deux, quatre ou six pattes peuvent également être utilisées (Fig.1.9) Elles ont l'avantage théorique de pouvoir se déplacer sur des terrains assez complexes, même si en pratique la plupart de ces plates-formes ne fonctionnent que sur des sols plans.

Nous avons dans ce paragraphe largement décrit les plateformes mobiles à roues par rapport à leurs caractéristiques propres. Nous allons dans ce qui suit évoquer les particularités de l'autre type de robots, les bras manipulateurs.

1.5 Robots manipulateurs

Nous n'allons considérer que les robots manipulateurs ayant une structure ouverte simple (en omettant d'étudier les structures fermées, arborescentes ou parallèles), car c'est ce type de bras qui sera utilisé de notre manipulateur mobile.

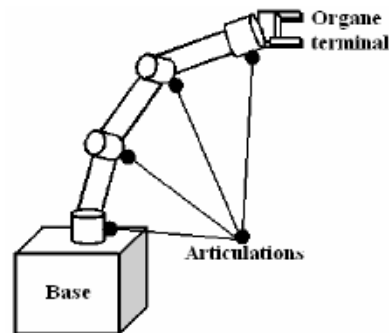


Fig. I.9 : Robot à chaîne ouverte simple.

Un bras manipulateur est un système électromécanique capable d'interagir avec son environnement. Il est constitué de deux parties distinctes :

- **Un organe terminal**

Sous le terme organe terminal, on regroupe tout dispositif destiné à manipuler des objets (dispositifs de serrage, dispositifs magnétiques, à dépression, ...), ou à les transformer (outils, torche de soudage, pistolet de peinture, ...). En d'autres termes, il s'agit d'une interface permettant au robot d'interagir avec son environnement. .. On utilisera indifféremment le terme organe terminal, préhenseur, outil ou effecteur pour nommer le dispositif d'interaction fixé à l'extrémité mobile de la structure mécanique.

- **Un système mécanique articulé (S.M.A.)**

C'est un mécanisme ayant une structure plus ou moins proche de celle du bras humain. Son rôle est d'amener l'*organe terminal* dans une situation (position et orientation) donnée, Son architecture est une chaîne cinématique de corps, généralement rigide, assemblés par des liaisons appelées *articulation* lient deux corps successifs.

I.5.1 Types d'articulations (liaisons)

Articulation *rotoïde* : Il s'agit d'une articulation de type pivot, notée R , réduisant le mouvement entre deux corps à une rotation autour d'un axe qui leur est commun. La situation relative entre les deux corps est donnée par l'angle autour de cet axe (Fig I.11 a).

Articulation *prismatique* : Il s'agit d'une articulation de type glissière, notée P , réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe (Fig I.11 b).

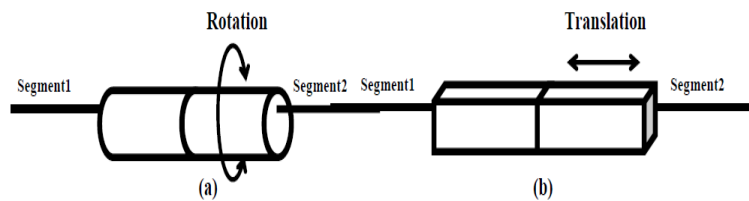


Fig. I.10 : types de liaisons : (a) liaison rotoïde (b) liaison prismatique.

I.6 Description des Manipulateurs Mobiles

Plusieurs types de manipulateurs mobiles existent dans le domaine robotique, nous citons dans ce qui suit les plus communs. Parmi les systèmes existant actuellement nous pouvons citer trois grandes familles.

Les humanoïdes

Les humanoïdes, qui représentent un domaine fascinant, puisqu'ils s'inspirent des êtres humains pour effectuer leurs tâches. Leurs utilisations possibles sont limitées (assistance aux personnes âgées, compagnons domestiques)[11] ou, telle robot QRIO de Sony (Fig I.12).



Fig. I.11 : Le robot humanoïde Qrio.

Les manipulateurs mobiles sous-marins

Les manipulateurs mobiles sous-marins sont aujourd'hui les manipulateurs mobiles les plus utilisés. Ils permettent l'accès à des zones maritimes non accessibles aux plongeurs. Nous pouvons, à titre d'exemple, citer le robot Victor 6000 (fig I.13 (a)) de l'IFREMER, équipé de deux bras manipulateurs et qui peut intervenir à des profondeurs de 6000 mètres ainsi que le robot Achille 2000 (Fig I.13 (b)) de la société Comex, lui aussi équipé de deux bras manipulateurs et dont la profondeur principale d'intervention est de 2000 mètres.



(a) Le robot Victor 6000



(b) Le robot Achille 2000

Fig. I.12 : Deux manipulateurs mobiles sous-marins.

Les manipulateurs mobiles à roues

Les manipulateurs mobiles à roues prennent aujourd'hui des formes diverses tant du point de vue de leur mode de locomotion et le nombre de bras manipulateur(s) dans le système. Les différences majeures entre ces systèmes à roues sont principalement liées au choix du type de locomotion. Ainsi, nous distinguons le manipulateur mobile au niveau du laboratoire de robotique du centre de développement des technologies avancées (CDTA) en Algérie, comportant une plateforme mobile (ROBUTER de Robosoft) non holonome comprenant 24 paires de capteurs à ultrason; quatre roues portant le véhicule dont une paire représentant des roues folles et les autres sont des roues différentielles. Un bras à six liaisons rotoïdes (ULM) se trouve juste au dessus de ces dernières. Il est doté d'un capteur d'effort et d'une caméra se trouvant au niveau de l'organe terminal qui est une pince. L'ordinateur de bord se trouve embarqué sur la plateforme.

Notre étude a été consacré à ce manipulateur mobile (ROBUTER/UML). Cependant, nous avons utilisé dans nos travaux de simulation une plateforme ayant les mêmes

caractéristiques que le robot mobile ROBUTER, sur laquelle nous avons embarqué un bras différent de bras ULM.



Fig. I.13 : Manipulateur mobile du CDTA.

I.7 Modes de fonctionnement d'un robot

Il existe deux principaux modes de fonctionnement pour un robot : télé-opéré et autonome.

I.7.1 Le mode télé-opéré

une personne pilote le robot à distance. Elle donne ses ordres via une interface de commande (joystick, clavier/souris...), et ceux-ci sont envoyés au robot via un lien de communication (internet, satellite ...). D'ailleurs, suivant le niveau de télé-opération, le terme « robotique » est plus ou moins justifié. Le robot doit donc obéir aux ordres de l'opérateur qui perçoit l'environnement autour du robot, par différents moyens (retour d'image, retour haptique...), de manière à donner des ordres adaptés au robot. Dans ce domaine, les efforts de recherche sont beaucoup portés sur les problèmes liés au réseau de télécommunication (retards dans le réseau de communication, problèmes de commande, pertes de données) et sur l'amélioration de la perception de l'environnement par l'opérateur (interfaces haptiques, retours d'efforts).

I.7.2 Le mode autonome

Le robot doit prendre ses propres décisions. Cela signifie qu'il doit être capable à la fois de percevoir correctement son environnement, mais également de savoir comment réagir en conséquence, suivant le niveau d'autonomie. C'est à lui de planifier son parcours et de

déterminer avec quels mouvements il va atteindre son objectif. Les recherches dans ce domaine portent principalement d'une part sur la localisation du véhicule autonome et la cartographie de son environnement, d'autre part sur le contrôle de tels véhicules (structure de contrôle, stratégies de commande, planification).

I.8 Conclusion

Dans ce chapitre, Nous avons présenté certaines notions fondamentales liées à la robotique en général, ensuite pour revenir au vif du sujet (manipulateurs mobiles), nous nous sommes intéressés aux notions relatives aux robots mobiles pour une meilleure compréhension de ce type de systèmes, et nous avons eu à relater les particularités liées aux bras manipulateurs par rapport à leurs constitutions, et enfin, une description des Manipulateurs Mobiles.

**Chapitre II:
Modélisation des
manipulateurs
mobiles.**

II.1 Introduction

Les plateformes mobiles et les manipulateurs mobiles à roues sont des robots à chaîne complexe. Avant de définir plus précisément le problème de planification de trajectoires de ces robots et de développer une approche pour le traiter, il est nécessaire de donner certaines définitions et notations et d'établir les modèles qui seront utiles à la compréhension et à la résolution de ce problème.

II.2 Notions fondamentales

II.2.1 Organe terminal

c'est un dispositif d'interaction fixé à l'extrémité mobile de la structure mécanique, il regroupe les procédés destinés à manipuler des objets, ou à les transformer, il s'agit donc d'une interface permettant au robot d'interagir avec son environnement [1].

II.2.2 Espace articulaire, Espace opérationnel

Définition 1 : on appelle espace de configuration articulaire d'un robot manipulateur (ou simplement configuration) l'état du robot représentant la situation de ses différents corps. Les variables ou coordonnées articulaires sont aussi appelées coordonnées généralisées. L'espace de ces variables noté N de dimension n , est appelé espace articulaire ou espace des configurations.

$$\mathbf{q} = [q_1 \ q_2 \dots \ q_n]^T$$

Définition 2 : l'espace opérationnel est celui dans lequel est représentée la situation de l'organe terminal. Les variables ou coordonnées au nombre minimum permettant de définir la situation de l'OT sont appelées les coordonnées opérationnelles. L'espace opérationnel de dimension m est noté M avec $m \leq 6$.

$$\mathbf{u} = [u_1, u_2, \dots, u_6]^T$$

Pour éviter la confusion entre les grandeurs liées au bras manipulateur et celles liées à la plate-forme mobile, on introduit les indices «_b», «_p» pour désigner leurs grandeurs.

II.2.3 Degrés de Libertés ddl

Le degré de liberté d'un robot manipulateur est égal au nombre de paramètres indépendants qui fixent la situation de l'organe terminal, il peut être fonction de la configuration du robot. Cela sera illustré dans l'exemple suivant en Fig.II.1 :

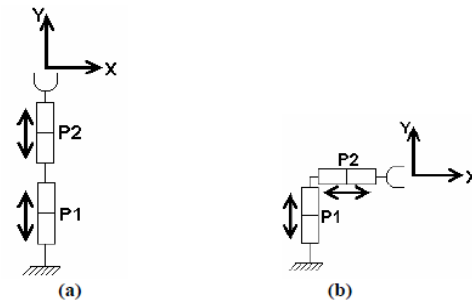


Fig.II.1 : Exemple explicatif pour la définition d'un degré de liberté :
(a) bras à un degré de liberté, (b) bras à deux degrés de libertés.

II.2.4 Redondance

C'est une situation dans laquelle le nombre de ddl de l'organe terminal est inférieur au nombre d'articulations motorisées. Les systèmes redondants ont la possibilité d'effectuer une tâche opérationnelle tout en exécutant une ou plusieurs tâches additionnelles et/ou en respectant une ou plusieurs contraintes comme éviter des obstacles ou s'éloigner des butées articulaires. En contre partie l'exploitation de la redondance complique la recherche d'une solution. Dans le cas des manipulateurs mobiles non holonomes, on distingue deux notions de redondance tirées des deux modèles, MDD et MDDR [5].

La notion de **redondance géométrique** [5] exprime le fait que le nombre de coordonnées généralisées n est strictement supérieur au degré de liberté global Δ de l'OT. L'ordre de redondance géométrique est égal à $n - \Delta$.

La notion de **redondance différentielle** [5] exprime le fait que le degré de mobilité, qui est égal à $(n-1)$, est strictement supérieur à $\Delta' = \max (r'(q))$, avec $r'(q) = \text{rang}(J'(q))$. L'ordre de redondance différentielle est égal à $(n - 1 - \Delta')$.

II.3 Non holonomie

Nous supposons que les contacts roues/sol sont soumis à des forces de frottement sec selon la loi de Coulomb. Les conditions de roulement et de pivotement sans glissement des roues sur le sol doivent être satisfaites, aux points de contacts roues/sol, lors du mouvement de la plateforme. Cette hypothèse se traduit par un système de n_h équations différentielles, non intégrables, exprimées en fonction des variables de configuration de la plateforme et de leurs dérivées par rapport au temps. Ces équations sont appelées contraintes non-holonomes. Elles sont données, dans le cas général, sous la forme matricielle suivante :

$$A(\mathbf{q}_p)\dot{\mathbf{q}}_p = \mathbf{0} \quad (\text{II.1})$$

avec $A(\mathbf{q}_p)$ matrice de dimension $(n_h \times n_p)$ appelée matrice des contraintes non-holonomes.

II.4 Description géométrique

Du point de vue mécanique, une plateforme mobile ou un manipulateur mobile à roues peut être considéré comme étant un système multi-corps à chaîne arborescente. Les roues de la plateforme et l'effecteur du bras manipulateur constituent les organes terminaux de cette chaîne. Afin de modéliser ces systèmes de façon systématique, il est nécessaire d'utiliser une méthode de description géométrique adéquate.

La méthode la plus utilisée pour ce type de systèmes est celle établie par W. Khalil et J. Kleinfinger . Cette méthode, dite de Denavit et Hartenberg Modifiée (DHM), permet une description homogène, avec un nombre minimum de paramètres géométriques, des structures à chaîne simple ou complexe des systèmes mécaniques poly- articulés. Elle est basée sur la décomposition des liaisons complexes du système en des articulations simples de type rotoïde ou prismatique . Par exemple, une liaison sphérique peut être décomposée en trois articulations rotoïdes à axes concourants. Ces articulations sont considérées comme idéales (pas de jeu mécanique, pas d'élasticité).

Soit Σ un système mécanique donné, constitué d'un ensemble de $n + 1$ corps rigides, notés C_j avec $j = 0, \dots, n$, reliés entre eux par n articulations (Fig II.2). Certains de ces corps sont munis d'organes terminaux. La description géométrique de ce système par la

méthode de DHM est basée sur les règles et les conventions suivantes :

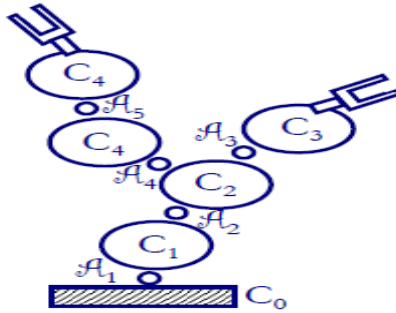


Fig II.2 : Numérotation des corps et des articulations dans une structure arborescente.

A) Conventions de la méthode de DHM

- Le corps C_0 désigne la base du système,
- Le corps C_j est soit réel, soit virtuel. Un corps est réel lorsqu'il a une représentation physique (une masse et une inertie) dans le système étudié. Un corps virtuel est un corps de la chaîne cinématique n'ayant pas d'existence physique dans la structure mécanique du système. Autrement dit, ce corps ne possède ni masse, ni inertie propre. Il sera utilisé dans la décomposition d'une liaison complexe en des liaisons simples,
- Les corps et les articulations sont numérotés de façon croissante lors du parcours d'une chaîne depuis la base vers l'organe terminal de la chaîne considérée (Fig II.2).
- Le corps C_j est articulé via l'articulation \mathcal{A}_j , c'est-à-dire que \mathcal{A}_j relie C_j et son antécédent, $C_{a(j)}$, lors du parcours d'une chaîne depuis la base (Fig II.2).

B) Règles de la méthode de DHM

Afin d'obtenir les modèles géométriques, cinématiques et dynamiques du système, il faut définir le repère lié à chaque corps de la structure. Soit C_i , avec $a(j) = i$, le corps antécédent au corps C_j . Le repère \mathcal{R}_i associé au corps C_i est défini selon les règles suivantes :

- l'axe \mathbf{z}_i est porté par l'axe de l'articulation \mathcal{A}_i ,
- l'axe \mathbf{x}_i est porté par la perpendiculaire commune aux axes \mathbf{z}_i et l'un des axes des articulations des corps successeurs au corps C_i . Si le corps C_i n'a pas d'arborescence, \mathbf{x}_i est porté par la perpendiculaire commune aux axes \mathbf{z}_i et \mathbf{z}_j .
- l'axe \mathbf{y}_i est pris de telle sorte que la base formée par $(\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$ soit orthonormée directe.

Deux cas doivent être envisagés pour définir la matrice ${}^i T_j$ de transformation homogène du repère \mathfrak{R}_j au repère \mathfrak{R}_i , avec $i = a(j)$:

1er Cas : si x_i est la perpendiculaire commune à z_i et z_j alors la transformation de \mathfrak{R}_j à \mathfrak{R}_i s'effectue à l'aide des quatre paramètres géométriques suivants :

- α_j : angle entre z_i et z_j autour de l'axe x_i .
- d_j : distance entre z_i et z_j selon l'axe x_i .
- θ_j : angle entre x_i et x_j autour de l'axe z_j .
- r_j : distance entre x_i et x_j selon l'axe z_j .

La matrice de transformation homogène sera donnée par :

$${}^i T_j = Rot(x_i, \alpha_j) \cdot Trans(x_i, d_j) \cdot Rot(z_j, \theta_j) \cdot Trans(z_j, r_j) \Rightarrow$$

$${}^i T_j = \left[\begin{array}{ccc|c} {}^i R_j & {}^i P_j & & \\ \hline 0 & 0 & 0 & 1 \end{array} \right] \quad (II.2)$$

$$\text{Avec } {}^i R_j = \begin{bmatrix} C\theta_j & -S\theta_j & 0 \\ C\alpha_j S\theta_j & C\alpha_j C\theta_j & -S\alpha_j \\ S\alpha_j S\theta_j & S\alpha_j C\theta_j & C\alpha_j \end{bmatrix} \text{ et } {}^i P_j = \begin{bmatrix} d_j \\ -r_j S\alpha_j \\ r_j C\alpha_j \end{bmatrix}$$

$C.$ et $S.$ désignent respectivement $\cos(.)$ et $\sin(.)$. La matrice ${}^i R_j$ définit l'orientation du repère \mathfrak{R}_j par rapport au repère \mathfrak{R}_i et le vecteur ${}^i P_j$ définit la position de l'origine du repère \mathfrak{R}_j par rapport au repère \mathfrak{R}_i .

