

UNIVERSITE SAAD DAHLAB DE BLIDA

Faculté des Sciences de l'Ingénieur
Département de Génie Mécanique

MEMOIRE DE MAGISTER

Spécialité : Construction mécanique

SUR LA DYNAMIQUE DES ROBOTS COOPERANTS

« PLANIFICATION DE TRAJECTOIRES »

Par :

AICHA Mohamed Yacine

Devant le jury composé de :

M ^r . A. AIAD	Maître de conférence, U. Blida.	Président
M ^r . A. BELAIDI	Maitre de conférence, U. de Boumerdes	Examineur
M ^r . M. MEGHATRIA	Chargé de cours, U. de Blida	Examineur
M ^r . M. OUALI	Professeur, U. de Blida	Rapporteur

Blida, Mars 2010

RÉSUMÉ

Dans ce mémoire, on a abordé la modélisation des bras manipulateurs dans un contexte général. L'objectif étant de proposer les différents algorithmes régissant la manipulation d'objets pour un transfert entre deux situations extrêmes données, en utilisant un seul ou plusieurs robots. On a procédé aux différents modèles régissant les positions, vitesses et accélérations des liens du robot pour appliquer l'algorithme de « *Newton-Euler* » et déterminer les réactions des articulations des liens de chaque robot contribuant à la manipulation. Les algorithmes ont été exécutés pour les structures de robots les plus connues et dans un ordre de complexité croissant. Les problèmes de résolution du modèle géométrique inverse, la planification de trajectoire et la génération de mouvement ont été abordés à travers un exemple de manipulation d'un objet par deux robots plans à deux degrés de liberté.

Mots clés : Robots sériels, Modélisation des robots, coopération de robots, modélisation dynamique des robots.

ABSTRACT

In this report, we approached the modelling of the manipulator arms on a general context. The objective is to propose the various algorithms governing the manipulation of objects for a transfer between two given extreme situations, by using the only one or several robots. We proceeded to the various models governing the positions, the speeds and the accelerations of the robot's links to apply the algorithm of "*Newton-Euler*" and determine the link's reactions of every robot contributing to the manipulation. The algorithm is executed for the structures of the most known robots and in an increasing order of complexity. The problems of resolution of the inverse geometrical model, the planning of trajectory and the generation of movement is approached through an example of manipulation of an object by two robots plans(shots) in two degrees of freedom.

Keywords : Serial Robots, Modelling of robots, Robots Cooperation, Robots dynamic modelling.

ملخص

في هذه المذكرة، تم تناول نمذجة الروبوتات بصفة عامة، بهدف إقتراح خوارزميات مختلفة تتناول حمل الأشياء و نقلها بين وضعيتين مختلفتين، وذلك باستعمال روبوت واحد أو عدة روبوتات. لقد تطرقنا إلى مختلف مختلف النماذج الممثلة للمواقع، السرعات و التسارعات و ذلك لتطبيق خوارزمية " نيوتن أولير" و تحديد ردود أفعال مختلف وصلات كل روبوت مساهم في عملية النقل. و قد تم تطبيق هذه الخوارزميات على مختلف هياكل الروبوتات المعروفة و تم إختيارها حسب تعقيد تصاعدي. بالنسبة لحلول النماذج الهندسية العكسية، تخطيط المسار و توليد الحركة، فقد تمت مناقشتها من خلال مثال لروبوت ذات درجتي حرّية.

كلمات مفتاحية : الروبوتات المتسلسلة، نمذجة الروبوتات، الروبوتات المتعاونة، ديناميكا الروبوتات.

REMERCIEMENT

الحمد لله الذي هدانا لهذا وما كنا لنهتدي لولا أن هدانا الله

Ce travail a été élaboré au département de mécanique de l'université SAAD DAHLAB – Blida. Je saisi cette opportunité pour remercier tous mes enseignants d'avoir contribué à enrichir mon esprit de leurs connaissances scientifiques.

Que mon Professeur, Monsieur Mohammed OUALI, trouve ici l'expression de mes meilleurs sentiments d'avoir dirigé ce travail.

Je remercie également tous les membres du jury pour avoir accepté de juger ce mémoire.

Enfin, ma chaleureuse reconnaissance est adressée à tous ceux qui m'ont aidé de loin ou de près à réaliser ce travail.

Je dédie ce modeste travail à mes parents et mes beaux-parents et leur petite famille, ma femme et mes enfants Yakoub, Sid Ahmed et Zineb.

SOMMAIRE

INTRODUCTION

CHAPITRE 1 : GÉNÉRALITÉS & ÉTAT DE L'ART.

1.1.	Introduction.....	17
1.2.	Historique de la technologie robotique [2].	17
1.3.	Composants d'une cellule robotisée.....	19
1.4.	Description des robots.....	19
1.5.	Constituants mécaniques d'un robot manipulateur.....	20
1.5.1.	Organe terminal.	20
1.5.2.	Structure mécanique articulée.....	21
1.5.2.1.	Couples cinématiques.....	22
1.5.2.2.	Chaînes cinématiques.....	26
1.6.	Architecture des robots manipulateurs.	27
1.6.1.	Structures générales des robots.	27
1.6.2.	Robots sériels.	30
1.6.3.	Robots parallèles.	30
1.7.	Définitions et terminologie.	32
1.7.1.	Espace opérationnel & Espace articulaire.....	32
1.7.2.	Redondance.....	33
1.7.3.	Singularités.	33
1.7.4.	Parcourabilité.	34
1.7.5.	Dextérité.....	35
1.7.6.	Généricité.....	35
1.8.	Coopération de robots.....	35
1.9.	Conclusion.....	38

CHAPITRE 2 : MODÉLISATION DES BRAS MANIPULATEURS.

2.1.	Introduction.....	39
2.2.	Formalismes de description des robots.....	40

2.2.1.	Transformation de coordonnées.	40
2.2.2.	Description des paramètres d'une chaîne articulée.	41
2.2.2.1.	Notation de Sheth-Uiker.	41
2.2.2.2.	Notation de Kleinfinger-Khalil.	42
2.2.2.3.	Notation de Denavit-Hartenberg.	43
2.2.3.	Matrice jacobienne.	45
2.3.	Modélisation géométrique.	45
2.3.1.	Modèle géométrique direct M.G.D.	46
2.3.2.	Modèle géométrique inverse M.G.I.	46
2.4.	Modélisation cinématique.	47
2.5.	Cinématique « <i>Directe & Inverse</i> » - Méthodes de résolution.	48
2.5.1.	Résolution du modèle géométrique inverse.	48
2.5.2.	Formation de la matrice jacobienne.	48
2.5.3.	Calcul pratique des vitesses & accélérations opérationnelles [1].	49
2.6.	Modélisation dynamique.	50
2.6.1.	Formalisme de Lagrange.	51
2.6.1.1.	Calcul des éléments de A , C et Q [1].	51
2.6.2.	Efforts exercés par l'organe terminal sur son environnement [1].	52
2.6.3.	Formalisme de Newton-Euler.	52
2.6.3.1.	Principe du formalisme de Newton Euler.	52
2.6.3.2.	Algorithme de Newton-Euler [1].	53
2.7.	Situation d'un robot dans son environnement.	55
2.8.	Modélisation de l'objet à manipuler.	55
2.8.1.	Sphère d'encombrement de l'objet.	56
2.8.2.	Positionnement de l'objet dans l'espace opérationnel.	57
2.8.3.	Prise de l'objet.	57
2.8.4.	Positionnement et cinématique d'un objet saisi par un ou plusieurs robots.	58
2.8.5.	Modèle dynamique de l'objet.	60
2.8.6.	Objet manipulé par plusieurs robots – Coopération de robots.	62
2.9.	Conclusion.	66

CHAPITRE 3 : TRAJECTOIRES & GÉNÉRATION DE MOUVEMENT.

3.1.	Introduction.	67
3.2.	Divers aspects de la planification de trajectoires.	67
3.3.	Classification des types de méthodes utilisées.	69

3.3.1.	Méthodes globales.	69
3.3.2.	Méthodes locales.	70
3.3.3.	Méthodes mixtes.	70
3.3.4.	Autres méthodes.	70
3.4.	Commandabilité, planification & suivi de trajectoire.	71
3.5.	Critère de performance dans la recherche de trajectoires.	72
3.6.	Espace de planification de la trajectoire.	74
3.6.1.	Génération de mouvement et système de commande [1].	74
3.6.2.	Planification de trajectoire dans l'espace articulaire.	75
3.6.2.1.	Trajectoire sans contraintes – Mouvement point à point (P.T.P).	76
3.6.2.2.	Trajectoire entre deux points avec des points intermédiaires.	78
3.6.3.	Planification de la trajectoire dans l'espace opérationnel.	78
3.6.3.1.	Interpolation cartésienne.	79
3.6.3.2.	Interpolation articulaire.	79
3.7.	Théorie des graphes et chemins optimaux.	79
3.8.	Conclusion.	81

CHAPITRE 4 : APPLICATIONS.

4.1.	Introduction.	82
4.2.	Notations.	82
4.3.	Descriptions des structures mécaniques étudiées.	84
4.3.1.	Considérations générales.	84
4.3.2.	Organes terminaux utilisés.	85
4.3.3.	Mécanisme plan à 2 ddl.	86
4.3.4.	Mécanisme Plan à 4 ddl.	87
4.3.5.	Structure plane à base articulée.	89
4.3.6.	Robot STAUBLI RX 90.	90
4.3.7.	Robot SCARA.	93
4.3.8.	Robot PUMA.	95
4.4.	Application du modèle géométrique direct MGD pour analyser la trajectoire de l'organe terminal.	97
4.4.1.	Paramètres cinématiques des actionneurs.	97
4.4.2.	Application du modèle géométrique direct pour un robot plan à 2 ddl.	97
4.4.3.	Application du modèle géométrique direct pour un robot plan à 4 ddl.	100

4.4.4.	Application du modèle géométrique directe pour une structure plane à base articulée.....	102
4.4.5.	Application du modèle géométrique direct pour une structure STAUBLI RX 90	104
4.4.6.	Application du modèle géométrique direct pour une structure SCARA.	105
4.4.7.	Application du modèle géométrique direct pour une structure PUMA.	107
4.5.	Application du modèle géométrique inverse pour un robot plan à deux ddl. par une méthode géométrique.	108
4.6.	Application du modèle dynamique pour un mouvement à vide & comparaison de résultats.	110
4.7.	Application du modèle dynamique pour la manipulation d'un objet par un seul robot	111
4.7.1.	Description de l'objet à manipuler.	111
4.7.2.	Manipulation de l'objet par un robot plan à 2ddl.....	112
4.8.	Application du modèle dynamique pour la manipulation d'un objet par deux robots coopérants.	113
4.8.1.	Description de l'objet à manipuler.	113
4.8.2.	Description de la scène de la manipulation.	114
4.8.3.	Description du mouvement de l'objet.	115
4.9.	Conclusion.....	119

CONCLUSION

REFERENCES

APPENDICES

LISTE DES FIGURES

Figure 1. 1 : Pince parallèle à mâchoires.....	20
Figure 1. 2 : Pince à deux doigts.....	21
Figure 1. 3 : Structure d'un poignet sphérique.....	21
Figure 1. 4 : Déplacements possibles (degrés de liberté) d'un solide dans l'espace.....	22
Figure 1. 5 : Exemples de chaînes cinématiques.....	26
Figure 1. 6 : Architecture des porteurs [1].....	28
Figure 1. 7 : Architecture des poignets [1].....	29
Figure 1. 8 : Structure générale d'un robot manipulateur [1].....	29
Figure 1. 9 : Exemple de structure parallèle « Robot DELTA » [6].....	32
Figure 1. 10 : Aspects et postures du manipulateur du type RR.....	34
Figure 1. 11 : Singularité parallèle du manipulateur de type 3PRR.....	34
Figure 1. 12 : Exemples de robots coopérants.....	37
Figure 2. 1 : Définition des repères pour une transformation homogène.....	40
Figure 2. 2 : Définition des paramètres géométriques suivant Sheth-Uiker [8].....	41
Figure 2. 3 : Définition des paramètres géométriques suivant Kleinfinger-Khalil [8].....	42
Figure 2. 4 : Définition des paramètres géométriques suivant Denavit-Hartenberg [8].....	43
Figure 2. 5 : Matrice jacobienne d'un manipulateur planaire à trois degrés de liberté.....	49
Figure 2. 6 : Repères associés à un objet manipulé par un ou plusieurs bras manipulateur.	57
Figure 2. 7 : Objet manipulé par un robot.....	61
Figure 2. 8 : Isolation d'un objet manipulé par un robot.....	61
Figure 2. 9 : Coopération de deux robots pour la manipulation d'un objet.....	62
Figure 2. 10 : Isolation d'un objet manipulé par deux robots.....	62
Figure 2. 11 : Distribution de la charge d'un objet manipulé par plusieurs robots.....	64
Figure 3. 1 : Principe de la détermination de la trajectoire de référence.....	72
Figure 3. 2 : Suivi d'une trajectoire.....	72
Figure 3. 3 : Système de commande pour la génération de mouvement dans l'espace articulaire.....	74

Figure 3. 4 : Système de commande pour la génération de mouvement dans l'espace opérationnel.	75
Figure 3. 5 : Évolution des positions, vitesses et accélérations sur l'articulation j avec une loi trapèze.	77
Figure 3. 6 : Description des différentes phases pour une tâche « <i>pick & place</i> ».	78
Figure 4. 1 : Description du repère atelier.	84
Figure 4. 2 : Moments d'inerties pour une barre.	85
Figure 4. 3 : Positionnement du point de saisie de l'organe terminal.	85
Figure 4. 4 : Description d'un mécanisme plan à deux degrés de liberté.	86
Figure 4. 5 : Description d'un mécanisme plan à quatre degrés de liberté.	87
Figure 4. 6 : Description d'une structure plane à base articulée.	89
Figure 4. 7 : Structure du robot STAUBLI RX 90.	91
Figure 4. 8 : Structure d'un robot du type SCARA.	93
Figure 4. 9 : Structure d'un robot du type PUMA.	95
Figure 4. 10 : Positions articulaires appliquées au robot plan à 2 ddl.	98
Figure 4. 11 : Vitesses articulaires appliquées au robot plan à 2 ddl.	98
Figure 4. 12 : Trajectoire du point de saisie de l'organe terminal du robot à 2 ddl.	99
Figure 4. 13 : Vitesses opérationnelles du point de saisie de l'organe terminal du robot à 2 ddl.	99
Figure 4. 14 : Accélérations opérationnelles du point de saisie de l'organe terminal du robot plans à 2 ddl.	100
Figure 4. 15 : Positions articulaires appliquées au robot plan à 4 ddl.	101
Figure 4. 16 : Vitesses articulaires appliquées au robot plan à 4 ddl.	101
Figure 4. 17 : Trajectoire du point de saisie de l'organe terminal du robot à 4 ddl.	101
Figure 4. 18 : Vitesses opérationnelles du point de saisie de l'organe terminal du robot à 4 ddl.	102
Figure 4. 19 : Accélérations opérationnelles du point de saisie de l'organe terminal du robot à 4 ddl.	102
Figure 4. 20 : Paramètres cinématiques articulaires appliquées à la structure plane à base articulée.	103
Figure 4. 21 : Trajectoire du point de saisie de la structure plane à base articulée.	103
Figure 4. 22 : Vitesses et accélérations opérationnelles du point de saisi de la structure plane à base articulée.	104
Figure 4. 23 : Paramètres cinématiques articulaires appliquées à la structure STAUBLI RX 90.	104
Figure 4. 24 : Trajectoire du point de saisie de la structure STAUBLI RX 90.	105

Figure 4. 25 : Vitesses et accélérations opérationnelles du point de saisi de la structure STAUBLI RX 90.	105
Figure 4. 26 : Positions articulaires appliquées à la structure SCARA.	106
Figure 4. 27 : Vitesses articulaires appliquées à la structure SCARA.	106
Figure 4. 28 : Trajectoire du point de saisi de la structure SCARA.	107
Figure 4. 29 : Vitesses et accélérations opérationnelles du point de saisi de la structure SCARA.	107
Figure 4. 30 : Paramètres cinématiques articulaires appliquées à la structure PUMA.	107
Figure 4. 31 : Trajectoire du point de saisi de la structure PUMA.	108
Figure 4. 32 : Vitesses et accélérations opérationnelles du point de saisi de la structure PUMA.	108
Figure 4. 33 : Résolution du modèle géométrique inverse par une méthode géométrique.	109
Figure 4. 34 : Description de l'objet à manipuler par un seul robot.	112
Figure 4. 35 : Analyse dynamique de la manipulation d'un objet par un seul robot plan à 2 ddl.	112
Figure 4. 36 : Description de l'objet à manipuler par deux robots.	113
Figure 4. 37 : Description de la scène de manipulation d'un objet par deux robots plans à 2 ddl.	115
Figure 4. 38 : Description du mouvement d'un objet manipulé par deux robots plans à 2 ddl.	116
Figure 4. 39 : Mouvement rotatif d'un objet en vu d'être manipulé par deux robots plans à 2 ddl.	116
Figure 4. 40 : Positions articulaires des liens des deux robots obtenues en appliquant le modèle géométrique inverse correspondant aux positions occupées par l'objet.	117
Figure 4. 41 : Vitesses articulaires des liens des deux robots obtenues par dérivation numérique des positions articulaires.	117
Figure 4. 42 : accélérations articulaires des liens des deux robots obtenues par une seconde dérivation numérique des positions articulaires.	118
Figure 4. 43 : Réactions des articulations des deux robots coopérants pour la manipulation d'un objet.	118

LISTE DES TABLEAUX

Tableau 1. 1 : Signes conventionnels des couples cinématiques.	24
Tableau 1. 2 : Nombre de morphologies possibles en fonction de nombre de degrés de liberté [1].	27
Tableau 3. 1 : Formes du critère de performance utilisées.	73
Tableau 3. 2 : Comparaison entre la planification de trajectoire dans l'espace articulaire et l'espace opérationnel.	75
Tableau 3. 3 : Récapitulatif des lois utilisée pour la génération de mouvement dans l'espace articulaire.	76
Tableau 4. 1 : Paramètres des liens du robot plan à 4 ddl.	86
Tableau 4. 2 : Paramètres de Denavit-Hartenberg d'un robot plan à 2 ddl.	87
Tableau 4. 3 : Paramètres des liens d'un robot plan à 4 ddl.	88
Tableau 4. 4 : Paramètres de Denavit-Hartenberg d'un robot plan à 4 ddl.	88
Tableau 4. 5 : Paramètres des liens d'une structure plane à base articulée.	89
Tableau 4. 6 : Paramètres de Denavit-Hartenberg d'une structure plane à base articulée. .	90
Tableau 4. 7 : Paramètres des liens d'une structure STAUBLI RX 90.	91
Tableau 4. 8 : Paramètres de Denavit-Hartenberg d'une structure STAUBLI RX 90.	92
Tableau 4. 9 : Paramètres des liens d'une structure SCARA.	93
Tableau 4. 10 : Paramètres de Denavit-Hartenberg d'une structure SCARA.	94
Tableau 4. 11 : Paramètres des liens d'une structure PUMA.	96
Tableau 4. 12 : Paramètres de Denavit-Hartenberg d'une structure PUMA.	96
Tableau 4. 13 : Paramètres des liens du robot plan à 2 ddl.	110
Tableau 4. 14 : Comparaison des résultats pour l'analyse dynamique d'un robot plan à deux degrés de liberté.	110

INTRODUCTION

Représentant un incroyable défi, les robots interprètent un important aboutissement en matière d'évolution des technologies : « *Que peut-il y avoir de plus difficile que de reproduire avec une machine, des caractéristiques du fonctionnement animal ou humain ?* ». Ce défi technologique mobilise depuis au moins un demi siècle des énergies considérables, et chacune des étapes majeures dans l'évolution de la robotique constitue une victoire de l'intelligence humaine.

Depuis le début des années 1970, les recherches dans le domaine de la robotique ont pris une allure croissante mettant en œuvre plusieurs résultats innovants dans d'autres disciplines connexes :

- Mathématiques : Développement des algorithmes efficaces pour la résolution des systèmes complexes ;
- Intelligence artificielle : Domaine connexe au mathématiques qui consiste généralement en l'imitation des systèmes biologiques donnant parfois la possibilité de l'auto apprentissage comme les réseaux de neurones, ainsi que la diversité dans la création de nouveaux systèmes et la sélection pseudo naturelle comme les algorithmes génétiques.
- Électronique : Développement des systèmes « *intelligents* » pouvant être même embarqués sur le robot afin de le doter de la perception nécessaire pour pouvoir prendre certaines décisions concernant son comportement ;
- Informatique : par le développement des langages de programmation de plus en plus évolués.

Entre autres, la modélisation consiste en une étape essentielle de tout sujet de robotique. Que le sujet ait l'aspect d' « analyse » ou de « conception », un modèle mathématique s'avère indispensable. Ce modèle doit dans les meilleurs cas, tenir compte de tous les paramètres qui peuvent influencer sur le comportement du robot, et où le résultat cherché doit figurer dans la liste de ces paramètres.

La complexité du modèle régissant un robot est en fonction du sujet traité. Elle peut varier d'un modèle simple qui ne régie que les positions occupées par le robot, ce modèle consiste en « *le modèle géométrique* », à un modèle complexe qui pourra tenir compte de la puissance nécessaire des moteurs (actuateurs) ou de la résistance mécanique des composants du robot ce qui fera appel au « *modèle dynamique* ». Ce dernier, tenant compte de tous les efforts agissant sur les liens du robot ou même sur l'objet manipulé, utilise à son tour les autres modèles de description du robot : le modèle géométrique pour déterminer les moments des forces et le modèle cinématique pour la détermination des efforts d'inertie.

Quoiqu'on ait pris le modèle géométrique pour un modèle simple, en réalité il représente l'un des sujets les plus complexes en robotique lorsqu'il s'agit du « *modèle géométrique inverse* ». L'absence de linéarité dans le passage entre les espaces « *opérationnel* » et « *articulaire* » ne permet d'utiliser directement les relations de l'algèbre linéaire, ainsi, le caractère de redondance mène à trouver plusieurs solutions dans l'espace articulaire correspondant à une seule situation dans l'espace opérationnel. Une ébauche de recherche bibliographique concernant la résolution du modèle géométrique inverse focalise les heuristiques basées généralement sur des variantes de la méthode de la descente [63], [72], [79], [106], ou des méthodes spécifiques pour certains types connus de robots [77], [111].

Le sujet le plus abordé par les roboticiens et celui des trajectoires. Dans un environnement encombré d'obstacles, le problème de planification d'un chemin menant le robot d'une situation initiale vers une situation finale s'avère parfois très difficile. Plusieurs approches ont été proposées pour la résolution de ce problème, toutes ayant l'aspect de recherche opérationnelle mettant en œuvre des techniques avancées comme « *les algorithmes du plus court chemin* » [66], [67], [70], ou des processus stochastiques [61], [69]. La difficulté de résolution du modèle géométrique inverse se répercute dans la génération de mouvements dans l'espace articulaire pour réaliser la trajectoire trouvée dans l'espace opérationnel. Les contraintes de différentes natures (géométriques, cinématique, dynamique ou même celle du temps de parcours) contribuent pour accentuer la complexité du problème.

Dans certaines situations, constituer une équipe de plusieurs robots s'avère crucial. Ceci peut-en tirer plusieurs profits comme la simplification de la conception où dans une équipe de robots conçus pour exécuter une tâche définie, chaque robot peut être réalisé de

façon beaucoup plus simple qu'une solution constituée d'un unique robot. La coopération de robots a fait l'objet de plusieurs travaux de recherches en robotique, on parle de la coopération multi agents pour désigner la coordination pour les actions distribuées entre plusieurs robots indépendants en action et en commande. Pour le cas de la manipulation d'objets par plusieurs robots, la modélisation doit tenir compte de la distribution de la charge qui est généralement un problème complexe.

Dans ce mémoire, on abordera la modélisation des robots sériels dans un objectif d'étudier le modèle dynamique pour un seul robot et aussi pour le cas d'une coopération de plusieurs robots. Ce mémoire est organisé comme suit : le premier chapitre est consacré à l'état de l'art et à la terminologie robotique. Dans le second chapitre, on présentera la modélisation des bras manipulateurs sériels, les modèles : géométrique, cinématique et dynamique seront abordé en détail pour proposer des algorithmes efficaces et ayant l'aspect généraliste pouvant être exécutés pour toutes les structures de robots sériels. Dans le troisième chapitre on abordera la planification des trajectoires et la génération de mouvement ainsi qu'une synthèse des types de méthodes utilisées. Le dernier chapitre consiste en une application des modèles décrits pour les structures de robots les plus répondues.

CHAPITRE 1

GÉNÉRALITÉS & ÉTAT DE L'ART.

1.1. Introduction.

Le robot a initialement été créé pour décharger l'homme des tâches fastidieuses, par exemple, les tâches de manutention sont souvent fatigantes, répétitives ou même infaisables manuellement. L'idée d'une machine se substituant à l'homme pour de telles tâches était donc séduisante. De ce besoin sont nés des manipulateurs relativement simples et souvent dédiés à une tâche bien définie.

Beaucoup de livres et articles en robotique, débutent avec une définition du mot « *robot* ». Malheureusement, il n'existe pas de définition formelle de ce terme très connu. La plus part des définition qu'on peut trouver dans la littérature tentent de dire quelque chose comme « *le robot est une machine capable à être reprogrammée* ». Le fait que la reprogrammation est aussi importante, elle est donc une caractéristique des robots. Mais cette dernière définition est trop grossière puisqu'elle peut inclure un four à micro-ondes ou une machine à laver vu l'aptitude de ces derniers à être reprogrammés, sachant bien sur, qu'on pensant à une machine à laver, il ne nous vient presque jamais à l'esprit qu'il s'agit d'un robot. La difficulté réside dans le fait que l'idéal soit que les robots seront des machines capables à tout faire ! Donc il est préférables de ne pas placer des limites concernant un robot en général : *que peut-il achever ? Ou comment est ce qu'il est construit ?* A présent la définition du mot robot peut être similaire à celle d'un éléphant ! « *difficile à décrire, mais vous allez le connaître dès que vous allez le voir !* ».

1.2. Historique de la technologie robotique [2].

Dans les années 60, Joe Engleberger a rencontré George Devol. Ce dernier avait juste fait breveter une conception pour un bras mécanique commandé par ordinateur. Engleberger a vu cette invention comme un robot primitif. Les deux d'entre eux ont formé la compagnie « *Unimation* ». L'ancêtre des bras original était le bras télé-actionné développé

dans les années 40. Ces bras « *maître-esclave* » étaient utilisés pour manoeuvrer les matériaux radioactifs derrière un écran de verre. Le « *maître* » humain a été remplacé par un ordinateur, en utilisant la technologie développée pour des machines-outils à commande numérique à M.i.t. Boston. Au début des années 70 Engleberger a visité le Japon, et c'était les japonais qui ont réalisé la première fois un robot industriel.

À l'origine, les robots étaient prévus pour remplacer les ouvriers humains. C'était particulièrement dans l'industrie automobile qui souffrait de divers problèmes. Alors, on a pensé que les robots seraient des machines universelles, capables d'être rapidement reprogrammées pour une grande variété de tâches. C'était ces idées qui ont motivé le développement des robots PUMA par « *Unimation* » dans un projet de recherche pour « *General Motors* ». Le bras du robot PUMA a été étroitement modelé sur le bras humain et pourrait soulever un poids à peu près identique.

Dans la pratique, les robots étaient difficiles à reprogrammer, et n'ont pas pu concurrencer les humains dans les tâches où l'environnement du travail n'était pas donné avec précision, mais c'était un début. Les premières demandes de ces nouvelles machines étaient : la peinture pulvérisante et le soudage. Ici le robot a pu être programmé par un opérateur humain qui donnait à la machine, l'ordonnancement des mouvements des bras avec un langage approprié. Le progrès a été accompli pour des tâches d'assemblage avec le développement du robot SCARA. Ceci a été développé au Japon, pour le montage des composants sur des cartes électroniques.

Autour de ce temps, l'idéologie de la robotique a changé : et au lieu de « *remplacer les ouvriers humains par des robots* » c'est devenue plutôt « *pouvoir faire les travaux dont l'humains ne pouvait pas* », comme le travail dans les environnements dangereux de l'exploitation, des industries nucléaires ou dans la lutte contre l'incendie. Les robots peuvent fonctionner dans les endroits inaccessibles aux humains, dans l'espace extra atmosphérique, sur le fond marin ou même à l'intérieur d'un pipe.

Avec la dépression économique au début des années 80, beaucoup de gens ont soutenu l'idée d'adoption de la nouvelle technologie. Cependant, ceci a mené à la crainte du chômage puisque les robots occupaient les postes des humains, mais les avantages économiques d'installer des robots ont relevé pas mal de défis. D'ailleurs, certainement, les robots n'ont pas besoin de vacances et ne font pas de grèves, mais ils sont seulement des machines. Et comme toutes les machines elles ont besoin d'entretien régulier. En plus,

l'installation des robots dans une usine n'est pas simplement une question de remplacer des ouvriers par des robots. La chaîne de production devra être remodelée, le procédé de production lui-même doit être changé. Souvent le produit devra être modifié de sorte que les robots puissent manoeuvrer les composants facilement. Se débarrasser des ouvriers non qualifiés implique l'emploi des techniciens et des programmeurs fortement qualifié dans ce sens. De tels problèmes ont souvent entravé l'installation des robots dans les usines.

Les problèmes de robotisation des chaînes industrielles ou l'automatisation des process méritent une étude approfondie qu'elle n'a pas été incluse dans le présent mémoire.

1.3. Composants d'une cellule robotisée.

Généralement, une cellule robotisée est constituée des composants suivants :

Le mécanisme : Ayant une structure proche de celle du bras humain, il permet de remplacer ou de prolonger son action. Ses mouvements sont menés par des actionneurs électrique, pneumatiques ou hydrauliques et qui transmettent leurs mouvements aux articulations par des systèmes appropriés.

La perception : Elle permet de gérer les relations entre le robot et son environnement. Les organes de perception sont des capteurs qui peuvent mesurer l'état interne du robot comme les positions et les vitesses des articulations, ou bien d'autres capteurs qui recueillent les informations sur l'environnement comme la détection d'objet, vision artificielle, mesure de distance etc.

La commande : Synthèse des consignes des asservissements pilotant les actionneurs.

Interface homme machine : Dispositif à travers lequel, l'utilisateur programme les tâches que le robot doit exécuter. Ce dispositif peut être une console de commande embarquée au robot, ou un micro-ordinateur distant.

Le poste de travail : L'endroit ou l'environnement dans lequel évolue le robot.

1.4. Description des robots.

Tenant compte de la norme AFNOR, un robot est un manipulateur commandé en position, reprogrammable, polyvalent, 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. Son unité de commande utilise, notamment, un dispositif de mémoire et éventuellement de perception et d'adaptation

à l'environnement et aux circonstances. Ces machines polyvalentes sont généralement étudiées pour effectuer la même fonction d'une façon cyclique et peuvent être adaptées à d'autres fonctions sans modification permanente du matériel [1].

1.5. Constituants mécaniques d'un robot manipulateur.

Un robot manipulateur est constitué par deux sous ensemble distincts : un (ou plusieurs) organe terminal et une structure mécanique articulée.

1.5.1. Organe terminal.

L'organe terminal est un dispositif fixé à l'extrémité mobile de la structure mécanique (bras). Il est destiné à manipuler des objets (dispositifs de serrage, dispositifs magnétiques, à dépression, etc...), ou à les transformer (outil, torche de soudage, pistolet de peinture etc.).

La forme la plus simple des organes terminaux, c'est les pinces, comme montré dans les figures (1.1) et (1.2).

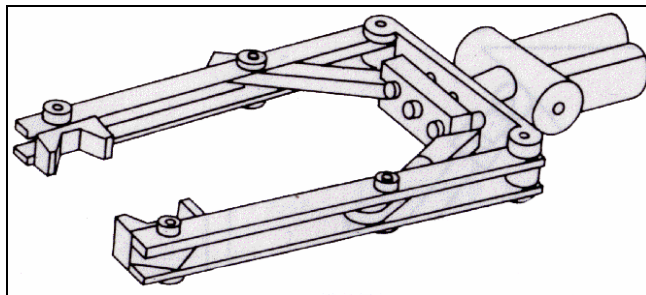


Figure 1. 1 : Pince parallèle à mâchoires.

Ces pinces sont des mécanismes à un degré de mobilité pouvant effectuer l'une des deux actions (ouvrir et fermer). Ceci est adéquat pour la manipulation des objets, mais inadéquat pour d'autres tâches comme la soudure, l'assemblage etc. Beaucoup de recherches ont été menées dans le sens du développement des organes terminaux. Ils existent des mécanismes comprenant plusieurs organes terminaux conçus de telle sorte qu'ils puissent être rapidement changés pour remplir une tâche définie.

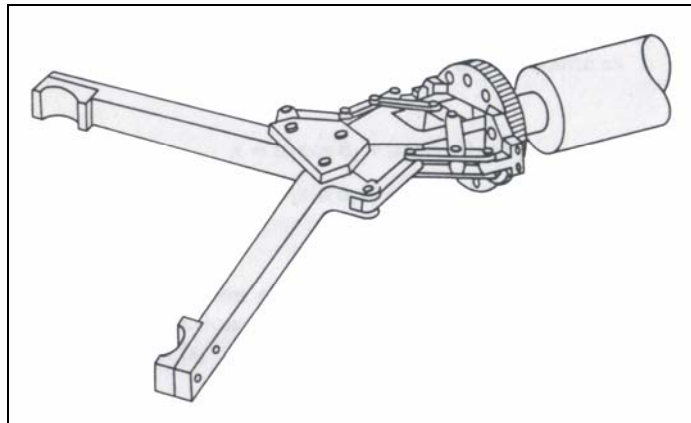


Figure 1. 2 : Pince à deux doigts.

D'autres recherches ont été menées dans le développement des poignets et mains anthropomorphes, afin de donner au robot la possibilité de s'adapter à l'objet à manipuler et à l'environnement de travail.

La liaison entre l'organe terminal et le bras est souvent réalisée par une jonction mécanique appelée poignet (Wrist en anglais). Cette jonction a pour tâche, l'orientation de l'organe terminal. Généralement on trouve des poignets sphériques qui permettent des rotations autour des trois axes du référentiel lié à l'organe terminal, une orientation quelconque dans l'espace est donc possible.

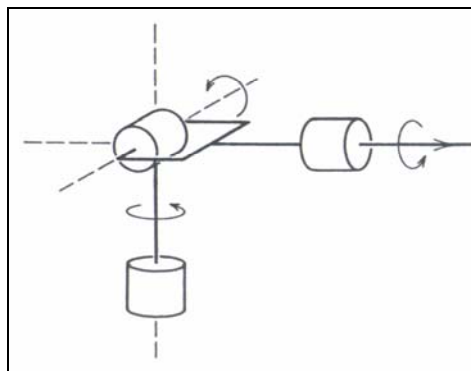


Figure 1. 3 : Structure d'un poignet sphérique.

1.5.2. Structure mécanique articulée.

Le rôle de la structure mécanique articulée est d'amener l'organe terminal dans une situation (position et orientation) donnée, selon des caractéristiques cinématiques données. Son architecture est une chaîne cinématique ce corps rigides, assemblés par des liaisons appelées articulations.

1.5.2.1. Couples cinématiques.

Un assemblage mobile de deux chaînons en contact permanent est appelé « *couple cinématique* ». Le contact en question peut se faire suivant un point, une ligne ou une surface.

On distingue deux sortes de couples cinématiques :

- **Couples cinématiques supérieurs** : Le contact se fait suivant une ligne ou un point.
- **Couples cinématiques inférieurs** : Le contact se fait suivant des surfaces.

Lorsqu'un chaînon constitue un couple cinématique avec un autre chaînon, cette condition impose certaines restrictions aux mouvements relatifs de ces chaînons, c'est à dire, qu'après assemblage, une partie des six degrés de liberté n'existe plus. Le nombre de ces restrictions, définit le nombre des « *conditions de liaisons* ».

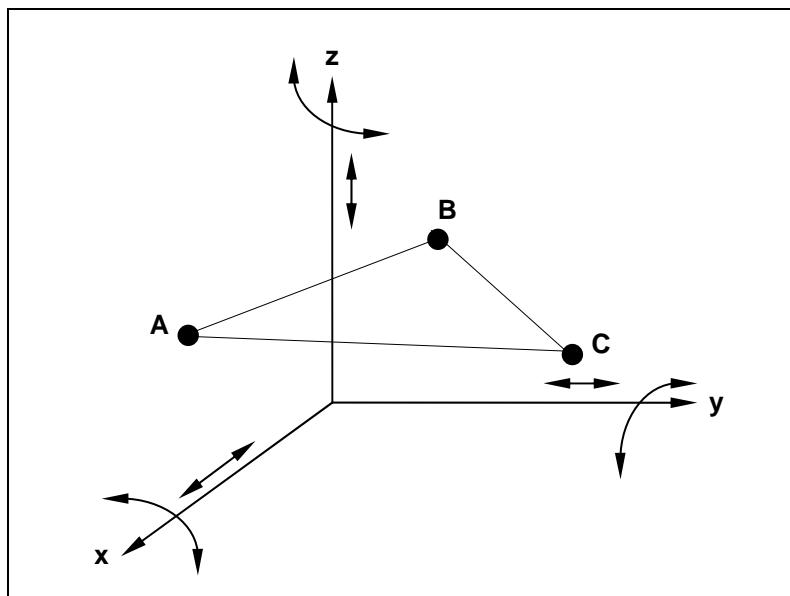


Figure 1. 4 : Déplacements possibles (degrés de liberté) d'un solide dans l'espace.

Il est notoire que dans le cas général, tout corps absolument rigide mobile dans l'espace (figure 1.4), dont la position est déterminée par trois points quelconques A , B et C choisis arbitrairement, possède six degrés de liberté. En effet la position d'un corps solide dans l'espace est déterminée par les coordonnées de ces trois points, c'est à dire par neuf coordonnées (X_A, Y_A, Z_A) , (X_B, Y_B, Z_B) , (X_C, Y_C, Z_C) ; ces coordonnées sont liées entre elles par trois conditions de constance de distance : AB , BC et CA . Donc les paramètres définissant la position d'un corps solide dans l'espace sont au nombre de six : le corps possède six « *degrés de liberté* » (H).

On peut toujours représenter le mouvement d'un tel corps comme la rotation autour de trois axes arbitraires réciproquement perpendiculaires X , Y et Z , et le glissement le long de ces trois axes. Donc, le cas général, un corps solide dans l'espace est libre d'effectuer six mouvements indépendants : Trois rotations autour des axes X , Y et Z et trois translations suivant les mêmes axes.

Le nombre de « conditions de liaison » (S) imposées au mouvement relatif de chaque élément du couple cinématique ne peut varier que de 1 à 5 en sorte que $1 \leq S \leq 5$.

Par conséquent, le nombre de degrés de liberté (H) d'un élément du couple cinématique animé d'un mouvement relatif peut être exprimé par la relation :

$$H = 6 - S \quad \text{--- (1. 1)}$$

De l'égalité (1.1), il suit que le nombre de degrés de liberté d'un élément du couple cinématique en mouvement relatif peut varier aussi de 1 à 5.

Les couples cinématiques sont classés en fonction du nombre de conditions de liaison qui s'imposent au mouvement relatif de leurs éléments. Comme le nombre de conditions de liaison varie de 1 à 5, on a cinq classes de couples ; il y a donc des couples cinématiques de classes *I*, *II*, *III*, *IV*, et *V*. On détermine aisément la classe d'un couple cinématique en partant de la relation (1.1) :


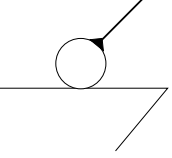
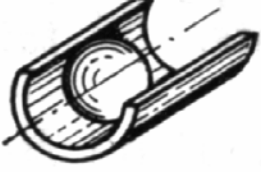
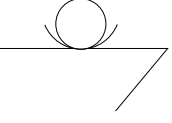

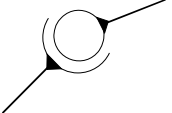
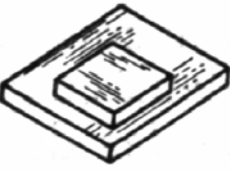
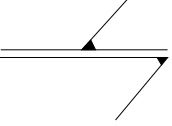
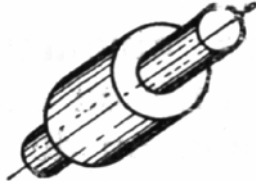
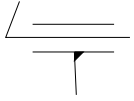
$$S = 6 - H \quad \text{--- (1. 2)}$$


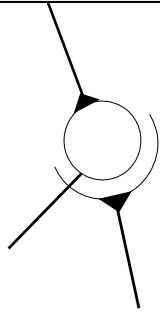

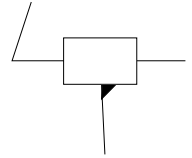

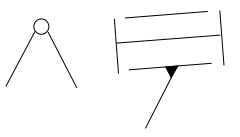

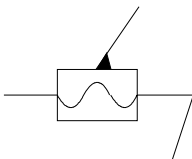
Remarque :

Dans certains ouvrages de mécanique, on peut trouver la notation inverse pour la classe d'un couple cinématique ; c'est à dire la classe sera égale à H au lieu de S .

Le tableau suivant nous donne les signes conventionnels des couples cinématiques les plus usuels en construction mécanique.

Tableau 1. 1 : Signes conventionnels des couples cinématiques.

Classe du couple	Nombre de conditions de liaisons	nombre de degrés de liberté	Appellation du couple	Dessin	Signe conventionnel
I	1	5	sphère-plan		
II	2	4	sphère-cylindre		
III	3	3	Couple sphérique		
III	3	3	Couple Plan		
IV	4	2	Couple cylindrique		

Classe du couple	Nombre de conditions de liaisons	nombre de degrés de liberté	Appellation du couple	Dessin	Signe conventionnel
IV	4	2	Couple sphérique guidé		
V	5	1	Couple de translation		
V	5	1	Couple de rotation		
V	5	1	Couple hélicoïdal		

Les couples cinématiques utilisés en robotiques sont les couples de translation appelés « *articulations prismatiques* » et les couples de rotations appelés « *articulations rotoïdes* » et même parfois, on peut trouver le couple hélicoïdal, qui peut représenter une articulation prismatique dont la translation est contrôlée par la rotation de l'un des deux chaînons.

1.5.2.2. Chaînes cinématiques.

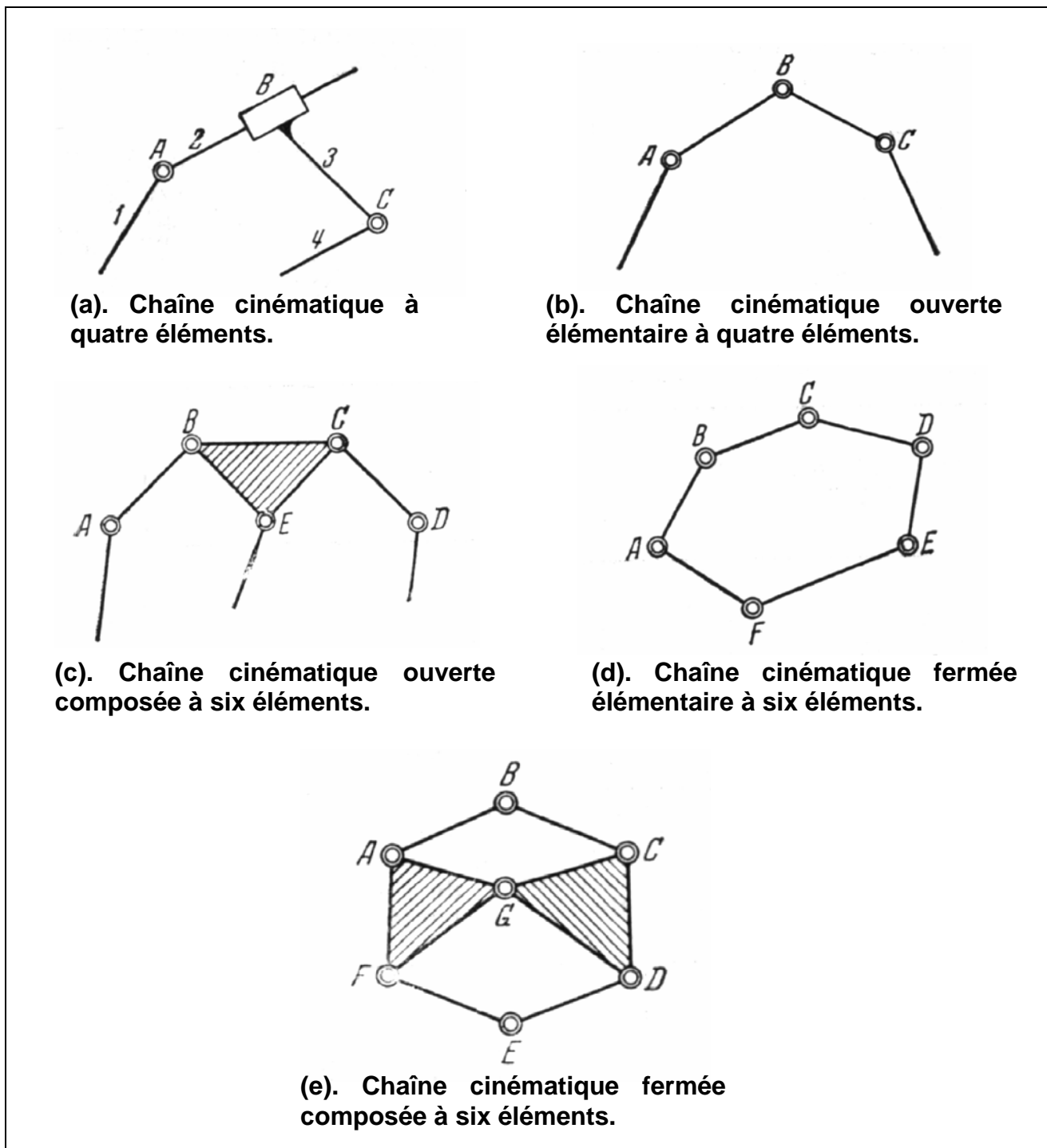


Figure 1. 5 : Exemples de chaînes cinématiques.

On entend par « *chaîne cinématique* », un système joint d'éléments qui forment entre eux des couples cinématiques. La figure (1.5.a) représente une chaîne cinématique composée de quatre éléments qui forment trois couples cinématiques. Les éléments 1 et 2 constituent un couple de rotation A (Classe V), les éléments 2 et 3, un couple de translation B (Classe V), enfin les éléments 3 et 4, un couple de rotation C (Classe V).

Les chaînes cinématiques peuvent être *élémentaires* ou *composées*. La chaîne est dite élémentaire si chacun de ces éléments ne forme que deux couples cinématiques au maximum. La figure (1.5.b) montre une chaîne élémentaires à couples de rotation A, B et C (Classe V).

La chaîne est dite composée si elle comporte au moins un élément qui forme plus de deux couples cinématiques. Une chaîne de ce type, comportant des couples de rotation A, B, C, D et E (Classe V) est représentée par la figure (1.5.c).

Les chaînes cinématiques élémentaires et composées se subdivisent à leur tour en chaînes « *fermées* » et « *non fermées* ». La chaîne est dite fermée si chacun de ces éléments forme au moins deux couples cinématiques. Les figures (1.5.d) et (1.5.e) montrent les chaînes de ce type à couple de rotation (Classe V).

On dit que la chaîne cinématique est non fermée si elle comporte des éléments qui ne forment qu'un seul couple cinématique, comme c'est le cas des chaînes représentées sur les figures (1.5.b) et (1.5.c).

1.6. Architecture des robots manipulateurs.

1.6.1. Structures générales des robots.

L'étude des structures arborescentes ainsi que celle des structures fermées se ramène à celle des chaînes ouvertes simples, qui est généralisable aux structures complexes.

En considérant le type d'articulation et l'angle que font deux axes articulaires successifs (généralement 0° ou 90°), le nombre de morphologies possibles en fonction du nombre d'articulations se déduit de la combinaison des quatre valeurs que peuvent prendre ces paramètres (tableau 1.2).

Tableau 1. 2 : Nombre de morphologies possibles en fonction de nombre de degrés de liberté [1].

Nombre de degrés de liberté	Nombre de structures
2	8
3	36
4	168
5	776
6	3508

Par convention, on appelle les trois premiers degrés de liberté d'une structure mécanique articulée d'un robot : « *le porteur du robot* », les degrés de libertés résiduels forment « *le poignet* », caractérisé par des dimensions beaucoup plus petites et une plus faible masse.

A partir du tableau (1.2), on constate qu'il existe 36 morphologie possibles pour le porteur, dont 12 seulement sont mathématiquement différentes et non redondantes¹ comme montré dans la figure (1.6).

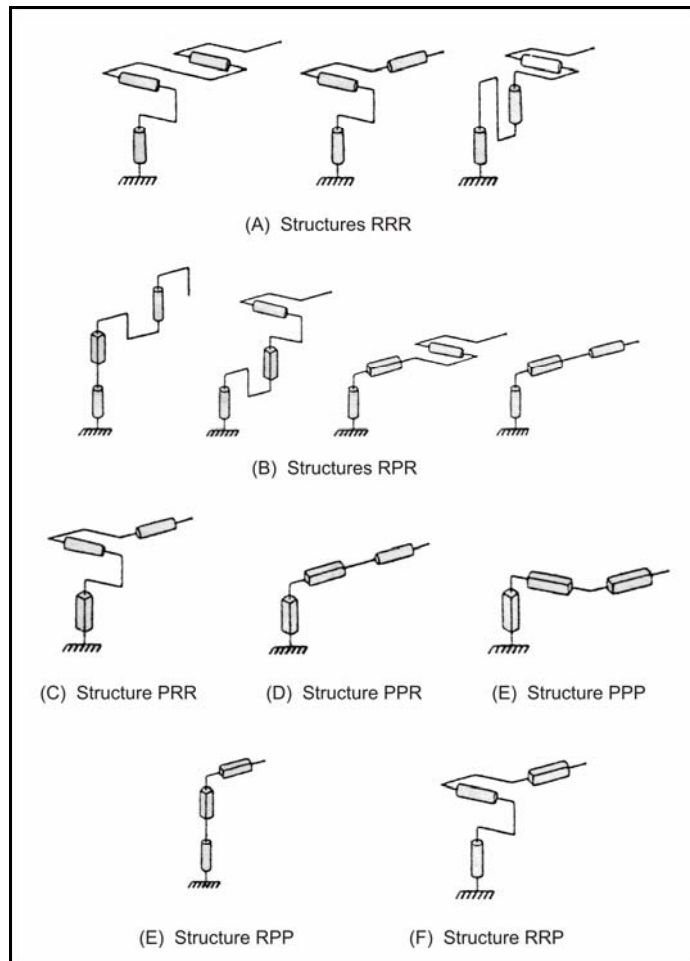


Figure 1. 6 : Architecture des porteurs [1].

¹ Voir définition de « *redondance* » § 1.7.2

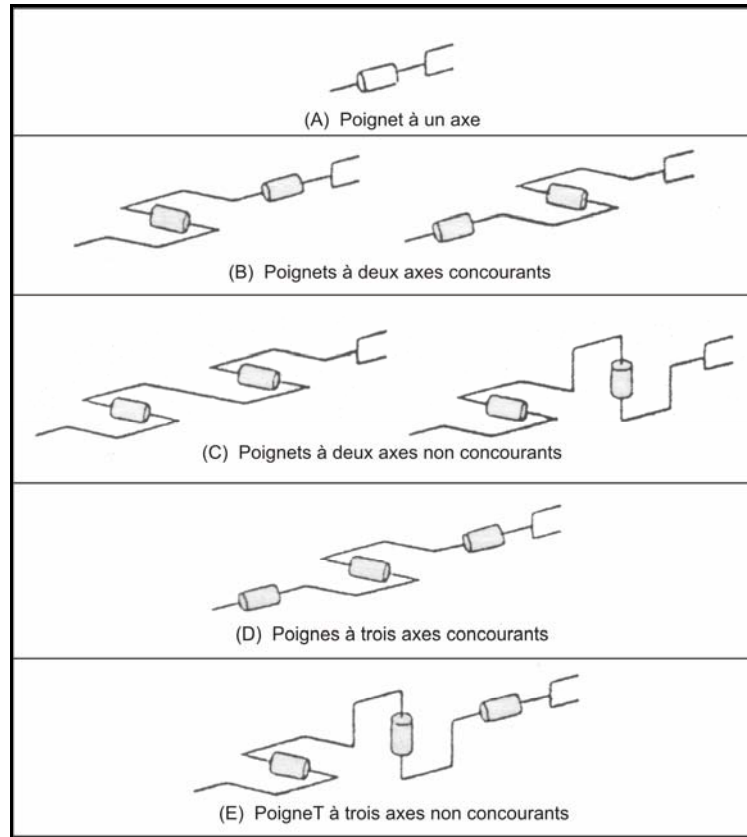


Figure 1. 7 : Architecture des poignets [1].

Dans la pratique, la structure la plus classique d'un robot est celle comportant un bras à trois degrés de liberté avec un poignet sphérique comme montré dans la figure (1.8)

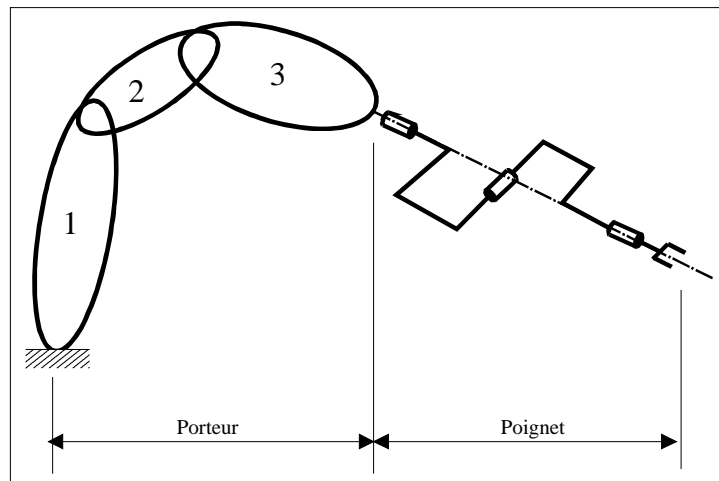


Figure 1. 8 : Structure générale d'un robot manipulateur [1].

1.6.2. Robots sériels.

Dans les années 70, pratiquement tous les robots étaient des bras manipulateurs dont les actionneurs sont placés en série (caractère anthropomorphique). Ce type de robot, parfois appelé sériel, est défini de la façon suivante : la structure articulée est une chaîne ouverte formée d'une succession de segments reliés entre eux par des liaisons à un degré de liberté ; chaque articulation est commandée par un actionneur situé à l'articulation ou sur l'un des segments précédents.

Ces robots ont pour avantage de disposer d'un grand volume de travail et être relativement simple sur le plan des calculs liés à leur commande. Par contre ces robots présentent des inconvénients qui peuvent se résumer en ce qui suit :

- Une inertie élevée due aux masses réparties sur toute la chaîne cinématique (actionneurs et organes de transmission) peut engendrer des erreurs de positionnement, des oscillations et surtout limite la rapidité. Ceci entraîne un rapport (charge utile / masse du robot) qui n'est jamais supérieur à 0.15 dans les meilleurs des cas ;
- Un manque de rigidité dû à la mise en série d'éléments souples. Le fait que les éléments soient disposés en série implique que chaque segment doit supporter la charge, mais aussi la masse des segments suivants. En rendant les segments plus rigides, ceci alourdira la structure dans la plus part des cas ;
- Une fatigue et une usure des liaisons de puissance assurant l'alimentation des actionneurs peut engendrer des pannes plus fréquentes.
- Une fatigue et une usure des liaisons assurant la circulation des informations entre les capteurs et la commande a des conséquences importantes sur la sécurité car une erreur de transmission peut engendrer des mouvements désastreux du robot.

