

UNIVERSITE SAAD DAHLAB DE BLIDA

Faculté de Technologie

Département de Mécanique

MEMOIRE DE MAGISTER

Spécialité : Construction Mécanique

CONTRIBUTION A LA PLANIFICATION DES TRAJECTOIRES

DES SYSTEMES ROBOTISES

Par

BENDALI NADIR

Devant le jury composé de:

A. AIAD	Professeur, U. de Blida	Président
A. ALLALI	Maitre de Conférences A, U. de Blida	Examineur
M. TEMMAR	Maitre de Conférences A, U. de Blida	Examineur
M. OUALI	Professeur, U. de Blida	Rapporteur

Blida, Juillet 2012

ملخص

هذا العمل يعالج مشكلة التخطيط الأمثل لمسارات الأذرع الآلية من نقطة إلى نقطة في عمليات (تلحيم بنقاط ، نقل القطع ، ...)، والمشكلة هي صعبة للغاية مما أدى بالعديد من الباحثين إلى اعتماد افتراضات وتطوير مناهج مخصصة لحالات معينة. اعتمادنا هنا على مناهج الذي تم التحقق من صحته لروبوتات المتنقلة، ونحن نحاول استغلال مزايا هذا المنهاج لمعالجة قضية التخطيط لمسارات الأذرع الآلية ، ولذلك فإن الخطوة الأولى هي توحيد مقياس الوقت، ثم تجزيء المسار إلى قسمين، حيث يتم تحميل القسم الواحد وظيفة الحد من فضاء البحث للجبرية الهندسية من خلال استغلال خصائص الخوارزميات الجينية، ولقد تم تمثيل المسارات باستخدام الدوال التكعيبة التي يتم تولدها بالطريقة العشوائية "الخوارزميات الجينية" لإيجاد مسار الأمثل للأذرع الآلية.

كلمات مفتاحية: الأذرع الآلية ، النمذجة الديناميكية، الدوال التكعيبة ، Stochastic Optimization ، الخوارزميات الجينية.

RESUME

Le présent travail traite le problème de planification optimale de trajectoires pour les robots manipulateurs dans les tâches point à point (Soudage par points, transfert d'objet, ...), le problème est assez difficile ce qui a poussé beaucoup de chercheurs à adopter des hypothèses et développer des approches dédiées à des cas particuliers. Nous adoptons ici une approche qui a été validée pour la robotique mobile et nous essayons d'exploiter ces avantages pour traiter le cas de problème de planification du trajectoire des robots manipulateurs, la démarche consiste au premier temps à normaliser l'échelle de temps puis décomposer la trajectoire en deux fonction, dont laquelle une de ces deux fonction sera charger de limiter l'espace de recherche des contrainte géométrique en exploitant les propriétés des algorithmes génétiques, les fonctions sont modéliser par les splines cubiques ensuite elles sont générées par une technique stochastiques "algorithmes génétiques" qui permet de trouver la trajectoire optimale pour le robot manipulateur.

Mots clés: Robots Manipulateurs, Modélisation Dynamique, Splines Cubiques, Optimisation Stochastique, Algorithmes Génétiques.

ABSTRACT

This work deals with the problem of optimal planning of the trajectories for the robots manipulators in the point-to-point tasks (tasks welding, transfer of object,...), the problem is rather difficult what pushed many researchers to adopt assumptions and to develop approaches dedicated to particular cases. We adopt here an approach which was validated for mobile robotics and we try to exploit these advantages to treat trajectory planning problem of robot manipulators, the first step is to standardize the scale of time then decompose the trajectory into two function, whose which one from these two function will be to charge with limiting the space of research of the geometrical constraint by exploiting the properties of the genetic algorithms, the functions are to model by the cubic spline then they are generated into a stochastic "genetic algorithms" which makes it possible to find the trajectory optimal for the robot manipulator.

Keywords: Robots Manipulators, Dynamic Modeling, Cubic Spline, Stochastic Optimization, Genetic Algorithms.

REMERCIEMENTS

Au terme de ce mémoire, je tiens à remercier en premier lieu **DIEU** le miséricordieux qui m'a donné la force et le courage pour achever ce mémoire

Je tiens à exprimer ma profonde gratitude à Monsieur le Directeur du mémoire le Professeur **M. OUALI** qui m'a beaucoup aidé au cours de la réalisation de ce travail. Je lui exprime ma sincère reconnaissance pour ses orientations pratiques et appréciables. Ses qualités scientifiques et humaines m'ont permis de mener à bien ce travail.

Que le Professeur Monsieur **A. AIAD** président du jury trouve ici ma profonde gratitude pour avoir accepté de présider ce jury.

Que Messieurs **A. ALLALI** et **M. TEMMAR**, soient vraiment remerciés pour avoir accepté de juger ce travail.

J'exprime ma profonde gratitude et mes sincères remerciements à notre enseignant Monsieur **A. MELBOUS** pour ses orientations et son aide.

Je remercie également et infiniment Monsieur **A. E. LEHTIHET, Ph.D**, à l'**EMP** pour son aide et ces conseils, et surtout pour les directives qu'il m'a donné.

Enfin, je tiens à remercier toutes les personnes qui ont contribué, de près ou de loin, au bon déroulement de ce travail, spécialement Monsieur **A. KIFOUCHE** enseignant à la faculté d'électronique à l'université de Saad Dahlab Blida.

DEDICACE

Je dédie ce modeste travail à:

Mes parents,

Mes frères et mes sœurs,

Ainsi qu'à tous ceux qui me sont chers.

TABLE DES MATIERES

RESUME	
REMERCIEMENTS	
TABLE DES MATIERES	
LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX	
INTRODUCTION GENERALE.....	7
1. GENERALITES ET TERMINOLOGIE	
1.1 Introduction.....	14
1.2 Définitions générales.....	14
1.3 Constituants mécaniques des robots manipulateurs.....	15
1.4 Description des robots.....	16
1.4.1 Articulation (ou joint).....	16
1.4.2 Mobilité du robot.....	17
1.4.3 Degré de la tâche Dr.....	17
1.4.4 Coordonnées articulaires.....	17
1.4.5 Coordonnées opérationnelles.....	17
1.4.6 Espace articulaire (espace de configuration).....	18
1.4.7 Espace opérationnel (espace de travail).....	18
1.4.8 Redondance.....	18
1.4.9 Configuration singulières.....	19
1.5 Morphologie des robots manipulateurs.....	19
1.6 Caractéristiques d'un robot.....	20
1.7 Modélisation des bras manipulateurs.....	21
1.7.1 Modélisation géométrique.....	22
1.7.1.1 Repérage de Denavit-Hartenberg (D-H).....	23
1.7.1.2 Modélisation géométrique directe (MGD).....	25
1.7.1.3 Modélisation géométrique inverse (MGI).....	26
1.7.2 Modélisation cinématique (différentiel).....	26
1.7.2.1 Modèle différentiel direct (MDD).....	26

1.7.2.2	Modèle différentiel inverse (MDI).....	27
1.7.3	Modélisation dynamique.....	28
1.7.3.1	Formalisme de LAGRANGE.....	29
1.7.3.2	Formalisme de NEWTON-EULER.....	35
1.8	prise en compte des frottements.....	37
1.9	Conclusion.....	38
2. PLANIFICATION § GENERATION DE MOUVEMENT		
2.1	Introduction.....	39
2.2	Espace de planification de la trajectoire.....	39
2.2.1	Génération de mouvement et système de commande.....	40
2.2.2	Génération de trajectoire dans l'espace articulaire.....	41
2.2.2.1	Trajectoire sans contraintes ou mouvement point à point.....	41
2.2.2.2	Trajectoire entre deux points avec des points intermédiaires.....	43
2.2.3	Planification de la trajectoire dans l'espace opérationnel.....	44
2.2.3.1	Interpolation cartésienne.....	44
2.2.3.2	Interpolation articulaire.....	44
2.3	Critère de performance «Fonction objectif» dans la recherche de trajectoires.....	45
2.3.1	Critère durée de parcours.....	46
2.3.2	Critère mixte temps-efforts quadratiques.....	46
2.3.3	Critère mixte avec puissance quadratique.....	46
2.4	Les contraintes.....	47
2.4.1	Contraintes liées à la tache.....	47
2.4.2	Contraintes sur les débattements.....	48
2.4.3	Contraintes cinématiques.....	48
2.4.4	Contraintes sur les couples moteurs.....	48
2.4.5	Contraintes dues aux obstacles dans l'espace opérationnel.....	49
2.5	Conclusion.....	49
3. FORMULATION DE LA METHODE DE RESOLUTION		
3.1	Introduction.....	50
3.2	Présentation de l'approche.....	50
3.3	La mise en œuvre de l'approche proposée.....	53

3.3.1 Reformulation des équations.....	53
3.3.1.1 Traitement des contraintes du problème.....	53
3.3.1.2 Traitement de la fonction objectif.....	57
3.3.1.3 Traitement des critères de performances avec frottements.....	61
3.3.1.3.1 Modèle dynamique avec frottements.....	61
3.3.1.3.2 Contrainte avec frottements.....	61
3.3.1.3.3 Critère mixte temps-efforts quadratiques avec frottements.....	62
3.3.2 Modélisation du chemin et du mouvement.....	63
3.3.3 Technique d'optimisation.....	64
3.3.3.1 Algorithmes Génétiques.....	65
3.3.3.2 Algorithme utilisé des algorithmes génétiques.....	66
3.3.3.3 Avantages des algorithmes génétiques.....	67
3.4 Récapitulation de la méthode de résolution.....	67
3.5 Conclusion.....	69

4. MISE EN ŒUVRE ET DISCUSSION DES RESULTATS

4.1 Introduction.....	70
4.2 Premier exemple: Robot IBM 7535 B04.....	70
4.3 Deuxième exemple: Robot IBM 7535 B04.....	77
4.4 Troisième exemple: Robot planaire (3R).....	79
4.5 Quatrième exemple: Robot PUMA 560 (4R).....	85
4.6 Cinquième exemple: Robot PUMA 560 (6R).....	88
4.7 Discussion des résultats.....	93
4.8 Conclusion.....	95

CONCLUSION GENERALE.....	96
--------------------------	----

APPENDICE

LISTE DES SYMBOLES

REFERENCES

LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX

Figures	Titre	Pages
1.1	Chaîne cinématiques des robots manipulateurs	15
1.2	Schéma d'une articulation prismatique	16
1.3	Schéma d'une articulation rotoïde	16
1.4	Architecture des porteurs	20
1.5	Repérage utilisé en robotique	22
1.6	Repérage de Denavit-Hartenberg	24
1.7	Bilan des efforts au niveau articulaire	36
2.1	Génération de mouvement dans l'espace articulaire	40
2.2	Génération de mouvement dans l'espace articulaire	40
2.3	Evaluation des positions, vitesses et accélérations sur l'articulation j avec une loi trapèze	43
2.4	Description des différentes phases pour une tâche "Pick § Place"	43
3.1	Allure de la fonction $J_Q(T)$ pour un profil $Q(\xi)$ donné (Cas ou $\alpha = 0$)	58
3.2	Allure de la fonction $J_Q(T)$ pour un profil $Q(\xi)$ donné (Cas ou $\beta = 0$)	59
3.3	Position de T_m relativement aux bornes de l'intervalle $[I_{ad}^{inf}, I_{ad}^{sup}]$	60
3.4	Allure générale de la fonction $J_Q(T)$ pour un critère mixte avec puissance quadratique	60
3.5	Allure de la fonction $J_Q(T)$ pour un profil $Q(\xi)$ donné avec introduction du frottement	62
3.6	Représentation du profil du chemin avec N_c+2 points de contrôle	64
3.7	Représentation du profil de mouvement avec N_m+2 points de contrôle	64
3.8	Organigramme des Algorithmes Génétiques utilisé	66
3.9	Organigramme de résolution	68
4.1	Robot industriel IBM 7535 B04 type SCARA simplifié	71
4.2	Résultats de l'exemple 1 correspondants aux $\alpha=1$, et $N_p =4$	72
4.3	Résultats de l'exemple 1 correspondants aux $\alpha=1$, et $N_p =5$	72
4.4	Résultats de l'exemple 1 correspondants aux $\alpha=1$, et $N_p =6$	73
4.5	Résultats de l'exemple 1 correspondants aux $\alpha=0.75$	74
4.6	Résultats de l'exemple 1 correspondants aux $\alpha=0.5$	75
4.7	Résultats de l'exemple 1 correspondants aux $\alpha=0.25$	76
4.8	Résultats de l'exemple 2 correspondants aux $\alpha=1$	77
4.9	Positions et vitesses articulaires trouvée par [82] avec $\alpha=1$	78

4.10	Couples moteurs trouvée par [82] avec $\alpha=1$	78
4.11	L'allure du mouvement optimisé correspondant à $\alpha=1$	72
4.12	L'allure du mouvement optimisé trouvée par [82] avec $\alpha=1$	78
4.13	Robot manipulateur planaire à trois articulations pivot	79
4.14	Résultats de l'exemple 3 pour $\alpha=1, \beta = 1$ on trouve $T_Q = J_Q = 4.77$	80
4.15	Résultats de l'exemple 3 pour $\alpha=0.75, \beta = 1$ on trouve $T_Q = 5.14,$ $J_Q = 4.69$	81
4.16	Résultats de l'exemple 3 pour $\alpha=0.5, \beta = 1$ on trouve $T_Q = 5.49,$ $J_Q = 3.66$	82
4.17	Résultats de l'exemple 3 pour $\alpha=0.3, \beta=0.5$ on trouve $T_Q = 5.89,$ $J_Q = 2.07$	83
4.18	Trajectoires optimales trouvées par [79]	84
4.19	Commandes optimales trouvées par [79]	84
4.20	Résultats de l'exemple 4 pour $\alpha=0.1$ on trouve $T_Q = 0.87s$	86
4.21	Commande optimale trouvée par [83], pour $\alpha = 0.1, T_Q = 0.83s$	86
4.22	Aspect optimal de mouvement de robot PUMA 560 (4R)	87
4.23	Robot manipulateur 6R de type PUMA 560	88
4.24	Résultats de l'exemple 5 pour $\alpha = 0.5, \beta = 1$ on trouve $J_Q = 1.427, T_Q = 2.755$	90
4.25	Résultats trouvée par [79], pour $\alpha = 0.5, \beta = 1$ $J_Q = 1.370, T_Q = 2.675$	91
4.26	Aspect optimal de mouvement de robot PUMA 560 (6R)	92
4.27	Reproductibilité des résultats du deuxième exemple	95

Tableaux	Titre	pages
3.1	Les intervalles admissibles de T liés aux contraintes couples du système.	56
4.1	Paramètres de robot manipulateur IBM 7535	70
4.2	Paramètres géométriques et inertiels du robot planaire 3R	79
4.3	Capacités technologiques du robot planaire 3R	79
4.4	Tâche à réaliser par le robot 4R PUMA 560	85
4.5	Paramètres du robot 4R PUMA 560	85
4.6	Tâche à réaliser par le robot 6R PUMA 560	89
4.7	Limites technologiques du robot 6R PUMA 560	89
4.8	Paramètres géométriques et inertiels du robot 6R PUMA 560	89

INTRODUCTION GENERALE

Le robot manipulateur est devenu un outil de travail par excellence dans différents secteurs qui touchent un large spectre des activités humains, et il n'a cessé d'augmenter ces dernières années, allant de l'industrie manufacturière, passant par l'agriculture, la nucléaire, la médecine, jusqu'à la conquête de l'espace. Cette présence très remarquable du robot manipulateur dans ces nombreux domaines peut être expliquée par sa versatilité, sa dextérité et sa précision dans l'exécution des différentes tâches considérées comme aliénantes, fatigantes ou dangereuse pour l'être humain.

En effet, l'utilisation courante des robots manipulateurs dans l'exécution des tâches répétitives fait que les recherches se sont naturellement orientées vers la planification de mouvements qui cherchent en général une meilleure gestion de la rapidité de l'exécution et de la fiabilité du fonctionnement du manipulateur.

En résumé, le problème de planification d'un mouvement optimal le long d'une tâche de transfert est formulé mathématiquement sous forme d'un problème d'optimisation sous contrainte dans lequel l'objectif est de minimiser une quantité physique représentée par un critère de performance tout en respectant certaines limitations et exigences de nature technologique liées au robot manipulateur, à la tâche et à l'environnement. La résolution de ce problème conduit à la détermination de la trajectoire optimale que le robot manipulateur doit suivre ainsi que les commandes optimisées qui permettent d'engendrer le mouvement le long de cette trajectoire.

Selon la tâche que doit accomplir le robot, on peut classer les différents mouvements selon deux catégories de trajectoires: l'une libre, l'autre imposée [6] [51] [52].

○ Trajectoire imposée:

De tels mouvements sont rencontrés lorsque l'outil en fin de chaîne agit sur son environnement sans interruption et selon un parcours déterminé. Il est nécessaire dans ce cas, de spécifier la trajectoire de l'effecteur dans l'espace opérationnel. C'est le cas par exemple les travaux de découpage ou de soudure en continu.

○ Trajectoire libre:

Appelée aussi mouvement point à point ou mouvement de transfert. Dans ce genre de mouvement, seules les configurations initiale et finale doivent être respectées en plus des obstacles à éviter. La trajectoire qui relie ces deux configurations est alors libre. Ceci peut être le cas de la manutention d'objets ou de soudure point à point.

Dans les deux cas, il existe généralement plusieurs trajectoires possibles. Il faut profiter de cette multiplicité de choix pour adopter la meilleure solution afin d'accomplir la tâche. Pour les mouvements libres, l'optimisation a pour but de rechercher la trajectoire à suivre ainsi que les modalités pour la parcourir tandis que pour les mouvements imposés, l'optimisation ne porte que sur les modalités de la parcourir.

Position de problème:

En toute généralité et de manière intuitive, le problème de la planification de mouvement pour un robot manipulateur peut se formuler de la façon suivante:

Etant donné un robot A quelconque placé dans un espace de travail W tout aussi quelconque. Etant donné p_i la position initiale de A et p_f la position finale où l'on désire amener A . Trouver un mouvement qui permette de faire passer A de p_i à p_f . Ce mouvement doit être exécutable par A dans l'espace W tout on prend en considération les divers aspects tels que:

- La prise en compte des caractéristiques physiques de A et de W (contraintes cinématiques et dynamiques, masse, gravité, frottement, etc.).
- L'optimisation d'un nombre défini de critères (temps, énergie, etc.).

- La prise en compte de la dimension temporelle pour l'évitement d'obstacles fixes ou mobiles.
- La prise en compte d'un espace de travail partiellement connu, voire inconnu, qui peut rendre nécessaire l'alternance entre planification et exécution dans le but d'obtenir les informations manquantes.

État de l'art:

De nombreux travaux dans le domaine de la robotique ont été ainsi consacrés à l'étude du problème de planification des mouvements des robots manipulateurs. Ces travaux cherchent, en général, une meilleure gestion de la rapidité de l'exécution et de la fiabilité du fonctionnement du manipulateur. La plupart de ces travaux tentent de planifier des mouvements de qualité en respectant certaines limitations caractéristiques liées à la capacité du manipulateur, à la tâche à réaliser et à l'environnement dans lequel évolue le robot.

L'efficacité et le fonctionnement rationnel du robot sont souvent traduits par un critère de performance à minimiser durant le transfert. La recherche des mouvements qui minimisent ce critère est conditionné par le respect des exigences imposées par les limitations géométriques, cinématiques, dynamiques et technologiques propres au manipulateur. La prise en compte de l'encombrement de l'espace de travail est un autre aspect complexe du problème de planification des mouvements. Elle consiste en général à définir des mouvements assurant l'évitement de toute collision avec d'éventuels obstacles lors de l'exécution de la tâche. Des contraintes supplémentaires dues à la nature de la tâche sont également à prendre en compte.

De nombreux travaux dans la littérature robotique tentent de proposer des approches permettant la planification, d'une façon optimale, des différents mouvements des bras manipulateurs. Les diverses méthodes proposées se distinguent entre elles par l'un des quatre facteurs suivants:

- ◆ La nature des tâches à exécuter : à savoir que la trajectoire à parcourir est libre ou imposée.
- ◆ La modélisation du comportement du robot: qui peut être dépassé la modélisation cinématique à la modélisation dynamique.
- ◆ Le critère d'optimisation considéré et les contraintes prise en compte.

- ◆ La technique employée pour la résolution du problème d'optimisation.

La première classe des problèmes de planification concerne les mouvements imposés, pour cela plusieurs approches de résolution ont été proposées en tenant compte la nature de modélisation adoptée cinématique ou dynamique.

Le modèle cinématique permet d'exprimer les vitesses opérationnelles en fonction des vitesses articulaires et présente l'avantage d'en être linéaire, la plupart des travaux sur la modélisation cinématique traitent le problème du suivi de trajectoire par l'effecteur, cette trajectoire est définie d'une manière purement géométrique dans l'espace opérationnel de sorte que l'effecteur accomplit la tâche sans entrer en collision avec les obstacles qui peuvent encombrer l'espace de travail. Pour certain travaux [53] [54] et [55], les auteurs s'intéressent principalement à l'évitement d'obstacles lors du suivi de trajectoire ainsi ils proposent la minimisation d'un critère dont le contenu est représentatif de la profondeur d'interpénétration des éléments en collision, l'évitement des obstacles est ainsi assuré par un processus d'optimisation, Pour [62] ont comparé différentes technique d'optimisation pour un temps de parcours minimal et sous des contraintes cinématiques et dynamiques pour un mouvement imposé. Tandis que [63] et [64] ont présenté une méthodologie pour la planification optimale de trajectoire des robots manipulateurs, dans [63] les trajectoires sont représentées par des fonctions B-Splines, et dans [64] les auteurs ont représentés les trajectoires par des fonctions splines cubiques et par b-splines ensuite les deux représentations ont été comparé sous de différentes contraintes cinématiques et dynamiques un critère mixte qui représente le temps d'exécution et le jerk est minimisé.

Cependant la planification de tels mouvements peut conduire à des vitesses d'exécution très élevées, c'est le cas notamment lorsque le temps de parcours est minimisé. Il apparait alors dans les mouvements réels des forces d'inertie qui peuvent atteindre de grandes valeurs. Le modèle cinématique qui est par définition une description mathématique du mouvement sans prise en compte des causes, ignore les forces ainsi que les caractéristiques inertielles de la chaîne articulée. Ainsi, dans certaines situations, il est obligatoire de passer à un niveau plus élevé pour une description plus réaliste du comportement du robot manipulateur d'où une modélisation dynamique s'impose car elle permet d'établir

les relations entre les paramètres de positions, de vitesses d'accélération articulaires et les efforts actionneurs pour assurer un tel mouvement. Cependant, le traitement du problème de planification des mouvements sur la base d'une modélisation dynamique est beaucoup plus compliqué que celui basé sur la modélisation cinématique. Pour cela plusieurs travaux essaient de cerner et de résoudre cette difficulté en utilisant des techniques standards et connues, ou en proposant parfois des techniques spécifiquement conçues pour ce type de problèmes. Les auteurs de [58] ont utilisé une méthode dite plan de phase pour la recherche d'un mouvement à durée de parcours minimale lorsque la trajectoire est prédéfinie. [59] ont amélioré cette méthode et étudié certains cas particuliers qui n'ont pas été abordés dans la formulation originale. D'autres auteurs [60] [61] proposent un critère mixte minimisant à la fois le temps de parcours et les efforts actionneurs, ce critère permet d'approcher la solution en temps minimal tout en conservant une continuité en accordant par le biais d'une pondération, une importance prédominante à la minimisation du temps de parcours. On évite ainsi les commandes de type bang-bang qui génèrent des mouvements comportant des à-coups qui peuvent être gênants pour le bon fonctionnement du robot manipulateur. D'autres techniques moins spécifiques ont été également conçues pour résoudre le même problème, Dans [65] l'auteur propose une procédure pour résoudre ce problème de planification de trajectoire optimale le long d'un mouvement imposé cette méthode basée sur une modélisation géométrique inverse, ensuite le mouvement est déterminé à travers des points intermédiaires tout en respectant les diverses contraintes imposées et par la suite une technique d'optimisation stochastique est utilisée.

La deuxième classe des problèmes de planification, concerne les mouvements libres ou à trajectoire non spécifiée. Pour ce type de problème très peu de recherches s'intéressent au comportement cinématique du bras manipulateur, Nous citons les travaux de [66] [67] qui développent une méthode du champ de potentiel afin de déterminer des trajectoires sans collision dans un espace opérationnel encombré, d'autres auteurs [68] [69] ont préoccupé à l'exécution des mouvements avec un temps de parcours optimal lorsque ce mouvement n'est pas imposé en présence d'obstacles. Tandis que dans [70] on introduit un critère

assurant l'optimisation des efforts actionneurs délivrés pendant l'exécution de la tâche.

Même pour les mouvements libres, la modélisation dynamique des manipulateurs reste la plus réaliste et la plus adéquate, pour cela la majorité des travaux est fondée sur la base d'une modélisation dynamique. Dans ce type de problèmes, la trajectoire est définie par le processus d'optimisation de telle sorte que le mouvement optimal permette au robot de suivre cette trajectoire en vérifiant un critère imposé. Ce critère doit inclure de façon significative la dynamique du robot.

Les auteurs de [71] considèrent un mouvement point à point pour robot manipulateur flexible dont lequel ont utilisé les algorithmes génétiques pour minimiser le temps d'exécution et / ou les vibrations, d'autres auteurs [72] traitent le problème de planification des trajectoires dans l'espace articulaire et opérationnel pour un mouvement libre, ainsi ils proposent la minimisation d'un critère mixte avec une considération des contraintes cinématiques et dynamique de robot manipulateur, par contre dans [73] proposent de résoudre le même problème avec l'influence de nombre des nœuds sur la trajectoire et en minimisant l'énergie des actionneurs. Et les auteurs de [74] [75] [76] et [77] qui proposent une technique d'optimisation pour résoudre le problème de planification de trajectoire dans un espace de travail encombré d'obstacles fixes et mobiles. D'autres auteurs [78] traitent le problème de planification de trajectoire des robots manipulateurs en considérant la dynamique des systèmes, ils s'intéressent à la minimisation d'une fonction objectif qui comporte plusieurs critères sous différentes contraintes cinématiques et dynamiques. Les auteurs de [79] proposent une nouvelle approche qui est basée sur la normalisation de l'échelle de temps pour traiter ce problème de planification de trajectoire, la minimisation de la fonction objectif s'effectue en utilisant une technique déterministe SQP et de différentes contraintes cinématiques, dynamiques, obstacles ont été imposées sur le mécanisme. Les auteurs de [80] [81] ont utilisés le Principe de Maximum de Pontriaguine PMP pour résoudre ce genre de problème dans un espace de travail libre et encombré d'obstacles.

Dans le cadre de ce mémoire, il est impossible de passer en revue tous les problèmes qui sont liés à la planification de mouvement dans toute sa généralité.

Aussi nous limiterons nous à l'étude des aspects pertinents compte tenu du problème particulier que l'on pose.

Notre mémoire a été organisé comme suit:

Le premier chapitre traite des généralités de la robotique, la terminologie utilisée, les différentes modélisations géométriques, cinématiques et dynamiques des bras manipulateurs. Et le deuxième chapitre aborde la planification des trajectoires et la génération de mouvement ainsi qu'une synthèse de la génération des trajectoires dans l'espace articulaire et opérationnel. En suite le troisième chapitre présente l'approche de résolution qu'on a adoptée ainsi que les formes de trajectoire et la technique d'optimisation choisis. Enfin, le quatrième chapitre se penche sur la mise en œuvre de l'approche avec des applications sur plusieurs robots manipulateurs planaires et spatiaux qui subissent différentes contraintes. La conclusion générale reflète fidèlement les résultats des chapitres suscités avec des analyses et des commentaires, suivi par des perspectives envisagées.

CHAPITRE 1

GENERALITES ET TERMINOLOGIE

1.1. Introduction:

Ce premier chapitre est consacré aux généralités, nous donnons une définition du terme " ROBOT " et une présentation des systèmes mécaniques articulés ainsi qu'un aperçu général sur la morphologie et la description des systèmes mécaniques articulés avec la terminologie des termes couramment utilisés. De même, on traite les différentes modélisations des bras manipulateurs.

1.2. Définitions générales:

D'après **(AFNOR) l'association Française de Normalisation**, 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 de façon cyclique et peuvent être adaptées à d'autres fonctions sans modification permanente du matériel". [1]

Le dictionnaire Robert propose comme définition : Mécanisme automatique pouvant se substituer à l'homme pour effectuer certaines opérations, et capable d'en modifier de lui-même le cycle et d'exercer un certain choix, (par détection photoélectrique des "cerveaux" électroniques de servomoteurs etc.).

La J.I.R.A. (Japan industrial Robot Association), quant à elle met plus l'accent sur la notion d'automorphisme. En donnant comme définition (1980), le dispositif

versatile et flexible offrant des fonctions de déplacement similaires à celle des membres humains ou dont les fonctions de déplacement sont commandées par ses capteurs et ses moyens de reconnaissance.

La R.I.A. (Robotic Industries Association) aux Etats Unis, insiste sur l'utilisation en proposant (1979) "(manipulateur multifonction reprogrammable conçu pour déplacer des matériaux, des pièces, des outils ou des dispositifs spécialisés de déplacements variables et programmables pour accomplir diverses tâches)". [2]

1.3. Constituants mécaniques des robots manipulateurs:

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

Sous le terme **organe terminal**, on regroupe tout dispositif destiné à manipuler des objets ou à les transformer. Dans la suite du mémoire on utilisera indifféremment le terme organe terminal, préhenseur, outil ou effecteur pour nommer le dispositif d'interaction fixé à l'extrémité mobile de la structure mécanique.