2ème Cas : si x_i est la perpendiculaire commune à z_i et à un autre axe z_k (Figure II.3), on construit la perpendiculaire commune u_j aux axes z_i et z_j . La transformation de \mathfrak{R}_j à \mathfrak{R}_i

s'effectue à l'aide des six paramètres suivants:

- γ_j : angle entre x_i et u_j autour de l'axe z_i .
- b_j : distance entre x_i et u_j selon l'axe z_i .
- α_j : angle entre z_i et z_j autour de l'axe u_j .
- d_j : distance entre z_i et z_j selon l'axe u_j .

- θ_j : angle entre u_j et x_j autour de l'axe z_j .
- r_j : distance entre u_j et x_j selon l'axe z_j .

Ces paramètres permettent de définir la matrice ${}^i T_j$ de transformation entre le repère \mathcal{R}_i et le repère \mathcal{R}_j de la manière suivante :

$${}^i T_j = Rot(z_i, \gamma_j) \cdot Trans(z_i, b_j) \cdot Rot(u_j, \alpha_j) \cdot Trans(u_j, d_j) \cdot Rot(z_j, \theta_j) \cdot Trans(z_j, r_j) \Rightarrow$$

$${}^i T_j = \begin{bmatrix} C\gamma_j C\theta_j - S\gamma_j C\alpha_j S\theta_j & -C\gamma_j S\theta_j - S\gamma_j C\alpha_j C\theta_j & S\gamma_j S\alpha_j & d_j C\gamma_j + r_j S\gamma_j S\alpha_j \\ S\gamma_j C\theta_j - C\gamma_j C\alpha_j S\theta_j & -S\gamma_j S\theta_j + C\gamma_j C\alpha_j C\theta_j & -C\gamma_j S\alpha_j & d_j S\gamma_j - r_j C\gamma_j S\alpha_j \\ S\alpha_j S\theta_j & S\alpha_j C\theta_j & C\alpha_j & r_j C\alpha_j + b_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (II.2)$$

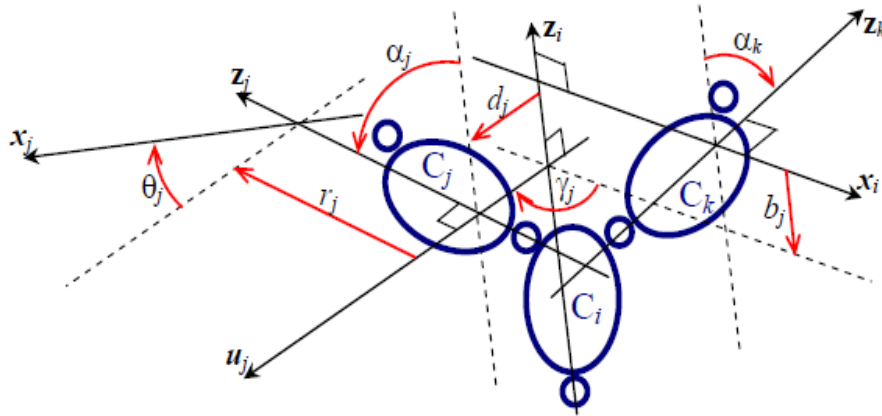


Fig II.3 : Paramètres géométriques pour un corps à plus de deux articulations.

De manière classique, la configuration du système Σ est connue quand la position de tous ses points dans \mathcal{R}_0 est connue . Dans la méthode de DHM, cette configuration est définie par le vecteur q des coordonnées généralisées composé de n variables articulaires q_j pour $j = 1$ à n . La variable articulaire q_j associée à l'articulation \mathcal{A}_j est définie par :

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j \text{ avec } \bar{\sigma}_j = 1 - \sigma_j \quad (II.3)$$

Où :

- $\sigma = 0$ si l'articulation \mathcal{A}_j est rotoïde,
- $\sigma = 1$ si l'articulation \mathcal{A}_j est prismatique.

Par convention, on prend $\sigma = 2$ pour une articulation bloquée. Dans ce cas les repères \mathcal{R}_i et \mathcal{R}_j sont liés au même corps C_i .

II.5 Définition des modèles

II.5.1 Modèle géométrique direct

Le Modèle Géométrique Direct (MGD) est l'ensemble des relations permettant d'exprimer la situation de l'organe terminal OT en fonction de la configuration du système, c'est-à-dire les coordonnées opérationnelles décrites par le vecteur $U=[U_1 \ U_2 \dots \ U_m]^T$ en fonction des coordonnées généralisées exprimées à travers le vecteur $q=[q_1 \ q_2 \dots \ q_n]^T$. Ce modèle est défini par la relation suivante :

$$\begin{aligned} f: N &\rightarrow M \\ q &\mapsto U = f(q) \end{aligned} \quad (\text{II.4})$$

II.5.2. Modèle Géométrique Inverse

Le Modèle Géométrique Inverse (MGI) est l'ensemble des relations inverses à celles du modèle géométrique direct. Ce modèle permet d'exprimer les coordonnées généralisées en fonction des coordonnées opérationnelles U .

$$\begin{aligned} f^{-1}: M &\rightarrow N \\ U &\mapsto q = f^{-1}(U) \end{aligned} \quad (\text{II.5})$$

II.5.3. Modèle Différentiel Direct (MDD) (cinématique)

C'est le modèle qui permet d'exprimer la différentielle de la situation de l'OT en fonction de la différentielle de la configuration du système étudié (les différentielles des coordonnées opérationnelles en fonction des différentielles des coordonnées généralisées). Le MDD est une application linéaire $J(q)$ qui est la matrice jacobienne du système telle que :

$$dU = J(q).dq \quad (\text{II.6})$$

$$\text{avec } \begin{bmatrix} \dot{U}_1 \\ \vdots \\ \dot{U}_m \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \dots & \frac{\partial f_1}{\partial q_n} \\ \vdots & \dots & \vdots \\ \frac{\partial f_m}{\partial q_1} & \dots & \frac{\partial f_m}{\partial q_n} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_n \end{bmatrix} \quad (\text{II.7})$$

II.5.4. Modèle Différentiel Inverse (MDI) (cinématique)

Un Modèle Différentiel Inverse (MDI) du système est une application linéaire inverse généralisée de $J(q)$. On peut avoir des difficultés à inverser ce modèle dans le cas où la matrice jacobienne J ne présente pas un rang plein (matrices carré non singulière).

II.6 Modélisation des bras manipulateurs

Pour un manipulateur quelconque constitué de n_b corps rigides reliés en série par n_b liaisons rotoïdes et/ou prismatiques, nous avons, dans un vecteur q_b à n_b coordonnées indépendantes.

Dans notre travail on a utilisé un bras de deux degré de liberté comme c'est illustré en (Fig.II.4). Cette figure représente les différents paramètres nécessaires à la description du mouvement du bras manipulateur, composé de deux corps rigides de longueurs respectives a_1 et a_2 (non nulles) et comportant deux liaisons rotoïdes.

Les coordonnées généralisées sont représentées dans le vecteur de configuration $q = [q_{b1} \ q_{b2}]^T$ (sachant que q_{b1} et q_{b2} sont les angles de rotation des deux liaisons). La dimension de l'espace généralisé est égale au nombre degré de liberté du système $n_b = 2$.

Le vecteur des coordonnées opérationnelles $U = [U_1 \ U_2 \ U_3]^T = [X_{OT} \ Y_{OT} \ \theta_{ot}]^T$ représente les coordonnées en position X_{OT} et Y_{OT} du point O_{B2} dans le repère de la base du bras $R_{B0} (O_{B0}, \vec{x}_{B0}, \vec{y}_{B0})$ ainsi que l'orientation θ_{ot} du second corps dans le même repère ; la dimension de cet espace opérationnel est égale à $\Delta=3$, car c'est le nombre de coordonnées opérationnelles du système [1].

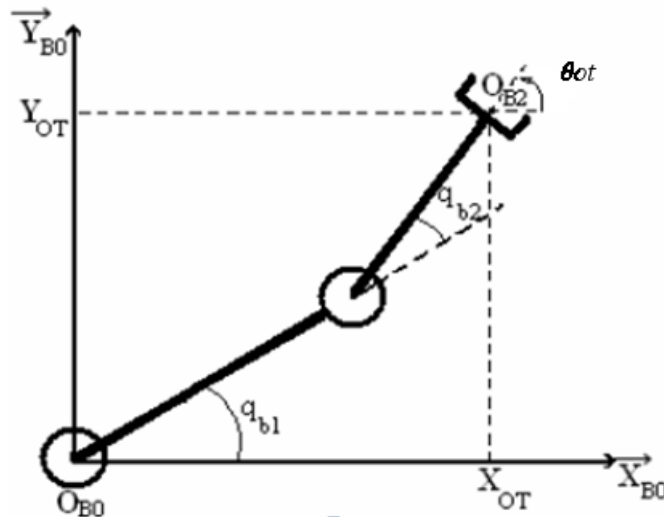


Fig.II.4 : paramétrisation d'un bras manipulateur à 2 degrés de liberté.

II.7 Modélisation d'une plateforme mobile à roues

Considérons une plateforme mobile à roues évoluant dans un environnement plan, La figure II.5 montre le repère lié à la plate-forme mobile et les coordonnées opérationnelles qui lui sont associées, trois coordonnées opérationnelles, deux pour la position et une pour l'orientation. Elle est donc définie sur un espace de dimension Δ égale à 3.

$$U_p = [x_p \ y_p \ \theta_p]^T \quad (\text{II.8})$$

où x_p et y_p sont, respectivement, l'abscisse et l'ordonnée du point O_p dans R_A et θ_p est l'angle d'orientation entre \vec{x} et \vec{x}_p .

Les coordonnées généralisées pour une plateforme à roue sont alors complètement décrites, sur un espace E_p , par un vecteur de dimension n_p dépendant du nombre et du type de roues considérées.

Soit ϕ_1, ϕ_2 les deux roues droite et gauche de notre plateforme mobile, avec $\phi = [\phi_1, \phi_2]^T$. La configuration d'une plateforme mobile à roues est connue lorsque sa situation dans le repère R_A est connue, et que la configuration de chacune de ses roues est connue, donc la configuration du système mobile est alors :

$$q_p = [\phi^T, U_p^T]^T \quad (II.9)$$

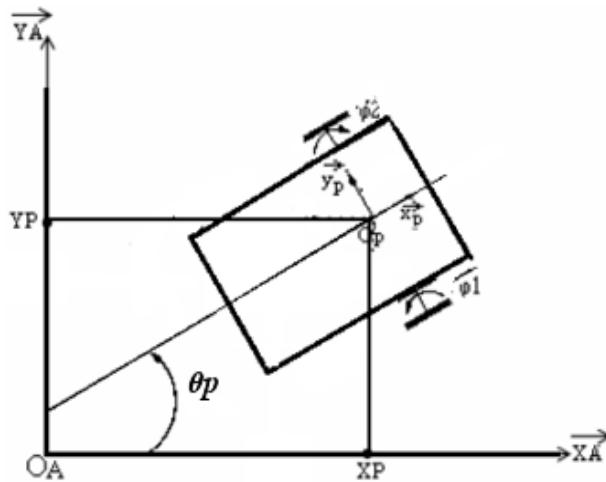


Fig.II.5 : Paramétrisation d'une plateforme mobile de type différentielle.

Il n'existe pas de modèle géométrique des robots mobiles, il existe cependant un modèle cinématique reliant les vitesses des roues aux coordonnées opérationnelles de ces systèmes. vitesses des roues en fonction des vitesses en situation:

r=rayon de la roue; R: distance entre l'axe central de la plateforme et l'axe de la roue:

$$V_{Rdroite} = 1/r(\sin(\theta_p) + \cos(\theta_p)) \dot{x}_p + 1/r(\sin(\theta_p) + \cos(\theta_p)) \dot{y}_p + R/r \quad (II.10)$$

$$V_{RGauche} = 1/r(\sin(\theta_p) + \cos(\theta_p)) \dot{x}_p + 1/r(\sin(\theta_p) + \cos(\theta_p)) \dot{y}_p - R/r \quad (II.11)$$

II.8 Modélisation d'un manipulateur mobile à roues

Nous considérons le cas général d'un manipulateur mobile constitué d'un bras manipulateur à chaîne continue ouverte simple embarqué sur une plateforme mobile à roues. La configuration d'un manipulateur mobile est connue lorsque celle de la plateforme et celle du bras manipulateur sont connues. Elle est définie sur un espace de dimension $n = n_p + n_b$. (n_b est la dimension de l'espace des configurations du bras et n_p la dimension de l'espace des configurations de la plateforme) par un vecteur q tel que :

$$q = [q_p \quad q_b]^T \quad (II.12)$$

Comme pour les bras manipulateurs, la situation d'un organe terminal pour un manipulateur mobile est égale au minimum à 6, dans un espace tridimensionnel.

Le manipulateur mobile auquel nous nous intéressons est présenté dans la figure Fig.II.6

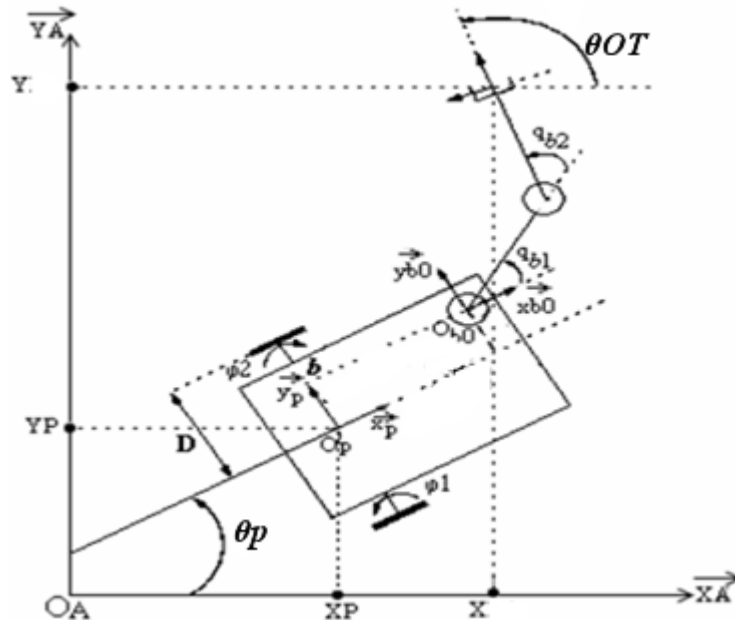


Fig. II.6 : Représentation d'un manipulateur mobile plan.

Dans le cadre de notre étude, le manipulateur mobile se présente comme plan, et évoluant dans un espace à deux dimensions, nous décrivons la situation de l' OT comme :

$U = [U_1 \ U_2 \ U_3]^T = [X_{ot} \ Y_{ot} \ \theta_{ot}]^T$, ces coordonnées opérationnelles seront toujours considérées dans $R_A = (O_A, \vec{x}_A, \vec{y}_A)$, elles spécifient les deux coordonnées cartésiennes ainsi que l'orientation de l' OT avec $\Delta=3$ représentant la dimension de l'espace opérationnel.

En considérant les coordonnées généralisées $q_b = [q_{b1} \ q_{b2}]^T$ du bras manipulateur, et nous ne nous intéressons qu'à la situation $q_p = [X_p \ Y_p \ \theta_p]^T$ de la plateforme, alors le degrés de liberté du système considéré est $n=2+2=4$.

II.9 Conclusion

Nous avons présenté dans ce chapitre des notions générales concernant les robots manipulateurs et les plateformes mobile. Les modèles géométriques et différentiels (cinématiques) du bras manipulateur utilisé ainsi que celui du système mobile ROBUTER ont été aussi définis.

Chapitre III: Position du problème.

III.1 Introduction

Les manipulateurs mobiles à roues sont souvent destinés à l'exécution de tâches complexes dans de vastes espaces de travail. Afin d'améliorer l'efficacité d'un manipulateur mobile lors de l'exécution d'une tâche donnée, il est nécessaire de déterminer la meilleure trajectoire permettant d'exécuter cette tâche. La recherche de cette trajectoire pourrait être conditionnée par la satisfaction d'un critère de performance et le respect d'un certain nombre de contraintes. Ce problème peut être défini comme un problème d'optimisation non linéaire sous contraintes. Ce chapitre est consacré à la formulation mathématique des tâches pouvant être assignés à un manipulateur mobile, aux critères de performance traités ainsi qu'aux contraintes existantes concernant les robots étudiés pour une meilleure compréhension de la problématique.

III.2 Définitions

III.2.1 Trajectoire et Chemin

Dans la littérature il existe certaines confusions entre génération de trajectoires, génération de chemin et génération de mouvement. En réalité, ce sont trois problèmes distincts. Il est important de définir avec précision les termes trajectoire, chemin et mouvement.

Une trajectoire généralisée $q(t)$ (ou opérationnelle $U_E(t)$, selon que la trajectoire est définie dans l'espace généralisé N ou dans l'espace opérationnel M) est une séquence continue de configurations (ou de situations), définie en fonction du temps t , reliant une configuration initiale q^{int} (ou une situation initiale U^{int} à l'instant $t = 0$) à une configuration finale q^{fin} (ou une situation finale U^{Fin}) à l'instant $t = T$, où T est la durée de la trajectoire.

Trajectoire généralisée :

$$\begin{aligned} q : [0, T] &\rightarrow N \\ t &\rightarrow q = q(t) \end{aligned} \quad (III.1)$$

Trajectoire opérationnelle :

$$\begin{aligned} U : [0, T] &\rightarrow M \\ &\rightarrow U_E = U_E(t) \end{aligned} \quad (III.2)$$

Par analogie avec la définition d'une trajectoire, un chemin généralisé $P(\lambda)$ (ou opérationnel $\chi(\lambda)$) est une séquence continue de configurations (ou de situations), définie

indépendamment du temps en fonction d'un paramètre λ , reliant une configuration initiale \mathbf{q}^{int} (ou une situation initiale \mathbf{U}^{int}) à une configuration finale \mathbf{q}^{fin} (ou une situation finale $\mathbf{U}_E^{\text{fin}}$). Généralement ce paramètre λ est normalisé ($\lambda \in [0, 1]$).