1.6.3. Robots parallèles.

Un robot parallèle est constitué d'une plate-forme fixe et d'une autre plate-forme mobile (souvent appelée nacelle). La plate-forme mobile est reliée à la plate-forme fixe par plusieurs chaînes cinématiques en parallèle. L'ensemble forme une chaîne cinématique fermée de telle sorte que la définition simplifiée d'un robot parallèle peut être la suivante :

« Un manipulateur parallèle est constitué d'un organe terminal à n degré de liberté et d'une base fixe, reliée entre eux par au moins deux chaînes cinématiques indépendantes, la motorisation s'effectuent par n actionneurs simples ».

Le fait de mettre en parallèle plusieurs chaînes cinématiques entraînée chacune par un actionneur offre les avantages suivants :

- La capacité de charge est généralement plus importante que celle d'un robot série avec un poids de robot équivalent ;
- Cette structure parallèle donne la possibilité d'obtenir des mouvements à haute dynamique car les accélérations sont élevées ;
- La masse mobile est très faible, car les actionneurs sont souvent solidaires de la plate-forme fixe contrairement aux robots sériels qui comportent des actionneurs sur certaines articulations ;
- La fréquence propre de tels robots est élevée donc les erreurs dues aux oscillations incontrôlées de la structure mobile, observées sur les robots sériels, sont minimisées ;
- Une bonne rigidité mécanique confère aux structures parallèles une bonne précision ;
- La possibilité de positionner les actionneurs directement sur la base fixe procure de nombreux avantages. Tout d'abord, cela offre un choix important pour les actionneurs car leur masse n'est plus un critère de sélection du fait qu'elle n'intervient pas dans l'inertie. Les problèmes de liaison entre actionneurs, capteurs et contrôleurs sont simplifiés. Les moteurs sont plus facilement isolables de l'espace de travail dans le cas d'applications en atmosphère propre ;
- La symétrie des robots parallèles donne la possibilité de fabrication en série car le robot est composé de plusieurs parties identiques.

En contre partie de ces avantages, les robots parallèles présentent des inconvénients par rapport aux robots sériels :

- Limitation du volume de travail ;
- Le modèle géométrique² est parfois difficile à obtenir ;
- La présence de singularités³ peut conduire à une perte de contrôle de la structure mobile ;
- Le fait qu'il existe un couplage fortement variable entre les différentes chaînes cinématiques complique souvent le réglage et oblige à surdimensionner les actionneurs pour contourner cette difficulté.

² Voir le chapitre 2 « Modélisation des bras manipulateurs ».

³ Voir la définition de « singularité » § 1.7.3

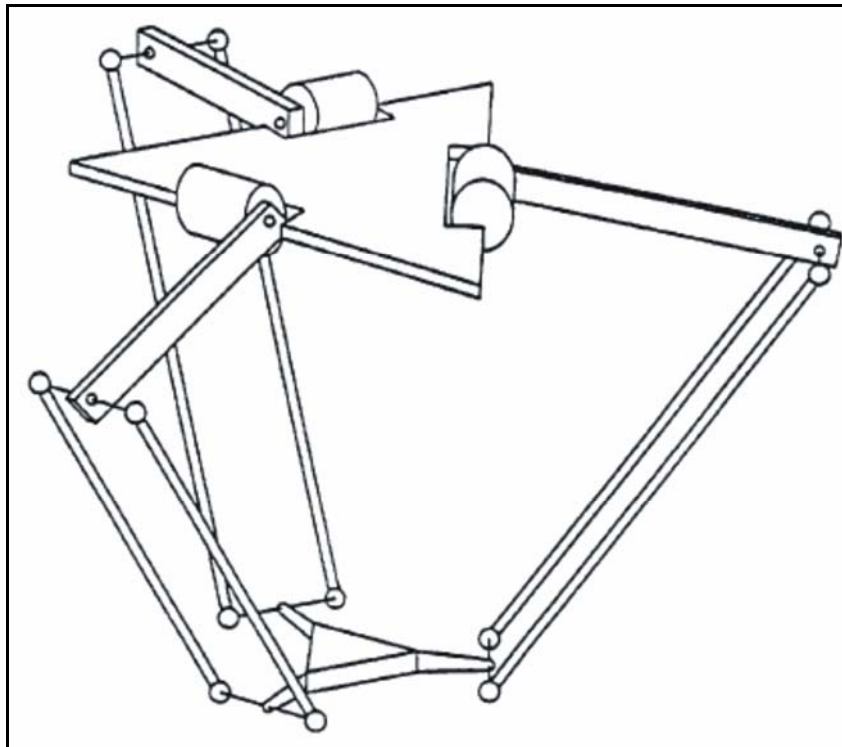


Figure 1. 9 : Exemple de structure parallèle « Robot DELTA » [6].

1.7. Définitions et terminologie.

1.7.1. Espace opérationnel & Espace articulaire.

L'espace opérationnel (appelé aussi espace de travail) d'un manipulateur est défini comme étant l'ensemble des positions et orientations accessibles par un repère lié à son organe terminal. La dimension de cet espace est définie par le nombre de paramètres indépendants nécessaires pour décrire la situation de ce repère. Si on note cet espace W , alors cet espace peut être défini comme suit :

$$W = \left\{ X \in \mathbb{R}^m / m \in \mathbb{N}^* ; \forall i = \overline{1, m}, (X_i)_{\min} \leq X_i \leq (X_i)_{\max} \right\} \quad \text{--- (1. 3)}$$

L'entier m est la dimension de W . Etant donné que le nombre de variables indépendantes nécessaires pour spécifier la situation d'un corps dans l'espace tridimensionnel est de six (trois rotations et trois translations), on peut déduire que :

$$0 < m \leq 6 \quad \text{--- (1. 4)}$$

On appelle espace articulaire (appelé parfois *configuration*), noté Q d'un robot manipulateur l'état du robot représentant la situation de ces différents corps. L'espace Q peut être défini comme suit :

$$Q = \left\{ q \in \mathbb{R}^n / n \in \mathbb{N}^* ; \forall i = \overline{1, n}, (q_i)_{\min} \leq q_i \leq (q_i)_{\max} \right\} \quad \text{--- (1.5)}$$

Où n est le nombre de variables articulaires.

1.7.2. Redondance.

Un robot est redondant lorsque le nombre de degrés de liberté de l'organe terminal est inférieur au nombre d'articulations motorisées. Les degrés de liberté supplémentaires autorisent le contournement des obstacles, cette propriété permet d'augmenter le volume du domaine opérationnel accessible.

1.7.3. Singularités.

Les manipulateurs de type sériel ou parallèle peuvent rencontrer des configurations dites singulières. Au voisinage de ces configurations, une augmentation des efforts dans les articulations du manipulateur ou une perte de rigidité peut avoir lieu alors que ses articulations motorisées sont bloquées. Son organe effecteur peut encore gagner ou perdre un ou plusieurs degré(s) de liberté. Une condition nécessaire à la robustesse d'un manipulateur est évidemment d'éviter les configurations singulières. Les singularités d'un manipulateur correspondent aux configurations annulant le déterminant de sa matrice jacobienne⁴ cinématique [1][7][10]. Les configurations singulières des manipulateurs non redondants font intervenir la notion d'*aspects*. Les aspects sont les plus grands domaines connexes ouverts de Q , ne possédant pas de configuration singulière. Ils réalisent une partition de Q .

La Figure (1.10) montre les deux aspects du manipulateur de type *RR* dans l'espace articulaire et leurs images dans l'espace opérationnel. Le premier aspect est associé à la posture « *coude haut* », le second correspond aux configurations de la posture « *coude bas* ». En outre, la jonction entre les deux aspects dans Q est à éviter de manière à s'éloigner des configurations singulières.

⁴ Voir définition de la « *matrice Jacobienne* » au chapitre 2.

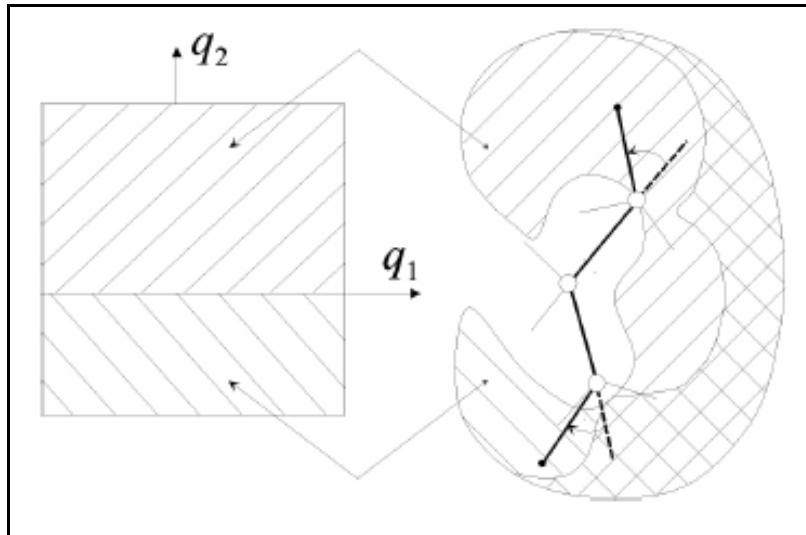


Figure 1. 10 : Aspects et postures du manipulateur du type RR.

La figure (1.11) représente le manipulateur parallèle de type $3PRR$ dans une configuration singulière parallèle. Dans ce cas, il est possible de bouger localement l'organe effecteur du manipulateur alors que ses actionneurs sont bloqués. Le manipulateur gagne donc un degré de liberté.

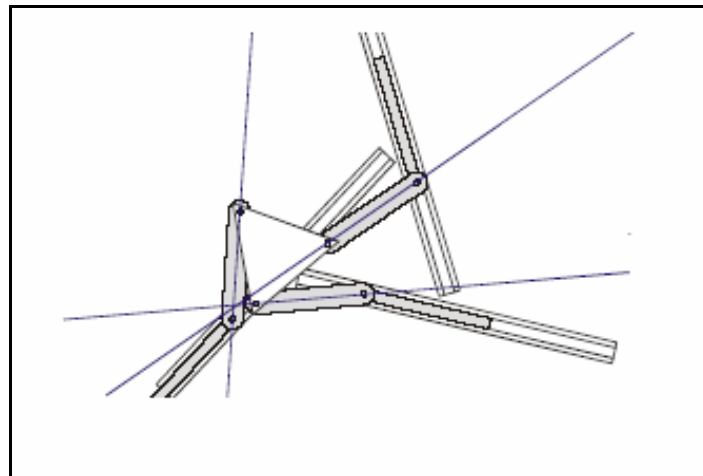


Figure 1. 11 : Singularité parallèle du manipulateur de type $3PRR$.

1.7.4. Parcourabilité.

L'espace de travail permet d'analyser les performances globales d'accessibilité d'un robot manipulateur. Or, une simple analyse d'accessibilité peut s'avérer insuffisante, même lorsque toutes les contraintes (butées, collisions) semblent avoir été prises en compte. Une condition nécessaire à la robustesse d'un manipulateur est bien évidemment que ses trajectoires soient parcourables. Si l'effecteur peut se déplacer entre deux points de l'espace de travail, cela signifie qu'il existe une trajectoire de liaison. Un changement de

posture en cours de mouvement est donc nécessaire, ce qui oblige l'organe terminal à passer par une configuration « *bras tendu* » et donc par un point de la frontière de l'espace de travail : la trajectoire doit être quittée et n'est donc pas parcourable. Il existe deux niveaux de parcourabilité, selon que la trajectoire entre les points soit spécifiée ou non. Pour chaque niveau de parcourabilité, il est utile de déterminer les régions de l'espace de travail qui sont réellement parcourables.

1.7.5. Dexterité.

Contrairement aux outils destinés à des analyses d'accessibilité et de parcourabilité, il existe des outils permettant d'évaluer des performances en un point particulier de l'espace de travail ou pour une configuration articulaire donnée. Ces analyses locales peuvent être complémentaires des précédentes. La dextérité, par exemple, peut être définie intuitivement comme mesurant l'aptitude de l'organe terminal du robot à effectuer « *facilement* » des petits déplacements arbitraires autour d'un point de l'espace de travail. La dextérité est étroitement liée au rapport entre les vitesses opérationnelles et les vitesses articulaires. Elle est généralement évaluée à l'aide de la notion de manipulabilité ou de l'indice d'isotropie [7].

1.7.6. Généricité.

Dans le cadre de la conception robuste de mécanismes, il est nécessaire de savoir si le mécanisme étudié est générique ou non puisque de faibles perturbations de ses paramètres peuvent changer ses propriétés globales lorsqu'il est non générique. La généricité d'un manipulateur implique que l'ensemble de ses configurations singulières forme une ou plusieurs surface(s) uniforme(s) et régulière(s) dans le domaine articulaire, c'est à dire qui ne se coupent pas. La propriété fondamentale d'un robot générique est sa stabilité vis-à-vis de petites perturbations de ses paramètres. Les robots non génériques sont reconnaissables par leurs singularités qui se coupent dans le domaine articulaire [7].

1.8. Coopération de robots.

D'une manière générale, on peut définir les systèmes multi-robots coopérants comme étant des systèmes regroupant des agents artificiels ayant une existence physique et étant engagés dans une tâche commune telle que :

- le fourragement qui signifie l'activité de recherche de nourriture ;
- la construction ;
- l'exploration des planètes ;

- l'industrie automatisée ;
- la recherche notamment dans les missions de secours ;
- le travail dans les environnements hostiles et les sites dangereux ;
- la maintenance et le nettoyage industrielle ou bien domestique ;
- la sécurité, la surveillance et la reconnaissance ;
- ou bien encore, dans un but plus ludique des robots footballeurs.

A première vue, on peut s'apercevoir que constituer une équipe de robot s'avère crucial dans plusieurs cas comme :

- Action distribuée : plusieurs robots peuvent être à plusieurs endroits différents au même moment ;
- Parallélisme : plusieurs robots peuvent réaliser des tâches différentes ou bien identiques à différents endroits au même moment ;
- La simplification de la conception : très souvent, dans une équipe de robots, chaque robot peut être réalisé de façon beaucoup plus simple qu'une solution constituée d'un unique robot.

Pour le cas des manipulateurs, confronté à des contraintes géométriques, cinématiques, énergétique, ou autres, un manipulateur se trouve parfois incapable d'effectuer la manutention d'un objet particulier, d'où naturellement vient à l'esprit l'idée d'utiliser plusieurs robots « *coopérants* » pour la même tâche.

En réalité, plusieurs variantes de robots coopérants peuvent apparaître sans qu'il s'agisse toujours de manipuler un objet lourd. On peut trouver deux manipulateurs qui coopèrent pour la même tâche, dont l'un manipule l'objet, et l'autre effectue des travaux sur cet objet comme la soudure, le pointage, etc. L'ensemble d'un manipulateur posé sur un chariot peut faire l'objet de deux robots coopérants dont l'un s'occupe du déplacement dans l'espace atelier, et l'autre effectue le travail demandé dans un espace plus proche.

Les robots de coopération sont aussi présent dans le domaine médical, on parle des « *robots médico-chirurgicaux* », dont l'objectif principal est de développer une coopération entre l'homme (le chirurgien) et une machine (le robot) qui cherche à exploiter les capacités des deux pour réaliser une tâche mieux que ce que pourrait faire l'un des deux tout seuls [13]. L'assistance des personnes handicapées fait aussi l'objet des robots coopérants avec l'homme dont l'objectif est d'imiter les mouvements de certains organes de l'être humain.

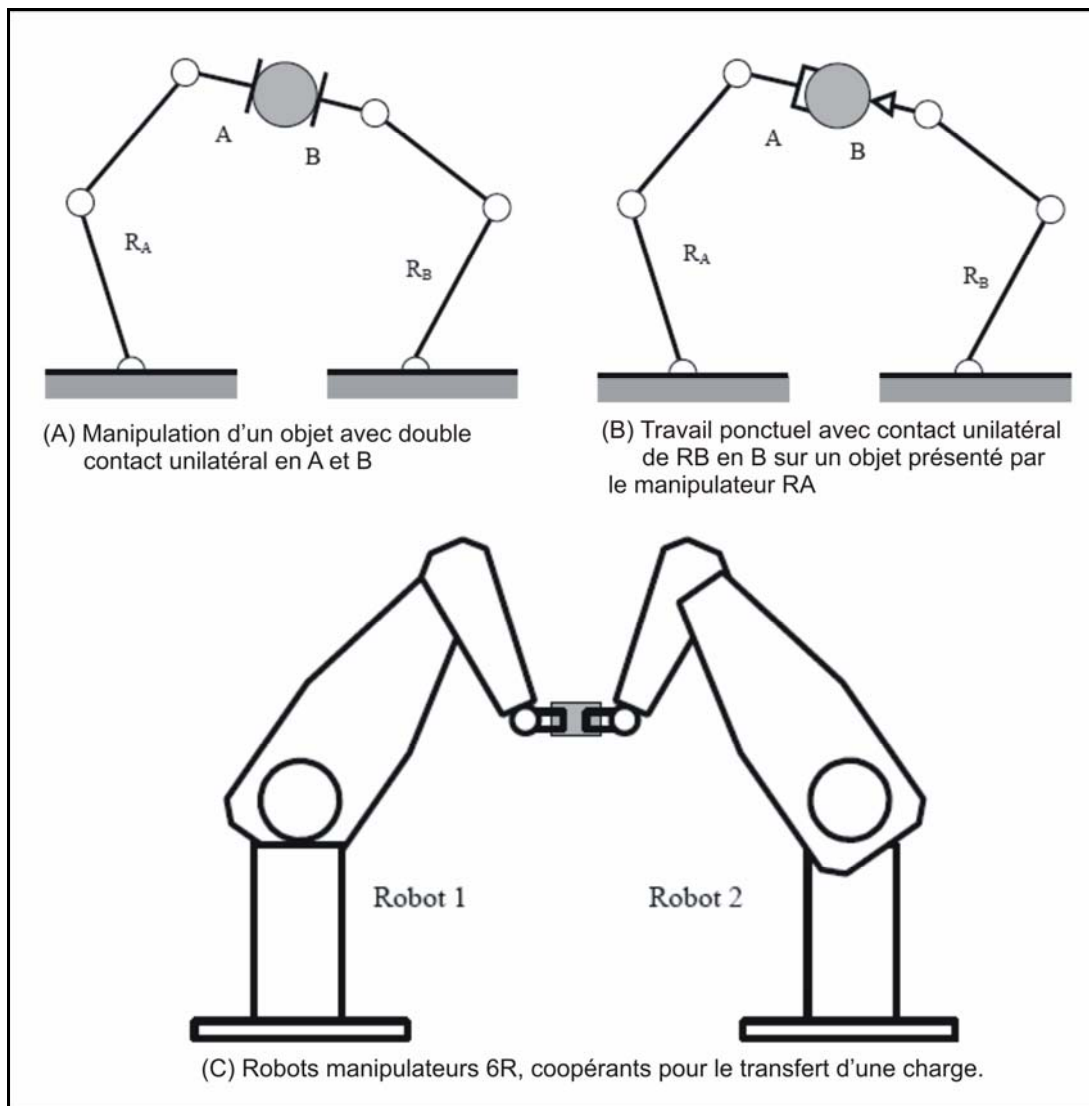


Figure 1. 12 : Exemples de robots coopérants.

La coopération entre plusieurs robots passe bien entendu par un partage de leurs connaissances sur l'environnement. Dans le cas des manipulateurs, il s'agit d'intégrer les données concernant l'environnement du travail, les obstacles ainsi que l'objet à manipuler.

Par rapport à l'opération d'un robot seul, la coopération entre plusieurs robots implique des algorithmes décisionnels supplémentaires (allocation de tâches, planification d'actions coopératives, coordination), et pose des problèmes particuliers aux processus de supervision des robots individuels et de supervision de l'ensemble des robots. Cela implique le développement d'une *architecture décisionnelle* qui permette l'intégration de ces algorithmes. Par ailleurs, les algorithmes décisionnels doivent pouvoir exploiter les modèles de l'environnement construits, et raisonner explicitement sur les capacités des robots à construire et mettre à jour ces modèles.

1.9. Conclusion.

Dans ce chapitre on a présenté d'une manière générale la terminologie utilisée en robotique ainsi que les différents constituants et structures mécaniques des robots industriels. Les deux architectures utilisées ont été brièvement définies. Dans ce qui suit, un intérêt particulier sera porté à l'architecture sérielle. Dans le chapitre suivant, on abordera d'une manière approfondie la modélisation mathématique des robots tenant compte du comportement cinématique et dynamique des bras manipulateurs, et proposer des méthodes algorithmiques directes et inverses pour l'analyse des robots.

CHAPITRE 2

MODÉLISATION DES BRAS MANIPULATEURS.

2.1. Introduction.

La réalisation d'un simulateur permettant de décrire l'état et le comportement global d'une structure mécanique articulée comme tout autre système, nécessite de combiner plusieurs concepts mathématiques.

Le problème principal dans la modélisation, est de trouver une relation entre les consignes données dans l'espace opérationnel de la tâche, et des postures des éléments du robot dans l'espace articulaire : cette relation permet de fournir une interface masquant le problème mathématique à l'utilisateur par le biais d'un programme informatique pour manipuler un robot de manière interactive.

Vu la complexité et la non linéarité des problèmes, un modèle complet du comportement réel d'un robot n'est pas réalisable. L'étude cinématique et dynamique des robots se fait généralement sur des modèles simplifiés, obtenus en introduisant des hypothèses simplificatrices, qui permettent de résoudre les équations qui représentent ces modèles.

Dans ce chapitre nous allons décrire la théorie sous-jacente aux problèmes de modélisation des robots dans le cas général avec un accent particulier qui sera mis sur les robots à chaînes ouvertes (sériels) tout en posant les hypothèses suivantes :

- Chaque corps composant le robot est assimilé à un solide indéformable ;
- Chaque articulation ne possède qu'un seul degré de liberté, les articulations de classes inférieures peuvent être décomposées en articulations élémentaires.

2.2. Formalismes de description des robots.

La description de la structure d'un robot en termes mathématiques est primordiale. Cette description est essentiellement basée sur la position relative ou absolue des différents corps du robot dans l'espace. Il est alors nécessaire de choisir des méthodes pour :

- décrire les transformations de coordonnées et leur composition afin de placer les différents corps dans l'espace ;
- obtenir une notation cohérente et universelle pour paramétrer une structure articulée.

2.2.1. Transformation de coordonnées.

Il existe une grande variété d'outils pour la représentation des mouvements des corps solides dans l'espace. La géométrie analytique donne des outils très performants pour la représentation et la résolution des équations représentant les transformations ponctuelles. Ainsi, dans le cas planaire, une projection dans le plan complexe peut être très utile vu la simplicité des calculs dans le corps des nombres complexes.

Dans ce projet, nous avons adopté la méthode des matrices de transformation homogènes 4x4, qui permettent la représentation de n'importe quelle transformation : translation, rotation ou même le mouvement hélicoïdal. Cette méthode est beaucoup rencontrée dans la littérature [1], [2], [18], vu son aptitude à décrire des robots de degré des liberté quelconque ainsi que la simplicité de la programmation informatique des calculs matriciels. Les concepts théoriques des transformations homogènes seront présentés en appendice A.

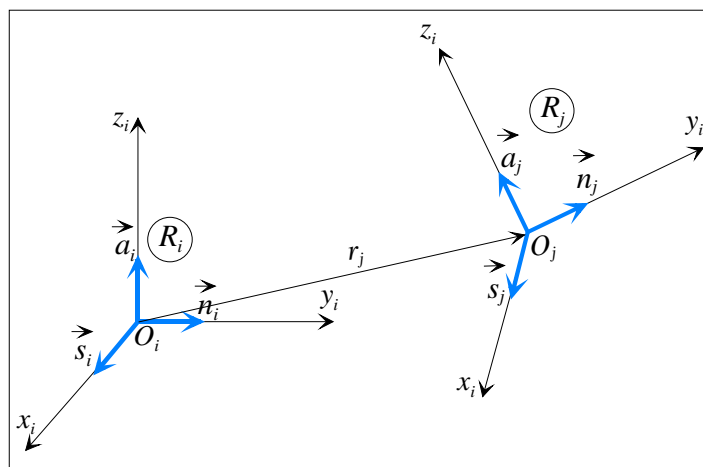


Figure 2. 1 : Définition des repères pour une transformation homogène.

La matrice homogène ${}^i T_j$ qui représente le repère R_j par rapport au repère R_i suivant les notations de la figure (2.1) est définie comme suit⁵ :

$${}^i T_j = \begin{bmatrix} s_x & n_x & a_x & r_x \\ s_y & n_y & a_y & r_y \\ s_z & n_z & a_z & r_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^i s_j & {}^i n_j & {}^i a_j & {}^i r_j \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^i A_j & {}^i r_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (2.1)}$$

Cette matrice ${}^i T_j$ représente l'opérateur qui permet de trouver les coordonnées d'un point ${}^j P$ ou sont vecteur position ${}^j V$ (exprimé dans le repère R_j) dans le repère R_i selon l'équation suivante :

$${}^i V = {}^i T_j \cdot {}^j V \quad \text{---- (2.2)}$$

2.2.2. Description des paramètres d'une chaîne articulée.

Dans La modélisation géométrique des robots, et avec la méthode des matrices homogènes qu'on a adopté pour notre projet, il faut assigner des repères aux différents composants du robot, afin de décrire toute chaîne cinématique de manière uniforme. Dans la littérature, une multitude de notations est présente. Nous présentons ici sommairement deux notations classiques, puis dans un troisième paragraphe nous détaillerons la notation adoptée pour ce projet.

2.2.2.1. Notation de Sheth-Uiker.

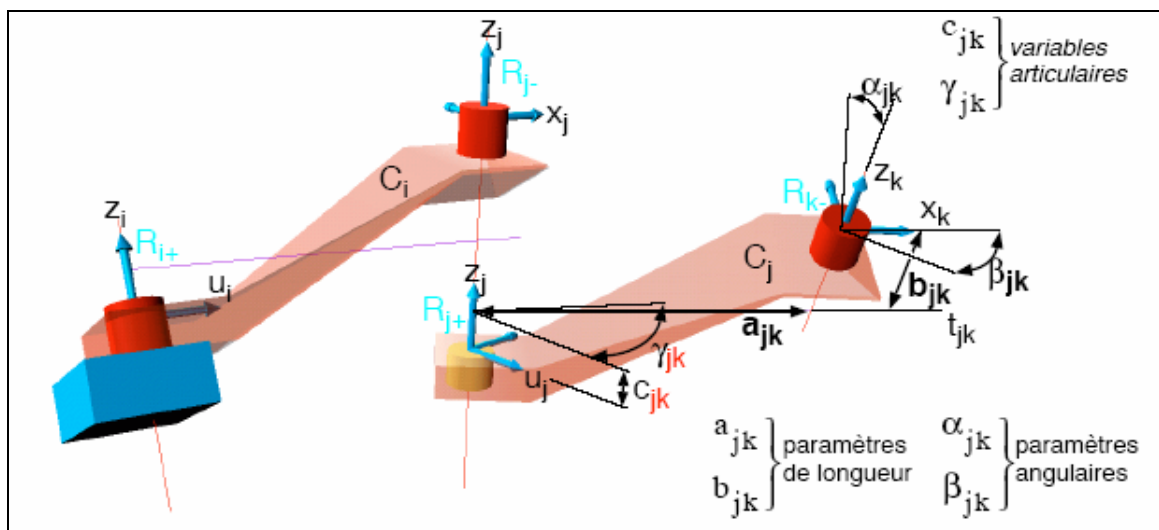


Figure 2. 2 : Définition des paramètres géométriques suivant Sheth-Uiker [8].

⁵ Dans l'appendice A, on a décrit les différentes représentations ainsi que les méthodes d'obtention des matrices de transformations homogènes dans les différents cas.

Pour pouvoir décrire des chaînes cinématiques complexes (arborescentes ou bouclée), Sheth et Uiker ont proposé une notation basée sur l'assignation de deux repères par lien. La figure (2.2) présente cette notation pour deux liens dans une vue éclatée, afin de visualiser correctement les différents repères. Chaque lien i se voit attribué un repère coïncidant avec son articulation i et un deuxième coïncidant avec l'articulation $i+1$.

Ainsi, chaque lien est modélisé par deux matrices :

- Une matrice décrivant la forme géométrique du lien et contenant donc des paramètres constants ;
- Une deuxième matrice décrivant le déplacement relatif de deux liens successifs, contenant donc des paramètres variables.

Remarque.

Dans le cas des chaînes cinématiques complexes contenant des chaînons ayant plus de deux couples cinématiques, on assigne à chaque lien, un nombre de repère égal au nombre de liaisons du lien considéré. L'utilisation de deux repères n'est donc valable que dans le cas des chaînes cinématiques simples.

L'avantage de cette notation se résume dans le fait que c'est une méthode complètement générale (non limitée aux robots classiques). Par contre, l'utilisation de deux repères (ou plus) par lien, rend cette méthode complexe à mettre en œuvre.

2.2.2.2. Notation de Kleinfinger-Khalil.

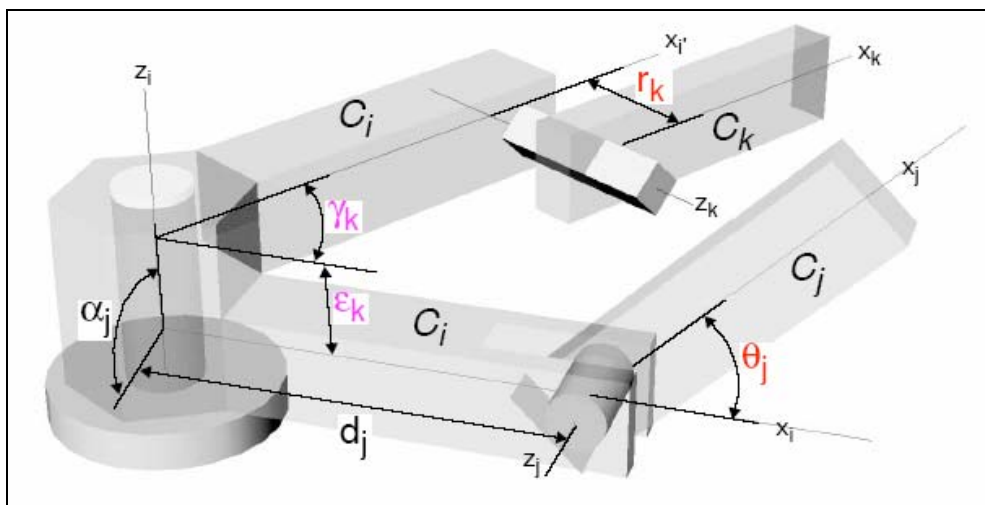


Figure 2. 3 : Définition des paramètres géométriques suivant Kleinfinger-Khalil [8].

Kleininger et Khalil ont proposé une méthode pour la description des chaînes cinématiques sérielles, arborescentes ou bouclées. La figure (2.3) présente l'assignation des repères pour cette notation et définit les paramètres nécessaires. Cette figure montre un corps C_j avec une articulation rotoïde, et un corps C_k avec une articulation prismatique. Ces deux corps sont reliés par leurs articulations à un corps rigide C_i qui forme un embranchement.

Cette notation utilise un repère par lien, et six paramètres pour la modélisation de chaque corps. Ce nombre élevé par rapport à d'autres méthodes rend la matrice de transformation beaucoup plus complexe à définir, mais il est avantageux pour donner à la méthode l'aptitude de représenter tout type de structure.

2.2.2.3. Notation de Denavit-Hartenberg.

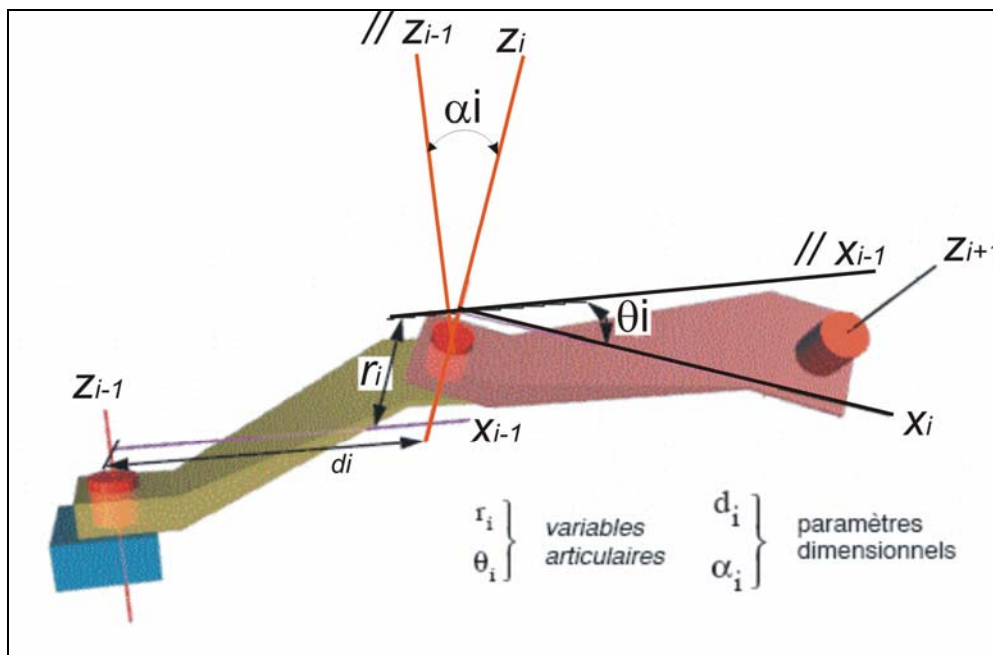


Figure 2. 4 : Définition des paramètres géométriques suivant Denavit-Hartenberg [8].

Denavit et Hartenberg ont proposé une méthode reposant sur l'assignation d'un repère unique pour chaque lien. La figure (2.4) présente l'utilisation de cette notation pour deux liens successifs.

Pour décrire les robots à chaîne ouverte simple qui feront l'objet de notre étude, on a adopté cette méthode qui consiste en ce qui suit [1] : Le système est composé de $n+1$ corps, notés C_0, \dots, C_n , et de n articulations. Le corps C_0 désigne la base du robot et C_n le

corps qui porte l'organe terminal. L'articulation i connecte le corps C_i avec le corps C_{i-1} . Le repère R_i , fixé au corps C_i défini de telle sorte que :

- L'axe z_i est porté par l'axe de l'articulation i ;
- L'axe x_i est porté par la perpendiculaire commune aux axes z_i et z_{i+1} ;

Le passage de R_{i-1} à R_i s'exprime en fonction des quatre paramètres suivants :

- α_i : angle entre les axes z_{i-1} et z_i , correspondant à une rotation autour de x_{i-1} ;
- d_i : distance entre z_{i-1} et z_i le long de x_{i-1} ;
- θ_i : angle entre les axes x_{i-1} et x_i , correspondant à une rotation autour de z_i ;
- r_i : distance entre x_{i-1} et x_i le long de z_i .

La variable articulaire q_i associée à la $i^{\text{ème}}$ articulation est définie comme suit [1] :

$$q_i = \bar{\sigma}\theta_i + \sigma_i r_i \quad \text{---- (2. 3)}$$

Avec :

$\sigma_i = 0$: si l'articulation est rotoïde

$\sigma_i = 1$: si l'articulation est prismatique

$$\bar{\sigma}_i = 1 - \sigma_i$$

La matrice de transformation définissant le repère R_i dans le repère R_{i-1} est donnée par [1] :

$${}^{i-1}T_i = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & d_i \\ C\alpha_i S\theta_i & C\alpha_i C\theta_i & -S\alpha_i & -r_i S\alpha_i \\ S\alpha_i S\theta_i & S\alpha_i C\theta_i & C\alpha_i & r_i C\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (2. 4)}$$

Pour l'obtention de la matrice de transformation définissant l'organe terminal dans le repère lié à la base du robot, on applique le produit matriciel suivant [1] :

$${}^0T_n = {}^0T_1 \cdot {}^1T_2 \cdot {}^2T_3 \cdot \dots \cdot {}^{n-1}T_n \quad \text{---- (2. 5)}$$

La notation de Denavit-Hartenberg souvent notée DH, est apparue très tôt dans le domaine de la robotique, mais elle est largement utilisée par la communauté scientifique pour décrire les robots en vue de leur analyse ou leur modélisation [1][5][16][17][20]. Elle présente certains avantages comme le nombre réduit de paramètres (quatre paramètres par lien), mais elle n'est pas utilisable pour décrire les chaînes arborescentes ou bouclées. Vu notre objectif qui est l'étude et la modélisation des robots coopérants, dont la majorité présente des chaînes ouvertes simples (robots sériels) surtout lorsqu'il s'agit des bras

manipulateurs, notre choix est tombé sur cette méthode afin de simplifier la construction des matrices de transformations en vue de leur programmation informatique.

2.2.3. Matrice jacobienne.

L'outil principalement utilisé pour traiter le problème de la cinématique, et particulièrement des modèles géométriques et cinématique des robots est « *la matrice jacobienne* ». Elle représente un opérateur permettant de lier les vitesses des corps d'un robot exprimées dans différents espaces vectoriels⁶.

En considérant la fonction f définie comme suit :

$$\begin{aligned} f: Q &\rightarrow W \\ q &\mapsto X = f(q) \end{aligned} \quad \text{---- (2.6)}$$

Où Q et W sont deux espaces vectoriels de dimensions (respectivement) m et n .

La différentiation de l'équation $X = f(q)$ dans (2.6), donne :

$$dX = \frac{\partial f}{\partial q} dq \quad \text{---- (2.7)}$$

On définit alors, la matrice jacobienne comme suit :

$$[J(q)] = \left[\frac{\partial f}{\partial q} \right] \quad \text{---- (2.8)}$$

Nous verrons par la suite, comment obtenir cette matrice pour le cas des robots, ainsi que son utilisation pour résoudre certains problèmes de cinématique.

2.3. Modélisation géométrique.

En robotique, pour un utilisateur comme pour un concepteur, la première interrogation qu'on se pose face à un robot est : « *comment calculer sa position ?* », ou plus précisément :

- « *Où va se trouver l'outil du robot si chaque articulation possède tel angle (moteur rotatif), ou tel allongement (actionneur linéaire) ?* » : modèle géométrique direct ;
- « *Comment trouver les valeurs de chaque articulation pour que l'outil du robot atteigne telle position avec telle orientation ?* » : modèle géométrique inverse.

⁶ Voir § 2.4 : « *Modélisation cinématique* ».

2.3.1. Modèle géométrique direct M.G.D.

Le modèle géométrique direct est l'ensemble des relations qui permettent d'exprimer la situation de l'organe terminal du robot en fonction de ces coordonnées articulaires [1].

$$\begin{aligned} f: Q &\rightarrow W \\ q &\mapsto X = f(q) \end{aligned} \quad \text{--- (2. 9)}$$

Le vecteur X peut être défini de plusieurs manières différentes, selon le mécanisme en question. A titre d'exemple, pour un mécanisme plan, W étant de dimension 3, le vecteur X peut être facilement calculé par des relations géométriques. En général, pour les cas tridimensionnels. Le vecteur X comporte trois composantes pour définir la position ponctuelle, ainsi que trois autres au minimum pour définir l'orientation selon le système d'angles d'Euler, ou bien, six composantes pour définir son orientation avec les cosinus directeurs.

Dans le cas des structures ouvertes simples, le calcul de X implique le calcul des matrices de transformation de l'organe terminal, la où les composantes du vecteur q font partie des composantes de la matrices de transformation qui exprime l'organe terminal dans le repère lié à la base ou bien dans le repère atelier.

2.3.2. Modèle géométrique inverse M.G.I.

Le problème du modèle géométrique inverse consiste à calculer les coordonnées articulaires qui amènent l'organe terminal à une situation désirée, spécifiée par ses coordonnées opérationnelles.

Si on analyse le problème du point de vue algébriste, le modèle géométrique direct étant une fonction définie sur un espace Q et qui trouve ses valeurs dans l'espace W . Le problème inverse consiste à trouver la fonction (application) réciproque qui n'existe que si cette fonction est une application bijective.

Même si la surjection est assurée en calculant auparavant l'espace opérationnel admissible par des méthodes géométriques ou même analytiques, toutes les valeurs de W seront donc réalisables, par contre l'injectivité est souvent un problème inévitable, à cause des fonction trigonométriques présente dans le modèle géométrique direct. Ces fonctions ont souvent le caractère non injectif, ce qui force dans la plus part des cas à avoir une fonction non bijective et par conséquent, l'inexistence de sa fonction réciproque.

La non injectivité entraîne donc la multitude de solutions pour l'équation $f(q) = X$, ce problème est connue en robotique sous le terme de « *redondance* ». Quoique c'est un problème de point de vue analytique, il est souvent bénéfique dans plusieurs cas, lorsqu'il s'agit d'éviter ou de contourner les obstacles.

Nous présentons par la suite, quelques méthodes pour la résolution du modèle géométrique inverse⁷.

2.4. Modélisation cinématique.

Le modèle géométrique du robot (qu'il soit direct ou inverse) est utile car il apporte une vision « *globale* » du comportement d'un robot. Toutefois, il reste limité pour le contrôle réel d'un robot. En effet, il correspond à un contrôle « *point à point* », qui bien qu'encore largement utilisé dans le monde industriel, répond mal au problème d'un déplacement entre deux points relativement distants. Il est nécessaire alors d'introduire les paramètres de vitesses et accélérations pour obtenir un contrôle du comportement plus efficace. Le contrôle en position n'est donc qu'un modèle restreint par rapport à la réalité robotique.

Le modèle cinématique direct d'un robot manipulateur décrit les vitesses des coordonnées opérationnelles en fonction des vitesses articulaires il est noté :

$$\dot{X} = J(q).\dot{q} \quad \text{--- (2. 10)}$$

Où $J(q)$ désigne la matrice jacobienne du mécanisme, cette matrice intervient aussi dans le calcul du modèle différentiel direct qui donne les variations élémentaires dX des coordonnées opérationnelles en fonction des variations élémentaires des coordonnées dq soit :

$$dX = J(q).dq \quad \text{--- (2. 11)}$$

Le modèle cinématique permet donc non seulement de compléter éventuellement le modèle géométrique en tenant compte des vitesses, mais aussi de le remplacer : en agissant par accroissements successifs.

Le modèle cinématique possède une propriété essentielle : il est une différentiation du modèle géométrique. Il est donc une linéarisation du système d'équations non linéaires

⁷ Voir § 2.5.1 « *Résolution du modèle géométrique inverse* ».

représentant le modèle géométrique. Par conséquent on peut toujours facilement obtenir les transformations inverses, puisqu'elles proviennent de l'inversion d'un problème linéaire. Toutefois le modèle cinématique comporte aussi des inconvénients :

- la non unicité du modèle géométrique inverse implique qu'il existe plusieurs "chemins" pour se rendre d'un point à un autre ;
- le traitement par incrément peut amener à des imprécisions provoquant des singularités, mécaniques et/ou mathématiques.

Une dérivation de l'équation (2.11) par rapport au temps, donne le modèle cinématique du deuxième ordre, qui permet de calculer les accélérations des coordonnées opérationnelles en fonction des positions, vitesses et accélérations articulaires.

$$\ddot{X} = J \cdot \ddot{q} + \dot{J} \cdot \dot{q} \quad \text{----- (2. 12)}$$

2.5. Cinématique « Directe & Inverse » - Méthodes de résolution.

2.5.1. Résolution du modèle géométrique inverse.

Plusieurs méthodes ont été proposées pour résoudre le M.G.I. (Modèle géométrique inverse), dont certaines sont analytiques, on citera par exemple [1] :

- Méthode de Paul : qui traite séparément chaque cas particulier et convient pour la plupart des robots industriels ;
- Méthode de Pieper qui permet de résoudre le problème pour les robots à six degrés de liberté possédant trois articulations rotoïdes d'axes concourants ou trois articulations prismatiques ;
- Méthode de Raghavan et Roth, donnant la solution générale des robots à six articulations à partir d'un polynôme de degré au plus égal à 16.

D'autres méthodes dont le principe, consiste traiter le problème de géométrie inverse, comme un problème d'optimisation : citons à titre d'exemple la méthode CCD (*Cyclic coordinate Descent*) [21].

2.5.2. Formation de la matrice jacobienne.

Il existe plusieurs méthodes pour calculer la matrice jacobienne. Si on écarte les méthodes de calcul symbolique permettant de dériver les équations du modèle géométrique direct, il faut calculer les influences de chaque articulation sur l'organe terminal comme le

montre la Figure (2.5). Cette approche, très répandue pour le calcul cinématique, peut s'effectuer suivant différents moyens qui sont plus ou moins efficaces au niveau des algorithmes et du nombre d'opérations nécessaires. Ces méthodes se caractérisent par le repère dans lequel la matrice jacobienne est exprimée ainsi que par le corps auquel elle correspond, c'est-à-dire par le choix de i et j dans le calcul de ${}^i J_j$: matrice jacobienne du corps j exprimée dans repère i .

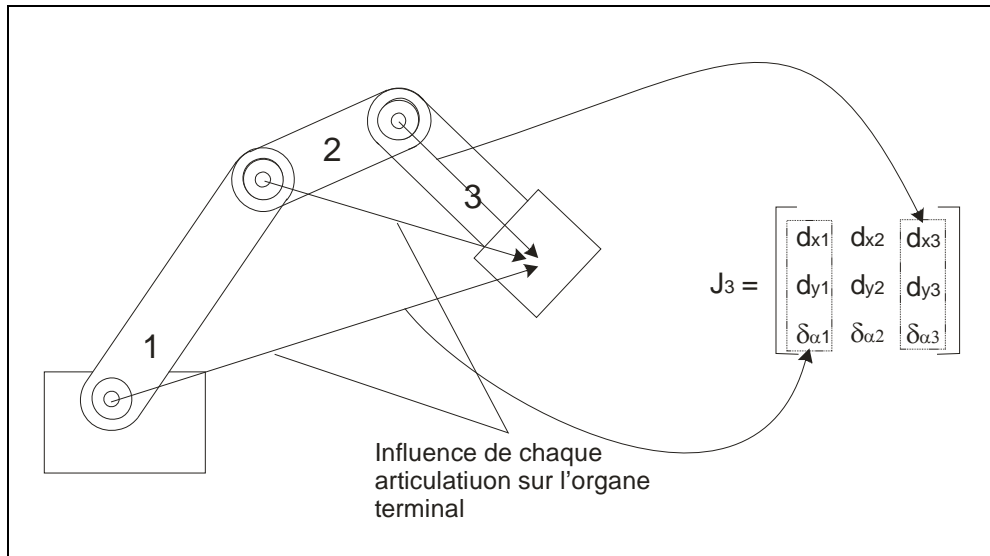


Figure 2. 5 : Matrice jacobienne d'un manipulateur planaire à trois degrés de liberté.

La matrice jacobienne ${}^l J_m$ est composée de $(m-l)$ colonne, la $k^{\text{ème}}$ colonne est donnée calculée comme suit [1] :

$${}^l j_{m:k} = \begin{bmatrix} \sigma_{k+l} \cdot {}^l a_{k+l} + \overline{\sigma_{k+l}} \left(-{}^{k+l} P_{my} \cdot {}^l s_{k+l} + {}^{k+l} P_{mx} \cdot {}^l n_{k+l} \right) \\ \overline{\sigma_{k+l}} \cdot {}^l a_{k+l} \end{bmatrix} \quad \text{---- (2. 13)}$$

Avec $k = 1, 2, \dots, m-l$

Où (s, n, a, P) sont les paramètres de description des matrices homogènes.

2.5.3. Calcul pratique des vitesses & accélérations opérationnelles [1].

Du point de vue du nombre d'opérations, pour trouver les vitesses et accélération opérationnelles (de l'organe terminal) il est plus judicieux d'utiliser les relations de récurrence suivantes :

Pour le calcul des vitesses :

$$\begin{cases} {}^j\omega_{j-1} = {}^jA_{j-1} \cdot {}^{j-1}\omega_{j-1} \\ {}^j\omega_j = {}^j\omega_{j-1} + \overline{\sigma_j} \cdot \dot{q}_j \cdot {}^j a_j \\ {}^jV_j = {}^jA_{j-1} \left[{}^{j-1}V_{j-1} + ({}^{j-1}\omega_{j-1}) \times ({}^{j-1}P_j) \right] + \sigma_j \cdot \dot{q}_j \cdot {}^j a_j \end{cases} \quad \text{---- (2. 14)}$$

Pour le calcul des accélérations :

$$\begin{cases} {}^j\dot{\omega}_j = {}^jA_{j-1} \cdot {}^{j-1}\dot{\omega}_j + \overline{\sigma_j} \left[\ddot{q}_j \cdot {}^j a_j + ({}^j\omega_{j-1}) \times (\dot{q}_j \cdot {}^j a_j) \right] \\ {}^jU_j = {}^j\dot{\omega}_j + {}^j\widehat{\omega}_j \cdot {}^j\widehat{\omega}_j \\ {}^j\dot{V}_j = {}^jA_{j-1} \left[{}^{j-1}\dot{V}_{j-1} + {}^{j-1}U_{j-1} \cdot {}^{j-1}P_j \right] + \sigma_j \left[\ddot{q}_j \cdot {}^j a_j + 2({}^j\omega_{j-1}) \times (\dot{q}_j \cdot {}^j a_j) \right] \end{cases} \quad \text{---- (2. 15)}$$

2.6. Modélisation dynamique.

Le modèle dynamique est la relation entre les charges appliqués aux actionneurs et les positions, vitesses, et accélérations articulaires. On représente le modèle dynamique par une relation de la forme :

$$\Gamma = f(q, \dot{q}, \ddot{q}, F_e) \quad \text{---- (2. 16)}$$

Avec

Γ : Vecteur des charges appliquées aux actionneurs ;

q : Vecteur des positions articulaires ;

\dot{q} : Vecteur des vitesses articulaires ;

\ddot{q} : Vecteur des accélérations articulaires ;

F_e : Vecteur représentant l'effort extérieur ;

Par convention, la relation (2.16) est appelée « *modèle dynamique inverse* », ou tout simplement « *modèle dynamique* ».

Le modèle dynamique direct, est celui qui exprime les accélérations articulaires en fonction des positions, vitesses et couples articulaires, il est représenté par la relation :

$$\ddot{q} = g(q, \dot{q}, \Gamma, F_e) \quad \text{---- (2. 17)}$$

Plusieurs formalismes ont été utilisés pour obtenir le modèle dynamique des robots, les plus utilisés sont : le *formalisme de Lagrange* et le *formalisme de Newton-Euler*.

2.6.1. Formalisme de Lagrange.

Le formalisme de Lagrange décrit les équations du mouvement en termes de travail et d'énergie du système, ce qui se traduit par l'équation suivante :

$$\Gamma_i = \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} \quad \text{---- (2. 18)}$$

Où :

- L : Lagrangien du système égal à $E - U$;
- E : Énergie cinétique totale du système ;
- U : Énergie potentielle totale du système ;

L'énergie cinétique du système est une fonction quadratique des vitesses articulaires :

$$E = \frac{1}{2} \dot{q}^T \cdot A \cdot \dot{q} \quad \text{---- (2. 19)}$$

Où A est appelée matrice d'inertie du robot, c'est une matrice de dimensions $(n \times n)$, symétrique définie positive, ses éléments sont en fonction des variables articulaires q_j .

L'énergie potentielle étant en fonction des variables articulaires q_j , le vecteur Γ peut se mettre sous la forme :

$$\Gamma = A(q) \cdot \ddot{q} + C(q, \dot{q}) \cdot \dot{q} + Q(q) \quad \text{---- (2. 20)}$$

$C(q, \dot{q}) \cdot \dot{q}$: Vecteur de dimensions $(n \times 1)$ représentant les couples/forces de Coriolis et les forces centrifuges ;

$Q(q)$: Vecteur des couples/forces de gravité.

2.6.1.1. Calcul des éléments de A , C et Q [1].

Après avoir calculer les énergies cinétique et potentielle, on peut calculer les éléments de A , C et Q comme suit :

- L'élément A_{ii} est égal au coefficient de $\frac{1}{2}(\dot{q}_i)^2$ dans l'expression de l'énergie cinétique ;
- L'élément A_{ij} si $i \neq j$ est égal au coefficient de $(\dot{q}_i \cdot \dot{q}_j)$ dans l'expression de l'énergie cinétique ;

- Le calcul de C se fait selon la relation suivante :

$$\begin{cases} C_{ij} = \sum_{k=1}^n c_{i,jk} \cdot \dot{q}_k \\ c_{i,jk} = \frac{1}{2} \left[\frac{\partial A_{ij}}{\partial q_k} + \frac{\partial A_{ik}}{\partial q_j} - \frac{\partial A_{jk}}{\partial q_i} \right] \end{cases} \quad \text{---- (2. 21)}$$

- Les éléments de Q se calculent selon la relation suivante :

$$Q_i = \frac{\partial U}{\partial q_i} \quad \text{---- (2. 22)}$$

2.6.2. Efforts exercés par l'organe terminal sur son environnement [1].

Pour accomplir une tâche donnée, l'organe terminal porté par le corps C_n doit exercer un effort statique sur l'environnement. Pour cela, on doit établir le modèle statique qui permet de calculer les couples et/ou forces articulaires correspondant à un effort donné par l'organe terminal.

Soit \mathbf{f}_{en} l'effort statique exercé par l'organe terminal sur son environnement.

$$\mathbf{f}_{en} = \begin{bmatrix} \mathbf{f}_{en} \\ \mathbf{m}_{en} \end{bmatrix} = \begin{bmatrix} f_x & f_y & f_z & m_x & m_y & m_z \end{bmatrix}^T \quad \text{---- (2. 23)}$$

Pour calculer l'effort Γ_e désignant les forces/moments que doivent fournir les actionneurs d'un robot pour que son organe terminal exerce l'effort \mathbf{f}_{en} sur son environnement, on utilise la relation suivante :

$${}^i \Gamma_e = {}^i J_n^T \cdot \mathbf{f}_{en} \quad \text{---- (2. 24)}$$

Avec \mathbf{f}_{en} exprimé dans le repère R_i lié au corps C_i .

En en tien compte en ajoutant le terme Γ_e au second membre de l'équation (2.20)

2.6.3. Formalisme de Newton-Euler.

2.6.3.1. Principe du formalisme de Newton Euler.

Les équations de Newton-Euler expriment le torseur dynamique des efforts extérieurs au centre de gravité G_j du corps C_j par les équations suivantes :

$$F_j = M_j \cdot \dot{V}_{C_j} \quad \text{---- (2. 25)}$$

$$M_{G_j} = I_{G_j} \cdot \dot{\omega}_j + \omega_j \times (I_{G_j} \cdot \omega_j) \quad \text{---- (2. 26)}$$

Où :

F_j : Résultantes des forces extérieures exercées sur le corps C_j ;

M_j : Masse du corps C_j ;

\dot{V}_{C_j} : Accélération du centre de gravité du corps C_j ;

M_{G_j} : Moment des efforts extérieurs exercés sur le corps C_j autour de son centre de gravité ;

I_{G_j} : Tenseur d'inertie du corps C_j par rapport à un repère parallèle à R_j et d'origine G_j le centre de gravité du corps.