Le rôle de **la structure mécanique articulée** est d'amener l'organe terminal dans une situation (position et orientation) donnée, son architecture est une chaîne cinématique de corps généralement rigides, assemblés par des liaisons appelées articulations. On distingue 4 types de chaînes polyarticulées:

- à chaîne ouverte, Figure I-1-a.
- à chaîne arborescente, Figure I-1-b.
- parallèle, Figure I-1-c.
- complexe à chaîne principale ouverte, Figure I-1-d. [3]

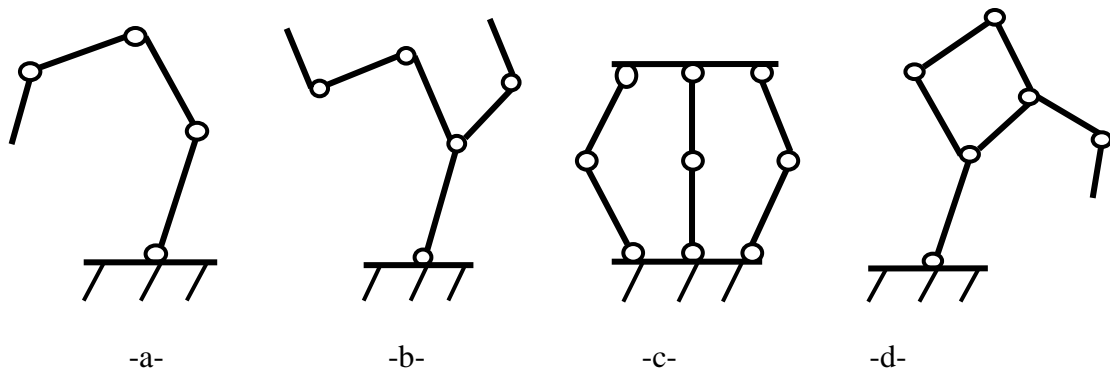


Figure 1.1 : Chaîne cinématiques des robots manipulateurs

1.4. Description des robots:

1.4.1. Articulation (ou joint):

Une articulation lie deux segments (corps) successifs en limitant le nombre de degrés de liberté de l'un par rapport à l'autre. Soit m le nombre de degrés de liberté résultant appelé aussi **mobilité** de l'articulation ($0 \leq m \leq 6$). En robotique, dans la majorité des cas, $m = 1$. Ce qui revient à dire que l'articulation est simple. On distingue deux types d'articulations simples:

- Articulation prismatique (P ou T) : type glissière. Le mouvement entre les deux corps se réduit à une translation le long d'un axe commun, La situation relative entre les deux corps est mesurée par la distance x le long de l'axe. Et elle est modélisée de la manière suivante :

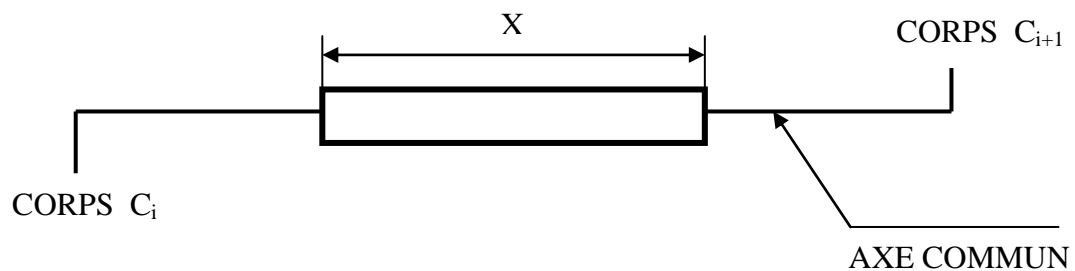


Figure 1.2 : Schéma d'une articulation prismatique

- Articulation rotoïde (R): type pivot. Le mouvement entre les deux corps se réduit à une rotation autour d'un axe commun, La situation relative entre les deux corps est donnée par l'angle θ autour de l'axe de l'articulation. Et elle est modélisée de la manière suivante :

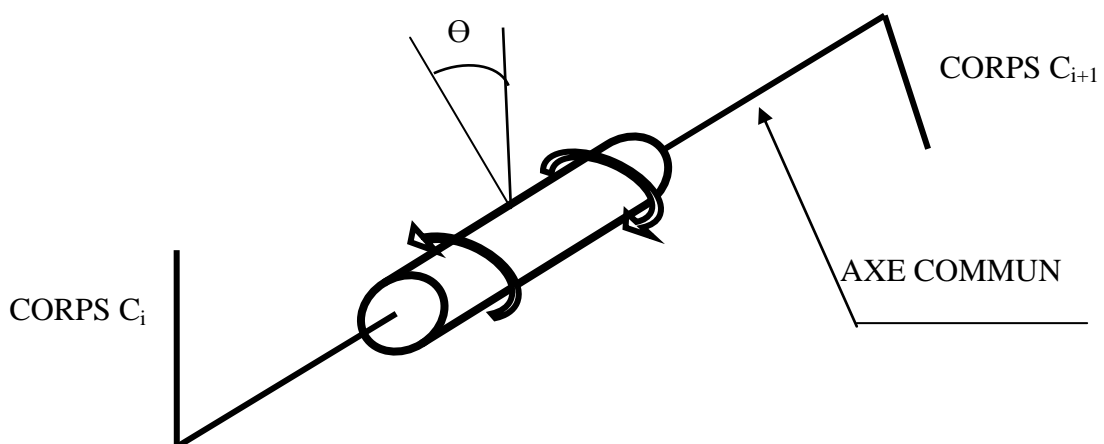


Figure 1.3 : Schéma d'une articulation rotoïde

1.4.2. Mobilité du robot:

C'est le nombre de variables indépendantes qui définissent la position du robot à un instant donné, la mobilité est également le nombre d'actionneurs du robot, c'est ce que l'on appelle le nombre d'axes de chaque actionneur commandant un axe, (m) est un nombre entier (en général $3 \leq m \leq 7$).

1.4.3. Degré de la tâche D_r :

C'est le nombre de paramètres indépendants qui permettent de définir la tâche c'est à dire la situation possible de l'organe terminal (position et orientation).

Lorsque $D_r < m$ il y a deux possibilités:

- $D_r < m$ pour toute configuration, le robot est dit redondant. C'est-à-dire qu'il y a plus d'axes qu'il ne faudrait pour effectuer la tâche. Cette redondance peut permettre d'augmenter le volume de travail (robot monté sur un chariot guidé par rails).
- $D_r < m$ pour certaines configurations particulières du robot appelées singulières. Le robot est dit localement redondant, les singularités peuvent poser des problèmes de commande. Par exemple, lorsque deux axes vont se trouver alignés. [2] [4]

1.4.4. Coordonnées articulaires:

Les coordonnées articulaires appelées aussi coordonnées généralisées sont les variables (angle ou longueurs) qui définissent la configuration du robot à un instant donné, elles sont fonction du temps, leur nombre est égale à n . on utilise souvent pour désigner ces variables la lettre q munie d'un indice ($q_1, q_2, q...$). On choisit généralement pour variables q celles qui sont directement commandées par les actionneurs (rotatif ou translation).

1.4.5. Coordonnées opérationnelles:

Les coordonnées opérationnelles sont les variables qui définissent la tâche. Si celle-ci consiste à déplacer l'outil terminal (cas le plus fréquent), ce sont les variables qui définissent la position et l'orientation de cet outil par rapport à un repère fixe lié au bâti du robot. On choisit le plus souvent les coordonnées cartésiennes d'un point particulier de l'organe terminal (point outil) pour la position. Pour l'orientation, on peut choisir trois angles (les angles d'Euler par exemple) ou

les composantes de la matrice de rotation, ou encore, les paramètres d'Euler. Le nombre de coordonnées opérationnelles indépendantes est égal à D_r . Dans le cas d'une tâche de positionnement d'un objet, $D_r \leq 6$.

1.4.6. Espace articulaire (espace de configuration):

L'espace articulaire d'un robot est celui dans lequel est représentée la situation de tous ses corps. La solution consiste à utiliser des variables ou coordonnées articulaires, L'espace de ces variable est appelé espace articulaire, sa dimension est égale au nombre de degrés de liberté de la structure mécanique.

Pour les structures à chaînes ouvertes simple ou arborescentes, les variables articulaires sont généralement indépendantes ce qui n'est pas le cas pour les structures à chaînes fermées pour lesquelles il existe des relations entre les variables.

1.4.7. Espace opérationnel (espace de travail):

L'espace opérationnel est celui dans lequel est représentée la situation de l'organe terminal (on considère donc autant d'espaces opérationnels que d'organes terminaux). La dimension M de cet espace est au maximum égale à 6: trois variables indépendantes de position et trois variables indépendantes d'orientation. [5]

1.4.8. Redondance:

Un robot est redondant lorsque le nombre de degrés de liberté de l'organe terminal est inférieur au nombre de degrés de liberté de l'espace articulaire (nombre d'articulations motorisées). Cette propriété permet d'augmenter le volume du domaine accessible et de préserver les capacités de déplacement de l'organe terminal en présences d'obstacles.

A titre d'exemple et pour les chaînes ouvertes simples, les combinaisons qui peuvent donner des structures redondantes sont:

- nombre d'articulations motorisées > 6
- nombre d'articulations rotoïdes d'axes concourants >3
- nombre d'articulations rotoïdes d'axes parallèles >3
- nombre d'articulations prismatiques > 3

1.4.9. Configuration singulières:

Pour tous les robots, qu'ils soient redondants ou non, il se peut que dans certaines configurations dites singulières, le nombre de degrés de liberté de l'organe terminal soit inférieur à la dimension de l'espace opérationnel.

La configuration singulière ou singularités se traduisent, physiquement, par la nullité du déterminant de la matrice Jacobéenne. [2]

1.5. Morphologie des robots manipulateurs:

On peut préciser que l'étude des structures arborescentes ainsi que celle des structures fermées se ramènent à celle des chaînes ouvertes simples, les quelques considérations que nous faisons s'adressent à ces morphologies mais sont généralisables aux structures complexes.

Afin de dénombrer les différentes architectures possibles, on ne considère que 2 paramètres :

- 1- le type d'articulation (rotoïde (R) ou prismatique (P)).
- 2- l'angle que font deux axes articulaires successifs (0° ou 90° ; sauf cas très particulier, les axes consécutifs d'un robot sont soit parallèles, soit perpendiculaires).

On convient d'appeler les trois premiers degrés de liberté le porteur du robot. Les degrés de liberté résiduels forment le poignet, caractérisé par des dimensions beaucoup plus petites et une plus faible masse. [1]

Dans la pratique, on trouve les cinq structures suivantes :

- Les porteurs anthropomorphes (RRR).
- Les porteurs sphériques (RRP).
- Les porteurs toriques (RPR).
- Les porteurs cylindriques (RPP).
- Les porteurs cartésiens (PPP).

Sont schématisées dans la figure qui suit les 12 morphologies possibles de porteur (ces morphologies sont non redondantes (on élimine a priori les structures limitant les mouvements du porteur à des déplacements linéaires ou planaires : 3

liaisons prismatiques d'axes parallèles, par exemple, ou 3 liaisons rotoïdes d'axes parallèles).

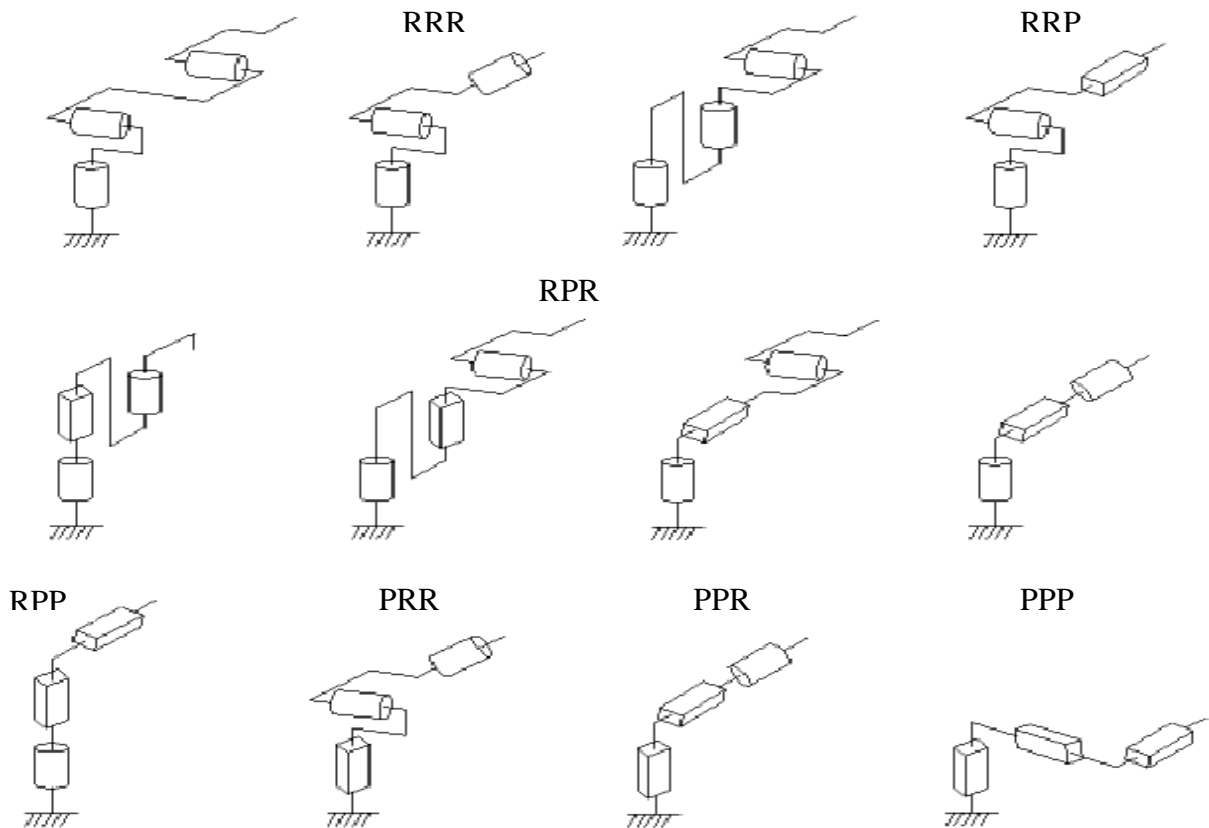


Figure 1.4 : Architecture des porteurs

1.6 Caractéristiques d'un robot:

Parmi ces caractéristiques on peut mentionner quelques-unes qui permettent de choisir un robot en fonction de l'application envisagée:

- L'espace de travail.
- La charge utile ou charge maximale transportable par le robot.
- Les vitesse et accélérations maximales, qui conditionnent le temps de cycle.
- La résolution correspond au plus petit incrément de mouvement qui peut être effectué par chaque axe ou articulation du robot.
- Les débattements articulaires et les distances inter-liaisons définissent l'ensemble des positions et orientations accessibles par le robot

manipulateur ou plus précisément par un repère lié à son effecteur. Cet ensemble est appelé volume ou espace de travail. C'est un paramètre intrinsèque au robot manipulateur et constitue un critère prédominant pour son exploitation. Plus les débattements sont importants, plus la souplesse d'emploi du robot est évidente; Cependant le choix des valeurs de ces débattements et des distances inter-liaisons est le résultat d'un compromis entre le souhait d'augmenter au maximum le champ d'action du robot manipulateur et l'exigence de précision qui passe forcément par la limitation des déformations donc par la rigidification des bras. De plus ils sont fixés de façon à ce que le robot manipulateur ne rentre pas en collision avec lui-même. [3]

Cependant, pour une tâche donnée, d'autres caractéristiques doivent être prises en compte: technique (énergie, commande, programmation...) et commerciales (cout, maintenance...). Les critères de choix sont donc parfois difficiles à formuler et souvent contradictoires, ce qui ne facilite pas la synthèse.

1.7 Modélisation des bras manipulateurs:

La conception et la commande des robots nécessitent le développement et la mise en œuvre de modèles mathématiques en mesure de traduire le comportement physique d'un système polyarticulé. On définit:

- Les modèles de transformation entre l'espace opérationnel (dans lequel est définie la situation de l'organe terminal) et l'espace articulaire (dans lequel est définie la configuration du robot). On distingue:
 - Les modèles géométriques, direct et inverse, qui expriment la situation de l'organe terminal en fonction des variables articulaires et inversement.
 - Les modèles cinématique, direct et inverse, qui expriment la vitesse de l'effecteur en fonction des vitesses articulaires et inversement.
- Les modèles dynamiques définissent les équations de mouvement du robot, qui permettent d'établir les relations entre les couples ou forces

exercées par les actionneurs et les positions, vitesses et accélérations articulaires.

Les modélisations sont donc de trois types: géométrique, cinématique et dynamique. Seuls les robots présentant une structure en forme de chaîne ouverte seront abordés. [3]

1.7.1 Modélisation géométrique:

En robotique, on associe à tout élément du poste de travail un ou plusieurs repères, ces repères sont généralement définis de telle sorte que leurs axes correspondent à des directions et leurs origines et à des points privilégiés ayant un rôle fonctionnel lors de l'exécution de la tâche. Sur la figure suivante on a représenté les principaux repères utilisés en robotique.

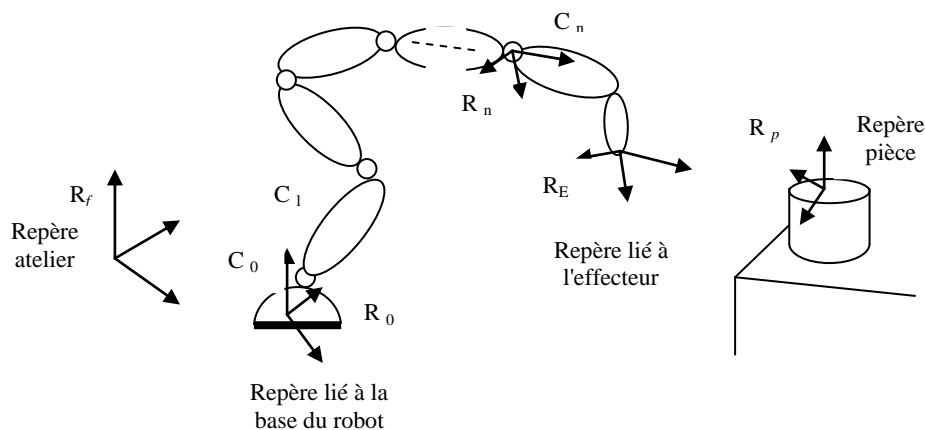


Figure I.5 : Repérage utilisé en robotique

On distingue alors,

R_f : référentiel atelier ou site.

R_0 : référentiel lié à la base du robot, il est défini par rapport à R_f .

R_E : référentiel lié à l'effecteur du robot, il est défini par rapport à R_0 .

R_p : référentiel lié à la pièce (tache), il est défini par rapport à R_f .

Cette liste n'est pas exhaustive et les notations peuvent changer. Une tâche assignée au robot consisterait en générale à faire coïncider R_E avec R_p . [3]

1.7.1.1 Repérage de Denavit-Hartenberg (D-H):

La modélisation des robots de façon systématique et automatique exige une méthode adéquate pour la description de leur morphologie. Plusieurs méthodes et notations ont été proposées, La plus répandue est celle de *J.Denavit* et *R.S.Hartenberg* modifiée plus tard par *W.Khalil* et *J.F.Kleifinger*. [3]

Pour analyser le comportement mécanique de la structure d'un bras manipulateur, il est nécessaire de lier un repère orthonormé à chaque de ces corps et attaché un référentiel R_0 à la base du bras manipulateur. L'étude des mouvements des corps revient alors à étudier les mouvements des repères.

Ce repérage convient aussi bien pour les chaînes ouvertes simples, complexes, arborescentes et fermées

Le repère R_i lié au corps C_i est défini selon les règles et les notations de Denavit-Hartenberg par:

- Les corps sont supposés rigides, reliés par des articulations (rotoïdes ou prismatiques) idéales, sans jeu mécanique et non élastiques.
- q_i est la variable de l'articulation i .
- L'axe z_i est porté par l'axe de l'articulation A_i .
- L'axe x_{i-1} 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 (figure 1.6).

- d_i : distance entre z_{i-1} et z_i le long de l'axe x_{i-1} (projection orthogonale de $\overline{O_{i-1}O_i}$ sur l'axe x_{i-1}).
- α_i : angle formé par les axes z_{i-1} et z_i qui correspond à la rotation autour de x_{i-1}
- r_i : distance entre x_{i-1} et x_i le long de l'axe z_i (projection orthogonale de $\overline{O_{i-1}O_i}$ sur l'axe z_i).
- θ_i : angle formé par les axes x_{i-1} et x_i qui correspond à la rotation autour de z_i .

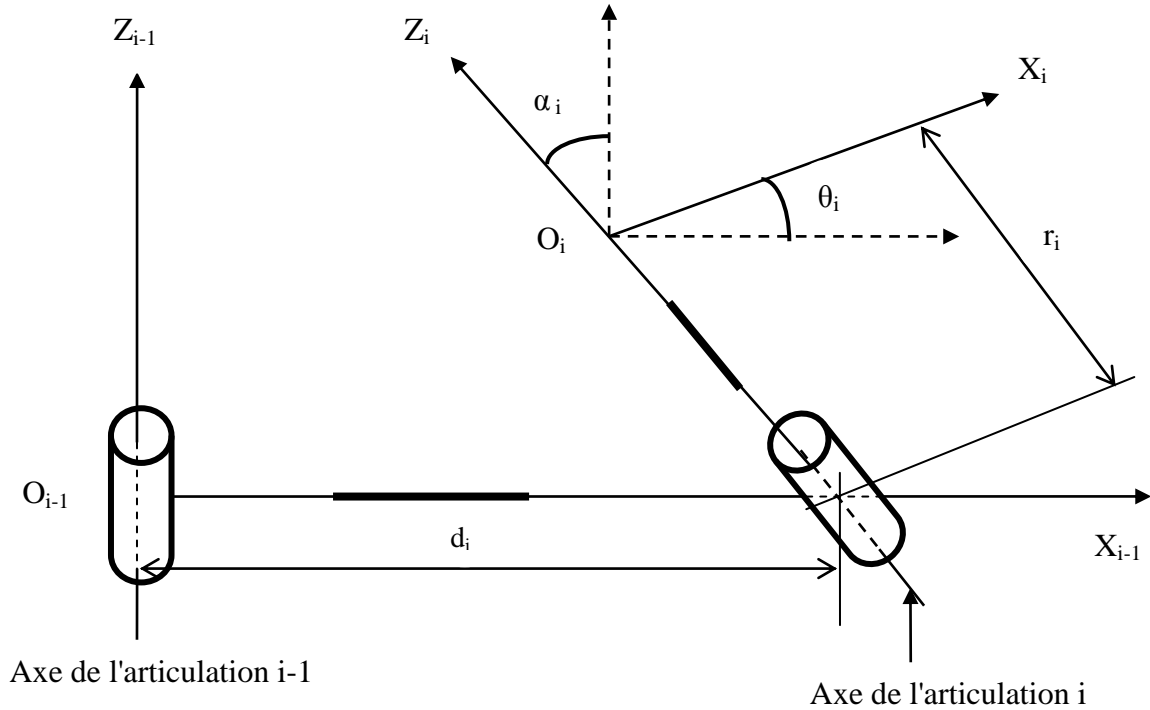


Figure 1.6 : Repérage de Denavit-Hartenberg

Si on prend q_i comme coordonnée généralisée de l'articulation A_i , cela veut dire que si :

A_i est rotoïde : $q_i = \theta_i$

A_i est prismatique : $q_i = r_i$

Pour résumer cela, on introduit la variable binaire σ_i telle que :

$$q_i = \bar{\sigma}_i \theta_i + \sigma_i r_i \quad \text{avec} \quad \bar{\sigma}_i = 1 - \sigma_i$$

$\sigma_i = 1$ si A_i est prismatique.

$\sigma_i = 0$ si A_i est rotoïde.

En définitive et en utilisant tous les résultats précédents:[5]

$${}^{i-1}T_i = \text{Rot}(x, \alpha_i) \cdot \text{Trans}(x, d_i) \cdot \text{Rot}(z, \theta_i) \cdot \text{Trans}(z, r_i) \quad (1.01)$$

Tout calcul fait, on obtient la matrice de transformation homogène (4 x 4) qui correspond à cette modélisation et qui permet le passage du repère R_i au repère R_{i-1} :

$${}^{i-1}T_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & d_i \\ c\alpha_i s\theta_i & c\alpha_i c\theta_i & -s\alpha_i & -r_i s\alpha_i \\ s\alpha_i s\theta_i & c\alpha_i c\theta_i & c\alpha_i & r_i c\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.02)$$

Remarque : on peut écrire la matrice ${}^{i-1}T_i$ de la manière suivante :

$${}^{i-1}T_i = \begin{bmatrix} {}^{i-1}A_i & & & \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{ou: } {}^{i-1}A_i = \text{rot}(x, \alpha_i) \cdot \text{rot}(z, \theta_i)$$

La matrice de transformation qui donne le passage de R_{i-1} à R_i sera:

$${}^i T_{i-1} = \text{Trans}(z, -r_i) \cdot \text{Rot}(z, -\theta_i) \cdot \text{Trans}(x, -d_i) \cdot \text{Rot}(x, -\alpha_i) \quad (1.03)$$

$${}^i T_{i-1} = \begin{bmatrix} & & -d_i s\theta_i \\ & {}^{i-1}A_i^T & -d_i c\theta_i \\ & & -r_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.04)$$

1.7.1.2 Modélisation géométrique directe (MGD):

C'est l'ensemble des relations qui permettent d'exprimer la situation de l'organe terminal en fonction des coordonnées articulaires. Ces variables sont définies par la matrice unicolonne $q = [q_1, q_2, \dots, q_n]^T$. La situation de l'organe terminal est définie par la matrice unicolonne formée de m coordonnées telle que:

$$X = [X_1, X_2, \dots, X_m]^T \quad \text{avec } m \leq 6.$$

n: nombre de coordonnées articulaires.

m: nombre de coordonnées opérationnelles.

Le modèle géométrique direct est défini par l'équation :

$$X = F(q) \quad (1.05)$$

Si la structure est ouverte simple ou arborescente, le calcul de X implique celui de l'organe terminal. Si la structure est fermée, il faut ajouter les relations liant les variables articulaires pour la fermeture de chaîne. [5] [3]

1.7.1.3 Modélisation géométrique inverse (MGI):

Le modèle géométrique inverse consiste à utiliser les coordonnées articulaires pour calculer les coordonnées opérationnelles (positions désirées de l'organe terminal).

C'est l'inverse du MGD :

$$q = F^{-1}(X) \quad (1.06)$$

En général, on n'a pas unicité de la solution au problème du MGI. En pratique, 3 cas peuvent se présenter :

Cas1 : Lorsqu'on a limitation du nombre de degrés de liberté et de la dimension des segments, l'organe terminal ne pouvant pas accéder à une position désirée, alors, il n'y a pas de solution au problème.

Cas 2 : Lorsque le robot se trouve dans une configuration singulière (redondance locale ou vis à vis de la tâche), l'organe terminal ne pouvant pas effectuer certaines tâches (limitation des déplacements et/ou des rotations) alors, il perd un certain nombre de degrés de liberté et les solutions au problème sont infinies.

Cas 3 : les solutions à la forme explicite de l'équation $q = F^{-1}(X)$ sont en nombre fini.

En général, pour résoudre le MGI, il n'existe pas de méthodes analytiques mais d'autres formes de résolution adaptées à des classes cinématiques données [6] [7]. (Méthode de PIEPER, méthode de PAUL, etc....). Ce modèle présente un grand intérêt pour les robots programmés par apprentissage ou langage de haut niveau.

1.7.2 Modélisation cinématique (différentiel):

1.7.2.1 Modèle différentiel direct (MDD):

Il décrit les variations élémentaires des coordonnées opérationnelles en fonction des variations élémentaires des coordonnées articulaires:

$$dX = J(q) dq \quad (1.07)$$

Le terme $J(q)$ désigne le jacobien (matrice de dimension $m \times n$) :

$$J(q) = \frac{\partial x}{\partial q}$$

Ce modèle permet de déterminer par itération les coordonnées opérationnelles en fonction des coordonnées articulaires. Si on remplace le calcul différentiel par le calcul des dérivées par rapport au temps, le MDD devient :

$$\dot{X} = J(q) \dot{q} \quad (1.08)$$

Ou: \dot{X} représentent les vitesses opérationnelles.

\dot{q} Représentent les vitesses articulaires.

Sous forme matricielle l'équation devient:

$$[\dot{X}] = [J_{ij}] [\dot{q}] \quad (1.09)$$

Ou:

$$J_{ij} = \frac{\partial F_i(q)}{\partial q_j} \quad i = (1, \dots, m) \quad \text{et} \quad j = (1, \dots, n)$$

1.7.2.2 Modèle différentiel inverse (MDI):

A partir d'une configuration q , on calcule la différentielle articulaire dq pour satisfaire une différentielle des coordonnées opérationnelles dX imposée. Si les différentielles sont remplacées par les dérivées par rapport au temps, on constitue le modèle cinématique inverse.

$$[\dot{q}] = [J_{ij}]^{-1} [\dot{X}] \quad (1.10)$$

Pour obtenir le modèle différentiel inverse, on peut soit dériver le modèle géométrique inverse (PAUL) (méthode analytique) soit inverser le modèle différentiel direct d'une manière plus générale (résolution analytique ou numérique).

1.7.3 Modélisation dynamique:

Elle exprime une équation mathématique qui donne la relation entre les couples (et/ou les moments) appliqués aux actionneurs et les positions, les vitesses et les accélérations articulaires. Elle est de la forme :

$$\Gamma = g(q, \dot{q}, \ddot{q}, F) \quad (1.11)$$

Où :

- Γ : vecteur des couples (et/ou forces) des actionneurs (On considère un couple pour une articulation rotoïde et une force pour une articulation prismatique).
- q : vecteur des positions articulaires.
- \dot{q} : vecteur des vitesses articulaires.
- \ddot{q} : vecteur des accélérations articulaires.
- F : forces extérieures (et couples) exercées par l'organe terminal.

Conventionnellement, on dira que la relation (1.11) représente le modèle dynamique inverse ou tout simplement le modèle dynamique. Alors le modèle dynamique direct est celui qui exprime les accélérations en fonction des positions, des vitesses et des couples des actionneurs.

$$\ddot{q} = h(q, \dot{q}, \Gamma, F) \quad (1.12)$$

Pour obtenir le modèle dynamique des robots, on dispose de deux formalismes qui sont les plus souvent utilisés. Il s'agit de :

- ❖ Formalisme de LAGRANGE.
- ❖ Formalisme de NEWTON – EULER.

Nous allons utiliser les notations suivantes:

a_j : vecteur unitaire sur l'axe x_j .

F_j : résultante des forces extérieures sur le corps C_j .

f_j : force extérieure sur C_j par son antécédent en O_j .

f_{ij} : force exercée par le corps C_j sur l'environnement en O_j .

g : accélération de la pesanteur.

G_j : centre de gravité du corps C_j .

r_j : tenseur d'inertie du corps C_j par rapport au repère parallèle à R_j d'origine G_j .

${}^j r_j$: tenseur d'inertie du corps C_j par rapport au repère R_j .

$$\text{Ou: } {}^j r_j = \begin{bmatrix} \iiint (y^2 + z^2) dm & - \iiint (xy) dm & - \iiint (xz) dm \\ - \iiint (xy) dm & \iiint (x^2 + z^2) dm & - \iiint (yz) dm \\ - \iiint (xz) dm & - \iiint (yz) dm & \iiint (x^2 + y^2) dm \end{bmatrix}$$

$$\text{On le note aussi: } {}^j r_j = \begin{bmatrix} xx_j & xy_j & xz_j \\ xy_j & yy_j & yz_j \\ xz_j & yz_j & zz_j \end{bmatrix}$$

l_j : vecteur $O_{j-1} O_j$ qui lie les origines des repères R_{j-1} et R_j .

m_j : masse du corps C_j .

N_j : résultante des moments extérieurs exercés sur C_j autour de G_j .

n : couple exercé sur C_j par son antécédent.

n_{ij} : couple exercé par C_j sur l'environnement.

ρ_j : vecteur $O_j G_j$.

v_j : vecteur vitesse absolue du point O_j .

v_{gj} : vecteur vitesse absolue du point G_j .

ω_j : vecteur vitesse de rotation du point G_j .

\dot{v}_j : vecteur accélération absolue du point O_j .

\dot{v}_{gj} : vecteur accélération absolue du point G_j .

$\dot{\omega}_j$: vecteur accélération de rotation du point G_j .

1.7.3.1 Formalisme de LAGRANGE:

Il décrit les équations en termes de travail et d'énergie du système qui donne ce qui suit quand l'effort extérieur sur l'organe terminal est supposé nul. [7] [8] [9]

$$\Gamma_i = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i}; \quad i = (1, \dots, n) \quad (1.13)$$

L est la fonction de LAGRANGE égale à la différence entre l'énergie cinétique totale (E) et l'énergie potentielle totale (U) du système. L'expression de l'énergie cinétique est:

$$E = \frac{1}{2} \dot{q}^T A \dot{q} \quad (1.14)$$

A : matrice d'inertie du robot d'éléments génériques A_{ij}

A_{ij} : sont fonction des variables articulaires q .

Avec : $i = (1, \dots, n)$ et $j = (1, \dots, n)$.

Le lagrangien s'écrit :

$$L = \frac{1}{2} \dot{q}^T A \dot{q} - U \quad (1.15)$$

Comme l'énergie potentielle est aussi fonction des variables articulaires q , alors d'après l'équation (1.13) et après dérivation:

$$\Gamma_i = [A_{i1}, \dots, A_{in}] \ddot{q} + \dot{q}^T \left[\frac{\partial A_{i1}}{\partial q}, \dots, \frac{\partial A_{in}}{\partial q} \right] \dot{q} - \frac{1}{2} \dot{q}^T \left[\frac{\partial A}{\partial q_i} \right] \dot{q} + \frac{\partial u}{\partial q_i} \quad (1.16)$$

Ou encore:

$$\Gamma_i = [A_{i1} \ddot{q}_1 + \dots + A_{in} \ddot{q}_n] + [B_i^{12} \dot{q}_1 \dot{q}_2 + \dots + B_i^{1n} \dot{q}_1 \dot{q}_n + B_i^{23} \dot{q}_2 \dot{q}_3 + \dots + B_i^{(n-1)n} \dot{q}_{n-1} \dot{q}_n] + [C_{i1} \dot{q}_1^2 + \dots + C_{in} \dot{q}_n^2] + Q_i \quad (1.17)$$

Elle s'écrit sous forme matricielle de la manière suivante:

$$\Gamma = A \ddot{q} + B \dot{q} \dot{q} + C \dot{q}^2 + Q \quad (1.18)$$

B : matrice des termes de Coriolis d'éléments génériques B_i^{jk} de dimension $[n \times n (n-1)/2]$.