Chemin généralisé :

$$\begin{aligned} \mathbf{P} : [0, 1] &\rightarrow \mathbf{N} \\ \lambda &\rightarrow \mathbf{q} = \mathbf{P}(\lambda) \end{aligned} \quad (\text{III.3})$$

Chemin opérationnel :

$$\begin{aligned} \chi : [0, 1] &\rightarrow \mathbf{M} \\ \lambda &\rightarrow \mathbf{U}_E = \chi(\lambda) \end{aligned} \quad (\text{III.4})$$

Remarque : On peut toujours construire une trajectoire à partir de la donnée d'un chemin et d'un mouvement sur ce chemin. La fonction mouvement est une fonction scalaire $\lambda(t)$ croissante qui permet de définir la manière selon laquelle on parcourt le chemin dans l'espace temporel. Cette fonction est définie par la relation suivante :

$$\begin{aligned} \lambda : [0, T] &\rightarrow [0, 1] \\ t &\rightarrow \lambda = \lambda(t) \end{aligned} \quad (\text{III.5})$$

III.2.2 Types de tâches pouvant être assigné à un Manipulateur Mobile [8]

Parmi les tâches auxquelles les manipulateurs mobiles sont dédiés, nous pouvons distinguer trois catégories principales :

- A) **Tâches Généralisées Point-à-Point (GPP) :** Cette catégorie concerne les tâches définies par une trajectoire libre d'une configuration initiale \mathbf{q}^{int} à une configuration finale \mathbf{q}^{fin} (Fig. III.1). C'est le cas notamment des tâches de transfert d'objets, de prise et de positionnement de pièces, de soudure point-à-point, d'opérations d'assemblage, etc., où les conditions aux limites en position sont définies dans l'espace généralisé.

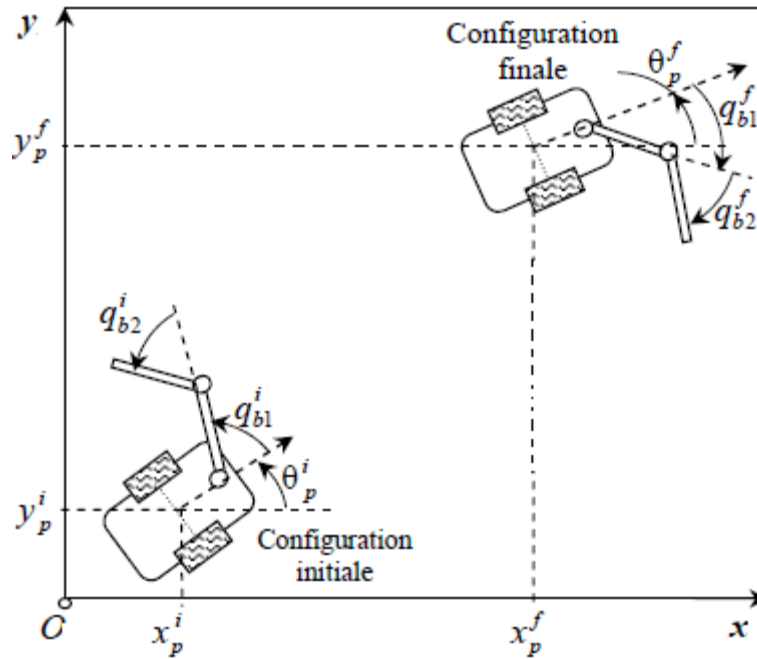


Fig.III.1 : Tâche généralisée point-à-point.

B) **Tâches Opérationnelles Point-à-Point (OPP)** : Ces tâches sont définies par une trajectoire libre d'une situation initiale U_E^{int} à une situation finale U_E^{fin} . Ce sont les mêmes tâches citées précédemment mais les conditions aux limites en position sont définies dans l'espace opérationnel. Dans la pratique, on connaît généralement non seulement U_E^{int} mais aussi la configuration q^{int} correspondante. Par conséquent, il est plus intéressant de définir les tâches OPP en considérant q^{int} et U_E^{fin} imposées (Fig.III.2) plutôt que de considérer U_E^{int} et U_E^{fin} [8].

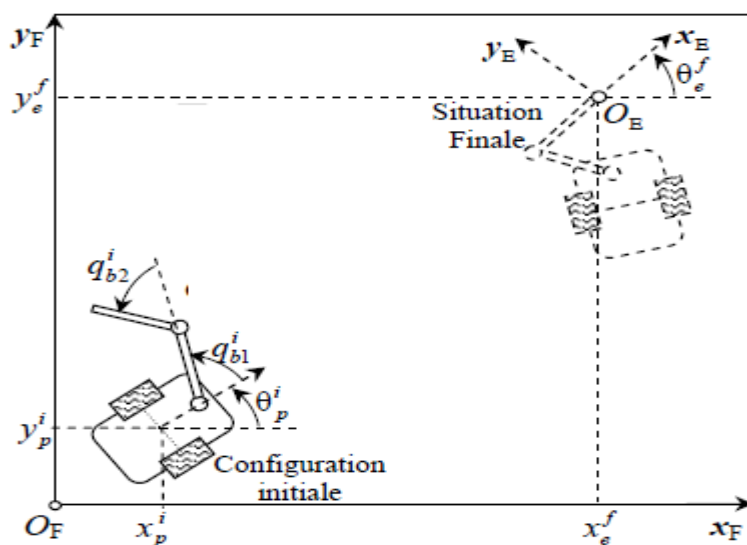


Fig.III.2 : Tâche opérationnelle point-à-point.

- C) **Tâches à Chemin Opérationnel Imposé (COI)** : Cette catégorie concerne les tâches nécessitant la poursuite d'un chemin opérationnel imposé $\chi(\lambda)$ (Fig.III.3). C'est le cas, par exemple des tâches de soudage à l'arc, de contrôle, de collage, de découpage, ...

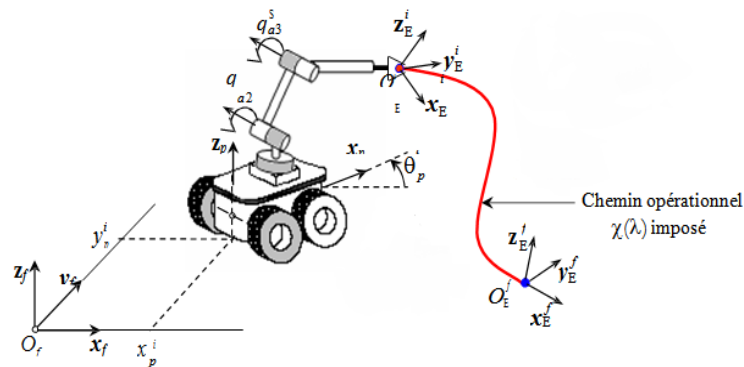


Fig.III.3 : Tâche à chemin opérationnel imposé.

III.3 Formulation mathématique du problème

Pour un manipulateur mobile à roues évoluant dans un environnement plan et réalisant une tâche GPP, OPP ou COI, le problème de génération de trajectoires peut être formulé sous forme d'un problème d'optimisation sous contraintes : déterminer la trajectoire généralisée $q(t)$, la durée T de cette trajectoire, tout en respectant les contraintes imposées.

III.3.1 Contraintes

Pour que la trajectoire soit réalisable, il est nécessaire que l'on tienne compte des contraintes imposées. Elles sont de diverses natures et elles peuvent être classées en trois catégories : les contraintes liées au système mécanique étudié, les contraintes liées à la tâche à réaliser, et les contraintes liées à l'état de l'environnement dans lequel évolue le système. Ces contraintes se traduisent souvent par une restriction de l'espace de recherche des solutions.

III.3.1.1 Contraintes liées au système mécanique

En plus des contraintes non holonomes, un manipulateur mobile est soumis aux différentes contraintes suivantes: les contraintes liées aux débattements articulaires, les

contraintes liées aux capacités cinématiques (vitesses et accélérations maximales admissibles). Il existe également des contraintes liées aux capacités dynamiques et la contrainte de stabilité dynamique que nous n'allons pas évoquer dans ce manuscrit. Il, existe également la contrainte d'évitement d'obstacles. Nous allons dans ce qui suit définir mathématiquement ces contraintes.

A) Contraintes liées aux débattements articulaires du manipulateur mobile

Les articulations d'un système mécanique sont conçues pour permettre une certaine liberté entre les solides rassemblés par cette liaison. En général, les débattements articulaires des corps sont limités par des butées mécaniques ou même par la conception de la liaison elle-même. Par conséquent, toute trajectoire engendrant des dépassements des capacités tolérées d'une articulation risque d'endommager celle-ci. De ce fait, les trajectoires générées doivent obligatoirement respecter ces contraintes. La formulation explicite de ces contraintes sur les débattements articulaires est donnée par :

$$\forall t \in [0, T] \quad |q_i(t)| \leq |q_i^{\max}| \quad \text{avec } i = 1 \dots n \quad (\text{III.6})$$

La valeur q_i^{\max} est une donnée intrinsèque au robot. Elle constitue la limite maximale de la position articulaire permise pour la $i^{\text{ème}}$ articulation.

B) Contraintes liées aux capacités cinématiques du manipulateur mobile

De même pour les caractéristiques cinématiques des mouvements articulaires, les vitesses et les accélérations permises sont bornées. Ces limitations sont dues principalement à la conception des liaisons et à la technologie des actionneurs motorisant les liaisons actives. Ces contraintes s'écrivent de la manière suivante :

- Pour les vitesses :

$$\forall t \in [0, T] \quad |\dot{q}_i(t)| \leq |\dot{q}_i^{\max}| \quad \text{avec } i = 1 \dots n \quad (\text{III.7})$$

- Pour les accélérations :

$$\forall t \in [0, T] \quad |\ddot{q}_i(t)| \leq |\ddot{q}_i^{\max}| \quad \text{avec } i = 1 \dots n \quad (\text{III.8})$$

Les valeurs limites \dot{q}_i^{\max} et \ddot{q}_i^{\max} peuvent être imposées par les caractéristiques techniques intrinsèques au robot.

III.3.1.2 Contraintes liées à la tâche

Ces contraintes sont définies par les conditions aux limites en positions et en vitesse et par les exigences liées à la tâche. En ce qui concerne les conditions aux limites en vitesse, elles sont les mêmes pour tous les types de tâches cités précédemment. Ces conditions sont exprimées par la relation suivante :

$$\dot{q}(t=0) = 0 \quad \text{et} \quad \dot{q}(t=T) = 0 \quad (\text{III.9})$$

Par contre, les autres contraintes varient selon la catégorie de la tâche à réaliser.

A) **Tâches GPP** : La réalisation d'une tâche GPP, entre une configuration initiale q^{int} et une configuration finale q^{fin} , est soumise aux conditions aux limites en positions suivantes :

$$q(t=0) = q^{\text{int}} \quad \text{et} \quad q(t=T) = q^{\text{fin}} \quad (\text{III.9a})$$

B) **Tâches OPP** : La réalisation d'une tâche OPP, entre une configuration initiale q^{int} et une situation finale U_E^{fin} , est soumise aux conditions aux limites en positions données par les relations suivantes :

$$q(t=0) = q^{\text{int}} \quad \text{et} \quad U_E(t=T) = U_E^{\text{fin}} \quad (\text{III.9b})$$

C) **Tâches COI**: La réalisation d'une tâche COI le long d'un chemin $\chi(\lambda)$, d'une configuration initiale q^{int} et une situation finale U_E^{fin} , est soumise, d'une part, à la contrainte (III.9b), et d'autre part, à la contrainte de poursuite de chemin. Cette dernière est donnée par la relation suivante:

$$\forall \lambda \in [0, 1] \quad U_E(\lambda) = \chi(\lambda) \quad (\text{III.10})$$

III.3.1.3 Contraintes liées à l'environnement

Les contraintes émanant de l'environnement concernent essentiellement la non collision du manipulateur mobile avec les obstacles fixes encombrant l'environnement de la tâche à réaliser. Ces contraintes prennent la forme générale suivante :

$$\forall t \in [0, T] \text{ Col}(\mathbf{q}(t)) = \text{faux} \quad (\text{III.11})$$

où Col est une fonction booléenne qui indique si le manipulateur mobile, à la configuration $\mathbf{q}(t)$, est en collision avec l'un des obstacles présents dans son environnement. On peut inclure dans cette contrainte l'auto-collision du robot.

III.4 Conclusion

Ce chapitre a été consacré à la position du problème de génération de trajectoire des manipulateurs mobiles, et à la présentation des différentes tâches faisable au système étudié. Les contraintes ont été classées en trois catégorisées : les contraintes liées au système, les contraintes liées à l'environnement et les contraintes liées à la tâche. Celles ci varient selon le type de la tâche assignée (GPP, OPP, COI,...).

Nous nous intéressons dans ce travail à l'exécution d'une tâche Généralisées Point-à-Point (GPP) par un manipulateur mobile à plateforme différentielle portant un bras manipulateur à deux degrés de libertés l'environnement est libre de collision. Nous allons présenter dans le chapitre suivant une méthode permettant d'aller au delà de la redondance des manipulateurs mobiles et traiter le problème posé comme étant un problème d'optimisation sous contrainte. L'approche est appelée Approche des Profils Aléatoire de trajectoires (random profile Approach RPA).

**Chapitre IV:
Approche des profils
aléatoires de
trajectoires (RPA).**

IV.1 Introduction

Dans ce chapitre, nous allons décrire l'approche des profils aléatoires de trajectoires en expliquant le principe général de cette méthode. Nous allons étendre le champ d'application de RPA à la planification de trajectoires des manipulateurs mobiles exécutant des tâches de type généralisée point-à-point GPP.

IV.2 Approche des profils aléatoires de trajectoires (RPA)

Cette méthode a été proposée dans [2] pour traiter du problème de planification de trajectoires des bras manipulateurs à base fixe. L'application des profils des trajectoires pour accomplir une tâche particulière, a pour but de trouver les configurations successives $q(t)$ du manipulateur mobile sous contraintes citées dans le chapitre III. La méthode RPA est basée sur des profils de trajectoires générée aléatoirement, grâce à des points de contrôle liées par des interpolations de type Beta-spline (voir Annexe B), afin d'accomplir la tâche assignées, en un temps d'exécution de la tâche T .

Le but principale de l'application de la méthode RPA consiste à obtenir une trajectoire **optimale** $q_{\text{best}}(t)$, se déroulant pendant l'intervalle de temps $[0, T_{\text{best}}]$.

IV.2.1 Profil de trajectoire

Une trajectoire généralisée $q(t)$ quelconque est définie dans un espace temps normalisé par le profil de trajectoire $Q(\xi)$. En effet, la recherche de la trajectoire optimale se fera plus aisément, en normalisant l'échelle du temps (avec la fonction linéaire définie par $\xi(t) = t / T$), et en réétudiant le problème dans un intervalle normalisé $[0,1]$.

La fonction vectorielle $Q(\xi)$ décrit complètement la succession des configurations prises par le système robotique étudié, lorsque celui-ci exécute la trajectoire considérée depuis un état initial, RPA échantillonne aléatoirement l'espace des classes de trajectoires, pour rechercher la trajectoire la plus optimale.

La méthode RPA a été quelque peu modifiée, pour découpler le déplacement du système mobile en un chemin $P(\lambda)$ et un mouvement sur ce chemin $\lambda(\xi)$ (Fig IV.1). Cette transformation est due à l'intégration de contraintes de non holonomie auxquels pourraient être soumis les robots manipulateurs mobiles .

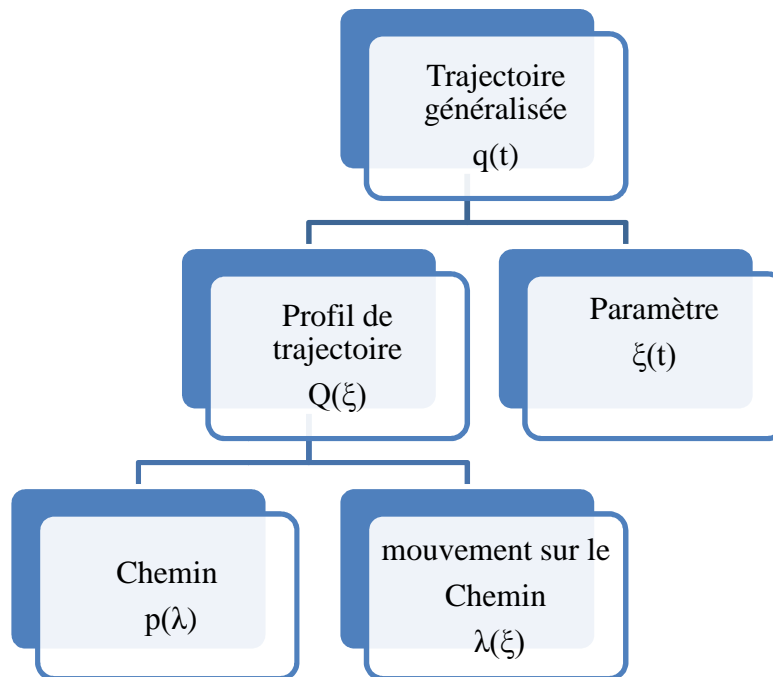


Fig IV.1 : Profil de la trajectoire généralisée selon la méthode RPA.

IV.2.1.1 Décompositions du profil d'une trajectoire

Nous allons dans cette partie formuler cette modification, le profil d'une trajectoire $Q(\xi)$ va être décomposée en un chemin $\mathbf{P}(\lambda)$ et en un mouvement sur ce chemin $\lambda(\xi)$.

- $\mathbf{P}(\lambda)$ est une fonction indépendante du temps. Elle décrit le chemin géométrique du robot dans l'espace des coordonnées généralisée, quand λ varie de façon continue sur $[0, 1]$. La configuration (ou situation) initiale correspond à $\lambda = 0$ et la configuration (ou situation) finale correspond à $\lambda = 1$.