ω_j : Vitesse de rotation du corps C_j ;

$\dot{\omega}_j$: Accélération de rotation du corps C_j ;

Nous présentons dans ce paragraphe, l'algorithme de Newton-Euler. Ce dernier étant fondé sur la double récurrence (avant et arrière), exprimant le torseur dynamique des efforts extérieurs en O_j (Origine de R_j) au lieu de G_j (Centre de gravité du corps C_j). La récurrence avant de la base du robot vers l'organe terminal, permet de calculer successivement les vitesses et accélérations des corps, puis leur torseur dynamique. La récurrence arrière de l'organe terminal vers la base, permet de calculer les efforts des actionneurs en exprimant pour chacune des articulations, le bilan des efforts.

2.6.3.2. Algorithme de Newton-Euler [1].

Dans les équations présentées dans ce paragraphe, toutes les grandeurs relatives à un corps C_j sont projetées dans le repère R_j qui lui est lié.

Notation :

${}^j\omega_k$, ${}^j\dot{\omega}_k$: Respectivement : vitesse et accélération angulaires du lien C_k exprimé dans le repère R_j ;

${}^j\dot{V}_k$: Accélération du point O_k exprimé dans le repère R_j ;

${}^j r_k$, ${}^j A_k$: (Voir § 2.2.1 : Transformation de coordonnées) ;

${}^j a_k$: Vecteur unitaire suivant l'axe z_k exprimé dans le repère R_j ;

M_k : Masse du corps C_k ;

${}^j F_k$: Résultante des forces extérieures appliquées sur le corps C_k exprimé dans le repère R_j ;

${}^j M_{O_k}$: Moments des efforts extérieurs appliqués sur le corps C_k autour du point O_k exprimé dans le repère R_j ;

${}^j J_j$: Tenseur d'inertie du corps C_j par rapport au repère R_j ;

S_j : Vecteur ayant pour origine le point O_j et pour extrémité, le centre de masse du corps C_j ;

${}^j MS_j$: Premier moment d'inertie du corps C_j autour de O_j , égal à : $M_j \cdot S_j$

${}^j f_k$: Résultante du torseur dynamique exercé sur le corps C_k par son antécédent, exprimé dans le repère R_j ;

${}^j m_k$: Moment du torseur dynamique exercé sur le corps C_k par son antécédent autour du point O_k , exprimé dans le repère R_j ;

${}^j f_{ek}$: Résultante du torseur dynamique exercé par le corps C_k sur l'environnement, exprimé dans le repère R_j ;

${}^j m_{ek}$: Moment du torseur dynamique exercé par le corps C_k sur l'environnement autour du point O_k , exprimé dans le repère R_j ;

${}^j g$: Accélération de la pesanteur exprimée dans le repère R_j ;

Récurrance avant :

La récurrence est initialisée par :

$${}^0 \omega_0 = 0, \quad {}^0 \dot{\omega}_0 = 0, \quad {}^0 \dot{V}_0 = -({}^0 g) \quad \text{---- (2. 27)}$$

On a pour $j=1,2,\dots,n$:

$${}^j \omega_{j-1} = {}^j A_{j-1} \cdot {}^{j-1} \omega_{j-1} \quad \text{---- (2. 28)}$$

$${}^j \omega_j = {}^j \omega_{j-1} + \bar{\sigma}_j \cdot \dot{q}_j \cdot {}^j a_j \quad \text{---- (2. 29)}$$

$${}^j \dot{\omega}_j = {}^j A_{j-1} \cdot {}^{j-1} \dot{\omega}_{j-1} + \bar{\sigma}_j \left(\ddot{q}_j \cdot {}^j a_j + {}^j \omega_{j-1} \times \dot{q}_j \cdot {}^j a_j \right) \quad \text{---- (2. 30)}$$

$${}^j U_j = {}^j \hat{\omega}_j + {}^j \hat{\omega}_j \cdot {}^j \hat{\omega}_j \quad \text{---- (2. 31)}$$

$${}^j \dot{V}_j = {}^j A_{j-1} \left({}^{j-1} \dot{V}_{j-1} + {}^{j-1} U_{j-1} \cdot {}^{j-1} r_j \right) + \sigma_j \left(\ddot{q}_j \cdot {}^j a_j + 2 \cdot {}^j \omega_{j-1} \times \dot{q}_j \cdot {}^j a_j \right) \quad \text{---- (2. 32)}$$

$${}^j F_j = M_j \cdot {}^j \dot{V}_j + {}^j U_j \cdot {}^j MS_j \quad \text{---- (2. 33)}$$

$${}^j M_{O_j} = {}^j J_j \cdot {}^j \dot{\omega}_j + {}^j \omega_j \times \left({}^j J_j \cdot {}^j \omega_j \right) + {}^j MS_j \times {}^j \dot{V}_j \quad \text{---- (2. 34)}$$

Récurrance arrière :

La récurrance est initialisée par :

$$\mathbf{f}_{n+1} = 0, \mathbf{m}_{n+1} = 0 \quad \text{---- (2. 35)}$$

On a pour $j=n, n-1, \dots, 1$:

$${}^j \mathbf{f}_j = {}^j F_j + {}^j \mathbf{f}_{j+1} + {}^j \mathbf{f}_{ej} \quad \text{---- (2. 36)}$$

$${}^{j-1} \mathbf{f}_j = {}^{j-1} A_j \cdot {}^j \mathbf{f}_j \quad \text{---- (2. 37)}$$

$${}^j \mathbf{m}_j = {}^j M_{Oj} + {}^j A_{j+1} \cdot {}^{j+1} \mathbf{m}_{j+1} + ({}^j r_{j+1} \times {}^j \mathbf{f}_{j+1}) + {}^j \mathbf{m}_{ej} \quad \text{---- (2. 38)}$$

Et enfin :

$$\Gamma_j = (\sigma_j \cdot {}^j \mathbf{f}_j + \bar{\sigma}_j \cdot {}^j \mathbf{m}_j)^T \cdot {}^j a_j \quad \text{---- (2. 39)}$$

2.7. Situation d'un robot dans son environnement.

L'environnement dans lequel évolue le robot est modélisé par un repère supposé absolu et fixe appelé « *repère atelier* ». Les formalismes décrits précédemment nous permettent de définir les paramètres géométriques, cinématiques et dynamiques relativement au repère lié à la base du robot. Pour calculer ces paramètres par rapport au repère atelier $(R_A, O_A, \vec{x}_A, \vec{y}_A, \vec{z}_A)$, on doit définir la situation de la base du robot dans le repère atelier par la matrice de transformation suivante :

$${}^A T_0 = \begin{bmatrix} {}^A s_0 & {}^A n_0 & {}^A a_0 & {}^A r_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^A A_0 & {}^A r_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (2. 40)}$$

Où :

$[{}^A A_0] = [{}^A s_0 \quad {}^A n_0 \quad {}^A a_0]$: Matrice (3x3) définissant l'orientation de la base par rapport au repère atelier ;

${}^A r_0$: Vecteur position de l'origine du repère $(R_0, O_0, \vec{x}_0, \vec{y}_0, \vec{z}_0)$ par rapport au repère atelier.

La situation d'un lien i relativement au repère atelier est définie par :

$${}^A T_i = {}^A T_0 \cdot {}^0 T_i \quad \text{---- (2. 41)}$$

2.8. Modélisation de l'objet à manipuler.

En pratique, l'utilisation des bras manipulateurs consiste comme son nom l'indique, dans la plus part des cas en la manipulation d'objets. Cette dernière peut désigner l'une de plusieurs tâches possibles comme la manutention, le soudage, l'assemblage etc.

Afin de pouvoir manipuler un objet quelconque avec un bras manipulateur, on doit connaître :

- Situation initiale de l'objet : Positionnement et orientation de l'objet dans le repère atelier ;
- La prise de l'objet : Dans le cas de la manutention de l'objet, on doit connaître la situation dans laquelle l'organe terminal doit se présenter relativement à la position et l'orientation de l'objet. A titre d'exemple pour ouvrir une porte, l'organe terminal doit se présenter devant la poignée et non pas devant le centre de gravité qui détermine la position de la porte ;
- Encombrement de l'objet : afin de pouvoir éviter les collisions avec les obstacles, on doit connaître la forme et les dimensions de l'objet à manipuler ;
- Paramètres inertiels et dynamiques : Masse total de l'objet, distribution de la masse (moments d'inertie) et les efforts extérieurs exercés sur l'objet.

Dans ce qui suit, on proposera un modèle qui pourra tenir compte de toutes les exigences citées ci-dessus.

2.8.1. Sphère d'encombrement de l'objet.

Afin que l'objet puisse être déplacé selon une trajectoire donnée, tout en évitant les obstacles présents dans l'espace de travail où le robot évolue, un objet ou un obstacle est représenté par une sphère qui l'enveloppe entièrement. La collision entre deux objets est détectée par un simple test qui calcul la distance entre les centres des deux sphères comme suit :

Soit deux objets représentés par les deux sphères (O_i, D_i) , (O_j, D_j) , avec

O_k : Centre de la sphère représentant l'objet k ;

D_k : Diamètre de la sphère représentant l'objet k ;

La distance entre les deux sphères est calculée selon la relation suivante :

$$d_{ij} = \|\overline{O_i O_j}\| \quad \text{---- (2. 42)}$$

Le test de collision est effectué comme suit :

$$Collision(i, j) = \begin{cases} \text{Vrai} & \text{si } d_{ij} \leq \frac{D_i + D_j}{2} \\ \text{Faux} & \text{sinon} \end{cases} \quad \text{---- (2. 43)}$$

2.8.2. Positionnement de l'objet dans l'espace opérationnel.

Pour représenter un objet Ob dans l'espace opérationnel, on lui fait associer un repère RO défini par son origine O_o qui coïncide avec le centre de la sphère d'encombrement de l'objet, ainsi que trois vecteurs so , no et ao qui forment un trièdre direct pour définir l'orientation de l'objet. Par conséquent, une matrice de transformation homogène sera associée à cet objet, cette dernière sera définie comme suit

$$TO = \begin{pmatrix} so_x & no_x & ao_x & ro_x \\ so_y & no_y & ao_y & ro_y \\ so_z & no_z & ao_z & ro_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{--- (2.44)}$$

La relation (2.44), définit le positionnement de l'objet i dans le repère atelier.

2.8.3. Prise de l'objet.

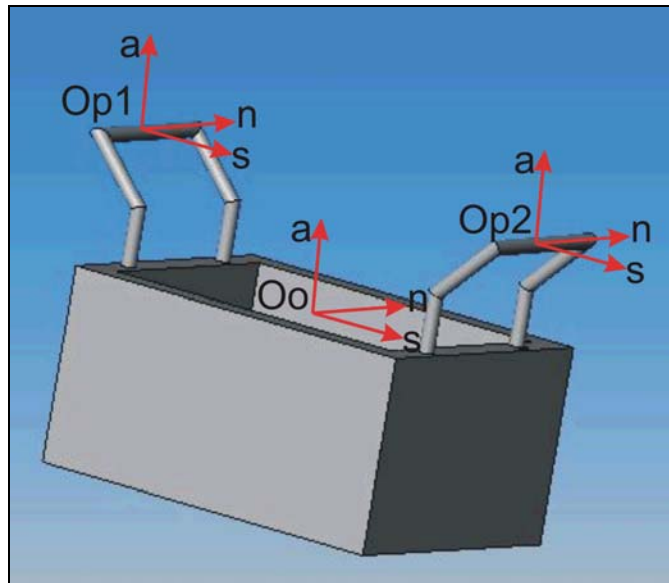


Figure 2. 6 : Repères associés à un objet manipulé par un ou plusieurs bras manipulateur.

Une prise détermine la position exacte où et dans quelle orientation doit se présenter l'organe terminal du bras manipulateur quand l'objet se trouve à une situation donnée. A une prise, on fait associer un repère défini relativement au repère lié à l'objet, et par conséquent, une prise est caractérisée par une matrice de transformation homogène TP définie relativement au repère de l'objet.

La situation de la prise de l'objet dans le repère atelier est calculée donc comme suit :

$${}^A TP = [TO][TP] \quad \text{--- (2. 45)}$$

En vue d'être manipulé par plusieurs bras manipulateurs, un objet peut avoir plusieurs prises comme montré dans la figure (2.6).

2.8.4. Positionnement et cinématique d'un objet saisi par un ou plusieurs robots.

On considère un objet *Ob* comprenant une ou plusieurs prises afin d'être manipulé par un ou plusieurs robots. On veut déterminer la position, la vitesse et l'orientation de cet objet à un instant donné selon la (les) situation(s) du (des) robot(s) portant cet objet. Une petite extension doit être faite dans la modélisation de l'organe terminal du robot exprimant la différence de ce lien par rapport aux autres liens du robot.

On considère que la base du robot se trouve à une situation ${}^A T_0$ relativement au repère atelier, et l'organe terminal se trouve à une situation ${}^0 T_{OT}$ par rapport à la base du robot. Le point prévu pour la saisie d'un objet est généralement loin de l'origine du lien (organe terminal) qui est liée à son axe de liaison avec le lien précédent. Il se peut aussi que la saisie doit être orientée autrement que l'organe terminal et ceci dépend de la géométrie de ce dernier. On considère dans le cas général, que la saisie se fait dans une situation définie relativement à celle de l'organe terminal par une matrice ${}^{OT} T_{Pr}$. Il en résulte que le point de saisi du robot sera défini dans le repère atelier par :

$${}^A T_{Pr} = \underbrace{{}^A T_0 \cdot {}^0 T_{OT}}_{{}^A T_{OT}} \cdot {}^{OT} T_{Pr} \quad \text{--- (2. 46)}$$

Deux cas sont à distinguer selon que cet objet soit manipulé par un ou par plusieurs robots :

a. Cas d'un objet manipulé par un seul robot.

Dans ce cas la liaison entre la prise de l'objet et le point de saisi du robot est généralement considérée complète. La prise de l'objet doit correspondre au point de saisi de l'organe terminal, ceci mène à que la situation du point de saisi et celle de la prise de l'objet, doivent être confondues (représentées dans le repère atelier).

Du côté de l'objet on a :

$${}^{atelier} T_{Prise} = {}^{atelier} T_{Objet} \cdot {}^{Objet} T_{Prise} \quad \text{--- (2. 47)}$$

Du côté du robot on a :

$${}^{atelier} T_{\text{point de saisi}} = {}^{atelier} T_{OT} \cdot {}^{OT} T_{\text{Point de saisi}} \quad \text{--- (2. 48)}$$

Les deux situations doivent être confondues, donc :

$${}^{atelier}T_{Prise} = {}^{atelier}T_{\text{point de saisie}} \Rightarrow {}^{atelier}T_{Objet} \cdot {}^{Objet}T_{Prise} = {}^{atelier}T_{OT} \cdot {}^{OT}T_{\text{Point de saisie}} \quad \text{---- (2. 49)}$$

Ou encore :

$${}^{atelier}T_{Objet} = {}^{atelier}T_{OT} \cdot {}^{OT}T_{\text{Point de saisie}} \cdot \left({}^{Objet}T_{Prise} \right)^{-1} \quad \text{---- (2. 50)}$$

La vitesse et l'accélération de l'objet peuvent être déterminées en utilisant les formules de distribution de vitesses et d'accélération d'un solide.

La vitesse linéaire du centre de gravité de l'objet exprimée dans le repère de l'organe terminal est donnée par [4] :

$${}^{Objet}V_{Objet} = {}^{Objet}V_n + {}^{Objet}\omega_n \wedge \left(\overrightarrow{r_{Objet}} \right) \quad \text{---- (2. 51)}$$

L'accélération est déterminée ainsi :

$${}^{Objet}\dot{V}_{Objet} = {}^{Objet}\dot{V}_n + {}^{Objet}\dot{\omega}_n \wedge \left(\overrightarrow{r_{Objet}} \right) + \left[{}^{Objet}\omega_n \wedge \left({}^{Objet}\omega_n \wedge \left(\overrightarrow{r_{Objet}} \right) \right) \right] \quad \text{---- (2. 52)}$$

Avec :

${}^{Objet}V_{Objet}$: Vitesse linéaire de l'origine du repère de l'objet exprimée dans le même repère ;

${}^{Objet}V_n$: Vitesse linéaire de l'origine de l'organe terminal exprimée dans le repère de l'objet ;

${}^{Objet}\omega_n$: Vitesse angulaire de l'objet qui est égale à celle de l'organe terminal (liaison complète) exprimée dans le repère de l'objet ;

${}^{Objet}\left(\overrightarrow{r_{Objet}} \right)$: Vecteur appliqué à l'origine de l'organe terminal et allant vers l'origine du repère de l'objet exprimée dans le même repère (celui de l'objet) ;

${}^{Objet}\dot{V}_{Objet}$: Accélération linéaire de l'origine du repère de l'objet exprimée dans le même repère ;

${}^{Objet}\dot{V}_n$: Accélération linéaire de l'origine de l'organe terminal exprimée dans le repère de l'objet ;

${}^{Objet}\dot{\omega}_n$: Accélération angulaire de l'objet qui est égale à celle de l'organe terminal (liaison complète) exprimée dans le repère de l'objet ;

b. Cas d'un objet manipulé par plusieurs robots.

Dans le cas d'un objet manipulé par plusieurs robots, l'utilisation d'une liaison complète entre le point de saisie et l'objet pose certaines limitations supplémentaires aux

déplacements des organes terminaux et mène parfois à des situations de blocage qui peuvent engendrer des détériorations de certains composants du robot ou même de l'objet à manipuler. L'utilisation de liaisons de classes inférieures pourra être plus utile dans ce cas. On donnera l'exemple de l'utilisation d'une liaison sphérique (rotule) pour le cas de deux robots.

Le mouvement relatif entre l'organe terminal et l'objet est réduit dans ce cas à la rotation dans tous les sens. La position de l'objet est définie selon les positions des organes terminaux contribuant à la manipulation, quant à l'orientation, elle est confiée à la position d'équilibre de l'objet.

2.8.5. Modèle dynamique de l'objet.

On considère un objet Ob possédant une masse M_{Ob} . Son centre de gravité par rapport au repère local (centre de la sphère) est donné par $CG_{Ob} = [CG_{Obx}, CG_{Oby}, CG_{Obz}]^T$. Sa distribution de masse par rapport au repère local de l'objet est donnée par le tenseur d'inertie.

$$T_{Ob} = \begin{pmatrix} (I_{Ob})_{xx} & -(I_{Ob})_{xy} & -(I_{Ob})_{xz} \\ -(I_{Ob})_{xy} & (I_{Ob})_{yy} & -(I_{Ob})_{yz} \\ -(I_{Ob})_{xz} & -(I_{Ob})_{yz} & (I_{Ob})_{zz} \end{pmatrix} \quad \text{----- (2. 53)}$$

En vue d'être manipulé par un robot pour effectuer un mouvement défini par la situation $X_{Ob}(t)$, la vitesse $\dot{X}_{Ob}(t) = \begin{bmatrix} v_{ob}(t) \\ \omega_{ob}(t) \end{bmatrix}$, et l'accélération $\ddot{X}_{Ob}(t) = \begin{bmatrix} \dot{v}_{ob}(t) \\ \dot{\omega}_{ob}(t) \end{bmatrix}$, l'organe terminal doit exercer un effort $f_{eo} = \begin{bmatrix} f_{eo} \\ m_{eo} \end{bmatrix}$ sur l'objet à manipuler. La force de cet effort est appliquée à un point défini (prise de l'objet) $P = [P_x, P_y, P_z]^T$ par rapport au repère lié à l'objet.

On a :

$X_{Ob}(t)$: Situation (position et orientation) de l'objet à l'instant t , exprimée relativement au repère de l'objet ;

$v_{ob}(t), \dot{v}_{ob}(t)$: Respectivement : vitesse et accélération linéaires du centre de gravité de l'objet à l'instant t , exprimées relativement au repère de l'objet ;

$\omega_{ob}(t), \dot{\omega}_{ob}(t)$: Respectivement : Vitesse et accélération angulaires de l'objet à l'instant t , exprimées relativement au repère de l'objet ;

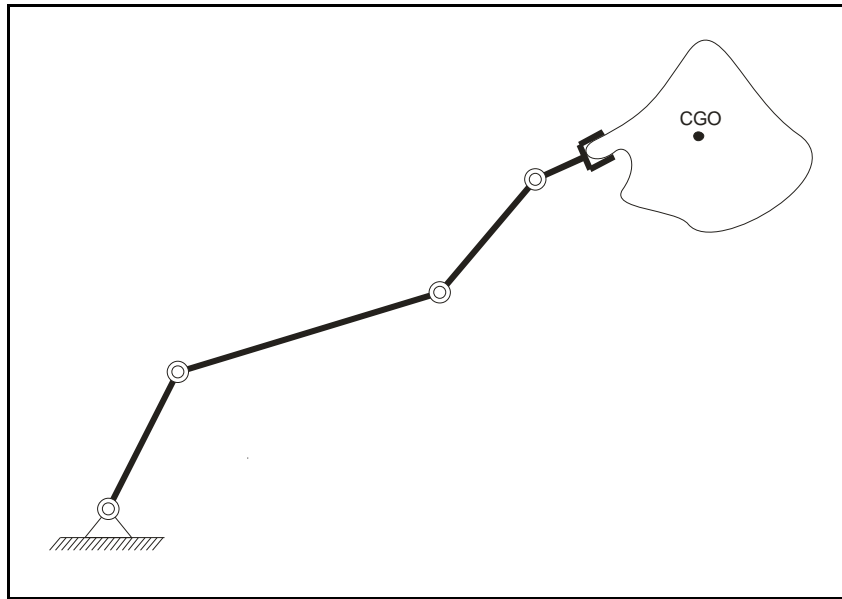


Figure 2. 7 : Objet manipulé par un robot.

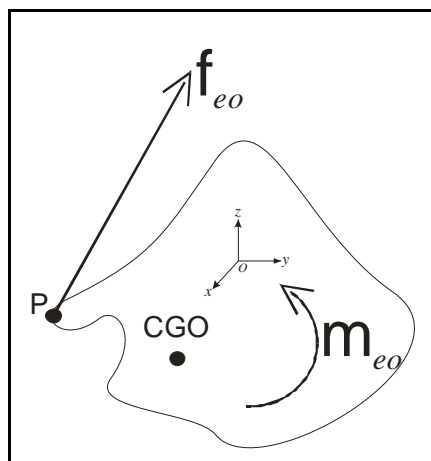


Figure 2. 8 : Isolation d'un objet manipulé par un robot.

On applique la loi fondamentale de la dynamique sur l'objet :

$$\begin{cases} \sum F = M_{Ob} \cdot \dot{v}_{Ob} \\ \sum \overrightarrow{M}_{/O} = T_{Ob} \cdot \dot{\omega}_{Ob} \end{cases} \quad \text{---- (2. 54)}$$

$$f_{eo} + M_{Ob} \cdot g = M_{Ob} \cdot \dot{v}_{Ob} \Rightarrow f_{eo} = M_{Ob} \cdot (\dot{v}_{Ob} - g) \quad \text{---- (2. 55)}$$

$$m_{eo} + \overrightarrow{OP} \wedge f_{eo} + \overrightarrow{OCG} \wedge M_{Ob} \cdot g = T_{Ob} \cdot \dot{\omega}_{Ob} \quad \text{---- (2. 56)}$$

$$\Rightarrow m_{eo} = -\overrightarrow{OP} \wedge f_{eo} - \overrightarrow{OCG} \wedge M_{Ob} \cdot g + T_{Ob} \cdot \dot{\omega}_{Ob} \quad \text{---- (2. 57)}$$

NB. : Lors de l'application de l'algorithme de Newton Euler pour la manipulation d'un objet, les efforts f_{eo} et m_{eo} doivent être injectés dans les équations (2.33) et (2.34) si $j = n$, pour tenir compte de la force exercée par l'organe terminal sur l'objet.

2.8.6. Objet manipulé par plusieurs robots – Coopération de robots.

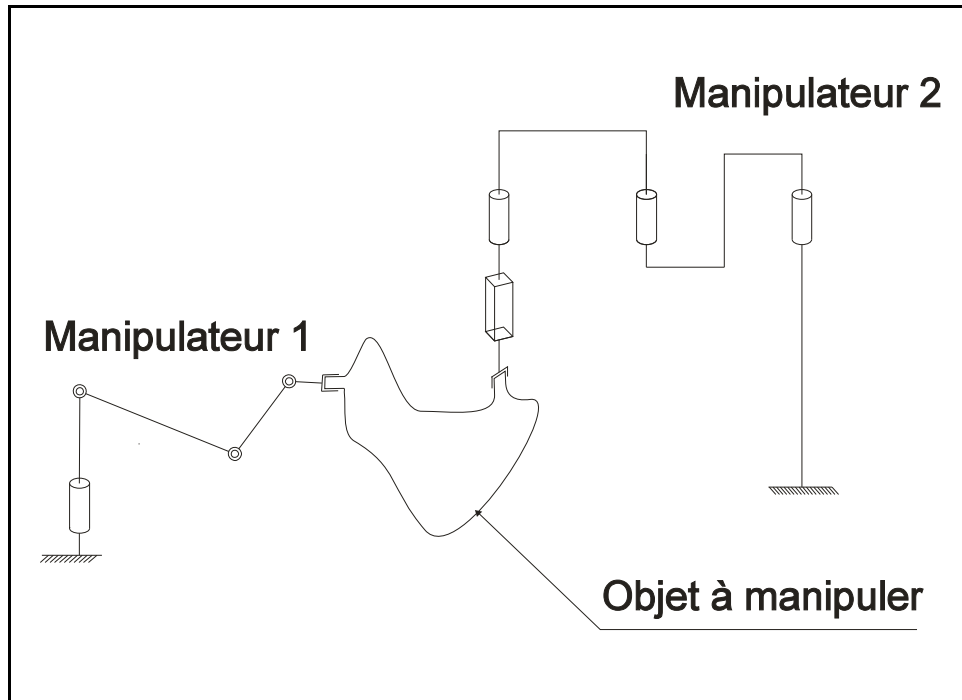


Figure 2. 9 : Coopération de deux robots pour la manipulation d'un objet.

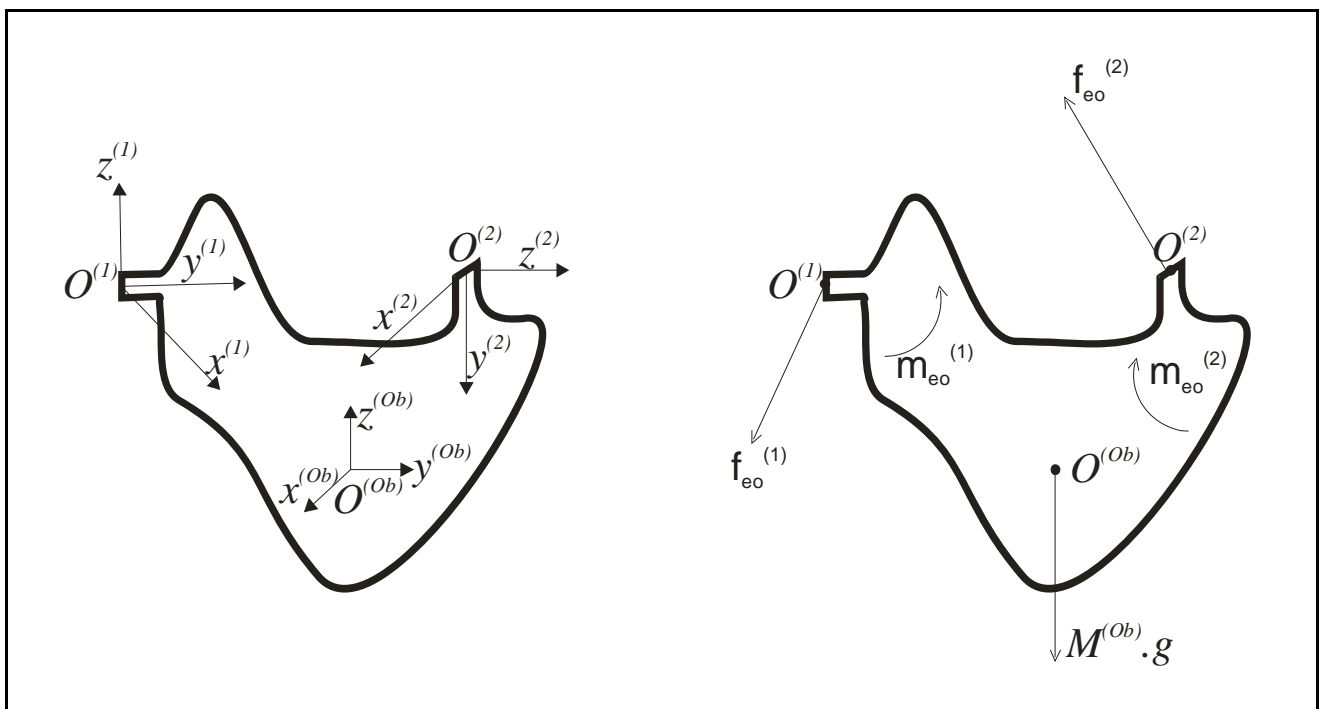


Figure 2. 10 : Isolation d'un objet manipulé par deux robots.

Dans le cas où un objet doit être manipulé par plusieurs robots, cet objet doit posséder autant de prises que de robots contribuant à la manipulation. Chaque organe terminal d'un robot k exerce un effort $\mathbf{f}_{eo}^k = \begin{bmatrix} \mathbf{f}_{eo}^k \\ \mathbf{m}_{eo}^k \end{bmatrix}$ sur l'objet à manipuler. Ce dernier subit la résultante des efforts extérieurs \mathbf{f}_{eo}^k .

On considère un objet Ob possédant p prises manipulé par p robots. En appliquant la loi de Newton sur l'objet on aura [19] :

$$Y = [L] \begin{bmatrix} \mathbf{f}_{eo}^1 & \mathbf{m}_{eo}^1 & \mathbf{f}_{eo}^2 & \mathbf{m}_{eo}^2 & \dots & \mathbf{f}_{eo}^p & \mathbf{m}_{eo}^p \end{bmatrix}^T \quad \text{---- (2. 58)}$$

Où :

$$Y = \begin{bmatrix} M_{Ob} \cdot I_3 & 0_{3 \times 3} \\ 0_{3 \times 3} & T_{Ob} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}}_{Ob} \\ \dot{\boldsymbol{\omega}}_{Ob} \end{bmatrix} + \begin{bmatrix} -M_{Ob} \cdot \mathbf{g} \\ \hat{\boldsymbol{\omega}}_{Ob} \cdot T_{Ob} \cdot \boldsymbol{\omega}_{Ob} \end{bmatrix} \quad \text{---- (2. 59)}$$

$$[L] = \begin{bmatrix} L^1 & L^2 & \dots & L^p \end{bmatrix} \quad \text{---- (2. 60)}$$

$$L^k = \begin{bmatrix} I_3 & 0_{3 \times 3} \\ -\tilde{\mathbf{r}}_{Ob}^k & I_3 \end{bmatrix}, \quad k = 1, 2, \dots, p \quad \text{---- (2. 61)}$$

\mathbf{r}_{Ob} : Vecteur allant du point d'application de la force \mathbf{f}_{eo} vers l'origine du repère lié à l'objet, exprimé relativement au même repère.

La résolution du système (2.58) repose sur l'inverse de la matrice L qui n'existe pas à cause de sa rectangularité, ce qui donnera naissance à un système redondant donnant possibilité à une infinité de solutions pour les efforts \mathbf{f}_{eo}^k .

Plusieurs approches ont été proposées dans la littérature, pour palier à ce problème comme en [19], on a proposé l'approche générale de la décomposition de la charge entre deux robots. En [191], deux méthodes ont été proposées : « *La méthode d'estimation de la charge* » et « *La méthode de la compensation de la charge* ». D'autres méthodes basées sur le calcul de la pseudo inverse de la matrice $[L]$ ont été utilisées avec plusieurs variantes selon le type de robots, ainsi que le mouvement donné à l'objet manipulé.

Plusieurs considérations entrent en jeu pour le choix de la méthode de résolution. Si on considère que la liaison entre l'objet et une liaison « *rotule sphérique* », l'effort transmis entre l'organe terminal du robot et l'objet, ne comporte que les composantes de la force, du fait que ce type de liaison élimine toute sorte de moment relatif.

Le problème peut être traité en deux phases :

- La première consiste à appliquer la loi fondamentale de la dynamique sur l'objet en ne considérant qu'une seule force f_{eo} extérieure appliquée à l'origine du repère lié à l'objet, ainsi qu'un seul moment m_{eo} appliqué à l'objet ;
- La deuxième phase consiste à équilibrer le moment m_{eo} par la décomposition de la force f_{eo} en plusieurs forces (selon le nombre de robots coopérant à l'opération de manipulation). La décomposition de f_{eo} suivra une loi de combinaison linéaire convexe et fera sorte que la décentralisation de cette force donnera un moment égal à m_{eo} .

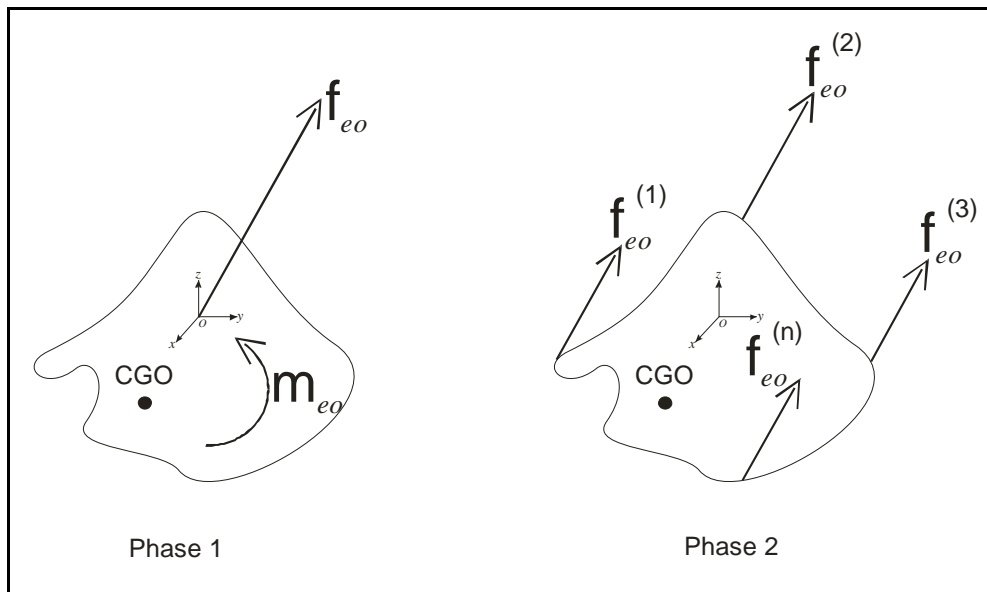


Figure 2. 11 : Distribution de la charge d'un objet manipulé par plusieurs robots.

Première phase :

On applique la loi fondamentale de la dynamique sur l'objet :

$$\begin{cases} \sum F = M_{Ob} \cdot \dot{v}_{Ob} \\ \sum \overrightarrow{M}_{/O} = T_{Ob} \cdot \dot{\omega}_{Ob} \end{cases} \quad \text{---- (2. 62)}$$

$$f_{eo} + M_{Ob} \cdot g = M_{Ob} \cdot \dot{v}_{Ob} \Rightarrow f_{eo} = M_{Ob} \cdot (\dot{v}_{Ob} - g) \quad \text{---- (2. 63)}$$

$$m_{eo} + \overrightarrow{OCG} \wedge M_{Ob} \cdot g = T_{Ob} \cdot \dot{\omega}_{Ob} \Rightarrow m_{eo} = -\overrightarrow{OCG} \wedge M_{Ob} \cdot g + T_{Ob} \cdot \dot{\omega}_{Ob} \quad \text{---- (2. 64)}$$

Deuxième phase :

On considère que l'on possède p robots coopérant à cette tâche. La force f_{eo} doit satisfaire les conditions suivantes :

$$\begin{cases} f_{eo} = \sum_{j=1}^p f_{eo}^j = \sum_{j=1}^p \lambda_j \cdot f_{eo} \\ \sum_{j=1}^p \lambda_j = 1 \\ \left[\sum_{j=1}^p \left(M \left(\frac{f_{eo}^j}{O} \right) \right) \right] + \overline{OCG} \wedge M_{ob} \cdot g = m_{eo} \end{cases} \quad \text{---- (2.65)}$$

La résolution du système (2.65) consiste à trouver les coefficients λ_j pour $j = 1, 2, 3, \dots, p$.

Cas de manipulation par deux robots.

On considère que l'objet possède deux prises, les points d'applications des deux forces f_{eo}^1 et f_{eo}^2 sont connus (respectivement P_1 et P_2). Le système (2.65) pourra s'écrire sous la forme suivante :

$$\begin{cases} f_{eo} = f_{eo}^1 + f_{eo}^2 = \lambda \cdot f_{eo} + (1 - \lambda) \cdot f_{eo} \\ \overline{OP_1} \wedge \lambda \cdot f_{eo} + \overline{OP_2} \wedge (1 - \lambda) \cdot f_{eo} = m_{eo} \\ \lambda \in \mathfrak{R} \end{cases} \quad \text{---- (2.66)}$$

Ou encore :

$$\begin{cases} f_{eo} = f_{eo}^1 + f_{eo}^2 = \lambda \cdot f_{eo} + (1 - \lambda) \cdot f_{eo} \\ \lambda \cdot (\overline{OP_1} \wedge f_{eo}) + (1 - \lambda) \cdot (\overline{OP_2} \wedge f_{eo}) = m_{eo} \\ \lambda \in \mathfrak{R} \end{cases} \quad \text{---- (2.67)}$$

On pose

$$\begin{cases} U = \overline{OP_1} \wedge f_{eo} \\ V = \overline{OP_2} \wedge f_{eo} \\ W = m_{eo} \end{cases} \quad \text{---- (2.68)}$$

De la deuxième équation du système (2.67) on aura :

$$\lambda U + (1 - \lambda)V = W \quad \text{---- (2.69)}$$

$$\lambda \cdot \begin{pmatrix} U_x \\ U_y \\ U_z \end{pmatrix} + (1 - \lambda) \cdot \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix} = \begin{pmatrix} W_x \\ W_y \\ W_z \end{pmatrix} \quad \text{---- (2.70)}$$

$$\begin{cases} \lambda(U_x - V_x) = W_x - V_x \\ \lambda(U_y - V_y) = W_y - V_y \\ \lambda(U_z - V_z) = W_z - V_z \end{cases} \quad \text{---- (2.71)}$$

L'une des équations du système (2.71) suffit pour trouver λ . On choisit celle qui comporte le coefficient de λ non nul.

N.B. Les quantités $(U_x - V_x)$, $(U_y - V_y)$ et $(U_z - V_z)$ ne peuvent s'annuler ensemble du fait que les deux points P_1 et P_2 ne sont pas confondus.

2.9. Conclusion.

Dans ce chapitre on a présenté les grandes lignes de la modélisation des robots en général avec un intérêt particulier qui a été porté aux bras manipulateurs à chaînes ouvertes. Commençant par la description géométrique directe qui a été décrite d'une manière générale en choisissant la notation de Denavit-Hartenberg et obtention des matrices de transformations homogènes régissant les positions des différents liens d'un robot. Quant à la géométrie inverse, la complexité du problème ne nous a pas été favorable pour donner une méthode générale pour décrire le modèle géométrique inverse. On s'est contenté de donner les définitions nécessaires et on a laissé la tâche de description du modèle pour le chapitre 4 en traitant un mécanisme spécifié.

Pour la modélisation dynamique, on a présenté les deux formalismes : « *Lagrange* » et « *Newton-Euler* » avec un accent qui a été porté sur le second pour son caractère récursif qui nous favorisera la programmation informatique.

L'étude faite dans ce chapitre nous a permis de préparer les différents modèles mathématiques régissant un robot ou un objet manipulé d'une manière algorithmique facilitant la programmation informatique à l'exception des modèles définissant le mouvement qui feront l'objet du prochain chapitre.

CHAPITRE 3

TRAJECTOIRES & GÉNÉRATION DE MOUVEMENT.

3.1. Introduction.

La planification des trajectoires a été le sujet d'un nombre très important de recherches aussi bien en robotique, que dans d'autres domaines. Qu'il soit un problème de stationnement ou chemin d'un véhicule auto guidé, trajectoires pour un bras manipulateur utilisé pour une tâche de soudure ou peinture en industrie automobile, marche d'un robot humanoïde, ou animation des personnages en jeux vidéo, Le terme « *Trajectoire* » trouvera toujours sa place dans la liste des tâches les plus complexes en modélisation ou en programmation du système étudié.

Vu la forte complexité du problème général de la planification de trajectoires, ce dernier ne semble pas soluble tel quel. Il se décompose naturellement en classes de sous problèmes dont certains, moins difficiles, peuvent trouver une solution. Cette subdivision de la complexité se trouve alors inmanquablement liée à plusieurs paramètres : la dimension de l'espace de travail, le nombre et la nature des objets mobiles, les composants constituant le robot ainsi que les exigences liées au mouvement et à l'énergie disponible.

Dans ce chapitre nous allons aborder le problème de planification de trajectoire par ses différents aspects, ainsi qu'une synthèse des recherches menées dans cet axe, et quelques méthodes de résolution utilisées.

3.2. Divers aspects de la planification de trajectoires.

Dans un premier temps, commençant par l'aspect algébriste et considérons l'espace vectoriel E , et P_1, P_2 deux éléments de cet espace. Un problème de planification de trajectoire peut être formulé comme suit : « *Trouver une séquence d'éléments de l'espace E , joignant les deux éléments P_1 et P_2* ». Une telle formulation peut avoir deux sous

aspects : « *continue* » si la séquence d'éléments est continue ou « *discret* », et dans ce cas les éléments doivent être rapprochés selon la précision voulue.

Une projection de la formulation précédente dans l'espace opérationnel d'un robot donné, donnera la définition suivante : « *Pour un robot A évoluant dans un environnement W donné, déterminer pour A un mouvement lui permettant de se déplacer entre deux configurations données* ». En effet, cette formulation n'est pas encore réaliste, le mouvement demandé doit satisfaire un certain nombre de critères. Ceux-ci découlent de plusieurs facteurs de natures diverses et dépendent généralement des caractéristiques du robot, de l'environnement et du type de la tâche à exécuter. En l'occurrence, les contraintes relatives au robot concernant sa géométrie, sa cinématique et sa dynamique et leur prise en compte peut être complexe selon l'architecture initiale considérée. Cette architecture pouvant correspondre à un système articulé d'objets rigides tel qu'un bras manipulateur, une main à plusieurs doigts ou un véhicule tractant des remorques, ou encore à plusieurs systèmes de robots à coordonner tels que des bras manipulateurs ou des robots mobiles de type voiture évoluant dans un réseau routier. Les contraintes émanant de l'environnement concernent essentiellement la non collision aux obstacles fixes encombrant W et la prise en compte d'interactions de contact avec le robot. L'évitement d'obstacles dépend de la géométrie de l'environnement et est commun à toutes les tâches robotiques.

De plus aucune hypothèse simplificatrice n'est faite sur la géométrie du robot ou sur celle de l'environnement. Les critères à satisfaire pendant la résolution du problème de planification concernent le fait qu'une solution doit optimiser une fonction de coût exprimée en terme de la distance parcourue par le robot entre les deux configurations extrêmes, de la durée ou de l'énergie nécessaires à l'exécution de son mouvement. D'autres critères peuvent être également considérés tels que la prise en compte de distance de sécurité aux obstacles pour un robot mobile ou manipulateur ou encore la « *qualité* » et la stabilité des prises pour une main articulée. Face à la nature aussi diverse de ces aspects et aux difficultés qu'elle peut induire sur un processus de résolution, la plupart des travaux proposés dans le domaine de la planification de mouvement ont porté sur la considération de certaines instances du problème général. Nous présentons dans la suite de ce chapitre les principaux concepts et approches développés à cet effet.

Ainsi planifier une trajectoire pour un corps solide dans l'espace cartésien revient à planifier la trajectoire d'un point dans l'espace de travail. Si l'on considère maintenant les obstacles de l'environnement, on se rend compte que les « *plonger* » dans l'espace des

configurations du système n'est pas une tâche aisée, surtout si le système comporte de nombreuses variables de configuration et que par conséquent la dimension de l'espace articulaire est élevée.

Après ce bref exposé de la problématique de la planification de trajectoire, plusieurs questions restent en suspens :

- Comment exprimer les obstacles dans l'espace des configurations ?
- Comment trouver un chemin dans l'espace opérationnel libre ?
- Si plusieurs chemins sont possibles, y a-t-il un moyen pour choisir un chemin optimum ?
- En présence de contraintes de natures différentes (géométrique, cinématique, dynamique) comment est-ce que le choix doit satisfaire un compromis entre toutes les contraintes ?
- Une fois que le chemin est désigné, comment générer un mouvement (dans l'espace articulaire) produisant la trajectoire trouvée dans l'espace opérationnel ?

3.3. Classification des types de méthodes utilisées.

Traditionnellement, les méthodes de planification de trajectoires ont été classées selon la connaissance de l'espace de travail. Ce dernier peut être totalement connue, ou connu dans un environnement partiel de l'espace du robot. On distingue donc les « *méthodes locales* » des « *méthodes globales* ». Il existe néanmoins de nombreux travaux de couplage entre les deux approches.

3.3.1. Méthodes globales.

Les méthodes globales [69, 70, 186] supposent une connaissance totale de l'environnement. Ces méthodes opèrent généralement en deux phases. La première phase consiste à modéliser l'espace libre accessible par le robot. Ce qui est fait généralement hors ligne une seule fois pour le robot dans son environnement. La deuxième phase se base sur le modèle généré par la première phase afin de trouver une trajectoire. Cette phase correspond à des requêtes effectuées en ligne qui sont rapides, vu que le travail le plus lourd en temps de calcul est effectué hors ligne. Les méthodes globales ne sont pas adaptées aux environnements dynamiques ou bien aux environnements qui changent lors du déplacement du robot. Tout changement dans l'environnement du robot, nécessite de recalculer la première phase qui est très gourmande en temps de calcul. De plus, le temps

de calcul augmente exponentiellement avec le nombre de degrés de liberté du robot. Parmi les méthodes globales les plus utilisées, on cite : « *La méthode des roadmaps* » et « *la méthode de décomposition en cellules* ».

3.3.2. Méthodes locales.

Les méthodes locales [69, 70, 186] ne nécessitent qu'une perception de l'environnement immédiat du robot. La plupart des stratégies locales se fondent sur une approche où un capteur, comme un sonar ou une caméra, fournit des informations sur le voisinage immédiat du robot, qui sont intégrées à un ensemble de règles hiérarchiques fournissant le comportement à adopter. De tels schémas prennent rarement en compte les critères d'optimalité dans la recherche du chemin à suivre. Ces méthodes locales utilisent la connaissance courante de l'état du système et de l'environnement pour en réduire le mouvement suivant. Ces stratégies ne nécessitant pas de calcul a priori, elles sont particulièrement appropriées aux tâches dont le seul objectif est d'atteindre une configuration désirée sans contrainte sur la trajectoire à réaliser. Parmi les méthodes locales utilisées, on trouve souvent « *la méthode des fonctions de potentiels* »[190]. Cette dernière est basée sur la construction de champs de potentiels artificiels qui peuvent être attractifs ou répulsifs selon qu'ils sont associés à l'objectifs ou à un obstacle de l'environnement. Le défaut majeur d'une telle approche réside dans le fait que des minima locaux peuvent éventuellement faire échouer le processus de planification [69].

3.3.3. Méthodes mixtes.

Les méthodes mixtes, sont une combinaison des deux types de méthodes globales et locales comme celle proposée en [186], et qui consiste à utiliser une méthode locale pour avancer vers le but. Dans le cas où une configuration de blocage est rencontrée, une méthode globale prend le relais afin de trouver la meilleure façon de contourner l'obstacle qui bloque le robot. Une fois cet obstacle contourné, le robot continue à évoluer vers son but en utilisant la méthode locale. Cette méthode est adaptée aux robots sériels placés dans des environnements fortement encombrés.

3.3.4. Autres méthodes.

Ces dernières années, les recherches se sont orientées vers des méthodes de planification de trajectoires basées sur les cartes probabilistes « *Probabilistic roadmaps* ». Ces méthodes ont été introduites par les travaux de Kavraki [84, 86, 89, 148, 154, 159, 160]

afin de remédier au problème de temps de calcul des méthodes globale. Cette méthode permet de modéliser l'espace de travail en choisissant des positions au hasard et en les plaçant dans un diagramme qui représente les chemins sans collisions entre les différentes positions. Pour chaque nouvelle position insérée, les chemins vers les autres positions sont vérifiées et rajoutées dans le diagramme. Plusieurs variantes de cette méthode ont été proposées pour la rendre soit plus rapide comme la méthode basée sur le graphe de visibilité [147], soit pour résoudre le problème des couloirs étroits avec la méthode de l'axe médian [186], soit pour essayer d'appliquer ces méthodes à des environnements dynamiques [154].

D'autres méthodes inspirées de la biologie ont également été proposées, comme par exemple les réseaux de neurones [75, 175], ou les algorithmes génétiques [127].

3.4. Commandabilité, planification & suivi de trajectoire.

En pratique, la résolution d'un problème de planification de trajectoire formulé comme suit « *Comment conduire un système dynamique d'un état initial X_0 vers un état final X_f* », exige l'existence d'au moins une fonction d'entrée $u(t)$ produisant une trajectoire passant par les états X_0 et X_f .

On écrit souvent les équations régissant l'évolution d'un système dynamique commandé sous la forme suivante :

$$\frac{dX}{dt} = f(X, u) \quad \text{--- (3. 1)}$$

Où X désigne l'état (la configuration), et u la commande du système.

Un état X_j est dit « *atteignable* » à partir de l'état X_i s'il existe un temps fini T et une fonction d'entrée $u(t)$ définie sur $[t_i, t_i + T]$ tels que :

$$X(t_i + T, X_0, u) = X_f \quad \text{--- (3. 2)}$$

Cette notion d'atteignabilité conduit au concept de commandabilité d'un système dynamique explicité dans la définition suivante : Le système (3.1) est « *localement commandable* » en X_i s'il existe un voisinage de X_i tel que X_i soit atteignable à partir de chaque élément du voisinage. Le système est « *globalement commandable* » si tout état de l'espace considéré est atteignable à partir de tout état initial de cet espace.

La détermination de la fonction de commande u consiste en la planification de la trajectoire, elle permet de calculer une trajectoire de référence définie sur $[0, T]$ satisfaisant les contraintes, et rejoignant les états $X(0)$ et $X(T)$.

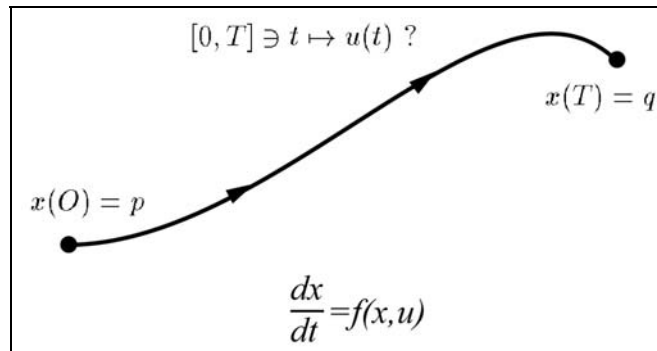


Figure 3. 1 : Principe de la détermination de la trajectoire de référence.

Si la condition initiale est exactement $X(0)$ spécifiée, alors on suit exactement la trajectoire de référence calculée. Si pendant l'exécution, la condition initiale est décalée, cette erreur initiale va se propager le long de la trajectoire. L'erreur s'atténue si le système est stable et s'amplifie si le système est instable. Dans ce cas il faut rajouter des termes correctifs Δu , pour assurer la convergence de cette erreur vers zéro, c'est ce qui consiste en « *le suivi de la trajectoire* ». Ceci peut se faire par des techniques classiques d'automatique linéaire reposant sur l'étude au premier ordre des écarts à la trajectoire de référence.

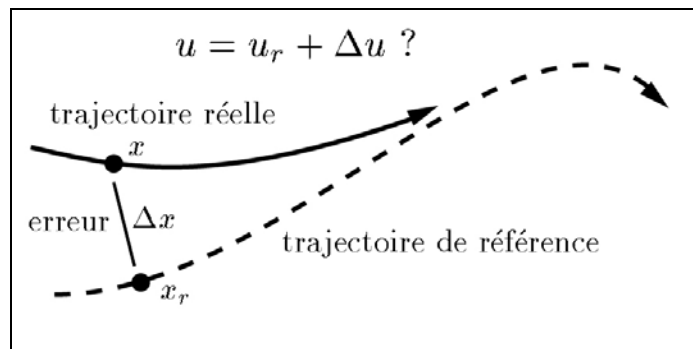


Figure 3. 2 : Suivi d'une trajectoire.

3.5. Critère de performance dans la recherche de trajectoires.

Comme vu précédemment, les diverses formulations du problème de planification de mouvement, mènent à résoudre un problème d'optimisation, où la fonction objectif appelée « *critère de performance* » représente un coût à minimiser lors du déplacement. Pouvant être représenté par plusieurs formes, le critère de performance est généralement formulé

par une fonctionnelle de type intégral. Cette dernière est donnée sous la forme générale suivante :

$$F(u, T) = \int_0^T L(x(t), u) dt \quad \text{--- (3.3)}$$

Où :

L : représente le lagrangien qui à son tour peut prendre diverses formes selon l'objectif visé ;

T : temps de transfert total entre les configurations initiale et finale ;

$x(t)$: vecteur des variables d'état à l'instant $t \in [0, T]$, pour les mouvements libres⁸, ce vecteur représente les variables articulaires de position, vitesse et accélération ;

u : Vecteur qui représente la commande à optimiser et qui est considéré comme une inconnue du problème, pour les mouvement libres⁸, il représente les couples moteurs ;

l'équation (3.3) peut donc prendre la forme suivante :

$$F(\Gamma, T) = \int_0^T L(q(t), \dot{q}(t), \ddot{q}(t), \Gamma) dt \quad \text{--- (3.4)}$$

Dans la littérature spécialisée, une variété de critères destinés à la génération des mouvements, comme le critère « *durée de parcours* », le critère « *effort quadratique* », ou une combinaison linéaire de ces deux critères d'où l'appellation « *critère mixte* ».

Dans le tableau suivant, on résume les différentes formes du critère de performance utilisée dans la littérature.

Tableau 3. 1 : Formes du critère de performance utilisées.