C : matrice des termes centrifuges d'éléments génériques C_{ij} de dimension $(n \times n)$.

$$\dot{q} \dot{q} = [\dot{q}_1 \dot{q}_2, \dots, \dot{q}_1 \dot{q}_n, \dot{q}_2 \dot{q}_3, \dots, \dot{q}_{n-1} \dot{q}_n]^T$$

$Q = [Q_1, \dots, Q_n]^T$: vecteur des efforts dus aux effets de gravité.

On identifie les expressions des équations (1.16) et (1.17):

$$B_i^{jk} = \frac{\partial A_{ij}}{\partial q_k} + \frac{\partial A_{ik}}{\partial q_j} - \frac{\partial A_{jk}}{\partial q_i} \quad (1.19)$$

$$C_{ij} = \frac{\partial A_{ij}}{\partial q_i} - \frac{1}{2} \frac{\partial A_{ij}}{\partial q_i} \quad (1.20)$$

$$Q_i = \frac{\partial A_{ij}}{\partial q_i} \quad (1.21)$$

Où A , B , C et Q représentent les coefficients dynamiques du robot qui sont fonction des paramètres géométriques et inertiels du mécanisme. Avec les équations d'un système mécanique articulé, on forme un système de n équations différentielles couplées, non linéaires et du second ordre. [5]

Propriétés des coefficients dynamiques :

Termes de Coriolis:

$$B_i^{jk} = \frac{\partial A_{ij}}{\partial q_k} + \frac{\partial A_{ik}}{\partial q_j} - \frac{\partial A_{jk}}{\partial q_i} \quad (1.22)$$

$$B_i^{jl} = -B_l^{ij} \quad \text{si } j \leq i \text{ et } j \leq 1$$

$$B_i^{jj} = -B_j^{ij} \quad \text{si } j \leq i$$

$$B_i^{jk} = 0 \quad \text{si } j \leq 1$$

Termes centrifuges:

$$C_{ij} = \frac{\partial A_{ij}}{\partial q_i} - \frac{1}{2} \frac{\partial A_{ij}}{\partial q_i} \quad (1.23)$$

Avec :

$$C_{ij} = -\frac{1}{2} B_i^{ij} \quad \text{si } j > i$$

$$C_{ii} = 0$$

Pour calculer les dérivées et dérivées partielles du lagrangien, on utilise la méthode d'UICKER qui fait appel aux notions essentielles suivantes :

- La notion de matrice de passage homogène.
- La notion de matrice d'inertie homogène.

On prend un système à chaîne cinématique simple et un corps S_i qui fait partie de cette chaîne. Soient les repères successifs $R_i (O_i, x_i, y_i, z_i)$ et M_i un point de S_i tels que:

$$\begin{aligned}\overline{O_0M_i} &= [O_0M_i] \\ \overline{O_iM_i} &= [O_iM_i] \\ \overline{O_0M_i} &= [{}^0T_i][O_0M_i]\end{aligned}\quad (1.24)$$

Avec : $\|O_iM_i\| = \text{cste}$ si le corps S_i est rigide.

En dérivant l'équation (1.24), on obtient:

$$V^0(M_i) = \frac{d^0}{dt} [O_0M_i] = \frac{d^0}{dt} \left(\begin{bmatrix} {}^0T_i \\ [O_iM_i] \end{bmatrix} \right) \quad (1.25)$$

Quant à l'énergie cinétique, elle sera égale à :

$$E_{ci} = \frac{1}{2} \int [V^0(M_i)]^2 dm \quad (1.26)$$

Alors:

$$E_{ci} = \frac{1}{2} \text{Trace} \sum_{k=1}^i \sum_{l=1}^i \text{Trace} \left(\begin{bmatrix} \frac{\partial {}^0T_i}{\partial q_k} \\ \left[\int [O_iM_i][O_iM_i]^T dm \right]_{sk} \end{bmatrix} \begin{bmatrix} \frac{\partial {}^0T_i}{\partial q_l} \\ \left[\frac{\partial {}^0T_i}{\partial q_l} \right] \end{bmatrix} \right) \dot{q}_k \dot{q}_l \quad (1.27)$$

$$[J_i] = \int_{S_i} [O_iM_i][O_iM_i]^T dm$$

Représente la pseudo-matrice d'inertie d'ordre (4x4) relative à S_i dans le repère R_i de la chaîne cinématique considérée. Elle est constituée par :

Le moment d'ordre 0 de S_i qui est en fait m_i (masse de S_i).

Le moment d'ordre 2 de S_i qui est $m_i \times \overline{O_iG_i}$ (G_i est le centre de masse de S_i).

Le moment d'ordre 2 de S_i qui est la matrice d'inertie $[I_i]$ en O_i dans le repère R_i .

$$[I_i] = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}$$

$$[U_i] = \begin{bmatrix} \frac{1}{2}(-I_{xx} + I_{yy} + I_{zz}) & -I_{xy} & -I_{xz} & m\bar{x} \\ -I_{xy} & \frac{1}{2}(I_{xx} - I_{yy} + I_{zz}) & -I_{yz} & m\bar{y} \\ -I_{xz} & -I_{yz} & \frac{1}{2}(I_{xx} + I_{yy} - I_{zz}) & m\bar{z} \\ m\bar{x} & m\bar{y} & m\bar{z} & m \end{bmatrix} \quad (1.28)$$

L'expression de E_{ck} sera alors:

$$E_{ci} = \frac{1}{2} \text{Trace} \sum_{k=1}^i \sum_{l=1}^i \text{Trace} \left(\left[\frac{\partial^0 T_i}{\partial q_k} \right] [U_i] \left[\frac{\partial^0 T_i}{\partial q_l} \right] \right) \dot{q}_k \dot{q}_l \quad (1.29)$$

Comme l'énergie cinétique totale est:

$$E_c = \sum_1^n E_{ci} \quad \text{donc:}$$

$$E_c = \frac{1}{2} \text{Trace} \sum_{i=1}^n \sum_{k=1}^i \sum_{l=1}^i \text{Trace} \left(\left[\frac{\partial^0 T_i}{\partial q_k} \right] [U_i] \left[\frac{\partial^0 T_i}{\partial q_l} \right] \right) \dot{q}_k \dot{q}_l$$

$$E_c = \frac{1}{2} \text{Trace} \sum_{k=1}^i \sum_{l=1}^i \left[\sum_{i=\max(k,l)}^n \text{Trace} \left(\left[\frac{\partial^0 T_i}{\partial q_k} \right] [U_i] \left[\frac{\partial^0 T_i}{\partial q_l} \right] \right) \right] \dot{q}_k \dot{q}_l \quad (1.31)$$

On pose:

$$A_{kl} = \sum_{i=\max(k,l)}^n \text{Trace} \left(\left[\frac{\partial^0 T_i}{\partial q_k} \right] [U_i] \left[\frac{\partial^0 T_i}{\partial q_l} \right] \right)$$

Alors:

$$E_c = \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^n A_{kl} \dot{q}_k \dot{q}_l$$

$$E_c = \frac{1}{2} \left[\dot{q} \right] [A] \left[\dot{q} \right] \quad (1.32)$$

On peut donner les expressions des coefficients dynamiques :

Expression de la matrice d'inertie:

Les éléments qui constituent la matrice [A] sont exprimés de la manière suivante :

$$A_{kl} = \sum_{i=\max(k,l)}^n \text{Trace} ([\Omega_{ok} {}^0T_i][J_i] [\Omega_{ol} {}^0T_i]^T) \quad (1.33)$$

Expression de la matrice des termes de Coriolis:

$$B_{i,kl} = \sum_{i=\max(k,l)}^n \text{Trace} ([\Omega_{ok} \ \Omega_{ol} {}^0T_i][J_i] [\Omega_{ok} {}^0T_i]^T) \quad (1.34)$$

Expression de la matrice des termes centrifuges:

$$C_{kl} = \sum_{i=\max(k,l)}^n \text{Trace} ([\Omega_{ol} \ \Omega_{ol} {}^0T_i][J_i] [\Omega_{ol} {}^0T_i]^T) \quad (1.35)$$

Expression du vecteur force de gravité:

L'expression de l'énergie potentielle étant

$$E_p = \sum_{i=1}^n (-m_i g T_i u_i)$$

Avec:

$$G_i = -\frac{\partial E_p}{\partial q_i}$$

Donc on a:

$$G = \sum_{i=1}^n -m_i g \left[\frac{\partial {}^0T_i}{\partial q_i} \right] \quad \text{ou encore:} \quad G_i = \sum_{i=1}^n -m_i g {}^0T_i \Omega_{oi}$$

Avec: $\Omega_{oi} = {}^0T_i \Omega_{ii} {}^0T_i$ et

$$\Omega_{ii} = \begin{bmatrix} 0 & -(1 - \sigma_i) & 0 & 0 \\ (1 - \sigma_i) & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_i \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$\sigma_i = 0$ si la liaison est rotoïde.

$\sigma_i = 1$ si la liaison est prismatique.

1.7.3.2 Formalisme de NEWTON-EULER:

Il est adapté à la construction du modèle dynamique inverse. Il permet le dimensionnement de la structure et des actionneurs. Le caractère itératif de ce formalisme réduit le temps de calcul par rapport au formalisme de LAGRANGE [1] [7] [8] [9]. Il est basé sur une double récurrence : une récurrence avant de la base du robot vers l'effecteur en utilisant la formule de composition pour calculer les vitesses et accélérations donc, le torseur dynamique et une récurrence arrière de l'effecteur vers la base pour calculer les couples des actionneurs en exprimant le bilan des efforts pour chaque corps. La composition des vitesses donne :

$$\omega_j = \omega_{j-1} + \bar{\sigma}_j \dot{q}_j a_j \quad (1.22)$$

$$v_j = v_{j-1} + \omega_{j-1} \wedge L_j + \sigma_j \dot{q}_j a_j \quad (1.23)$$

On dérive ces 2 expressions par rapport au temps pour obtenir la composition des accélérations:

$$\dot{\omega}_j = \dot{\omega}_{j-1} + \bar{\sigma}_j (\ddot{q}_j a_j + \omega_{j-1} \wedge \dot{q}_j a_j) \quad (1.24)$$

$$\dot{v}_j = \dot{v}_{j-1} + \dot{\omega}_{j-1} \wedge L_j + \omega_{j-1} (\omega_{j-1} \wedge L_j + \sigma_j \dot{q}_j a_j) + \sigma_j (\ddot{q}_j a_j + \omega_{j-1} \wedge \dot{q}_j a_j) \quad (1.25)$$

On arrange l'expression (1.25) comme suit:

$$\dot{v}_j = \dot{v}_{j-1} + \dot{\omega}_{j-1} \wedge L_j + \omega_{j-1} (\omega_{j-1} \wedge L_j) + \sigma_j (\ddot{q}_j a_j + 2\omega_{j-1} \wedge \dot{q}_j a_j) \quad (1.26)$$

On peut utiliser cette dernière expression pour déterminer la vitesse du centre de gravité, à savoir:

$$\dot{v}_{Gj} = \dot{v}_j + \omega_j \wedge (\omega_j \wedge s_j) \quad (1.27)$$

Ce qui permet d'obtenir le torseur dynamique:

$$F_j = m_j \dot{v}_{Gj} \quad (1.28)$$

$$N_j = \tau_j \dot{\omega}_j + \omega_j \wedge (\tau_j \wedge \omega_j) \quad (1.29)$$

F_j et N_j représentent respectivement la somme totale des forces extérieures et la somme totale des moments extérieurs. On initialise la récurrence par:

$$\omega_0 = 0 \quad , \quad \dot{\omega}_0 = 0 \quad \text{et} \quad \dot{v}_0 = 0$$

Bilan des efforts:

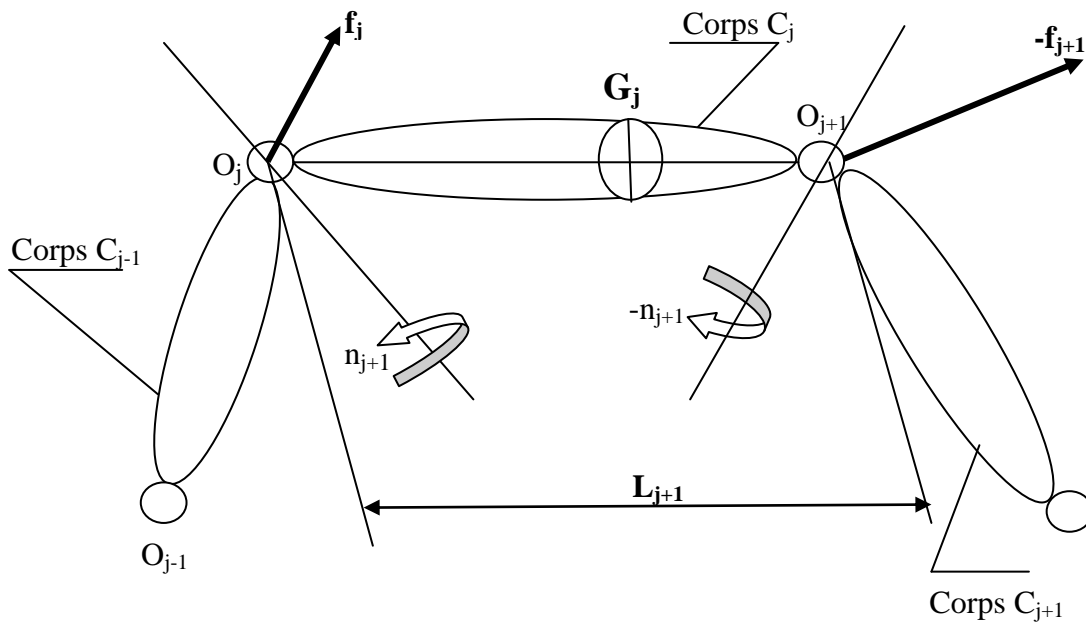


Figure 1.7 : Bilan des efforts au niveau articulaire

Les équations qui composent la récurrence arrière sont obtenues à partir du bilan des forces sur chaque corps:

$$F_j = f_j - f_{j+1} + m_j g \quad (1.30)$$

$$N_j = n_j - n_{j+1} + (s_j - L_{j+1}) \wedge f_{j+1} - s_j \wedge f_j \quad (1.31)$$

Pour faire intervenir l'effort de la gravité sans le prendre en compte dans le bilan des forces, on prend $\dot{v}_0 = -g$. Les équations deviennent:

$$f_j = F_j + f_{j+1} \quad (1.32)$$

$$n_j = N_j - n_{j+1} L_{j+1} \wedge f_{j+1} + s_j \wedge F_j \quad (1.33)$$

Cette récurrence est initialisée par les efforts f_{n+1} et n_{n+1} notés respectivement f_m et n_m qu'exerce l'organe terminal sur l'environnement en projetant les vecteurs f_j et n_j sur l'axe du mouvement. On obtient les couples Γ_j aux actionneurs et cela, suivant la nature de l'articulation j par la forme suivante: [5]

$$\Gamma_j = (\sigma_j f_j + \bar{\sigma}_j n_j) z_i \quad (1.34)$$

Cette forme de représentation est surtout orientée vers la construction des algorithmes de calcul, elle peut être réécrite aussi sous la forme suivante:

$$\Gamma_i = \sum_{j=1}^n M_{ij}(q) \ddot{q}_j + C_i(q, \dot{q}) + G_i(q) \quad i = 1, \dots, n \quad (1.35)$$

Où les coefficients M_{ij} sont les composantes de la matrice masse $M(q)$ du robot, C_i sont les composantes du vecteur des forces de Coriolis et centrifuge et G_i les efforts gravitationnels.

1.8 prise en compte des frottements:

L'introduction des frottements, malgré qu'elles rendent le modèle plus réaliste, augmente encore la complexité du traitement du modèle dynamique et par conséquent la résolution du problème de planification des mouvements. Le modèle de type frottement sec (ou de Coulomb) fait l'hypothèse d'un couple constant de frottement en opposition au mouvement. La composante du frottement visqueux est considérée dans plusieurs applications, comme proportionnelle à la vitesse articulaire, l'expression du couple de frottement est donnée par [1]:

$$\Gamma_{fi} = F_{si} \text{sign}(\dot{q}_i) + F_{vi} \dot{q}_i \quad (1.36)$$

Dans cette expression le couple de frottement de l'articulation i est noté Γ_{fi} ; F_{si} et F_{vi} désignent respectivement les paramètres de frottement sec et visqueux de l'articulation i . la forme générale du modèle incluant les frottements sera donc:

$$\Gamma_i = \sum_{j=1}^n M_{ij}(q) \ddot{q}_j + C_i(q, \dot{q}) + G_i(q) + F_{si} \text{sign}(\dot{q}_i) + F_{vi} \dot{q}_i \quad (1.37)$$

1.9 Conclusion:

Ce chapitre a été consacré à la terminologie en robotique et à certaines définitions et notations de base qui reviennent souvent dans l'étude des robots manipulateurs. On présente les constituants mécaniques des bras manipulateurs, leur morphologie, leurs caractéristiques, et les modélisations géométrique, cinématique et dynamique. Le but est de comprendre et d'introduire des aspects utiles à notre démarche dans la suite de notre travail.

CHAPITRE 2

PLANIFICATION § GENERATION DE MOUVEMENT

2.1. Introduction:

On abordera dans ce chapitre le problème de planification, génération de mouvement pour les robots manipulateurs, pour cela nous allons présenter le problème de suivi de trajectoire dans l'espace opérationnel ainsi que la génération de mouvement dans l'espace articulaire, ensuite nous allons aborder le critère de performance dans la recherche de trajectoire. Et enfin nous présentons les différentes contraintes et exigences liées à la capacité du manipulateur car la recherche des mouvements qui minimisent un tels critère est conditionnée par le respect de ces limitations, ainsi que l'environnement dans lequel évolue le robot.

2.2 Espace de planification de la trajectoire:

Comme le 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;
- Le mouvement entre deux points via des points intermédiaires avec trajectoire contrainte entre les points intermédiaire;

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. [10]

2.2.1 Génération de mouvement et système de commande:

Les deux approches; génération de mouvement dans l'espace articulaire et génération de mouvement dans l'espace opérationnel dont schématiser sur les figurent suivantes:

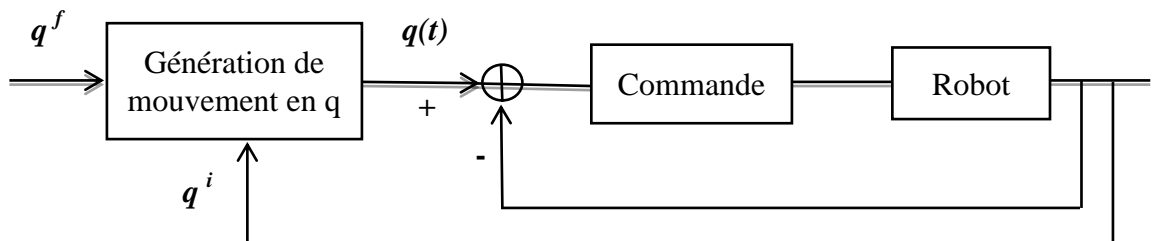


Figure 2.1 : Génération de mouvement dans l'espace articulaire

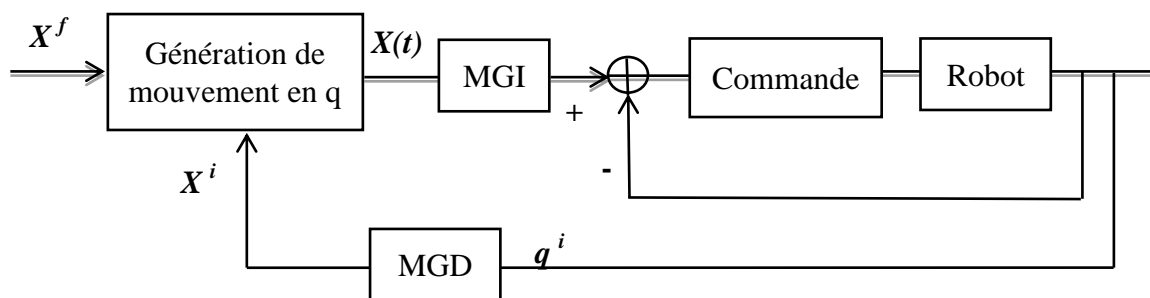


Figure 2.2 : Génération de mouvement dans l'espace opérationnel

La génération de mouvement dans l'espace articulaire présente plusieurs avantages :

- elle nécessite moins de calcul en ligne, puisqu'il n'y a pas d'appel au modèle géométrique ou cinématique inverse,
- le mouvement n'est pas affecté par le passage sur les configurations singulières,
- les contraintes de vitesse et de couples maximaux sont connues avec précision puisqu'elles correspondent aux limites physiques des actionneurs.

En contrepartie, la géométrie de la trajectoire dans l'espace opérationnel ne peut être imposée. Entre 2 points donnés, l'organe terminal se déplace de façon imprévisible mais répétitive, ce qui peut occasionner des collisions lorsque le robot

évolue dans un environnement encombré). Ce type de mouvement est par conséquent approprié pour réaliser des déplacements rapides dans un espace dégagé.

La génération de mouvement dans l'espace opérationnel permet de contrôler la géométrie de la trajectoire (mouvement rectiligne par exemple). Par contre :

- 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,
- elle peut être mise en échec chaque fois que les points de la trajectoire engendrée ne sont pas dans le volume accessible du robot ou chaque fois que la trajectoire impose une reconfiguration du mécanisme (changement d'aspect en cours de trajectoire).

Le choix d'une méthode de génération de mouvement dépend de l'application considérée. Chaque approche a ses propres limites, inhérentes au fait que les contraintes sont exprimées soit dans l'espaces articulaire (vitesses, couples, butées) soit dans l'espace opérationnel (précision, obstacles).

2.2.2 Génération 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 initiales et finale. On distingue k_v et k_a les vecteurs des vitesses et accélérations articulaires maximales, ces derniers étant caractéristiques techniques des actionneurs utilisés.

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

$$\begin{cases} q(0) = q^i \\ q(t_f) = q^f \end{cases} \quad (2.1)$$

2.2.2.1 Trajectoire sans contraintes ou mouvement point à point:

Dans le tableau suivant, on récapitule les différentes fonctions permettant de satisfaire le (2.1) avec: $D_j = (q^f)_j - (q^i)_j$.

Fonction d'interpolation	Expression horaire	Temps de transfert global minimum
Interpolation linéaire	$q_j(t) = (q^i)_j + \frac{t}{t_f} D_j$	$(t_f)_j = \frac{ D_j }{(k_v)_j}$
Polynôme de degré trois	$q_j(t) = a_0 + a_1 t + a_2 \cdot t^2 + a_3 \cdot t^3$ Pour satisfaire les conditions aux limites, nous avons: $a_0 = q^i; \quad a_1 = 0; \quad a_2 = \frac{3}{t_f^2} \cdot D_j \quad ;$ $a_3 = \frac{3}{t_f^3} \cdot D_j$	$(t_f)_j = \text{Max} \left[\frac{3 D_j }{2(k_v)_j}, \sqrt{\frac{6 D_j }{(k_a)_j}} \right]$
Polynôme de degré cinq	$q_j(t) = (q^i)_j + 10 \cdot \lambda^3 - 15 \cdot \lambda^4 + 6 \cdot \lambda^5$ Avec: $\lambda = \frac{t}{(t_f)_j}$	$(t_f)_j = \text{Max} \left[\frac{15 D_j }{8(k_v)_j}, \sqrt{\frac{10 D_j }{\sqrt{3}(k_a)_j}} \right]$
Loi Bang-Bang	Pour: $0 \leq t \leq \frac{(t_f)_j}{2}$ $q_j(t) = (q^i)_j + 2 \cdot \lambda^2 D_j$ Pour: $\frac{(t_f)_j}{2} \leq t \leq t_f$ $q_j(t) = (q^i)_j + [-1 + 4 \cdot \lambda - 2 \cdot \lambda^2] \cdot D_j$ Avec: $\lambda = \frac{t}{(t_f)_j}$	$(t_f)_j = \text{Max} \left[\frac{2 D_j }{(k_v)_j}, 2 \sqrt{\frac{ D_j }{(k_a)_j}} \right]$

La loi de Bang-Bang ou " Loi de trapèze", comporte trois phases de mouvement (Accélération- Mouvement uniforme- Décélération) et il est représenté par les relations suivantes:

$$\begin{cases} q_j(t) = (q^i)_j + \frac{1}{2} t^2 \cdot (k_a)_j \cdot \text{Sign}(D_j) & \text{pour } 0 \leq t \leq \tau_j \\ q_j(t) = (q^i)_j + \left(t - \frac{\tau_j}{2}\right) \cdot (k_v)_j \cdot \text{Sign}(D_j) & \text{pour } \tau_j \leq t \leq (t_f)_j - \tau_j \\ q_j(t) = (q^f)_j - \frac{1}{2} \left[(t_f)_j - t\right]^2 \cdot (k_a)_j \cdot \text{Sign}(D_j) & \text{pour } (t_f)_j - \tau_j \leq t \leq (t_f)_j \end{cases} \quad (2.2)$$

$$\text{Avec : } \tau_j = \frac{(k_v)_j}{(k_a)_j}$$

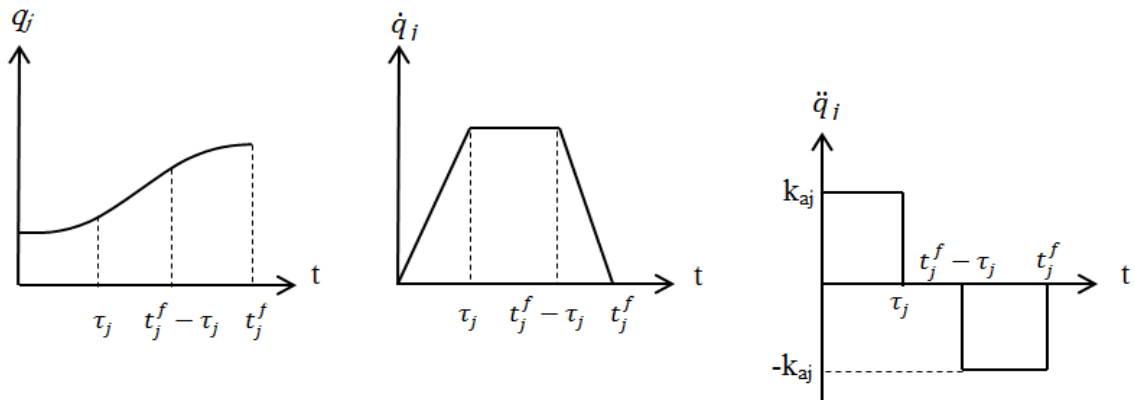


Figure 2.3: Evaluation des positions, vitesses et accélérations sur l'articulation j avec une loi trapèze

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

$$(t_f)_j = \frac{(k_v)_j}{(k_a)_j} + \frac{|D_j|}{(k_v)_j} \quad (2.3)$$

2.2.2.2 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 dépose. Dans ce dernier cas, il suffira de spécifier une position intermédiaire sur la normale à la surface de pose 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 à la dépose 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.

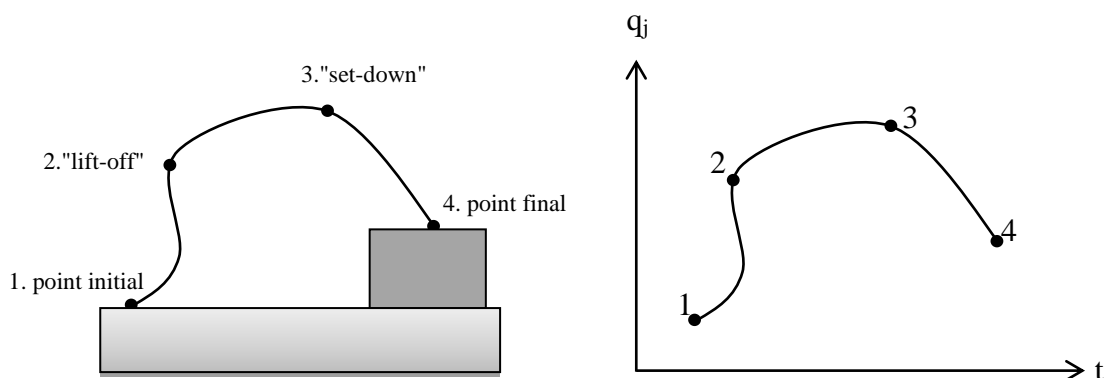


Figure 2.4: Description des différentes phases pour une tâche "Pick & Place"

Afin d'avoir un mouvement doux lors des phases, "lift-off" et "set-down", on exige que les vitesses et les accélérations soient continues et lisses. Plusieurs méthodes se basant sur l'interpolation ou les "Splines cubiques" que nous allons les utiliser prochainement.

2.2.3 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. Pour cela, deux approches sont utilisées:

2.2.3.1 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 cette approche, on cite: la méthode de Paul, et la méthode de Taylor.

2.2.3.2 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 , de les transposer dans l'espace articulaire et ensuite d'interpoler plus fin dans cet espace pour subvenir aux besoins du régulateur.

Plusieurs méthodes ont été proposées dans cette approche, on cite: la méthode de Lin, et la méthode de Taylor.

2.3 Critère de performance «Fonction objectif» dans la recherche de trajectoires:

Le choix du critère de performance est déterminant quant à la qualité du mouvement et conditionne l'efficacité des méthodes de résolution du problème. Le critère à minimiser peut se présenter sous 3 formes distinctes : une fonctionnelle de type intégrale, de type terminal ou de type mixte. Bien que formulés différemment, ces 3 problèmes sont équivalents. La fonctionnelle intégrale est la mieux adaptée pour les mouvements de transfert. Les critères à optimiser dans ce cas se mettent sous la forme générale suivante:

$$F(u, T) = \int_0^T L(x(t), u) dt \quad (2.4)$$

Où:

L : représente le lagrangien qui prend la forme selon l'objectif visé.

$x(t)$: vecteur des variables d'état à l'instant $t \in [0, T]$.

u : vecteur qui représente la commande à optimiser, considéré comme une inconnue du problème.

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

Pour le cas d'une planification de mouvements libres, u représente les couples moteurs, $x(t)$ les variables articulaires de position $q(t)$, de vitesse $\dot{q}(t)$ et d'accélération $\ddot{q}(t)$

Dans ce cas, le critère s'écrit de la manière suivante :

$$F(\Gamma, T) = \int_0^T L(q(t), \dot{q}(t), \ddot{q}(t), \Gamma) dt \quad (2.5)$$

Pour générer des mouvements de qualité, on peut citer parmi les critères les plus significatifs la durée de parcours, mixte temps - efforts quadratiques et mixte avec puissance quadratique.

2.3.1 Critère durée de parcours:

Il suscite l'intérêt d'un bon nombre de chercheurs roboticiens [6] [11] car pour les robots qui travaillent d'une manière cyclique, la rapidité d'exécution (rentabilité) est un critère très significatif. La dynamique du robot n'est pas représentée directement dans cette formulation. Ceci conduit à des commandes optimales discontinues. Les couples articulaires ne sont pas représentés dans le lagrangien d'où une forme simple comparée aux autres.

$$L(q(t), \dot{q}(t), \ddot{q}(t), \Gamma) = 1 \quad (2.6)$$

Alors:

$$F(\Gamma, T) = F(T) = \int_0^T dt = T \quad (2.7)$$

2.3.2 Critère mixte temps-efforts quadratiques:

Comparativement au critère précédant qui implique des discontinuités lors des transferts de charges, donc, nuisible au bon fonctionnement du manipulateur, ce critère est un moyen efficace pour régulariser les vitesses et les couples. En introduisant un facteur de pondération μ , on peut favoriser soit le temps, soit les efforts (couples) [6] [12]. Le lagrangien prend la forme suivante:

$$L(q(t), \dot{q}(t), \ddot{q}(t), \Gamma) = \mu + \frac{1-\mu}{2} \sum_{i=1}^n \left(\frac{\Gamma_i(t)}{\Gamma_{imax}} \right)^2 \quad : \quad \mu \in]0,1[\quad (2.8)$$

Γ_{imax} Représente le couple maximum délivré par la $i^{\text{ème}}$ articulation. Quand μ se rapproche de 0, l'optimisation se fait en faveur des couples moteurs tandis que quand μ est au voisinage de 1, on privilégie plutôt l'optimisation du temps. Le choix de μ détermine alors le compromis entre ces deux quantités. Ceci conduit à une certaine souplesse d'exécution en tenant compte de la durée de la tâche. Néanmoins, on rencontre des difficultés dans la formulation de ce critère vu l'existence des efforts dans le lagrangien.