- $\lambda(\xi)$ est une fonction représentant le profil du mouvement. Elle définit la manière selon laquelle le chemin $\mathbf{P}(\lambda)$ sera parcouru dans le temps normalisé.

Les avantages de cette décomposition du profil de trajectoire sont les suivants:

-Les conditions aux limites en position/orientations et en vitesse peuvent être maintenant distribuées sur deux fonctions. La fonction chemin $\mathbf{P}(\lambda)$ doit satisfaire les conditions aux limites en positions, selon le type de la tâche assignée. La fonction mouvement $\lambda(\xi)$ doit satisfaire, d'une part, les conditions suivantes :

$$\lambda(\xi = 0) = 0 \quad \text{et} \quad \lambda(\xi = 1) = 1 \quad (\text{IV.1})$$

$$\lambda'(\xi) > 0 \quad \forall \xi \in]0, 1[\quad (\text{IV.2})$$

et d'autre part, les conditions aux limites en vitesse à savoir :

$$\lambda'(\xi = 0) = 0 \quad \text{et} \quad \lambda'(\xi = 1) = 0 \quad (\text{IV.3})$$

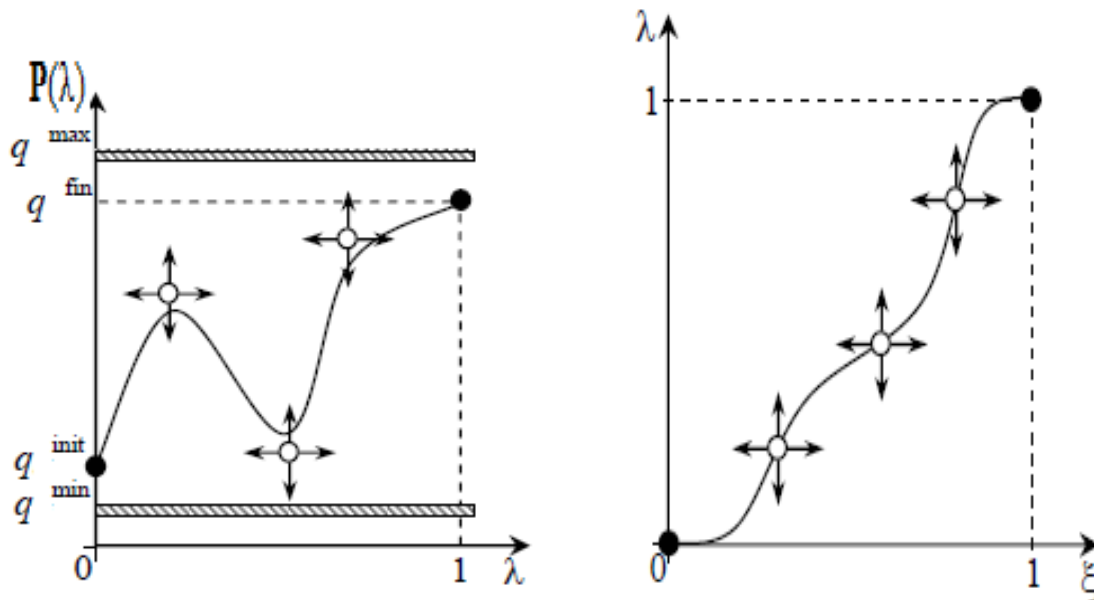
IV.2.2 Modélisation paramétrique d'un profil

Pour calculer la trajectoire la plus optimale, il suffit de modéliser les profils des trajectoires en exploitant des modèles splines.

Nous allons dans ce qui suit présenter la procédure, à adopter, dans le cadre de profils décomposés en profils de chemins, et en profils de mouvement sur ce chemin.

IV.2.2.1 Modélisation paramétrique du chemin et du mouvement

En découplant le profil de la trajectoire courante en chemin, et en mouvement sur ce chemin, Le modèle Beta-spline à été considéré pour le chemin $P(\lambda)$, et un modèle Cubic spline pour le modèle du mouvement sur ce chemin $\lambda(\xi)$ (FigIV.2). L'ordre de ces modèles est élevé (4 ou plus) afin d'assurer la continuité de la trajectoire en position, en vitesse et en accélération.



FigIV.2 : Modèles spline du profil de chemin et du profil du mouvement sur ce chemin.

Après avoir décrit les profils de trajectoires générées aléatoirement par la méthode RPA afin d'accomplir une tâche assignées (définies dans le temps). Nous allons dans ce qui suit présenter la méthodologie pour le calcul du temps d'exécution T d'une tâche, que nous appelons Fonction objective.

IV.2.3 Méthodologie pour le calcul de la fonction objectif

Le calcul de la fonction objectif (Fobj) nécessite toutes les contraintes cinématiques (vitesses et accélérations) qui seront converties en des limites admissibles sur la durée T

d'exécution de la tâche pour un profil donné. Le but envisagé par ce calcul constitue le fond de RPA.

En appliquant la normalisation de l'échelle du temps pour une trajectoire généralisée $q_i(t)$ donnée, les vitesses généralisées $\dot{q}_i(t)$ et les accélérations généralisées $\ddot{q}_i(t)$ de cette trajectoire peuvent être réécrites comme suit :

$$\dot{q}_i(t) = \frac{\partial \xi}{\partial t} \frac{\partial Q_i(\xi)}{\partial \xi} = \frac{1}{T} Q'_i(\xi) \tag{IV.4}$$

$$\ddot{q}_i(t) = \frac{\partial \xi}{\partial t} \frac{\partial Q'_i(\xi)}{\partial \xi} = \frac{1}{T} Q''_i(\xi) \tag{IV.5}$$

En considérant \dot{q}_i^{\max} comme étant la vitesse maximale admissible, Par définition on a Pour $t \in [0, T]$:

$$|\dot{q}_i(t)| \leq \dot{q}_i^{\max}$$

Remplaçons $\dot{q}_i(t)$ par (IV.4) et pour $\xi \in [0, T]$ on aura

$$\frac{1}{T} |Q'_i(\xi)| \leq \dot{q}_i^{\max}$$

La valeur du paramètre T doit être supérieure à $T_V = \max [\max |Q'_i(\xi) / \dot{q}_i^{\max} |]$.

La même procédure doit être appliquée, pour calculer le temps minimal de déplacement, avec prise en compte de la contrainte en accélération, sachant que pour $t \in [0, T]$:

$$|\ddot{q}_i(t)| \leq \ddot{q}_i^{\max}$$

Remplaçons $\ddot{q}_i(t)$ par (IV.5) et pour $\xi \in [0, T]$ on aura

$$\frac{1}{T} |Q''_i(\xi)| \leq \ddot{q}_i^{\max}$$

Dans ce cas, T doit être supérieure à $T_A = \max [\max |(Q''_i(\xi) / \ddot{q}_i^{\max})^{1/2} |]$.

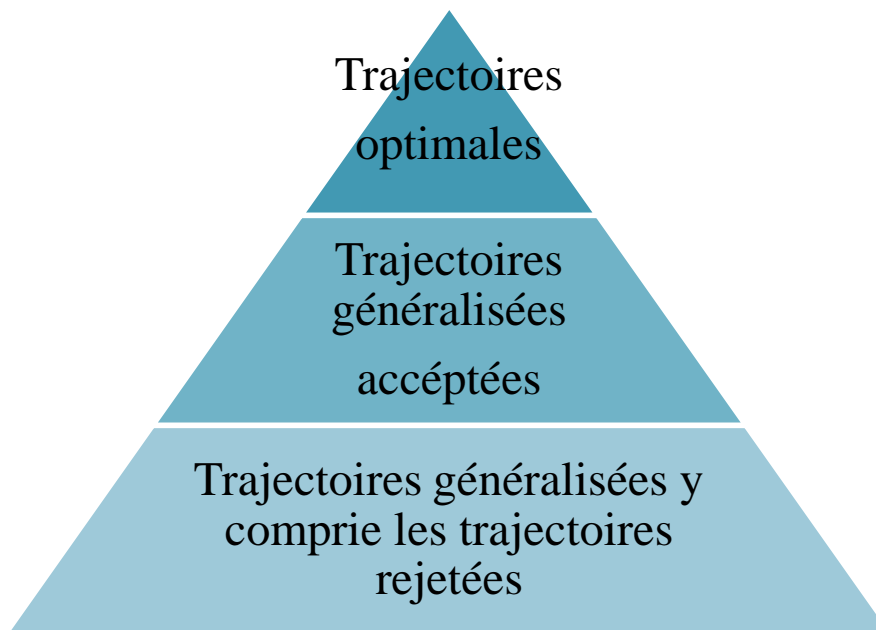
Le temps de déplacement minimal T_M , en considérant les vitesses et accélérations, est calculé comme le maximum des deux valeurs déduites précédemment $T_M = \max (T_V, T_A)$. D'où on considère la fonction objectif **Fobj = T_M** .

IV.2.4 Détermination de la trajectoire optimale

Pour atteindre une trajectoire optimale RPA modifie la position des points de contrôle libres jusqu'à atteindre la trajectoire la plus optimale, Le calcul de la fonction objectif Fobj a pour objectif d'évaluer la performance de tout profil donné. Il sera exploité de façon répétée par un procédé stochastique d'optimisation basé sur l'algorithme du recuit simulé (voir Annexe A). Celui-ci a pour objectif de diriger la recherche vers le profil le plus performant. Il explore aléatoirement

l'espace des points de contrôle libres et génère des profils qu'il soumet à évaluation. La particularité du recuit simulé est sa remarquable faculté de ségrégation graduelle [Had02] qui lui permet de cibler progressivement les profils ayant le plus de chance de fournir le meilleur résultat. La solution finale espérée de cette optimisation imbriquée est la trajectoire la plus performante $Q_{\text{best}}(t)$ se déroulant pendant l'intervalle de temps $[0, T_{\text{best}}]$.

La détermination de la trajectoire optimale est schématisée sur la figure ci-dessous (FigIV.3).

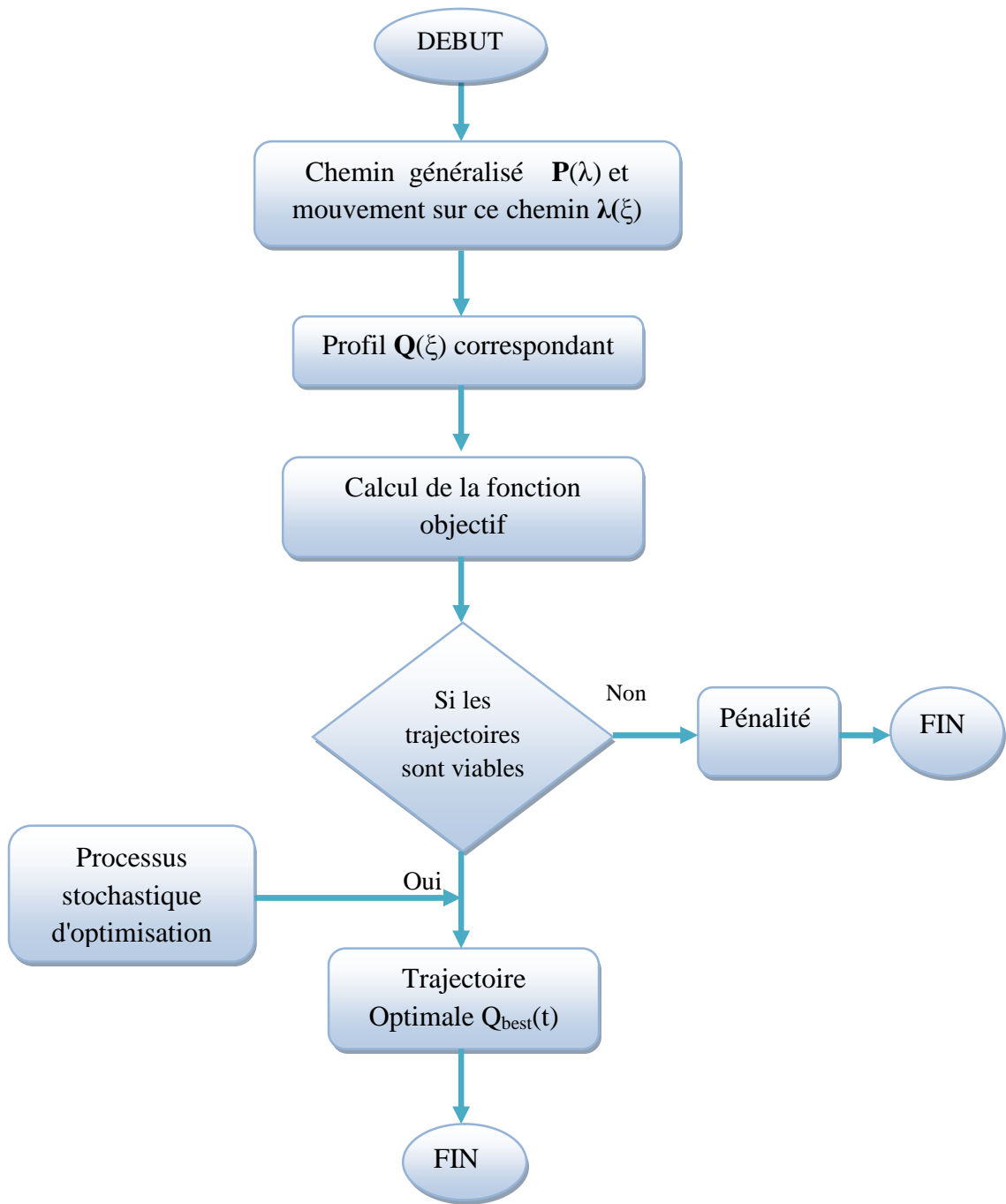


FigIV.3 : Portée des calculs effectués par RPA.

IV.2.5 Récapitulatif

En résumé, l'approche utilisée consiste à générer aléatoirement un chemin généralisé $P(\lambda)$ et un mouvement $\lambda(\xi)$ puis à déduire le profil $Q(\xi)$ correspondant, ces profils seront soumis au calcul de la fonction objectif. Certaines trajectoires calculées ne respectent pas les contraintes imposées d'où elles sont traitées par rejet (pénalité). En conséquence, seules les trajectoires viables seront retenues et soumises au processus stochastique d'optimisation. Ce dernier mesure la meilleure performance pouvant être obtenue avec une trajectoire ayant le profil considéré. Toutes ces étapes sont répétées jusqu'à obtenir la trajectoire optimale.

L'organigramme présenté sur la Fig.IV.4 présente les étapes générales pour le calcul de la trajectoire optimale selon la méthode RPA.



FigIV.4 : L'organigramme de la recherche d'une trajectoire optimale selon la méthode RPA.

Nous allons dans ce qui suit étendre le champ d'application de la méthode RPA à la planification des tâches GPP, en utilisant un manipulateur mobile constitué d'une plateforme de type différentiel et d'un bras à deux d.d.l.

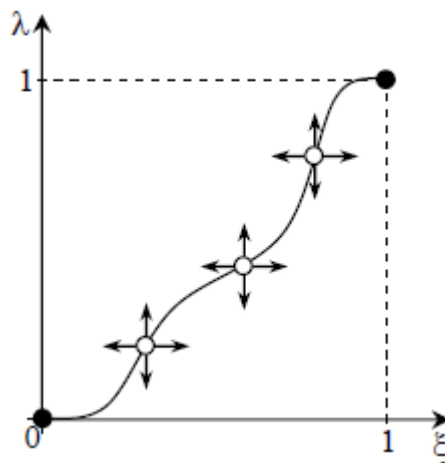
IV.3 Adaptation de RPA aux tâches GPP

Dans cette section, nous allons proposer une adaptation de RPA à notre système mobile. Cette adaptation permettra, d'une part, de contourner l'ensemble des difficultés et qui concernent, notamment, la prise en compte des contraintes non holonomes et des conditions aux limites en position, en orientation et en vitesse. D'autre part, elle ne perdra aucun des avantages déjà offerts par la technique de base, à savoir, la transformation du problème en la recherche de la position optimale de quelques points de contrôle et la transformation des contraintes cinématiques en des limites admissibles sur la durée d'exécution T

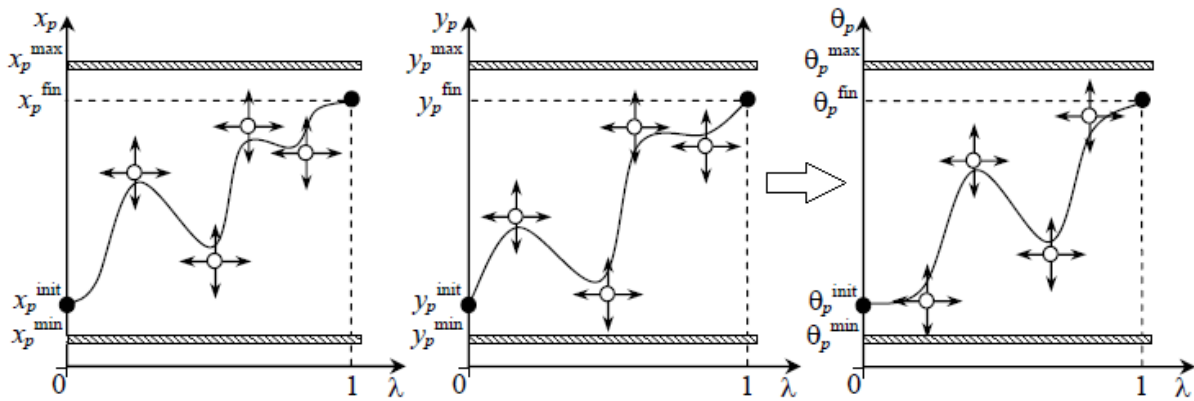
L'approche que nous avons appliquée sur le manipulateur mobile est basée sur les étapes suivantes:

- La décomposition du profil de la trajectoire en un chemin et en un mouvement sur ce chemin.
- Le calcul de la fonction objectif.
- L'optimisation de la trajectoire correspondante.