Critère de performance	Forme mathématique du Lagrangien
Durée de parcours	$L(q(t), \dot{q}(t), \ddot{q}(t), \Gamma) = 1$
Critère mixte durée de parcours / effort quadratique	$L(q(t), \dot{q}(t), \ddot{q}(t), \Gamma) = \beta + (1 - \beta) \sum_{j=1}^n \left(\frac{\Gamma_j(t)}{(\Gamma_j)_{\max}} \right)$ <p>Avec :</p> <p>Avec $\beta \in [0,1]$</p> <p>$(\Gamma_j)_{\max}$: l'effort maximum délivré par la $j^{\text{ème}}$ articulation.</p>

⁸ Voir « *Mouvement point à point* » (§ 3.6.2.1)

3.6. Espace de planification de la trajectoire.

Comme un robot peut être décrit dans les deux espaces : Espace cartésien et/ou espace articulaire, il en est de même pour la trajectoire. Distinguons d'abord les classes de mouvements suivantes :

- Le mouvement entre deux points avec trajectoire libre entre les points ;
- Le mouvement entre deux points via des points intermédiaires, spécifiés pour éviter des obstacles, avec trajectoire libre entre les points intermédiaires ;
- Le mouvement entre deux points avec trajectoire contrainte entre les points (rectiligne ou arc de cercle par exemple) ;
- Le mouvement entre deux points via des points intermédiaires avec trajectoire contrainte entre les points intermédiaires.

De ces caractéristiques de mouvements cités, nous pouvons faire une classification primaire des trajectoires entre deux points, et nous distinguons les « *trajectoires libres* » des « *trajectoires contraintes* ».

Pour le cas des trajectoires libres, la génération de mouvement peut se faire directement dans l'espace articulaire, par contre, pour les trajectoires contraintes, il est préférable de raisonner dans l'espace opérationnel.

3.6.1. Génération de mouvement et système de commande [1].

Les deux figures suivantes montrent une schématisation du système de commande pour les deux cas : « *génération de mouvement dans l'espace articulaire* » et « *génération de mouvement dans l'espace opérationnel* ».

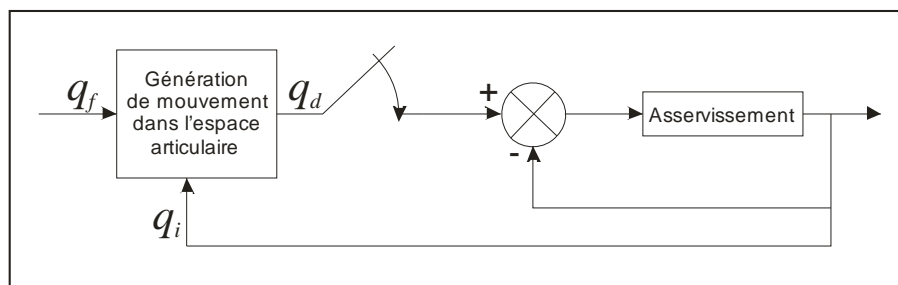


Figure 3. 3 : Système de commande pour la génération de mouvement dans l'espace articulaire.

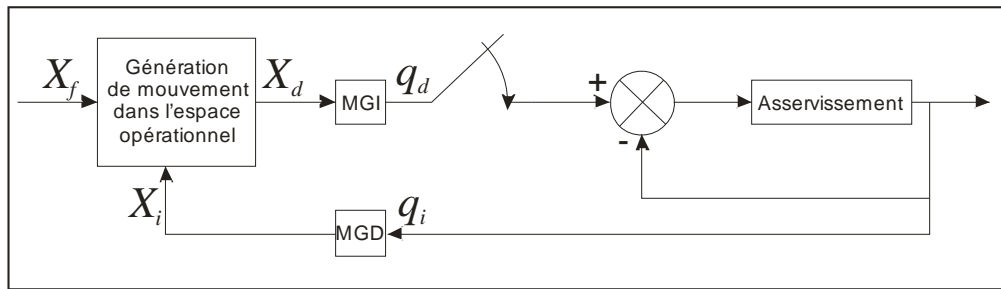


Figure 3. 4 : Système de commande pour la génération de mouvement dans l'espace opérationnel.

Le tableau suivant récapitule la différence entre les deux approches.

Tableau 3. 2 : Comparaison entre la planification de trajectoire dans l'espace articulaire et l'espace opérationnel.

Génération de mouvement dans l'espace articulaire	Génération de mouvement dans l'espace opérationnel.
<ul style="list-style-type: none"> • Elle nécessite moins de calcul en ligne, puisqu'il n'y a pas d'appel aux modèles inverses ; • Le mouvement n'est pas affecté par le passage sur les configurations singulières ; 	<ul style="list-style-type: none"> • Elle implique la transformation en coordonnées articulaires de chaque point de la trajectoire ; • Elle peut être mise en échec lorsque la trajectoire calculée passe par une position singulière ;
<ul style="list-style-type: none"> • Les contraintes de vitesses et de couples maximaux sont directement déduites des limites physiques des actionneurs. 	<ul style="list-style-type: none"> • Elle est mise en échec chaque fois que les points de la trajectoire engendrée ne sont pas dans le volume accessible du robot. • Les limites en vitesse et en couple sont exprimées par des valeurs de performance moyennes, on risque donc d'imposer au robot de travailler en deçà de ses capacités réelles.

3.6.2. Planification de trajectoire dans l'espace articulaire.

On considère un robot à n degrés de liberté. Soit q^i et q^f les vecteurs des coordonnées articulaires correspondant respectivement aux configurations initiale et finale. On désigne

par k_v et k_a les vecteurs des vitesses et accélérations articulaires maximales. Ces deux derniers étant caractéristiques techniques des actionneurs utilisés [1].

Comme déjà montré au (§ 3.4), le temps de transfert t_f devra être calculé auparavant.

Le problème étant de trouver une fonction $q(t)$ satisfaisant le système suivant :

$$\begin{cases} q(0) = q^i \\ q(t_f) = q^f \end{cases} \quad \text{--- (3.5)}$$

3.6.2.1. Trajectoire sans contraintes – Mouvement point à point (P.T.P).

Dans le tableau suivant, on récapitule les différentes fonctions permettant de satisfaire le système (3.5) trouvées dans la littérature [1], [41].

Tableau 3. 3 : Récapitulatif des lois utilisées pour la génération de mouvement dans l'espace articulaire.

Intitulé de la loi	Expression horaire	Temps de transfert global minimum
Interpolation linéaire	$q_j(t) = (q^i)_j + \frac{t}{t_f} D_j$ <p>Avec $D_j = (q^f)_j - (q^i)_j$</p>	$(t_f)_j = \frac{ D_j }{(k_v)_j}$
Polynôme de degré trois	$q_j(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$ <p>Avec :</p> $a_0 = q^i ; a_1 = 0 ; a_2 = \frac{3}{(t_f)^2} D_j ;$ $a_3 = \frac{2}{(t_f)^3} D_j ; D_j = (q^f)_j - (q^i)_j$	$(t_f)_j = \max \left[\frac{3 D_j }{2(k_v)_j}, \sqrt{\frac{6 D_j }{(k_a)_j}} \right]$
Polynôme de degré cinq	$q_j(t) = (q^i)_j + 10.\lambda^3 - 15.\lambda^4 + 6.\lambda^5$ <p>Avec $\lambda = \frac{t}{(t_f)_j} ; D_j = (q^f)_j - (q^i)_j$</p>	$(t_f)_j = \max \left[\frac{15 D_j }{8(k_v)_j}, \sqrt{\frac{10 D_j }{\sqrt{3}(k_a)_j}} \right]$

Intitulé de la loi	Expression horaire	Temps de transfert global minimum
Loi Bang-Bang	<ul style="list-style-type: none"> Pour $0 \leq t \leq \frac{(t_f)_j}{2}$ $q_j(t) = (q^i)_j + 2\lambda^2 \cdot D_j$ Pour $\frac{(t_f)_j}{2} \leq t \leq t_f$ $q_j(t) = (q^i)_j + [-1 + 4\lambda - 2\lambda^2] D_j$ <p>Avec $\lambda = \frac{t}{(t_f)_j}$; $D_j = (q^f)_j - (q^i)_j$</p>	$(t_f)_j = \max \left[\frac{2 D_j }{(k_v)_j}, 2\sqrt{\frac{ D_j }{(k_a)_j}} \right]$

La loi qui est souvent utilisée s'appelle « *Bang-Bang avec palier de vitesse* » ou encore « *Loi trapèze* ». Cette loi comporte trois phases de mouvement (Accélération – Mouvement uniforme – Décélération) et elle est décrite comme suit :

$$\begin{cases} q_j(t) = (q^i)_j + \frac{1}{2} t^2 \cdot (k_a)_j \cdot \text{sgn}(D_j) & \text{pour } 0 \leq t \leq \tau_j \\ q_j(t) = (q^i)_j + \left(t - \frac{\tau_j}{2} \right) \cdot (k_v)_j \cdot \text{sgn}(D_j) & \text{pour } \tau_j \leq t \leq (t_f)_j - \tau_j \\ q_j(t) = (q^f)_j - \frac{1}{2} [(t_f)_j - t]^2 \cdot (k_a)_j \cdot \text{sgn}(D_j) & \text{pour } (t_f)_j - \tau_j \leq t \leq (t_f)_j \end{cases} \quad \text{----(3. 6)}$$

Avec

$$\begin{cases} \tau_j = \frac{(k_v)_j}{(k_a)_j} \\ D_j = (q^f)_j - (q^i)_j \end{cases} \quad \text{----(3. 7)}$$

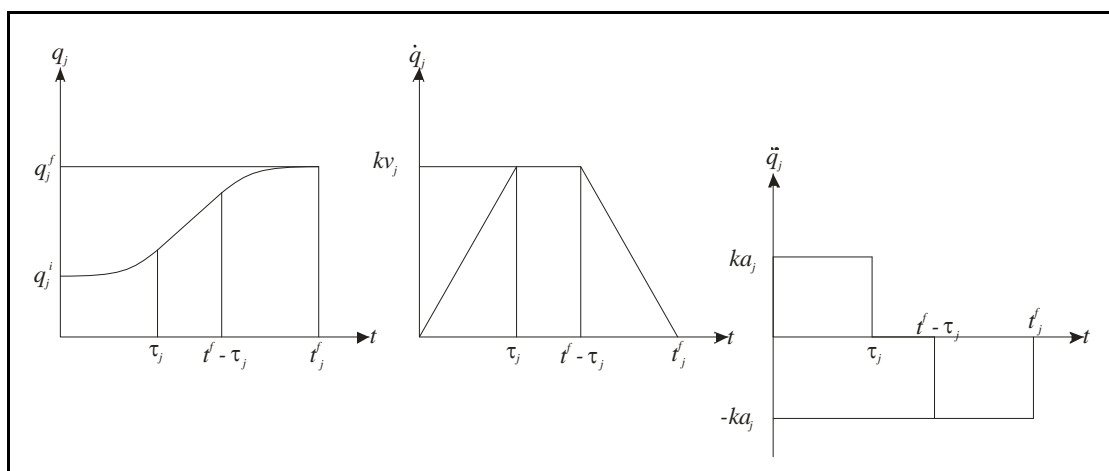


Figure 3. 5 : Évolution des positions, vitesses et accélérations sur l'articulation j avec une loi trapèze.

Et le temps de parcours minimal pour l'articulation j :

$$(t_f)_j = \frac{(k_v)_j}{(k_a)_j} + \frac{|D_j|}{(k_v)_j} \quad \text{---(3. 8)}$$

3.6.2.2. Trajectoire entre deux points avec des points intermédiaires.

Ce type de trajectoire est contraint par des configurations intermédiaires imposées. Cela peut être le cas si la tâche du manipulateur est définie de manière à éviter des obstacles et/ou pour des opérations de « *pick and place* » simples si on veille à éloigner/approcher l'objet correctement des sites de prise et de dépôt. Dans ce dernier cas, il suffira de spécifier une position intermédiaire sur la normale à la surface de dépôt qui passe par le point de prise : en imposant à la pince de passer par cette position intermédiaire, on obtiendra un mouvement admissible pour le départ « *lift-off* ». On fera de même pour la phase d'approche « *set-down* » nécessaire au dépôt de l'objet manipulé. De plus, en imposant les temps de parcours relatifs aux phases de « *lift-off* » et « *set-down* », il sera possible de contrôler la vitesse du manipulateur pour ces opérations délicates.

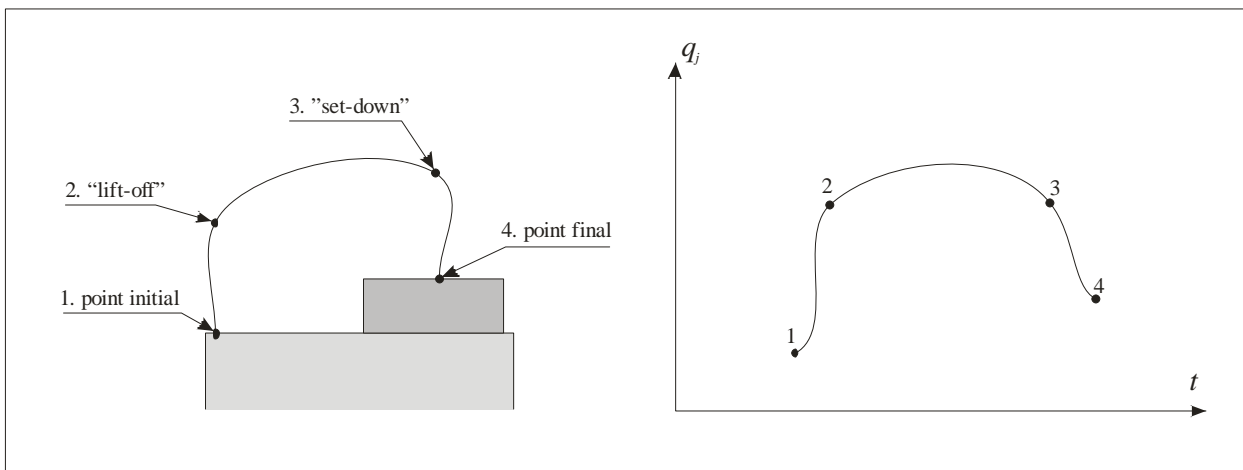


Figure 3. 6 : Description des différentes phases pour une tâche « *pick & place* ».

Afin d'avoir un mouvement doux lors des phases, « *lift-off* » et « *set-down* », on exige que les vitesses et parfois les accélérations soient continues et lisses. Plusieurs méthodes se basant sur l'interpolation ou les « *splines cubiques* » sont utilisées [1], [41].

3.6.3. Planification de la trajectoire dans l'espace opérationnel.

Le problème sera toujours formulé de la même manière, sauf que dans l'espace opérationnel, les deux positions : « *initiale* » et « *finale* » sont dans ce cas décrites dans l'espace opérationnel.

Deux approches sont utilisées : « *interpolation cartésienne* » et « *interpolation articulaire* »

3.6.3.1. Interpolation cartésienne.

Réalisée dans l'espace naturel de l'opérateur, l'interpolation cartésienne rend plus aisée la programmation de trajectoires sans collisions. Deux difficultés apparaissent néanmoins :

1. Si la notion d'interpolation (en ligne droite, en arc de cercle, etc.) est évidente en ce qui concerne la position de l'outil, il n'en est pas de même pour son orientation ;
2. Toute trajectoire comportant un tracé anguleux de l'outil nécessiterait des accélérations infinies : il sera nécessaire de prévoir des "raccords" pour le tracé de carrés, rectangles, etc.

Plusieurs méthodes ont été proposées dans ce sujet, on cite : La méthode de Paul, et la méthode de Taylor [41].

3.6.3.2. Interpolation articulaire.

L'interpolation cartésienne coûte cher en calcul si chaque point interpolé doit être transposé $Q = f^{-1}(X)$ dans l'espace articulaire. En outre, il est difficile de veiller aux respects des contraintes de limitations k_v et k_a . D'où l'idée de ne calculer dans l'espace cartésien qu'un point sur n (entier déterminé en fonction de la précision voulue), de les transposer dans l'espace articulaire et ensuite on procède à une interpolation plus fine dans cet espace.

Plusieurs méthodes sont utilisées dans cette approche, on cite : La méthode de Lin et la méthode de Taylor [41].

3.7. Théorie des graphes et chemins optimaux.

Les graphes représentent un outil puissant pour modéliser de nombreux problèmes combinatoires qui seraient sans cela difficilement abordables par des techniques classiques comme l'analyse mathématique.

En plus de son existence en tant qu'objet mathématique, le graphe est aussi une structure de données puissante pour l'informatique.

Les graphes sont irremplaçables dès qu'il s'agit de décrire la structure d'un ensemble complexe, en exprimant les relations, les dépendances entre ses éléments. Des exemples sont les diagrammes hiérarchiques, les arbres génétiques, et les diagrammes de succession de tâches en gestion de projets.

Il existe deux familles des graphes : orientés et non orientés selon que les relations entre les éléments d'un ensemble soient orientés ou non.

Un graphe est défini par un couple $G = (X, U)$ de deux ensembles :

- X est un ensemble (x_1, x_2, \dots, x_N) de sommets, également appelés noeuds ;
- $U = (u_1, u_2, \dots, u_M)$ est une famille de couples ordonnés de sommets appelés arêtes (arcs).

L'approche générale consiste à modéliser l'espace des solutions comme un graphe valué et muni de poids ou coûts sur ses arêtes, et par la suite, l'utilisation d'une heuristique pour le parcourir [44].

Etant donné :

- Un nœud initial ;
- Une fonction $goal(n)$ qui retourne « *Vrai* » si le nœud n est un nœud objectif (but) ;
- Une fonction $Transition(n)$ qui retourne les transitions immédiates du nœud n ;

L'algorithme retourne un chemin partant d'un nœud initial à un nœud objectif :

- Le coût d'un chemin est la somme des coûts des arcs qui le composent ;
- Il peut y avoir plusieurs nœuds satisfaisant l'objectif ;
- Il peut aussi y avoir plusieurs chemins menant à un nœud.

L'enjeu de l'algorithme se résume dans les points suivants :

- » Trouver un chemin solution ;
- » Dans le cas où il existe plusieurs chemins possibles, choisir un chemin optimal ;
- » Le trouver vite ;

3.8. Conclusion.

Dans ce chapitre on a présenté d'une manière générale le sujet des trajectoires en robotique. Après avoir donné les différents aspects du problème de planification, et avoir classifié les types de méthodes utilisées, on a abordé le problème de suivi de trajectoires dans l'espace opérationnel, et la génération de mouvements dans l'espace articulaire. Et enfin, on a présenté brièvement l'utilité de la théorie des graphes dans la recherche de la trajectoire optimale entre deux situations extrêmes.

Le passage de la trajectoire de l'espace opérationnel vers l'espace articulaire est lié directement avec le modèle géométrique inverse qui n'a pas été traité d'une manière générale dans le chapitre précédent. Dans le chapitre suivant, nous allons traiter en détails le modèle géométrique inverse ainsi que la génération de mouvement pour deux robots plans coopérants à la manipulation d'un objet.

CHAPITRE 4

APPLICATIONS.

4.1. Introduction.

Partant des fondements théoriques abordés dans les chapitres précédents, qui consistent à la modélisation en robotique avec tous ces axes d'études : la modélisation des robots qui comporte les modèles géométrique, cinématique et dynamique pour décrire le comportement du robot dans une évolution connue, ou inversement, trouver l'évolution du robot connaissant son comportement dynamique ou cinématique. Pour aborder les tâches de manipulation et manutention, la modélisation s'étend aussi à l'objet afin de décrire son comportement dynamique selon le mouvement désiré et son influence sur le robot intervenant. Et finalement, la modélisation du mouvement du robot et de l'objet à manipuler. L'objectif du présent chapitre est d'appliquer cette théorie pour des exemples réels, où la structure du robot est connue, et résoudre les modèles décrits pour différents exemples de robots.

4.2. Notations.

- $R_W = (O_W, x_W, y_W, z_W)$: Repère atelier – référentiel considéré absolu ;
- C_j^k : Lien n° j du robot k ;
- $R_j^k = (O_j^k, x_j^k, y_j^k, z_j^k)$: Repère lié au lien C_j^k ;
- $R_{PS}^k = (O_{PS}^k, x_{PS}^k, y_{PS}^k, z_{PS}^k)$: Repère lié au point de saisie de l'organe terminal du robot k ;
- $R_{Objet} = (O_{Objet}, x_{Objet}, y_{Objet}, z_{Objet})$: Repère lié à l'objet considéré.
- $R_{Pr}^k = (O_{Pr}^k, x_{Pr}^k, y_{Pr}^k, z_{Pr}^k)$: Repère lié à la prise k de l'objet considéré.

- ${}^i A_j^k = \begin{bmatrix} (i s_j^k)_x & (i n_j^k)_x & (i a_j^k)_x \\ (i s_j^k)_y & (i n_j^k)_y & (i a_j^k)_y \\ (i s_j^k)_z & (i n_j^k)_z & (i a_j^k)_z \end{bmatrix}$: Matrice définissant l'orientation du lien C_j^k par rapport au repère du lien C_i^k ;
- ${}^i A_{PS}^k$: Matrice définissant l'orientation du point de saisie du robot k par rapport au repère du lien C_i^k ;
- ${}^i r_j^k = \left[(i r_j^k)_x, (i r_j^k)_y, (i r_j^k)_z \right]^T$: Vecteur définissant la position du lien C_j^k par rapport au repère du lien C_i^k ;
- ${}^i r_{PS}^k$: Vecteur définissant la position du point de saisie du robot k par rapport au repère du lien C_i^k ;
- ${}^i T_j^k = \begin{bmatrix} {}^i A_j^k & {}^i r_j^k \\ 0 & 0 & 0 & 1 \end{bmatrix}$: Matrice de transformation homogène qui décrit le positionnement (position & orientation) du lien C_j^k par rapport au repère du lien C_i^k ;
- ${}^i T_{PS}^k$: Matrice de transformation homogène qui décrit le positionnement du point de saisie du robot k par rapport au repère du lien C_i^k ;
- \vec{g} : Accélération de la pesanteur ;
- $\sigma_j^k, \bar{\sigma}_j^k$ paramètres dépendant de l'articulation du lien C_j^k définis comme suit :
 - $\sigma_j^k = 0$: si l'articulation est rotoïde ;
 - $\sigma_j^k = 1$: si l'articulation est prismatique ;
 - $\bar{\sigma}_j^k = 1 - \sigma_j^k$;
- $\alpha_j^k, d_j^k, \theta_j^k, r_j^k$: Paramètres de Denavit-Hartenberg du lien C_j^k ;
- q_j^k : Variable articulaire du lien C_j^k ;
- $\dot{q}_j^k, \ddot{q}_j^k$: Vitesse et accélération articulaires du lien C_j^k ;
- T_{Objet} : Tenseur d'inertie de l'objet considéré par rapport au repère lié à l'objet.

4.3. Descriptions des structures mécaniques étudiées.

4.3.1. Considérations générales.

Dans tous les exemples traités dans le présent chapitre, on considère ce qui suit :

- Les robots étudiés évoluent dans un environnement défini par un référentiel global supposé absolu, appelé « *repère atelier* », noté $R_W = (O_W, x_W, y_W, z_W)$. Ce repère doit avoir l'axe z vertical orienté vers le haut comme montré sur la figure suivante :

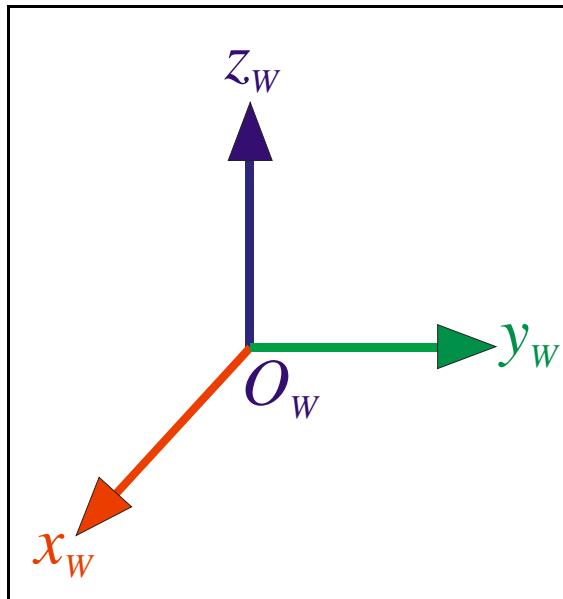


Figure 4. 1 : Description du repère atelier.

- La situation du robot dans le repère atelier est définie (comme montré au chapitre2, § 2.7) par la situation de la base du robot par :

$${}^wT_0 = \begin{bmatrix} {}^w s_0 & {}^w n_0 & {}^w a_0 & {}^w r_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^w A_0 & {}^w r_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{--- (4. 1)}$$

- On considère que le robot est fixe (non mobile), donc les paramètres de l'équation (4.1) sont donnés et restent fixes ;
- Le vecteur accélération de la pesanteur est défini selon le référentiel global R_W par :

$$\vec{g} = (0,0,-9.81)^T ;$$

- Tous les liens des robots sont considérés comme des barres cylindriques pleines en acier (masse volumique $\rho = 7850 \text{ kg/m}^3$) avec $\phi_{\text{barre}} = 5\%$ de la longueur de la barre ;
- La masse d'une barre de longueur l est calculée de la manière suivante :

$$m = \rho.l.\frac{\pi.D^2}{4} = \frac{25.\rho.\pi.l^3.10^{-4}}{4} \quad \text{--- (4. 2)}$$

- Les moments d'inertie d'une barre de masse m et de longueur $l = 2a$ sont calculés de la manière suivante :

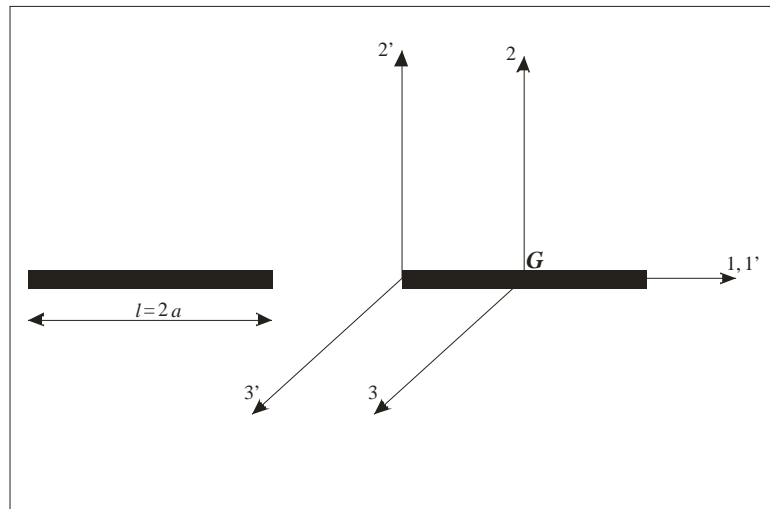


Figure 4. 2 : Moments d'inerties pour une barre.

$$I_{[1,2,3]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{ma^2}{3} & 0 \\ 0 & 0 & \frac{ma^2}{3} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{ml^2}{12} & 0 \\ 0 & 0 & \frac{ml^2}{12} \end{bmatrix} \quad \text{----(4. 3)}$$

$$I_{[1',2',3']} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{4}{3}.ma^2 & 0 \\ 0 & 0 & \frac{4}{3}.ma^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{ml^2}{3} & 0 \\ 0 & 0 & \frac{ml^2}{3} \end{bmatrix} \quad \text{----(4. 4)}$$

- Les masses, inerties des actionneurs (articulations) sont supposées négligeables.

4.3.2. Organes terminaux utilisés.

Pour des raisons de simplification, dans tous les exemples traités, l'organe terminal a été considéré comme une barre de 20 cm de longueur. On considère que le point de saisie est situé au bout libre de la barre et orienté parallèlement au repère lié au lien, comme montré dans la figure suivante.

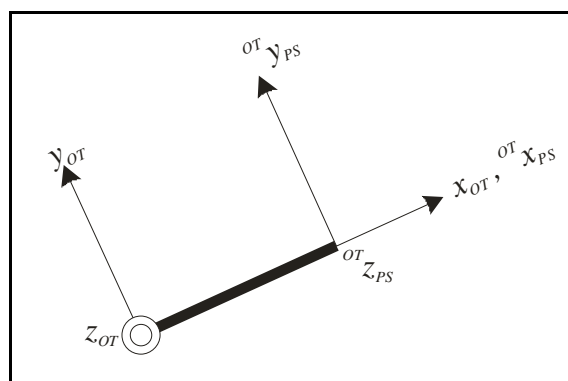


Figure 4. 3 : Positionnement du point de saisie de l'organe terminal.

La matrice définissant l'orientation du point de saisie de l'organe terminal par rapport au repère de l'organe terminal est donnée par :

$${}^{OT}A_{PS} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{---- (4.5)}$$

4.3.3. Mécanisme plan à 2 ddl.

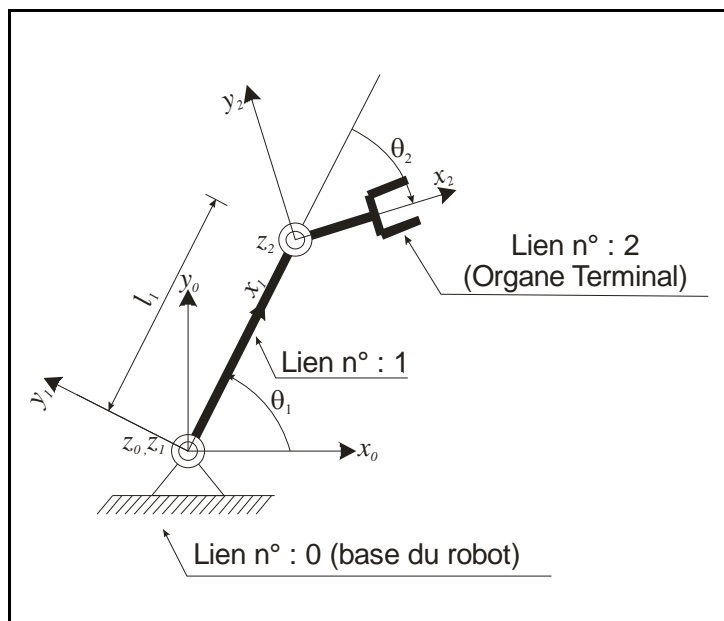


Figure 4. 4 : Description d'un mécanisme plan à deux degrés de liberté.

Paramètres des liens.

Tableau 4. 1 : Paramètres des liens du robot plan à 4 ddl.

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
1	$l_1 = 0.5$	1.9267	(0.25,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.1606 & 0 \\ 0 & 0 & 0.1606 \end{bmatrix}$
2	$l_2 = 0.2$	0.1233	(0.1,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.0016 & 0 \\ 0 & 0 & 0.0016 \end{bmatrix}$

Paramètres Denavit-Hartenberg (DH).

Tableau 4. 2 : Paramètres de Denavit-Hartenberg d'un robot plan à 2 ddl.

N° du lien	σ_j	α_j (Rad)	d_j (m)	θ_j (Rad)	r_j (m)	$(q_j)_{\min}$	$(q_j)_{\max}$
1	0	0	0	θ_1	0	0	π
2	0	0	$l_1 = 0.5$	θ_2	0	$-\pi/2$	$3\pi/2$

Point de saisie de l'organe terminal.

Le positionnement du point de saisie de l'organe terminal par rapport à son repère est donné par la matrice de transformation homogène suivante :

$${}^2T_{PS} = \begin{bmatrix} 1 & 0 & 0 & 0.2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{--- (4.6)}$$

4.3.4. Mécanisme Plan à 4 ddl.

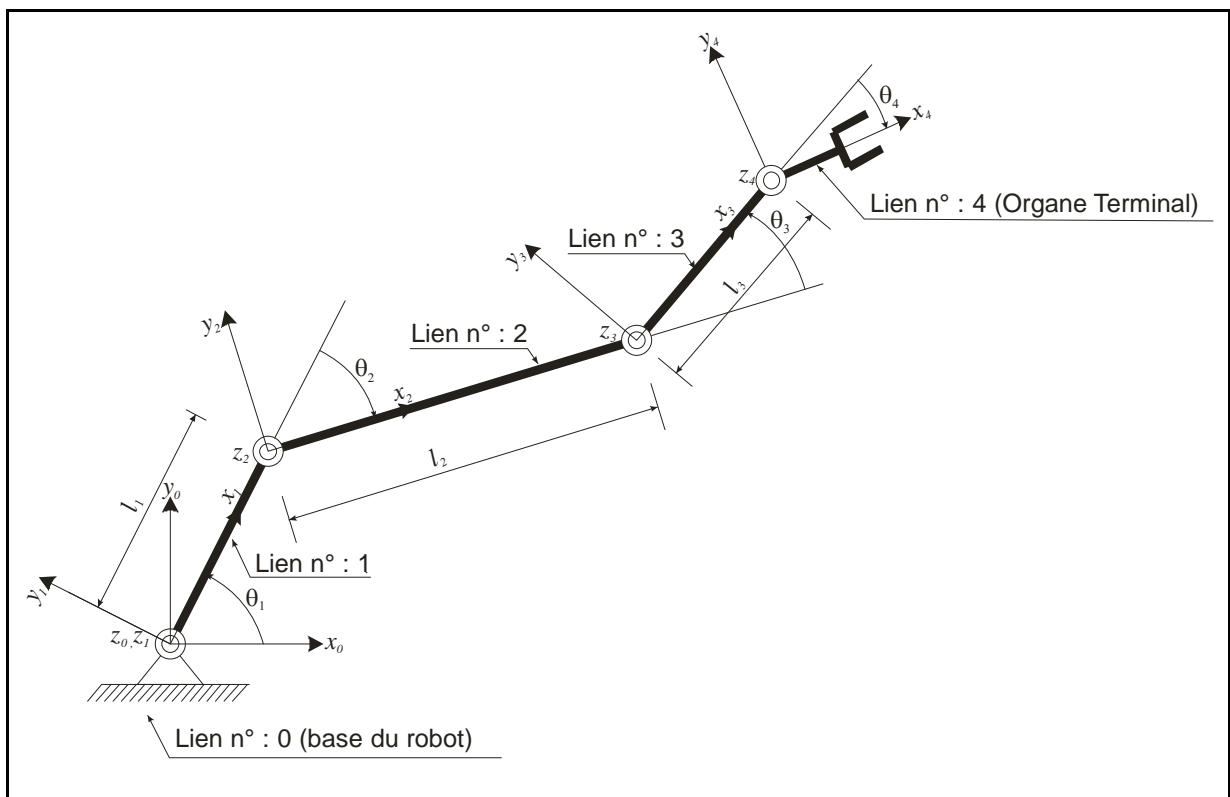


Figure 4. 5 : Description d'un mécanisme plan à quatre degrés de liberté.

Paramètres des liens.

Tableau 4. 3 : Paramètres des liens d'un robot plan à 4 ddl.

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
1	$l_1 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
2	$l_2 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
3	$l_3 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
4	$l_4 = 0.2$	0.1233	(0.1,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.0016 & 0 \\ 0 & 0 & 0.0016 \end{bmatrix}$

Paramètres Denavit-Hartenberg (DH).

Tableau 4. 4 : Paramètres de Denavit-Hartenberg d'un robot plan à 4 ddl.

N° du lien	σ_j	α_j (Rad)	d_j (m)	θ_j (Rad)	r_j (m)	$(q_j)_{\min}$	$(q_j)_{\max}$
1	0	0	0	θ_1	0	0	π
2	0	0	$l_1 = 1$	θ_2	0	$-\pi/2$	$3\pi/2$
3	0	0	$l_2 = 1$	θ_3	0	$-\pi/2$	$3\pi/2$
4	0	0	$l_3 = 1$	θ_4	0	$-\pi/2$	$3\pi/2$

Point de saisie de l'organe terminal.

Le positionnement du point de saisie de l'organe terminal par rapport à son repère est donné par la matrice de transformation homogène suivante :

$${}^4T_{PS} = \begin{bmatrix} 1 & 0 & 0 & 0.2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----(4. 7)}$$

4.3.5. Structure plane à base articulée.

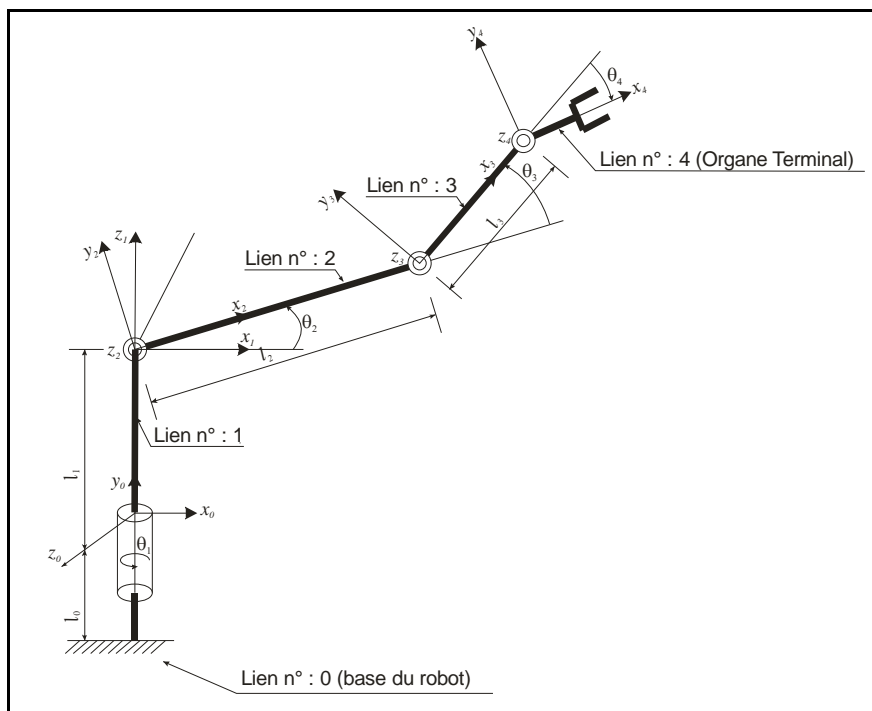


Figure 4. 6 : Description d'une structure plane à base articulée.

Paramètres des liens.

On considère que la base du robot possède une hauteur initiale l_0 (voir figure 4.6) ;

Tableau 4. 5 : Paramètres des liens d'une structure plane à base articulée.

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
1	$l_1 = 1$	15.4134	(0,0,-0.5)	$\begin{bmatrix} 5.1378 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
2	$l_2 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
3	$l_3 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
4	$l_4 = 0.2$	0.1233	(0.1,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.0016 & 0 \\ 0 & 0 & 0.0016 \end{bmatrix}$

Paramètres Denavit-Hartenberg (DH).

Tableau 4. 6 : Paramètres de Denavit-Hartenberg d'une structure plane à base articulée.

N° du lien	σ_j	α_j (Rad)	d_j (m)	θ_j (Rad)	r_j (m)	$(q_j)_{\min}$	$(q_j)_{\max}$
1	0	$-\pi/2$	0	θ_1	$l_1 = 1$	$-\pi$	π
2	0	$\pi/2$	0	θ_2	0	$-\pi/3$	π
3	0	0	$l_2 = 1$	θ_3	0	$-\pi/2$	$\pi/2$
4	0	0	$l_3 = 1$	θ_4	0	$-\pi/2$	$\pi/2$

Point de saisie de l'organe terminal.

Le positionnement du point de saisie de l'organe terminal par rapport à son repère est donné par la matrice de transformation homogène suivante :

$${}^4T_{PS} = \begin{bmatrix} 1 & 0 & 0 & 0.2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{--- (4. 8)}$$

4.3.6. Robot STAUBLI RX 90.

La figure (4.7) montre la structure du robot STAUBLI RX 90. Le schéma de gauche donne la structure générale du robot [1], et le schéma de droite montre la structure simplifiée qu'on a utilisé.

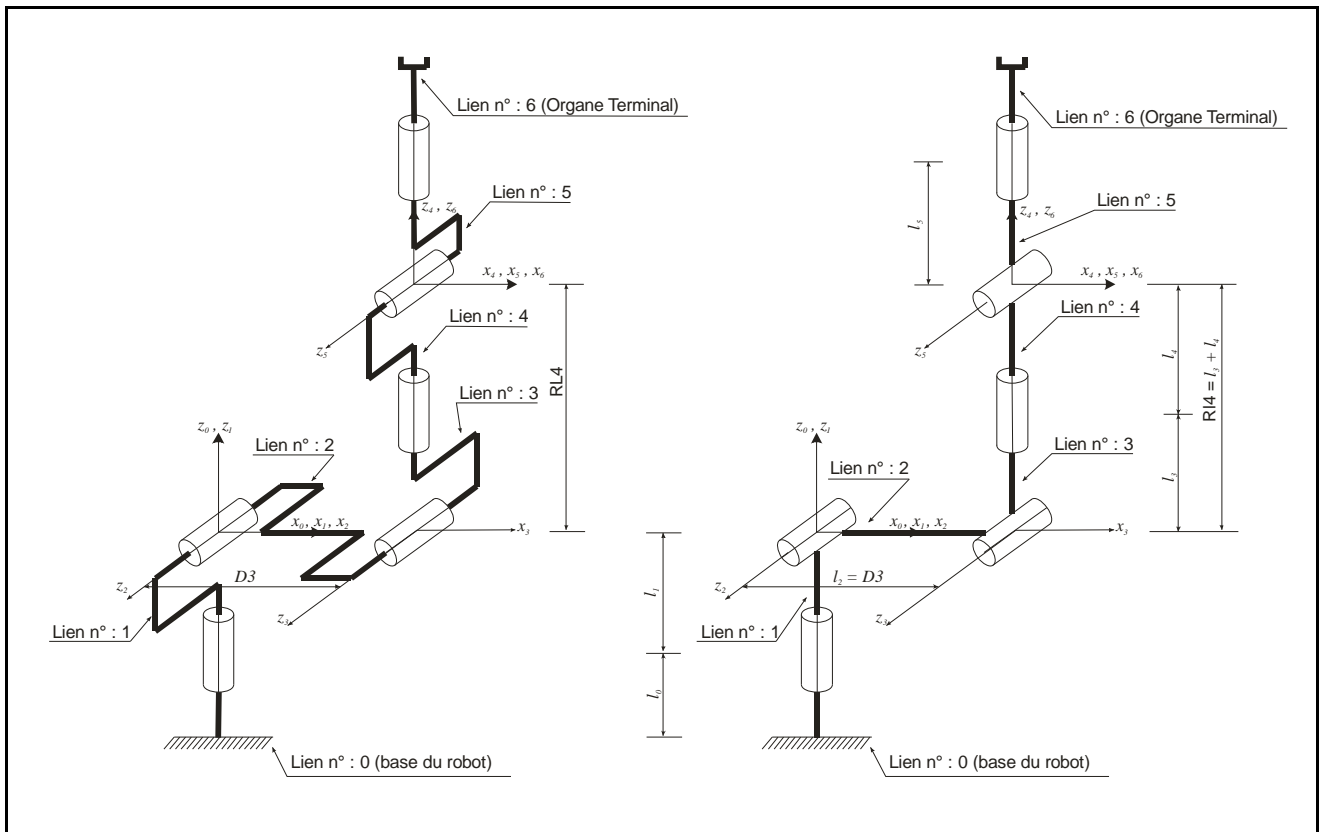


Figure 4. 7 : Structure du robot STAUBLI RX 90.

Paramètres des liens.

On considère que la base du robot possède une hauteur initiale $l_0 + l_1$ (voir figure 4.7) ;

Tableau 4. 7 : Paramètres des liens d'une structure STAUBLI RX 90.

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
1	$l_1 = 1$	15.4134	(0,0,-0.5)	$\begin{bmatrix} 5.1378 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
2	$l_2 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
3	$l_3 = 1$	15.4134	(0,0.5,0)	$\begin{bmatrix} 5.1378 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
4	$l_4 = 1$	15.4134	(0,0,-0.5)	$\begin{bmatrix} 5.1378 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

Tableau 4.7 (suite)

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
5	$l_5 = 1$	15.4134	(0,0.5,0)	$\begin{bmatrix} 5.1378 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
6	$l_6 = 0.2$	0.1233	(0,0,1.1)	$\begin{bmatrix} 0.1249 & 0 & 0 \\ 0 & 0.1249 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

Remarque : Le tenseur d'inertie du lien 6 a été calculé en utilisant le théorème de HUYGENS.

Paramètres Denavit-Hartenberg (DH).

Tableau 4. 8 : Paramètres de Denavit-Hartenberg d'une structure STAUBLI RX 90.

N° du lien	σ_j	α_j (Rad)	d_j (m)	θ_j (Rad)	r_j (m)	$(q_j)_{\min}$	$(q_j)_{\max}$
1	0	0	0	θ_1	0	$-\pi$	π
2	0	$\pi/2$	0	θ_2	0	$-\pi/3$	π
3	0	0	$D3 = l_2 = 1$	θ_3	0	$-\pi/2$	$\pi/2$
4	0	$-\pi/2$	0	θ_4	$RL4 = l_3 + l_4 = 2$	$-\pi/2$	$\pi/2$
5	0	$\pi/2$	0	θ_5	0	$-\pi/2$	$\pi/2$
6	0	$-\pi/2$	0	θ_6	0	$-\pi$	π

Point de saisie de l'organe terminal.

Le positionnement du point de saisie de l'organe terminal par rapport à son repère est donné par la matrice de transformation homogène suivante :

$${}^4T_{PS} = \begin{bmatrix} 1 & 0 & 0 & 1.2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----(4. 9)}$$

4.3.7. Robot SCARA.

La figure (4.8) montre la structure du robot SCARA. Le schéma supérieur donne la structure générale du robot [1], et le schéma inférieur montre la structure simplifiée qu'on a utilisé.

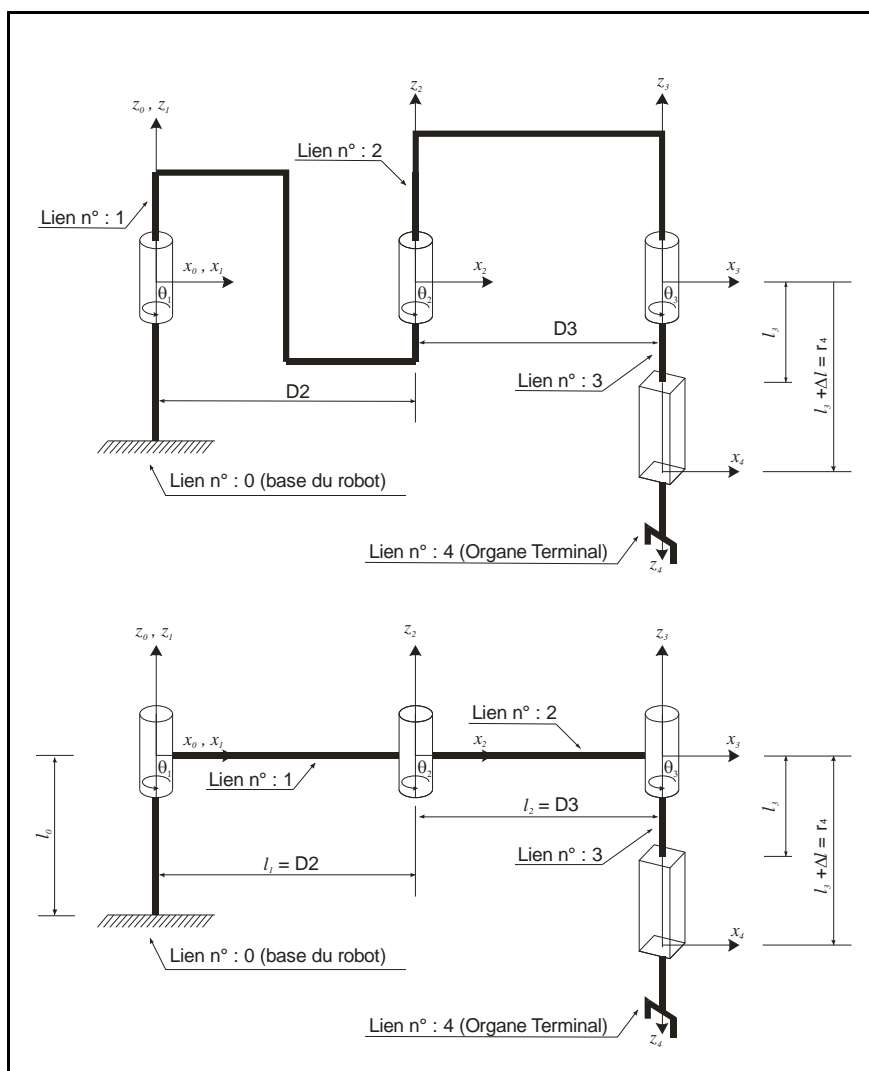


Figure 4. 8 : Structure d'un robot du type SCARA.

Paramètres des liens.

On considère que la base du robot possède une hauteur initiale l_0 (voir figure 4.8) ;

Tableau 4. 9 : Paramètres des liens d'une structure SCARA.

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
1	$l_1 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$

Tableau 4.9 (suite)

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
2	$l_2 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
3	$l_3 = 0.5$	1.9267	(0,0,-0.25)	$\begin{bmatrix} 0.1606 & 0 & 0 \\ 0 & 0.1606 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
4	$l_4 = 0.2$	0.1233	(0,0,0.1)	$\begin{bmatrix} 0.0016 & 0 & 0 \\ 0 & 0.0016 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

Remarque : Pour le lien n°4, on considère que c'est l'articulation qui s'allonge ou se rétrécit et non pas le lien 3 ni le lien 4, et donc on n'aura aucune influence sur le moment d'inertie de ces deux liens du moment qu'on a posé l'hypothèse que les masses des articulations sont supposées nulles.

Paramètres Denavit-Hartenberg (DH).

Tableau 4. 10 : Paramètres de Denavit-Hartenberg d'une structure SCARA.

N° du lien	σ_j	α_j (Rad)	d_j (m)	θ_j (Rad)	r_j (m)	$(q_j)_{\min}$	$(q_j)_{\max}$
1	0	0	0	θ_1	0	$-\pi$	π
2	0	0	$D_2 = l_1 = 1$	θ_2	0	$-\pi/2$	$\pi/2$
3	0	0	$D_3 = l_2 = 1$	θ_3	0	$-\pi$	π
4	1	π	0	0	r_4	$l_3 = 0.5$	1

Point de saisie de l'organe terminal.

Le positionnement du point de saisie de l'organe terminal par rapport à son repère est donné par la matrice de transformation homogène suivante :

$${}^4T_{PS} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---(4. 10)}$$

4.3.8. Robot PUMA.

La figure (4.9) montre la structure du robot PUMA. Le schéma supérieur donne la structure générale du robot [1], et le schéma inférieur montre la structure simplifiée qu'on a utilisé.

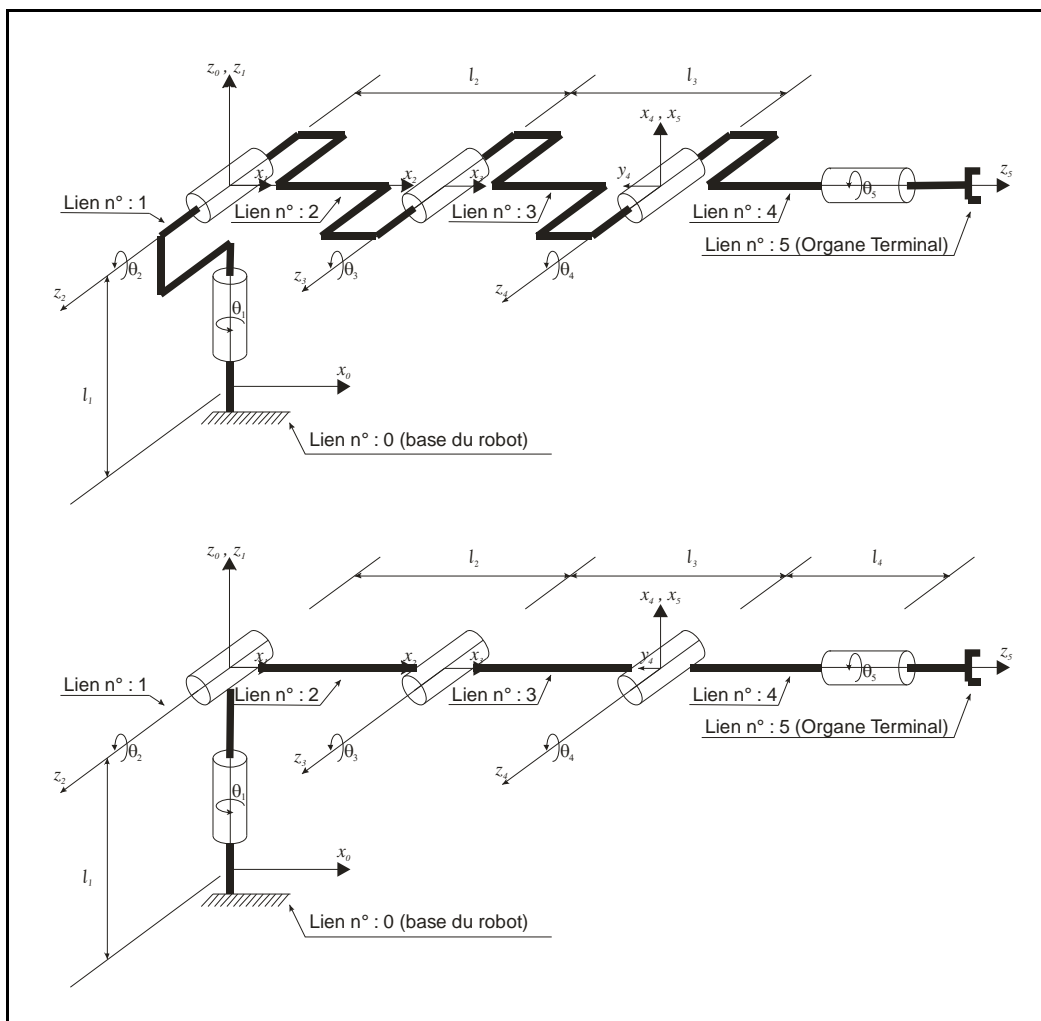


Figure 4. 9 : Structure d'un robot du type PUMA.

Paramètres des liens.

Tableau 4. 11 : Paramètres des liens d'une structure PUMA.

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
1	$l_1 = 1$	15.4134	(0,0,-0.5)	$\begin{bmatrix} 5.1378 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
2	$l_2 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
3	$l_3 = 1$	15.4134	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 5.1378 & 0 \\ 0 & 0 & 5.1378 \end{bmatrix}$
4	$l_4 = 0.5$	1.9267	(0,-0.25,0)	$\begin{bmatrix} 0.1606 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0.1606 \end{bmatrix}$
5	$l_5 = 0.2$	0.1233	(0,0,0.6)	$\begin{bmatrix} 0.0325 & 0 & 0 \\ 0 & 0.0325 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

Remarque : Le tenseur d'inertie du lien 5 a été calculé en utilisant le théorème de HUYGENS.

Paramètres Denavit-Hartenberg (DH).

Tableau 4. 12 : Paramètres de Denavit-Hartenberg d'une structure PUMA.

N° du lien	σ_j	α_j (Rad)	d_j (m)	θ_j (Rad)	r_j (m)	$(q_j)_{\min}$	$(q_j)_{\max}$
1	0	0	0	θ_1	$l_1 = 1$	$-\pi$	π
2	0	$\pi/2$	0	θ_2	0	$-\pi/3$	$\pi/2$
3	0	0	$l_2 = 1$	θ_3	0	$-\pi/2$	$\pi/2$
4	0	0	$l_3 = 1$	θ_4	0	$-\pi/2$	$\pi/2$
5	0	$\pi/2$	0	θ_5	0	$-\pi$	π

Point de saisie de l'organe terminal.