2.3.3 Critère mixte avec puissance quadratique:

La minimisation du critère précédent permet la régularisation des couples articulaires ce qui évite tous problème de discontinuité ou de saturation. Par ailleurs, les mouvements générés peuvent associer des vitesses articulaires

élevées à des efforts actionneurs importants à des instants quelconques du transfert. Une manière de maîtriser l'évolution simultanée des vitesses et des efforts actionneurs, consiste à minimiser les puissances articulaires développées, ceci nécessite l'introduction d'un nouveau terme dans le lagrangien du critère de performance, son rôle est de quantifier les puissances articulaires à minimiser. La puissance instantanée s'écrit sous la forme :

$$p(t) = \Gamma(t) \cdot \dot{q}(t) \quad (2.9)$$

L'expression du lagrangien dans ce cas est la suivante :

$$L(q(t), \dot{q}(t), \ddot{q}(t), \Gamma) = \mu + \frac{1-\mu}{2} \left[\alpha \sum_{i=1}^n \left(\frac{\Gamma_i(t)}{\Gamma_{imax}} \right)^2 + (1-\alpha) \sum_{i=1}^n \left(\frac{P_i(t)}{P_{imax}} \right)^2 \right] \quad (2.10)$$

$: \mu \in]0,1[, \alpha \in]0,1[$

Avec $P_{imax} = \Gamma_{imax} \dot{q}_{imax}$ ou \dot{q}_{imax} est la vitesse articulaire maximale permise à la $i^{\text{ème}}$ articulation.

Le facteur α joue le même rôle que celui de μ , mais sur le terme des puissances quadratiques, il permet ainsi d'accorder plus au moins d'importance à la minimisation de ce terme.

2.4 Les contraintes:

Pour que la planification du mouvement soit possible, on tient compte des capacités technologiques du robot et des spécificités de la tâche à effectuer. Ceci implique une restriction des solutions. Pour planifier un mouvement, on s'intéresse aux contraintes suivantes:

2.4.1 Contraintes liées à la tâche:

La réalisation d'une tâche entre une configuration initiale q^{ini} et une configuration finale q^{fin} , est soumise aux conditions aux limites en positions et en vitesses suivantes:

$$q(t = 0) = q^{ini} \quad \text{et} \quad q(t = T) = q^{fin} \quad (2.11)$$

$$\dot{q}(t = 0) = 0 \quad \text{et} \quad \dot{q}(t = T) = 0 \quad (2.12)$$

2.4.2 Contraintes sur les débattements:

Du fait de la conception de la liaison et de la limitation des mouvements entre des butées mécaniques, ces derniers ne doivent pas dépasser les capacités de la structure.

Tout ceci se traduit par la formulation suivante:

$$\forall t \in [0, T] , |q_i(t)| \leq q_{imax} \quad i = 1, \dots, n \quad (2.13)$$

t : représente le temps.

$q_i(t)$: représente les coordonnées généralisées de la $i^{\text{ème}}$ articulation.

q_{imax} : représente la valeur maximale de la $i^{\text{ème}}$ articulation (valeur intrinsèque du robot).

2.4.3 Contraintes cinématiques:

La conception des articulations et de la technologie des actionneurs implique une limitation des caractéristiques cinématiques notamment pour les objets manipulés sensibles aux survitesses, accélérations et freinages (par exemple des liquides qui peuvent se déverser). Les formulations se traduisent par :

vitesses:

$$\forall t \in [0, T] , |\dot{q}_i(t)| \leq \dot{q}_{imax} \quad i = 1, \dots, n \quad (2.14)$$

Accélérations:

$$\forall t \in [0, T] , |\ddot{q}_i(t)| \leq \ddot{q}_{imax} \quad i = 1, \dots, n \quad (2.15)$$

\dot{q}_{imax} et \ddot{q}_{imax} Sont imposées par les caractéristiques techniques et la nature de la tâche à effectuer.

2.4.4 Contraintes sur les couples moteurs: [13]

Les capacités maximales des actionneurs impliquent la formule suivante:

$$\forall t \in [0, T] , |T_i(t)| \leq T_{imax} \quad i = 1, \dots, n \quad (2.16)$$

2.4.5 Contraintes dues aux obstacles dans l'espace opérationnel: [14]

Le robot travaille généralement dans des sites encombrés : sur des machines ou avec d'autres robots, il est rare qu'il soit placé dans un espace dégagé. La planification des mouvements doit bien évidemment tenir compte de l'existence de ces obstacles. Le mouvement optimal, dans ce cas, doit nécessairement éviter tout risque de collision.

Les contraintes d'évitement d'obstacles ont la forme générale suivante :

$$g[q(t)] \leq 0 \quad (2.17)$$

Où g est une fonction traduisant, en générale, la distance entre les différents corps constituant la structure du robot et les différents obstacles présents dans son environnement. Ce genre de contraintes rend le problème de planification optimale des mouvements libres d'un robot très complexe.

2.5 Conclusion:

Ce chapitre a permis de présenter différentes méthodes de planification et de génération de mouvement. Le problème a été traité tout d'abord en considérant une génération de trajectoires dans l'espace articulaire ensuite nous avons étudié la planification de trajectoires dans l'espace opérationnel. Nous avons vu aussi que le choix d'un espace dépend du mode de description des trajectoires et de la façon dont sont exprimées les contraintes.

Dans le prochain chapitre nous allons présenter les principales approches d'optimisation pour résoudre ce genre de problème.

CHAPITRE 3

FORMULATION DE LA METHODE DE RESOLUTION

3.1 Introduction:

Dans ce chapitre nous allons présenter l'approche de résolution qu'on a adoptée pour résoudre le problème de planification de trajectoire libre des robots manipulateurs, nous allons utiliser cette approche avec des problèmes compliqué ou le nombre de degré de liberté devient plus en plus élevé avec des contraintes supplémentaires pour chaque cas traitée.

En effet, cette approche a été proposée par [13] pour traiter ce problème pour les robots manipulateurs à base fixe, elle consiste à transformer le problème de temps finale inconnu (difficile à résoudre) en un problème équivalent de bornes définies (plus facile à résoudre), par la suite cette méthode modélise les trajectoires par des fonctions splines cubiques et obtient des résultats satisfaisants avec la prise en compte les contraintes imposées et le critère de performance choisi; cette approche a fait par la suite évaluée et développée dans les travaux de [30] pour prendre en compte d'autre types de robot tels que les plateformes mobiles et les manipulateurs mobiles à roues, cette amélioration se manifeste spécialement dans la décomposition des trajectoires (normalisées) pour mieux distribuer les différentes contraintes, puis les modélisées par des fonctions appropriées (splines cubiques et b-spline), il est à noter que cette décomposition offre un avantage indéniable car le traitement des contraintes du problème se fera séparément et de manière séquentielle.

Notre but consiste à adopter la méthode décrite par [30] en exploitant les avantages qu'elle offre pour trouver des meilleures solutions à des problèmes de planification de trajectoires pour les robots manipulateurs en utilisant d'autres fonctions d'interpolations plus une technique d'optimisation adéquates (les algorithmes génétiques).

3.2 Présentation de l'approche:

Les inconnus d'un problème de planification de trajectoire sont:

- $q(t)$: l'évolution dans le temps du vecteur des coordonnées généralisées.

- T : le temps final de transfert.
- $\tau(t)$: l'évolution du vecteur des efforts actionneurs.

Cette approche considère que $q(t)$ et T sont les inconnues des problèmes, le vecteur $\tau(t)$ n'étant qu'une conséquence de la détermination de ces deux variables via le modèle dynamique inverse. Ainsi que cette approche compte traiter le problème de planification de trajectoire sans devoir intégrer le système d'équations différentielles issu de modèle dynamique.

Cependant, la difficulté rencontrée est la définition de la borne supérieure de l'intervalle du temps $[0, T^{best}]$ pour une trajectoire optimale $q(t)^{best}$. De ce fait, il devient intéressant d'effectuer une normalisation de l'échelle du temps en introduisant la notion de *profil de trajectoire* qui transforme le problème d'une recherche sur une intervalle à borne indéfinie $[0, T^{best}]$ vers un autre équivalent dont la borne de recherche sont complètement connue $[0, 1]$ et qui sera alors plus simple à résoudre.

Profil de trajectoire:

Soit $q(t)$ une trajectoire donnée avec un temps de transfert T quelconque. Il est toujours possible d'exprimer cette trajectoire de la façon suivante:

$$q(t) = Q(\xi(t)) = Q(\xi) \circ \xi(t) \quad (3.1)$$

Avec $\xi(t) = t / T$ est une fonction linéaire de normalisation d'échelle du temps de sorte que $\xi \in [0, 1]$. La fonction vectorielle $Q(\xi)$ constitue le profil de la trajectoire considérée, ce profil décrit complètement la succession des configurations prises par le système lorsque celui-ci exécute la trajectoire considérée depuis le départ ($\xi = 0$) jusqu'à l'arrivée ($\xi = 1$). Ce profil de trajectoire est valide s'il mène à une trajectoire qui est géométriquement faisable (aucune collision) et qui satisfait les conditions aux limites exigées. Le symbole \circ représente l'opérateur de composition de fonctions.

En conséquence, le problème difficile qui consistait à obtenir la trajectoire optimale $q(t)^{best}$ ainsi que sa durée d'exécution T^{best} , se réduit maintenant à

trouver seulement le profil optimal $Q(\xi)^{best}$ car le temps de transfert T sera traité comme un paramètre dépendant.

En effet, soit $Q(\xi)$ un profil de trajectoire donné nous pouvons décomposer ce profil en un chemin et en mouvement sur ce chemin en réécrivant $Q(\xi)$ comme une combinaison de deux fonctions paramétriques:

$$Q(\xi) = P(\lambda) \circ \lambda(\xi) \quad (3.2)$$

La fonction vectorielle $P(\lambda)$ avec $\lambda \in [0, 1]$, est une fonction indépendante du temps et elle a la même dimension que Q , Elle décrit le chemin géométrique du robot dans l'espace des coordonnées généralisées quand λ varie de façon continue sur $[0, 1]$. La configuration initiale correspond à $\lambda = 0$ et la configuration finale correspond à $\lambda = 1$.

La fonction $\lambda(\xi)$ est une fonction scalaire croissante représentant le profil du mouvement. Elle définit la manière selon laquelle le chemin $P(\lambda)$ sera parcouru dans le temps normalisé. Évidemment, un chemin faisable doit satisfaire toutes les contraintes géométriques pour $\lambda \in [0, 1]$ et il doit également satisfaire les conditions aux limites imposées sur les positions articulaires du robot manipulateur, à savoir:

$$P(\lambda = 0) = q^{ini} \quad \text{et} \quad P(\lambda = 1) = q^{fin} \quad (3.3a)$$

Cette décomposition du profil de la trajectoire présente un avantage indéniable, car les conditions aux limites peuvent être maintenant distribuées sur deux fonctions. La fonction chemin $P(\lambda)$ doit satisfaire les conditions aux limites selon (3.3a). Tandis que, la fonction mouvement $\lambda(\xi)$ doit satisfaire, d'une part les conditions suivantes:

$$\lambda(\xi = 0) = 0 \quad \text{et} \quad \lambda(\xi = 1) = 1 \quad (3.3b)$$

$$\lambda'(\xi) \geq 0 \quad (3.3c)$$

Et d'autre part, les conditions aux limites en vitesses, à savoir:

$$\lambda'(\xi = 0) = \lambda'(\xi = 1) = 0 \quad (3.3d)$$

Nous allons voir par la suite comment adapter cette approche avec les différents aspects de ce problème de planification de trajectoire.

3.3 La mise en œuvre de l'approche proposée:

La mise en œuvre de cette approche nécessite le partage du problème en trois parties importantes qui sont distinctes mais indissociables car complémentaires et interdépendantes:

- ◆ La reformulation des équations pour extraire explicitement la durée de la tâche T.
- ◆ Le choix de la technique à utiliser pour modéliser le profil de trajectoire.
- ◆ Détermination de la trajectoire optimale par une technique d'optimisation appropriée.

3.3.1 Reformulation des équations:

3.3.1.1 Traitement des contraintes du problème:

Un autre avantage qu'offre cette approche et qui constitue le fond de son application, concerne la réduction considérable des difficultés relatives au traitement des différentes contraintes ainsi que la fonction objectif pour ce problème de planification. Pour ce faire, nous reprenons la formulation de ces contraintes et de la fonction objectif présentée dans le chapitre 2 en utilisant la normalisation de l'échelle du temps (3.1).

◆ Les contraintes cinématiques:

En appliquant la normalisation (3.1) de l'échelle du temps pour une trajectoire généralisée $q(t)$ donnée, les vitesses généralisées $\dot{q}(t)$ et les accélérations généralisées $\ddot{q}(t)$ de cette trajectoire peuvent être réécrites comme suit:

$$\dot{q}(t) = \frac{\partial \xi}{\partial t} \frac{\partial Q(\xi)}{\partial \xi} = \frac{1}{T} Q'(\xi) \quad (3.4a)$$

$$\ddot{q}(t) = \frac{\partial \xi}{\partial t} \frac{\partial Q'(\xi)}{\partial \xi} = \frac{1}{T^2} Q''(\xi) \quad (3.4b)$$

Ce qui permet de reformuler les contraintes (2.12) et (2.13) liées aux capacités cinématiques du système comme suit:

- Contraintes sur les vitesses:

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

$$\Rightarrow \forall \xi \in [0, 1] \quad \frac{1}{T} |Q'_i(\xi)| \leq \dot{q}_i^{max}, \quad i = 1, \dots, n$$

$$\Rightarrow T \geq \max_{i=1, \dots, n} \left[\max_{\xi \in [0, 1]} \left(\frac{|Q'_i(\xi)|}{\dot{q}_i^{max}} \right) \right]$$

Soit: $T \geq T_V$ (3.5a)

Avec:

$$T_V = \max_{i=1, \dots, n} \left[\max_{\xi \in [0, 1]} \left(\frac{|Q'_i(\xi)|}{\dot{q}_i^{max}} \right) \right]$$

- Contraintes sur les accélérations:

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

$$\Rightarrow \forall \xi \in [0, 1] \quad \frac{1}{T^2} |Q''_i(\xi)| \leq \ddot{q}_i^{max}, \quad i = 1, \dots, n$$

$$\Rightarrow T \geq \max_{i=1, \dots, n} \left[\max_{\xi \in [0, 1]} \left(\frac{|Q''_i(\xi)|}{\ddot{q}_i^{max}} \right)^{1/2} \right]$$

Soit: $T \geq T_A$ (3.5b)

Avec:

$$T_A = \max_{i=1, \dots, n} \left[\max_{\xi \in [0, 1]} \left(\frac{|Q''_i(\xi)|}{\ddot{q}_i^{max}} \right)^{1/2} \right]$$

Les deux contraintes cinématiques sont ainsi transformées en une borne inférieure des valeurs permises pour T . Donc, pour un profil donné de trajectoires $Q(\xi)$, le temps optimal T_Q doit satisfaire les contraintes cinématiques:

$$T_Q \geq T^* \quad (3.6)$$

Avec:

$$T^* = \max_{i=1, \dots, n} \left(\max_{\xi \in [0,1]} \left[\left(\frac{|Q'_i(\xi)|}{\dot{q}_i^{max}} \right), \left(\frac{|Q''_i(\xi)|}{\ddot{q}_i^{max}} \right)^{1/2} \right] \right)$$

♦ Les contraintes dynamiques (dues aux couples):

Les commandes $U = \tau_1, \tau_2, \tau_3, \dots, \tau_n$ sont données par le modèle dynamique du robot:

$$\tau_i(t) = \sum_{j=1}^n M_{ij}(q(t)) \ddot{q}_j(t) + C_i(q(t), \dot{q}(t)) + G_i(q(t)) \quad (3.7)$$

En utilisant l'équation (3.1) et les équations des vitesses et des accélérations généralisées (3.4) dans les équations du modèle dynamique, on obtient:

$$\tau_i(\xi) = \frac{1}{T^2} \sum_{j=1}^n M_{ij}(Q(\xi)) Q''_j(\xi) + \frac{1}{T^2} C_i(Q(\xi), Q'(\xi)) + G_i(Q(\xi)) \quad (3.8)$$

Si on pose:

$$h_i(\xi) = \sum_{j=1}^n M_{ij}(Q(\xi)) Q''_j(\xi) + C_i(Q(\xi), Q'(\xi))$$

L'équation (3.8) devient:

$$\tau_i(\xi) = \frac{1}{T^2} h_i(\xi) + G_i(Q(\xi)) \quad (3.9)$$

Pour simplifier encore les calculs numériques, on divise l'équation (3.9) par τ_i^{max} .

Et on pose:

$$\bar{\tau}_i(\xi) = \frac{\tau_i(\xi)}{\tau_i^{max}} \quad ; \quad \bar{h}_i(\xi) = \frac{h_i(\xi)}{\tau_i^{max}} \quad \text{et} \quad \bar{G}_i(\xi) = \frac{G_i(Q(\xi))}{\tau_i^{max}}$$

L'équation (3.9) devient:

$$\xi \in [0, 1] : \bar{\tau}_i(\xi) = \frac{1}{T^2} \bar{h}_i(\xi) + \bar{G}_i(\xi) \quad (3.10)$$

Ce qui permet de réécrire les contraintes (2.16), liées aux capacités en couple des actionneurs du système de la manière suivante:

$$\xi \in [0, 1] : -1 \leq \frac{1}{T^2} \bar{h}_i(\xi) + \bar{G}_i(\xi) \leq 1 \quad \text{avec } i = 1, \dots, n$$

Cette double inéquation peut être réécrite comme suit:

$$-B \leq \frac{1}{T^2} \bar{h}_i(\xi) \leq A$$

Avec: $A = 1 - \bar{G}_i(\xi)$, $B = 1 + \bar{G}_i(\xi)$

La solution de cette inéquation par rapport à T pour une valeur donnée de ξ , donne un intervalle admissible pour T :

$$T = [T_{L_i}, T_{R_i}] \quad (3.11)$$

T_{L_i} et T_{R_i} Représentent respectivement la limite inférieure (left) et la limite supérieure (right), ces valeurs en fonction du signe de $\bar{h}_i(\xi)$, A et B , sont résumées dans le tableau 3.1.

Tableau 3.1: les intervalles admissibles de T liés aux contraintes couples du système.

$\bar{h}_i(\xi) \geq 0$		$\bar{h}_i(\xi) < 0$			
$A \leq 0$	$A > 0$	$B \leq 0$	$B > 0$		
	$B < 0$	$B \geq 0$	$A < 0$	$A \geq 0$	
$[T_{L_i}, T_{R_i}] = \emptyset$	$T_{L_i}(\xi) = \sqrt{\frac{\bar{h}_i(\xi)}{A}}$ $T_{R_i}(\xi) = \sqrt{\frac{\bar{h}_i(\xi)}{-B}}$	$T_{L_i}(\xi) = \sqrt{\frac{\bar{h}_i(\xi)}{A}}$ $T_{R_i}(\xi) = +\infty$	$[T_{L_i}, T_{R_i}] = \emptyset$	$T_{L_i}(\xi) = \sqrt{\frac{\bar{h}_i(\xi)}{A}}$ $T_{R_i}(\xi) = \sqrt{\frac{\bar{h}_i(\xi)}{-B}}$	$T_{L_i}(\xi) = \sqrt{\frac{\bar{h}_i(\xi)}{-B}}$ $T_{R_i}(\xi) = +\infty$

L'intersection de tous les intervalles $[T_{L_i}, T_{R_i}]$, pour $i = 1, \dots, n$ le long de la trajectoire donne un intervalle admissible pour le temps de déplacement respectant les contraintes dynamiques dues aux couples:

$$T \in [T_g, T_d] \quad (3.12)$$

Avec:

$$T_g = \max_{i=1, \dots, n} \left[\max_{\xi \in [0,1]} T_{L_i}(\xi) \right] \text{ et } T_d = \min_{i=1, \dots, n} \left[\min_{\xi \in [0,1]} T_{R_i}(\xi) \right]$$

Si on note I_{ad} l'intervalle des durées T (temps optimal) qui vérifient toutes les contraintes (cinématiques et dynamiques), alors:

$$I_{ad} = [T^*, \infty[\cap [T_g, T_d] = [I_{ad}^{inf}, I_{ad}^{sup}] \quad (3.13)$$

Quand $I_{ad} = \emptyset$ cela signifie que le système ne pourra jamais parcourir le profil de la trajectoire donnée $Q(\xi)$ sans violer l'une des contraintes du problème.

4.3.1.2 Traitement de la fonction objectif:

Nous montrons dans ce paragraphe l'effet de la reformulation sur la fonction objectif et sur les différents critères de performance cités dans le deuxième chapitre. Pour cela, en utilisant (3.4) et (3.12) dans (2.5) nous obtiendrons pour un profil Q donné, l'expression de la fonction objectif J_Q uniquement en fonction du temps:

$$J_Q(T) = T \left(\alpha + (1 - \alpha) \left(\beta \left(E_0 + \frac{E_1}{T^2} + \frac{E_2}{T^4} \right) + (1 - \beta) \left(\frac{W_0}{T^2} + \frac{W_1}{T^4} + \frac{W_2}{T^6} \right) \right) \right) \quad (3.14)$$

Avec:

$$\begin{aligned} E_0 &= \int_0^1 \sum_{i=1}^n \bar{G}_i^2 d\xi, & E_1 &= 2 \int_0^1 \sum_{i=1}^n \bar{G}_i \bar{h}_i d\xi, & E_2 &= \int_0^1 \sum_{i=1}^n \bar{h}_i^2 d\xi \\ W_0 &= \int_0^1 \sum_{i=1}^n \bar{Q}_i'^2 \bar{G}_i^2 d\xi, & W_1 &= 2 \int_0^1 \sum_{i=1}^n \bar{Q}_i'^2 \bar{G}_i \bar{h}_i d\xi, & W_2 &= \int_0^1 \sum_{i=1}^n \bar{Q}_i'^2 \bar{h}_i^2 d\xi \\ \bar{h}_i &= \frac{h_i}{\tau_i^{max}}, & \bar{G}_i &= \frac{G_i}{\tau_i^{max}}, & \bar{Q}_i' &= \frac{Q_i'}{\dot{q}_i^{max}} \end{aligned}$$

E_0, E_1, E_2, W_0, W_1 et W_2 Sont des coefficients réels indépendants du temps de transfert T . E_0, E_2, W_0 et W_2 Sont des quantités qui sont toujours positives quelle que soit l'allure du profil $Q(\xi)$ considérée.

Pour chaque profil de trajectoires $Q(\xi)$, la meilleure durée de transfert T_Q qui minimise la fonction objectif dans l'intervalle I_{ad} est:

$$J_Q(T_Q) = \min_{T \in I_{ad}} J_Q(T)$$

L'évaluation de T_Q s'effectue analytiquement ou numériquement selon la forme de la fonction $J_Q(T)$ considérée.

♦ Critère à temps minimal: ($\alpha = 1$)

Dans ce cas, l'expression de la fonction objectif $J_Q(T)$ devient:

$$J_Q(T) = T$$

L'allure de cette fonction est représentée par une droite (Figure 3.1). La valeur de T_Q correspondante est définie par la limite inférieure du domaine admissible I_{ad} .

$$T_Q = T_1 \quad (3.15)$$

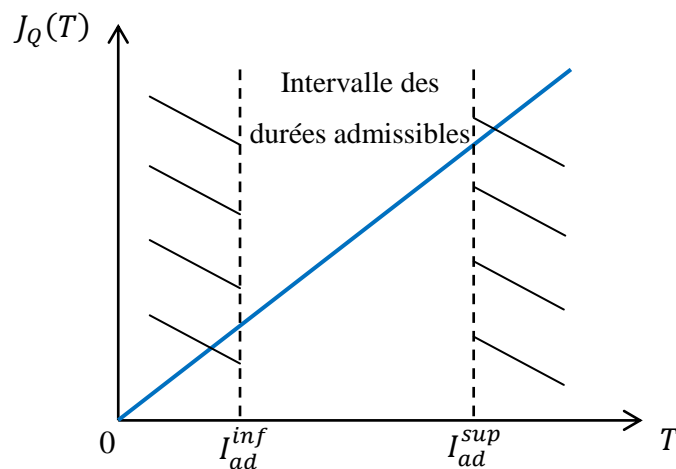


Figure 3.1: Allure de la fonction $J_Q(T)$ pour un profil $Q(\xi)$ donné (Cas ou $\alpha = 1$)

♦ Critère mixte temps efforts-quadratiques: ($\beta = 1$)

Dans le cas d'un critère mixte temps efforts-quadratiques, l'expression de la fonction objectif devient:

$$J_Q(T) = T \left(\alpha + (1 - \alpha) \left(E_0 + \frac{E_1}{T^2} + \frac{E_2}{T^4} \right) \right) \quad (3.16)$$

L'allure de cette fonction est représentée sur la figure 3.2.

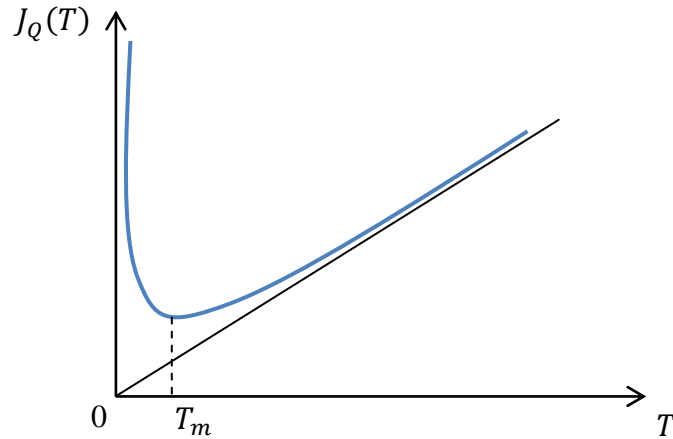


Figure 3.2: Allure de la fonction $J_Q(T)$ pour un profil $Q(\xi)$ donné (Cas ou $\beta = 1$)

Analytiquement le temps de déplacement T_m qui minimise la fonction objectif pour le profil $Q(\xi)$ considéré mais sans toutefois tenir compte des contraintes, à la fin on trouve:

$$\begin{cases} T_m = \left(\frac{E_1 + \sqrt{E_1^2 + 12(E_0 + (\alpha/1 - \alpha))E_2}}{2(E_0 + (\alpha/1 - \alpha))} \right)^{1/2} \\ T_m = \left(\frac{6E_2}{\sqrt{E_1^2 + 12(E_0 + (\alpha/1 - \alpha))E_2} - E_1} \right)^{1/2} \end{cases} \quad (3.17)$$

Les deux expressions sont équivalentes, on utilisera l'une ou l'autre selon que le signe de E_1 (E_0 et E_2 sont tous les deux positifs), est respectivement positif ou négatif de manière à réduire les erreurs d'arrondis.

Quant au temps de transfert optimal T_Q qui minimise la fonction $J_Q(T)$ sous les contraintes cinématiques et dynamiques imposées, il va être déduit selon la position de T_m par rapport à l'intervalle I_{ad} :

$$T_Q = \begin{cases} T_m & \text{si } T_m \in I_{ad} \\ T_j / J_Q(T_j) = \min(J_Q(T_j)) \quad i = 1, \dots, k & \text{si } T_m \notin I_{ad} \end{cases} \quad (3.18)$$

Si T_m n'est pas à l'intérieur de ces limites permises alors il ne coïncidera pas avec la durée T_Q optimale pour le profil Q considéré, pour cela, nous avons trois cas distinctes (a, b et c) à examiner. La durée de transfert optimale T_Q sera alors, respectivement:

$$T_Q = I_{ad}^{sup}, \quad T_Q = T_m, \quad T_Q = I_{ad}^{inf}.$$

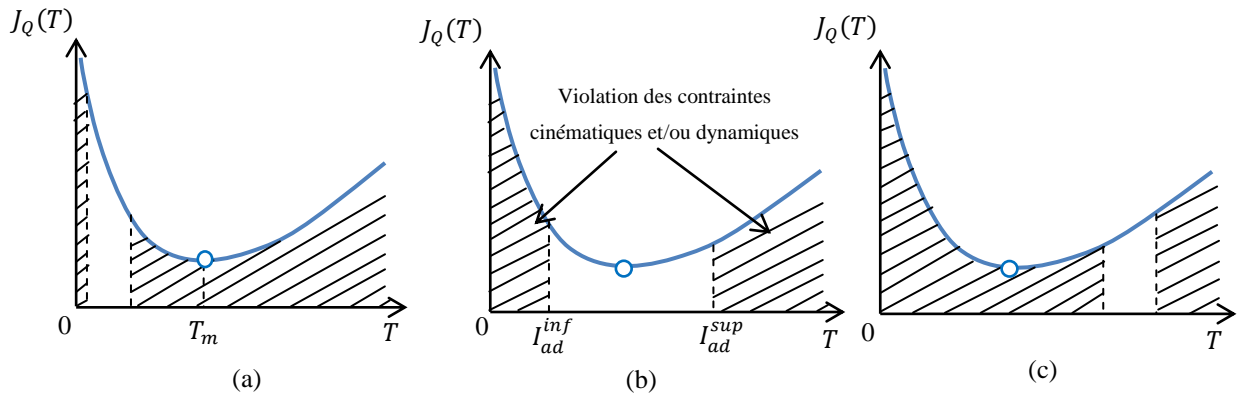


Figure 3.3: Position de T_m relativement aux bornes de l'intervalle $[I_{ad}^{inf}, I_{ad}^{sup}]$

◆ Critère mixte avec puissance quadratiques:

La fonction $J_Q(T)$ correspondante est donnée par l'expression (3.14), On prend cette expression on la dérive et on l'annule on obtient l'équation non linéaire suivante:

$$A_3 T^6 - A_2 T^4 - A_1 T^2 - A_0 = 0$$

Avec: $A_0 = 5(1 - \alpha)(1 - \beta)W_2$, $A_1 = 3(1 - \alpha)(\beta E_2 + (1 - \beta)W_1)$

$A_2 = (1 - \alpha)(\beta E_1 + (1 - \beta)W_0)$, $A_3 = \alpha + (1 - \alpha)\beta E_0$

A_0, A_1, A_2 et A_3 Sont des coefficients réels indépendants du temps de transfert T . A_0 et A_3 sont des quantités positives. La solution analytique de cette équation donne trois solutions T_{m1}, T_{m2} et T_{m3} . L'allure générale de la fonction $J_Q(T)$ est représentée par la figure (3.4).

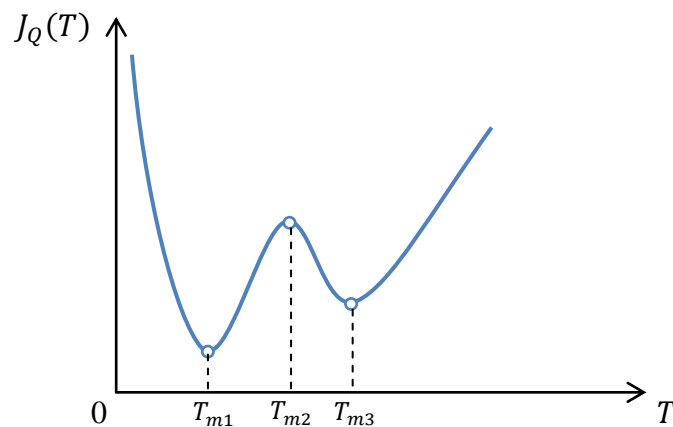


Figure 3.4: Allure générale de la fonction $J_Q(T)$ pour un critère mixte avec puissance quadratique

3.3.1.3 Traitement des critères de performances avec frottements:

3.3.1.3.1 Modèle dynamique avec frottements:

La forme générale du modèle dynamique avec introduction des termes de frottements sec et de frottements visqueux est la suivante:

$$\sum_{j=1}^n M_{ij}(q) \ddot{q}_j + C_i(q, \dot{q}) + G_i(q) + F_i^s \text{sign}(\dot{q}_i) + F_i^v \dot{q}_i = \Gamma_i \quad (3.19)$$

F_i^s et F_i^v sont respectivement les coefficients du frottement sec et visqueux de $i^{\text{ème}}$ articulation.