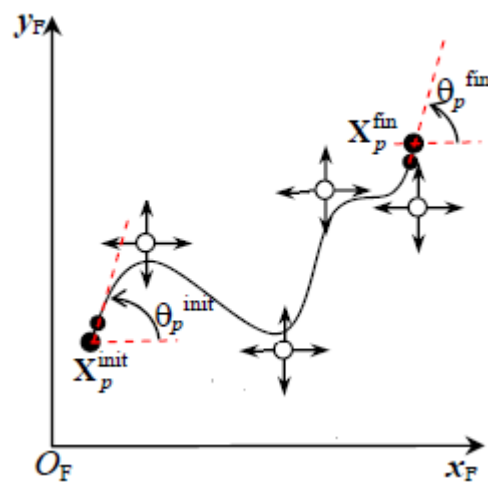
Dans notre cas, l'application de la méthode RPA pour le manipulateur mobile étudié est basée sur les mêmes étapes présentées dans le paragraphe §IV.2, sauf que ici on précise tout les paramètres concernant l'étude notre manipulateur mobile dans Modélisation paramétrique du chemin et du mouvement donnons $\lambda(\xi), P_p(\lambda) = [X_p(\lambda), Y_p(\lambda), \theta_p(\lambda)]^T$ et $P_b(\lambda) = [q_{b1}(\lambda), q_{b2}(\lambda)]^T$ comme nous montre les figures FigIV.5 FigIV.6 FigIV.7 FigIV.8.



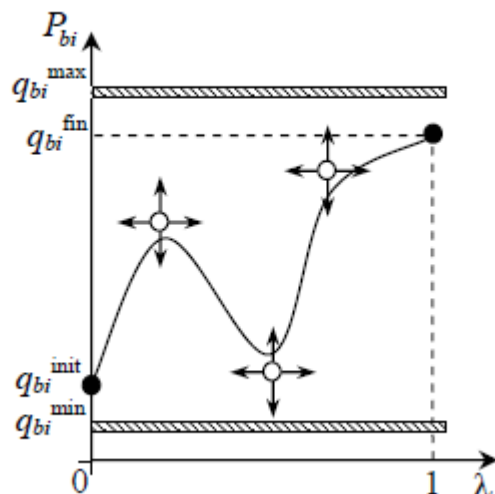
FigIV.5 : Représentation discrète de la fonction mouvement.



FigIV.6 : Représentation discrète du chemin des paramètres de la plateforme X_p, Y_p et du chemin de θ_p déduit.



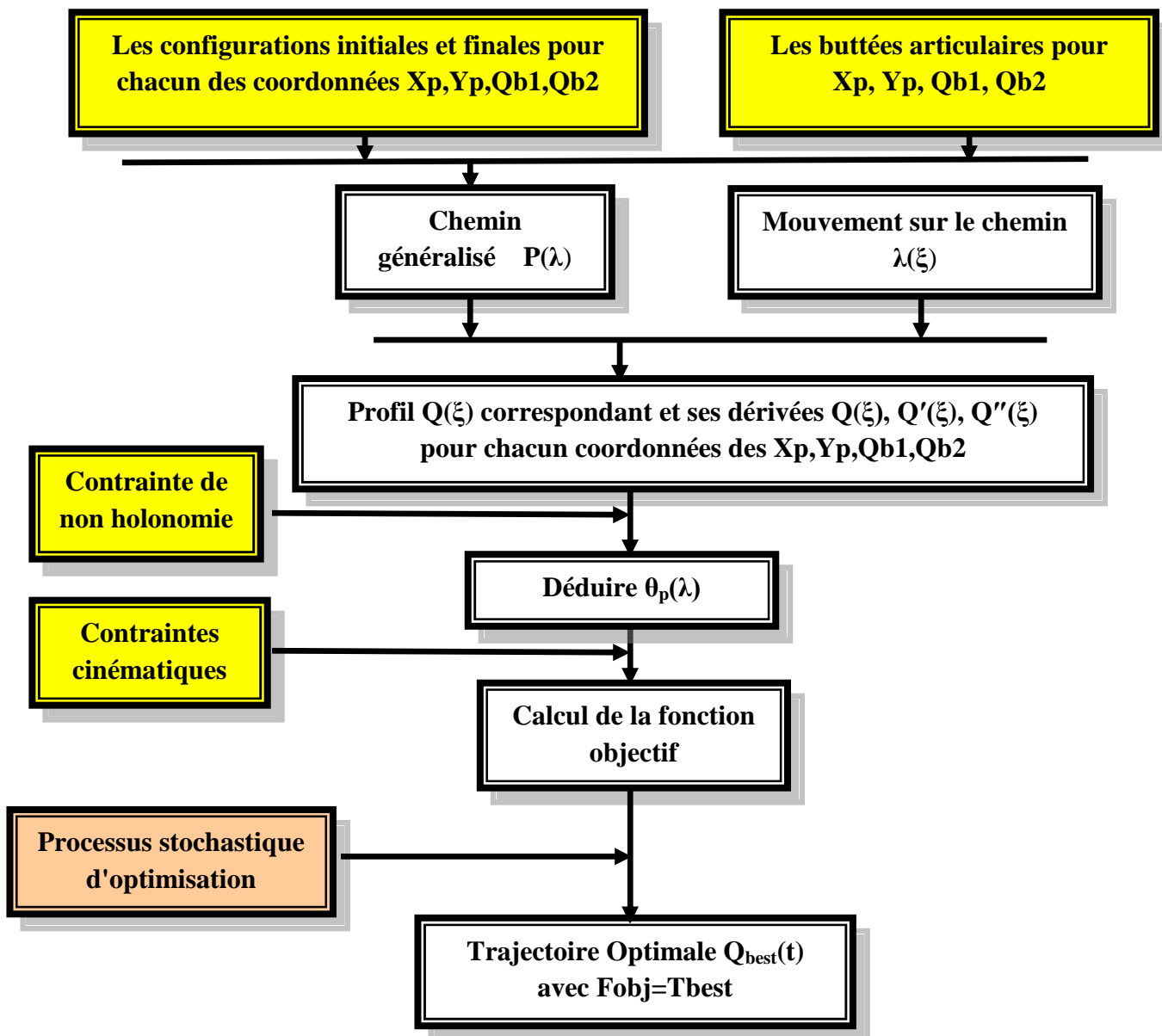
FigIV.7 : Représentation discrète du chemin de la plateforme.



FigIV.8 : Représentation discrète du chemin généralisé du bras pour une articulation.

Nous pouvons générer aléatoirement un mouvement $\lambda(\xi)$ en positionnant les points de contrôle libres et un chemin généralisé $\mathbf{P}(\lambda)$ en positionnant les points de contrôle. Notons que, nous

effectuerons les calculs avec $N_p=6$ points de contrôle pour construire le chemin de la plateforme et du bras manipulateur au lieu de $N_p = 4$ points pour la tâche GPP car avec $N_p=4$ on obtient des profils moins lisse que celle obtenues avec $N_p=6$ (voir chapitre V), et nous fixons les configurations initiales et finales des paramètres X_p, Y_p, q_{b1} et q_{b2} avec prise en compte de leur valeurs maximales et minimales, Pour les composantes θ_p nous fixons les configurations initiales et finales pour l'orientation du chemin de la plateforme (FigIV.7) par ailleurs nous déduisons θ_p par intégration des contraintes non holonomes (CNH), et en ce qui concerne le calcul de la fonction objective et le calcul de la trajectoire optimale leur description est la même que celle qui est citée dans le paragraphe §IV.2, l'organigramme (FigIV.9) résume toutes ces étapes.



FigIV.9 : l'organigramme de l'adaptation de RPA aux tâches GPP.

IV.4 Conclusion

Ce chapitre a été consacré à l'extension de RPA à la planification des manipulateurs mobiles exécutant des tâches de type GPP.

La manière selon laquelle cette extension a été établie montre l'adaptabilité de RPA aux problèmes de planification de trajectoires donnés pour des tâches GPP, consiste uniquement, à libérer (ou fixer) certains points de contrôle, à déplacer certains inconnus de l'ensemble indépendant vers un autre ensemble d'inconnus déduits ou à ajouter un ensemble de points de contrôle supplémentaires.

Les différents problèmes traités montrent l'efficacité de RPA que ce soit en temps de calcul ou en qualité des résultats (convergence vers des comportements naturels). Par exemple, pour les problèmes à temps minimal d'exécution de tâches, RPA converge vers des solutions qui tendent vers la saturation des contraintes le long de la trajectoire, et nous allons montrer cette efficacité en simulation dans le chapitre suivant.

Chapitre V : Mise en œuvre : Simulations.

V.1 Introduction

En vue de montrer la flexibilité de la démarche appliquée dans le chapitre précédent (méthode des Profils Aléatoires de Trajectoires RPA), nous présentons dans ce dernier chapitre des simulations numériques relatives aux Tâches Généralisée Point à Point(GPP) .

Le manipulateur mobile utilisé pour le traitement des deux types de problèmes posés est constitué, d'une plateforme mobile à roues de type différentiel et d'un bras manipulateur à 2 d.d.l (présenté dans le chapitre II). Les caractéristiques numériques concernant les entrées de taches traitées (conditions initiales et finales ainsi que les vitesses et accélérations maximales) sont données en Annexe C.

V.2 Description du simulateur

Nous avons effectuée nos simulations avec le langage de programmation Pascal. La plateforme logicielle utilisée est Delphi. La figure suivante représente la forme principale.

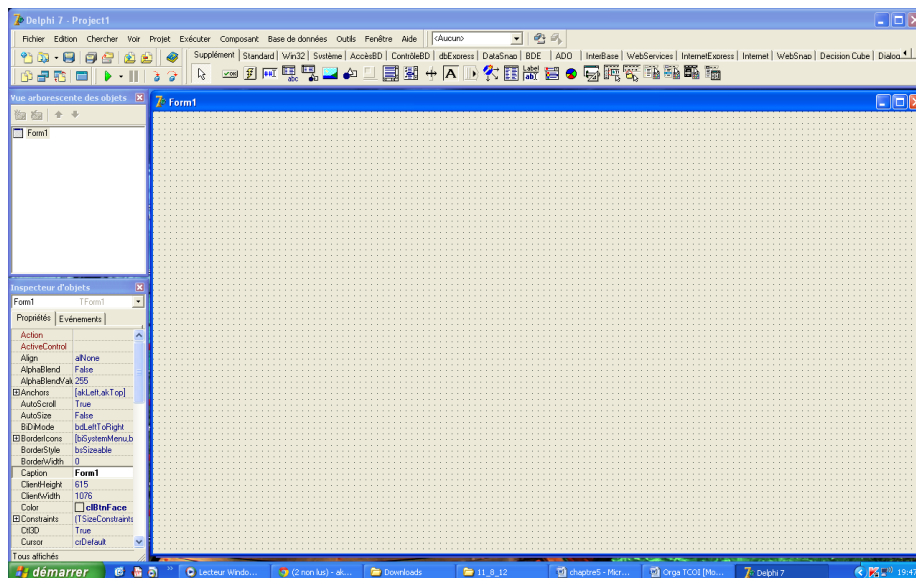


Fig V.1 : Vue de la forme principale.

Delphi est un outil de développement puissant pour la programmation d'application pour Windows [9]. il assemble les applications à l'écran de manière fort simple. Il dispose pour cela d'une palette richement garnie de composants, cette dernière est désignée par le sigle VCL (Visual Component Library : bibliothèque des composant visuels), au plus de ces composants qui permettent de créer facilement des interfaces d'applications. Il dispose aussi de zones de listes , de

boîtes de dialogue toute prêtes, une table de contrôle complète pour la création d'application multimédia et on trouve aussi les outils de connexion aux bases de données...etc.

Delphi est un outil à double usage puisqu'il permet de créer des applications de deux façons différentes : la première façon est visuelle, elle utilise le glisser-déplacer à l'écran. La seconde consiste une programmation traditionnelle(en langage Pascal). On peut exclusivement appliquer une de ces techniques, mais plus souvent ces dernières sont combinées.

Avec Delphi nous permettons de développer, tester et maintenir des applications au sein d'un environnement unique et intégré et on peut aussi créer ou modifier une application (fiche de saisie, modules de données et composants). Toutes ces fonctions évoluent dans un environnement RAD (Rapid Application Développement : Développement rapide d'applications).

V.3 Résultats et description des expérimentations

V.3.1 Tâche généralisée point à point (GPP)

Afin d'appliquer la méthode RPA nous avons créé l'interface représentée dans Fig. V.2) qui nous permet de voir toutes les solutions possibles générées aléatoirement, ainsi que d'avoir la solution optimale.

V.3.1.1 Les démarches du programme

Le programme principal (Unités principale) conçu, définit toutes les procédures des composants utilisés. Ce dernier fait appel également a d'autres unités (calcul, optim et simopt, Hdon_MOB_Man, U_planif, Ram,Optim, Stat, Simopt) permettent de faire appel à des procédures spécifiques utiles dans notre travail. Cependant ces procédures ont été tirées d'une librairie mathématique spécialement conçue pour Delphi

Pour trouver une trajectoire optimale, l'opérateur doit suivre les étapes suivantes. L'interface principale est présentée dans la figure Fig.V.2.

1. **Lire les paramètres géométriques, cinématiques**(Bouton Début) Récupérer les données géométriques et cinématiques ($Q_{int}, Q_{fin}, Q_{min}, Q_{max}, Q_{pmax}, Q_{semax}$) des paramètres (q_1, q_2, q_3, q_4, q_5) d'un fichier texte et entrer les valeurs n_p (nombre du point de contrôle B-Spline), m_{BS} (nombre de degrés de la bétat spline), et n_s (nombre du point de contrôle Cub-Spline).
2. **Générer une solution initiale aléatoirement** (Bouton Init)

-Imposer une solution aléatoire initiale en chargeant un vecteur X_r tel que $X_r[i]=X_{rmin}+Ram*(X_{rmax}[i]+X_{rmin}[i])$. Ram est une fonction qui génère aléatoirement des nombres variant entre X_{rmax} et X_{rmin} de sorte à respecter les limites établies comme les butées articulaires par exemple.

-Calculer la fonction objective **Fobj_Temps_min** pour la solution aléatoire courante.

3. **Calculer la solution optimale**(Bouton optimum)

-Optimiser la trajectoire et calculer **Tfinal=Fbest** par recuit simulé.

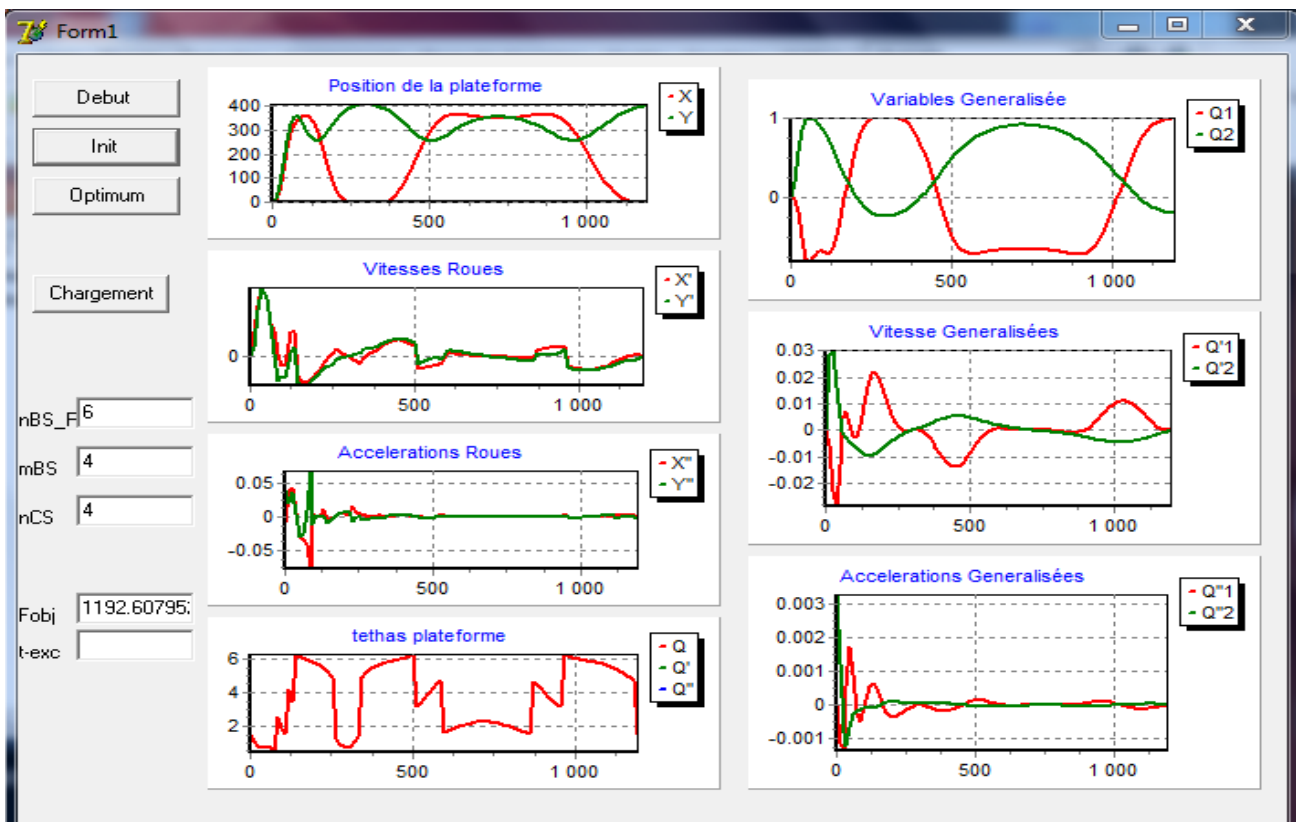


Fig V.2 : résultat obtenu pour une solution initiale.

V.3.1.2 Résultats de simulation

Nous avons appliqué les démarches du programme citées dans le paragraphe précédent en appuyant sur le bouton «debut», puis sur le bouton «int». les figure V.3 et la figure V.4 présentent deux essais d'initialisation différents.