Le positionnement du point de saisie de l'organe terminal par rapport à son repère est donné par la matrice de transformation homogène suivante :

$${}^4T_{PS} = \begin{bmatrix} 1 & 0 & 0 & 0.7 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{--- (4.11)}$$

4.4. Application du modèle géométrique direct MGD pour analyser la trajectoire de l'organe terminal.

4.4.1. Paramètres cinématiques des actionneurs.

Pour l'ensemble des exemples qu'on a traité dans ce sous-chapitre, on considère ce qui suit :

- Les actionneurs rotatifs correspondant aux articulations rotoïdes, sont considérés comme des moteurs asynchrones dont chacun est pilotés par un variateur électronique nous permettant d'avoir le profil de vitesse voulu ;
- Les actionneurs linéaires correspondant aux articulations prismatiques, sont considérés comme des systèmes mécaniques « *vis-mère* » pour la transformation du mouvement provenant d'un moteur asynchrone piloté de la même manière que celui utilisé pour les articulations rotoïdes.

4.4.2. Application du modèle géométrique direct pour un robot plan à 2 ddl.

On considère le robot décrit en (§ 4.3.3).

Le robot est positionné horizontalement dans le repère atelier à une distance de (2m,2m,0) par rapport au repère atelier.

La matrice de transformation homogène qui décrit le robot dans son environnement est donnée donc par :

$${}^wT_0 = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{--- (4.12)}$$

Pour les deux articulations du robot, on exécute le mouvement décrit par les figures suivantes :

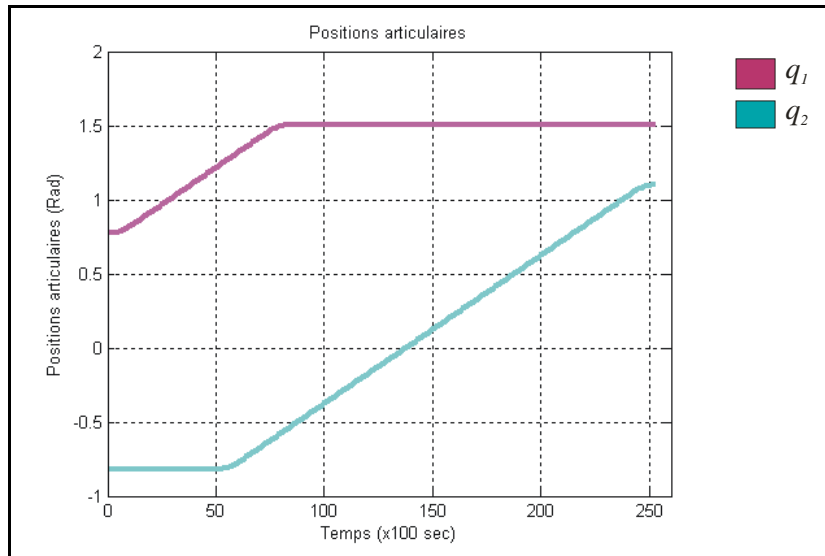


Figure 4. 10 : Positions articulaires appliquées au robot plan à 2 ddl.

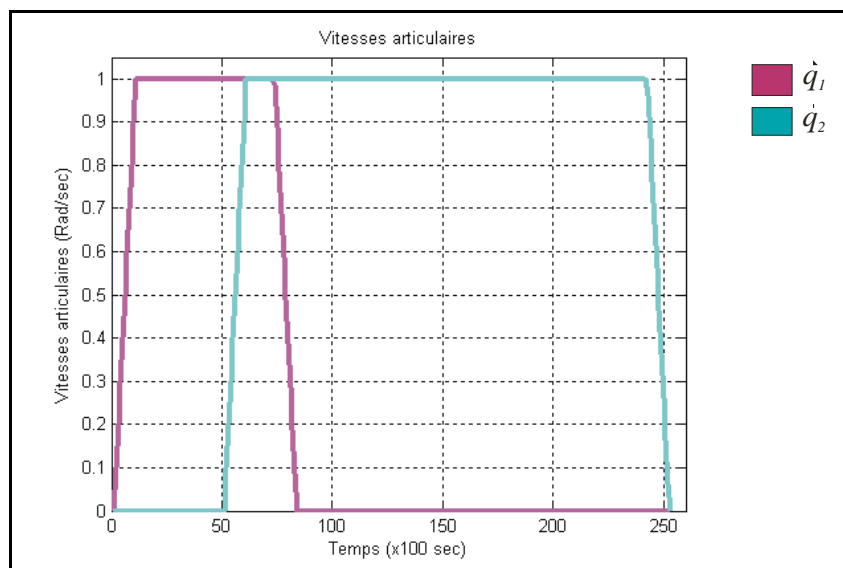


Figure 4. 11 : Vitesses articulaires appliquées au robot plan à2 ddl.

L'application du modèle géométrique direct MGD a permis d'obtenir les positions opérationnelles décrivant la trajectoire de l'organe terminal, montrées dans la figure suivante :

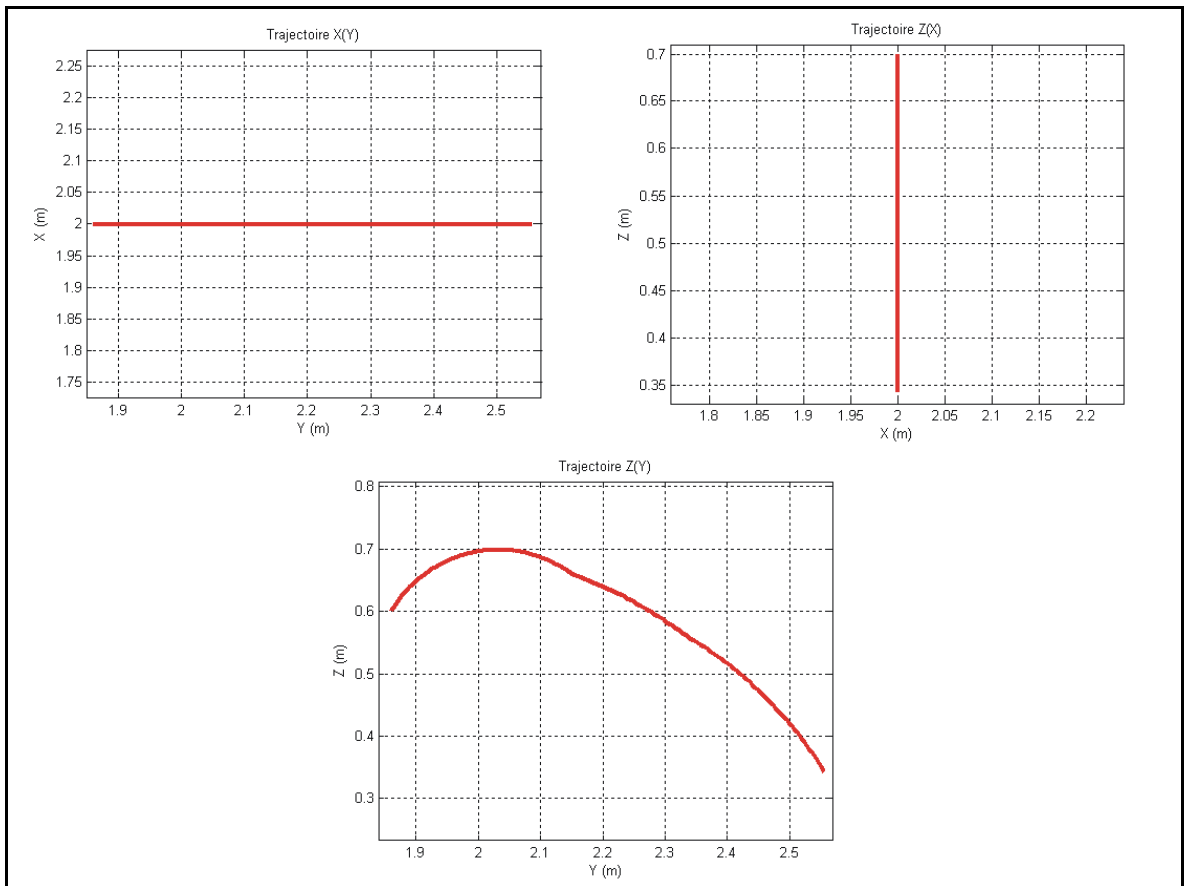


Figure 4. 12 : Trajectoire du point de saisie de l'organe terminal du robot à 2 ddl.

Une dérivation numérique des données obtenues (trajectoires), a permis d'avoir les vitesses et accélérations opérationnelles du point de saisie de l'organe terminal.

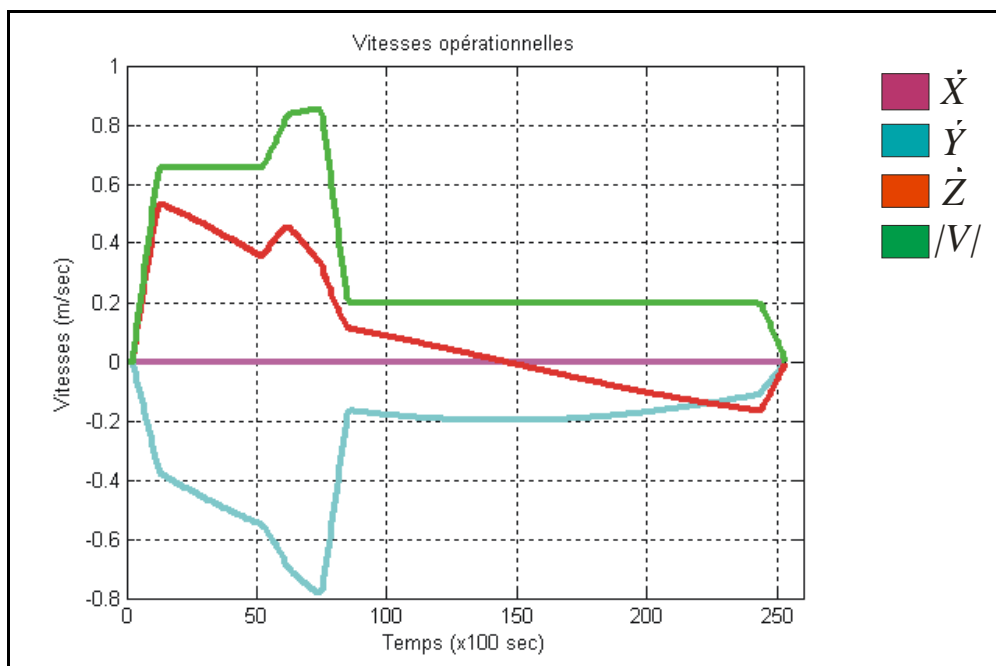


Figure 4. 13 : Vitesses opérationnelles du point de saisie de l'organe terminal du robot à 2 ddl.

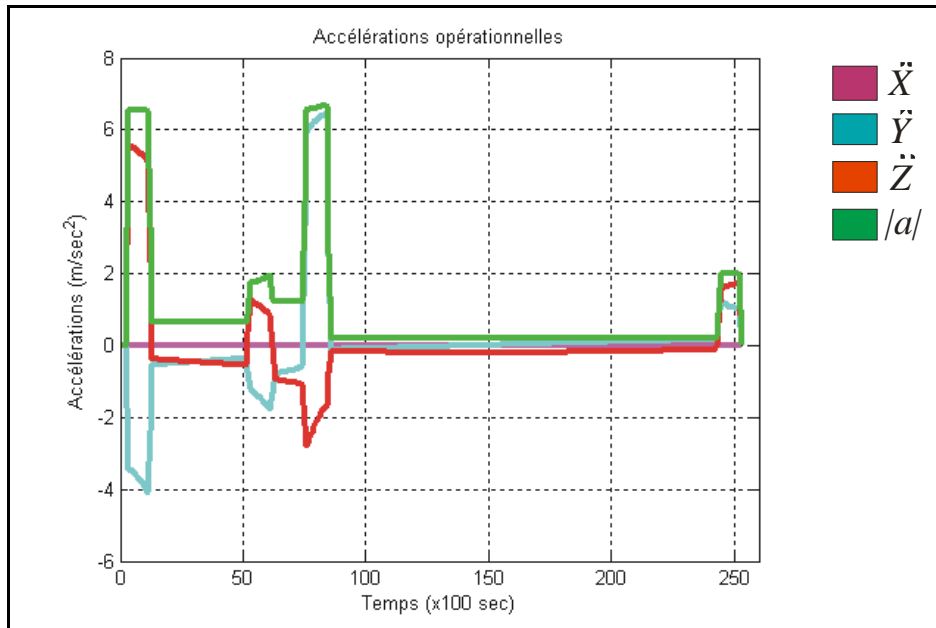


Figure 4. 14 : Accélérations opérationnelles du point de saisie de l'organe terminal du robot plans à 2 ddl.

4.4.3. Application du modèle géométrique direct pour un robot plan à 4 ddl.

On considère le robot décrit en (§ 4.3.4)

La matrice de transformation homogène qui décrit le robot dans son environnement est donnée donc par :

$${}^wT_0 = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (4. 13)}$$

On applique le mouvement décrit par les figures (4.15 et 4.16).

Une analyse similaire à celle de l'exemple précédant a donné les résultats montrés dans les figures (4.17, 4.18 et 4.19).

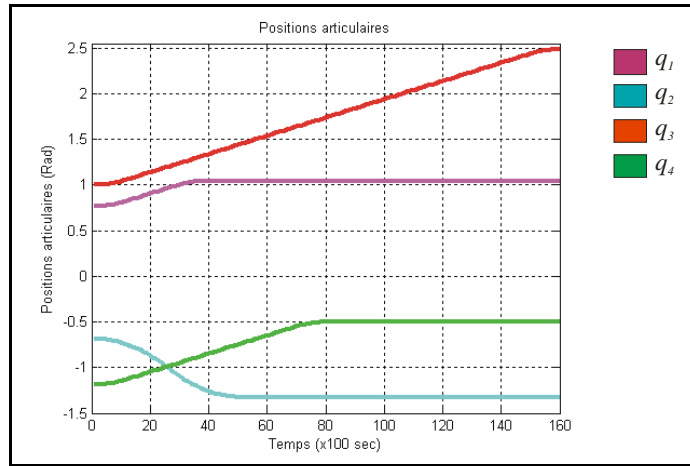


Figure 4. 15 : Positions articulaires appliquées au robot plan à 4 ddl.

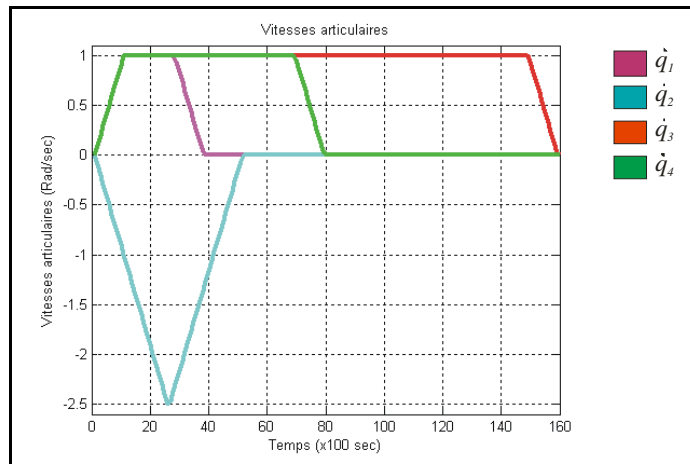


Figure 4. 16 : Vitesses articulaires appliquées au robot plan à 4 ddl.

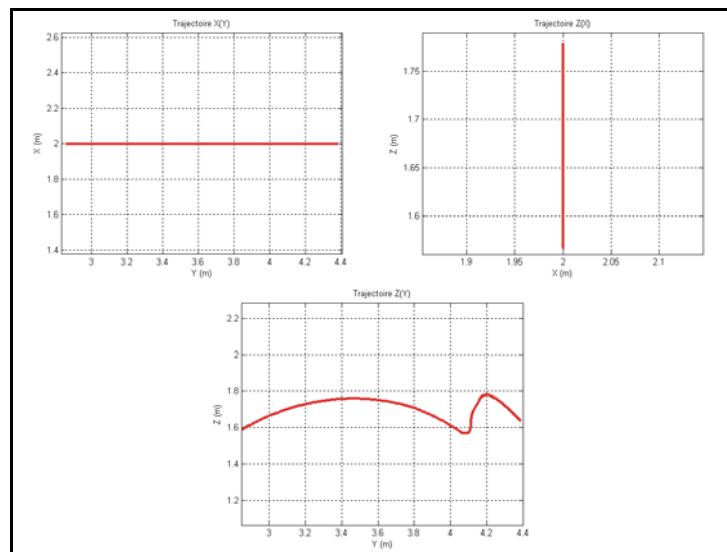


Figure 4. 17 : Trajectoire du point de saisie de l'organe terminal du robot à 4 ddl.

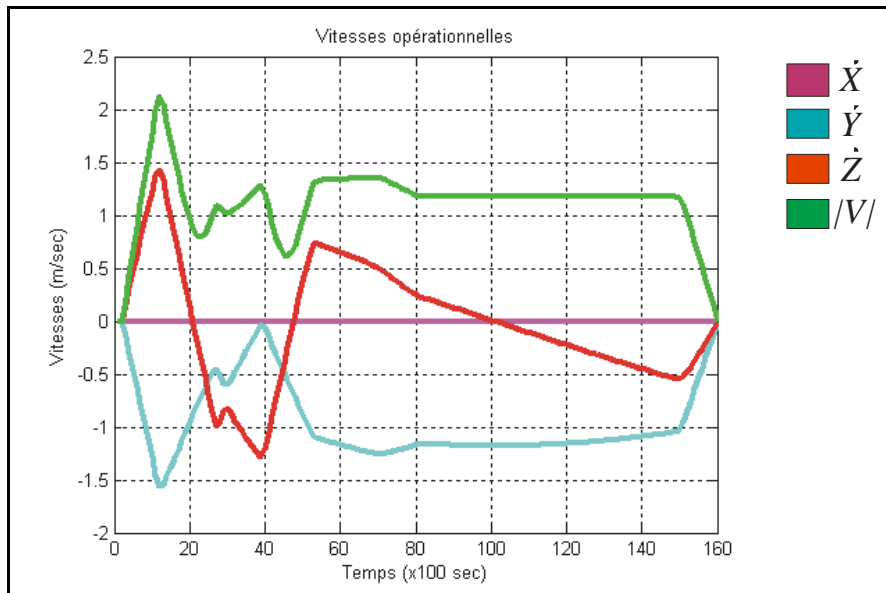


Figure 4. 18 : Vitesses opérationnelles du point de saisie de l'organe terminal du robot à 4 ddl.

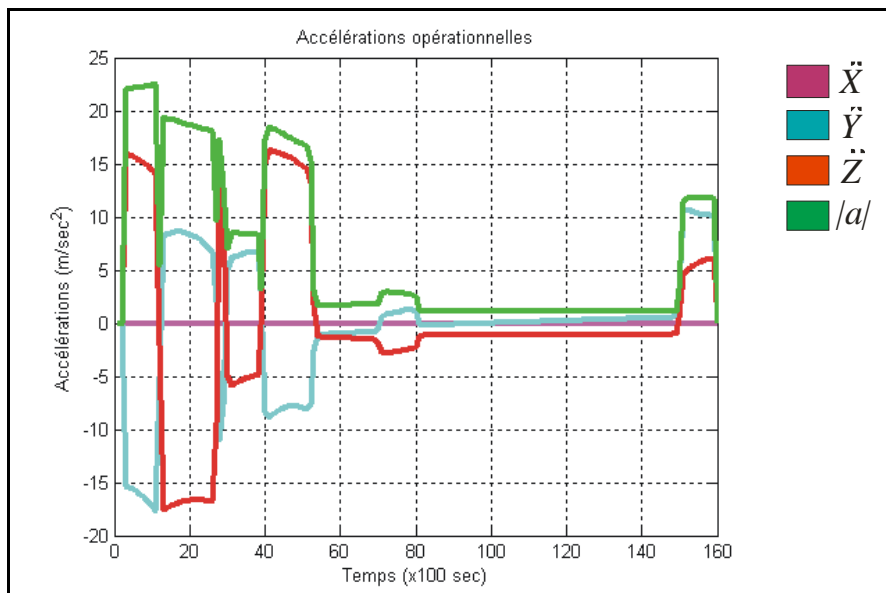


Figure 4. 19 : Accélérations opérationnelles du point de saisie de l'organe terminal du robot à 4 ddl.

4.4.4. Application du modèle géométrique directe pour une structure plane à base articulée.

On considère le robot décrit en (§ 4.3.5). On positionne le robot selon la matrice de transformation homogène suivante :

$${}^wT_0 = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----- (4. 14)}$$

On résume toute l'analyse dans les figures suivantes :

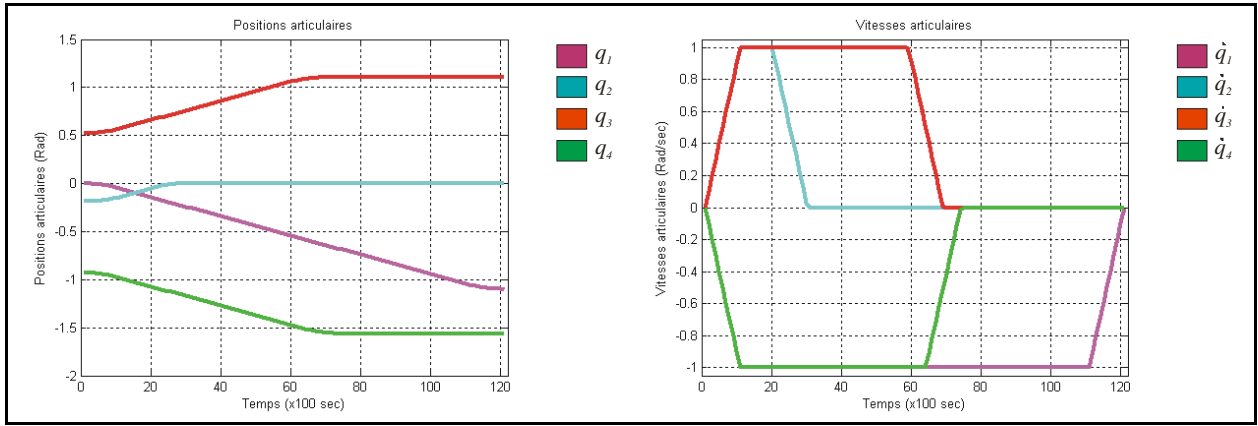


Figure 4. 20 : Paramètres cinématiques articulaires appliquées à la structure plane à base articulée.

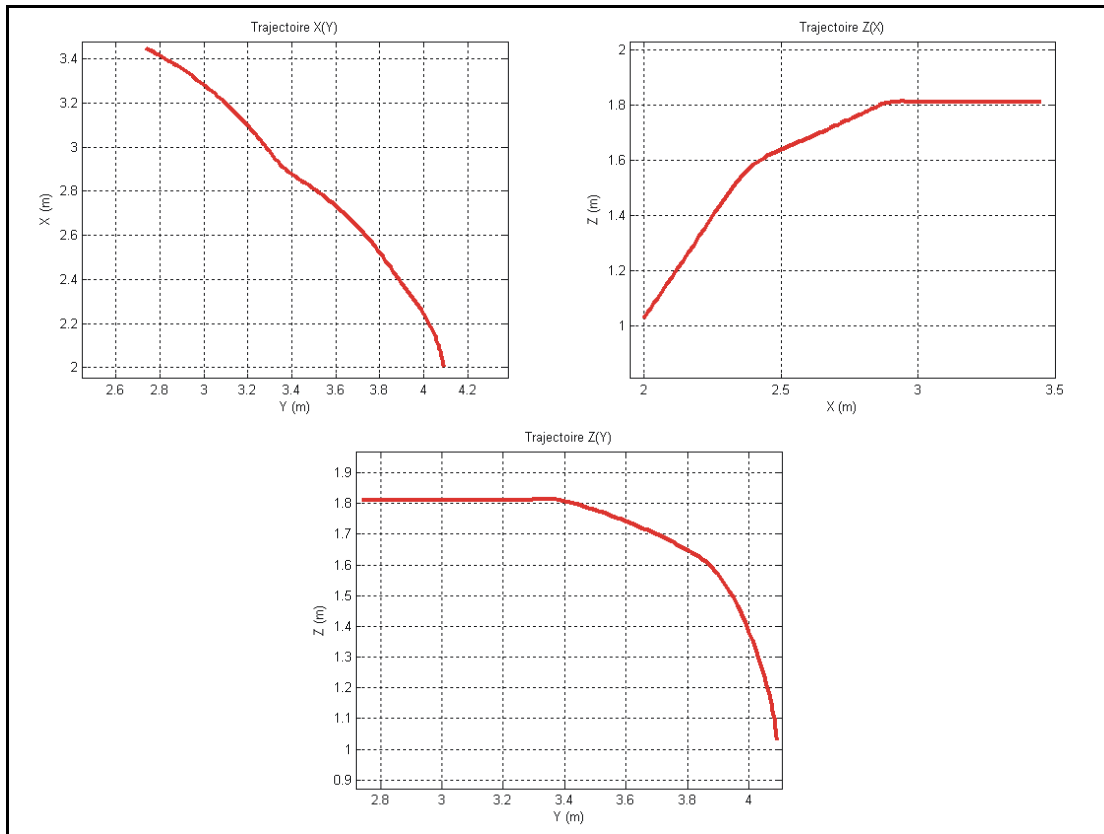


Figure 4. 21 : Trajectoire du point de saisie de la structure plane à base articulée.

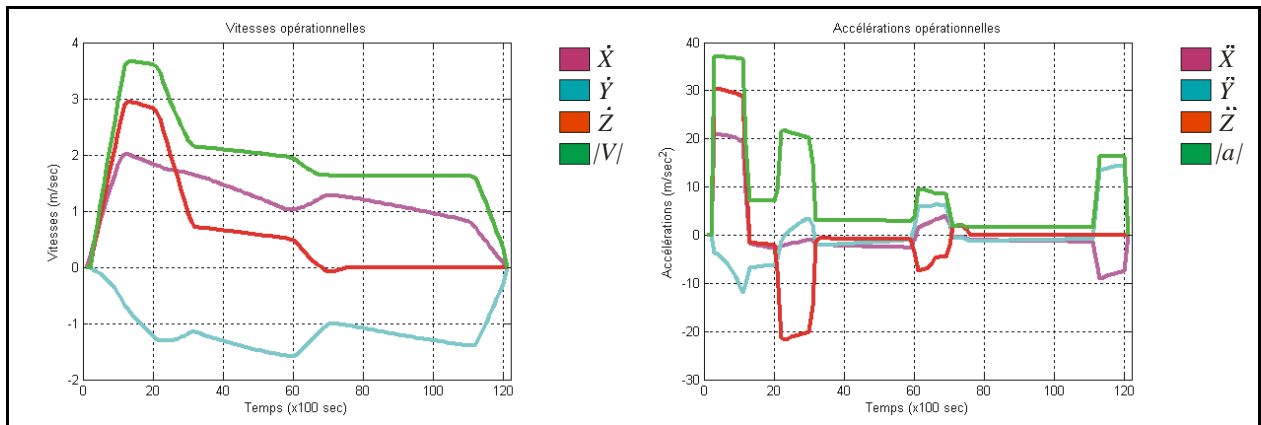


Figure 4. 22 : Vitesses et accélérations opérationnelles du point de saisi de la structure plane à base articulée.

4.4.5. Application du modèle géométrique direct pour une structure STAUBLI RX 90.

On considère le robot décrit en (§ 4.3.6). On positionne le robot selon la matrice de transformation homogène suivante :

$${}^wT_0 = \begin{bmatrix} 0 & -1 & 0 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{--- (4. 15)}$$

On résume toute l'analyse dans les figures suivantes :

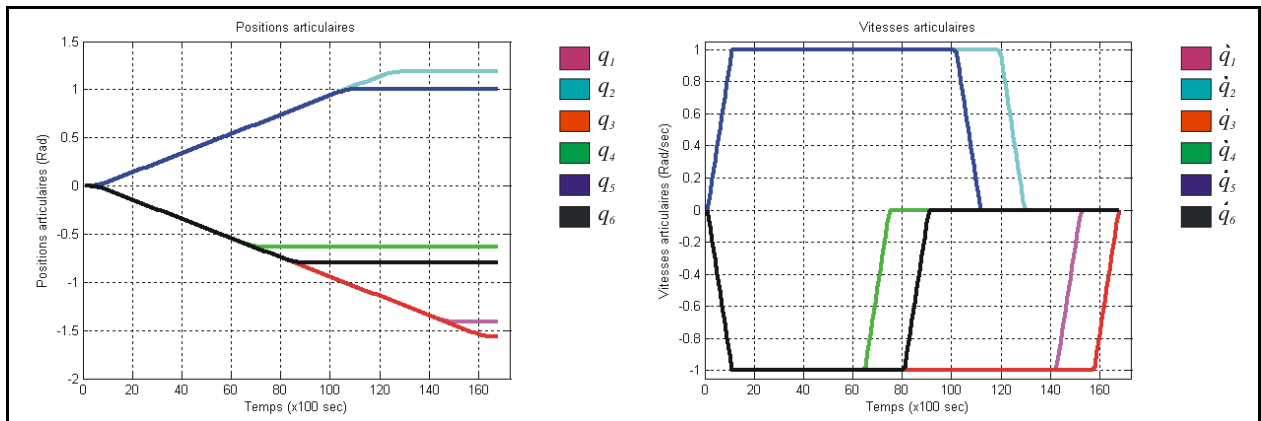


Figure 4. 23 : Paramètres cinématiques articulaires appliquées à la structure STAUBLI RX 90.

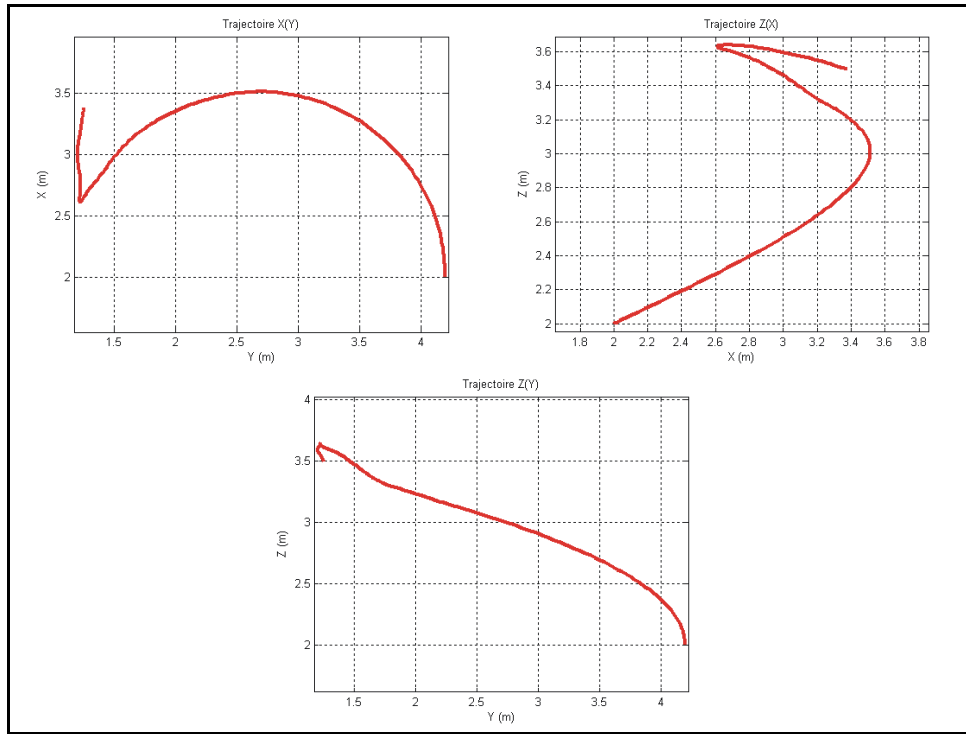


Figure 4. 24 : Trajectoire du point de saisie de la structure STAUBLI RX 90.

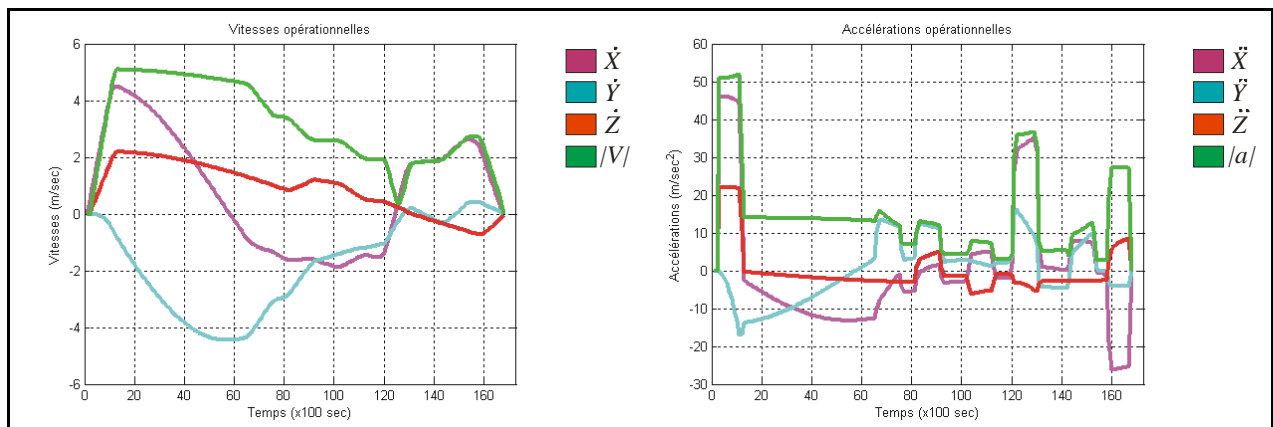


Figure 4. 25 : Vitesses et accélérations opérationnelles du point de saisie de la structure STAUBLI RX 90.

4.4.6. Application du modèle géométrique direct pour une structure SCARA.

On considère le robot décrit en (§ 4.3.7). On positionne le robot selon la matrice de transformation homogène suivante :

$${}^wT_0 = \begin{bmatrix} 0 & -1 & 0 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (4. 16)}$$

On résume toute l'analyse dans les figures suivantes :

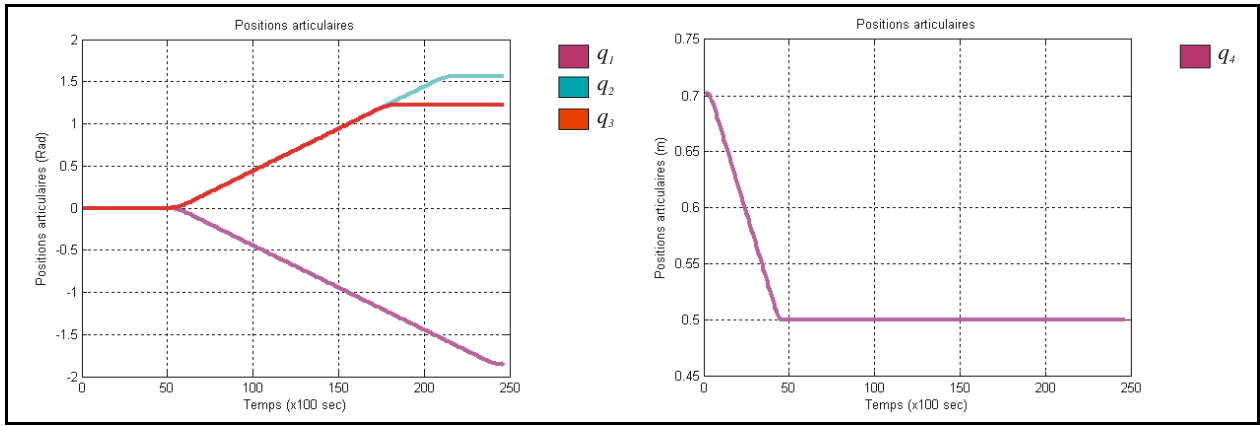


Figure 4. 26 : Positions articulaires appliquées à la structure SCARA.

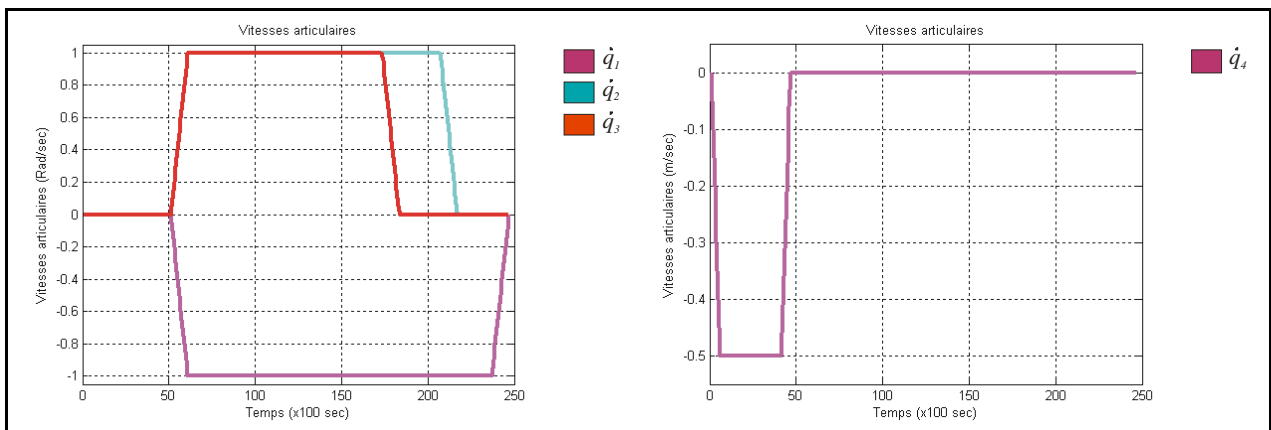


Figure 4. 27 : Vitesses articulaires appliquées à la structure SCARA.

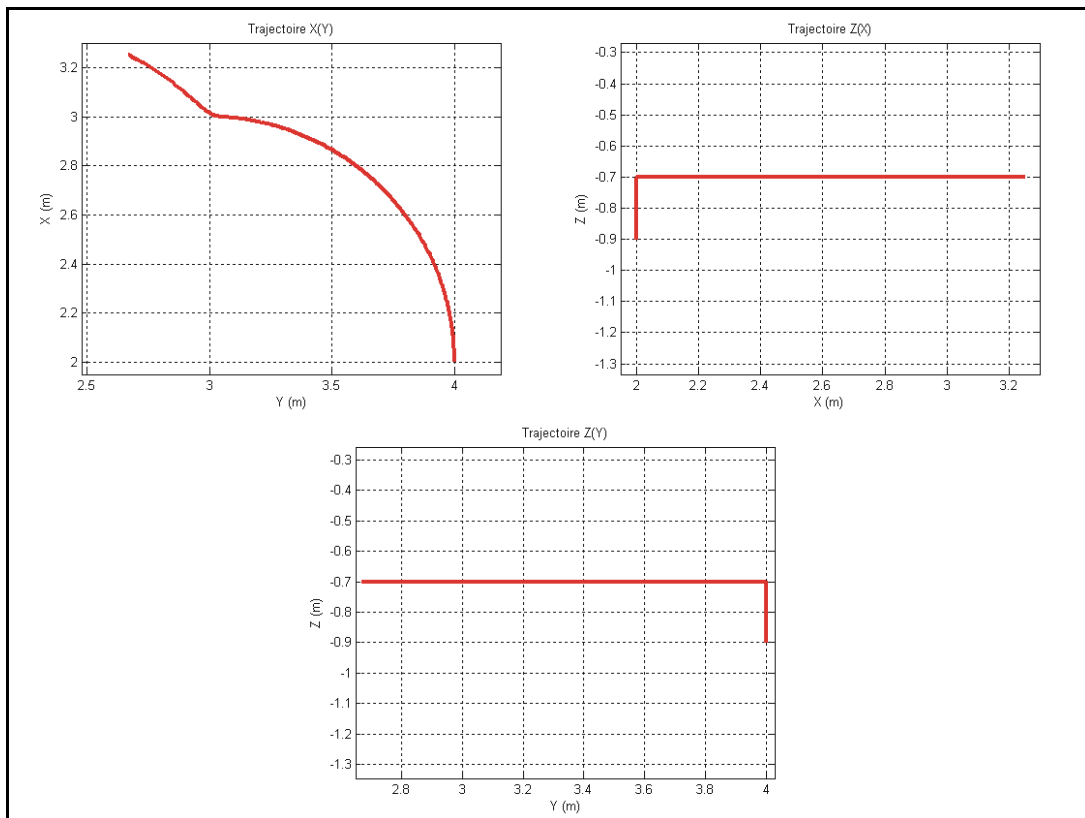


Figure 4. 28 : Trajectoire du point de saisie de la structure SCARA.

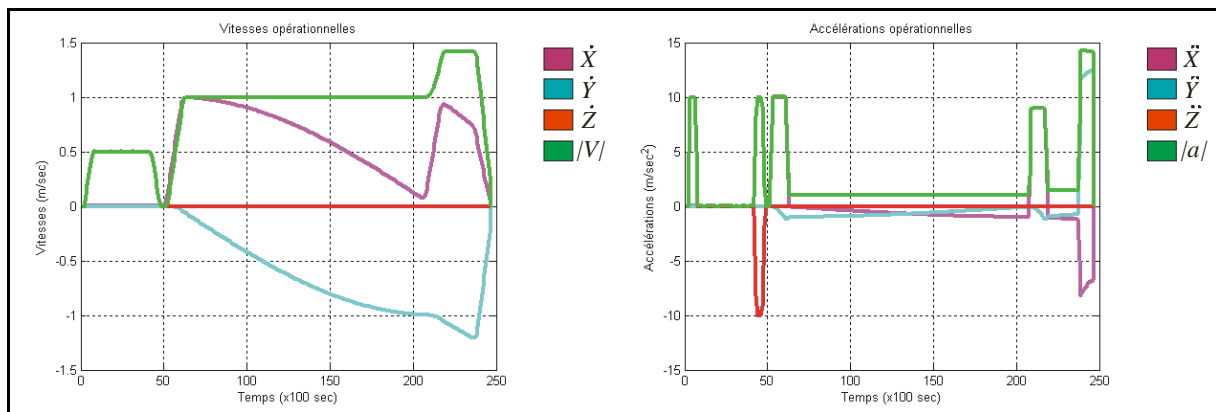


Figure 4. 29 : Vitesses et accélérations opérationnelles du point de saisi de la structure SCARA.

4.4.7. Application du modèle géométrique direct pour une structure PUMA.

On considère le robot décrit en (§ 4.3.8). On positionne le robot selon la matrice de transformation homogène suivante :

$${}^wT_0 = \begin{bmatrix} 0 & -1 & 0 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (4. 17)}$$

On résume toute l'analyse dans les figures suivantes :

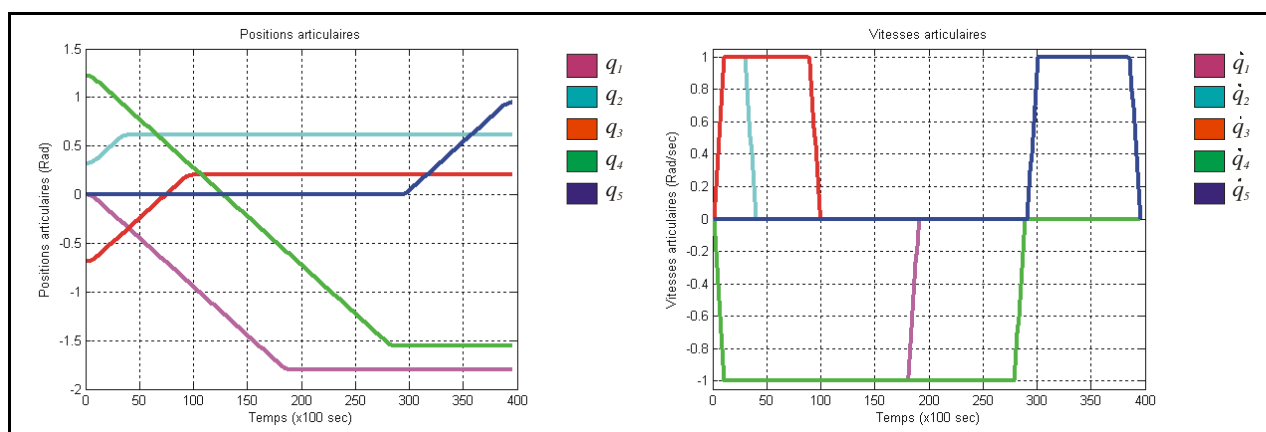


Figure 4. 30 : Paramètres cinématiques articulaires appliquées à la structure PUMA.

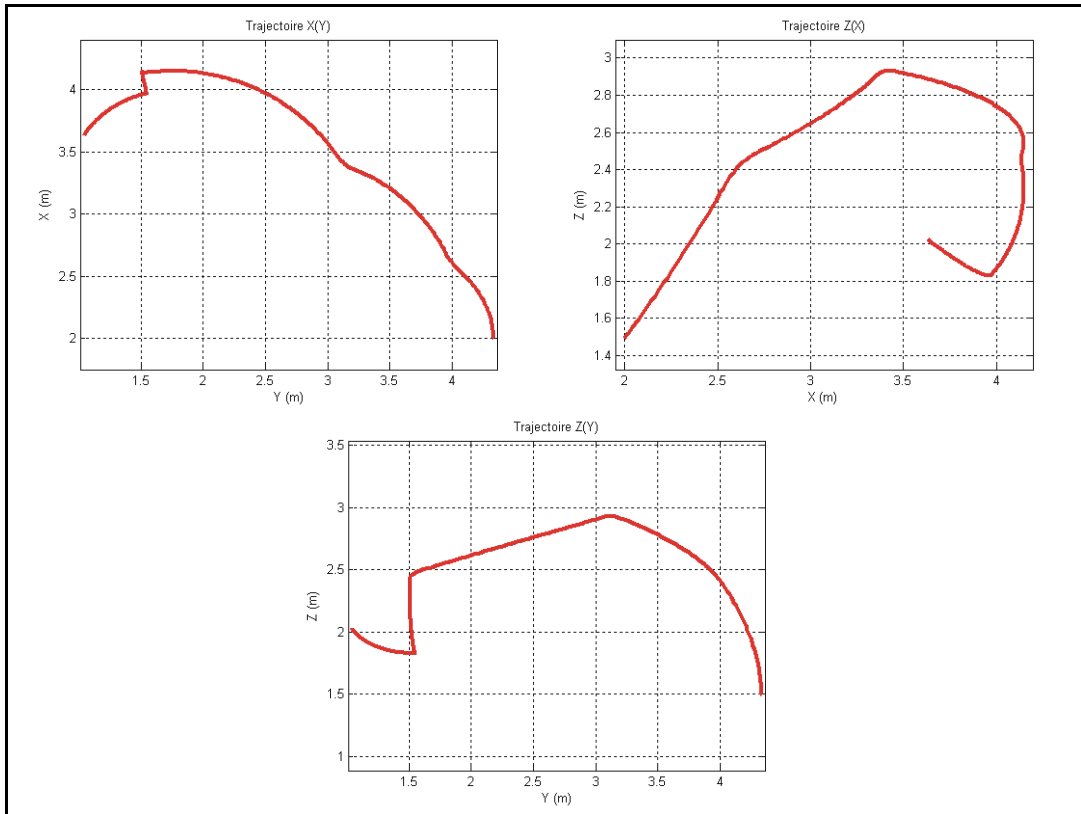


Figure 4. 31 : Trajectoire du point de saisie de la structure PUMA.

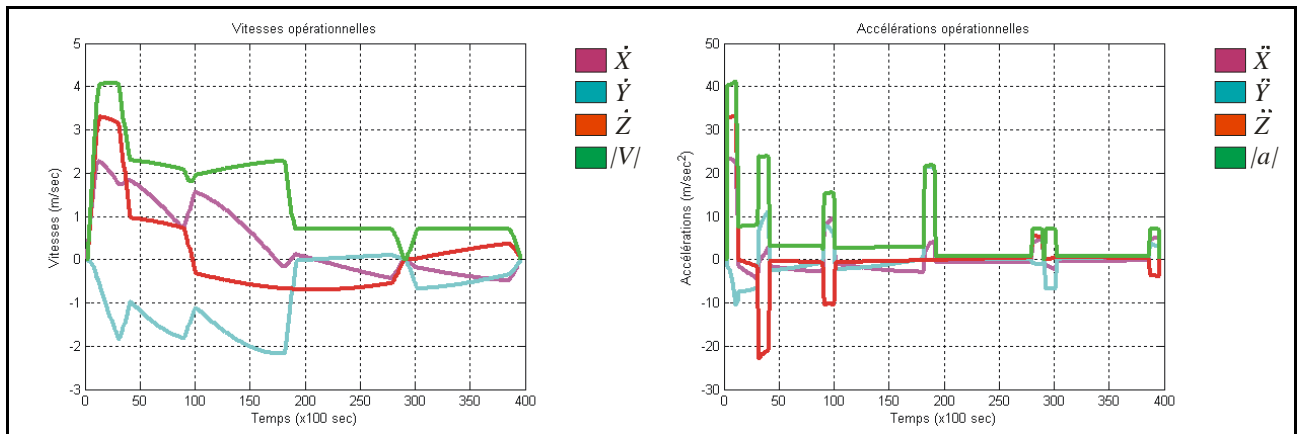


Figure 4. 32 : Vitesses et accélérations opérationnelles du point de saisie de la structure PUMA.

4.5. Application du modèle géométrique inverse pour un robot plan à deux ddl. par une méthode géométrique.

On considère le robot décrit en (§ 4.3.3)

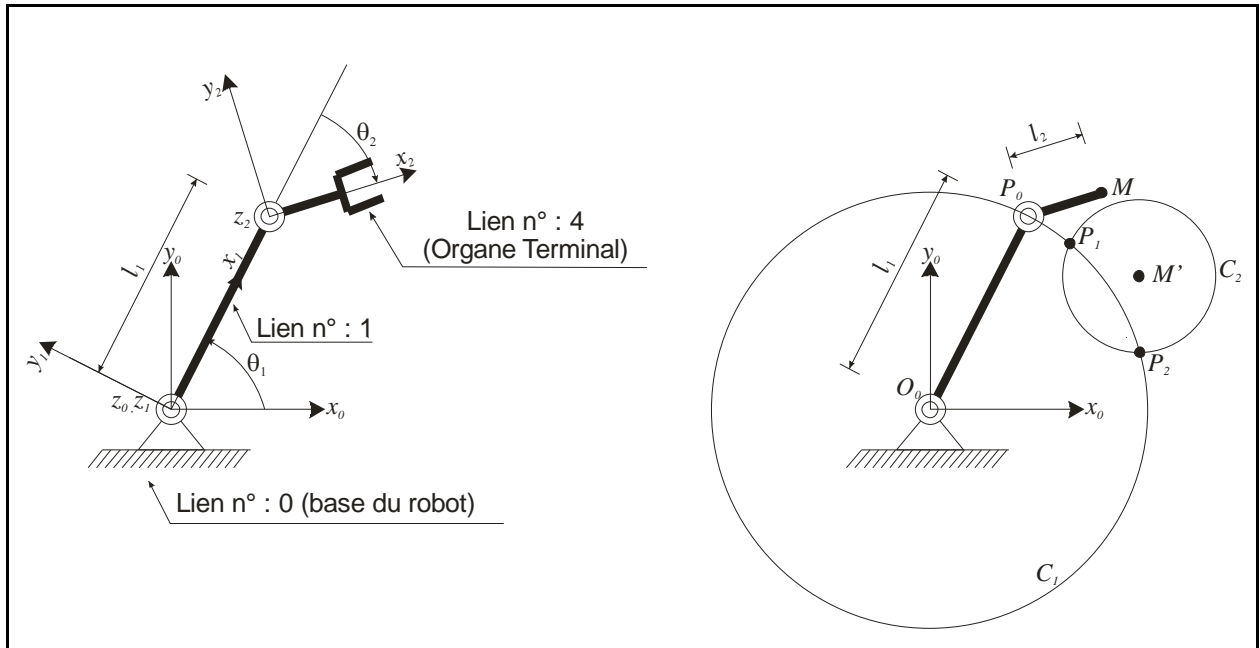


Figure 4. 33 : Résolution du modèle géométrique inverse par une méthode géométrique.

Le point de saisie de l'organe terminal se trouve initialement à la position M , et on veut atteindre la position M' . Dans l'ensemble, on possède deux points fixes : l'origine qui appartient à la base et le point objectif M' . L'ensemble des solutions cherchées est donc $\{P_1, P_2\}$: les deux points qui consistent en l'intersection des deux cercles $C_1(O_0, \text{Rayon} = l_1)$ et $C_2(M', \text{Rayon} = l_2)$. En général l'intersection des deux cercles donne l'une des trois possibilités :

- Ensemble vide : donc le point cherché n'appartient pas au domaine opérationnel du robot ;
- Ensemble contenant un seul point : dans ce cas, on essaye d'éviter cette solution, car elle présente une position singulière pour le robot ;
- Ensemble contenant deux points $\{P_1, P_2\}$: on choisira celui qui donnera la configuration la plus proche de la situation initiale. On peut choisir le point P qui donnera le plus petit vecteur en module entre les deux vecteurs $\overrightarrow{P_0P_1}$ et $\overrightarrow{P_0P_2}$.

La méthode de résolution de l'intersection de deux cercles dans un plan, est bien détaillée en annexe C.

4.6. Application du modèle dynamique pour un mouvement à vide & comparaison de résultats.

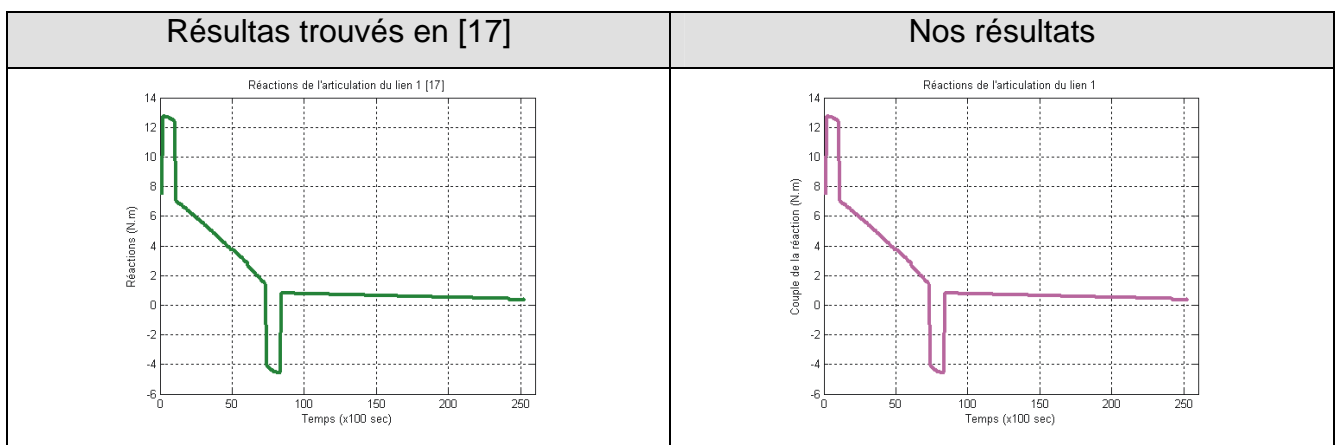
Afin d'analyser la fiabilité des résultats obtenus en appliquant l'algorithme de Newton-Euler, on a pris un exemple traité en [17] pour un mécanisme plan à deux degrés de liberté d'une structure similaire à celle du robot décrit en (§ 4.3.3) à la différence de la distribution de masse qui est décrite par le tableau (4.13).