On divise l'équation(3.19)par τ_i^{max} , on trouve:

$$\bar{\tau}_i(t) = \sum_{j=1}^n \bar{M}_{ij}(q(t)) \ddot{q}_j(t) + \bar{C}_i(q(t), \dot{q}(t)) + \bar{G}_i(q(t)) + \bar{F}_i^s \text{sign}(\dot{q}_i(t)) + \bar{F}_i^v \dot{q}_i(t)$$

Avec : $\bar{\tau}_i(t) = \frac{\tau_i(t)}{\tau_i^{\text{max}}}$, $\bar{F}_i^s = \frac{F_i^s}{\tau_i^{\text{max}}}$ et $\bar{F}_i^v = \frac{F_i^v}{\tau_i^{\text{max}}}$

Introduisons maintenant le changement d'échelle (3.1) et les expressions des (3.4.a), (3.4.b) et (3.4.c). Nous obtenons:

$$\begin{aligned} \bar{\tau}_i(\xi) = & \frac{1}{T^2} \sum_{j=1}^n \bar{M}_{ij}(Q(\xi)) Q_j''(\xi) + \frac{1}{T^2} \bar{C}_i(Q(\xi), Q'(\xi)) + \bar{G}_i(Q(\xi)) + \frac{1}{T} \bar{F}_i^v Q_i'(\xi) \\ & + \bar{F}_i^s \text{sign}(Q_i'(\xi)) \end{aligned} \quad (3.20)$$

Si on pose:

$$\bar{h}_i(\xi) = \sum_{j=1}^n \bar{M}_{ij}(Q(\xi)) Q_j''(\xi) + \bar{C}_i(Q(\xi), Q'(\xi))$$

$$K_i(\xi) = \bar{F}_i^v Q_i'(\xi)$$

$$\bar{G}_i(\xi) = \bar{G}_i(Q(\xi)) + \bar{F}_i^s \text{sign}(Q_i'(\xi))$$

L'équation (3.20) devient:

$$\bar{\tau}_i(\xi) = \frac{1}{T^2} \bar{h}_i(\xi) + \frac{1}{T} K_i(\xi) + \bar{G}_i(\xi) \quad (3.21)$$

3.3.1.3.2 Contrainte avec frottements:

La contrainte couple en présence du frottement donne toujours une borne inférieure et une borne supérieure sur les valeurs admissibles de la solution. Cette contrainte se traite de la manière suivante:

$$\xi \in [0, 1] : -1 \leq \frac{1}{T^2} \bar{h}_i(\xi) + \frac{1}{T} K_i(\xi) + \bar{G}_i(\xi) \leq 1 \quad : \quad T \in [T_{L_i}, T_{R_i}]$$

3.3.1.3.3 Critère mixte temps-efforts quadratiques avec frottements:

Rapportons (3.21) dans l'expression du critère mixte avec effort quadratiques (3.16). Nous aurons:

$$J_Q(T) = T \left((1 - \alpha) + \alpha \left(E_0 + \frac{E_1}{T} + \frac{E_2}{T^2} + \frac{E_3}{T^3} + \frac{E_4}{T^4} \right) \right) \quad (3.22)$$

Avec:

$$\begin{aligned} E_0 &= \int_0^1 \sum_{i=1}^n \bar{G}_i^2 d\xi \\ E_1 &= 2 \int_0^1 \sum_{i=1}^n \bar{G}_i K_i d\xi \\ E_2 &= \int_0^1 \sum_{i=1}^n (K_i^2 + 2H_i \bar{G}_i) d\xi \\ E_3 &= 2 \int_0^1 \sum_{i=1}^n H_i K_i d\xi \\ E_4 &= \int_0^1 \sum_{i=1}^n H_i^2 d\xi \end{aligned} \quad (3.23)$$

Pour un profil particulier $Q(\xi)$, le critère J_Q prend en fonction du temps de transfert T l'allure suivante:

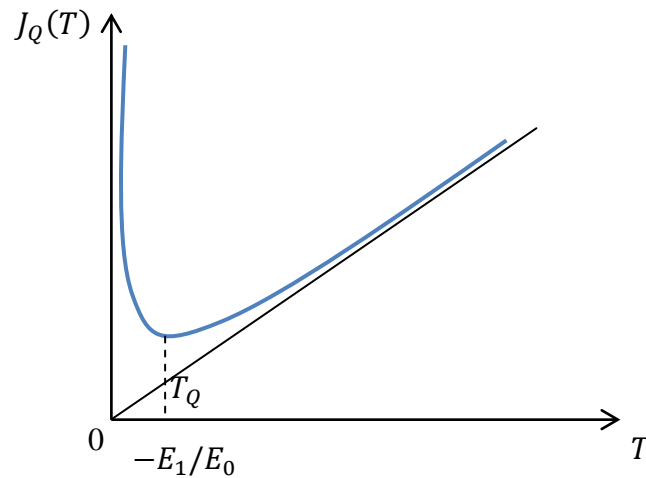


Figure 3.5: Allure de la fonction $J_Q(T)$ pour un profil $Q(\xi)$ donné avec introduction du frottement

Les critères de performance même avec frottement, sont traités avec cette technique comme des fonctions à variable unique T . la prise en compte des frottements ne changera pas le principe de base de la méthode, pour optimiser ces critères discontinus la stratégie de recherche reste valable, il nous faut seulement évaluer la fonction cout pour chaque forme des positions articulaires que le processus d'optimisation tire aléatoirement, ceci est faisable tant qu'on connait les valeurs numériques des coefficients du frottement F^v et F^s .

3.3.2 Modélisation du chemin et du mouvement:

Jusqu'à présent nous avons présenté comment trouver la fonction objectif $J_Q(T)$, en tenant compte les différentes contraintes pour un profil de trajectoire quelconque $Q(\xi)$ composé de deux fonction paramétrique du chemin $P(\lambda)$ et l'autre du mouvement $\lambda(\xi)$. Il nous faut maintenant modéliser ces profils via des modèles d'interpolation appropriés qui doivent répondre à des exigences liées à la qualité désirée des résultats ainsi qu'à l'efficacité de la technique en temps de calcul.

Afin d'éviter les modèles qui induisent des oscillations exagérées et en même temps de permettre la détermination explicite des dérivées, nous avons choisi une modélisation par des fonctions splines cubiques dites "Natural" pour la fonction chemin, ces fonctions sont composées par des morceaux de polynômes de degré trois, raccordés en garantissant la continuité des pentes et des courbures aux nœuds, elles permettent d'assurer la continuité de la trajectoire en position en vitesse et en accélération et par conséquent la continuité des efforts actionneurs, ce profil du chemin est constitué par N_c+2 points de contrôle placés dans le plan (q_i, λ) de tels sorte que les points fixes assurent les conditions aux limites en position (Figure 3.4). Et pour modéliser le profil du mouvement nous avons opté pour une modélisation par des fonctions splines cubiques dites "clamped" ce modèle est bien adapté pour prendre en charge les conditions aux limites en vitesses (3.3d), le profil de mouvement est composé de N_m+2 points de contrôle placés dans un plan normalisé de façon que le premier et le dernier points sont fixes selon (3.3b), tandis que les N_m points intérieurs sont placés librement en respectant les conditions (3.3c) et (3.3d); les profils de chemin et de mouvement seront par la suite générés en interpolant pour chaque profil une courbe lisse sur leurs points de contrôles.

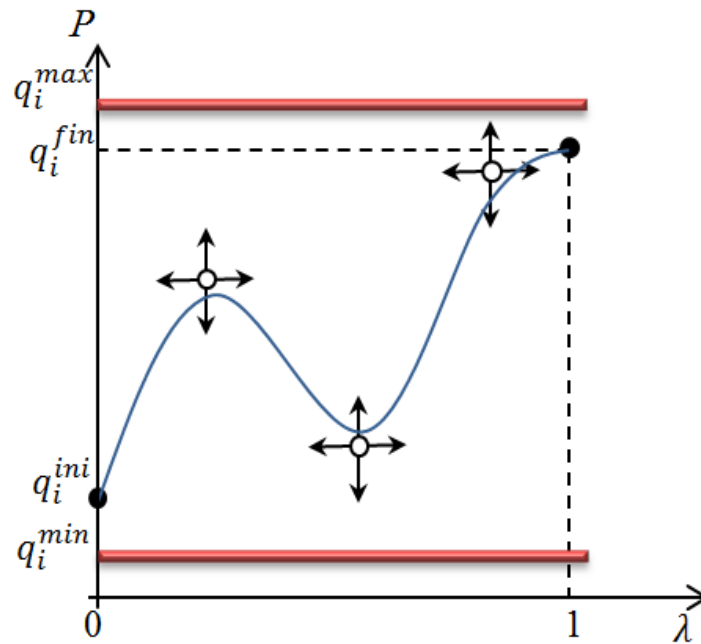


Figure 3.6: Représentation du profil du chemin avec N_c+2 points de contrôle
Les points libres (fixes) sont représentés par des cercles blancs (noirs)

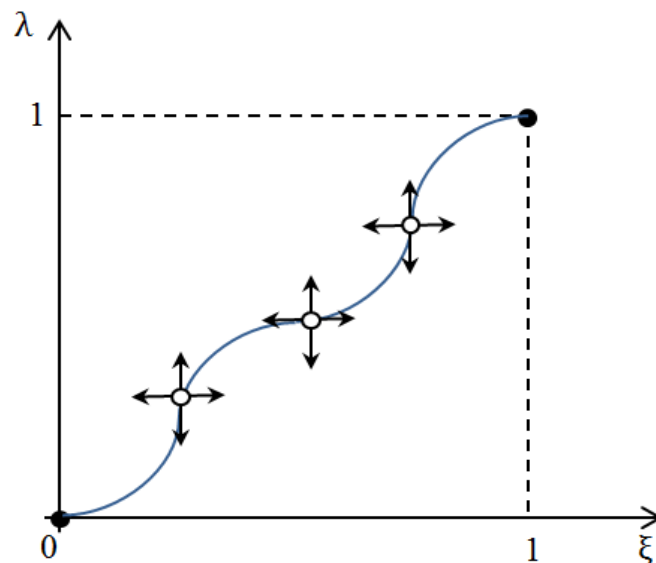


Figure 3.7: Représentation du profil de mouvement avec N_m+2 points de contrôle
Les points libres (fixes) sont représentés par des cercles blancs (noirs)

3.3.3 Technique d'optimisation:

Afin que cette approche soit complète et utilisable, il est nécessaire d'avoir un moyen efficace pour la recherche du meilleur profil possible parmi l'ensemble de tous les profils possibles accessibles, ceci revient à résoudre un problème

d'optimisation pour trouver une fonction objectif qui fournira une valeur optimale à un critère recherché.

3.3.3.1 Algorithmes génétiques:

Pour résoudre ce problème, nous avons opté pour une technique d'optimisation stochastique "**algorithmes génétiques**", nous allons citer à la fin de ce chapitre les raisons pour lesquelles nous avons choisi d'utiliser cette technique.

En effet, l'utilisation d'un algorithme génétique commence par la création d'une population initiale ou chromosome en génétique, ce chromosome est composé des gènes ou leur nombre est défini selon le nombre des points des contrôles utilisés pour générer la fonction de chemin et en même temps la fonction de mouvement, il est à noter que cette étape nécessite un codage de l'ensemble de ces gènes, pour cela nous avons choisi un codage réel des gènes qui correspondent aux points de control intermédiaires, la figure (3.6) montre l'organigramme utilisé dans le prochain chapitre pour optimiser les fonctions décrites antérieurement.

Par la suite cette population initiale va être générée aléatoirement et on obtient pour chaque chromosome une solution qui correspondant à sa fonction de cout.

Pour créer la prochaine génération, trois opérateurs génétiques sont appliqués:
La reproduction: habituellement la stratégie générale de reproduction est que les chromosomes (parents) avec meilleurs fonction de cout ont la possibilité plus de reproduire.

Le croisement: c'est l'opérateur qui va permettre le brassage des caractères génétiques de la population, cet opérateur va créer deux enfants en effectuant un mélange des chromosomes de deux parents.

La mutation: elle consiste à altérer le codage d'un chromosome. Son rôle est de faire émerger de nouvelles gènes en explorant des zones de l'espace de recherche qui pourraient ne pas être visités par simple application de l'opérateur de croisement; en pratique il existe de nombreuses manières de muter un chromosome soit par la modification d'un ou plusieurs gène(s), ou par changement de position d'un gène, ou bien la suppression en ajoutant un gène .
[31] [32] [33] [34] [35]

Dans le prochain chapitre, nous allons fixer le taux ou la probabilité de croisement égale à 65% et la probabilité de mutation égale 4%.

3.3.3.2 Algorithme utilisé des algorithmes génétiques:

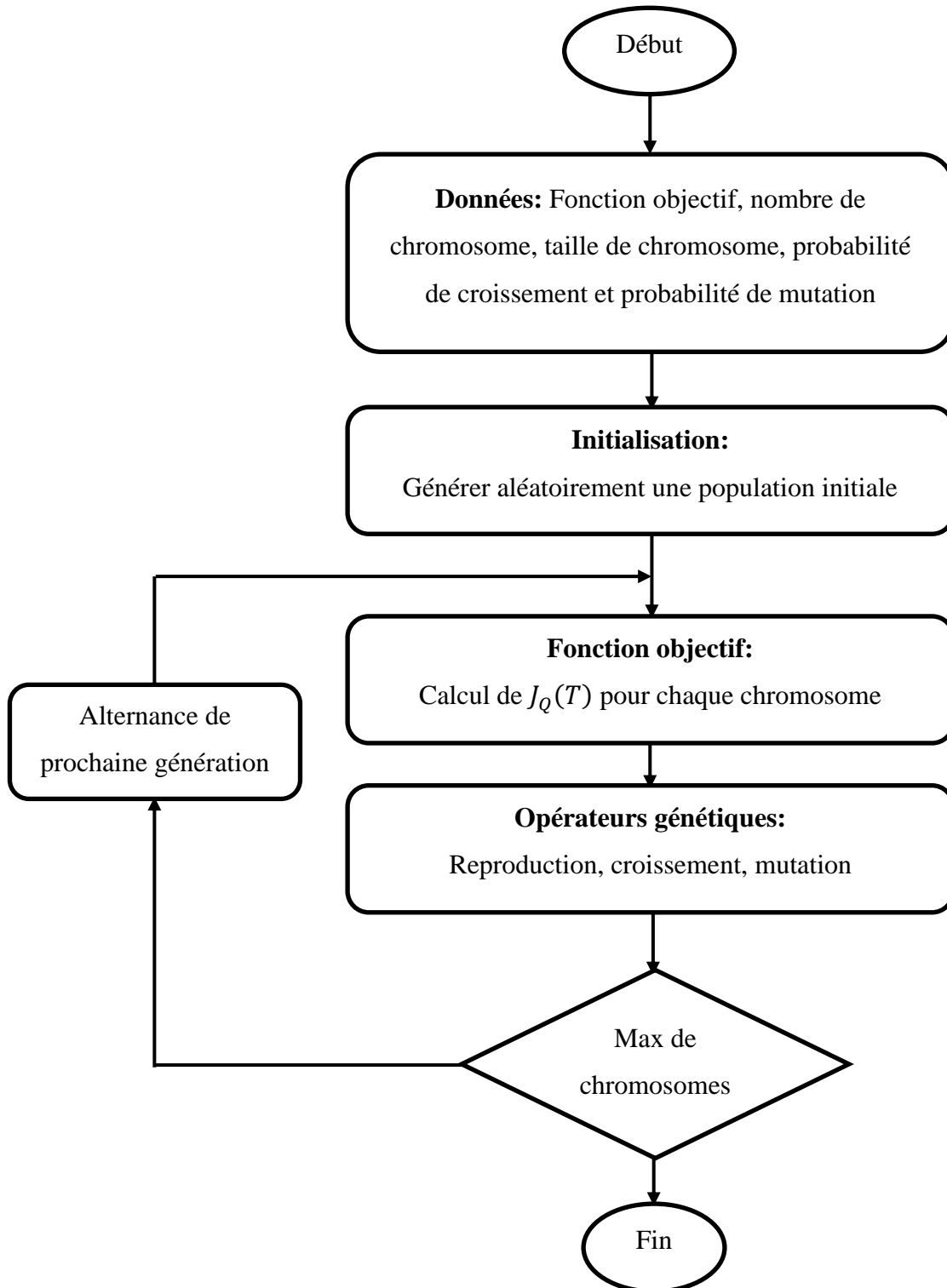


Figure 3.8: Organigramme des Algorithmes Génétiques utilisé

3.3.3.3 Avantages des algorithmes génétiques:

Les avantages de cette technique par rapport aux autres techniques stochastiques ainsi que les techniques déterministes sont:

- La mise en œuvre est plus simple que celle des techniques déterministes et ne nécessite pas une réécriture des codes de calcul lorsqu'on change les particularités du problème.
- Elle n'a besoin de connaître que les valeurs numériques de la fonction objectif et ne font pas appel comme c'est le cas avec les techniques déterministes, aux diverses dérivées de cette fonction
- Elle reste applicable quand le nombre de degré de liberté et le nombre des nœuds de raccordement soient important et même si le modèle de comportement du robot est discontinu (prise en compte des frottements); ce qui n'est pas le cas pour les technique déterministes dont la mise en œuvre devient une tâche épuisante.
- L'apparition des architectures massivement parallèles (les opérateurs génétique) ouvre de nouveaux horizons pour ces algorithmes, cette qualité est une des grandes différences qui sépare les algorithmes génétiques de leurs concurrents les plus proches, les algorithmes de type "Recuit simulé".
- Le codage des paramètres est un avantage car il nous permet de réduire l'espace de perturbation des nœuds intermédiaires, et par conséquent diminuer le nombre des profils de chemin rejetés.
- Enfin, et surtout, les algorithmes génétiques ne sont pas du tout adaptés pour trouver une solution unique optimale, ils sont par contre très performants pour découvrir rapidement une solution acceptable (sachant qu'il y en a beaucoup).

3.4 Récapitulation de la méthode de résolution:

Pour chercher la trajectoire optimale, on doit générer aléatoirement selon la technique d'optimisation des algorithmes génétique une forme de chemin et une forme de mouvement qui vont nous donner par la suite une forme de profil candidat cette dernière sera par la suite évaluée et comparée avec d'autre, cette opération est répétée pour tous les chromosomes introduits, est le meilleurs résultat c'est celui qui satisfait le critère de convergence donné. Il est à noter que tout profil de chemin qui violerait l'une des contraintes géométriques, ainsi que tout

profil de trajectoire qui violerait l'une des contraintes cinématique ou dynamique seront automatiquement rejetés.

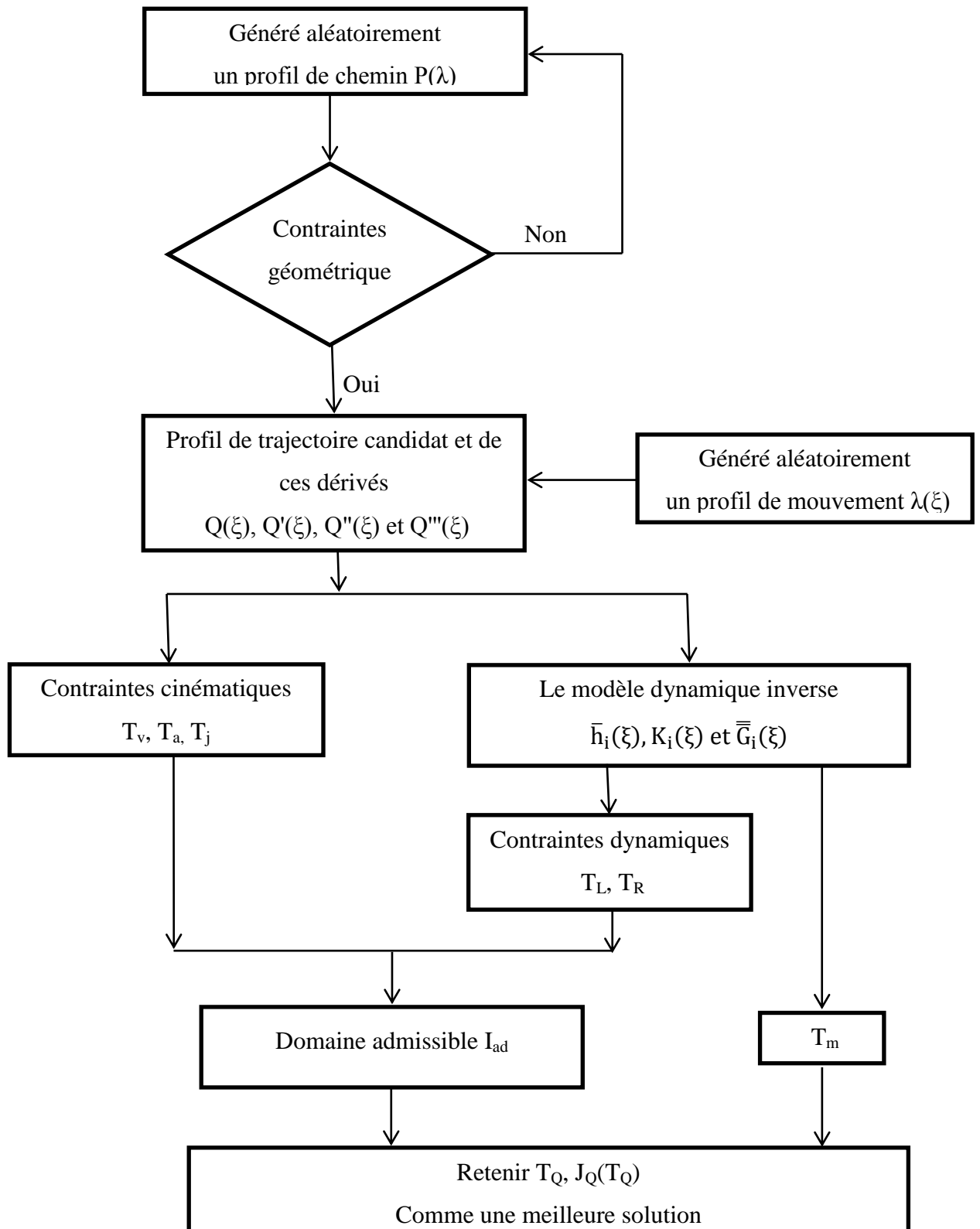


Figure 3.9 : Organigramme de résolution

3.5 Conclusion:

Dans ce chapitre nous avons montré l'idée de base de l'approche qu'on a adoptée pour résoudre le problème de planification des trajectoires des robots manipulateurs. Cette idée consiste à dissocier le temps de transfert de la forme de la trajectoire en utilisant une normalisation de l'échelle de temps et par la suite la trajectoire (normalisée) sera décomposée en une fonction de chemins qui permet de prendre en compte les contraintes géométriques et une fonction de mouvement qui permet de prendre en charge les conditions aux limites. Nous avons vu par la suite une reformulation de différentes contraintes et les critères de performance de problème sans frottements ainsi que dans les cas où les frottements sont inclus dans le modèle dynamique de robot, et pour modéliser les deux fonctions de chemin et de mouvement nous avons choisi les fonction spline cubique de type "natural" et "clamped" ce choix a été effectué selon les propriétés de chaque fonction, la dernière partie de ce chapitre a été consacré au technique d'optimisation "les algorithmes génétiques" qui consiste à perturber les nœuds intermédiaires des fonctions chemin et mouvement pour générer des profils susceptibles de réduire la valeur d'une fonction objectif définie par un critère de performance donné, nous avons vu que cette technique offre la possibilité de réduire l'espace de recherche (en exploitant la propriété de codage des chromosomes) et par conséquent réduire le nombre de profils de chemins rejetés qui va influencer sur la rapidité de trouver des profils de trajectoires candidats et d'accélérer la convergence du processus numérique vers la solution optimale. Et à la fin de ce chapitre nous avons présenté l'organigramme qu'on a utilisé pour traiter ce problème.

Dans le prochain chapitre, nous allons essayer d'appliquer les étapes décrites précédemment pour trouver les trajectoires optimales pour des robots manipulateurs qui possèdent de différents degrés de liberté et soumis aux contraintes composées.

CHAPITRE 4

MISE EN ŒUVRE ET DISCUSSION DES RESULTATS

4.1 Introduction:

L'objectif visé dans ce chapitre est de juger l'efficacité de l'approche qu'on a adoptée pour résoudre les problèmes liés à la planification des trajectoires des robots manipulateurs. Pour ce faire, il est bien évident que nous devons exploiter non seulement les avantages de cette approche mais aussi celles de la modélisation par les fonctions splines cubiques et l'optimisation par les algorithmes génétiques. A cet effet, l'élaboration nécessite de faire des tests comparatifs avec d'autres méthodes qu'ont été présentés des solutions optimales pour valider nos résultats. Nous allons traiter dans nos exemples différents robots manipulateurs qui présentent des variations dans leurs dynamiques soit par le nombre de degré de liberté, les contraintes imposées et le critère à minimiser, les détails concernant chaque robots vont être présenter dans l'exemple qui correspond.

4.2 Premier exemple: Robot (IBM 7535 B04)

Le robot considéré dans cet exemple est planaire à deux degrés de liberté, il représente une simplification du robot industriel IBM 7535 B04 type SCARA (figure 4.1), on lui demande d'effectuer un déplacement d'une masse $m_3 = 6 \text{ Kg}$ et d'inertie $I_{z3} = 0.01 \text{ Kg.m}^2$, d'une configuration initiale $q^{init} = [0, 0.689]^T$ vers une configuration finale $q^{fin} = [0.433, -0.689]^T$.

Les paramètres de DHM et inertiels de ce robot manipulateur sont regroupés dans le tableau 4.1.

Tableau 4.1 : Paramètres de robot manipulateur IBM 7535

Segment <i>i</i>	σ_i	L_i	α_i	d_i	θ_i	$m_i[\text{Kg}]$	$x_{gi}[\text{m}]$	$I_{zi}[\text{Kg.m}^2]$	$\tau_i^{max}[\text{N.m}]$
1	0	0.4	0	0	q_1	/	/	1.6	25
2	0	0.25	0	0	q_2	15	0.125	0.43	9

Le critère à minimiser est de type mixte, durée de transfert et moyenne quadratique des couples moteurs ou le coefficient de pondération α prend des valeurs différentes. Dans un premier temps pour le critère de minimisation du temps de parcours nous allons voir l'influence du nombre de nœuds (intervalles) sur la qualité des résultats obtenus, et dans un second temps c'est l'influence du coefficient de pondération qui est étudiée.

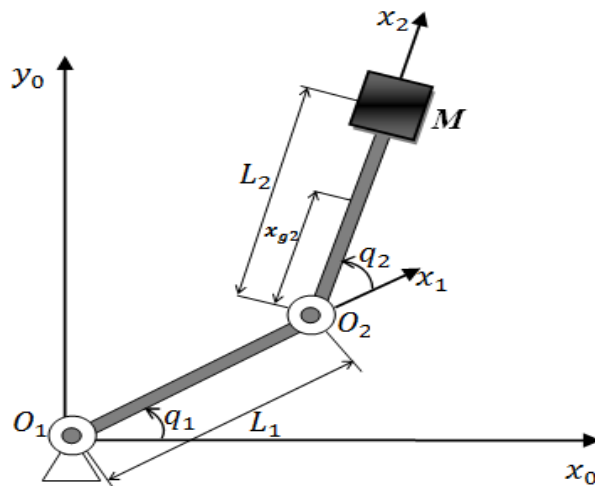
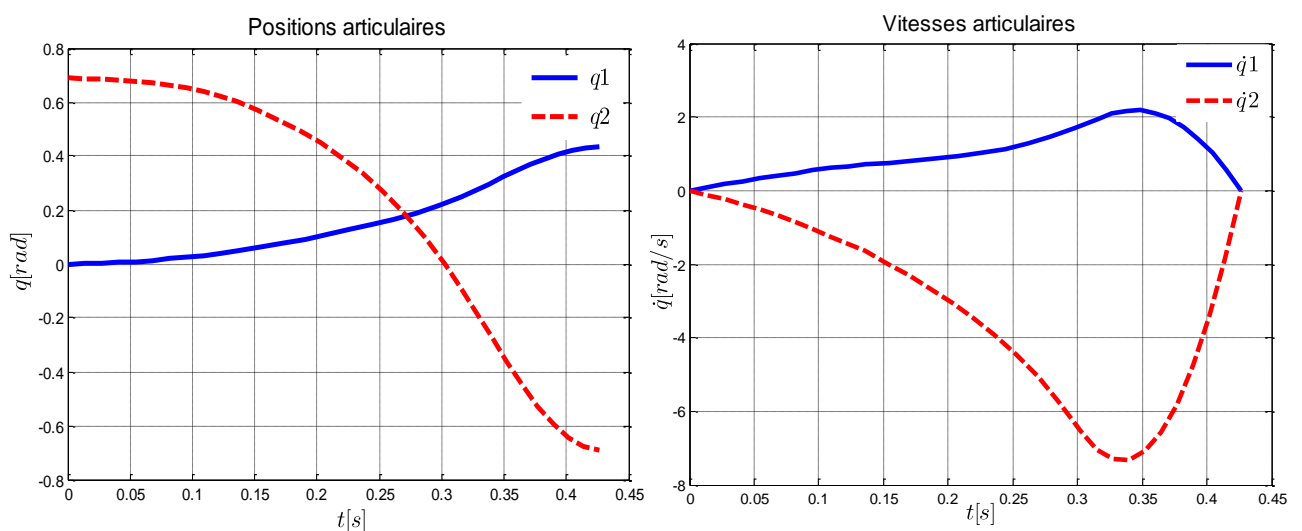


Figure 4.1: Robot industriel IBM 7535 B04 type SCARA simplifié

1^{ère} cas: $\alpha = 1$ (minimisation du temps de transfert)

Nombre de nœuds	4
Nombre de chromosomes	100
Temps d'exécution de la tâche $T_Q(s) = J_{obj}$	0.431



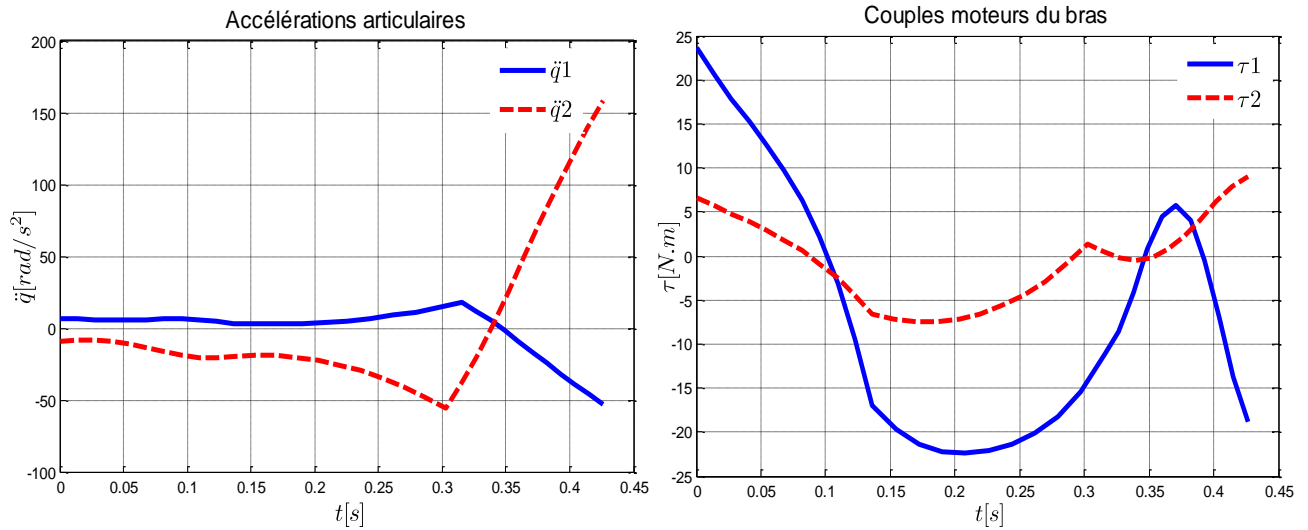


Figure 4.2: Résultats de l'exemple 1 correspondants aux $\alpha = 1$, et $N_p = 4$