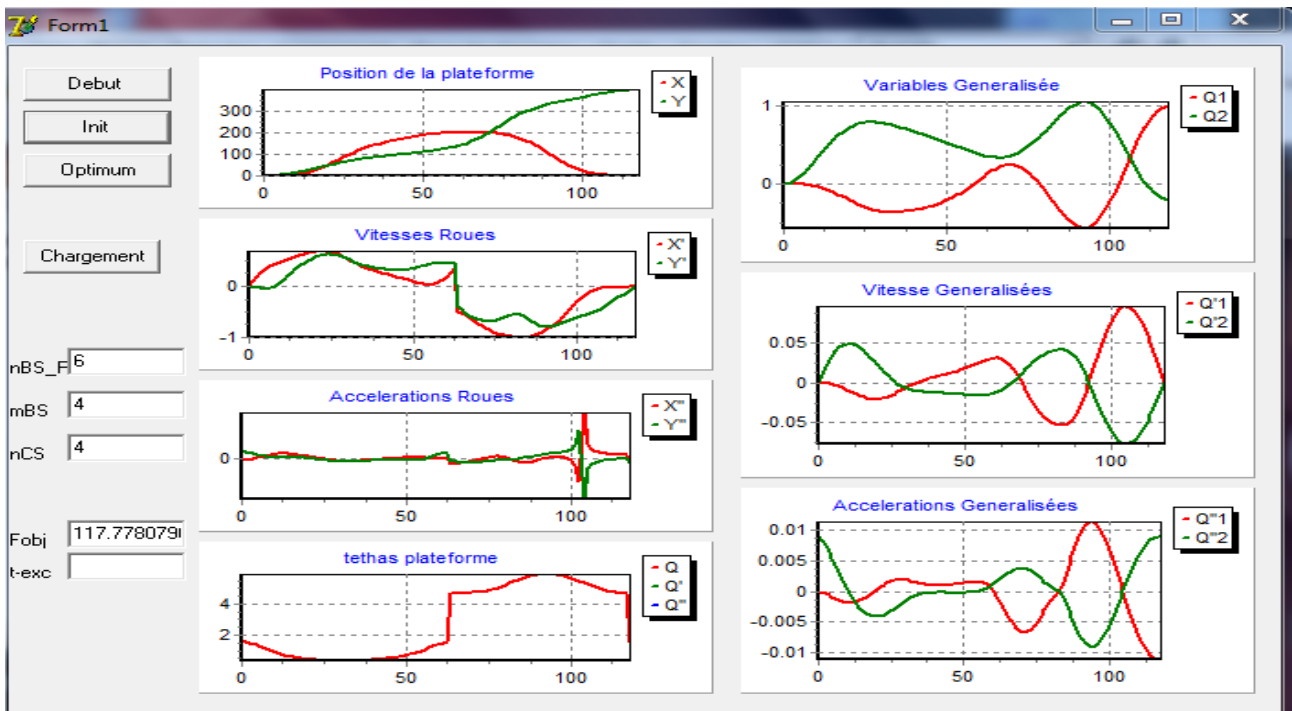


Fig V.3 : résultat obtenu pour une première solution initiale.

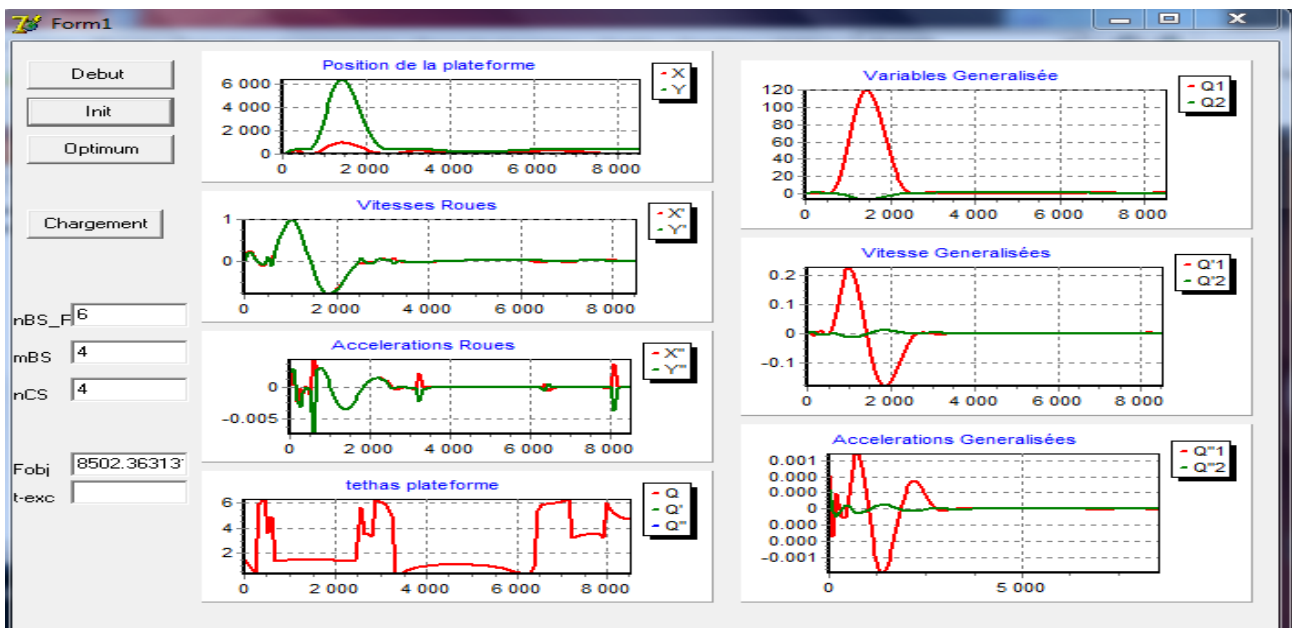


Fig V.4 : résultat obtenu pour une seconde solution initiale.

D’après ces deux figures, on constate que le temps calculé Fobj sont différents sur les figures V.3 et V.4. Le résultat de la figure FigV.4 est rejeté, et celui de Fig.V.3, est accepté.

Pour les solutions acceptées on remarque que le robot se déplace de la configuration initiale $(q^{init}(1), q^{init}(2), q^{init}(3), q^{init}(4), q^{init}(5)) = (0,0,0,0,1.57)$ radians, à la configuration finale $(q^{fin}(1), q^{fin}(2),$

$q^{\text{fin}}(3), q^{\text{fin}}(4), q^{\text{fin}}(5) = (1, -0.2, 0, 400, 1.57)$ radians . On remarque également que les variables généralisées ne dépassent pas les limites des butées articulaires, en ce qui concerne les vitesses généralisées et les vitesses des roues , elles sont nulles au début et à la fin du parcours, cela montre que les conditions aux limites de la tâche ont été respectées lors de la génération de la trajectoire, et elles sont aussi limitées à la vitesse maximale Q_{pmax} , Les accélérations généralisées aussi ne dépassent pas l'accélération maximale Q_{semax} .

Pour la solution rejetée on constate que les contraintes géométriques ne sont pas respectées, $q(1) > Q_{\text{max}}$ et $q(4) > Q_{\text{max}}$ (voir Annexe C) à $t=8502$ secondes.

Maintenant nous allons rechercher la trajectoire optimale $q_{\text{best}}(t)$, (se déroulant pendant l'intervalle de temps $[0, T_{\text{best}}]$) en appuyant sur le bouton «optimum» et le résultat obtenu est donné par la figure V.5

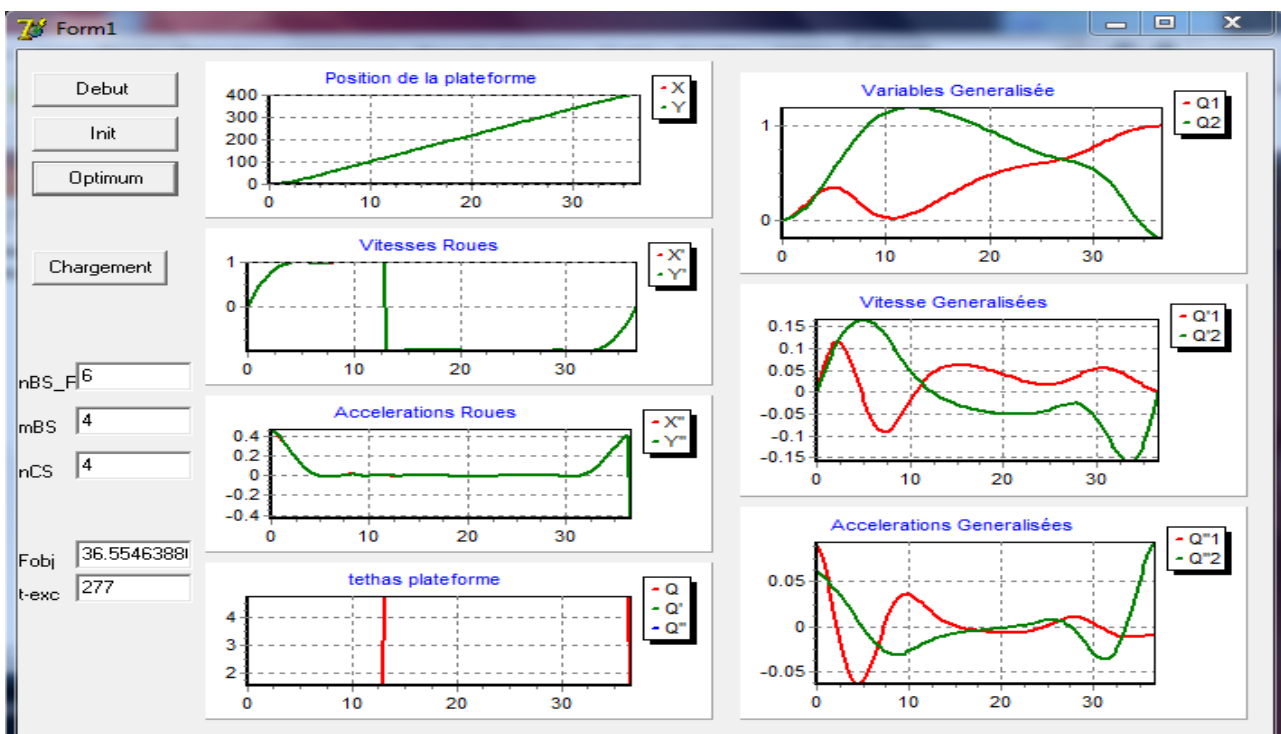


Fig V.5 : solution optimale.

Le temps calculé est $F_{\text{obj}}=36.5546388$ secondes qui est bien inférieur à ceux que nous avons obtenu précédemment, pour un temps d'exécution de l'algorithme de 277 secondes.

Nous constatons que les configurations initiales et finales sont respectées ainsi que toutes les contraintes géométriques et cinématiques sont respectées elles aussi.

Nous pouvons constater aussi que la variables généralisée $q_3=0$ qui est la composante X de la plateforme et q_4 qui est la composante Y de la plateforme varie linéairement, cela signifie que le robot mobile se dirige tout droit suivant l'axe des Y.

Nous pouvons également constater que la vitesse et l'accélération généralisée des roues ont atteint leurs valeurs maximales (saturation en vitesse). Cela signifie que c'est la vitesse et l'accélération des roues qui ont guidé la recherche de la trajectoire optimale.

Remarque

Pour la tâche généralisée point à point on a posé N_p le nombre de point de contrôle égale à 6 et non pas à 4 car avec $N_p=4$ on obtient des trajectoires moins lisse par rapport à celle obtenue avec $N_p=6$. (voir Fig V.6)

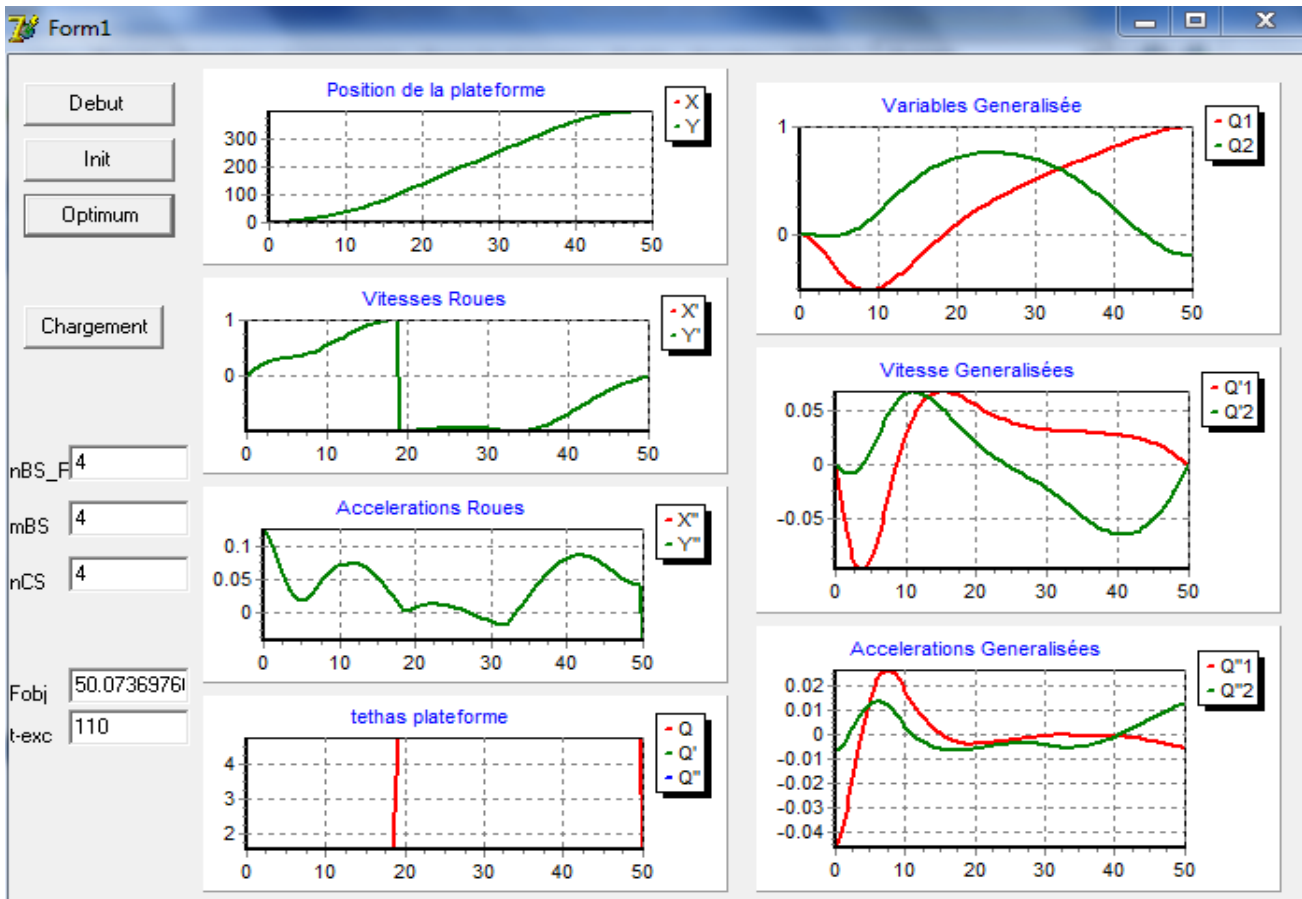


Fig V.6 :solution optimale avec $N_p=4$.

V.4 Conclusion

A travers le présent chapitre ,nous avons montré les avantages fondamentaux qu'offre RPA. Cette approche est capable de traiter des problèmes de planification de trajectoires pour des manipulateurs mobiles exécutant des tâches GPP selon une variété de critères d'optimisation et sous divers type de contraintes (contraintes géométriques et cinématiques).

Les résultats obtenus ont montré que le processus d'optimisation incorporé dans RPA arrive à produire des comportements naturels de bonne qualité dans des temps de calcul qui sont généralement très acceptables.

Enfin nous pouvons conclure que RPA reste une méthode générale. La simplicité algorithmique qu'elle présente est acquise des propriétés des techniques stochastiques d'optimisation. De ce fait, elle est applicable à plusieurs types de problèmes de planification de trajectoires pour des tâches GPP.

Conclusion Générale.

Conclusion Générale

Le travail présenté dans ce mémoire est une continuité des travaux, sur la planification de trajectoires des systèmes mécaniques robotisés. Il porte en particulier sur l'exploitation optimale des manipulateurs mobiles à roues exécutant divers types de tâches.

L'objectif que nous nous sommes fixé a été d'élargir à la robotique mobile le champ d'application de l'approche des profils aléatoires de trajectoires (RPA). Cette dernière, utilise une optimisation imbriquée sur deux niveaux. Au niveau interne, elle utilise une technique déterministe d'optimisation unidimensionnelle sous contraintes qui obtient le meilleur temps d'exécution de la tâche pour un profil donné de trajectoire. Au niveau externe, elle utilise une technique stochastique d'optimisation qui cible le meilleur profil de trajectoire satisfaisant le critère de performance considéré.

Cependant cette approche, telle qu'elle a été initialement proposée, n'est pas directement applicable à la robotique mobile, principalement en raison de la difficulté de prise en compte des contraintes non holonomes. Celles-ci devant être satisfaites sur tout le parcours du robot, leur traitement par un simple rejet dans le processus stochastique de sélection n'est pas envisageable car il mènerait à un alourdissement intolérable des calculs.

Afin de résoudre ce problème, tout en sauvegardant les avantages les plus intéressants de RPA, nous avons été amenés à lui apporter les modifications suivantes :

- a. la décomposition du profil de la trajectoire en un chemin généralisé et en un mouvement sur ce chemin ;
- b. la modélisation du chemin et du mouvement par une fonction paramétrique adaptée à chacun,
- c. le choix adéquat de l'ensemble des inconnues indépendantes (ou paramètres libres) du problème.