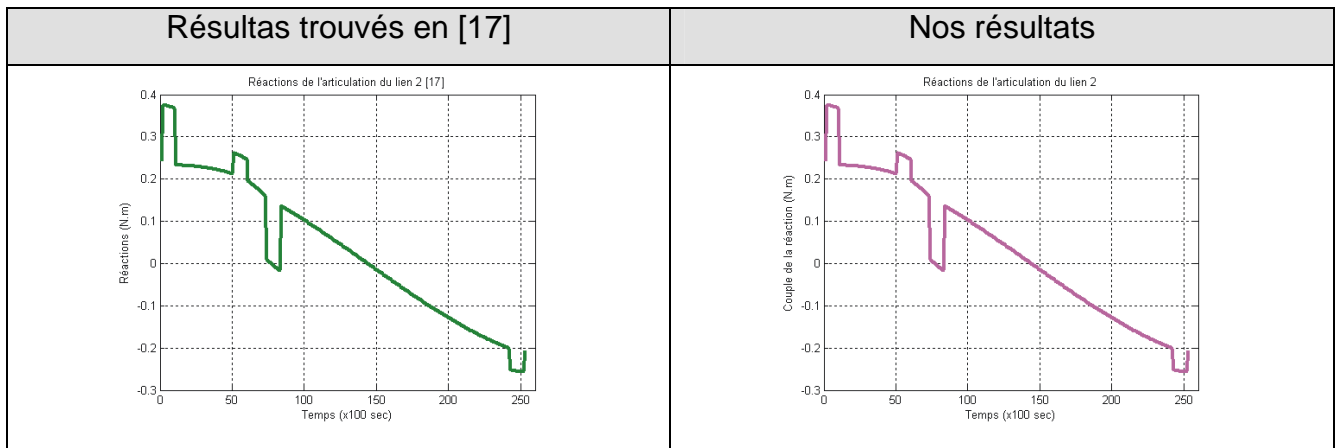
Tableau 4. 13 : Paramètres des liens du robot plan à 2 ddl.

N° du lien	Longueur du lien (m)	Masse du lien (kg)	Position du centre de masse (/ Repère du lien)	Tenseur d'inertie (/ Repère du lien) (kg.m ²)
1	$l_1 = 0.5$	1.9267	(0.5,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.4817 & 0 \\ 0 & 0 & 0.4817 \end{bmatrix}$
2	$l_2 = 0.2$	0.1233	(0.2,0,0)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.0049 & 0 \\ 0 & 0 & 0.0049 \end{bmatrix}$

En considérant un mouvement similaire à celui décrit en (§ 4.4.2), on compare les résultats de l'analyse dynamique à vide (sans porter d'objet) pour ce robot entre l'exemple traité en référence [17] et celui traité par notre algorithme par le tableau suivant :

Tableau 4. 14 : Comparaison des résultats pour l'analyse dynamique d'un robot plan à deux degrés de liberté.





La similitude apparente des courbes donne une lueur de fiabilité des résultats obtenus aussi bien pour le modèle dynamique que pour les modèles géométrique et cinématique, du fait que le modèle dynamique utilise à son tour les deux autres modèles. On peut tirer profit de cette comparaison favorable pour proposer d'autres différents exemples de manipulation d'objet avec différentes structures et différents mouvements et d'en étudier les comportements cinématiques et dynamiques et même de pouvoir dimensionner certains composants de robots.

4.7. Application du modèle dynamique pour la manipulation d'un objet par un seul robot.

4.7.1. Description de l'objet à manipuler.

On considère un objet sphérique de diamètre $\phi_{Objet} = 10cm$ en acier (masse volumique $\rho = 7850 \text{ kg/m}^3$). On détermine donc la masse de l'objet comme suit :

$$M_{Objet} = \rho \cdot \frac{4\pi \left(\frac{\phi_{Objet}}{2} \right)^3}{3} = 4.1103 \text{ kg} \quad \text{---- (4. 18)}$$

Le moment d'inertie de l'objet par rapport à un axe passant par son centre de gravité est donc :

$$I = \frac{2.M_{Objet} \left(\frac{\phi_{Objet}}{2} \right)^2}{5} = 0.0041 \text{ kg.m}^2. \quad \text{---- (4. 19)}$$

Et le tenseur d'inertie de l'objet :

$$TI_{Objet} = \begin{bmatrix} 0.0041 & 0 & 0 \\ 0 & 0.0041 & 0 \\ 0 & 0 & 0.0041 \end{bmatrix} \quad \text{---- (4. 20)}$$

En vu d'être manipulé par un robot, l'objet possède une prise dont la situation relative au repère de l'objet est donnée par la matrice suivante :

$${}_{Objet}T_{Pr} = \begin{bmatrix} 1 & 0 & 0 & -0.05 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (4. 21)}$$

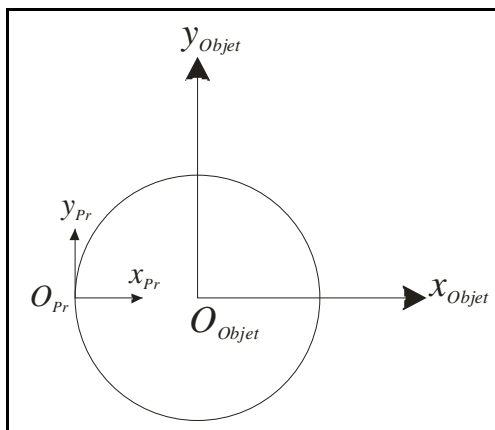


Figure 4. 34 : Description de l'objet à manipuler par un seul robot.

La situation de l'objet quand il est manipulé par un robot est déterminée par les relations décrites au chapitre 2 (§ 2.8.4).

4.7.2. Manipulation de l'objet par un robot plan à 2ddl.

On considère le robot décrit en (§ 4.3.3). Et on veut manipuler l'objet décrit en (§ 4.5.1) suivant le mouvement décrit en (§ 4.4.2). L'analyse dynamique a donné les résultats résumés dans les courbes suivantes :

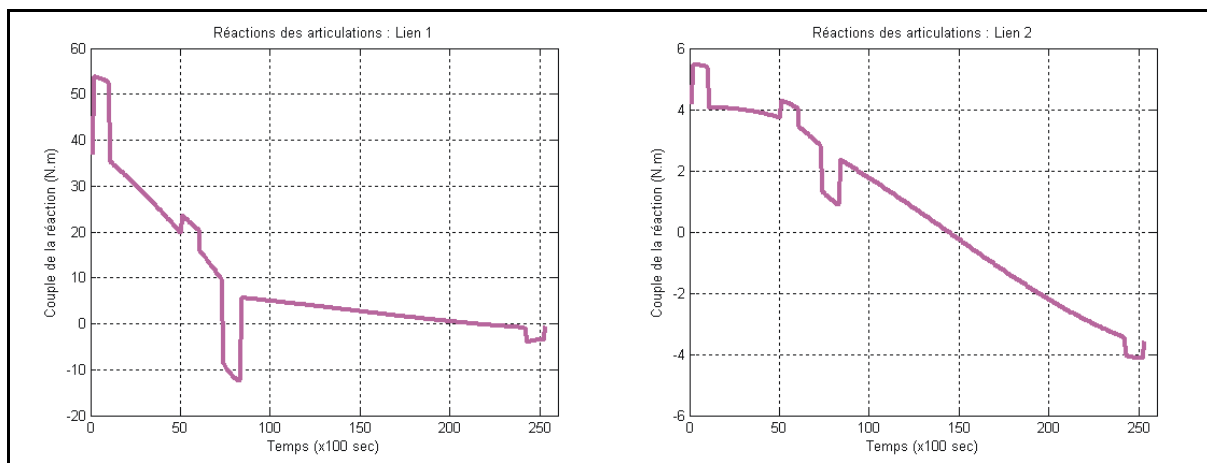


Figure 4. 35 : Analyse dynamique de la manipulation d'un objet par un seul robot plan à 2 ddl.

On remarque une similitude des formes des courbes avec celles obtenues en (§ 4.6) et ceci est traduit par la similitude des deux structures et des dimension des robots utilisés dans les deux exemples, tandis que les valeurs sont beaucoup plus importantes que celles du premier exemples, et ceci est évident du fait que dans cet exemple on manipule un objet à la différence de l'exemple précédent qui consistait à un mouvement à vide.

On remarque également dans les deux exemples que les couples du premier lien sont plus importantes que celles de l'organe terminal, ceci est du à la distance (bras de levier) entre l'articulation de la base et l'objet qui est généralement plus importante que celle entre l'articulation de l'organe terminal et l'objet.

4.8. Application du modèle dynamique pour la manipulation d'un objet par deux robots coopérants.

4.8.1. Description de l'objet à manipuler.

On considère un objet sous la forme d'une barre de longueur $l_{barre} = 30cm$ et de diamètre $\phi_{barre} = 5cm$ en acier (masse volumique $\rho = 7850 \text{ kg/m}^3$). L'objet possède deux prises P_1 et P_2 afin d'être manipulé par deux robots. Les prises sont considérées comme des sphères pour que la liaison (Organe terminal / Prise de l'objet) soit liaison rotule sphérique. On considère que les sphères sont de masses négligeables en comparaison avec celle de la barre.

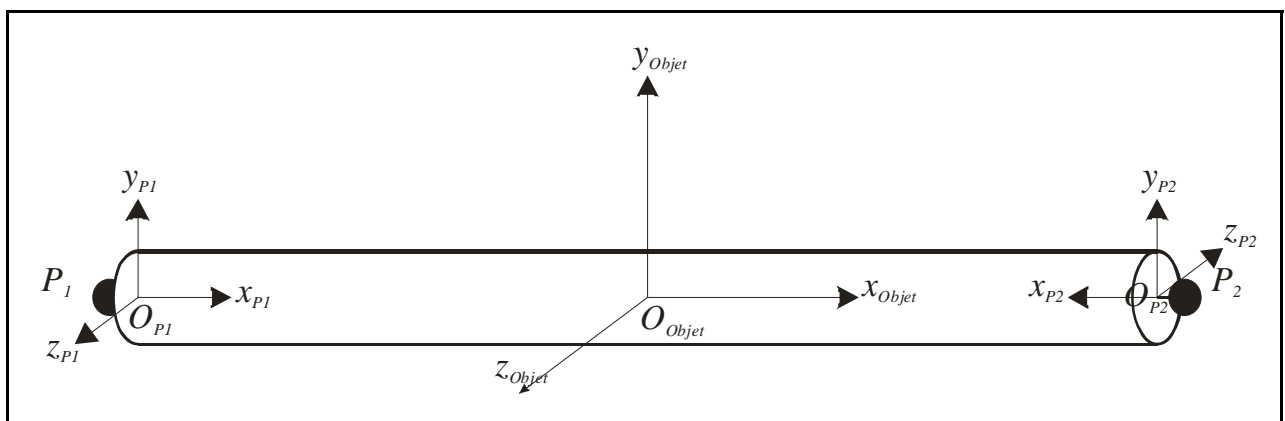


Figure 4. 36 : Description de l'objet à manipuler par deux robots.

Les situations des deux prises P_1 et P_2 par rapport au repère de l'objet sont déterminées respectivement par :

$${}_{Objet}T_{P1} = \begin{bmatrix} 1 & 0 & 0 & -\frac{l_{barre}}{2} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -0.15 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{-----(4. 22)}$$

$${}_{Objet}T_{P2} = \begin{bmatrix} -1 & 0 & 0 & \frac{l_{barre}}{2} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0.15 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{-----(4. 23)}$$

On détermine la masse de l'objet comme suit :

$$M_{Objet} = \rho \cdot l_{barre} \cdot \pi \cdot \frac{(\phi_{barre})^2}{4} = 4.624 \text{ kg} \quad \text{---- (4. 24)}$$

La position du centre des masses de l'objet étant $CG_{Objet} = [0,0,0]^T$ exprimée dans le repère de l'objet.

Et le tenseur d'inertie de l'objet :

$$TI_{Objet} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.03468 & 0 \\ 0 & 0 & 0.03468 \end{bmatrix} \quad \text{---- (4. 25)}$$

4.8.2. Description de la scène de la manipulation.

On considère deux robots plan : le premier est identiques à celui décrit en (§ 4.3.3), le second possède la même structure que le premier à la différence des valeurs maximale et minimale de l'articulation du lien n° 02 données par : $(q_2)_{\min} = -\frac{\pi}{2}$ et $(q_2)_{\max} = \frac{3\pi}{2}$. Les deux robots coopèrent pour manipuler l'objet décrit en (§ 4.1.1).

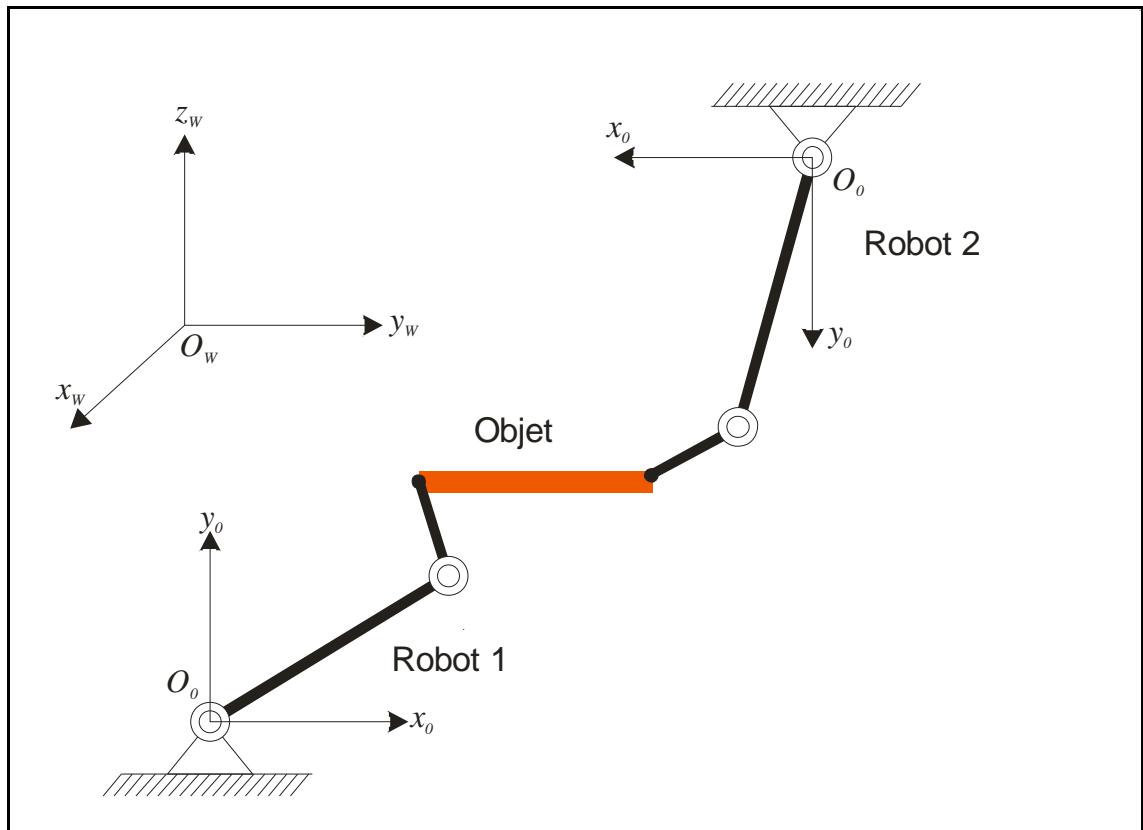


Figure 4. 37 : Description de la scène de manipulation d'un objet par deux robots plans à 2 ddl.

Les bases des deux robots sont positionnées relativement au repère atelier respectivement selon les matrices suivantes :

$${}^wT_0^1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0.6560539 \\ 0 & 1 & 0 & 0.0178444 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----(4. 26)}$$

$${}^wT_0^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1.5253869 \\ 0 & -1 & 0 & 1.0491524 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----(4. 27)}$$

4.8.3. Description du mouvement de l'objet.

Initialement l'objet se situe dans le repère atelier selon la matrice suivante :

$${}^wT_{Objet} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1.1447357 \\ 0 & 1 & 0 & 0.4351698 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----(4. 28)}$$

Afin que l'objet soit saisi par les deux robots, ces derniers doivent avoir les positions articulaires suivantes (Paramètres de Denavit-Hartenberg) :

$$\text{Pour le robot 1 : } \begin{cases} \theta_1 = 0.507766 \text{ (Rad)} \\ \theta_2 = 1.576467 \text{ (Rad)} \end{cases}$$

$$\text{Pour le robot 2 : } \begin{cases} \theta_1 = 1.430702 \text{ (Rad)} \\ \theta_2 = -0.794176 \text{ (Rad)} \end{cases}$$

On considère que l'objet doit effectuer une rotation d'un angle $\theta = \frac{\pi}{2}$

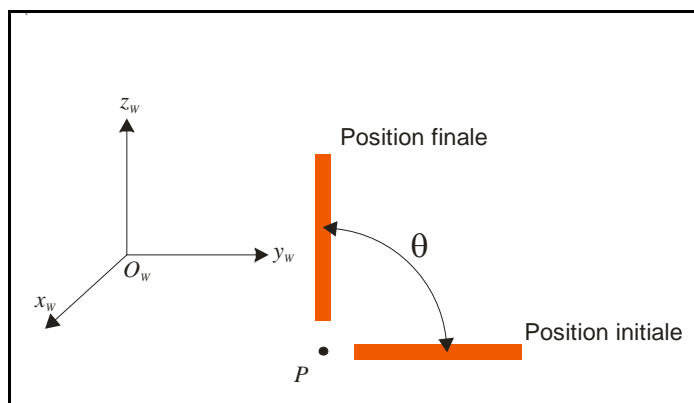


Figure 4. 38 : Description du mouvement d'un objet manipulé par deux robots plans à 2 ddl.

Suivant une loi « trapèze », l'objet effectuera le mouvement rotatif autour du point P. le mouvement est décrit par les courbes suivantes :

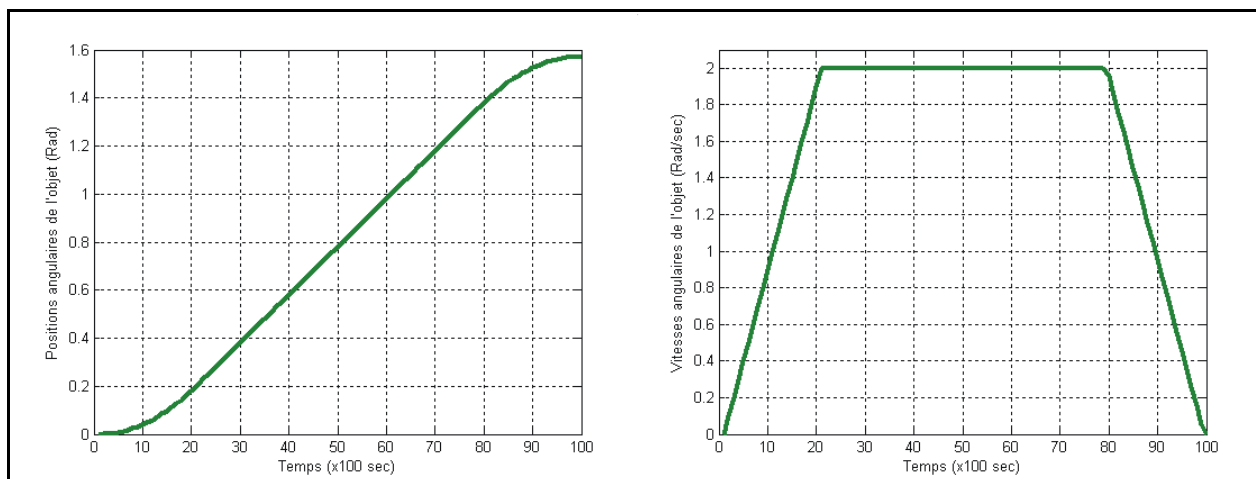


Figure 4. 39 : Mouvement rotatif d'un objet en vue d'être manipulé par deux robots plans à 2 ddl.

On a pris le point $P = (0,0.9444908,0.4351698)$ comme centre pour le mouvement circulaire que l'objet doit effectuer.

Une discrétisation du mouvement de l'objet à un pas de 0.01 secondes, et l'application du modèle géométrique inverse pour les deux robots d'après chaque point occupé de l'objet, ont donné les positions articulaires correspondantes des deux robots qui sont résumés dans les figures suivantes :

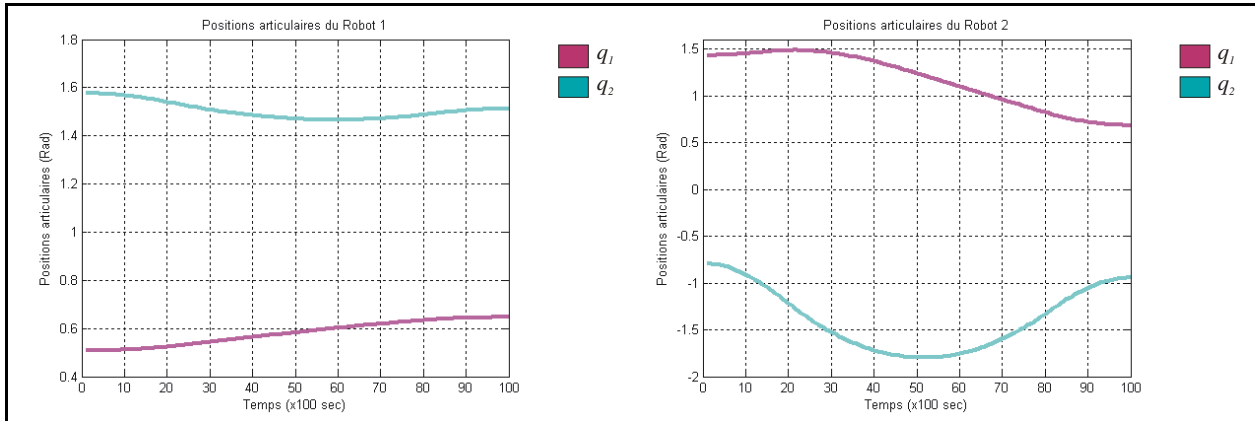


Figure 4. 40 : Positions articulaires des liens des deux robots obtenues en appliquant le modèle géométrique inverse correspondant aux positions occupées par l'objet.

Une dérivation numérique des positions articulaires a donné les vitesses articulaires montrées dans les figures suivantes :

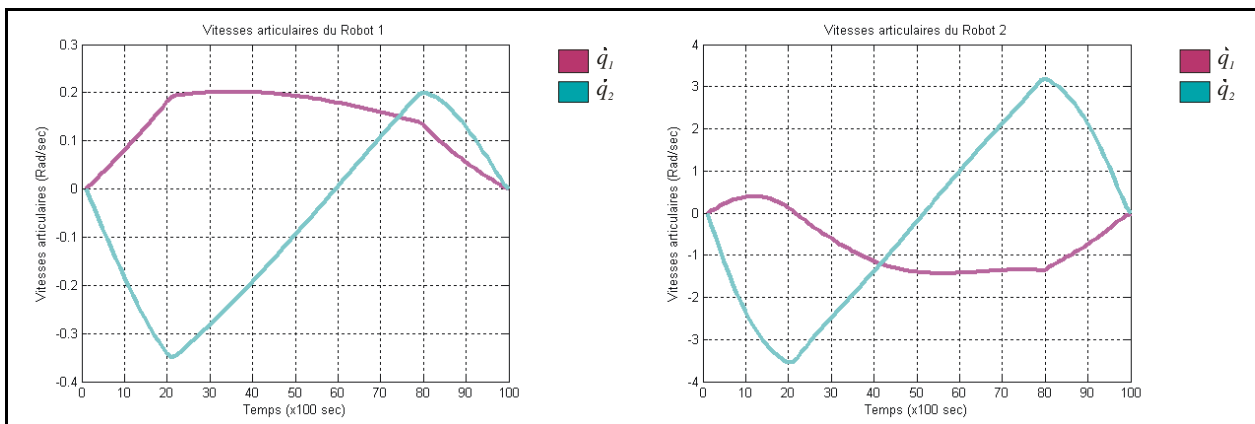


Figure 4. 41 : Vitesses articulaires des liens des deux robots obtenues par dérivation numérique des positions articulaires.

Et finalement une seconde dérivation numérique a donné les accélérations articulaires :

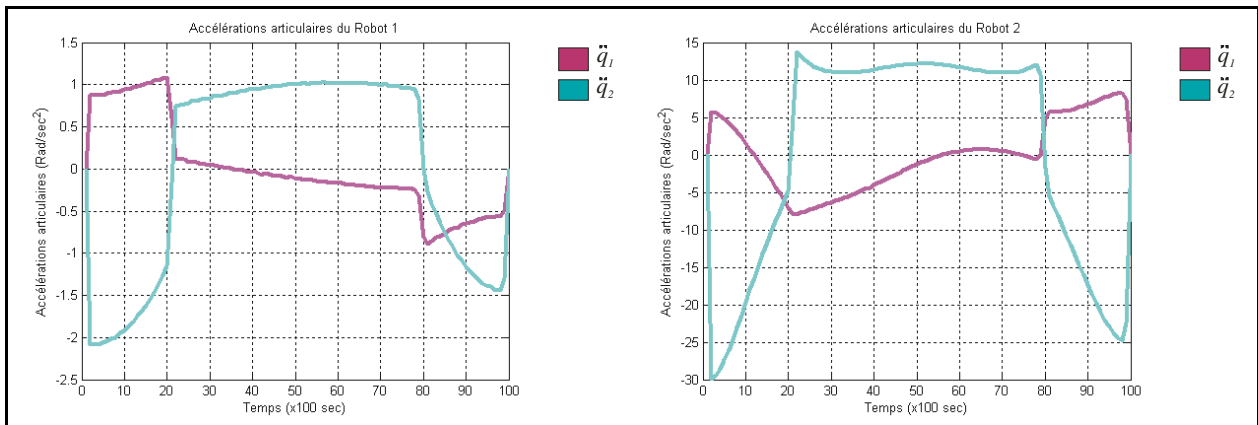


Figure 4. 42 : accélérations articulaires des liens des deux robots obtenues par une seconde dérivation numérique des positions articulaires.

On a appliqué l'algorithme de Newton-Euler avec une décomposition de la charge selon la méthode décrite au chapitre 2 (§ 2.8.6), et on a obtenu les résultats suivants :

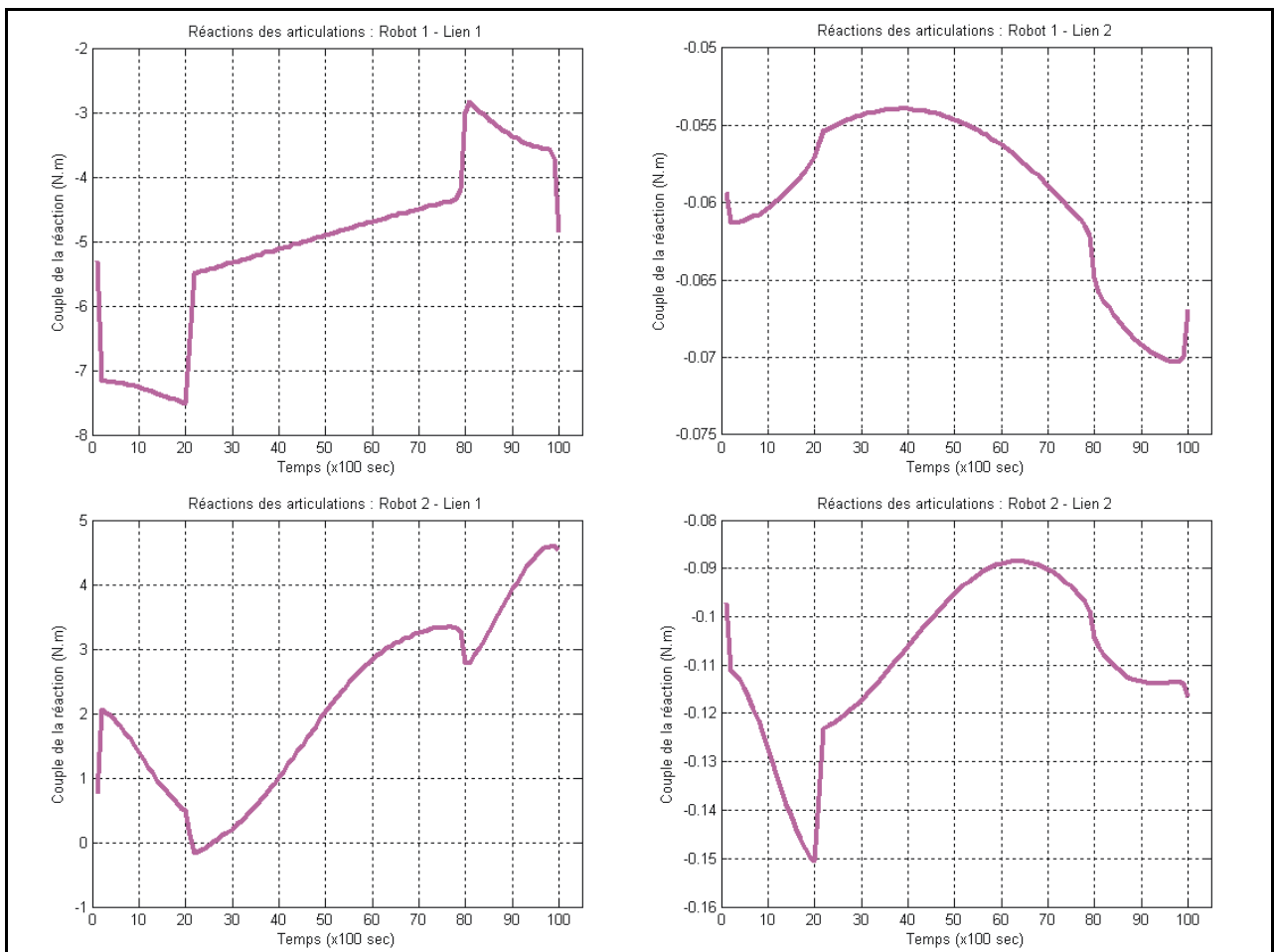


Figure 4. 43 : Réactions des articulations des deux robots coopérants pour la manipulation d'un objet.

D'après les courbes précédentes, on remarque que les réactions des liens du robot 1 sont légèrement plus petites que celles du second robot, ceci est traduit par la difficulté des mouvements demandés à chacun des deux robots. Selon la scène décrite dans la figure (4.37), on peut déduire que le travail demandé du second robot est plus important que celui demandé du robot 1.

D'autre part, en comparaison avec l'exemple traité en (§ 4.7.2), tenant compte que les objets manipulés dans les deux exemples sont de masses voisines, on remarque que les efforts articulaires sont considérablement plus faible dans le cas d'une coopération, ceci traduit donc l'efficacité de la coopérations de robots du point de vue « *économie d'énergie* ».

4.9. Conclusion.

Dans ce chapitre les problèmes ont été abordés suivant un ordre de complexité croissant. D'abord, le choix de six (06) structures les plus répondues dans la littérature robotique, et ensuite, l'application du modèle géométrique direct pour l'ensemble des structures étudiées. Quand au modèle géométrique inverse, on a choisi une structure plane simple, proposé une méthode géométrique pour décrire la géométrie inverse et obtenu la cinématique inverse par des méthodes numériques. Affin de s'assurer de la fiabilité des modèles proposés, on a choisi un exemple traité dans la littérature, et on a confronté les résultats du modèle dynamique qui repose sur tous les autres modèles.

Pour la coopération de robots, on a traité un problème de manutention avec deux robots plans, et ensuite comparé les résultats avec la manutention d'un objet similaire avec un seul robot. La comparaison nous a permis de valoriser qualitativement l'efficacité de la coopération.

CONCLUSION

La modélisation est une étape cruciale dans un sujet de robotique. Qu'il s'agit du dimensionnement des composants du robot ou de la planification d'une trajectoire pour l'organe effecteur, un modèle régissant le comportement du robot et tenant compte de tous les paramètres y influant s'avère indispensable. La modélisation d'une scène de manipulation d'un objet par des robots comporte aussi bien pour l'objet que pour les robots contribuant à la manipulation les modèles suivant : le modèle géométrique, le modèle cinématique et le modèle dynamique. Ces modèles peuvent être directes si les grandeurs cherchées sont exprimées dans l'espace opérationnel, ou inverses si elles sont exprimées dans l'espace articulaire.

Dans un esprit de proposer la démarche suivie lors de l'étude d'un sujet de robotique, notre travail portait particulièrement sur la modélisation des bras manipulateurs. Une recherche bibliographique approfondie nous a aidé à exposer les différents concepts théoriques et formalismes nécessaires pour ceux qui veulent s'initier dans ce domaine.

Notre thème comme son intitulé l'indique, porte sur : « *la modélisation dynamique* », « *les robots coopérants* » et « *la planification des trajectoires* », trois axes différents dont on a présenté chacun d'eux en détails, et trouvé un point de liaison à travers une application de manipulation d'un objet avec deux robots.

Le modèle dynamique a été le sujet de plusieurs travaux de recherches dont tous pratiquement s'articulaient sur l'utilisation de l'un des deux formalismes : Le formalisme du « *lagrangien* » et celui de « *Newton-Euler* ». Une présentation des deux formalismes nous a mené d'adopter le deuxième pour son caractère « *généraliste* » et sa récursivité favorisée pour la programmation informatique.

Inévitablement, le modèle dynamique utilise les modèles : géométrique et cinématique pour calculer les efforts auxquels les différents liens sont assujettis. Le chapitre 2 a été présenté dans un ordre favorisant la compréhension pour le lecteur. Commencant par la description des positions des liens relativement aux différents repères du robot ce qui

consiste en le modèle géométrique direct, ensuite, une méthode algorithmique pour le calcul du jacobien utilisé pour le calcul des vitesses et accélérations, et enfin, la dynamique pour un objet se déplaçant suivant une loi donnée ainsi que pour un robot sollicité à des efforts extérieurs.

Pour le cas de la manipulation d'un objet par deux ou plusieurs robots, deux approches ont été fortement utilisées par les auteurs. La première considère le système « *Robots/Objet* » comme un système fermé, et donc utiliser le modèle dynamique concernant les robots parallèles. La seconde approche s'appuie sur la décomposition de la charge de l'objet en deux ou plusieurs (selon le nombre de robots coopérants), et traiter chaque robot comme s'il était seul à supporter cette partie de la charge. Notre choix a été porté sur la deuxième approche pour bénéficier des algorithmes développés pour le cas d'un seul robot.

Notre troisième axe d'étude s'est porté sur les trajectoires. D'abord une synthèse des approches présentes dans la littérature dans le chapitre 3, et ensuite une application dans le chapitre 4 consistant à la génération du mouvement pour deux robots coopérants pour la manipulation d'un objet dont la trajectoire opérationnelle est connue. La méthode géométrique qu'on a proposé pour résoudre le modèle géométrique inverse, nous a aussi servi de support de comparaison de résultats pour les paramètres cinématiques obtenu par les relations de la cinématique du solide, et ceux obtenus par nos algorithmes après dérivation numérique.

Concernant la fiabilité de nos résultats, de la littérature on a pris pour référence, un exemple traitant la dynamique d'un robot à deux ddl, et on a exécuté notre algorithme pour le même exemple, et on s'est appuyé sur la similitude des résultats obtenus avec ceux considérés comme référence. De ce fait on a proposé d'autres exemples traitant aussi bien le modèle dynamique que les modèles : géométrique et cinématique ainsi que les trajectoires dans l'espace opérationnel.

Notre travail peut être considéré comme une base de départ pour une recherche approfondie dans le domaine des bras manipulateurs et des robots coopérants. Dans un travail futur, une étude complémentaire peut être envisagée en proposant des méthodes pour :

- La résolution du modèle géométrique inverse pour les structures complexes ;

- Incorporer la perception pour tenir compte des environnements non connus où le robot peut évoluer ;
- Tenir compte du frottement et des inerties des actionneurs pour mieux approcher les résultats du modèle dynamique à la réalité.

RÉFÉRENCES

1. E. DOMBRE – W. KHALIL
« *Modélisation identification et commande des robots.* ». HERMES 1999.
2. J.M. SELIG
« *Introductory Robotics.* ». Prentice Hall International (UK) Ltd. 1992
3. I. ARTOBOLEVSKI
« *Théorie des mécanismes et des machines.* ». MIR – Moscou 1977.
4. J. HERING
« *Cours de Mécanique* ». OPU – Alger 1993
5. M. W. SPONG - M. VIDYASAGAR.
« *Robot Dynamics and control.* ». Library of Congress Cataloging. 1989. ISBN 0-471-61243-X
6. C. PRELLE – Thèse de Doctorat – INSA Lyon 1997.
« *Contribution au contrôle de la compliance d'un bras de robot à actionnement électropneumatique.* ».
7. F. BENNIS, P. WENGER, S. CARO – Article : Institut de Recherche en communications et Cybernétique de Nantes
« *Etat de l'art de la conception robuste de mécanismes.* ».
8. L. FLÜCKIGER - Thèse de Doctorat – Lausanne 1998.
« *Interface pour le pilotage et l'analyse des robots basée sur un générateur de cinématiques* »
9. D. CHABLAT - Thèse de Doctorat – Nantes 1998.
« *Domaines d'unicité et parcourabilité pour les manipulateurs pleinement parallèles.* ».
10. M.J.D.HAYES^{1,2}, M.L.HUSTY³, P.J.ZSOMBOR-MURRAY⁴ – Article : April 2, 2003
¹ : Mechanical & Aerospace Engineering, Carleton University, Canada.
² : Institut Für Automation, Montan Universität Leoben, Austria.
³ : Institut Für Technische Mathematik, Geometrie und Bauinformatik, Universität Innsbruck, Austria.
⁴ : Centre For Intelligent Machines, McGill University, Canada.
« *Singular Configurations of Wrist-Partitioned 6R Serial Robots: a Geometric Perspective for Users.* ».
11. N. ANDRE – Document de cours - Mai 2006
« *Des droites et des robots.* »
12. N. ANDREFF¹, C.B. BOUZGARROU¹, T. CANO¹, H. CHANAL¹, F. CHAPELLE¹, J. C. FAUROUX¹, G. GOGU¹, N. RAT¹, P. RAY¹, R. RIZK¹, D. CHABLAT², O. IBRAHIM², S. CARO², F. BENNIS², W. KHALIL², P. WENGER², P. MARTINET³, D. DANAY⁴, J.-P. MERLET⁴, A. FOURNIER⁵, O. COMPANY⁵, E. DOMBRE⁵, S. KRUT⁵, V. NABAT⁵, P. POIGNET⁵, S. RONCHI⁵, F. PIERROT⁵.
¹LaRAMA, ²IRCCyN, ³LASMEA, ⁴INRIA, ⁵LIRMM.
« *Projet ROBEA MP2.* ».

13. P. Poignet¹, G. Poisson² – Article :
¹ Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier
² Laboratoire de Vision et Robotique de Bourges
 « *La conception de robots pour l'assistance aux gestes Médicaux et chirurgicaux* ».
14. S. CHESSE – Thèse de Doctorat – Poitiers 1992
 « *Dynamique optimale de systèmes articulés à cinématique fermée - Application à la synthèse d'allures de marche optimales.* ».
15. S. Lacroix¹, P. Fabiani², D. Dufourd³. Projet Acrobate.
¹ LAAS/CNRS - ² Onera/DCSD - ³ DGA/CEP
 « *Algorithmes pour la coopération entre robots terrestres et aériens.* ».
16. H. ASADA - J.J.E. SLOTINE.
 « *Robot analysis and control.* ». Library of Congress Cataloging 1985 – ISBN 0-471-83029-1
17. J.J. CRAIG.
 « *Introduction to ROBOTICS – Mechanics and control.* ». Library of Congress Cataloging – 1986 – ISBN 0-201-10326-5
18. M. CAZIN – J. METGE.
 « *Introduction to ROBOTICS – Mechanics and control.* » BORDAS Paris 1989. ISBN 2-04-018794-4
19. P. CHIACCHIO – S. CHIAVERINI
 « *Complex robotics systems.* » - SPRINGER – LONDON 1998
20. J. Angeles.
 « *Fundamentals of Robotics Mechanical Systems.* » SPRINGER 1997.
21. C. WELMAN – Thèse Master – SIMON FRASER 1993.
 « *Inverse Kinematics and Geometrics Constraints For Articulated Figure Manipulation.* ».
22. R. KULPA – Thèse de Doctorat – INSA Rennes 2005.
 « *Adaptation interactive et performante des mouvements d'humanoïdes synthétiques : aspect cinématique – cinétique et dynamique .* ».
23. R. BOULIC – Article : Virtual Reality Laboratory. Swiss Federal Institute of Technology Lausanne, Switzerland.
 « *Cinématique inverse pour l'optimisation de postures humaines soumises à des contraintes multiples* ».
24. P. GLARDON – Thèse de Doctorat – LAUSANNE 2005.
 « *On-Line Locomotion synthesis for virtual humans.* ».
25. F. FAURE – Article : UJF-Grenoble 2002.
 « *Outils Mathématiques : Position, Orientation et Mouvement* »
26. C. TOMASI – Document de cours : Stanford 2000.
 « *Mathematical Methods for Robotics and Vision* »
27. J. A. MEYER, A. GUILLOT – Article : POURLASCIENCE – N° 284 Juin 2001.
 « *La robotique évolutionniste* »
28. S. GUEGAN, W. KHALIL, D. CHABLAT, P. WENGER : Article : Institut de Recherche en Communications et Cybernétique de Nantes.
 « *Modélisation Dynamique d'un Robot Parallèle à 3-DDL : l'Orthoglide* »
29. F. L. CHERNOUSKO, N. N. BOLOTNIK, V. G. GRADETSKY
 « *Manipulation robots* »
 CRC Press 1994. ISBN 0-8493-4457-3
30. A. A. KOBRINSKI, A. E. KOBRINSKI
 « *Bras manipulateurs des robots- Architecture et théorie* »
 MIR Moscou 1989

31. K. KOZTOWSKI
« *Modeling and Identification in Robotics* »
Springer 1998 - ISBN 3-540-76240-X
32. J. DUFFY
« *Statics and Kinematics with Applications to Robotics* »
Library of Congress Cataloging 1996 - ISBN 0-521-48213-5
33. P. DUYSINX, M. GERADIN – Document de cours – University Of LIEGE 2004
« *An introduction to Robotics : Mechanical aspects* ».
34. P. CUBAUD – Document de cours – Dpt. Informatique CNAM 2003.
« *Animatique* ».
35. J. F. BONNANS – Document de cours - INRIA 2000.
« *Autour de l'optimisation robuste* ».
36. J. C. SAMIN – Document de cours – Université catholique de Louvain
« *Dynamics Models for Multibody Systems* ».
37. X. BAGUENARD, M. DAO, L. JAULIN, W. KHALIL – Journées Doctorales d'Automatique 2003.
« *Etalonnage géométrique d'un robot à six degrés de liberté* ».
38. J. C. SAMIN – Document de cours – Université catholique de Louvain
« *Génération de trajectoires* ».
39. B. BAYLE – Document de cours – Université Louis Pasteur de Strasbourg – 2004
« *Introduction à la Robotique* »
40. B. BAYLE – Document de cours – Université Louis Pasteur de Strasbourg – 2007
« *Introduction à la Robotique* ».
41. P. FISETTE, H. BUYSE, J. C. SAMIN – Document de cours – Université catholique de Louvain
« *Introduction à la Robotique* ».
42. A. POPIER – Document de cours – Septembre 2005
« *Méthode de Monte Carlo et simulation de variables aléatoires* ».
43. A. MILLET - Document de cours – Université de Paris.
« *Méthodes de Monte Carlo* ».
44. F. KABANZA. – Document de cours – Département d'informatique – Université de Sherbrooke.
« *Planification en intelligence artificielle – Planification des trajectoires* ».
45. M.P. GROOVER, M. WEISS, R.N. NAGEL, N.G. ODREY. - Document de cours – Université catholique de Louvain
« *Programmation des robots industriels* ».
46. J. C. SAMIN, P. FISETTE – Document de cours – Juillet 2003.
« *Symbolic Modeling of Multibody Systems* ».
47. Document de cours - Université catholique de Louvain 2003.
« *Symbolic Modeling of Multibody Systems* »
48. Document de cours – FH DARMSTADT 2000
« *Introduction to Robotics – Trajectory Generation and Robot Programming* »
49. R. HORAUD – Document de cours – INRIA 2000
« *Vision 3D Projective, affine et euclidienne* »
50. D. MANOCHA – Thèse de Doctorat 1992
« *Algebraic and Numeric Techniques for Modeling and Robotics* »
51. K. DOBRINDT – Thèse de Doctorat – L'Ecole Nationale supérieur de Mines – Paris 1994.
« *Algorithmes dynamiques randomisés pour les arrangements. Application à la planification de trajectoires* »

52. B. AROUNA – Thèse de Doctorat – Ecole Nationale des pont et chaussées – 2004
« *Algorithmes Stochastiques et Méthodes de Monte Carlo* ».
53. M. BAILI – Thèse de Doctorat – Ecole centrale de Nantes – Décembre 2004
« *Analyse et classification de manipulateurs 3R à axes orthogonaux* »
54. S. VOROS – Thèse de Doctorat – Université de Paris - Décembre 2006
« *Vers la commande en boucle fermée d'un robot porte-optique : Apport de l'analyse contextuelle d'images endoscopiques* ».
55. G. DUCHEMIN – Thèse de Doctorat – Université de Montpellier II – Décembre 2002
« *Commande et programmation d'un robot d'assistance au geste médical pour des tâches de suivi au contact de tissu mou* »
56. R. PASSAMA - Thèse de Doctorat – Université de Montpellier II – Juin 2006
« *Conception et développement de contrôleurs de robots – Une méthodologie basée sur les composants logiciels* ».
57. S. CARO – Thèse de Doctorat – Ecole centrale de Nantes – Décembre 2004.
« *Conception robuste de mécanismes* »
58. J. M. BOURGEOT – Thèse de Doctorat – Institut National Polytechnique de Grenoble – Octobre 2004.
« *Contribution à la commande de systèmes mécaniques non-réguliers* »
59. P. BARAONA – Thèse de Doctorat – Université PAUL SABATIER de Toulouse – Juillet 1981.
« *Contribution à la réalisation d'un robot manipulant des objets en contact . Commande par retour d'efforts* »
60. A. PICKEL – Thèse de Doctorat – University of Western Australia – May 2003
« *Control for a biped robot with minimal number of actuators* »
61. J. C. HIMMELSTEIN – Thèse de Doctorat – Institut National des Sciences Appliquées de Toulouse – Septembre 2008.
« *Geometric Operators for Motion Planning* »
62. R. BALANIUK – Thèse de Doctorat – Institut National Polytechnique de Grenoble – Septembre 1996.
« *Identification Structurelle* ».
63. C. WELMAN – Thèse Master – Simon Fraser University – Septembre 1993.
« *Inverse kinematics and Geometric constraints for articulated figure manipulation* »
64. B. BAYLE – Thèse de Doctorat – Université Paul SABATIER – Décembre 2001
« *Modélisation et commande cinématiques des manipulateurs mobiles à roues* »
65. E. L. DAMIAN – Thèse de Doctorat – Institut National des Sciences Appliquées de Toulouse – Juillet 2006.
« *Planification de saisie pour la manipulation d'objets par un robot autonome* »
66. S. LAZARD – Thèse de Doctorat – Université de Paris 6 – Mai 1996.
« *Planification de trajectoires de robots mobiles non-holonomes et de robots à pattes* »
67. A. KENZAI – Mémoire DEA – Université d'Angers – Juillet 2005
« *Planification de trajectoires de robots mobiles via des méthodes ensemblistes* ».
68. B. KISS – Thèse de Doctorat – Ecoles des mines de Paris – Avril 2001.
« *Planification de trajectoires et commande d'une classe de systèmes mécaniques plats et liouviliens* »
69. Y. MEZOUAR – Thèse de Doctorat – Université de Rennes I – Novembre 2001.
« *Planification de trajectoires pour l'asservissement visuel* ».
70. M. PASQUIER – Thèse de Doctorat – Institut National Polytechnique de Grenoble – Janvier 1989.
« *Planification de trajectoires pour un robot manipulateur* »

71. E. MAZER – Thèse de Doctorat - Institut National Polytechnique de Grenoble – Mai 2003.
« *Programmation bayésienne des bras manipulateurs* »
72. A. CASTELLET – Thèse de Doctorat – Université Polytechnique de Catalunya – Juin 1998.
« *Solving inverse kinematics problems using an interval method* ».
73. B. NEVEU – Thèse de Doctorat – Université de Nice – Février 2005.
« *Techniques de résolution de problèmes de satisfaction de contraintes* »
74. E. A. ŞİŞBOT. Thèse de Doctorat – Université de Toulouse III – Octobre 2008.
« *Towards Human Aware Robot Motion* »
75. S. BRETON – Thèse de Doctorat – Université de Haute Alsace - Juin 1999.
« *Une approche neuronale du contrôle robotique utilisant la vision binoculaire par reconstruction tridimensionnelle* »
76. C. BOMBIN, L. ROS, F. THOMAS – Article : Institut de Robotica i informatica Industrial (UPC-CSIC) – Barcelona, Spain.
« *A concise Bézier Clipping Technique for solving inverse kinematics problems* »
77. D. MANOCHA, Y. ZHU – Article : IEEE Conf. On Robotics and Automation 1994.
« *A Fast Algorithm and system for the inverse kinematics of general serial manipulators* ».
78. J. P. MERLET. Article : Journal of Mechanical Design – Vol. 123 – Dec. 2001.
« *A Generic Trajectory Verifier for the Motion Planning of Parallel Robots* ».
79. U. BEYER, F. SMIEJA. Article : Research Group for Adaptive Systems. Rep. N° 1994/10.
« *A Heuristic Approach to the inverse Differential Kinematics Problem* ».
80. L. HAN, N. M. AMATO. Article : WAFR 2000 Preprint.
« *A Kinematics Based Probabilistic Roadmap Methode for Closed Chain Systems* »
81. T. SIMEON, J. CORTES, A. SAHBANI, J.P. LAUMOND. Article : IEEE. International Conference on Robotics and Automation, 2002.
« *A Manipulation Planner for Pick and Place Operations under Continious Grasps and Placement* »
82. J. M. AHUACTZIN, K. GUPTA – Article : ICRA 1997
« *A Motion Planning Based Aproach For Inverse Kinematics of Redundant Robots The Kinematic Roadmap* ».
83. O. C. SCHREMPF, U.D. HANEBECK, A. J. SCHMID, H. WORN – Article : IEEE International Workshop on Robots and Human Interactive Communication 2005.
« *A Novel Approach To Proactive Human Robot Cooperation* ».
84. S. M. LAVALLE, J. H. YAKEY, L. E. KAVRAKI – Article : IEEE Intenational Conf. On Robotics and Automation 1999.
« *A Probabilistic Roadmap Approach for Systems with Closed Kinematic Chain* ».
85. M. H. OVERMARS, P. SVESTKA – Article : Utrecht University. Department of Computer Science. UU-CS-1994-03 – January 1994.
« *A Probabilistic learning approach to Motion Planning* ».
86. J. BARRAQUAND, L. KAVRAKI, J. C. LATOMBE, T. Y. LI, R. MOTWANI, P. RAGHAVAN – Article : 7th International Symposium On robotics Research.
« *A Random Sampling Scheme For Path Planning* ».
87. L. FLÜCKIGER – Article : 29th International Symposium On robotics 1998 (ISR98) pp 123-126. Birmingham (UK) April 1998.
« *A Robot Interface using virtual reality and automatic kinematics generator* »

88. P. I. CORKE, B. A. HELOUVRY- Article : Div. Manufacturing Technology CSIRO & Dept. Electrical Engineering and Computer Science, University of Wisconsin.
« *A Search for Consensus Among Model Parameters Reported for the PUMA 560 Robot* »
89. C. L. Nielsen, L. E. KAVRAKI – Article : IROS 2000.
« *A Two Level Fuzzy PRM For Manipulation Planning* »
90. O. IVLEV, A. GRÄSER – Article : Intelligent Automation and Active Systems. Bremen, Sept. 15-17, 1997.
« *An Analytical Method for the Inverse Kinematics of Redundant Robots* »
91. L. Chaimowicz^{1,3}, T. SUGAR², V. KUMAR¹, M. F. M. CAMPOS³ – Article :
¹ : GRASP Laboratory – University of Pennsylvania, Philadelphia, PA, USA.
² : Mechanical and Aerospace Engineering – Arizona State University, Tempe, AZ, USA
³ : DCC – Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
« *An Architecture for Tightly Coupled MultiRobot Cooperation* »
92. J. J. KUFFNER, S. M. LAVALLE – Article : IEEE International Conf. On Robotics and Automation 2000.
« *RRT-Connect : An Efficient Approach To Single-Query Path Planning* »
93. D. BERTRAM^{1,2}, J. KUFFNER², R. DILLMANN¹, T. ASFOUR¹ – Article :
¹ : Institute Of Computer Science and Engineering, University of Karlsruhe, Germany.
² : The Robotics Institute, Carnegie Mellon University, USA.
« *An Integrated Approach to Inverse Kinematics and Path Planning for Redundant Manipulators* »
94. N. M. AMATO, O. B. BAYAZIT, L. K. DALE, C. JONES, D. VALLEJO – Article : Texas A & M University, College Station, TX, USA.
« *OBPRM : An Obstacle-Based PRM for 3D Workspaces* »
95. J. F. ALLAN, C. M. GOSSELIN – Article : Département de Génie Mécanique, Université LAVAL, Québec, Canada.
« *Analyse Dynamique d'un Nouveau Manipulateur Parallèle Spatial à 3 Degrés de Liberté* »
96. P. LUCIDARME – Article : LIRMM (UMR 55060).
« *Apprentissage et Adaptation pour des Ensembles de Robots Réactifs Coopérants* »
97. Y. KUNIYOSHI – Article : Electrotechnical Laboratory, Ibaraki, Japan.
« *Behavior Matching by Observation for MultiRobot Cooperation* »
98. J. MOZARYN, J.E. KUREK- Article : Warsaw University of Technology, Institute of Automatic Control and Robotics.
« *Calculation of Model of the Robot by Neural Network with Robot Joint Distinction* »
99. Y. WU, T. S. HUANG – Article : IEEE int'l Conf. On Computer Vision, pp.606-611, Greece, 1999.
« *Capturing Articulated Human Hand Motion : A Divide-and-Conquer Approach* ».
100. D. KEE, G. WYETH – Article : School of Information Technology and Electrical Engineering, University of Queensland, Australia.
« *Cerebellar Joint Compensation for a Humanoid Robot* »
101. L. CHANGHONG, L. MINQIANG, K. JISONG : Article : IEEE International Conference on Artificial Intelligence systems (ICAIS02) 2002.
« *Cooperation Structure of Milti-Agent an Algorithms* »
102. E. ANSHELEVICH, S. OWENS, F. LAMIRAUX, L. E. KAVRAKI – Article : Department of Computer Science, Rice University, HOUSTON.
« *Deformable Volumes in Path Planning Application* »