Nombre de nœuds	5
Nombre de chromosomes	100
Temps d'exécution de la tâche $T_Q(s) = J_{obj}$	0.428

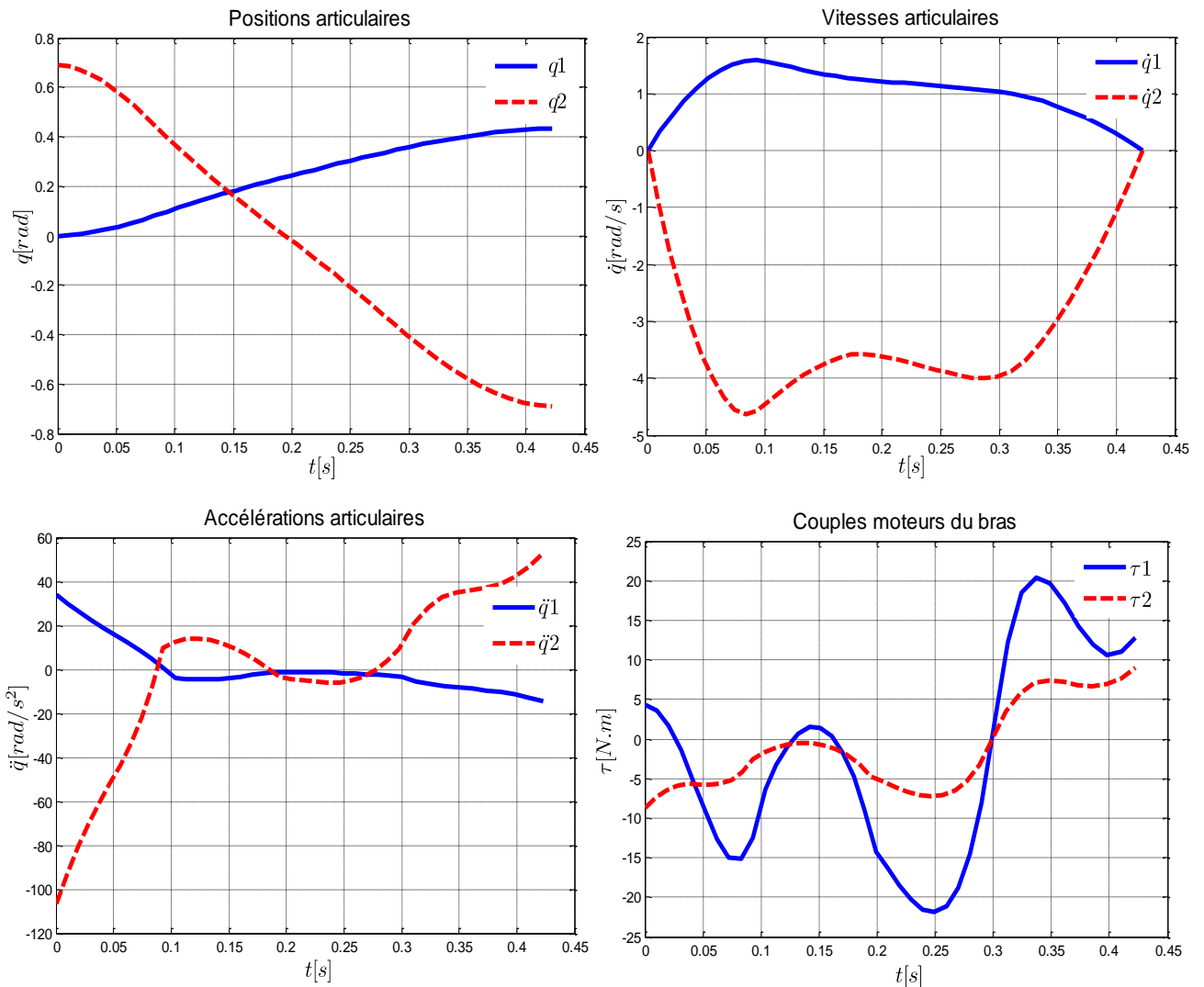


Figure 4.3: Résultats de l'exemple 1 correspondants aux $\alpha = 1$, et $N_p = 5$

Nombre de nœuds	6
Nombre de chromosomes	100
Temps d'exécution de la tâche $T_Q(s)=J_{obj}$	0.426

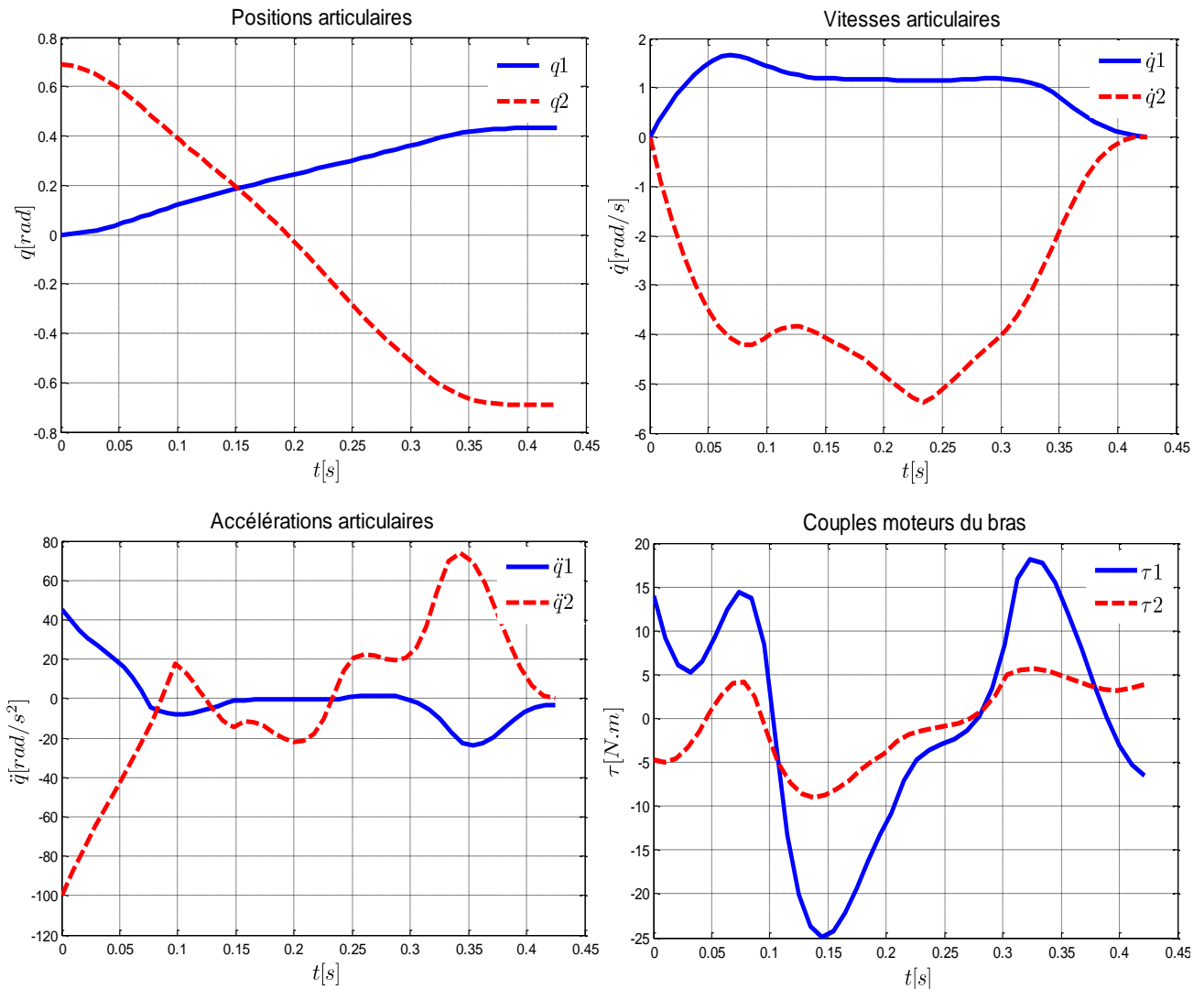


Figure 4.4: Résultats de l'exemple 1 correspondants aux $\alpha = 1$, et $N_p = 6$

Les résultats obtenus montrent qu'au fur et à mesure qu'on augmente le nombre de nœuds des fonctions splines cubiques la qualité des résultats est meilleure, ceci traduit qu'on se rapproche de plus en plus à des fonctions lisses et en même temps continu. Cependant l'utilisation d'un nombre important de nœuds va pénaliser l'effort de calcul (codage des chromosomes, le nombre d'itérations) ainsi que le temps d'exécution du code de calcul ceci peut être très remarquable lorsque le robot étudié possède un nombre de degré de liberté élevé. Sur la base de ces résultats (amélioration de l'ordre de 20 ms par nœuds) et afin d'éviter ces

difficultés nous avons retenu le nombre de nœud égal à 05 (03 intermédiaires) pour les prochains exemples.

2^{ème} cas: $\alpha = 0.75$

Nombre de chromosomes	100
Temps d'exécution de la tâche T_Q (s)	0.452
Valeur de la fonction objectif $J_{obj}(s)$	0.435

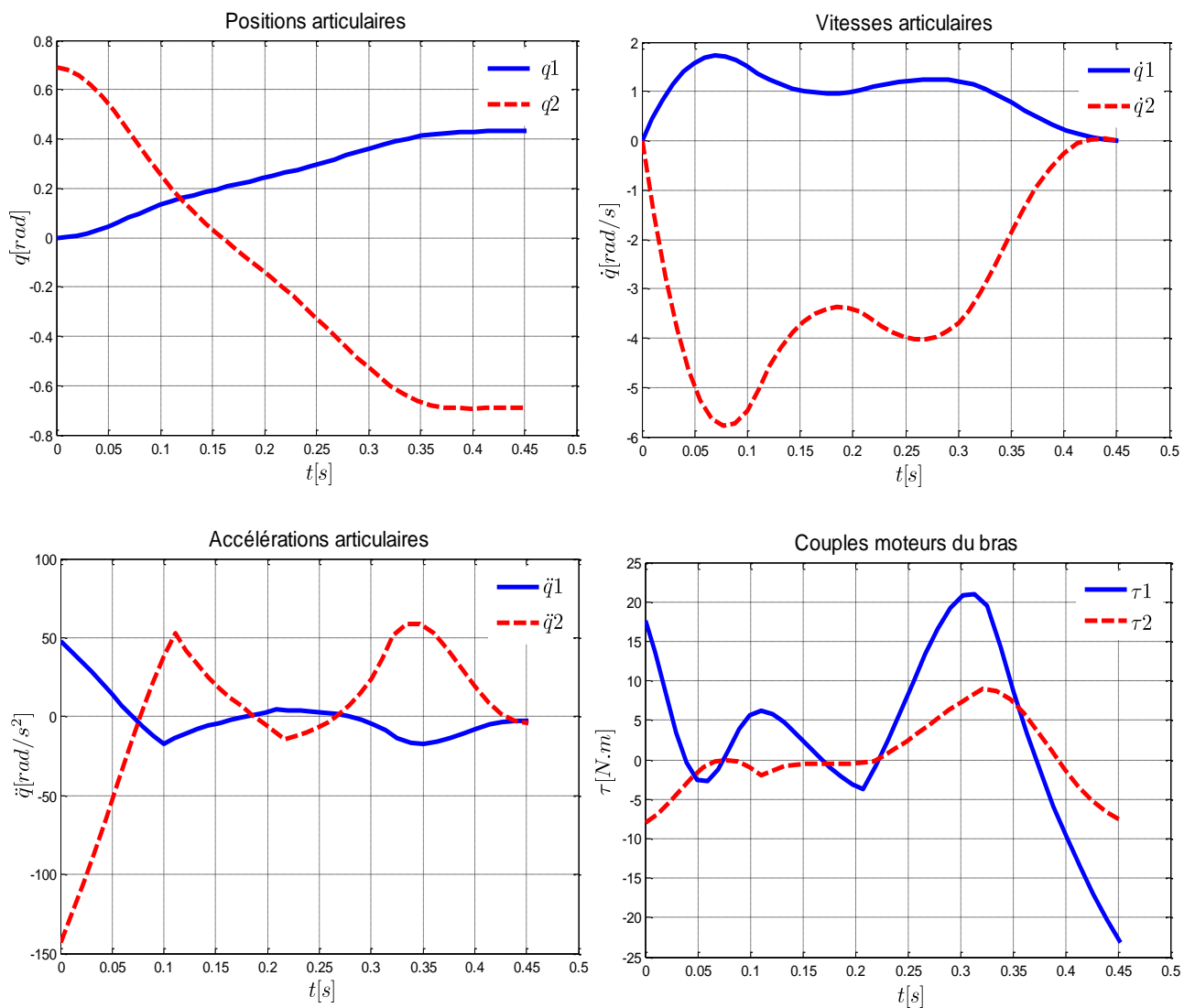


Figure4.5: Résultats de l'exemple 1 correspondants aux $\alpha = 0.75$

3^{ème} cas: $\alpha = 0.5$

Nombre de chromosomes	100
Temps d'exécution de la tâche T_Q (s)	0.550
Valeur de la fonction objectif J_{obj} (s)	0.366

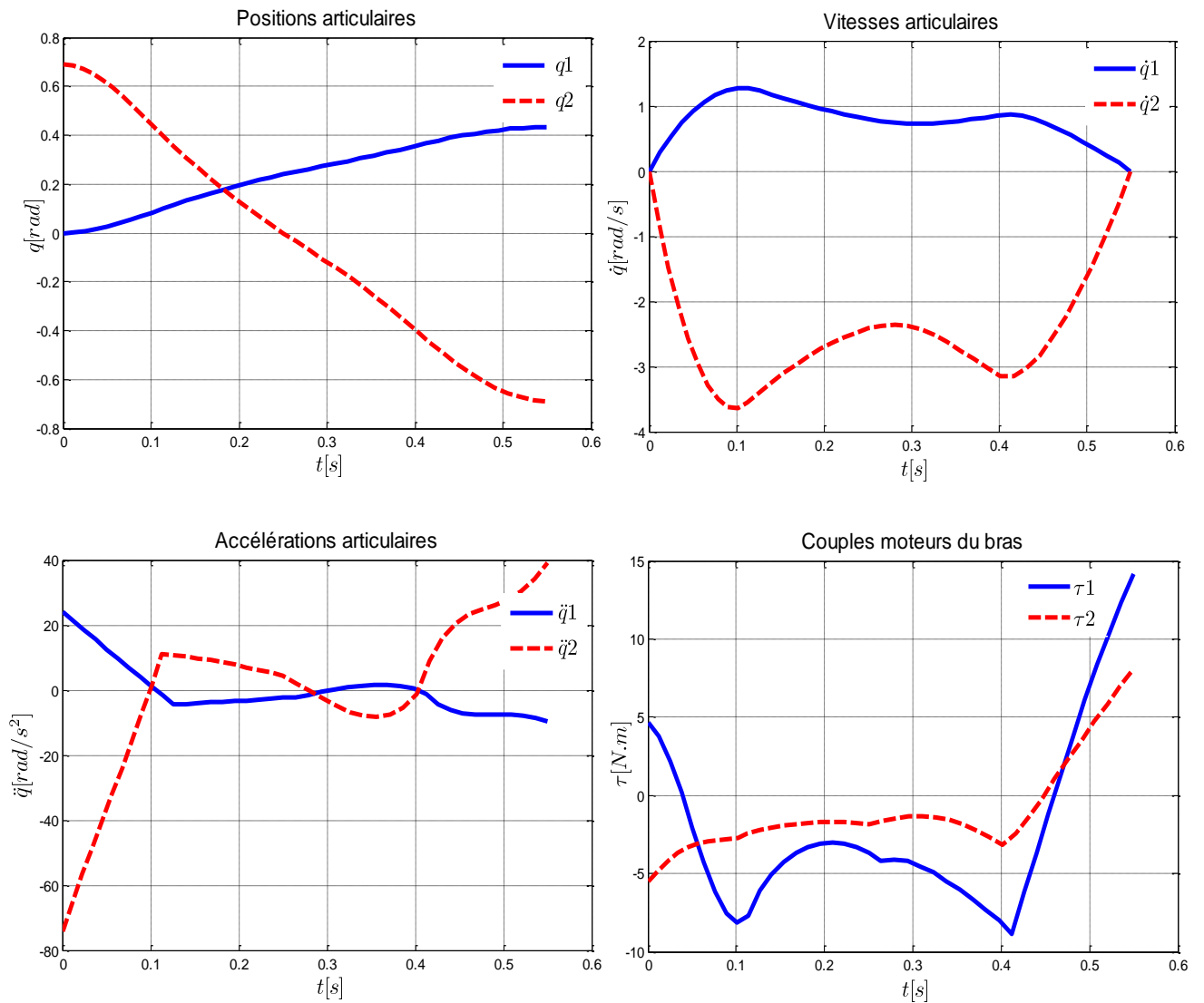


Figure 4.6: Résultats de l'exemple 1 correspondants aux $\alpha = 0.5$

4^{ème} cas: $\alpha = 0.25$

Nombre de chromosomes	100
Temps d'exécution de la tâche T_Q (s)	0.815
Valeur de la fonction objectif J_{obj} (s)	0.271

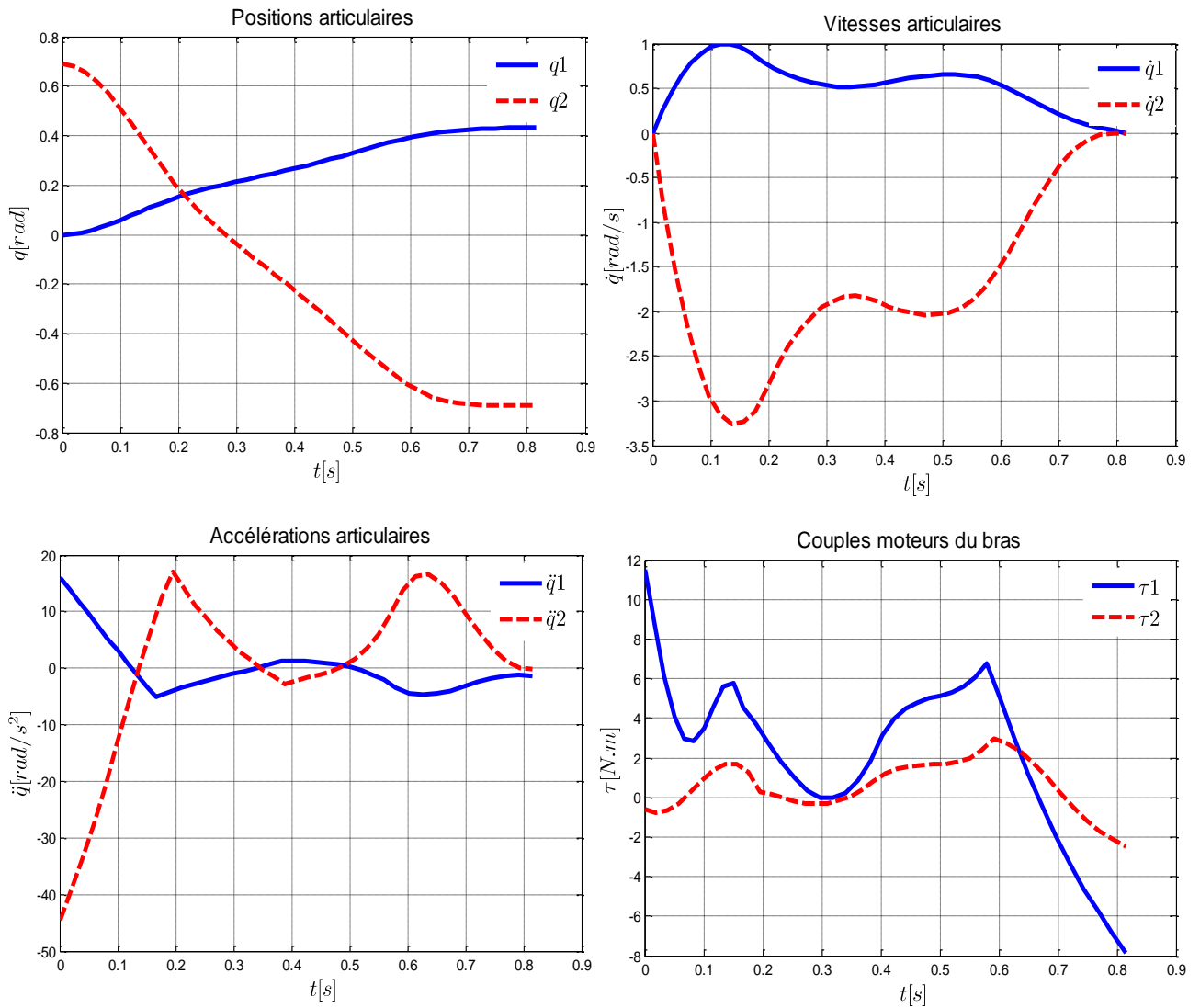


Figure 4.7: Résultats de l'exemple 1 correspondants aux $\alpha = 0.25$

Les figures précédentes montrent que l'introduction de l'effet de la moyenne quadratique des efforts actionneurs dans la fonction objectif par le biais (introduction) du facteur de pondération α augmente le temps de transfert T_Q au fur et au mesure avec la minimisation du α , cela se traduit qu'on donne moins d'importance à la minimisation du temps de transfert de la trajectoire et par conséquent on évite le stade de la saturation des efforts actionneurs (profil bang-bang) ainsi que les amplitudes maximales des vitesses articulaires devient minimales et les mouvements obtenus sont doux et plus facile à implanter sur le système.

5.3 Deuxième exemple: Robot planaire (2R)

Le robot considéré ici est celui de l'exemple précédent IBM 7535, on lui demande d'effectuer une autre tâche définie par des valeurs des variables articulaires de la configuration initiale $q^{init} = [0, 0.78]^T$ vers une configuration finale $q^{fin} = [0, 0]^T$, les contraintes imposées sur ce robot manipulateur sont: $|\tau_1| \leq 25$ et $|\tau_2| \leq 9$.

Les mouvements seront planifiés pour une valeur fixe du coefficient de pondération $\alpha = 1$, nos résultats seront par la suite comparés avec celui présentes dans la référence [82].

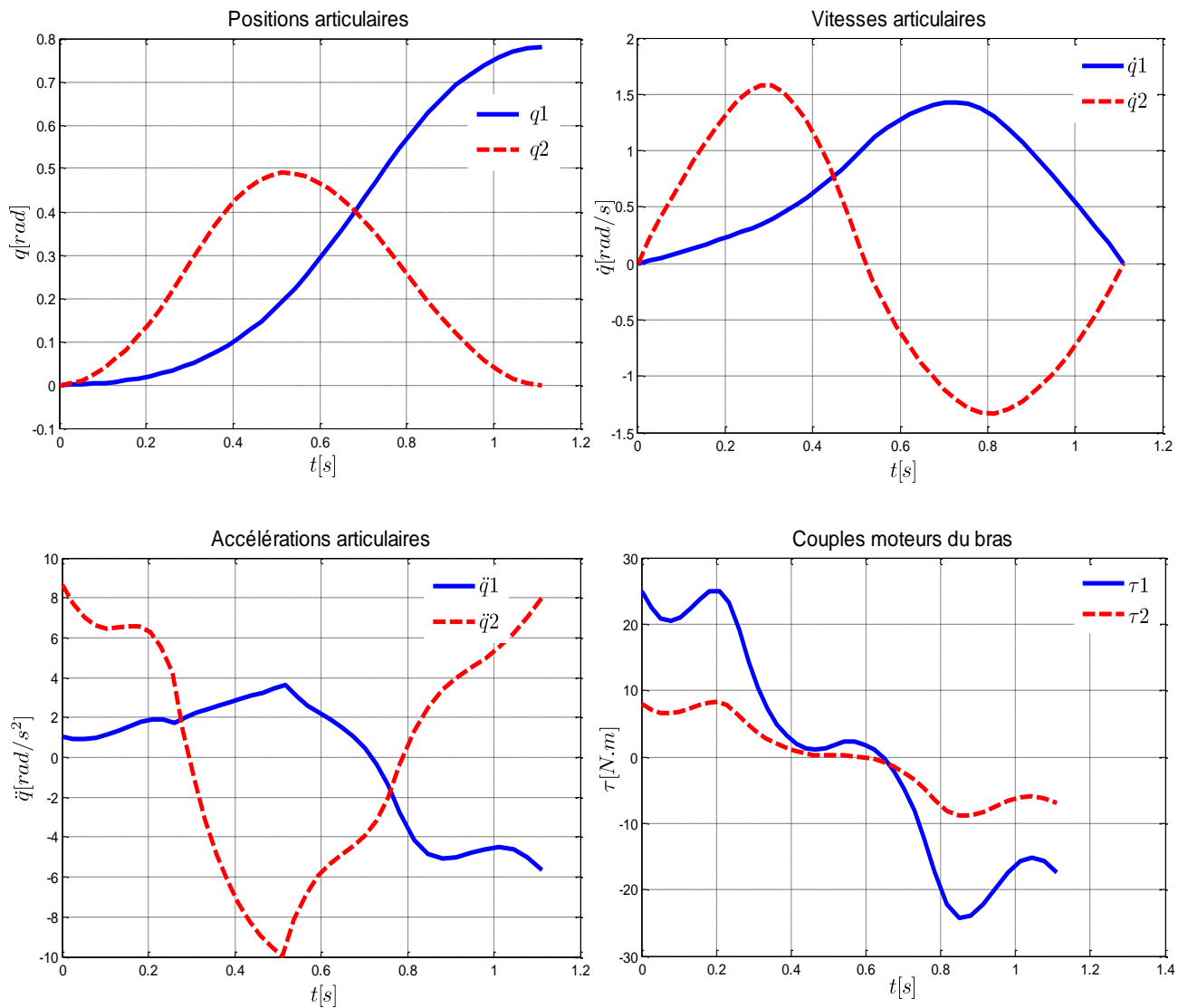


Figure 4.8: Résultats de l'exemple 2 correspondant aux $\alpha = 1$

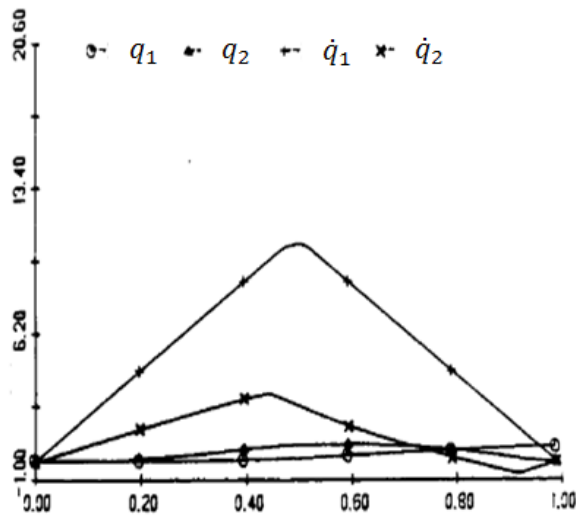


Figure 4.9: Positions et vitesses articulaires
trouvées par [82] avec $\alpha = 1$

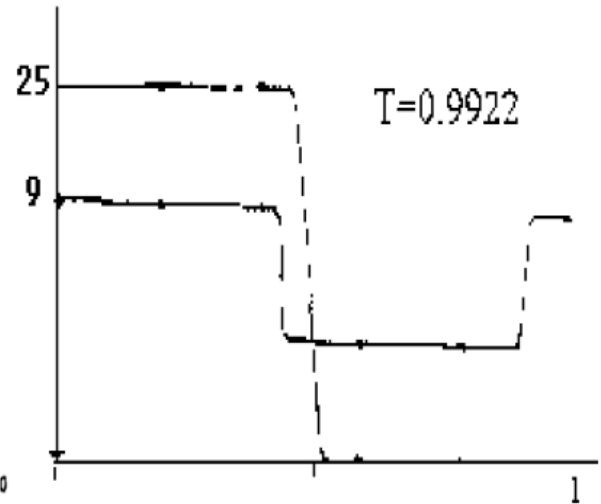


Figure 4.10: Couples trouvés
par [82] avec $\alpha = 1$

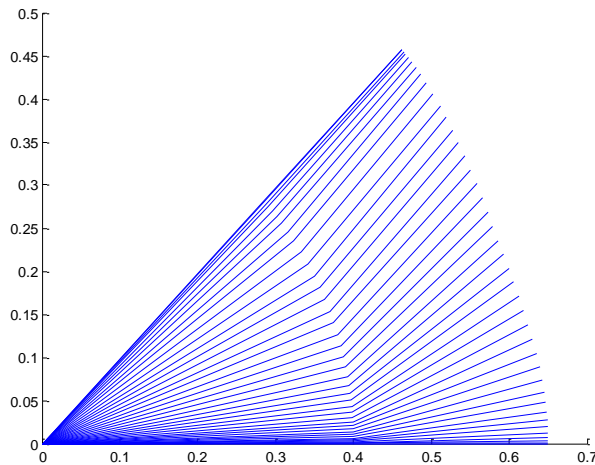


Figure 4.11: L'allure du mouvement
optimisé correspondant à $\alpha=1$

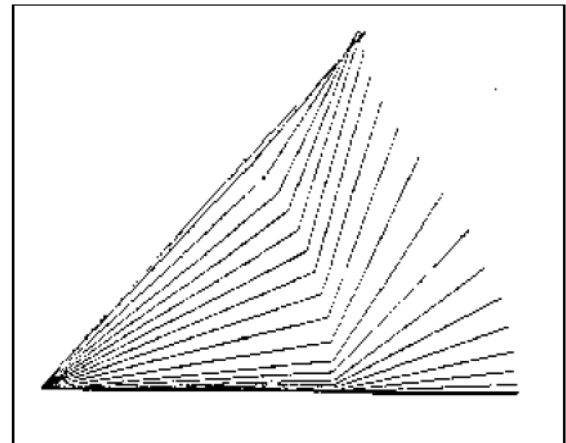


Figure 4.12: L'allure du mouvement
optimisé trouvée par [82] avec $\alpha = 1$

Les résultats présentés (figure 4.8) montrent que le temps optimal pour exécuter la tâche est de 1.11s ce qui reste très raisonnable en comparant le temps trouvé par [82] qui a appliqué le Principe de Maximum de Pontriaguine PMP.

Au niveau des commandes on remarque qu'elles sont continuées avec des basculements entre les limites des couples moteurs, et l'apparition des courtes zones de saturation sans qu'il y ait des violations des contraintes imposées sur les couples moteurs du robot ce qui reflète sur le temps optimal de transfert, toute fois l'auteur [82] trouve des commandes saturées de type Bang-Bang (figure 4.10).

5.4 Troisième exemple: Robot planaire (3R)

Nous allons considérer un robot planaire à trois articulations rotoïde (3R), ces paramètres D-H et inertiels sont regroupés dans (tableau 4.2), ce robot déplace d'une configuration initiale $q^{init} = [0, -\pi/3, -\pi/10]^T$ vers une configuration finale $q^{fin} = [\pi, 0, 0]^T$ avec des vitesses aux limites nulles.