La décomposition du profil de la trajectoire joue un rôle crucial car elle permet de simplifier encore plus le traitement des diverses contraintes du problème. Les contraintes

Conclusion Générale

qui dépendent de la durée d'exécution de la tâche ont été transformées, en des limites admissibles sur cette durée pour un profil donné de trajectoire. Les autres contraintes, qui ne dépendent que du profil de la trajectoire, ont été redistribuées sous forme de conditions sur les fonctions qui modélisent, respectivement, le mouvement, le chemin généralisé de la plateforme et le chemin généralisé du bras manipulateur. Par exemple, les conditions aux limites en vitesse ne concerneront que le profil du mouvement alors que les conditions aux limites en position/orientation ne concerneront que le chemin généralisé du système. La prise en compte proprement dite de ces contraintes s'effectue, quand il n'est pas possible de faire autrement, par un simple test de conformité suivi du rejet des candidats non viables. Mais dans la plupart des autres cas, les contraintes sont incluses lors de la construction des profils de mouvement et de chemin. Ceci est fait soit directement, c'est le cas notamment des conditions aux limites en vitesse et en position/orientation, soit indirectement en scindant tout d'abord les inconnues du problème en deux parties: la première regroupe les variables qui peuvent être librement générées par le processus stochastique et la deuxième celles qui seront déduites à partir des contraintes imposées. Le résultat évidemment est l'amélioration de la qualité de l'échantillonnage des profils aléatoires de trajectoires, la réduction de la proportion des rejets et la diminution des temps de calcul.

Comme perspectives, nous envisageons les thèmes suivants :

- Planification de trajectoires d'un manipulateur mobile pour l'exécution des tâches opérationnelle point à point, à chemin opérationnel imposé et à mouvement opérationnel imposé.
- Planification d'un manipulateur mobile pour l'exécution de plusieurs tâches enchainées.
- Planification de trajectoires des plateformes mobiles et des manipulateurs mobiles avec prise en compte dynamique des contraintes de non glissement.
- Planification de trajectoire des plateformes mobiles et des manipulateurs mobiles évoluant dans un environnement encombrés.
- Planification de trajectoire des plateformes mobiles et des manipulateurs mobiles évoluant dans un environnement 3D.

Annexe A: Techniques stochastiques d'optimisation.

La résolution d'un problème d'optimisation consiste à trouver, parmi un ensemble Ψ des solutions admissibles, une solution \mathbf{S}^{best} qui minimise une fonction coût $J(\mathbf{S})$ donnée.

La majorité des méthodes envisagées pour résoudre ce problème sont basées sur des techniques itératives de recherche. Leur principe est le suivant : à partir d'une solution de départ \mathbf{S}_0 , considérée temporairement comme étant la meilleure solution trouvée. On engendre, par une technique déterministe ou aléatoire (stochastique), une suite finie de voisins $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_k$ pour arriver à la fin à une approximation \mathbf{S}^* de la solution finale \mathbf{S}^{best} .

Pour un problème d'optimisation compliqué ayant plusieurs minima locaux, les techniques de résolution préférées sont les techniques stochastiques (les méthodes de Monte Carlo, recuit simulé, recherche Tabou, les algorithmes génétiques, ...). Ces dernières se basent sur une recherche aléatoire uniformément distribuée dans l'espace Ψ des solutions. Elles peuvent, de ce fait, éviter les pièges des minima locaux et assurer une bonne approximation de la solution globale \mathbf{S}^{best} recherchée. Elles ne font appel qu'à l'évaluation de la fonction coût et à la vérification des contraintes imposées.

On présente, ici, le principe de quelques techniques stochastiques d'optimisation, en l'occurrence : la méthode de descente, la méthode du recuit simulé, la recherche Tabou et les algorithmes génétiques. Nous avons utilisé dans le présent travail la méthode du recuit qui est une technique de recherche dans lequel on introduit de l'aléatoire.

A.1 Méthode de descente

Le principe de la méthode de descente (dite aussi basic local Search) consiste à partir d'une solution « S » et à choisir une solution « S' » dans un voisinage de « S », telle que s'améliore la recherche (généralement telle que $f(S') < f(S)$).

On peut décider soit d'examiner toutes les solutions du voisinage et prendre la meilleure de toutes (ou prendre la première trouvée), soit d'examiner un sous-ensemble du voisinage.

La méthode de recherche locale la plus élémentaire est la méthode de descente. On peut la schématiser comme suit :

Procédure descente simple (solution initiale s)

Répéter :

Choisir S' dans $N(S)$

Si $f(S') < f(S)$ alors $S \leftarrow S'$

Jusqu'à ce que $f(S') \geq f(S), \forall S' \in S$

Fin

● Inconvénients

En général, l'efficacité des méthodes de recherche locale simples (descente, ou plus grande descente) est très peu satisfaisante. D'abord, par définition, la recherche s'arrête au premier minimum local rencontré, c'est là leur principal défaut. Pour améliorer les résultats, on peut lancer plusieurs fois l'algorithme en partant d'un jeu de solutions initiales différentes, mais la performance de cette technique décroît rapidement.

● Avantages

Le principal avantage de la recherche locale simple est évidemment sa grande simplicité de mise en œuvre: la plupart du temps, elle ne fait que calculer $f(S+i)-f(S)$, où i correspond à un déplacement élémentaire, et si cette expression peut se simplifier algébriquement, alors on pourra évaluer très rapidement cette différence.

A.2 Méthode de recuit simulé

Le recuit est un processus physique de chauffage. Il est utilisé en métallurgie pour améliorer la qualité d'un solide. On cherche à atteindre un état d'énergie minimale qui correspond à une structure stable du métal. En partant d'une haute température à laquelle la matière est devenue liquide, la phase de refroidissement conduit la matière à retrouver sa forme solide par une diminution progressive de la température. Ainsi, lorsqu'on chauffe un métal solide, il devient liquide à une certaine température dans ce cas les atomes qui le compose ont vu leur degré de liberté augmenter. Inversement lorsque l'on baisse la température, le degré de liberté diminue jusqu'à obtenir un solide. Suivant la façon dont on diminue la température on obtient différent solide :

-Baisse brutale de la température (La trempe) cela produit une structure amorphe, un verre. On a alors un minimum local d'énergie.

-Baisse progressive de la température de façon à atteindre le minimum global d'énergie. On obtient dans ce cas un cristal.

L'idée du recuit simulé en optimisation combinatoire est de simuler numériquement une opération de recuit thermique. Le principe en est le suivant :

On considère un système composé de N éléments. A chaque configuration du système est associée une fonction à optimiser. Si l'on recherche la configuration qui minimise la fonction, celle-ci joue le rôle de l'énergie. Si l'on recherche au contraire un maximum, alors on prend comme énergie l'opposée de la fonction. Les configurations du système peuvent être modifiées par des changements d'états discrets des composants du système. Bien entendu, les systèmes qui nous intéressent sont ceux qui ont un grand nombre d'optima.

On part d'une configuration aléatoire (ou choisie astucieusement en fonction du problème). Au départ de l'itération, on fixe un paramètre de température en relation avec la gamme des énergies accessibles au système. On itère alors le processus suivant (dit de Metropolis ou de Monte-Carlo): on tire au sort une modification de la configuration actuelle qui change l'énergie du système d'une quantité ΔE . Si l'énergie diminue, on effectue le changement. Si l'énergie augmente, on effectue le changement avec une probabilité $\exp - \Delta E/T$. L'itération se poursuit tant que l'énergie du système diminue. Lorsque l'énergie reste stationnaire, on diminue un peu la température et l'on reprend le processus Monte-Carlo de décroissance de l'énergie. On arrête lorsque les diminutions de température restent inefficaces.

Schéma d'algorithme de recuit simulé

- *Engendrer une configuration initiale S_0 de S ;*

$$S := S_0$$

- *Initialiser T en fonction du schéma de refroidissement*
- *Répéter*
- *Engendrer un voisin aléatoire S' de S*
- *calculer $\Delta = f(S') - f(S)$*
- *Si CritMetropolis (Δ, T), alors $S := S'$*
- *Mettre T à jour en fonction du schéma de refroidissement*
- *Jusqu'à <condition fin> (si les conditions d'arrêt soient atteintes)*

- *Retourner la meilleure configuration trouvée*

Le critère de Metropolis dépend de la fonction (Δ, T) , si $\Delta \leq 0$ donc la solution est vraie sinon avec une probabilité de $\exp(-\Delta/T)$ la solution peut être vraie et peut être aussi fautive.

- Un voisin qui améliore ($\Delta < 0$) ou à coût égal ($\Delta = 0$) est toujours accepté.

Schéma de refroidissement

La fonction qui spécifie l'évolution de la température est appelée le schéma de refroidissement. Dans le recuit simulé standard la température décroît par paliers. Par exemple, on pourrait avoir trois paramètres : la température initiale, la longueur d'un palier (nombre d'itérations avant de changer la température) et le coefficient de décroissance (si décroissance géométrique). On peut aussi utiliser d'autres schémas de refroidissement : On peut faire décroître la température à chaque itération. On utilise parfois une température constante (algorithme de Metropolis). On peut également utiliser des schémas plus complexes, dans lesquels la température peut parfois remonter.

• Avantages et inconvénients

Le recuit simulé présente l'avantage d'offrir des solutions de bonne qualité, tout en restant simple à programmer et à paramétrer. Il offre autant de souplesse d'emploi que l'algorithme de recherche local classique : on peut inclure facilement des contraintes dans le corps du programme. Le recuit simulé est un algorithme facile à implémenter. Il possède également des propriétés de convergence intéressantes. Aarts, Korst et Laarhoven (1997) ont par ailleurs démontré que, sous certaines conditions de décroissance de la température, l'algorithme du recuit simulé converge en probabilité vers un optimum global lorsque le nombre d'itérations tend vers l'infini. L'un des inconvénients du recuit simulé est qu'une fois l'algorithme piégé à basse température dans un minimum local, il lui est impossible de s'en sortir tout seul.

A.3 La méthode Tabou

La méthode de recherche Tabou est l'une des méthodes d'amélioration locale. Elle consiste à se déplacer d'une solution \mathbf{S} à une autre \mathbf{S}^* , qui se trouve au voisinage $V(\mathbf{S})$ de \mathbf{S} , tout en évitant de revenir à une solution déjà rencontrée. Afin de choisir le meilleur voisin de \mathbf{S} , on évalue la fonction coût J en un nombre fini de points \mathbf{S}_i dans $V(\mathbf{S})$, et on retient le meilleur voisin qui améliore la valeur de la fonction coût, ce qui permet à cette technique de

sortir d'un minimum local. Pour éviter le risque de revenir vers une solution déjà visitée, l'algorithme conserve la trace des dernières meilleures solutions dans une liste de longueur L donnée, appelé la *liste tabou*. Une nouvelle solution n'est acceptée que si elle n'appartient pas à cette liste tabou. Ce qui permet de diriger l'exploration de la méthode vers des régions de l'espace des solutions Ψ non encore visitées. La liste tabou est généralement gérée comme une liste "*circulaire*" : à chaque itération la mauvaise solution sera remplacée par la nouvelle solution retenue. Pour cela, on remplace la liste tabou de solutions interdites par une liste de "*transformations interdites*", en interdisant la transformation inverse d'une transformation faite récemment.

Algorithme Recherche Tabou

Générer aléatoirement une solution initiale S_0 dans Ψ ;

$$S^* = S_0 ;$$

$$J^* = J(S_0) ;$$

Tabou = Vide; (liste de solutions de longueur L)

Répéter :

Générer N solutions S_i dans $V(S^) \in \Psi / S_i \notin \text{Tabou}$;*

Calculer $S / J(S) = \min \{J(S_i), i = 1, N\}$;

Ajouter S à la liste Tabou ;

Si $J(S) < J(S^)$ alors $S^* = S$ et $J^* = J(S)$;*

Fin de répéter (si les conditions d'arrêt sont atteintes) ;

•Avantages et Inconvénients

La méthode Tabou est une méthode de recherche locale, et la structure de son algorithme de base est finalement assez proche de celle du recuit simulé. On passera donc de l'un à l'autre facilement, avec l'avantage, par rapport au recuit simulé, d'avoir un paramétrage simplifié : dans un premier temps, le paramétrage consistera d'abord à trouver une valeur indicative t d'itérations pendant lesquelles les mouvements sont interdits. Il faudra également décider d'une stratégie de mémorisation à long terme – sur la qualité des solutions, sur leur récurrence, ou sur leur qualité. L'efficacité de la méthode Tabou fait qu'elle est largement employée dans les problèmes d'optimisation combinatoire : elle a été testée avec succès sur les grands problèmes classiques (voyageur de commerce, ordonnancement d'ateliers).

A.4 Les algorithmes génétiques :

Les algorithmes génétiques sont des techniques de recherche stochastiques, introduites par Holland dans les débuts des années 70, Leur fonctionnement est relativement simple. On part avec une population de solutions potentielles (*chromosomes*) initiales arbitrairement choisies. On évalue leurs performances (*fitness*) relatives sur la base desquelles on crée une nouvelle population de solutions potentielles en utilisant des opérateurs d'évolution simples : la sélection, le croisement et la mutation. On recommence ce cycle jusqu'à ce que l'on trouve une solution satisfaisante. Ces algorithmes se distinguent des autres méthodes stochastiques par les deux principaux points suivants : ils utilisent un codage des paramètres (solutions) et non pas les paramètres eux-mêmes et ils travaillent sur une population de points, au lieu d'un point unique.

Un algorithme génétique se présente en général de la manière suivante :

Générer une population initiale

Répéter :

- *Évaluer les performances de chaque individu (chromosome),*
- *Sélectionner et regrouper les chromosomes par paire selon leurs performances,*
- *générer une nouvelle population par les opérations de croisement, et de mutation),*

Fin de répéter (si les conditions d'arrêt soient atteintes) ;

● **Avantages et inconvénients**

D'abord, les algorithmes génétiques sont coûteux en temps de calcul, puisqu'ils manipulent plusieurs solutions simultanément. C'est le calcul de la fonction de performance qui est le plus pénalisant, et on optimise généralement l'algorithme de façon à éviter d'évaluer trop souvent cette fonction.

L'ajustement d'un algorithme génétique est également délicat. L'un des problèmes les plus caractéristiques est celui de la dérive génétique, qui fait qu'un bon individu se met, en l'espace de quelques générations, à envahir toute la population. On parle dans ce cas de convergence prématurée, qui revient à lancer à une recherche locale autour d'un minimum, qui n'est pas forcément l'optimum attendu. Les méthodes de sélection proportionnelle peuvent en particulier favoriser ce genre de dérive. Un autre problème surgit lorsque les différents individus se mettent à avoir des performances similaires : les bons éléments ne sont alors plus sélectionnés, et l'algorithme ne progresse plus.

Les techniques d'optimisation stochastiques sont des heuristiques établies permettant d'approcher des solutions à des problèmes complexes pour lesquels les techniques déterministes ne sont pas applicables ou moins efficaces. Ces techniques n'assurent pas une solution exacte du problème mais présentent l'avantage d'éviter les optimums locaux en faisant une exploration globale dans l'espace des solutions. Elles aboutissent souvent à une très bonne approximation de l'optimum global.

Leur principe consiste à faire des tirages aléatoires uniformément repartis dans le domaine de recherche jusqu'à l'obtention d'une solution satisfaisante. Ce principe favorise un traitement général des problèmes d'optimisation indépendamment de la nature de la physique de ces problèmes. Cependant, l'efficacité de ces techniques dépend de certains paramètres qui doivent être adaptés selon la dimension du problème à résoudre.

Annexe B :
Interpolations,
courbes de bézier,
 β -spline et spline
cubique.

Cet annex est consacré à l'étude des courbes d'interpolation et d'approximation. Notre but est de trouver des courbes très régulières passant par un grand nombre de points. On peut mesurer la régularité d'une fonction par le biais de ses dérivées. En effet, plus une fonction est différentiable, plus la courbe qui lui est associée est lisse et plus La fonction est régulière.

Nous allons présenter dans cette partie les courbes de Bézier et enfin les B_splines, et enfin les interpolations par splines cubique.

B.1 Courbes de Bézier

Soit $M(t)$ le Barycentre du segment $(A, 1-t) (B, t)$.

t est la **proportion du segment** [AB] où se situe le point $M(t)$:

$$t = 0 \Rightarrow M = A$$

$$t = 0,5 \Rightarrow M = \text{milieu de [AB]}$$

$$t = 1 \Rightarrow M = B$$

Quand t parcourt l'intervalle $[0,1]$, il est clair que le point $M(t)$ décrit tout le segment [AB].

Le segment [AB] est la Courbe de Bézier de degré 1 avec points de contrôle A et B .

Les Polynômes $1-t$ et t sont les Polynômes - Poids de Bernstein de degré 1.

Construisons une courbe paramétrée en rajoutant une **2ème étape** à ce qui précède :

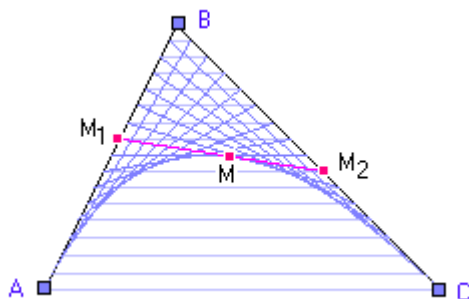
1ère étape :

- Soit $M_1(t)$ le Barycentre de $(A, 1-t) (B, t)$; $M_1(t)$ décrit [AB].

- Soit $M_2(t)$ le Barycentre de $(B, 1-t) (C, t)$; $M_2(t)$ décrit [BC].

2ème étape :

- Soit $M(t)$ le Barycentre de $(M_1, 1-t) (M_2, t)$.



(FigB.1.)

B.1 .1Propriété de la construction

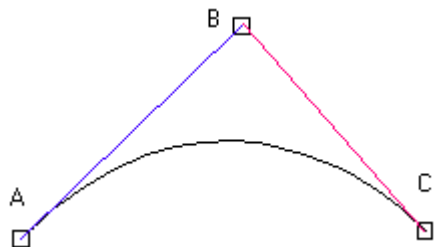
La courbe obtenue est l'enveloppe des segments $[M_1 M_2]$: en tout point M , la tangente à la

courbe est le segment $[M_1 M_2]$.

$M(t)$ décrit alors une Courbe de Bézier de degré 2, qui, par construction :

commence en A et finit en C, et a pour tangentes (AB) en A et (BC) en C.