103. A. CROSNIER – Article : Laboratoire d'informatique, de Robotique et de Micro-électronique de Montpellier LIRMM, 17/07/2009.
« *Détection de Collision pour la Robotique* »
104. J.P MERLET – Article : INRIA Sofia-Antipolis, France.
« *Determination of 6D Workspaces of Gough-Type parallel Manipulator and Comparison Between Different Geometries* »
105. D. NIKOVSKI, S. RAMAKRISHNA, R. K. KONERU, S. JAMHED, A. KONIJETI – Article : Southern Illinois University at Carbondale, May 12, 1994.
« *Distal Supervised Learning for solving Inverse Kinematic Problems* »
106. S. REGNIER, F. B. OUEZDOU, P. BIDAUD – Article : Laboratoire de Robotique de Paris, Université Paris 6, Octobre 1997.
« *Distributed Method for Inverse Kinematics of All serial Manipulators* »
107. M. M. BEN KHELIFA, M. BOUCHOUICHA – Article : Laboratoire SIS/AI – Université de Toulon.
« *Du Traitement de l'Information Visuelle à la Planification de Trajectoires : Application à la Robotique Mobile d'Assistance* »
108. A. JOUKHADAR – Rapport de recherche: INRIA N° 2546, Mai 95.
« *Dynamic Modeling system for Robotics applications* »
109. J. M. HOLLERBACH – Article : Massachusetts Institute of Technology – Artificial Intelligence Laboratory, A. I. Memo N° 700n, January 1983.
« *Dynamic Scaling of Manipulator Trajectories* ».
110. A. MCLEAN, S. CAMERON – Article : International Conf. On Advanced Robotics and Computer Vision, Singapore, 8th-12th November 1994.
« *Effective Path Planning and collision Avoidance for Redundant Manipulators* »
111. D. MANOCHA, J. CANNY – Article : IEEE Transaction on Robotics and Automation.
« *Efficient Inverse Kinematics for General 6R Manipulators* »
112. F. GIANNI, B. LENSEIGNE, P. DALLE – Article : IRIT, Université Paul Sabatier, Toulouse.
« *Estimation mono-vue de la posture du bras en utilisant un modèle biomécanique* »
113. B. LENSEIGNE, F. GIANNI, P. DALLE – Article : IRIT, Université Paul Sabatier, Toulouse.
« *Estimation mono-vue de la posture du bras , Méthode et évaluation* »
114. S. EL-KHOURY, A. SAHBANI et V. PERDEREAU – Article : Laboratoire des Instruments et Systèmes d'Ile-de-France, Université Pierre et Marie Curie, Paris 6.
« *Extraction d'axes de préhension d'un objet de forme quelconque pour l'apprentissage par imitation de la saisie humaine* ».
115. F. LAMIRAUX, J. P. LAUMOND – Article : LAAS-CNRS Toulouse – France.
« *Flatness and small time controllability of multibody mobile robots : application to motion planning* »
116. M. HAEGELE, J. NEUGEBAUER, R. D. SCHRAFT – Article : Proc. Of the 32nd ISR (International Symposium on Robotics), 19-21 April 2001.
« *From Robots to Robot Assistants* ».
117. Y. XU, M. C. NECHYBA – Article : The Robotics institute, Carnegie Mellon University 1993.
« *Fuzzy Inverse Kinematic Mapping : Rule generation, Efficiency, and Implementation* »
118. L. FLUCKIGER, L. PIGUET, C. BAUR – Article : SPIE Telemanipulator and Telepresence Technologies III, Vol 2901, pp 185-195, Boston, Nov. 1996.
« *Generic Robotic Kinematic Generator for virtual Environment Interfaces* »

119. F. BOYER – Article : Institut de Recherche en Cybernétique de Nantes.
« *Géométrie et dynamique en robotique* »
120. D. DEMERS, K. K. DELGADO – Article : Dept. Of Computer science & Engr.
« *Global Regularization of Inverse Kinematics for Redudant Manipulators* »
121. P. I. CORKE – Article : CSIRO Division of Manufacturing Technology, Australia
« *In situ Measurement of Robot Motor Electrical Constant* »
122. S. R. BUSS – Article : Department of Mathematics, University of California –
April 17, 2004.
« *Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse
and Damped Least Squares Methods* »
123. J. DE LOPE, R. G. CAREAGA, T. ZARRAONANDIA, D. MARAVALL – Article :
Dept. Of Artificial Intelligence, Faculty of Computer Science, Universidad
Politécnica de Madrid.
« *Inverse Kinematics for humanoid Robot Using Artificial Neural Networks* »
124. I. M. CHEN, G. YANG – Article : IEEE International Conf. On Robotics &
Automation Leuven, Belgium, May 1998.
« *Inverse Kinematics for Modular Reconfigurable Robots* »
125. E. OYAMA, N. Y. CHONG – Article : IEEE International Conf. On Robotics &
Automation Seoul, Korea, May 21-26, 2001.
« *Inverse Kinematics Learning by Modular Architecture Neural Networks with
Performance Prediction Networks* »
126. E. OYAMA¹, S. TACHI² – Article :
¹ : Mechanical Engineering Laboratory, Namiki 1-2, Tsukuba Science City
Ibaraki, Japan.
² : Faculty of Engineering, The University of Tokyo, Japan.
« *Inverse Kinematics Learning by modular Architecture Neural Networks* »
127. A. A. KHWAJA, M. O. RAHMAN, M. G. WAGNER – Article : Arizona State
University, USA.
« *Inverse Kinematics of Arbitrary Robotic Manipulators Using Genetic
Algorithms* »
128. D. PARK – Article : Computer Graphics, Department of Comuter Science,
University of Buenos Aires, Argentina.
« *Inverse Kinematics* »
129. R. KINDEL, D. HSU, J. C. LATOMBE, S. ROCK – Article : IEEE International
Conference on Robotics and Automation, 2000.
« *Kinodynamic Motion Planing Amidst Moving Obstacles* »
130. G. FAGES – Article : ROBAUT, J'automatise N°4 – Juin – Juillet 1999.
« *La Vérité sur les robots parallèles* ».
131. A. DEARDEN, Y. DEMIRIS – Article : Department of Electrical and Electronic
Engineering, Imperial College, London.
« *Learning Forward Models for Robots* »
132. D. DEMERS¹, K. K. DELGADO² – Article :
¹ : Comuter Science & Engineering, UC, San Diego.
² : Electrical & Computer Engineering, UC, San Diego.
« *Learning Global Direct Inverse Kinematics* »
133. S. ROWEIS – Article :
« *Levenberg Marquardt Optimization* »
134. L. GUILAMO¹, J. KUFNER^{1,2}, K. NISHIWAKI¹, S. KAGAMI¹ – Aricle :
¹ : The National Institute of Advanced Industrial Science and Technology, Tokyo,
Japan.
² : The Robotics Institute, Carnegie, Mellon University, USA.
« *Manipulability Optimization for Trajectory Generation* »

135. R. W. SUMNER, M. ZWICKER, C. GOTSMAN, J. POPOVIE – Article : SIGGRAPH 2005.
« *Mesh-Based Inverse Kinematics* »
136. A. MOUADIB – Article : GREYC – Université de Caen.
« *Modèles de décision de groupe pour la coordination et le contrôle d'un système multi-robots* ».
137. F. KHATOUNIAN, S. MOREAU, J.P. LOUIS, E. MONMASSON, F. LOUVEAU, J.M. ALEXANDRE – Article : Electrimacs 2005, Hammamet, Tunisie, April 17-20-2005.
« *Modeling and Simulation of a Hybrid Dynamic System used in Haptic Interfaces* »
138. B. BAYLE – Article : LAAS-CNRS, Toulouse.
« *Modélisation des manipulateurs mobiles à roues, en vue de leur commande cinématique* »
139. E. OYAMA¹, S. TACHI² – Article :
¹ : Mechanical Engineering Laboratory, Namiki 1-2, Tsukuba Science City Ibaraki, Japan.
² : Faculty of Engineering, The University of Tokyo, Japan.
« *Modular Neural Net System for Inverse Kinematics Learning* »
140. A. HAIT, T. SIMEON – Article : IEEE, Int. Conf. On Intelligent Robots and Systems, Osaka, Japan, November 4-8, 1996.
« *Motion Planning on Rough Terrain for an Articulated Vehicle in Presence of Uncertainties* ».
141. F. HAO, J. P. MERLET – Article : ELSEVIER, Mechanism and Machine Theory 997, 16 August 2004.
« *Multi-Criteria Optimal design of Parallel Manipulators based on interval analysis* »
142. R. ALAMI – Article : LAAS-CNRS, France.
« *Multi-robot Cooperation : Architectures and Paradigms* »
143. S. LEROY, J. P. LAUMOND, T. SIMEON – Article : LAAS-CNRS, France.
« *Multuple Path Coordination for Mobile Robots : a Geometric Algorithm* »
144. K. ABDEL-MALEK, S. OTHMAN – Article : ELSEVIER, Computer-Aided Design 31 (1999) 567-583.
« *Multiple Sweeping using the Denavit-Hartenberg Representation Method* »
145. J. MOZARYN, J. E. KUREK – Article : Warsaw University of Technology, Institute of Automatic Control and Robotics, Warsaw, POLAND.
« *Neural Network Robot Model with Not-Inverted Inertia Matrix* »
146. L. L. WHITCOMB – Article : Department of Mechanical Engineering, The Johns Hopkins University.
« *Notes on the Dynamics of Rigid-Body Open Kinematics Chains* ».
147. J. P. LAUMOND, T. SIMEON – Article : LAAS-CNRS, Toulouse, France.
« *Notes on Visibility Roadmaps and Path Planning* »
148. D. HSU, L. E. KAVRAKI, J. C. LATOMBE, R. MOTWANI, S. SORKIN – Article : Stanford University, USA.
« *On Finding Narrow Passages with Probabilistic Roadmap Planners* ».
149. Y. KOGA, J. C. LATOMBE – Article : 1050-4729/94 , IEEE 1994.
« *On Multi-Arm Manipulation Planning* »
150. I. M. REKLEITIS, G. DUDEK – Article : Centre for Intelligent Machines, Mc Gill University, Monreal, Canada.
« *On the Positional Uncertainty of Multi-Robot Cooperative Localization* »
151. M. HADDAD, A. YOUSNADJ – Article : 16^{ème} Congrès Français de Mécanique, Nice 1-5 septembre 2003.
« *Optimisation de la puissance nécessaire à un manipulateur mobile* »

152. A. HAYASHI, B. J. KUIPERS – Article : Ninth National Conference on Artificial Intelligence (AAAI91), Cambridge 1991.
« *Path Planning for Highly Redundant Manipulators using a Continuous Model* »
153. D. HSU, J. C. LATOMBE, R. MOTWANI – Article : Department of Computer Science, Stanford University.
« *Path Planning in Expansive Configuration Spaces* »
154. R. BOHLIN, L. E. KAVRAKI – Article : ICRA 2000.
« *Path Planning Using Lazy PRM* »
155. K. ABDEL-MALEK, W. YU – Article : Department of Mechanical Engineering, University of IOWA.
« *Placement of Robot Manipulators to Maximize Dexterity* »
156. D. HSU, J. C. LATOMBE, S. SORKIN – Article : IEEE International Symposium on Assembly and Task Planning 1999.
« *Placing a Robot Manipulator Amid Obstacles for Optimized Execution* »
157. S. A. VELINSKY, R. MATHURIN, D. M. DUCROS – Article : Bulletin des laboratoires des pont et chaussées – 235 – Novembre – Décembre 2001 – Ref. 4385 – pp. 77-85.
« *Planification de la trajectoire optimale du bras robotisé d'une machine de réparation de fissures* »
158. M. BENOSMAN – Article : Journées des Jeunes Chercheurs en Robotique ; 12^{ème} édition, 3,4 Février 2000, Bourges.
« *Planification de trajectoire opérationnelle pour le mouvement d'un bras manipulateur flexible* »
159. L. E. KAVRAKI, P. SVESTSKA, J. C. LATOMBE, M. H. OVERMARS – Article : IEEE Trans. On Robotics and Automation, 12 (4), 566-580, 1996.
« *Probabilistic Roadmaps For Path Planning in High-Dimensional Configuration Spaces* »
160. L. E. KAVRAKI, J. C. LATOMBE – Article : Department of Computer Science, Rice University, Houston.
« *Probabilistic Roadmaps for Robot Path Planning* »
161. B. R. DONALD¹, P. XAVIER² – Article :
¹ : Department of Computer Science, Cornell University, Ithaca.
² : Sandia National Laboratories, Albuquerque.
« *Provably Good Approximation Algorithms for Optimal Kinodynamic Planning : Robots With Decoupled Dynamics Bounds* »
162. M. S. BRANICKY, S. M. LAVALLE, K. OLSON, L. YANG – Article : IEEE International Conference on Robotics and Automation 2001.
« *Quasi-Randomized Path Planning* »
163. J. BAJON – Article : Techniques de l'ingénieur R 7 749.
« *Vision 2D en robotique* »
164. I. ESTEBE – Article : Techniques de l'ingénieur R 7 735.
« *Simulation et programmation hors ligne en robotique* »
165. C. LAURGEAU – Article : Techniques de l'ingénieur R 7 750.
« *Perception géométrique tridimensionnelle en robotique* »
166. R. CAEN, M. LAJOIE-MAZENC, B. TRANNOY – Article : Techniques de l'ingénieur R 7 760.
« *Actionneurs en robotique* »
167. D. MANOCHA, J. F. CANNY – Article : Computer Science Division, Department of Electrical Engineering and Computer Science.
« *Real Time Inverse Kinematics for General 6R Manipulators* »
168. D. TOLANI, A. GOSWAMI, N. I. BADLER – Article : Graphical Models, May 30, 2000, pp 62, 353-388.
« *Real-Time Inverse Kinematics Techniques for Anthropomorphic Limbs* »

169. V. PERDEREAU, C. REMOND, M. DROUIN – Article : Laboratoire des Instruments et Systèmes d’Ile-de-France, Université Pierre et Marie Curie, France.
« *Reconfiguration de la prise d’un objet par une main articulée* »
170. L. MICLO – Article : Séminaire de probabilités (Starsbourg), tome 26 (1992), p. 47-60.
« *Recuit simulé sans potentiel sur un ensemble fini* »
171. J. BURLET, O. AYCARD, T. FRAICHARD – Article : Inria Rhône-Alpes & Gravir, France. 08.03.2004.
« *Robust Motion Planning using Markov Decision Processes and Quadtree Decomposition* »
172. M. LLIBRE, R. MAMPEY, J. P. CHRETIEN – Article : Congrès AFCET Automatique 83, BESANÇON 15-17 Novembre 1983.
« *Simulation de la Dynamique des Robots Manipulateurs Motorisés* »
173. A. JOUKHADAR – Rapport de recherche : INRIA N° 3072, Décembre 96.
« *Simulation Dynamique et Applications Robotiques* »
174. J. P. MERLET – Article : The International Journal of Robotics Research, Vol. 23, N° 3, March 2004, pp.221-235.
« *Solving the Forward Kinematics of a Gough-Type Parallel Manipulator with Interval Analysis* ».
175. J. DE LOPE, T. ZARRAONANDIA, R. G. CAREAGA, D. MARAVALL – Article : Department of Artificial Intelligence, Faculty of computer Science, Universidad Politécnica de Madrid.
« *Solving the Inverse Kinematics in Humanoid Robots : A Neuronal Approach* »
176. K. GROCHOW, S. L. MARTIN, A. HERTZMAN, Z. POPOVIE – Article : ACM Tran. On Graphics (Proc. SIGGRAPH 2004).
« *Style-Based Inverse Kinematics* ».
177. E. DELASSUS – Article : Annales Scientifiques de l’E.N.S. 3^e série, tome 33 (1916), p. 71-125.
« *Sur les mouvements holonomes dont les équations admettent des formes multiples de Lagrange* ».
178. D. VILLE – Article : Bulletin de la S.M.F., tome 37 (1909), p. 120-132.
« *Sur les systèmes non holonomes* »
179. A. DEQUIDT, D. BERNIER, E. VALDES – Article : 16^e Congrès Français de Mécanique, Nice, 1-5 Septembre 2003.
« *Sur une extension du paramétrage de Denavit & Hartenberg pour la description systématique des chaînes cinématiques* »
180. S. GUERRY, f. B. OUEZDOU, S. REGNIER – Article : 2nd International Conference on Integrated Design and Manufacturing in Mechanical Engineering, May 27-29, 1998, Compiègne, France.
« *Synthèse géométrique de manipulateurs sous contraintes cinématiques* »
181. K. DJATH – Article : Journées des Jeunes Chercheurs, 11^{ème} édition, 8-9 Avril 1999, Lausanne.
« *Système de Navigation d’un Robot Mobile par Sous-Objectifs* ».
182. Z. TANG, C. ZHOU, Z. SUN – Article : Proc. Of then 2003 IEEE, International Conf. On Robotics & Automation, Taipei, Taiwan, September 14-19, 2003.
« *Trajectory Planning for Smooth Transition of a Biped Robot* »
183. T. SIMEON, J. P. LAUMOND, C. NISSOUX – Article : LAAS-CNRS, Toulouse, France.
« *Visibility-Based Probabilistic Roadmap for Motion Planning* »
184. C. Y. E. WANG, W. K. TIMOSZYK, J. E. BOBROW – Article : Proc. of the 1999 IEEE, International Conf. On robotics & Automation, Detroit, Michigan, May 1999.
« *Weightlifting Motion Planning for a PUMA 762 Robot* »

185. J. M. HOLLERBACH, G. SAHAR – Article : Massachusetts Institute of Technology, Artificial Intelligence Laboratory, A.I. Memo N°. 717, April 1983.
« *Wrist-Partitioned Inverse Kinematic Accelerations and Manipulator Dynamics* »
186. S. LAHOUAR – Thèse de Doctorat : Université de Poitier – 30 janvier 2008.
« *Contribution à la Modélisation et à la Planification de Trajectoires de Robots Parallèles Evitant les Singularités et les Collisions* »
187. K. I. KIM, Y. F. ZHENG – Article : Proc. of the 1991 IEEE.
« *The Explicit Dynamic Model and Inertial Parameters of the PUMA 560 Arm* »
188. O. KHATIB – Article : JAPAN USA Symposium on Flexible Automation.
« *Constrained Motion and Redundancy in Robot Manipulator Control* »
189. D. C. RUSPINI, K. KOLAROV, O. KHATIB – Article : Proc. IEEE 1997.
« *Haptic Interaction in Virtual Environments* »
190. O. KHATIB – Article : The International Journal of Robotics Research, Vol. 5, N°. 1, Spring 1986.
« *Real Time Obstacle Avoidance for Manipulators and Mobile Robots* »
191. D. C. RUSPINI, K. KOLAROV, O. KHATIB – Article : Proc. IEEE 1997, International Conf. On Robotics and Automation, Sacramento, California – April 1991.
« *Unknown Load Distribution of Two Industrial Robots* »
192. W. S. LU, Q. H. MENG – Article : Department of Electrical and Computer Engineering, University of Victoria, Canada.
« *A Load Distribution Scheme for Multi-Arm Coordinating Robots* »
193. J. GRIFONE
« *Algèbre Linéaire* » - CEPAD 1990 – I.S.B.N. 285428.239.6

APPENDICE A
TRANSFORMATIONS HOMOGÈNES.

A.1. Coordonnées homogènes [1].

A.1.1. Représentation d'un point.

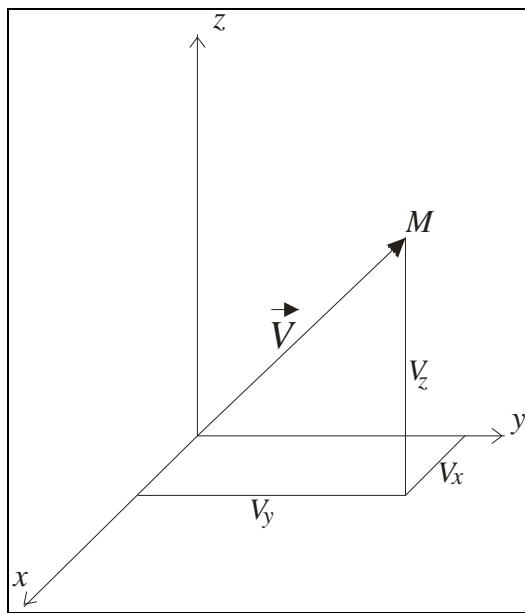


Figure A. 1 : Représentation d'un point dans l'espace.

En coordonnées homogènes, la représentation du vecteur position d'un point $M = (M_x, M_y, M_z)^t$ se fait par un vecteur colonne à quatre composantes. Les trois premières sont les coordonnées cartésiennes du point, tandis que la dernière composante est considérée comme un facteur d'échelle (égale à 1 pour notre cas).

$$V_M = \begin{bmatrix} M_x & M_y & M_z & 1 \end{bmatrix}^t \quad \text{---- (A. 1)}$$

A.1.2. Représentation d'une direction.

La représentation d'un vecteur libre se fait ainsi par un vecteur colonne de quatre composantes, mais la quatrième est nulle.

$$U = [U_x \quad U_y \quad U_z \quad 0]^T \quad \text{---- (A. 2)}$$

A.1.3. Représentation d'un plan.

Le plan Q défini par son équation cartésienne $\alpha x + \beta y + \gamma z + \delta = 0$ est représenté par le vecteur ligne V_Q comme suit :

$$V_Q = [\alpha \quad \beta \quad \gamma \quad \delta] \quad \text{---- (A. 3)}$$

Ainsi, pour qu'un point $M = (M_x, M_y, M_z)^T$ appartienne au plan Q il doit satisfaire la condition suivante :

$$V_Q \cdot V_M = 0 \Rightarrow [\alpha \quad \beta \quad \gamma \quad \delta] [M_x \quad M_y \quad M_z \quad 1]^T = 0 \quad \text{---- (A. 4)}$$

A.2. Transformations homogènes [1].

A.2.1. Matrice de transformation homogène.

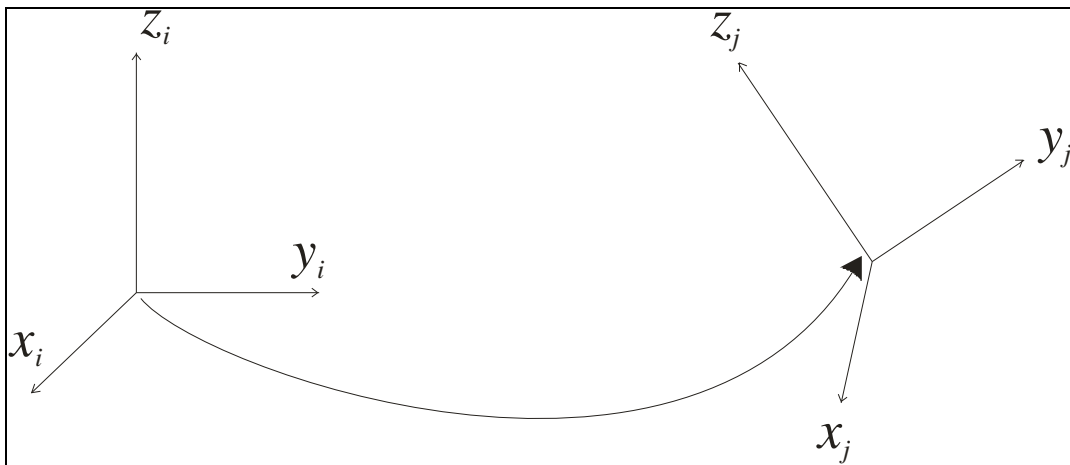


Figure A. 2 : Transformation homogène d'un repère.

La transformation qui ramène un repère R_i sur un repère R_j (comme montré dans la Figure A. 2 : Transformation homogène d'un repère.) est définie par la matrice ${}^i T_j$ appelée matrice de transformation homogène, de dimension (4x4) telle que :

$${}^i T_j = \begin{bmatrix} {}^i s_j & {}^i n_j & {}^i a_j & {}^i P_j \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----- (A. 5)}$$

Où ${}^i s_j$, ${}^i n_j$ et ${}^i a_j$ désignent respectivement les vecteurs unitaires suivant les axes x_j , y_j et z_j du repère R_j exprimés dans le repère R_i et ${}^i P_j$ est l'origine du repère R_j exprimée dans le repère R_i .

A.2.2. Transformations rigides [2].

Une transformation rigide est une transformation homogène qui préserve les distance entre les points du domaine exemple rotation, et translation.

A.2.2.1. Matrice de translation pure [1].

Soit $Trans(\vec{u})$ une translation définie par le vecteur \vec{u} , cette transformation est définie par la matrice :

$${}^i T_j = Trans(\vec{u}) = \begin{bmatrix} 1 & 0 & 0 & u_x \\ 0 & 1 & 0 & u_y \\ 0 & 0 & 1 & u_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----- (A. 6)}$$

Une translation orientée par un vecteur \vec{u} d'une valeur d est définie par :

$${}^i T_j = Trans(\vec{u}, d) = Trans\left(\left(\frac{d}{|\vec{u}|}\right)\vec{u}\right) = \begin{bmatrix} 1 & 0 & 0 & \left(\frac{d}{|\vec{u}|}\right)u_x \\ 0 & 1 & 0 & \left(\frac{d}{|\vec{u}|}\right)u_y \\ 0 & 0 & 1 & \left(\frac{d}{|\vec{u}|}\right)u_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----- (A. 7)}$$

Une translation $Trans(\vec{u})$ peut être décomposée en un produit de trois translations :

$$Trans(\vec{u}) = Trans(\vec{i}, u_x) Trans(\vec{j}, u_y) Trans(\vec{k}, u_z) \quad \text{----- (A. 8)}$$

A.2.2.2. Matrices de rotation autour des axes principaux [1].

Soit $Rot(s, \theta)$ une rotation d'un angle θ autour de l'axe s ; ou l'axe s représente l'un des axes principaux du repère initial R_i . le développement de l'équation (A.5) donne le tableau suivant :

Tableau A. 1 : Matrices de rotation autour des axes principaux.

Axe de rotation	Matrice de transformation
x	${}^i T_j = Rot(x, \theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\theta & -S\theta & 0 \\ 0 & S\theta & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} & & & 0 \\ & A(x, \theta) & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
y	${}^i T_j = Rot(y, \theta) = \begin{bmatrix} C\theta & 0 & S\theta & 0 \\ 0 & 1 & 0 & 0 \\ -S\theta & 0 & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} & & & 0 \\ & A(y, \theta) & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
z	${}^i T_j = Rot(z, \theta) = \begin{bmatrix} C\theta & -S\theta & 0 & 0 \\ S\theta & C\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} & & & 0 \\ & A(z, \theta) & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

A.2.2.3. Matrice de rotation autour d'un axe quelconque passant par l'origine [1].

Considérons une rotation $Rot(u, \theta)$ autour d'un axe portant un vecteur unitaire $u = [u_x, u_y, u_z]^T$ quelconque et passant par l'origine.

La matrice de transformation est donnée par la relation suivante :

$${}^i T_j = Rot(u, \theta) = \begin{bmatrix} u_x^2(1-C\theta)+C\theta & u_x u_y(1-C\theta)-u_z S\theta & u_x u_z(1-C\theta)+u_y S\theta & 0 \\ u_x u_y(1-C\theta)+u_z S\theta & u_y^2(1-C\theta)+C\theta & u_y u_z(1-C\theta)-u_x S\theta & 0 \\ u_x u_z(1-C\theta)-u_y S\theta & u_y u_z(1-C\theta)+u_x S\theta & u_z^2(1-C\theta)+C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{--- (A. 9)}$$

Ou encore :

$$\left\{ \begin{array}{l} \begin{bmatrix} 0 \\ A(u, \theta) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \\ A(u, \theta) = u.u^T(1 - C\theta) + I_3 C\theta + \hat{u}.S\theta \\ \hat{u} = \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix} \end{array} \right. \quad \text{---- (A. 10)}$$

A.2.3. Propriétés des matrices de transformations rigides.

- Une matrice de transformation rigide peut se mettre sous la forme

$${}^i T_j = \begin{bmatrix} {}^i s_j & {}^i n_j & {}^i a_j & {}^i P_j \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A & P \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (A. 11)}$$

La matrice A représente la rotation, alors que la matrice colonne P représente la translation.

- La matrice A est orthogonale $A^{-1} = A^T$;
- L'inverse de la matrice ${}^i T_j$ définit la matrice ${}^j T_i$;
- L'inverse d'une matrice de transformation donnée par la relation (A.5), peut être mise sous la forme :

$$\left({}^i T_j\right)^{-1} = {}^j T_i = \begin{bmatrix} A^T & -s^T.P_x \\ -n^T.P_y \\ -a^T.P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A^T & -A^T.P \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (A. 12)}$$

- Si un repère R_0 a subi k transformations consécutives, alors la transformation ${}^0 T_k$ peut être déduite par :

$${}^0 T_k = {}^0 T_1 \cdot {}^1 T_2 \cdot {}^2 T_3 \cdot \dots \cdot {}^{k-1} T_k \quad \text{---- (A. 13)}$$

A.2.4. Axe et angle de rotation équivalents à une transformation de rotation donnée [1].

Soit T une matrice de transformation de rotation pure telle que :

$$T = \begin{bmatrix} s_x & n_x & a_x & 0 \\ s_y & n_y & a_y & 0 \\ s_z & n_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{---- (A. 14)}$$

Pour avoir les termes u et θ de l'expression $T = Rot(u, \theta)$, on applique :

$$C\theta = \frac{1}{2}(s_x + n_y + a_z - 1) \quad \text{---- (A. 15)}$$

$$S\theta = \frac{1}{2}\sqrt{(n_z - a_y)^2 + (a_x - s_z)^2 + (s_y - n_x)^2} \quad \text{---- (A. 16)}$$

Remarque.

De la relation (A.16), on remarque que le signe de $S\theta$ est positif, donc θ sera forcément dans l'intervalle $[0, \pi]$, d'où on peut en déduire l'insuffisance de la méthode, et la complexité du problème de géométrie inverse.

Et ensuite, on peut avoir les composantes du vecteur u par les relations suivantes

$$\begin{cases} u_x = Sgn(n_z - a_y) \sqrt{\frac{s_x - C\theta}{1 - C\theta}} \\ u_y = Sgn(a_z - s_z) \sqrt{\frac{n_y - C\theta}{1 - C\theta}} \\ u_z = Sgn(s_y - n_x) \sqrt{\frac{a_z - C\theta}{1 - C\theta}} \end{cases} \quad \text{---- (A. 17)}$$

A.2.5. Matrices de transformation entre torseurs.

A.2.5.1. Torseur cinématique.

Considérons le point O_i du corps C_i . On appelle éléments de réduction du torseur cinématique en O_i :

V_i : la vitesse absolue de l'origine O_i par rapport à un repère R_0 telle que

$$V_i = \frac{d}{dt}(O_0O_i) ;$$

ω_i : le vecteur de rotation instantanée du corps C_i par rapport à R_0 .

La connaissance de ces deux éléments permet de calculer la vitesse d'un point O_j par la relation fondamentale suivante :

$$V_j = V_i + \omega \times O_i O_j \quad \text{----- (A. 18)}$$

On définit le vecteur $\mathbb{V}_i = \begin{bmatrix} V_i \\ \omega_i \end{bmatrix}$ appelé vecteur du torseur cinématique en O_i .

Si on veut calculer le torseur cinématique dans le repère R_j , on procède comme suit :

$L_{i,j}$ étant le vecteur d'origine O_i et d'extrémité O_j , de la relation (A.18) on a :

$$\begin{bmatrix} V_j \\ \omega_j \end{bmatrix} = \begin{bmatrix} I_3 & -\hat{L}_{i,j} \\ 0_3 & I_3 \end{bmatrix} \begin{bmatrix} V_i \\ \omega_i \end{bmatrix} \quad \text{----- (A. 19)}$$

Une projection dans le repère R_i donne :

$$\begin{bmatrix} {}^i V_j \\ {}^i \omega_j \end{bmatrix} = \begin{bmatrix} I_3 & -{}^i \hat{P}_j \\ 0_3 & I_3 \end{bmatrix} \begin{bmatrix} {}^i V_i \\ {}^i \omega_i \end{bmatrix} \quad \text{----- (A. 20)}$$

En exprimant cette relation dans le repère R_j , on obtient :

$$\begin{bmatrix} {}^j V_j \\ {}^j \omega_j \end{bmatrix} = \begin{bmatrix} {}^j A_i & 0_3 \\ 0_3 & {}^j A_i \end{bmatrix} \begin{bmatrix} I_3 & -{}^i \hat{P}_j \\ 0_3 & I_3 \end{bmatrix} \begin{bmatrix} {}^i V_i \\ {}^i \omega_i \end{bmatrix} = \begin{bmatrix} {}^j A_i & -{}^j A_i \cdot {}^i \hat{P}_j \\ 0_3 & {}^j A_i \end{bmatrix} \begin{bmatrix} {}^i V_i \\ {}^i \omega_i \end{bmatrix} \quad \text{----- (A. 21)}$$

Ou encore :

$${}^j \mathbb{V}_j = {}^j \mathbb{T}_i \cdot {}^i \mathbb{V}_i \quad \text{----- (A. 22)}$$

Avec

$${}^j \mathbb{T}_i = \begin{bmatrix} {}^j A_i & -{}^j A_i \cdot {}^i \hat{P}_j \\ 0_3 & {}^j A_i \end{bmatrix} \quad \text{----- (A. 23)}$$

A.2.5.2. Torseur dynamique

Un effort en O_i est représenté par un torseur dynamique $\mathbf{f}_i = \begin{bmatrix} \mathbf{f}_i \\ \mathbf{m}_i \end{bmatrix}$ où \mathbf{f}_i et \mathbf{m}_i représentent respectivement la résultante des forces et le moment autour de O_i .

Étant donné un effort ${}^i \mathbf{f}_i$ exprimé dans un repère R_i , pour calculer l'effort équivalent exprimé dans un repère R_j , on applique la transformation suivante :

$${}^j \mathbf{f}_i = \begin{bmatrix} {}^j \mathbf{f}_i \\ {}^j \mathbf{m}_i \end{bmatrix} = {}^j \mathbb{T}_i \begin{bmatrix} {}^i \mathbf{f}_i \\ {}^i \mathbf{m}_i \end{bmatrix} \quad \text{----- (A. 24)}$$

Ou encore :

$${}^j \mathbf{f}_i = {}^j A_i \cdot {}^i \mathbf{f}_i \quad \text{----- (A. 25)}$$

$${}^j \mathbf{m}_i = {}^j A_i \cdot ({}^i \mathbf{f}_i \times {}^j P_i + {}^i \mathbf{m}_i) \quad \text{----- (A. 26)}$$

APPENDICE B

DÉRIVATION NUMÉRIQUE & OBTENTION DES VITESSES ET ACCÉLÉRATIONS.

B.1. Description du problème.

On considère une fonction f dont on ne connaît pas l'expression analytique. Étant données les $(n+1)$ valeurs $y_i = f(t_i)$ prises par la fonction f aux points de collocation t_i , on souhaite évaluer les dérivées de cette fonction en ces mêmes points t_i .

B.2. Approximation de la dérivée première.

B.2.1. Approche.

Soit $f : [a, b] \rightarrow \mathbb{R}$ une fonction continue ; on définit le pas e discrétisation $h = \frac{b-a}{n}$; $n \in \mathbb{N}$ et on suppose que l'on connaît $f(a)$, $f(a+h)$, $f(a+2h)$,, $f(a+i.h)$,, $f(b)$.

On rappelle la définition de la dérivée :

$$f'(t) = \lim_{h \rightarrow 0} \frac{f(t+h) - f(t)}{h} \quad \text{----- (B. 1)}$$

soit au point t_i :

$$f'(t_i) = \lim_{h \rightarrow 0} \frac{f(t_i+h) - f(t_i)}{h} \quad \text{----- (B. 2)}$$

Le développement de la fonction f en série de Taylor au voisinage d'un point t est donné par :

$$f(t+h) = f(t) + h.f'(t) + \frac{h^2}{2} f''(t) + \frac{h^3}{3!} f'''(t) + \dots \quad \text{----- (B. 3)}$$

Soit à l'ordre 1 :

$$f(t+h) = f(t) + h.f'(t) + o(h) \quad \text{----- (B. 4)}$$

D'où l'on obtient la première approche de la dérivée au point t :

$$f'(t) = \frac{f(t+h) - f(t)}{h} + o(h) \quad \text{----- (B. 5)}$$

B.2.2. Schémas numériques.

Il existe trois méthodes de calcul approché de la dérivée première :

B.2.2.1. Schéma aux différences ascendantes.

$$\begin{cases} f_i = f(t_i) \\ f_{i+1} = f(t_i + h) \end{cases} \quad \text{----- (B. 6)}$$

$$f'(t_i) = \frac{f_{i+1} - f_i}{h} + o(h) \quad \text{----- (B. 7)}$$

Remarque : On ne peut pas calculer la dérivée au point b .

B.2.2.2. Schéma aux différences descendantes.

$$\begin{cases} f_i = f(t_i) \\ f_{i-1} = f(t_i - h) \end{cases} \quad \text{----- (B. 8)}$$

$$f'(t_i) = \frac{f_i - f_{i-1}}{h} + o(h) \quad \text{----- (B. 9)}$$

Remarque : On ne peut pas calculer la dérivée au point a .

B.2.2.3. Schéma aux différences centrées.

$$\begin{cases} f_{i-1} = f(t_i - h) \\ f_{i+1} = f(t_i + h) \end{cases} \quad \text{----- (B. 10)}$$

$$f'(t_i) = \frac{f_{i+1} - f_{i-1}}{2h} + o(h^2) \quad \text{----- (B. 11)}$$

Remarque : On ne peut pas calculer la dérivée aux points a et b .

B.3. Approximation de la dérivée d'ordre 2.

B.3.1. Schéma aux différences ascendantes.

$$\begin{cases} f_i = f(t_i) \\ f_{i+1} = f(t_i + h) \\ f_{i+2} = f(t_i + 2.h) \end{cases} \quad \text{---- (B. 12)}$$

$$f''(t_i) = \frac{f_{i+2} - 2f_{i+1} + f_i}{h^2} + o(h) \quad \text{---- (B. 13)}$$

B.3.2. Schéma aux différences descendantes.

$$\begin{cases} f_i = f(t_i) \\ f_{i-1} = f(t_i - h) \\ f_{i-2} = f(t_i - 2.h) \end{cases} \quad \text{---- (B. 14)}$$

$$f''(t_i) = \frac{f_i - 2f_{i-1} + f_{i-2}}{h^2} + o(h) \quad \text{---- (B. 15)}$$

B.3.3. Schéma aux différences centrées.

$$\begin{cases} f_{i-1} = f(t_i - h) \\ f_i = f(t_i) \\ f_{i+1} = f(t_i + h) \end{cases} \quad \text{---- (B. 16)}$$

$$f''(t_i) = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + o(h^2) \quad \text{---- (B. 17)}$$

B.4. Application au calcul des vitesses et accélérations.

On considère un mouvement donné par $(n+1)$ positions y_i aux instants correspondants t_i .
Le mouvement démarre du repos donc et s'arrête à la fin donc :

$$\dot{y}_0 = \ddot{y}_0 = \dot{y}_n = \ddot{y}_n \quad \text{---- (B. 18)}$$

Pour une précision optimale on choisi les différences centrées du fait que les valeurs cherchées sont préalablement connues aux instants extrêmes t_0 et t_n .

Le programme suivant (sous MATLAB) nous calculera les vitesses et les accélérations pour un mouvement donné.


```

%Programme permettant de calculer les vitesses V et les accélérations A
%pour un mouvement donné par un tableau P de n lignes corespondants aux
%différents instants discrets et deux colonnes la première pour ti et la
%seconde pour les positions yi
[n,m]=size(P);
V=zeros(n,1);
A=zeros(n,1);
nombre=n-1;
h=P(2,1)-P(1,1);
%Calcul des vitesses
for i=2:nombre
    f1=P(i-1,2);
    f2=P(i+1,2);
    V(i)=(f2-f1)/(2*h);
end
%Calcul des accélérations
for i=2:nombre
    f1=P(i-1,2);
    f2=P(i,2);
    f3=P(i+1,2);
    V(i)=(f1-2*f2+f3)/(h^2);
end

```

APPENDICE C

INTERSECTION GÉOMÉTRIQUE ENTRE DEUX CERCLES.

C.1. Description du problème.

Étant donné deux cercles de centres $C_1 \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$ et $C_2 \begin{pmatrix} x_2 \\ y_2 \end{pmatrix}$ et de rayons respectivement R_1 et R_2 , on veut élaborer une procédure informatique qui analyse l'intersection entre les deux cercles, et donne le point d'intersection le plus proche à un point donné $P_0 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$ si l'intersection a lieu dans deux points.

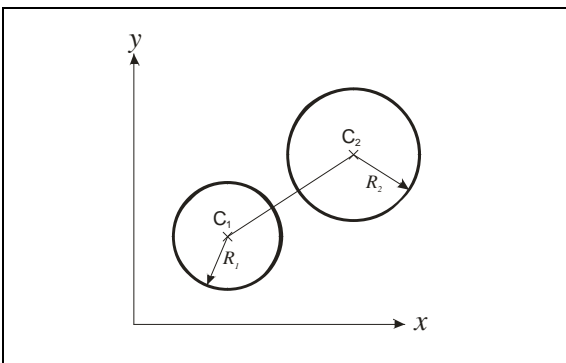
C.2. Analyse géométrique.

L'analyse géométrique commence par l'évaluation de la longueur du segment $[C_1C_2]$.

$$\| \overrightarrow{C_1C_2} \| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad \text{---- (C. 1)}$$

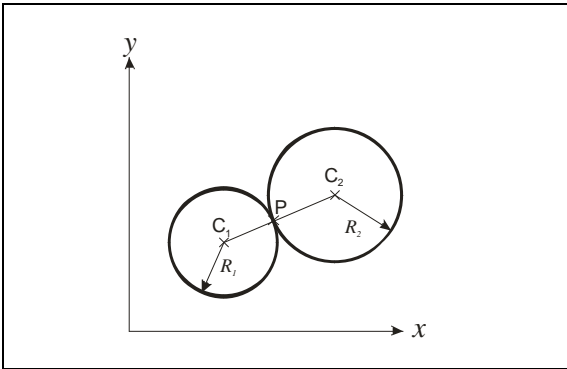
On distingue les trois cas suivants :

a. Premier cas : $\| \overrightarrow{C_1C_2} \| > R_1 + R_2$



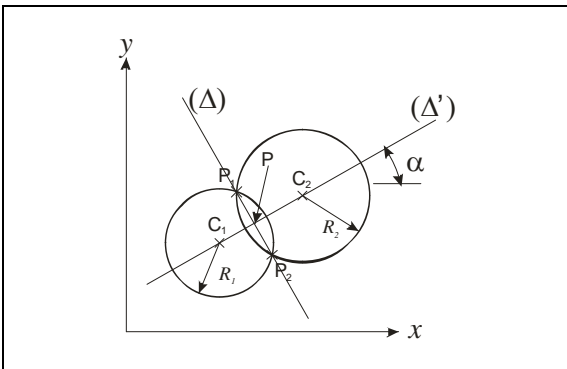
Dans ce cas l'intersection entre les deux cercles est vide.

b. Deuxième cas : $\|C_1 C_2\| = R_1 + R_2$



Dans ce cas l'intersection entre les deux cercles comporte un seul point P .

c. Troisième cas : $\|C_1 C_2\| < R_1 + R_2$



Dans ce cas l'intersection entre les deux cercles comporte les deux points P_1 et P_2 .

Pour le deuxième et le troisième cas, la droite (Δ) passe par les points d'intersection cherchés : cette droite est déterminée de la manière suivante :

Les équations cartésiennes des deux cercles :

$$(x - x_1)^2 + (y - y_1)^2 = R_1^2 \Rightarrow x^2 + y^2 - 2(xx_1 + yy_1) = R_1^2 - (x_1^2 + y_1^2) \quad \text{---- (C. 2)}$$

$$(x - x_2)^2 + (y - y_2)^2 = R_2^2 \Rightarrow x^2 + y^2 - 2(xx_2 + yy_2) = R_2^2 - (x_2^2 + y_2^2) \quad \text{---- (C. 3)}$$

(C.2)-(C.3) donne :

$$-2(xx_1 + yy_1) + 2(xx_2 + yy_2) = R_1^2 - R_2^2 - (x_1^2 + y_1^2) + (x_2^2 + y_2^2) \quad \text{---- (C. 4)}$$

Ou encore

$$-2(x_1 - x_2)x - 2(y_1 - y_2)y = R_1^2 - R_2^2 - (x_1^2 + y_1^2) + (x_2^2 + y_2^2) \quad \text{---- (C. 5)}$$

L'équation (C.5) représente l'équation cartésienne de la droite (Δ) .

On détermine ainsi l'équation cartésienne de la droite (Δ') qui passe par les deux points C_1 et C_2 .

On distingue les trois cas suivant :

- Si $(C_1)_x = (C_2)_x$ alors la droite (Δ') est verticale, son équation est la suivante :
 $x = (C_1)_x$;
- Si $(C_1)_y = (C_2)_y$ alors la droite (Δ') est horizontale, son équation est la suivante :
 $y = (C_1)_y$;
- Si $(C_1)_x \neq (C_2)_x$ et $(C_1)_y \neq (C_2)_y$, alors la droite est oblique, on détermine son équation de la manière suivante :

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1} \Rightarrow (y_2 - y_1)(x - x_1) = (x_2 - x_1)(y - y_1) \quad \text{---- (C. 6)}$$

$$\Rightarrow (y_2 - y_1)x - (x_2 - x_1)y = (y_2 - y_1)x_1 - (x_2 - x_1)y_1 \quad \text{---- (C. 7)}$$

Dans les trois cas, l'équation de la droite (Δ') est :

$$ax + by = c \quad \text{---- (C. 8)}$$

Avec les coefficients a , b et c connus.

Le point P est l'intersection des deux droites (Δ) et (Δ') . Cette intersection est déterminée par le système suivant :

$$\begin{cases} -2(x_1 - x_2)x - 2(y_1 - y_2)y = R_1^2 - R_2^2 - (x_1^2 + y_1^2) + (x_2^2 + y_2^2) \\ ax + by = c \end{cases} \quad \text{---- (C. 9)}$$

Ou encore :

$$\begin{pmatrix} -2(x_1 - x_2) & -2(y_1 - y_2) \\ a & b \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} R_1^2 - R_2^2 - (x_1^2 + y_1^2) + (x_2^2 + y_2^2) \\ c \end{pmatrix} \quad \text{---- (C. 10)}$$

Et on aura :

$$\begin{pmatrix} x_P \\ y_P \end{pmatrix} = \begin{pmatrix} -2(x_1 - x_2) & -2(y_1 - y_2) \\ a & b \end{pmatrix}^{-1} \begin{pmatrix} R_1^2 - R_2^2 - (x_1^2 + y_1^2) + (x_2^2 + y_2^2) \\ c \end{pmatrix} \quad \text{---- (C. 11)}$$

Le point P représente l'intersection des deux cercles dans le cas où $\|\overrightarrow{C_1 C_2}\| = R_1 + R_2$.

Pour le cas où $\|\overrightarrow{C_1 C_2}\| < R_1 + R_2$, on doit trouver les deux points P_1 et P_2 .

L'angle α est déterminé par le système suivant :

$$\begin{cases} \cos \alpha = \frac{(\overrightarrow{C_1 C_2})_x}{C_1 C_2} = \frac{x_2 - x_1}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \\ \sin \alpha = \frac{(\overrightarrow{C_1 C_2})_y}{C_1 C_2} = \frac{y_2 - y_1}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \end{cases} \quad \text{---- (C. 12)}$$

On considère le triangle $C_1\widehat{P}P_1$ droit en P .

On applique le théorème de Pythagore :

$$\overline{C_1P_1}^2 = \overline{C_1P}^2 + \overline{PP_1}^2 \quad \text{----- (C. 13)}$$

$$\Rightarrow \overline{PP_1} = \sqrt{R_1^2 - \overline{C_1P}^2} \quad \text{----- (C. 14)}$$

De la figure, on remarque que l'angle que fait le vecteur $\overrightarrow{PP_1}$ avec l'horizontale est égale à $\alpha + \frac{\pi}{2}$ et par conséquent :

$$\overrightarrow{PP_1} = \overline{PP_1} \begin{pmatrix} \cos\left(\alpha + \frac{\pi}{2}\right) \\ \sin\left(\alpha + \frac{\pi}{2}\right) \end{pmatrix} = \overline{PP_1} \begin{pmatrix} -\sin \alpha \\ \cos \alpha \end{pmatrix} \quad \text{----- (C. 15)}$$

Et on trouve le point P_1 :

$$\begin{cases} (P_1)_x = P_x + (\overrightarrow{PP_1})_x = -\overline{PP_1} \cdot \sin \alpha + P_x \\ (P_1)_y = P_y + (\overrightarrow{PP_1})_y = \overline{PP_1} \cdot \cos \alpha + P_y \end{cases} \quad \text{----- (C. 16)}$$

Le point P_2 représente le point symétrique de P_1 par rapport au point P , on a donc :

$$\overrightarrow{PP_2} = -\overrightarrow{PP_1} = \overline{PP_1} \begin{pmatrix} \sin \alpha \\ -\cos \alpha \end{pmatrix} \quad \text{----- (C. 17)}$$

Et finalement le point P_2 :

$$\begin{cases} (P_2)_x = P_x + (\overrightarrow{PP_2})_x = \overline{PP_1} \cdot \sin \alpha + P_x \\ (P_2)_y = P_y + (\overrightarrow{PP_2})_y = -\overline{PP_1} \cdot \cos \alpha + P_y \end{cases} \quad \text{----- (C. 18)}$$

C.3. Choix du point le plus proche de P_0 .

Dans le cas où le système comporte deux solutions possibles P_1 et P_2 , on choisit l'un des deux points en cherchant parmi les deux vecteurs $\overrightarrow{P_0P_1}$ et $\overrightarrow{P_0P_2}$, celui qui possède le plus petit module.

C.4. Numérisation.

Le programme suivant écrit en langage MATLAB, résout le problème posé en trouvant l'intersection entre les deux cercles (C_1, R_1) et (C_2, R_2) et renvoi le point d'intersection le plus proche à P_0 ;

```
function y=Inter_Cercles(C1,R1,C2,R2,P0)
x1=C1(1);
y1=C1(2);
x2=C2(1);
y2=C2(2);
MC1C2=sqrt((x2-x1)^2+(y2-y1)^2);
R=R1+R2;%La somme des deux rayons
if MC1C2<=R
    if x1==x2
        a=1;
        b=0;
        c=x1;
    elseif y1==y2
        a=0;
        b=1;
        c=y1;
    else
        a=y2-y1;
        b=x1-x2;
        c=(y2-y1)*x1-(x2-x1)*y1;
    end
    A=[-2*(x1-x2),-2*(y1-y2);a,b];
    V=[R1^2-R2^2-x1^2-y1^2+x2^2+y2^2;c];
    P=inv(A)*V;
    if MC1C2==R
        y=P
    else
        C=(x2-x1)/MC1C2;
        S=(y2-y1)/MC1C2;
        C1P=P-C1;
        MPP1=sqrt(R1^2-(norm(C1P))^2);
        PP1=MPP1*[-S;C];
        P1=PP1+P;
        PP2=-PP1;
        P2=PP2+P;
        V1=P1-P0;
        MV1=norm(V1);
        V2=P2-P0;
        MV2=norm(V2);
        if MV1<=MV2
            y=P1;
        else
            y=P2;
        end
    end
end
end
```

APPENDICE D

DÉTERMINATION DES PARAMÈTRES CINÉMATIQUES D'UN MOUVEMENT SUIVANT UNE LOI BANG-BANG.

D.1. Description du problème.

Étant donné un actuateur correspondant à une articulation donnée. Cette dernière est représentée par une variable généralisée X . On veut que cet actuateur génère un mouvement menant le lien correspondant de la position initiale X_i à la position finale X_f . On suppose que le mouvement suit une loi Bang-Bang divisant le mouvement en deux ou trois phases successives.

Pour un mouvement en deux phases, on distingue les phases suivantes :

- Phase d'accélération : le mouvement démarre d'une vitesse initiale nulle pour atteindre une vitesse maximale V_{max} donnée avec une accélération a_1 donnée, le temps nécessaire pour effectuer ce mouvement est t_1 ;
- Phase de décélération : mouvement décéléré à une accélération a_2 pour atteindre une vitesse nulle donc l'arrêt du lien pendant une durée t_3 .

Pour un mouvement en trois phases, on ajoute entre les deux phases citées, une phase intermédiaire, la où le mouvement se fait à une vitesse constante V_{max} pendant une durée t_2 .

Le signe de $(X_f - X_i)$ déterminera les signes des deux accélérations a_1 et a_2 , comme suit :

- Si $X_f > X_i$ alors :

$$\begin{cases} V_{max} > 0 \\ a_1 > 0 \\ a_2 < 0 \end{cases} \quad \text{---- (D. 1)}$$

- Si $X_f < X_i$ alors :

$$\begin{cases} V_{\max} < 0 \\ a_1 < 0 \\ a_2 > 0 \end{cases} \quad \text{---- (D. 2)}$$

D.2. Problème 1 : Détermination des paramètres cinématiques pour un mouvement en deux phases.

Le problème est posé de la manière suivante : étant donné t_0 , a_1 , a_2 , X_i et X_f trouver les équations horaires d'un mouvement en deux phases menant la position X de X_i vers X_f .

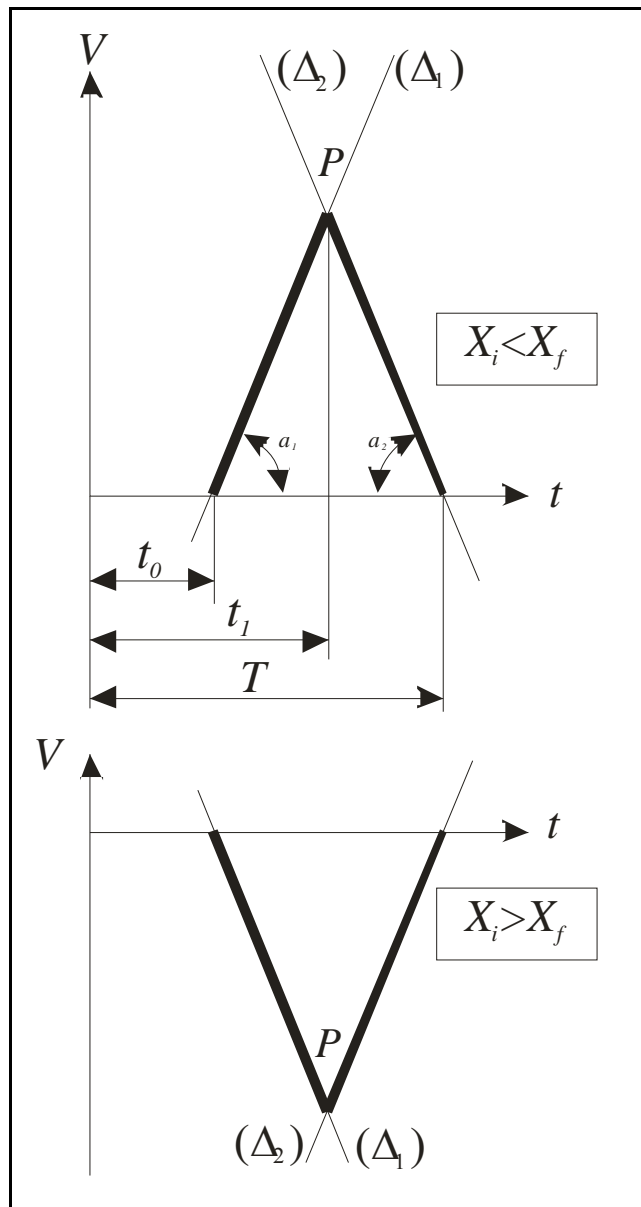


Figure D. 1 : Description d'un mouvement suivant une loi Bang-Bang en deux phases.