Tableau 4.2: Paramètres géométriques et inertiels du robot planaire 3R

Segment i	σ_i	L_i	α_i	d_i	θ_i	m_i [Kg]	x_{gi} [m]	I_{zi} [Kg.m ²]
1	0	0.7	0	0	q_1	7	0.35	0.8
2	0	0.5	0	0	q_2	5	0.5	0.5
3	0	0.5	0	0	q_3	5	0.5	0.5

Les capacités technologiques sont regroupées dans (tableau 4.3):

Tableau 4.3: Capacités technologiques du robot planaire 3R

Segment i	q_i^{max} [rad]	\dot{q}_i^{max} [rad/s]	\ddot{q}_i^{max} [rad/s ²]	\ddot{q}_i^{max} [rad/s ³]	τ_i^{max} [N.m]
1	π	3	8	10	30
2	$3\pi/4$	3	8	15	25
3	$3\pi/4$	3	8	20	25

Le critère à minimiser pour cette tâche est un critère mixte avec puissances quadratique, nos résultats seront comparés avec ceux présentes dans la référence [79].

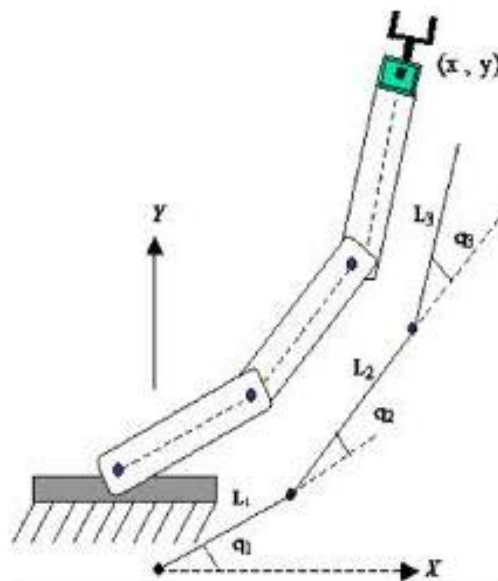


Figure 4.13: Robot manipulateur planaire à trois articulations rotoïde

1^{ère} cas: $\alpha = 1$ (minimisation du temps de transfert)

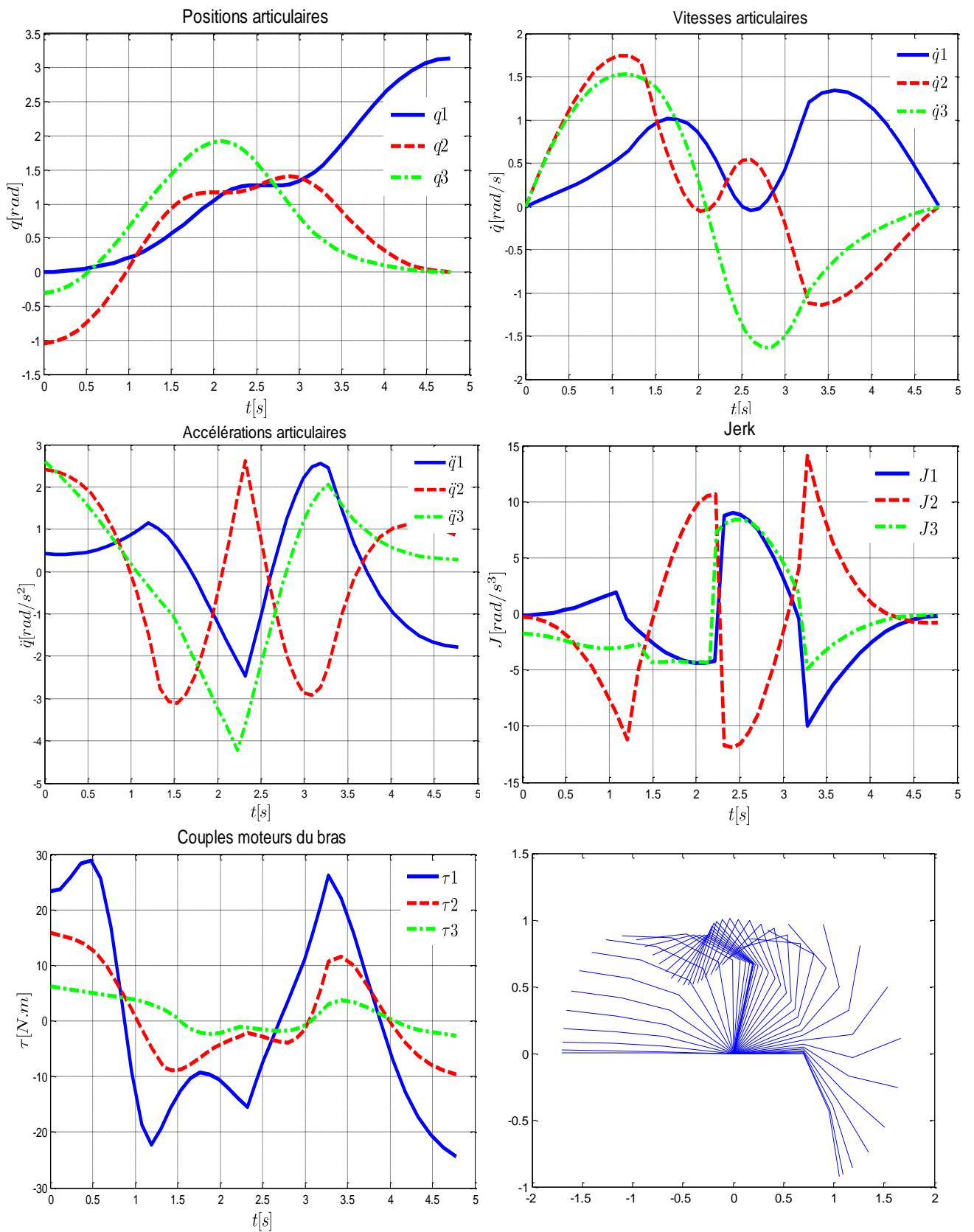


Figure 4.14: Résultats de l'exemple 3 pour $\alpha = 1, \beta = 1$ on trouve

$$T_Q = J_Q = 4.77$$

2^{ème} cas: $\alpha = 0.75$

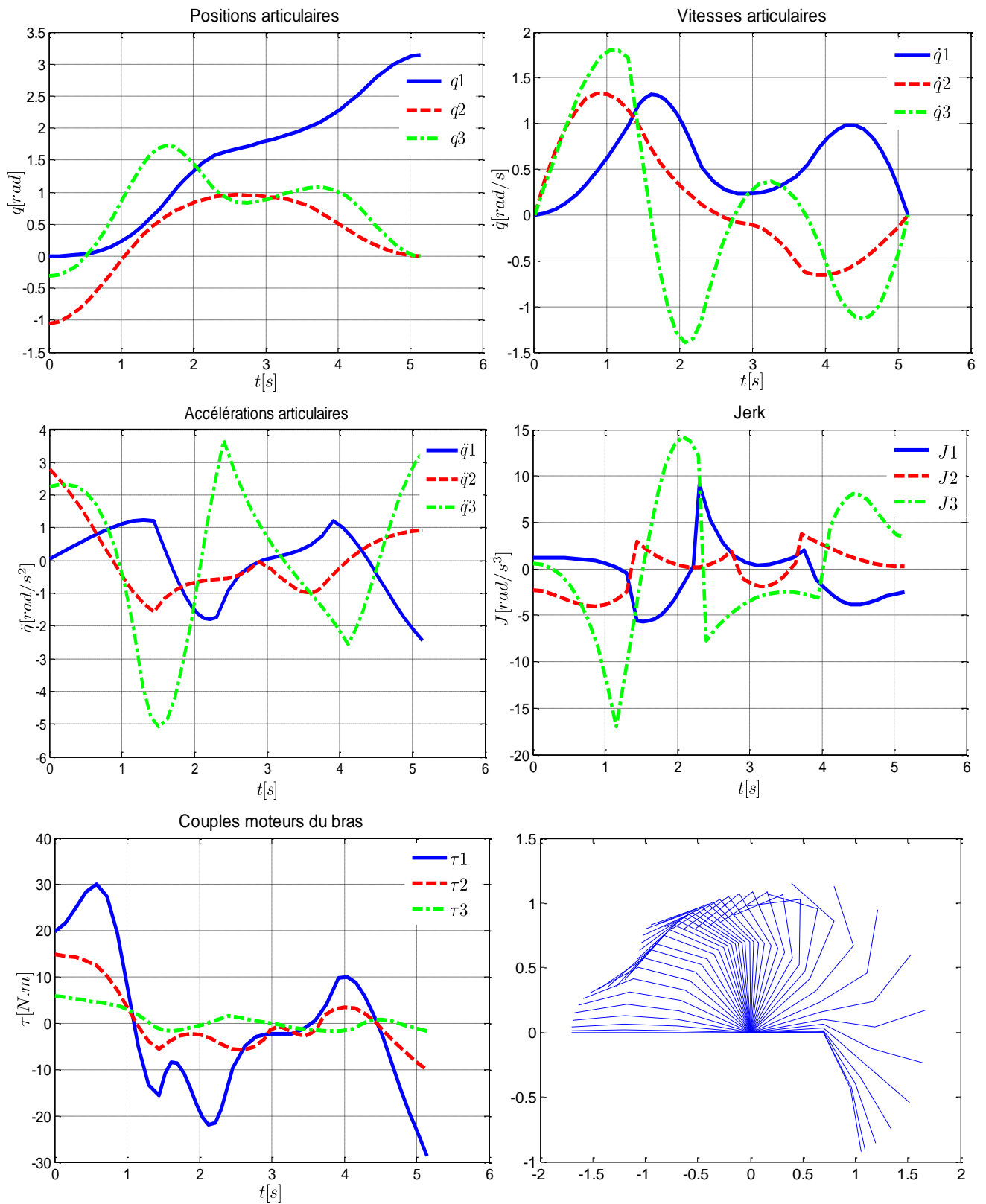
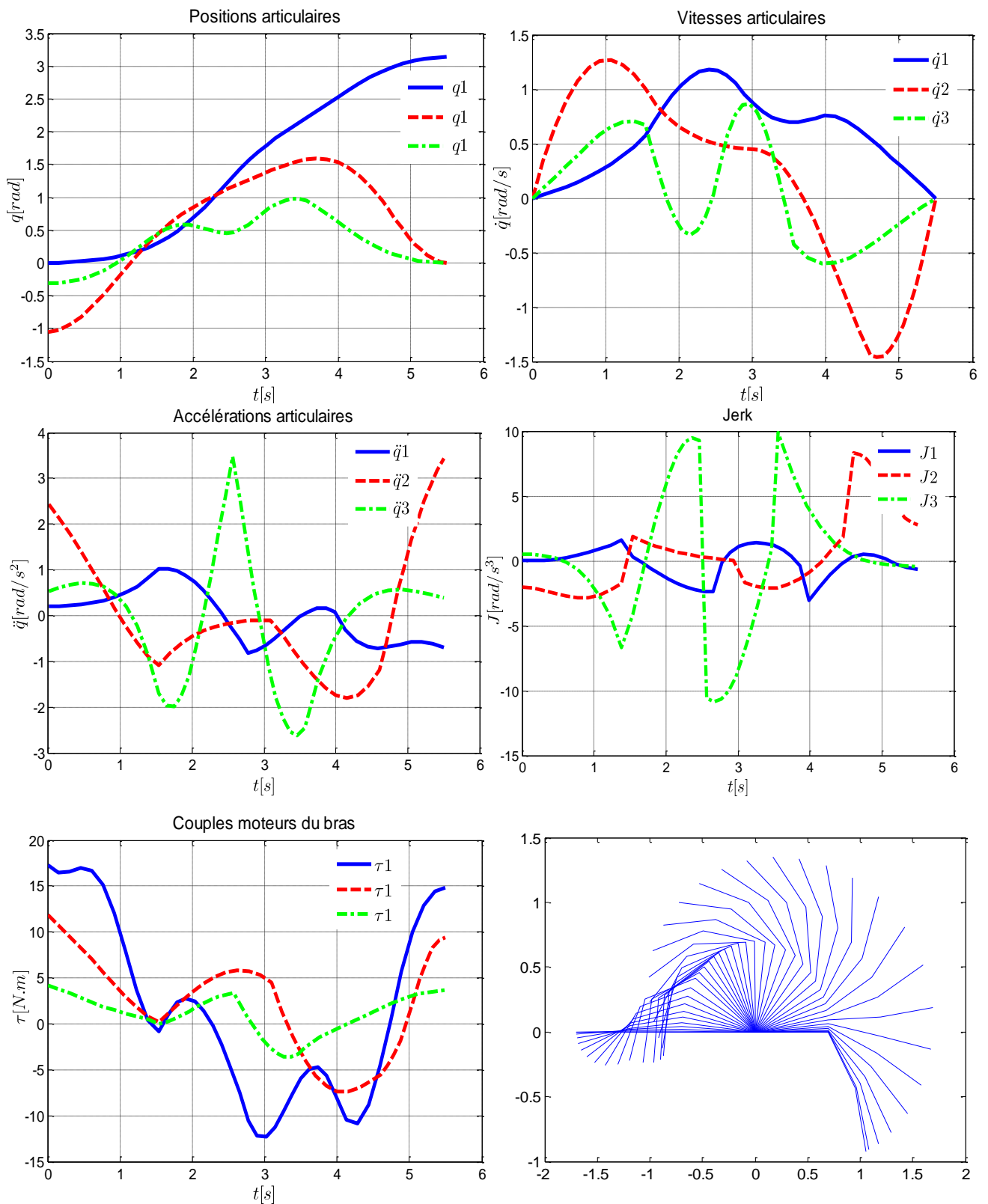
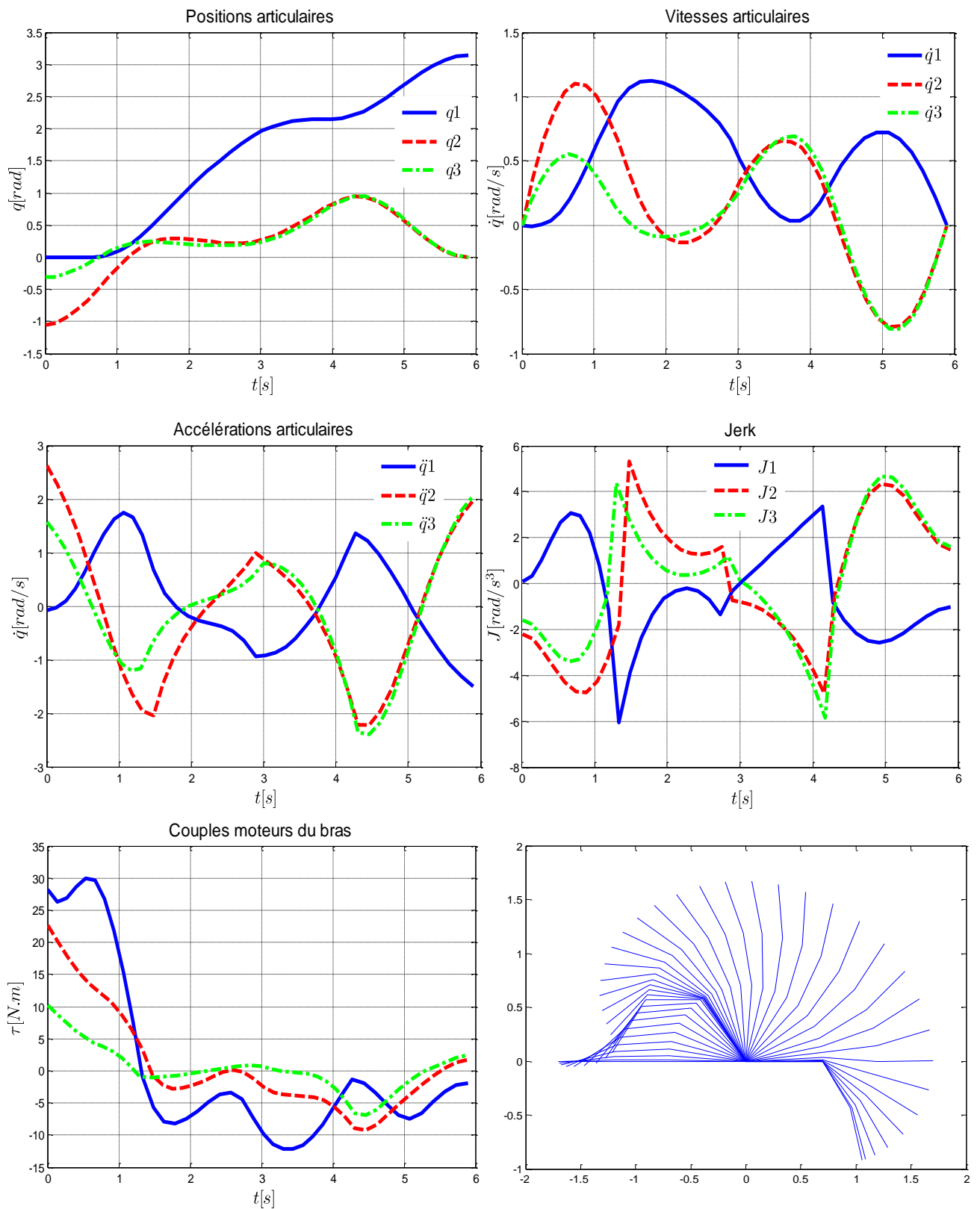


Figure 4.15: Résultats de l'exemple 3 pour $\alpha = 0.75, \beta = 1$ on trouve $T_Q = 5.14$,

$$J_Q = 4.69$$

3^{ème} cas: $\alpha = 0.5$ Figure 4.16: Résultats de l'exemple 3 pour $\alpha = 0.5, \beta = 1$ on trouve $T_Q = 5.49$,

$$J_Q = 3.66$$

4^{ème} cas: $\alpha = 0.3$ Figure 4.17: Résultats de l'exemple 3 pour $\alpha = 0.3, \beta = 0.5$ on trouve $T_Q = 5.89, J_Q = 2.07$

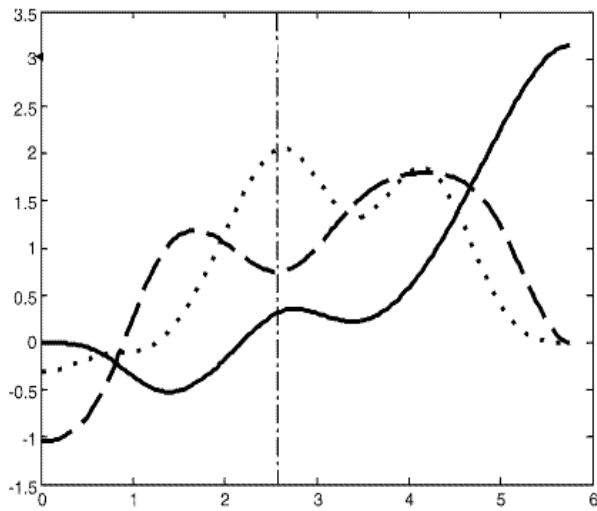


Figure 4.18: Trajectoires optimales
trouvées par [79]

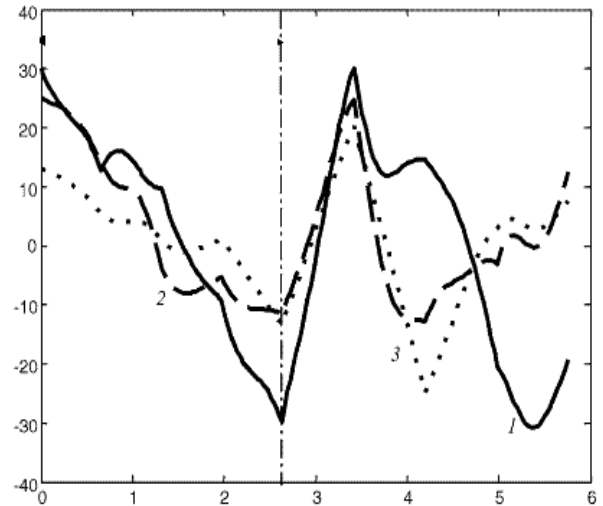


Figure 4.19: Commandes optimales
trouvées par [79]

L'observation des résultats obtenus pour ce robot manipulateur qui a subi des contraintes géométriques, cinématiques et dynamiques montre qu'au fur et à mesure que le coefficient de pondération α augmente le robot prend moins de temps pour exécuter la tâche, ceci signifie qu'on a attribué beaucoup d'importance à l'exploitation des capacités au niveau des actionneurs; on remarque aussi que la première articulation développe des vitesses de déplacement toujours positives tandis que la deuxième et la troisième articulation ont des allures positives et négatives ceci du essentiellement au couplage dynamique entre les corps du robot.

Il est à noter que ce même exemple a été traité par [79] en utilisant la méthode SQP (Sequential Quadratic Programming) dont les ont considéré que la fonction objectif est un critère mixte avec puissance quadratique, pour $\alpha = 0.3, \beta = 0.5$ le temps de transfert trouvé est 5.76 ce qui signifie que nos résultats sont très attendants.

5.5 Quatrième exemple: Robot PUMA 560 (4R)

L'exemple que nous allons traiter dans ce paragraphe est un robot universel à quatre degrés de liberté de type PUMA 560, sa structure contient quatre articulations pivot, on lui demande d'effectuer un déplacement entre les deux configurations initiale et finale définissent dans (tableau 4.4), les paramètres de ce robot manipulateur sont représentés dans (tableau 4.5).

Nous allons utiliser un critère mixte temps-efforts quadratiques avec un coefficient de pondération $\alpha=0.1$ les résultats obtenus sont représentés dans (figure 4.18) et comparés avec ceux présentes dans la référence [83].

Tableau 4.4 : Tâche à réaliser par le robot 4R PUMA 560

Articulation i	1	2	3	4
$q^{init} [rad]$	0	$\pi/18$	$-\pi/4$	$-\pi/6$
$q^{fin} [rad]$	$5\pi/9$	$\pi/18$	$-7\pi/36$	$-\pi/9$

Tableau 4.5 : Paramètres du robot 4R PUMA 560

Articulation i	1	2	3	4
$\alpha_i [rad]$	0	$\pi/2$	0	0
$d_i [m]$	0	0	0.4318	0.4331
$r_i [m]$	0	0	-0.15	0
$a_i [m]$	0	0.068	0.12	0.128
$b_i [m]$	0	-0.006	0	0
$c_i [m]$	0	-0.2275	0.012	0
$M_i [kg]$	0	17.4	5.62	2.34
$I_{xx_i} [kg.m^2]$	0	1.031	0.0143	0.00503
$I_{yy_i} [kg.m^2]$	0	1.505	0.2324	0.00503
$I_{zz_i} [kg.m^2]$	0.35	0.620	0.2514	0.05504
$I_{yz_i} [kg.m^2]$	0	-0.023	0	0
$I_{xz_i} [kg.m^2]$	0	0.269	0.0047	0
$I_{xy_i} [kg.m^2]$	0	0.00709	0	0
$\tau_i^{max} [N.m]$	97.6	186.4	89.4	20.1

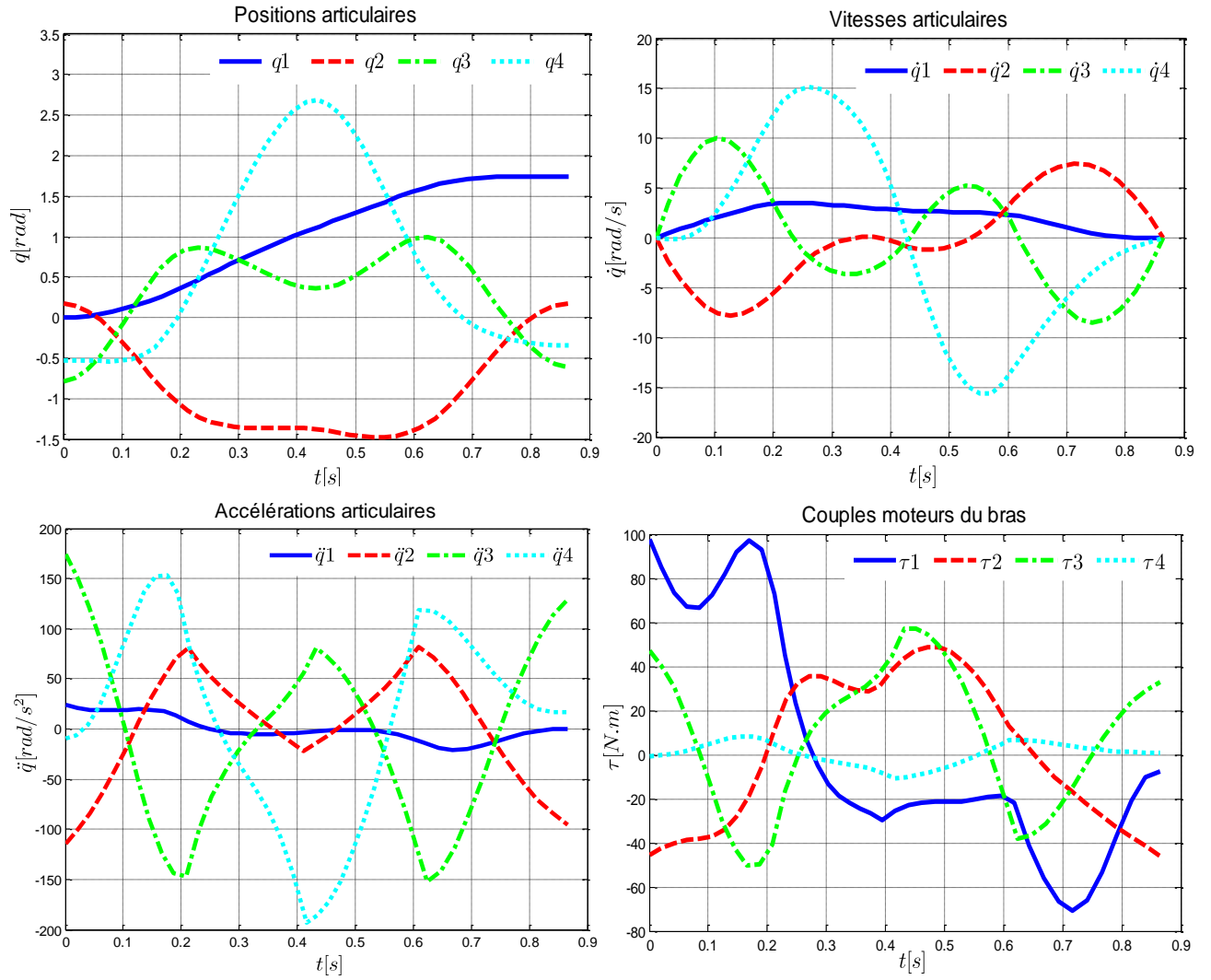


Figure 4.20: Résultats de l'exemple 4 pour $\alpha = 0.1$ on trouve $T_Q = 0.87$ s

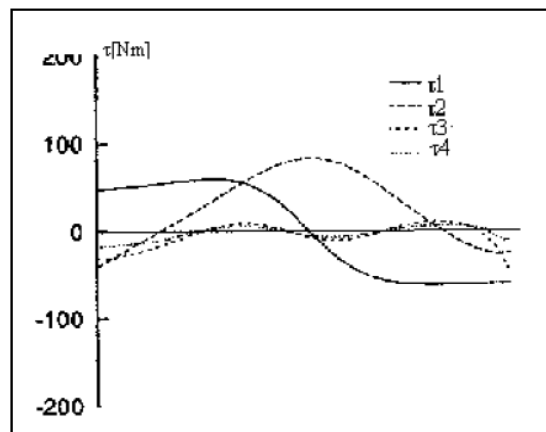


Figure 4.21: Commande optimale trouvée par [83], pour $\alpha = 0.1$, $T_Q = 0.83$ s

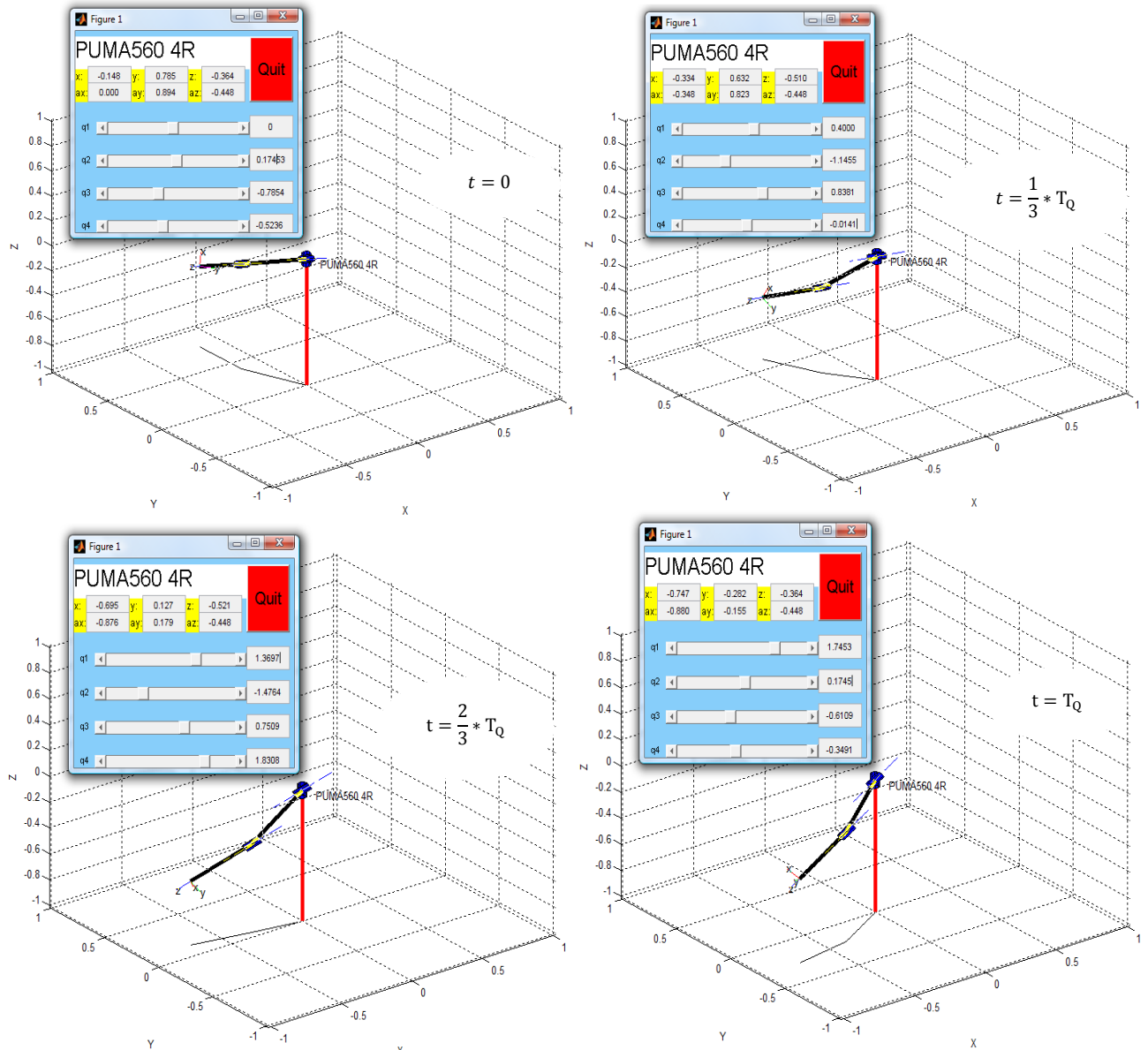


Figure 4.22: Aspect optimal de mouvement de robot PUMA 560 (4R)

Les résultats obtenus pour le robot universel PUMA 560 (4R) montrent que lorsqu'on donne plus d'importance à la minimisation des couples pour un facteur de pondération $\alpha = 0.1$ on trouve un temps optimal de transfert égale à 0.87s, on remarque aussi que les allures des commandes (figure 4.18) sont similaires à celles présentés dans la référence [83] qui a trouvé un temps de transfert égale à 0.83s en utilisant le Principe de Maximum de Pontriaguine PMP, malheureusement que cette référence ne donne pas les allures de déplacements et de vitesses articulaires.

En effet, la simulation de ce robot manipulateur d'une vue 3-D (figure 4.19), montre le mouvement optimal de ce robot qui commence à partir de la configuration initiale et termine à la configuration finale.