C'est en fait un arc de Parabole (que nous pourrions noter très logiquement $[ABC]$) :



(FigB.2)

Les propriétés d'association du Barycentre nous permettent d'exprimer $M(t)$ plus directement :

$M(t)$ est ainsi le Barycentre de $(A, (1-t)^2) (B, 2t(1-t)) (C, t^2)$.

$M(t)$ décrit la Courbe de Bézier de degré 2 avec 3 points de contrôle A, B et C.

Les Polynômes $(1-t)^2$, $2t(1-t)$ et t^2 sont les Polynômes - Poids de Bernstein de degré 2

Construisons une courbe paramétrée en rajoutant une 3ème étape à ce qui précède :

1ère étape : 3 Courbes de Bézier de degré 1 :

- Soit $M_1(t)$ le Barycentre de $(A, 1-t) (B, t)$; $M_1(t)$ décrit $[AB]$.
- Soit $M_2(t)$ le Barycentre de $(B, 1-t) (C, t)$; $M_2(t)$ décrit $[BC]$.
- Soit $M_3(t)$ le Barycentre de $(C, 1-t) (D, t)$; $M_3(t)$ décrit $[CD]$.

2ème étape : 2 Courbes de Bézier de degré 2 :

- Soit $N_1(t)$ le Barycentre de $(M_1, 1-t) (M_2, t)$
- Soit $N_2(t)$ le Barycentre de $(M_2, 1-t) (M_3, t)$

3ème étape : 1 Courbe de Bézier de degré 3 :

- Soit $M(t)$ le Barycentre de $(N_1, 1-t) (N_2, t)$

Les propriétés d'association du Barycentre nous permettent d'exprimer $M(t)$ plus directement :

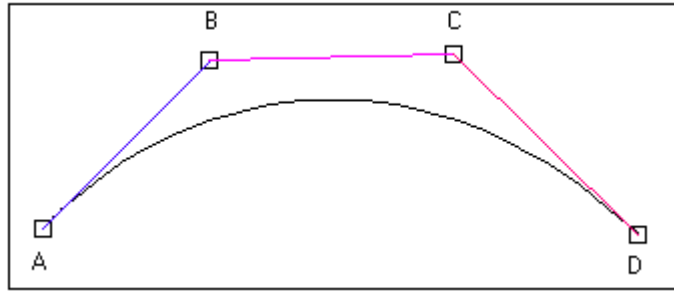
$M(t)$ est ainsi le Barycentre de $(A, (1-t)^3) (B, 3t(1-t)^2) (C, 3t^2(1-t)) (D, t^3)$.

$M(t)$ décrit la Courbe de Bézier de degré 3 avec 4 points de contrôle A, B, C et D.

Elle part de A pour finir en D.

C'est en fait un arc de Cubique (que nous pourrions noter très logiquement $[ABCD]$) :

Les Polynômes $(1-t)^3$, $3t(1-t)^2$, $3t^2(1-t)$ et t^3 sont les Polynômes - Poids de Bernstein de degré 3.



(FigB.3)

B.2 Les Courbes B-Splines définies à partir des Courbes de Bézier

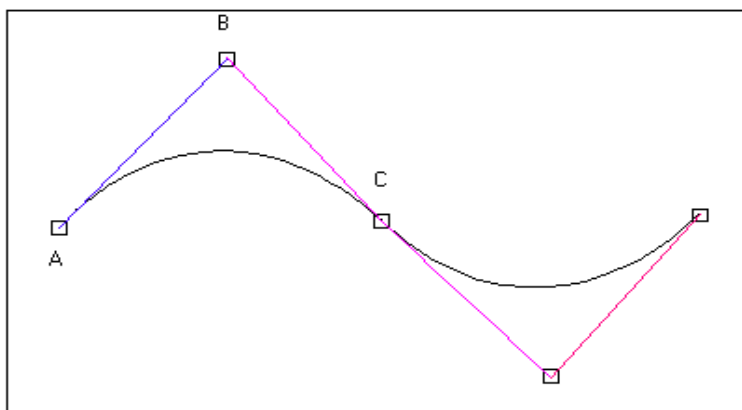
Les courbes B-Splines ont été inventées par Schoenberg dès 1947, qui les définit de façon purement analytique, par morceaux, avec conditions imposées de régularité.

Une B-Spline Uniforme, considérée uniquement du point de vue géométrique, peut être sommairement définie comme une suite de Courbes de Bézier accolées les unes aux autres de façon à ce que les raccordements entre elles soient suffisamment lisses aux points de raccord

Exemple :

B-spline Uniforme de degré 2, construite à partir de deux courbes de Bézier de degré 2, notées $[ABC]$ et $[CDE]$:

Pour avoir l'égalité des vitesses au point de raccordement C, il faut et il suffit que C soit le milieu de $[BD]$:



(FigB.4)

B.2 .1 Correspondance des Points de Contrôle Spline - Bézier :

Degré 2 : permet de comprendre la construction de B-spline Uniforme à partir de Bézier

Les points de contrôle A, B, C de Bézier sont remplacés par P, Q, R tels que :

$Q = B$, seul point originel conservé .

$P =$ symétrique de B par rapport à A .

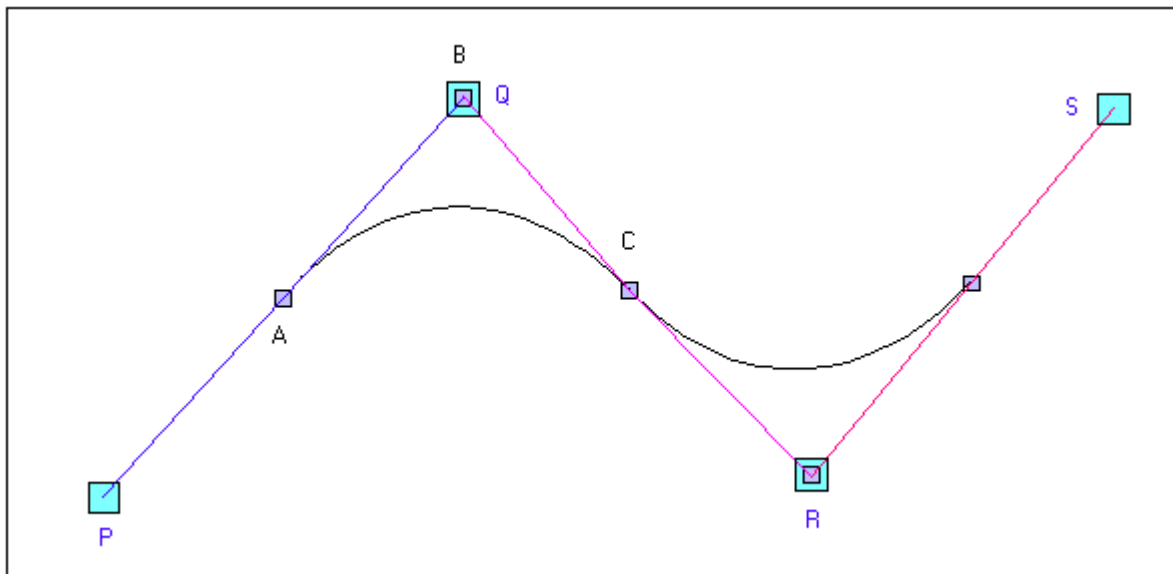
$R =$ symétrique de B par rapport à C .

Ainsi le segment $[PQ]$ définit à la fois le point de départ (son milieu) et le vecteur vitesse (donc la tangente) en ce point.

De même, le segment $[QR]$ définit à la fois le point d'arrivée (son milieu) et le vecteur vitesse (donc la tangente) en ce point.

Considérons alors la courbe de Bézier définie par les "nouveaux" points de contrôle Q, R, S :

le segment $[QR]$ définit à la fois le point de départ (son milieu) et le vecteur vitesse (donc la tangente) en ce point : le raccord avec la courbe de Bézier précédente est continu et avec le même vecteur vitesse (donc la même tangente). La réunion de ces deux Courbes de Bézier définit ainsi la **B-spline Uniforme** de degré 2 de points de contrôle P, Q, R et S :



(FigB.4)

B.3 Spline cubique

Les splines cubiques représentent un compromis très intéressant entre la régularité de la courbe obtenue et le degré des polynômes utilisés.

On considère donc, ici encore, $(n + 1)$ points d'interpolation $(x_i, f(x_i))$, $i=0, 1, 2, \dots, n$ par les quels on souhaite faire passer une courbe autant de fois différentiable que possible. Dans chaque intervalle $[x_i, x_{i+1}]$ (de longueur $h_i = x_{i+1} - x_i$), nous allons utiliser un polynôme de degré 3 de la forme :

$$P_i(x) = f_i + f'_i(x-x_i) + \frac{f''_i}{2!}(x-x_i)^2 + \frac{f'''_i}{3!}(x-x_i)^3 \quad \text{pour } i=0, 1, 2, \dots, n-1 \quad (1)$$

et relier ces différents polynôme de façon à ce que la courbe résultante soit deux fois différentiable. La situation est décrite à la figure 2 pour $n = 4$. C'est l'interpolation par splines cubiques. les coefficients à déterminer (f_i, f'_i, f''_i et f'''_i), qui sont alors respectivement les valeurs de la spline et de ses trois premières dérivées en x_i , avec $p_i(x_i)=f_i, p'_i(x_i)=f'_i, p''_i(x_i)=f''_i$ et enfin $p'''_i(x_i)=f'''_i$.

Puisque l'on a $(n + 1)$ points d'interpolation, il y a n intervalles $[x_i, x_{i+1}]$ qui résultent en $4n$ coefficients inconnus (f_i, f'_i, f''_i et f'''_i pour $i = 0, 1, 2, \dots, n - 1$). Nous allons en effet exprimer toutes ces inconnues en fonction des dérivées secondes f''_i aux nœuds . On complète donc notre ensemble d'inconnues en introduisant la dérivée seconde f''_n au nœud x_n de sorte que nous aurons un grand total de $4n + 1$ inconnues que nous réduirons en un système de dimension $n + 1$.

Puisque il ya trous de formules mathématiques concernant le spline cubique on a essayé de simplifier beaucoup de chose, donc on présente ici toute les formules mathématiques qui sont néssésaire pour calculer le spline , les équations sont :

$$\frac{1}{2} f''_i + 2f''_{i+1} + \frac{1}{2} f''_{i+2} = 6(f[x_i, x_{i+1}, x_{i+2}]) \quad (2)$$

pour $i = 0, 1, 2, \dots, n - 2$

La manière la plus simple de compléter le système d'équations consiste à imposer les valeurs des dérivées secondes aux deux extrémités soit $p''_0(x_0)= a$ et $p''_{n-1}(x_n)=b$ aux deux extrémités (en supposant toujours que nous les connaissions).

$$f''_0 = a \text{ et } f''_n = b \quad (3)$$

$$f''_0 = f''_1 \text{ et } f''_{n-1} = f''_n \quad (4)$$

On impose aussi les valeurs des dérivées premières aux deux extrémités soit $p'_0(x_0) = a$ et $p'_{n-1}(x_n) = b$ aux deux extrémités (en supposant toujours que nous les connaissons). On a donc

$$2f''_0 + f''_1 = -\frac{6}{h_0} (f[x_0, x_1] - a)$$

$$p'_{n-1}(x_n) = f[x_{n-1}, x_n] + \frac{hn-1}{6} f''_{n-1} + 2f''_n \quad (5)$$

Imposer $b = p'_{n-1}(x_n)$ revient alors à :

$$f''_{n-1} + 2f''_n = \frac{6}{hn-1} (b - f[x_{n-1}, x_n])$$

et on doit donc ajouter au système 2 les équations :

- $2f''_0 + f''_1 = -\frac{6}{h_0} (f[x_0, x_1] - a) \quad (6)$
- $f''_{n-1} + 2f''_n = \frac{6}{hn-1} (b - f[x_{n-1}, x_n])$

on impose la continuité de la troisième dérivée soit :

$$p'''_0(x_1) = p'''_1(x_1) \text{ ainsi que } p'''_{n-2}(x_{n-1}) = p'''_{n-1}(x_{n-1})$$

ce qui revient à :

$$f'''_0 = f'''_1 \text{ et } f'''_{n-2} = f'''_{n-1}$$

On doit donc ajouter au système 2 les équations :

$$h_1 f''_0 - (h_0 + h_1) f''_1 + f''_2 = 0 \quad (7)$$

$$h_{n-1} f'''_{n-2} - (h_{n-2} + h_{n-1}) f'''_{n-1} + h_{n-2} f'''_n = 0$$

Pour effectuer une interpolation à l'aide des splines cubiques, il faut en premier lieu calculer les dérivées secondes f''_i on résolvant le système 2 complété par les conditions aux extrémités 3, 4,6,7.

Par la suite, on doit déterminer l'intervalle dans lequel se situe le point d'interpolation x et calculer le polynome dans cet intervalle en utilisant la formule 1 dans laquelle on remplace :

$$f_i = f(x_i)$$

$$f'_i = f[x_i, x_{i+1}] - \frac{hif''_i}{3} - \frac{hif''_{i+1}}{6} \quad (8)$$

$$f'''_i = \frac{f''_{i+1} - f''_i}{hi}$$

Annexe C :
Caractéristiques
numériques des
tâches traitées.

C.1 CARACTERISTIQUES NUMERIQUES POUR TACHE GPP**C.1.1 ESPACE DE TRAVAIL**

Corps	Qmin	Qmax	
1	-94	94	(qb1 en rad)
2	-22	87	(qb2 en rad)
3	0	500	(x en cm)
4	0	500	(y en cm)

C.1.2 LA CONFIGURATION INITIALE ET FINALE

Corps	Qinit	Qfin	
1	0	1	(qb1 en rad)
2	0	-0.2	(qb2 en rad)
3	0	0	(x en cm)
4	0	400	(y en cm)
5	90	90	(Tetap en rad)

C.1.3 LES VITESSES MAXIMALES en rad/s

Corps	Qpmax	
1	0.4	(qb1)
2	0.4	(qb2)
3	1	(roue gauche)
4	1	(roue droite)

C.1.4 LES ACCELERATIONS MAXIMALES en rad/s²

Corps	Qsemax	
1	0.1	(qb1)
2	0.1	(qb2)

3	0.5 (roue gauche)
4	0.5 (roue droite)

Références Bibliographiques

- [1] I.Akli,"Elaboration d'une stratégie de coordination de mouvement pour un Manipulateur Mobile redondant ",Memoire de magister,USTHB, 2007.
- [2] T.Chettebi and HE. Lehtihet, 2002 ,"A new approach for point to point optimal motion planning problems of robotics manipulators" .In Proc .6 th Beinnial Conf .on Engineering Systems Desing and Analysis , APM10.
- [3] J.Foret,O.Bruneau,JB.Fontaine, "Unified approach for m-stability analysis and control of legged robots",2003.
- [4] D.Filliat,"Robotique Mobile-Cours C10-2",Ecole Nationale Supérieure des Techniques Avancées (ENSTA) ,France,2005.
- [5] G.Foulon,"Génération de mouvements coordonnés pour un ensemble constitué d'une plateforme mobile à roues et d'un bras manipulateur", thèse de doctorat , LAAS-CNRS, Institut National des Sciences Appliquées,Toulouse ,France.1998.
- [6] M.Fruchard,"Méthodologie pour la commande de manipulateurs mobiles non-holonomes",thèse de doctorat ,Ecole nationale supérieure des mines de paris- Sophia antipolis , préparée à l'INRIA Antipolis,2005.
- [7] J.Ghommam,J.poisson,"Motion coordination control of multiple marine craft",10 IA Workshop on Advanced Motion Control,2008,AMC 08.Publication Date:26-28 March 2008.
- [8] M.Haddad,"Exploitation optimale des plateformes mobiles et des manipulateurs mobiles à roues. Extension de l'approche des profils aléatoires de trajectoires",thèse de doctorat,Ecole Militaire Polytechnique,Alger,2008.
- [9] M.C.Belaidi Programmation en Delphi,Collection le champion 2000.
- [10] N.Morette,"Contribution à la navigation de robots mobiles :approche par modèle directe",thèse de doctorat,Université d'Orléans,2009.
- [11] V.Padois,"Enchaînement dynamique de tâches pour des manipulateurs mobiles à roues",thèse de doctorat,Laboratoire Génie de Production de l'Ecole Nationale d'Ingénieurs de Tarbes ,Institut national polytechnique de Toulouse,France,2005.
- [12] Pontryagin, L., Boltianski, V., Gamkrelidze, R., Michtchenko, E., 1965. "Théorie mathématique des processus optimaux". Ed. Mir.

Références Internet

[W-amt] :<http://www.robot amateur.com>.

[W-ensta] :<http://www.esnta.fr/filliat/cours/polys/polyrobotiquemobile>.

Références Bibliographiques

[W-unilim] : [http://www.unilim.fr/pages perso/jean.debord](http://www.unilim.fr/pages_perso/jean.debord).



Titre : «Génération de trajectoire pour un Robot Manipulateur Mobile pour tâche généralisée point à point. Extension de l'approche des profils aléatoires de trajectoires»

Résumé

Le présent travail a pour but la résolution du problème d'exploitation optimale de manipulateurs mobile à roues portant un bras manipulateur à deux degrés de libertés, pour l'exécution de tâche généralisée point à point, avec prise en compte des contraintes géométriques (les limites des débattements, ...), cinématiques (les capacités limites en vitesse et en accélération des roues) . Nous avons conçu un algorithme de génération de trajectoire existant, en optimisant le temps. mais pour ce type de problèmes, il existe une infinité de trajectoires permettant d'atteindre chaque point de la trajectoire vue que le robot est caractérisé par sa redondance pour cela nous avons proposé une extension de l'approche des profils aléatoires de trajectoires (RPA). Elle est basée sur la transformation du problème original en un problème d'optimisation paramétrique sous contraintes. elle utilise le processus de stochastique d'optimisation (recuit simulé) afin de mesurer la meilleure performance et obtenir une trajectoire optimale $q_{best}(t)$, se déroulant pendant l'intervalle de temps $[0, T_{best}]$.

Mots clés : génération de trajectoire , planification de trajectoires, manipulateur mobile, optimisation, recuit simulé.