Le problème peut être résolu en trouvant le point P dans le plan (t, V) comme montré sur la figure (D.1), ou encore trouver l'instant t_1 où le mouvement change de phase de l'accélération vers le freinage.

On suppose que t_1 est connu, et on exprime la relation entre t_1 et $\Delta X = X_f - X_i$.

(Δ_1) est une droite dont l'équation cartésienne est la suivante :

$$V = a_1 \cdot (t - t_0) \quad \text{----- (D. 3)}$$

Le point P aura donc les coordonnées suivantes :

$$P = \begin{pmatrix} t_1 \\ a_1 \cdot (t_1 - t_0) \end{pmatrix} \quad \text{----- (D. 4)}$$

(Δ_2) est une droite dont l'équation cartésienne est la suivante :

$$V = a_2 \cdot t + (a_1 - a_2)t_1 - a_1 \cdot t_0 \quad \text{----- (D. 5)}$$

On peut déterminer la valeur de T (intersection de la droite (Δ_2) avec l'axe des abscisses) :

$$T = \frac{(a_2 - a_1)t_1 + a_1 \cdot t_0}{a_2} \quad \text{----- (D. 6)}$$

L'aire délimitée par les deux droites (Δ_1) , (Δ_2) ainsi que l'axe des abscisses détermine le déplacement effectué durant les deux phases.

$$|\Delta X| = \left| \frac{1}{2} \cdot (T - t_0) \cdot a_1 \cdot (t_1 - t_0) \right| = \left| \frac{1}{2} \cdot \left(\frac{(a_2 - a_1)t_1 + a_1 \cdot t_0}{a_2} - t_0 \right) \cdot a_1 \cdot (t_1 - t_0) \right| \quad \text{----- (D. 7)}$$

$$|\Delta X| = \left| \frac{1}{2a_2} \cdot ((a_2 - a_1)t_1 + a_1 \cdot t_0 - a_2 \cdot t_0) \cdot a_1 \cdot (t_1 - t_0) \right| \quad \text{----- (D. 8)}$$

$$|\Delta X| = \left| \frac{1}{2a_2} \cdot ((a_2 - a_1)t_1 + (a_1 - a_2)t_0) \cdot a_1 \cdot (t_1 - t_0) \right| \quad \text{----- (D. 9)}$$

$$|\Delta X| = \frac{1}{2} \cdot \left| \frac{a_1(a_2 - a_1)}{a_2} \right| \cdot (t_1 - t_0)^2 \quad \text{----- (D. 10)}$$

Connaissant $\Delta X = X_f - X_i$ (données du problème), on déterminera la valeur de t_1 :

$$t_1 = \sqrt{\frac{2\Delta X \cdot a_2}{a_1 \cdot a_2 - a_1^2}} + t_0 \quad \text{----- (D. 11)}$$

On utilise la fonction « *Parabole* » décrite dans l'annexe E pour trouver les équations horaires du déplacement dans les différentes phases. On admet que pour les deux phases, le déplacement est une fonction polynomiale de degré 2.

Phase 1 : $t \in [t_0, t_1]$

$$\begin{cases} X(t_0) = X_i \\ \left[\frac{dX}{dt} \right]_{t=t_0} = 0 \\ \left[\frac{dX}{dt} \right]_{t=t_1} = V_1 = a_1 \cdot (t_1 - t_0) \end{cases} \quad \text{---- (D. 12)}$$

Le déplacement est donc déterminé ainsi par la fonction Parabole comme suit :

$$X = \text{Parabole}(0, V_1, t_0, t_1, X_i) \quad \text{---- (D. 13)}$$

On détermine la position finale de la phase 1, X_1 en évaluant la fonction Position trouvée en (D.13) à l'instant t_1 .

Phase 2 : $t \in [t_1, T]$

$$\begin{cases} X(t_1) = X_1 \\ \left[\frac{dX}{dt} \right]_{t=t_1} = V_1 = a_1 \cdot (t_1 - t_0) \\ \left[\frac{dX}{dt} \right]_{t=T} = 0 \end{cases} \quad \text{---- (D. 14)}$$

Le déplacement est donc déterminé ainsi par la fonction Parabole comme suit :

$$X = \text{Parabole}(V_1, 0, t_1, T, X_1) \quad \text{---- (D. 15)}$$

Le mouvement est donc entièrement déterminé par les trois systèmes suivants :

$$\begin{cases} \ddot{X} = 0 & \text{si } 0 \leq t < t_0 \\ \ddot{X} = a_1 & \text{si } t_0 \leq t < t_1 \\ \ddot{X} = a_2 & \text{si } t_1 \leq t \leq T \end{cases} \quad \text{---- (D. 16)}$$

$$\begin{cases} \dot{X} = 0 & \text{si } 0 \leq t < t_0 \\ \dot{X} = a_1 \cdot (t - t_0) & \text{si } t_0 \leq t < t_1 \\ \dot{X} = a_2 \cdot t + (a_1 - a_2) \cdot t_1 - a_1 \cdot t_0 & \text{si } t_1 \leq t \leq T \end{cases} \quad \text{---- (D. 17)}$$

$$\begin{cases} X = X_i & \text{si } 0 \leq t < t_0 \\ X = \text{Parabole}(0, V_1, t_0, t_1, X_i) & \text{si } t_0 \leq t < t_1 \\ X = \text{Parabole}(V_1, 0, t_1, T, X_1) & \text{si } t_1 \leq t \leq T \end{cases} \quad \text{---- (D. 18)}$$

D.3. Problème 2 : Détermination des paramètres cinématiques pour un mouvement en trois phases.

Le problème est posé de la manière suivante : étant donné t_0 , a_1 , a_2 , V_{max} , X_i et X_f trouver les équations horaires d'un mouvement en trois phases menant la position X de X_i vers X_f .

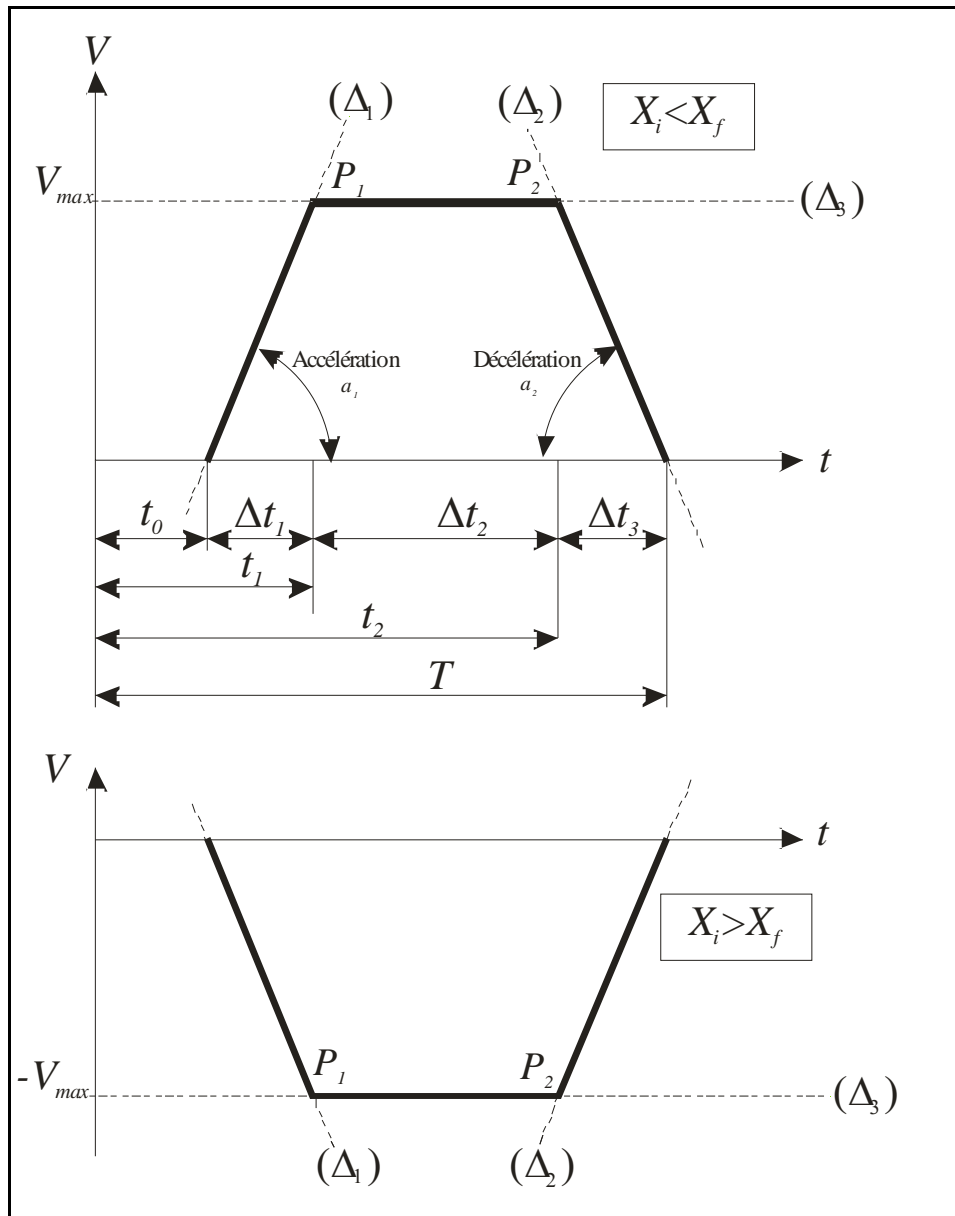


Figure D. 2 : Description d'un mouvement suivant une loi Bang-Bang en trois phases.

Le problème peut être résolu en trouvant le point P_2 dans le plan (t, V) comme montré sur la figure (D.2), ou encore trouver l'instant t_2 (ou la durée Δt_2) où le mouvement change de phase d'un mouvement uniforme vers un freinage.

On suppose que Δt_2 est connu.

(Δ_1) est une droite dont l'équation cartésienne est la suivante :

$$V = a_1 \cdot (t - t_0) \quad \text{----- (D. 19)}$$

On détermine t_1 et t_2 :

$$t_1 = \frac{V_{\max}}{a_1} + t_0 \quad \text{----- (D. 20)}$$

$$t_2 = t_1 + \Delta t_2 = \frac{V_{\max}}{a_1} + \Delta t_2 + t_0 \quad \text{----- (D. 21)}$$

(Δ_2) est une droite dont l'équation cartésienne est la suivante :

$$V = a_2 \cdot t + V_{\max} \left(\frac{a_1 - a_2}{a_1} \right) - a_2 \cdot \Delta t_2 - a_2 \cdot t_0 \quad \text{----- (D. 22)}$$

(Δ_3) est une droite dont l'équation cartésienne est la suivante :

$$V = V_{\max} \quad \text{----- (D. 23)}$$

On détermine les durées du mouvement de la phase 1 et la phase 3 :

$$\begin{cases} \Delta t_1 = \left| \frac{V_{\max}}{a_1} \right| \\ \Delta t_3 = \left| \frac{V_{\max}}{a_2} \right| \end{cases} \quad \text{----- (D. 24)}$$

On détermine T (intersection de (Δ_2) avec l'axe des abscisses) :

$$T = V_{\max} \left(\frac{a_2 - a_1}{a_1 \cdot a_2} \right) + \Delta t_2 + t_0 \quad \text{----- (D. 25)}$$

Donc :

$$\begin{cases} |\Delta X_1| = \frac{1}{2} \frac{(V_{\max})^2}{|a_1|} \\ |\Delta X_2| = |V_{\max}| \cdot \Delta t_2 \\ |\Delta X_3| = \frac{1}{2} \frac{(V_{\max})^2}{|a_2|} \end{cases} \quad \text{----- (D. 26)}$$

Connaissant $\Delta X = X_f - X_i$ (données du problème), on déterminera la valeur de Δt_2 :

$$|\Delta X| = |\Delta X_1| + |\Delta X_2| + |\Delta X_3| \quad \text{----- (D. 27)}$$

$$|\Delta X| = |\Delta X_1| + |V_{\max}| \cdot \Delta t_2 + |\Delta X_3| \Rightarrow \Delta t_2 = \frac{|\Delta X| - |\Delta X_1| - |\Delta X_3|}{|V_{\max}|} \quad \text{----- (D. 28)}$$

Et on détermine t_2 en utilisant l'équation (D.21).

On utilise la fonction « *Parabole* » décrite dans l'annexe E pour trouver les équations horaires du déplacement dans les différentes phases. On admet que pour les trois phases, le déplacement est une fonction polynomiale de degré 2 (ou inférieur).

Phase 1 : $t \in [t_0, t_1]$

$$\left\{ \begin{array}{l} X(t_0) = X_i \\ \left[\frac{dX}{dt} \right]_{t=t_0} = 0 \\ \left[\frac{dX}{dt} \right]_{t=t_1} = V_{\max} \end{array} \right. \quad \text{----- (D. 29)}$$

Le déplacement est donc déterminé ainsi par la fonction Parabole comme suit :

$$X = \text{Parabole}(0, V_{\max}, t_0, t_1, X_i) \quad \text{----- (D. 30)}$$

On détermine la position finale de la phase 1, X_1 en évaluant la fonction Position trouvée en (D.30) à l'instant t_1 .

Phase 2 : $t \in [t_1, t_2]$

$$\left\{ \begin{array}{l} X(t_1) = X_1 \\ \left[\frac{dX}{dt} \right]_{t=t_1} = V_{\max} \\ \left[\frac{dX}{dt} \right]_{t=t_2} = V_{\max} \end{array} \right. \quad \text{----- (D. 31)}$$

Le déplacement est donc déterminé ainsi par la fonction Parabole comme suit :

$$X = \text{Parabole}(V_{\max}, V_{\max}, t_1, t_2, X_1) \quad \text{----- (D. 32)}$$

Remarque :

Il est évident que pour la phase 2, la fonction parabole nous renvoi une équation d'une droite et non pas d'une parabole.

On détermine la position finale de la phase 2, X_2 en évaluant la fonction Position trouvée en (D.32) à l'instant t_2 .

Phase 3 : $t \in [t_2, T]$

$$\begin{cases} X(t_2) = X_2 \\ \left[\frac{dX}{dt} \right]_{t=t_2} = V_{\max} \\ \left[\frac{dX}{dt} \right]_{t=T} = 0 \end{cases} \quad \text{---- (D. 33)}$$

Le déplacement est donc déterminé ainsi par la fonction « *parabole* » comme suit :

$$X = \text{Parabole}(V_{\max}, 0, t_2, T, X_2) \quad \text{---- (D. 34)}$$

Le mouvement est donc entièrement déterminé par les trois systèmes suivants :

$$\begin{cases} \ddot{X} = 0 & \text{si } 0 \leq t < t_0 \\ \ddot{X} = a_1 & \text{si } t_0 \leq t < t_1 \\ \ddot{X} = 0 & \text{si } t_1 \leq t < t_2 \\ \ddot{X} = a_2 & \text{si } t_1 \leq t \leq T \end{cases} \quad \text{---- (D. 35)}$$

$$\begin{cases} \dot{X} = 0 & \text{si } 0 \leq t < t_0 \\ \dot{X} = a_1 \cdot (t - t_0) & \text{si } t_0 \leq t < t_1 \\ \dot{X} = V_{\max} & \text{si } t_1 \leq t < t_2 \\ \dot{X} = a_2 \cdot t + V_{\max} \left(\frac{a_1 - a_2}{a_1} \right) - a_2 \cdot \Delta t_2 - a_2 \cdot t_0 & \text{si } t_1 \leq t \leq T \end{cases} \quad \text{---- (D. 36)}$$

$$\begin{cases} X = X_i & \text{si } 0 \leq t < t_0 \\ X = \text{Parabole}(0, V_{\max}, t_0, t_1, X_i) & \text{si } t_0 \leq t < t_1 \\ X = \text{Parabole}(V_{\max}, V_{\max}, t_1, t_2, X_1) & \text{si } t_1 \leq t < t_2 \\ X = \text{Parabole}(V_{\max}, 0, t_2, T, X_2) & \text{si } t_1 \leq t \leq T \end{cases} \quad \text{---- (D. 37)}$$

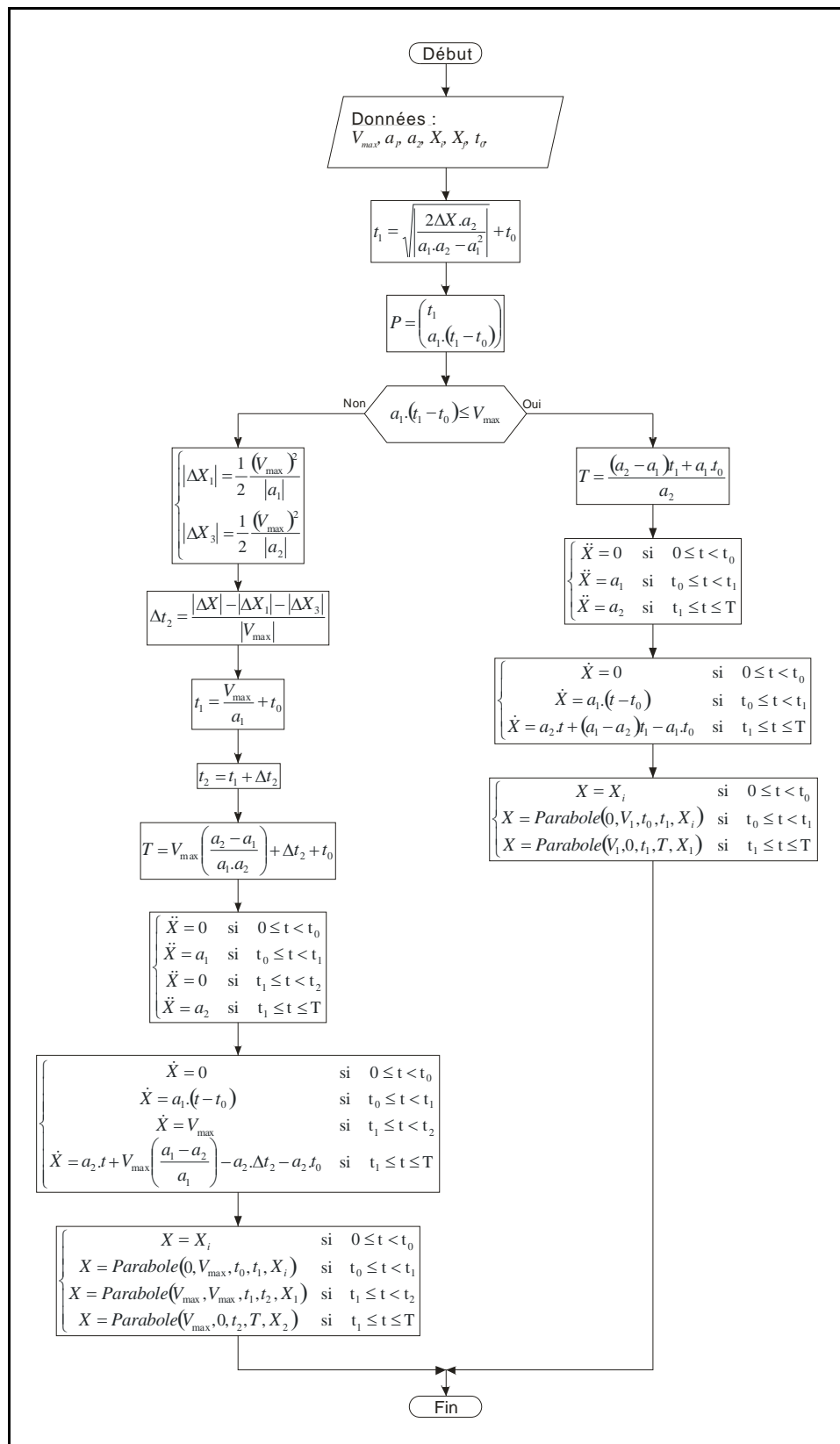
D.4. Cas général.

Dans le cas général, on dispose d'un actuateur (dans la plus part des cas, il s'agit d'un moteur électrique piloté par un variateur de fréquence) qui dispose à son tour de certains paramètres fixés par le constructeur, comme l'accélération de démarrage et la décélération du freinage ainsi que la vitesse maximale que cet actuateur peut atteindre. On demande un déplacement $\Delta X = X_f - X_i$, on ne sait donc auparavant si le mouvement doit se faire en deux ou en trois phases.

On présente ici, un organigramme pour l'automatisation de cette tâche. Cet organigramme se tâche de vérifier d'abord si le mouvement peut se faire en deux phases

sans dépasser la vitesse maximale spécifiée, autrement le mouvement doit se faire en trois phases.

Organigramme D. 1 : Détermination des paramètres cinématiques d'un mouvement suivant la loi Bang-Bang



APPENDICE E

OBTENTION DE L'ÉQUATION D'UNE PARABOLE RELIANT DEUX POINTS.

E.1. Description du problème.

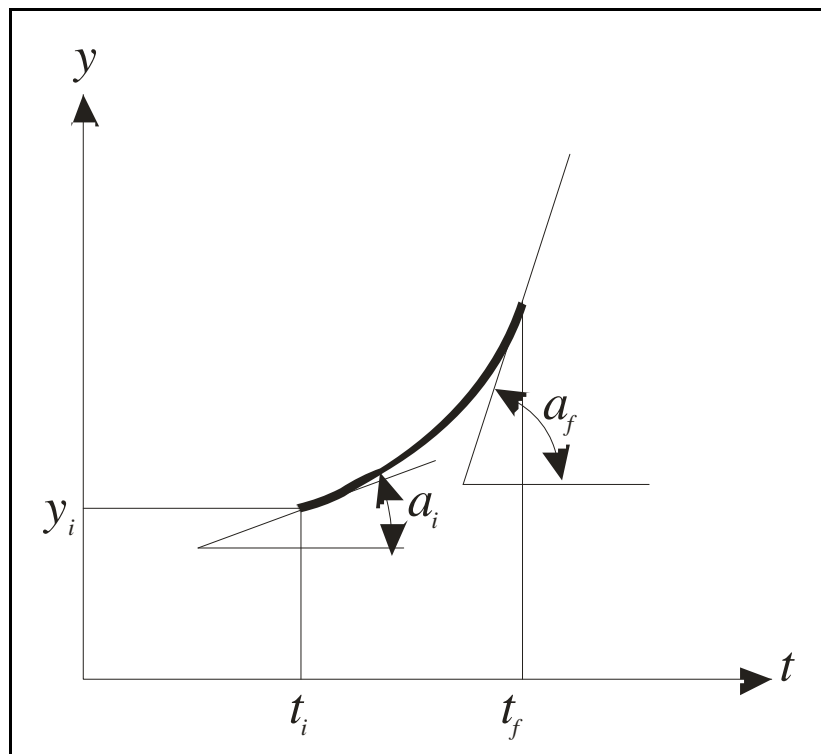


Figure E. 1 : Parabole reliant deux points d'un plan.

Étant donné deux points $P_i = \begin{pmatrix} t_i \\ y_i \end{pmatrix}$ et $P_f = \begin{pmatrix} t_f \\ y_f \end{pmatrix}$, on veut relier les deux points par une courbe dont l'équation cartésienne est un polynôme d'ordre 2, et connaissant les valeurs de la dérivée de la fonction (parabole) au début et à la fin du segment $[t_i, t_f]$ ainsi que la valeur de la fonction à l'instant t_i .

E.2. Analyse du problème.

L'équation cherchée est de la forme :

$$y = at^2 + bt + c \quad \text{---- (E. 1)}$$

On connaît les valeurs de la dérivée aux instants t_i et t_f , donc :

$$\left[\frac{dy}{dt} \right]_{t=t_i} = 2at_i + b = a_i \quad \text{---- (E. 2)}$$

$$\left[\frac{dy}{dt} \right]_{t=t_f} = 2at_f + b = a_f \quad \text{---- (E. 3)}$$

Ou encore :

$$\begin{bmatrix} 2t_i & 1 \\ 2t_f & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} a_i \\ a_f \end{bmatrix} \quad \text{---- (E. 4)}$$

La résolution du système (E.4) donne :

$$\begin{cases} a = \frac{a_i - a_f}{2(t_i - t_f)} \\ b = \frac{a_f t_i - a_i t_f}{t_i - t_f} \end{cases} \quad \text{---- (E. 5)}$$

Remarque :

Si $a_i = a_f$, le coefficient a deviendra nul, donc la courbe qui relie les deux points est une droite et non pas une parabole.

Il nous reste de déterminer la constante c :

On a pour $t = t_i$; $y = y_i$, donc :

$$c = y_i - at_i^2 - bt_i \quad \text{---- (E. 6)}$$

E.3. Numérisation.

On veut élaborer une procédure informatique qui nous calcule les coefficients a , b et c , nous donnerons un programme en langage Matlab qui réalise cette tâche avec les données : a_i , a_f , t_i , t_f et y_i .

```
function [a b c]=Parabole(ai,af,ti,tf,yi)
M=[2*ti 1;2*tf 1];
V=[ai;af];
W=inv(M)*V;
a=W(1);
b=W(2);
c=yi-a*ti^2-b*ti;
```

APPENDICE F

LISTING DES PROGRAMMES SOUS « MATLAB ».

```
function y=Mh(alpha,d,theta,r)
%fonction qui retourne la matrice de transformation homogène définissant le
%repère (Rj) dans le repère précédent (Rj-1). les paramètres alpha, d,
%theta et r sont les paramètres DH du lien j.
y=[cos(theta) -sin(theta) 0 d;...
   cos(alpha)*sin(theta) cos(alpha)*cos(theta) -sin(alpha) -r*sin(alpha);...
   sin(alpha)*sin(theta) sin(alpha)*cos(theta) cos(alpha) r*cos(alpha);...
   0 0 0 1];
```

```
function y=MHR(R,i,k)
%fonction de calcul de la matrice de passage entre le lien i et le lien k
%(Ramener k en i) du robot R
%le Robot R est une matrice de (n+3)x10 Les trois premières lignes
%représentent la position et l'orientation du robot dans le repère atelier
%et les autres lignes chacune représente un lien partant du lien N° 1
%les colonnes représentent respectivement : Sigma, Alpha, d, Theta, r,
%qmin, qmax, Vit_nom, Tps_Acc et Tps_Dec.
%Si i=-1, alors on calcul par rapport au repère atelier
y=eye(4);
id=min(i,k);
ig=max(i,k);
if id==-1
    for j=4:ig+3
        y=y*Mh(R(j,2),R(j,3),R(j,4),R(j,5));
    end
    y=[R(1,1) R(1,2) R(1,3) R(1,4);R(2,1) R(2,2) R(2,3) R(2,4);R(3,1) R(3,2)
R(3,3) R(3,4);0 0 0 1]*y;
else
    for j=id+4:ig+3
        y=y*Mh(R(j,2),R(j,3),R(j,4),R(j,5));
    end
end
if i>k
    y=inv(y);
end
```

```
function Y=MGD(R,x,y,z,lien,lien_relatif)
%fonction donnant le Modèle géométrique direct pour calculer la position du
%point (x,y,z) appartenant au lien "lien" relativement au lien
%"lien_relatif"
%la fonction retourne une matrice 4x4 donnant l'orientation dans la partie
%3x3 haut gauche et le vecteur position à droite
Y=MHR(R,lien_relatif,lien);
ZZ=Y*[x;y;z;1];
Y(1,4)=ZZ(1);
Y(2,4)=ZZ(2);
Y(3,4)=ZZ(3);
```

```

function y=Rot(i,theta)
% fonction qui donne la matrice de rotation pure autour d'un des axes principaux
% i : représente l'axe de rotation (1:x) (2:y) (3:z)
% theta : représente l'angle de rotation en radian
if i==1
    y=[1 0 0 0;0 cos(theta) -sin(theta) 0;0 sin(theta) cos(theta) 0;0 0 0 1];
elseif i==2
    y=[cos(theta) 0 sin(theta) 0;0 1 0 0;-sin(theta) 0 cos(theta) 0;0 0 0 1];
elseif i==3
    y=[cos(theta) -sin(theta) 0 0;sin(theta) cos(theta) 0 0;0 0 1 0;0 0 0 1];
end

```

```

function [y]=Rot_Gen(ux,uy,uz,theta)
%Rotation générale d'un angle theta autour d'un axe passant par l'origine
%et dirigé par le vecteur u=(ux,uy,uz)
uu=[ux;uy;uz]/norm([ux;uy;uz]);
pu=[0 -uu(3) uu(2);uu(3) 0 -uu(1);-uu(2) uu(1) 0];
z=uu*uu'*(1-cos(theta))+eye(3)*cos(theta)+pu*sin(theta);
y=[z [0;0;0];[0 0 0 1]];

```

```

function y=MRot(R,i,j)
%fonction qui retourne la matrice de rotation qui ramène le repère j dans
%le repère i du robot R
A=MHR(R,i,j);
y=A(1:3,1:3);

```

```

function y=Trans(a,b,c);
% Donne la matrice de Translation pure
% a,b,c : représentent les composantes du vecteur translation
y=[1 0 0 a;0 1 0 b;0 0 1 c;0 0 0 1];
end

```

```

function y=VTrans(R,i,j)
%fonction qui retourne le vecteur translation qui ramène le lien j dans le
%lien i du robot R;
A=MHR(R,i,j);
y=A(1:3,4:4);

```

```

function y=Pre_Cross(V)
%fonction qui donne la matrice du pré produit vectoriel du vecteur V:
y=zeros(3,3);
y(1,2)=-V(3);
y(1,3)=V(2);
y(2,1)=V(3);
y(2,3)=-V(1);
y(3,1)=-V(2);
y(3,2)=V(1);

```

```

function y=Vecteur(P1,P2)
%fonction qui retourne les composante du vecteur dont l'origine est le
%point P1 et l'extrémité est le point P2
for i=1:3
    y(i)=P2(i)-P1(i);
end
y=y';

```

```

function y=Moment(V,P,O)
%fonction qui retourne le moment du vecteur V appliquée au point P autour
%du point O
z=Vecteur(O,P);
y=cross(z,V);

```

```

function y=Situation_OT(R,lien_relatif)
%Fonction qui retourne la situation de l'outil du robot R relativement au
%repère lié au "Lien_relatif"

%Récupérer la situation locale de l'outil dans La matrice SO
SO=R(1:3,5:8);
SO=[SO;0 0 0 1];
%Récupérer l'indice de l'organe terminal
n=size(R);
n=n(1)-3;
y=MGD(R,0,0,0,n,lien_relatif)*SO;

```

```

function y=jacobien(R,l,m)
%fonction de calcul du jacobien du robot R
%la fonction retourne une matrice de dimension 6x(m-1)
%Il faut absolument que l<m
if l<m
y=zeros(6,m-1);
for k=1:m-1
A=MHR(R,k+1,m);
B=MHR(R,l,k+1);
s=[B(1,1);B(2,1);B(3,1)];
n=[B(1,2);B(2,2);B(3,2)];
a=[B(1,3);B(2,3);B(3,3)];
Px=A(1,4);
Py=A(2,4);
Sigma=R(3+k+1);
Z=[Sigma*a+(1-Sigma)*(-Py*s+Px*n);(1-Sigma)*a];
for i=1:6
y(i,k)=Z(i);
end
end
end

```

```

function y=Dynamique_Objet(Tab_I,Tab_Vit,Tab_Acc,Prise,g);
%Fonction de calcul dynamique d'un objet manipulé par un seul robot
%Tab_I=Paramètres inertiels/Rep. local=Masse,I11,I12,I13,I22,I23,I33,CGX,CGY,CGZ
%Tab_Vit = Vitesses = Vx,Vy,Vz,OMEGAx,OMEGAY,OMEGAZ
%Tab_Acc = Accélérations = VPx,VPy,VPz,OMEGAPx,OMEGAPy,OMEGAPz
%Prise = Coordonnées de la prise relativement au rep. local
%g:Accélération de la pesanteur

R1=(Tab_I(8)-Prise(1);Tab_I(9)-Prise(2);Tab_I(10)-Prise(3));
R2=Pre_Cross(R1);
A=[eye(3),zeros(3,3);R2,eye(3)];
T1=[Tab_I(2),-Tab_I(3),-Tab_I(4);-Tab_I(3),Tab_I(5),-Tab_I(6);-Tab_I(4),-
Tab_I(6),Tab_I(7)];
S1=[Tab_I(8);Tab_I(9);Tab_I(10)];
S2=Pre_Cross(S1);
T2=T1+Tab_I(1)*(S2*S2);
B=[Tab_I(1)*eye(3),zeros(3,3);zeros(3,3),T2];
V=[Tab_Acc(1);Tab_Acc(2);Tab_Acc(3);Tab_Acc(4);Tab_Acc(5);Tab_Acc(6)];
C=[eye(3),zeros(3,3);R2,eye(3)];
OMEGA=[Tab_Vit(4);Tab_Vit(5);Tab_Vit(6)];
OMEGA1=Pre_Cross(OMEGA);
W=[-Tab_I*g;OMEGA1*(T2*OMEGA)];
y=(A*B)*V+C*W;

```

```

function y=Newton_Euler(R,Tab_Cin,Tab_I,Tab_F,Tab_M,g)
%Fonction qui calcul le modèle dynamique en utilisant l'algorithmme de
%Newton-Euler pour le robot R avec
%R : Tableau du robot avec les positions mis à jours
%Tab_Cin : nx5 :Sigma,Sigma_,q, q'"
%Tab_I :nx10: Masse,I11,I12,I13,I22,I23,I33,CGx,CGy,CGz;
%Tab_F :nx3: Fx,Fy,Fz : Forces extérieures
%Tab_M :nx3: Mx,My,Mz : Moments extérieurs
%g:3x1:Accélération de la pesanteur

%Récupération de n : Nombre de liens
[n,m]=size(R);
n:=n-3;
for j=1:n
    if j==1
        OMEGA1=[0;0;0];%Initialisé à 0
        OMEGAP1=[0;0;0];%Initialisée à 0
        U1=zeros(3,3);%Initialisée à 0 avec les vitesses nulles et les
accélérationes nulles
        VP1=-g;%Initialisée à -g
    else
        OMEGA1=OMEGA3;
        OMEGAP1=OMEGAP2;
        U1=U2;
        VP1=VP2;
    end
    %Récurrence
    A1=MRot(R,j,j-1);
    OMEGA2=A1*OMEGA1;
    OMEGA3=OMEGA2+Tab_Cin(j,2)*Tab_Cin(j,4)*[0;0;1];

    OMEGAP2=A1*OMEGAP1+Tab_Cin(j,2)*(Tab_Cin(j,5)*[0;0;1]+cross(OMEGA2,Tab_Cin(j,4)*[
0;0;1]));
    P1=VTrans(R,j-1,j);

    VP2=A1*(VP1+U1*P1)+Tab_Cin(j,1)*(Tab_Cin(j,5)*[0;0;1]+2*cross(OMEGA2,Tab_Cin(j,4)
*[0;0;1]));
    U2=Pre_Cross(OMEGAP2)+Pre_Cross(OMEGA3)*Pre_Cross(OMEGA3);
    MS1=Tab_I(j,1)*[Tab_I(j,8),Tab_I(j,9),Tab_I(j,10)];
    F1=Tab_I(j,1)*VP2+U2*MS1;
    Tab_FM(j,1)=F1(1);
    Tab_FM(j,2)=F1(2);
    Tab_FM(j,3)=F1(3);
    J1=[Tab_I(j,2),-Tab_I(j,3),-Tab_I(j,4);-Tab_I(j,3),Tab_I(j,5),-Tab_I(j,6);-
Tab_I(j,4),-Tab_I(j,6),Tab_I(j,7)];
    M1=J1*OMEGAP2+cross(OMEGA3,J1*OMEGA3)+cross(MS1,VP2);
    Tab_FM(j,4)=M1(1);
    Tab_FM(j,5)=M1(2);
    Tab_FM(j,6)=M1(3);
end
%Récurrence arrière
for j=n:-1:1
    if j==n %Première itération
        F1=[0;0;0];
        P1=[0;0;0];
        M1=[0;0;0];
        A2=zeros(3,3);
    else
        F1=F4;
        P1=VTrans(R,j,j+1);
        M1=M3;
        A2=MRot(R,j,j+1);
    end
    F2=[Tab_FM(j,1);Tab_FM(j,2);Tab_FM(j,3)];
    F3=F2+F1;

```

```

A1=MRot(R,j-1,j);
F4=A1*F3;
M2=[Tab_FM(j,4);Tab_FM(j,5);Tab_FM(j,6)];
M3=M2+A2*M1+cross(P1,F1);
y(j)=(Tab_Cin(j,1)*F3+Tab_Cin(j,2)*M3)'*[0;0;1];
end

```

```

%Programmes permettant de tracer les courbes suivantes
%Data_P,Data_V,X(t),Y(t),Z(t),X(Y),Z(X),Z(Y),Vx(t),Vy(t),Vz(t),V(t),ax(t),
%ay(t),az(t),a(t)
%Utilise les données DATA, DATA_P,DATA_V,DATA_A
%Définir l'ordre des couleurs des courbes
Couleur=[1 1 0;1 0 1;0 1 1;1 0 0;0 1 0;0 0 1;0 0 0;1 1 1];
%Tracer les positions articulaires
figure('Color',[1 1 1],'Name','Positions articulaires');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(DATA_P,'LineWidth',3);
title('Positions articulaires');
xlabel('Temps (x100 sec)');
ylabel('Positions articulaires (Rad)');
grid on
%Tracer les vitesses articulaires
figure('Color',[1 1 1],'Name','Vitesse articulaires');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(DATA_V,'LineWidth',3);
title('Vitesse articulaires');
xlabel('Temps (x100 sec)');
ylabel('Vitesse articulaires (Rad/sec)');
grid on
%Préparer X(Y),Z(X) et Z(Y)
X=DATA(:,2);
Y=DATA(:,3);
Z=DATA(:,4);
%Tracer X(Y)
figure('Color',[1 1 1],'Name','Trajectoire X(Y)');
set(gcf,'DefaultAxesColorOrder',[1 0 0]);
plot(Y,X,'LineWidth',3);
title('Trajectoire X(Y)');
xlabel('Y (m)');
ylabel('X (m)');
grid on
axis equal
%Tracer Z(X)
figure('Color',[1 1 1],'Name','Trajectoire Z(X)');
set(gcf,'DefaultAxesColorOrder',[1 0 0]);
plot(X,Z,'LineWidth',3);
title('Trajectoire Z(X)');
xlabel('X (m)');
ylabel('Z (m)');
grid on
axis equal
%Tracer Z(Y)
figure('Color',[1 1 1],'Name','Trajectoire Z(Y)');
set(gcf,'DefaultAxesColorOrder',[1 0 0]);
plot(Y,Z,'LineWidth',3);
title('Trajectoire Z(Y)');
xlabel('Y (m)');
ylabel('Z (m)');
grid on
axis equal
clear X;
clear Y;
clear Z;
%Préparer le traçage des Vitesses

```

```

t=DATA(:,1);
DX=DATA(:,5);
DY=DATA(:,6);
DZ=DATA(:,7);
V=DATA(:,8);
Tab_V=[t DX DY DZ V];
figure('Color',[1 1 1],'Name','Vitesses opérationnelles');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tab_V,'LineWidth',3);
title('Vitesses opérationnelles');
xlabel('Temps (x100 sec)');
ylabel('Vitesses (m/sec)');
grid on
clear DX;
clear DY;
clear DZ;
clear V;
clear Tab_V;
%Préparer le traçage des Accélérations
t=DATA(:,1);
D2X=DATA(:,9);
D2Y=DATA(:,10);
D2Z=DATA(:,11);
A=DATA(:,12);
Tab_A=[t D2X D2Y D2Z A];
figure('Color',[1 1 1],'Name','Accélérations opérationnelles');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tab_A,'LineWidth',3);
title('Accélérations opérationnelles');
xlabel('Temps (x100 sec)');
ylabel('Accélérations (m/sec^{2})');
grid on
clear D2X D2Y D2Z A Tab_A t Couleur;

```

```

%Programme permettant d'analyser les données de coopération de deux robots
%plan à 2 ddl
clear Th11 Th12 Th21 Th22 Th1 Th2 dTh1 dTh2 d2Th1 d2Th2;
clear n m i;
clear h fjl fj_1;
%Importation des données
load('D:\Magister\Résultats\COOP\COOP');

Couleur=[1 1 0;1 0 1;0 1 1;1 0 0;0 1 0;0 0 1;0 0 0;1 1 1];
Temps=COOP(:,1:1);
Th11=COOP(:,23:23);
Th12=COOP(:,24:24);
Th21=COOP(:,25:25);
Th22=COOP(:,26:26);
Sit_Objet=COOP(:,27:38);
Sit_Objet=[Temps Sit_Objet];
Th1=[Temps Th11 Th12];
Th2=[Temps Th21 Th22];
%Obtention des positions opérationnelles de l'objet
X_Objet=COOP(:,30:30);
Y_Objet=COOP(:,34:34);
Z_Objet=COOP(:,38:38);
POS_OP_Objet=[Temps X_Objet Y_Objet Z_Objet];

%Obtention des vitesses et des accélérations articulaires et opérationnelles
[n,m]=size(COOP);
dTh1=[Temps zeros(n,2)];
dTh2=[Temps zeros(n,2)];
d2Th1=[Temps zeros(n,2)];
d2Th2=[Temps zeros(n,2)];
POS_ANG_Objet=[Temps zeros(n,3)];

```

```

VIT_ANG_Objjet=[Temps zeros(n,3)];
ACC_ANG_Objjet=[Temps zeros(n,3)];
VIT_OP_Objjet=[Temps zeros(n,3)];
ACC_OP_Objjet=[Temps zeros(n,3)];
h=Temps(2)-Temps(1);%Pas de discrétisation

%Dérivation numérique des positions Robot : 1 - Lien : 1
nbre=n-1;
for i=2:nbre
    fjl=Th1(i+1,2);
    fj_1=Th1(i-1,2);
    dTh1(i,2)=(fjl-fj_1)/(2*h);
end
%Dérivation numérique des positions Robot : 1 - Lien : 2
for i=2:nbre
    fjl=Th1(i+1,3);
    fj_1=Th1(i-1,3);
    dTh1(i,3)=(fjl-fj_1)/(2*h);
end

%Dérivation numérique des positions Robot : 2 - Lien : 1
for i=2:nbre
    fjl=Th2(i+1,2);
    fj_1=Th2(i-1,2);
    dTh2(i,2)=(fjl-fj_1)/(2*h);
end
%Dérivation numérique des positions Robot : 2 - Lien : 2
for i=2:nbre
    fjl=Th2(i+1,3);
    fj_1=Th2(i-1,3);
    dTh2(i,3)=(fjl-fj_1)/(2*h);
end

%Dérivation numérique d'ordre 2 des positions Robot : 1 - Lien : 1
for i=2:nbre
    fjl=Th1(i+1,2);
    fj_1=Th1(i-1,2);
    fj=Th1(i,2);
    d2Th1(i,2)=(fjl+fj_1-2*fj)/(h^2);
end
%Dérivation numérique d'ordre 2 des positions Robot : 1 - Lien : 2
for i=2:nbre
    fjl=Th1(i+1,3);
    fj_1=Th1(i-1,3);
    fj=Th1(i,3);
    d2Th1(i,3)=(fjl+fj_1-2*fj)/(h^2);
end

%Dérivation numérique d'ordre 2 des positions Robot : 2 - Lien : 1
for i=2:nbre
    fjl=Th2(i+1,2);
    fj_1=Th2(i-1,2);
    fj=Th2(i,2);
    d2Th2(i,2)=(fjl+fj_1-2*fj)/(h^2);
end
%Dérivation numérique d'ordre 2 des positions Robot : 2 - Lien : 2
for i=2:nbre
    fjl=Th2(i+1,3);
    fj_1=Th2(i-1,3);
    fj=Th2(i,3);
    d2Th2(i,3)=(fjl+fj_1-2*fj)/(h^2);
end

```



```

%Détermination des positions angulaires de l'objet
for i=1:n
    C=COOP(i,32);%Cosinus
    S=COOP(i,36);%Sinus
    POS_ANG_Objjet(i,2)=atan2(S,C);
end
%Dérivation des positions angulaires pour avoir les vitesses angulaires de
l'objet
for i=2:nbre
    fjl=POS_ANG_Objjet(i+1,2);
    fj_1=POS_ANG_Objjet(i-1,2);
    VIT_ANG_Objjet(i,2)=(fjl-fj_1)/(2*h);
end
%Dérivation d'ordre 2 des positions angulaires pour avoir les accélérations
angulaires de l'objet
for i=2:nbre
    fjl=POS_ANG_Objjet(i+1,2);
    fj_1=POS_ANG_Objjet(i-1,2);
    fj=POS_ANG_Objjet(i,2);
    ACC_ANG_Objjet(i,2)=(fjl+fj_1-2*fj)/(h^2);
end
%Obtention des vitesses et accélérations opérationnelles de l'objet
for i=2:nbre
    fjl=POS_OP_Objjet(i+1,2);
    fj_1=POS_OP_Objjet(i-1,2);
    fj=POS_OP_Objjet(i,2);
    VIT_OP_Objjet(i,2)=(fjl-fj_1)/(2*h);%Vx
    ACC_OP_Objjet(i,2)=(fjl+fj_1-2*fj)/(h^2);%Ax

    fjl=POS_OP_Objjet(i+1,3);
    fj_1=POS_OP_Objjet(i-1,3);
    fj=POS_OP_Objjet(i,3);
    VIT_OP_Objjet(i,3)=(fjl-fj_1)/(2*h);%Vy
    ACC_OP_Objjet(i,3)=(fjl+fj_1-2*fj)/(h^2);%Ay

    fjl=POS_OP_Objjet(i+1,4);
    fj_1=POS_OP_Objjet(i-1,4);
    fj=POS_OP_Objjet(i,4);
    VIT_OP_Objjet(i,4)=(fjl-fj_1)/(2*h);%Vz
    ACC_OP_Objjet(i,4)=(fjl+fj_1-2*fj)/(h^2);%Az
end

%Tracer les positions articulaires du Robot 1
figure('Color',[1 1 1],'Name','Positions articulaires du robot 1');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Th1,'LineWidth',3);
title('Positions articulaires du Robot 1');
xlabel('Temps (x100 sec)');
ylabel('Positions articulaires (Rad)');
grid on
%Tracer les positions articulaires du Robot 2
figure('Color',[1 1 1],'Name','Positions articulaires du robot 2');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Th2,'LineWidth',3);
title('Positions articulaires du Robot 2');
xlabel('Temps (x100 sec)');
ylabel('Positions articulaires (Rad)');
grid on

%Tracer les Vitesses articulaires du Robot 1
figure('Color',[1 1 1],'Name','Vitesses articulaires du robot 1');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(dTh1,'LineWidth',3);
title('Vitesses articulaires du Robot 1');
xlabel('Temps (x100 sec)');

```

```

ylabel('Vitesses articulaires (Rad/sec)');
grid on
%Tracer les vitesses articulaires du Robot 2
figure('Color',[1 1 1],'Name','Vitesses articulaires du robot 2');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(dTh2,'LineWidth',3);
title('Vitesses articulaires du Robot 2');
xlabel('Temps (x100 sec)');
ylabel('Vitesses articulaires (Rad/sec)');
grid on

%Tracer les accélérations articulaires du Robot 1
figure('Color',[1 1 1],'Name','Accélérations articulaires du robot 1');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(d2Th1,'LineWidth',3);
title('Accélérations articulaires du Robot 1');
xlabel('Temps (x100 sec)');
ylabel('Accélérations articulaires (Rad/sec^{2})');
grid on
%Tracer les accélérations articulaires du Robot 2
figure('Color',[1 1 1],'Name','Accélérations articulaires du robot 2');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(d2Th2,'LineWidth',3);
title('Accélérations articulaires du Robot 2');
xlabel('Temps (x100 sec)');
ylabel('Accélérations articulaires (Rad/sec^{2})');
grid on

clear C S Couleur Temps Th11 Th12 Th21 Th22 X_Objet Y_Objet Z_Objet;
clear fj fj1 fj_1 h i m n nbre;

```

```

%Programmes permettant de tracer les courbes des réactions des
%articulations pour le transfert d'un objet par un seul robot plan à 2 ddl à
%vide (sans objet)
%Importation des données
load('D:\Magister\Résultats\Validation\DATA_DYN1');
Temps=DATA_DYN1(:,1:1);
Tho1=[Temps DATA_DYN1(:,2:2)];
Tho2=[Temps DATA_DYN1(:,3:3)];
%Définir l'ordre des couleurs des courbes
Couleur=[1 1 0;1 0 1;0 1 1;1 0 0;0 1 0;0 0 1;0 0 0;1 1 1];

%Tracer les réactions Lien 1
figure('Color',[1 1 1],'Name','Réactions de l''articulation du lien 1');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tho1,'LineWidth',3);
title('Réactions de l''articulation du lien 1');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on
%Tracer les réactions Lien 2
figure('Color',[1 1 1],'Name','Réactions de l''articulation du lien 2');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tho2,'LineWidth',3);
title('Réactions de l''articulation du lien 2');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on

clear DATA_DYN1 Tho1 Tho2 Temps;
clear Couleur;

```

```

%Programme permettant d'analyser les données dynamique de la coopération
%entre deux robots plans à 2 ddl. (données issues de SCDRC)
load('D:\Magister\Résultats\COOP\DATA_DYN_COOP');
Temps=DATA_DYN_COOP(:,1:1);
Tho_R11=DATA_DYN_COOP(:,2:2);%Robot 1 lien 1
Tho_R12=DATA_DYN_COOP(:,3:3);%Robot 1 Lien 2
Tho_R21=DATA_DYN_COOP(:,4:4);%Robot 2 lien 1
Tho_R22=DATA_DYN_COOP(:,5:5);%Robot 2 Lien 2
Tho_R11=[Temps Tho_R11];
Tho_R12=[Temps Tho_R12];
Tho_R21=[Temps Tho_R21];
Tho_R22=[Temps Tho_R22];
%Définir l'ordre des couleurs des courbes
Couleur=[1 1 0;1 0 1;0 1 1;1 0 0;0 1 0;0 0 1;0 0 0;1 1 1];

%Tracer les réactions du robot1 lien 1
figure('Color',[1 1 1],'Name','Réactions des articulations Robot1 - Lien 1');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tho_R11,'LineWidth',3);
title('Réactions des articulations : Robot 1 - Lien 1');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on
%Tracer les réactions du robot1 lien 2
figure('Color',[1 1 1],'Name','Réactions des articulations Robot1 - Lien 2');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tho_R12,'LineWidth',3);
title('Réactions des articulations : Robot 1 - Lien 2');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on

%Tracer les réactions du robot2 lien 1
figure('Color',[1 1 1],'Name','Réactions des articulations Robot2 - Lien 1');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tho_R21,'LineWidth',3);
title('Réactions des articulations : Robot 2 - Lien 1');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on
%Tracer les réactions du robot2 lien 2
figure('Color',[1 1 1],'Name','Réactions des articulations Robot2 - Lien 2');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tho_R22,'LineWidth',3);
title('Réactions des articulations : Robot 2 - Lien 2');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on

clear DATA_DYN_COOP Couleur;

```

```

%Programmes permettant de tracer les courbes suivantes des réaction
%dynamique pour le transfert d'un objet par un seul robot plan à 2 ddl
%Importation des données
load('D:\Magister\Résultats\MDI\DATA_DYN1');
Temps=DATA_DYN1(:,1:1);
Tho1=[Temps DATA_DYN1(:,2:2)];
Tho2=[Temps DATA_DYN1(:,3:3)];
%Définir l'ordre des couleurs des courbes
Couleur=[1 1 0;1 0 1;0 1 1;1 0 0;0 1 0;0 0 1;0 0 0;1 1 1];

%Tracer les réactions Lien 1
figure('Color',[1 1 1],'Name','Réactions des articulations : Lien 1');
set(gcf,'DefaultAxesColorOrder',Couleur);

```

```

plot(Tho1,'LineWidth',3);
title('Réactions des articulations : Lien 1');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on
%Tracer les réactions Lien 2
figure('Color',[1 1 1],'Name','Réactions des articulations : Lien 2');
set(gcf,'DefaultAxesColorOrder',Couleur);
plot(Tho2,'LineWidth',3);
title('Réactions des articulations : Lien 2');
xlabel('Temps (x100 sec)');
ylabel('Couple de la réaction (N.m)');
grid on

clear DATA_DYN1 Tho1 Tho2 Temps;
clear Couleur;

```

```

%Programme exemple pour la validation du modèle dynamique
%Un robot Plan à deux degrés de liberté REF.17 Pages 173 à 176
%Les données : m1 m2 l1 l2 g
%Les vecteurs Positions Vitesses et accélérations des deux liens du robot
%DATA_P DATA_V DATA_A : Resp. Positions, Vitesses et accélérations articulaires
%Le programme renvoi le vecteur Tho de trois colonnes : t Tho1 Tho2
l1=0.5;
l2=0.2;
m1=1.9267;
m2=0.1233;
g=9.8;
[n,m]=size(DATA_P);
Tho=zeros(n,3);
for i=1:n
    Tho(i,1)=DATA_P(i,1);
    Th1=DATA_P(i,2);
    Th2=DATA_P(i,3);
    dTh1=DATA_V(i,2);
    dTh2=DATA_V(i,3);
    d2Th1=DATA_A(i,2);
    d2Th2=DATA_A(i,3);
    c1=cos(Th1);
    c2=cos(Th2);
    s1=sin(Th1);
    s2=sin(Th2);
    c12=cos(Th1+Th2);
    s12=sin(Th1+Th2);

    A=m2*(l2^2)*(d2Th1+d2Th2);
    B=m2*l1*l2*c2*(2*d2Th1+d2Th2);
    C=(m1+m2)*(l1^2)*d2Th1;
    D=-m2*l1*l2*s2*(dTh2^2);
    E=-2*m2*l1*l2*s2*dTh1*dTh2;
    F=m2*l2*g*c12;
    G=(m1+m2)*l1*g*c1;
    Tho(i,2)=A+B+C+D+E+F+G;

    A=m2*l1*l2*c2*d2Th1;
    B=m2*l1*l2*s2*(dTh1^2);
    C=m2*l2*g*c12;
    D=m2*(l2^2)*(d2Th1+d2Th2);
    Tho(i,3)=A+B+C+D;
end
Temps=Tho(:,1:1);
Tho1=Tho(:,2:2);
Tho2=Tho(:,3:3);
Tho1=[Temps Tho1];

```

```

Tho2=[Temps Tho2];

%Traçage des courbes
%Tracer les réactions du lien 1
figure('Color',[1 1 1],'Name','Réactions de l''articulation du lien 1 [17]');
plot(Tho1,'LineWidth',3);
title('Réactions de l''articulation du lien 1 [17]');
xlabel('Temps (x100 sec)');
ylabel('Réactions (N.m)');
grid on
%Tracer les réactions du lien 2
figure('Color',[1 1 1],'Name','Réactions de l''articulation du lien 2 [17]');
plot(Tho2,'LineWidth',3);
title('Réactions de l''articulation du lien 2 [17]');
xlabel('Temps (x100 sec)');
ylabel('Réactions (N.m)');
grid on

clear Th1 Th2 dTh1 dTh2 d2Th1 d2Th2 c1 c2 s1 s2 c12 s12;
clear A B C D E F G Temps Tho1 Tho2 i m n;

```