5.7 Cinquième exemple: Robot PUMA 560 (6R)

Nous allons traiter dans cette application un robot manipulateur à six degré de liberté de type PUMA560, sa structure contient six articulations pivot (figure 4.23). Ce robot manipulateur est chargé de faire un mouvement entre les deux configurations définies dans (tableau 4.6), le transfert doit être fait sans violation des contraintes géométriques, cinématiques et dynamiques représentées dans (tableau 4.7), les paramètres géométriques et inertiels de ce robot manipulateur sont présentés dans (tableau 4.8).

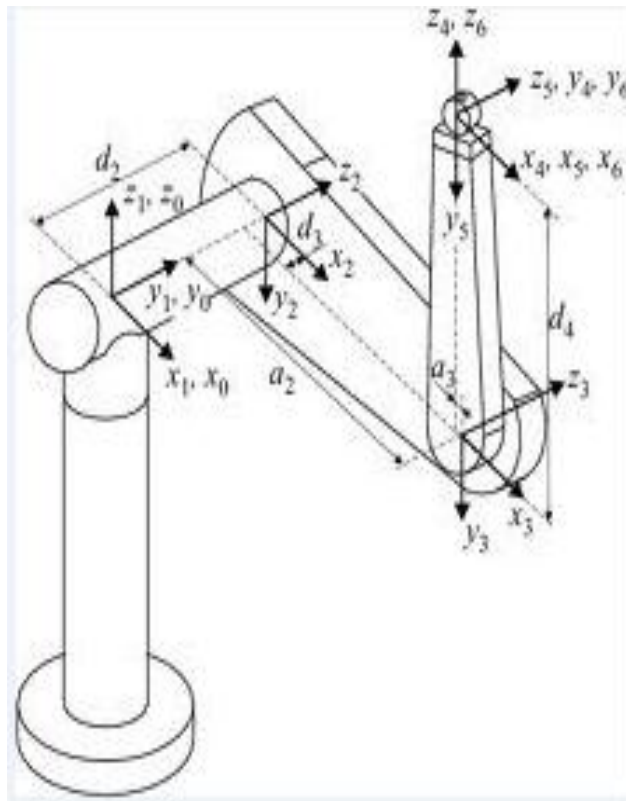


Figure 4.23: Robot manipulateur 6R de type PUMA 560

Tableau 4.6 : Tâche à réaliser par le robot 6R PUMA 560

Articulation i	1	2	3	4	5	6
q_i^{init} [rad]	0	$-\pi/6$	0	$-\pi/3$	0	0
q_i^{fin} [rad]	$2\pi/3$	$\pi/6$	$\pi/4$	$\pi/3$	$-\pi/4$	$\pi/6$

Nous allons considérer dans cette application un critère mixte avec puissances quadratiques définis dans (3.14).

Les contraintes imposées sont :

Tableau 4.7 : Limites technologiques du robot 6R PUMA 560

Articulation i	1	2	3	4	5	6
q_i^{max} [rad]	π	$3\pi/4$	$3\pi/4$	π	$3\pi/4$	$3\pi/4$
\dot{q}_i^{max} [rad/s]	8	10	10	5	5	5
\ddot{q}_i^{max} [rad/s ²]	10	12	12	8	8	8
\dddot{q}_i^{max} [rad/s ³]	30	40	40	20	20	20
τ_i^{max} [N.m]	97.6	186.4	89.4	24.2	20.1	21.3

Tableau 4.8 : Paramètres géométriques et inertiels du robot 6R PUMA 560

Articulation i	1	2	3	4	5	6
α (rad)	$\pi/2$	0	$-\pi/2$	$\pi/2$	$-\pi/2$	0
d_i (m)	0	0.4318	0.0203	0	0	0
r_i (m)	0	0	0.15005	0.4318	0	0
M (kg)	-	17.4	4.8	0.82	0.34	0.09
p_x (m)	0	-0.3638	-0.0203	0	0	0
p_y (m)	0	0.006	-0.0141	0.19	0	0
p_z (m)	0	0.2275	0.07	0	0	0.032
I_{xx} (kg.m ²)	0	0.13	0.066	1.8e-3	0.3e-3	0.15e-3
I_{yy} (kg.m ²)	0.35	0.542	0.086	1.3e-3	0.4e-3	0.15e-3
I_{zz} (kg.m ²)	0	0.539	0.0125	1.8e-3	0.3e-3	0.04e-3
I_{xy} (kg.m ²)	0	0	0	0	0	0
I_{yz} (kg.m ²)	0	0	0	0	0	0
I_{xz} (kg.m ²)	0	0	0	0	0	0

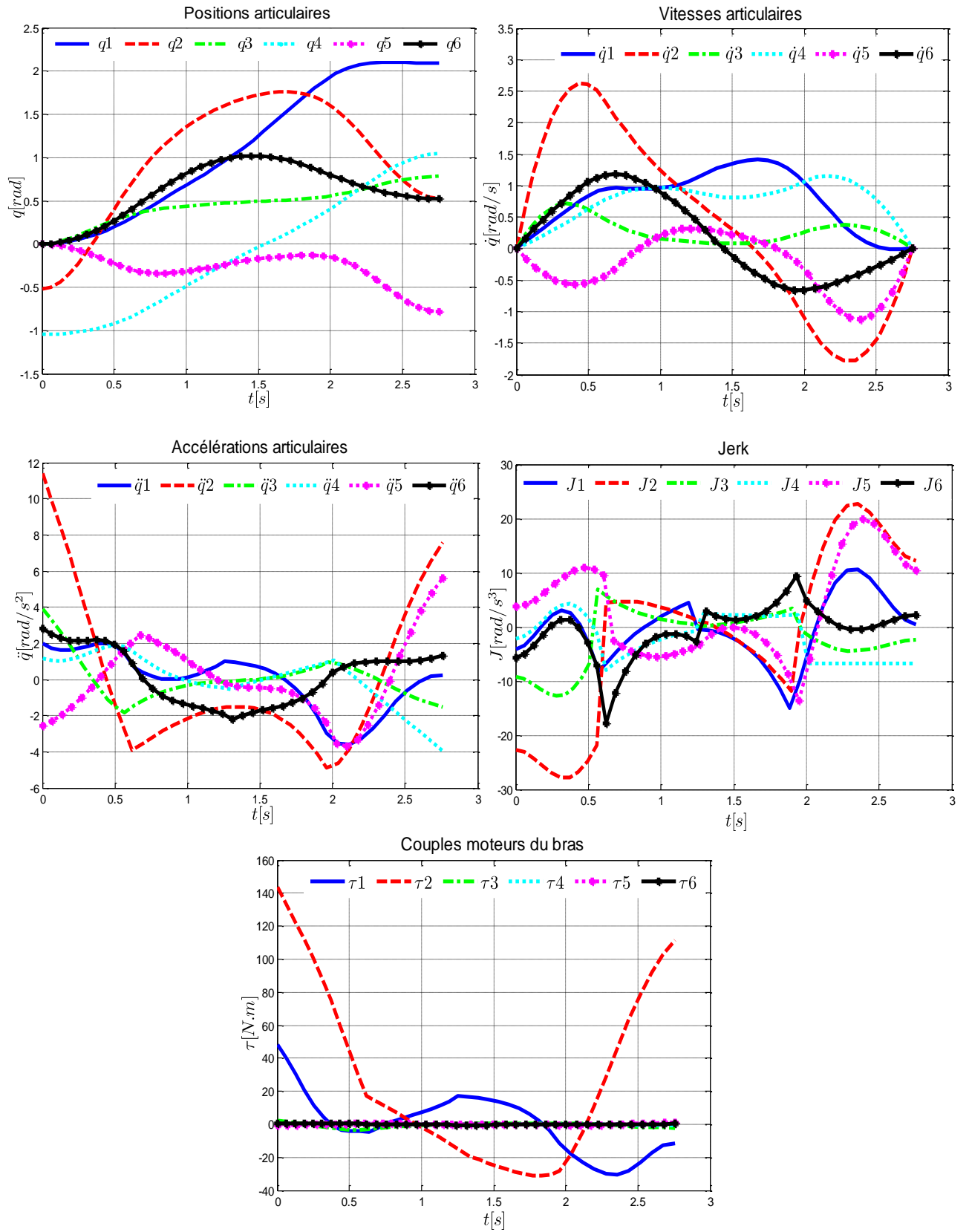


Figure 4.24: Résultats de l'exemple 5 pour $\alpha = 0.5, \beta = 1$ on trouve

$$J_Q = 1.427, T_Q = 2.755$$

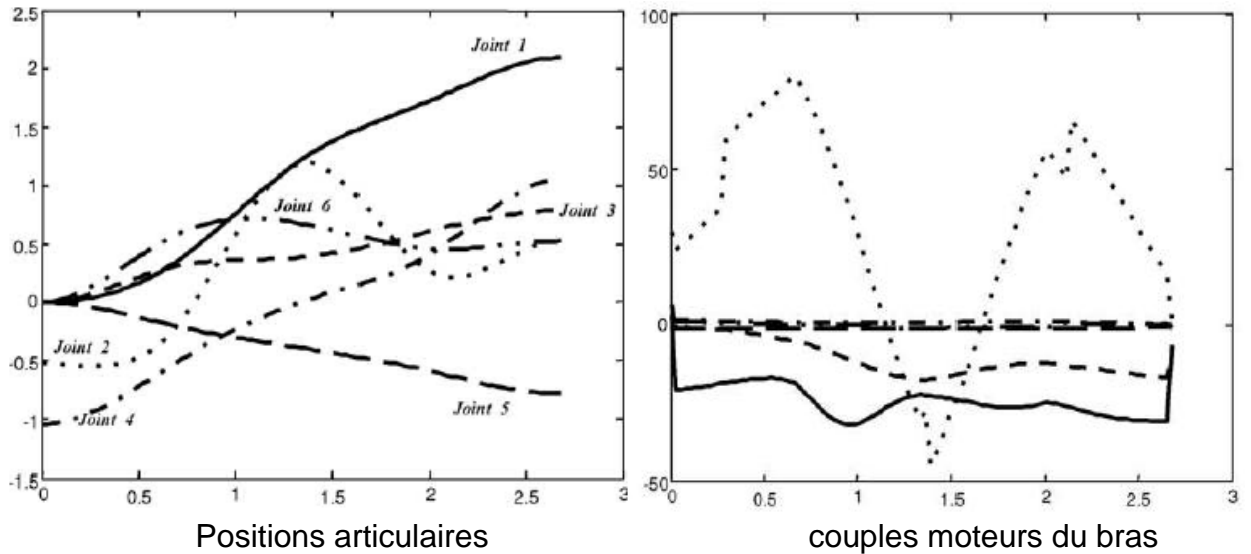


Figure 4.25: Résultats trouvés par [79], pour $\alpha = 0.5$, $\beta = 1$

$$J_Q = 1.370, T_Q = 2.675$$

L'observation des résultats obtenus pour le robot manipulateur PUMA 560 (6R) montre que lorsqu'on a attribué une égalité entre le temps de transfert et la moyenne quadratique des efforts actionneurs soit $\alpha = 0.5, \beta = 1$, on obtient une fonction de coût égale à 1.427s qui correspond à un temps de transfert optimal égale à 2.755s.

Sur le plan allure de trajectoires articulaires nous avons trouvé des allures similaires avec celles trouvées dans la référence [79], la différence réside dans les allures des dérivées des articulations articulaires et par conséquent dans les commandes articulaires car ce sont déterminées à partir de modèle dynamique inverse.

En effet, les résultats que nous avons trouvés sont de même ordre de grandeur que ceux présentés dans la référence [79] soit $J_Q = 1.370, T_Q = 2.675$ en utilisant le Principe de Maximum de Pontriaguine (figure 4.25).

La simulation de robot manipulateur PUMA 560 montre les aspects des mouvements optimaux pour les six articulations rotoïde (les trois premières articulations qui représentent les corps plus les trois dernières articulations qui représentent l'organe terminal)

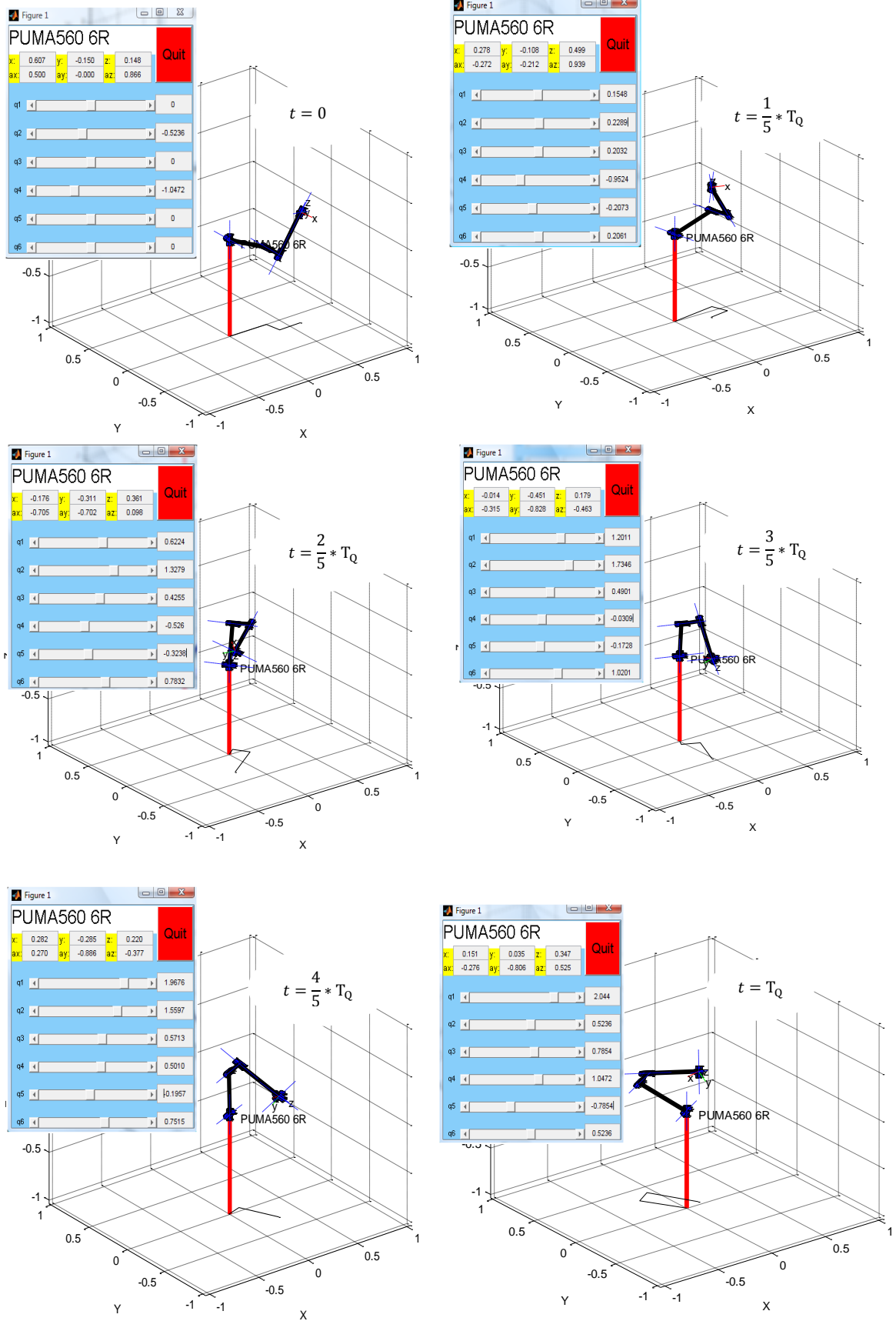


Figure 4.26: Aspect optimal de mouvement de robot PUMA 560 (6R)

4.7 Discussion des résultats:

A travers les exemples traités auparavant, et pour lesquels nous avons optimisé la trajectoire de différents robots manipulateurs on constate que les résultats obtenus ont le même ordre de grandeur que ceux trouvés dans les références citées.

Pour le premier exemple, dans premier temps nous avons testé l'influence des nombres de nœuds sur la qualité des résultats obtenus, en conséquence nous avons trouvé qu'au fur et à mesure qu'on augmente le nombre de nœuds des fonctions splines cubiques la qualité des résultats est meilleure, par ailleurs l'utilisation d'un nombre important des nœuds peut pénaliser l'effort de calcul principalement le codage des chromosomes pour des systèmes qui possèdent un nombre de degré de liberté élevé, de ce fait nous avons fixé le nombre des nœuds égale à cinq (trois intermédiaire) pour les prochaines exemples, et dans un second temps nous avons étudié l'influence du coefficient de pondération sur les résultats, par conséquent nous avons trouvés une augmentation de temps de transfert au fur et à mesure avec la minimisation du facteur de pondération. Rappelons que ce premier exemple est mis juste pour voir l'influence de ces deux termes sur notre démarche.

Le deuxième exemple a été traité par [82], cette référence a utilisé le Principe de Maximum de Pontriaguine (PMP) ceci aurait pu constituer un test important de validation de notre démarche car PMP est connu pour fournir la solution optimale du problème. En effet, les allures des positions articulaires que nous nous trouvés sont similaires à celles de la référence [82] la différence réside dans l'allure de la vitesse de la première articulation qui fournit une grande valeur ce qui reflète automatiquement sur les accélérations et par conséquent sur les commandes (type bang-bang) et le temps final de transfert.

Pour le troisième exemple, le critère minimisé est un critère mixte avec puissances quadratiques, le résultat obtenu est de même ordre de grandeur que celui publié dans [79] obtenu par la méthode SQP (temps de transfert égal à 5.76), la référence citée ne donne pas la valeur de la fonction objectif. En outre, les mêmes allures de trajectoires sont trouvées, la différence réside toujours dans les autres allures ceci s'explique comme nous avons déjà dit aux modèles des formes de trajectoires choisies (Splines cubiques).

Pour le quatrième exemple, l'auteur de la référence [83] a utilisé le Principe de Maximum de Pontriaguine (PMP), malheureusement les résultats fournis par cette référence ne permettent pas d'effectuer une comparaison complète. En effet, la valeur de la fonction cout ainsi que l'évolution des variables articulaires et de ces dérivées ne sont pas données. Les seules indications disponibles sont le temps de transfert (0.83 secondes) et l'évolution des couples articulaires. Pour notre part, nous avons trouvés une durée de transfert égale à 0.87 secondes ce qui reste très acceptables.

Dans le dernier exemple, nous avons traité le problème de planification de trajectoires pour un robot manipulateur universel de type PUMA 560 qui possède six articulations rotoïde, cet exemple est considéré comme un cas concret pour juger notre démarche car le robot possède un nombre de degré de liberté élevé ainsi qu'il subit des contraintes géométriques, cinématiques et dynamiques, par conséquence les résultats que nous avons trouvés concernant la fonction cout et le temps optimale de transfert sont très proches à ceux obtenus par la référence [79] qui a utilisé une technique d'optimisation déterministe, la seule différence trouvées concernant les allures des dérivées des trajectoires articulaires s'exprime par le choix des fonctions splines cubiques utilisées pour modéliser ces trajectoires.

Cependant, l'utilisation d'une technique d'optimisation stochastique basée sur une recherche de caractère aléatoire ne donne pas exactement les mêmes résultats pour chaque exécution des codes de calculs. Pour cela, nous avons effectué une centaine d'exécutions du code de calcul pour le deuxième problème (2 d.d.l), nous présentons dans (figure 4.20) les résultats finaux trouvés.

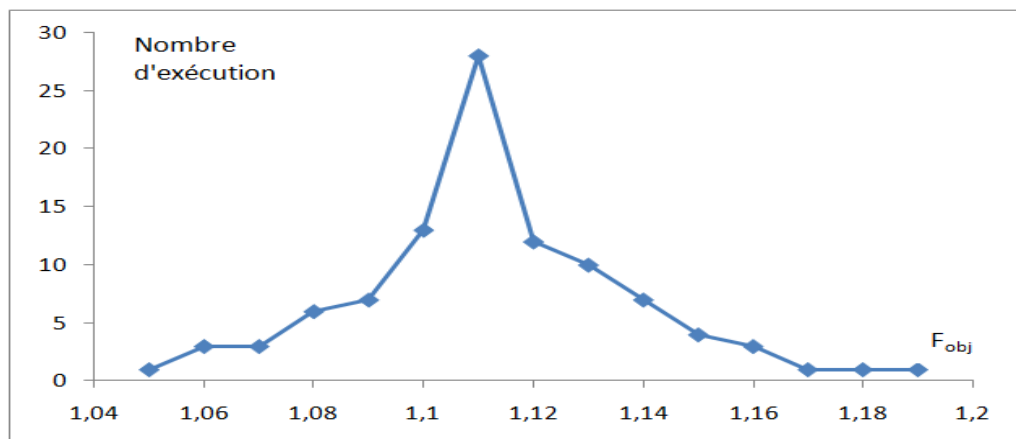


Figure 4.27: Reproductibilité des résultats du deuxième exemple.

Concernant le temps de résolution, notre approche donne les résultats dans des temps de calcul acceptables: le premier problème à deux degré de liberté est résolu dans 6 seconds environs, celui de quatrième problème relatif à quatre degré de liberté est résolu dans 35 secondes et pour le dernier exemple il nécessite un temps de calcul de 210 seconds sur un PC de 2 GHz.

4.8 Conclusion:

A travers ce chapitre, nous avons montré la possibilité de l'approche de nous donner des résultats plausibles et très satisfaisants aux problèmes de planification de trajectoires de robots manipulateurs. Nous avons pu traiter plusieurs exemples ou on a considéré toujours la dynamique du système avec un changement des autres paramètres (d.d.l, les contraintes imposées, la fonction objectif,..) de manière à tester l'habilité de cette approche à traiter les problèmes compliqués. Toutefois, signalons les difficultés que nous avons déjà rencontrées pour trouver des cas typiques de comparaisons, en outre les problèmes résolus que nous avons pu obtenir ne donnent pas suffisamment d'informations (fonction objectif, les allures de mouvement, de vitesses et d'accélérations, temps de calcul) pour pouvoir effectuer des comparaisons complètes. En particulier, nous n'avons pas pu trouver de cas résolus incluant les frottements.

CONCLUSION GENERALE ET PERSPECTIVES

Le travail présenté dans ce mémoire traite le problème de planification optimale des trajectoires libres pour un bras manipulateurs il constitue en particulier une tentative d'amélioration des travaux réalisés auparavant sur les robots manipulateurs puis ont été élargies à la robotique mobile, notre objectif était d'exploiter les avantages obtenues dans les travaux de la robotique mobile et ensuite les appliques sur les robots manipulateurs. Ce travail est aussi, une recherche de nouvelle technique de résolution des problèmes de planification de trajectoires des bras manipulateurs qui offre l'efficacité et la facilité de la mise en œuvre et en même temps la qualité des résultats obtenus, en se référant à d'autres techniques disponibles dans la littérature.

Le problème de planification optimale du trajectoires le long d'une tâche de transfert est formulé mathématiquement sous forme d'un problème d'optimisation sous contraintes dans lequel l'objectif est de minimiser une quantité physique représenté par une fonction objectif, tout en respectant certaines limitations et exigences de nature technologiques liées particulièrement au robot, à la tâche et à l'environnement, la résolution de ce problème conduit à la détermination de la trajectoire optimale que ce robot doit suivre ainsi que les commandes optimisées qui permettent d'engendrer un mouvement le long de cette trajectoire. Pour cela, ce mémoire a été partagé en quatre chapitres, nous avons consacré le premier chapitre aux définitions de certaines notions utilisées couramment et les différentes modélisations nécessaires à l'étude des bras manipulateurs. Tandis que dans le deuxième chapitre nous avons présenté les différentes méthodes de planification de trajectoire tout d'abord dans l'espace articulaire ensuite dans l'espace opérationnel nous avons vu que le choix d'un espace dépend du mode de description des trajectoires ainsi que la façon dont sont exprimées les contraintes.

Dans le troisième chapitre, notre premier objectif été de choisir une technique appropriée qui garde la simplicité de mise en œuvre indépendamment de la complexité du problème, et qui sera ainsi de caractère

générale pour pouvoir l'appliquer pour plusieurs type de problème sans qu'il sera nécessaire de modifier le code de calcul pour chaque cas traité. Pour cela nous avons adopté une approche qui a donnée des résultats sub-optimaux pour les plateformes mobiles et nous avons exploité ces avantages pour la rendre applicable à notre problème, cette approche consiste en premier temps à dissocier la recherche des positions articulaires de la détermination du temps de transfert final T , en utilisant une normalisation de l'échelle de temps ce qui permet de transformer le problème à temps final inconnu en un problème équivalent dont les bornes sont totalement connus, par la suite ce profil de trajectoire sera décomposé en un chemin et en un mouvement sur ce chemin. La mise en œuvre de cette approche à nécessité de reformuler le problème sur la base de ces idées, pour cela les contraintes géométriques, cinématiques et dynamiques seront transformées sous forme d'intervalles admissibles pour les valeurs du temps de transfert T , alors que les critères de performances sont réécrits sous formes d'une fonction à variable unique simple à optimiser. Il est à noter que la prise en compte des contraintes géométriques se fait par la fonction de chemin cela permet de sélectionner que les profils faisables pour poursuivre le processus d'optimisation.

Pour modéliser la forme de chemin et la forme de mouvement nous avons proposé un modèle constitué d'un ensemble de fonction splines cubiques de type "natural" pour la fonction chemin et "clamped" pour la fonction mouvement, les fonctions splines cubiques permettent d'assurer la continuité des vitesses, des accélérations et même des jerks et elles sont raccordées entre eux avec des nœuds qui sont laissés libres et peuvent prendre n'importe quelle position horizontale ou verticale, ceci permet de donner plus de flexibilité au modèle pour que la forme de profil peut prendre les formes voulues. Finalement, pour cette approche sera finalisée elle nécessite une technique d'optimisation appropriée pour générer les formes de profils et chercher parmi eux la forme optimale, pour cela nous avons choisi les algorithmes génétiques, l'avantage de cette technique par rapport aux autres et qu'on peut coder les paramètres de solution (chromosomes) pour limiter l'espace de recherche dans le cas des systèmes qui possèdent des contraintes géométriques, en effet le principe de base de cette technique

consiste à perturber les nœuds intermédiaires pour générer aléatoirement des profils des deux fonctions chemin et mouvement et par conséquent pour trouver un profil de trajectoire viable ce profil doit satisfaire toutes les contraintes, cette opération est refaite plusieurs fois est la meilleure valeur va être retenue.

Le quatrième chapitre a été consacré à l'examinait de l'approche proposée, la mise en œuvre à nécessité au début d'effectuer un choix concernant le nombre suffisant de nœuds libre. A cet effet, et après des tests effectués nous avons décidé de prendre cinq nœuds pour les prochains exemples. Tous les exemples que nous avons traités ont été comparés avec d'autres traiter par des techniques déterministes, les résultats obtenus ont montrés que la méthode peut approcher des solutions de type sub-optimal, ainsi que cette méthode possède une bonne capacité de reproduire les mêmes résultats et elle peut converger dans des temps de calcul acceptables.

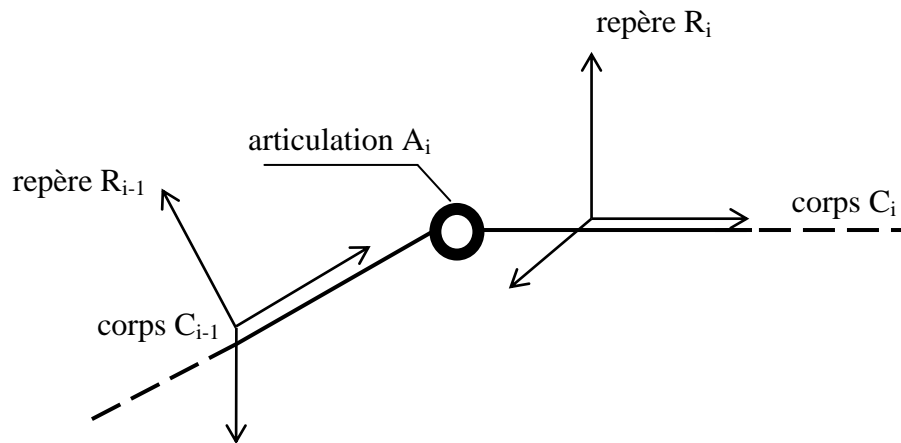
Comme perspectives, nous envisageons divers aspect concernant le problème d'optimisation dynamique de trajectoires pour les systèmes robotisés:

- La prise en compte des contraintes dans l'espace de travail à savoir l'évitement des obstacles stables ou mobiles, en exploitant l'avantage de décomposition du profil de la trajectoire.
- Traiter des problèmes similaires en incluant les frottements dans la dynamique des robots.
- Utilisation d'autres fonctions d'interpolation ou d'approximation (Bézier, B-Spline, ...).
- Extension aux robots manipulateurs sous actionnés.
- Utilisation d'autres techniques d'optimisation (Réseaux de neurone, les algorithmes de fourmi, ...).

APPENDICE A

TRANSFORMATIONS HOMOGÈNES - OPERATIONS MATRICIELLES

Coordonnées généralisées:



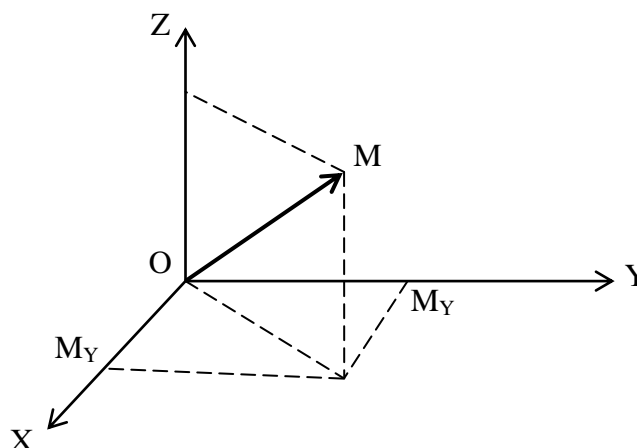
Les paramètres (réels et indépendants) qui permettent de définir la situation du corps C_i par rapport au corps C_{i-1} sont appelés coordonnées généralisées. Dans l'espace, 6 paramètres suffisent à connaître la situation d'un corps.

Matrices de transformations homogènes:

On introduit la notion de coordonnées homogènes (largement utilisées en informatique graphique) pour calculer les projections et perspectives d'un objet sur un écran, maintenant, très répandues en robotique.

Coordonnées homogènes d'un point:

C'est un vecteur colonne à 4 composantes:



$$\overrightarrow{OM} \begin{vmatrix} M_x \\ M_y \\ M_z \\ 1 \end{vmatrix}$$

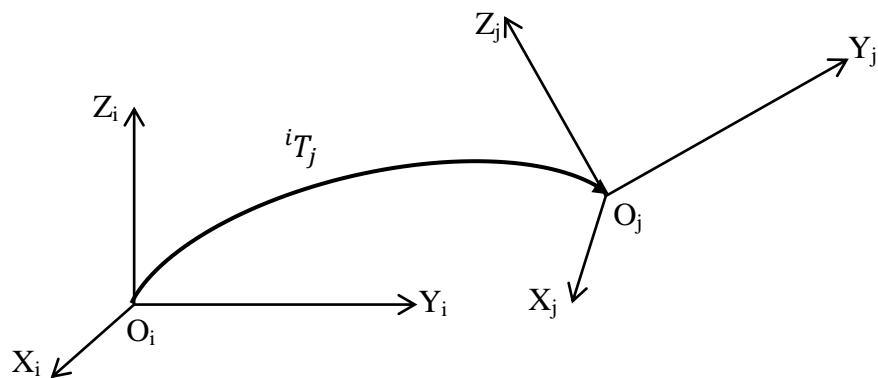
M_x , M_y et M_z sont les coordonnées de M dans le système $Oxyz$, la 4^{ième} valeur est un facteur d'échelle égal à 1 en robotique.

Coordonnées homogènes d'un vecteur libre:

C'est 1 vecteur constitué de 4 facteurs. Les trois premiers étant les composantes du vecteur libre, le 4^{ième} est égal à 0.

$$\vec{v} \begin{vmatrix} v_x \\ v_y \\ v_z \\ 0 \end{vmatrix}$$

Transformations homogènes:



${}^i T_j$ transforme le repère R_i en R_j . Elle peut être une (des) translation (s) et/ou une (des) rotation (s). Cette transformation se définit par une matrice (4x4).
Telle que :

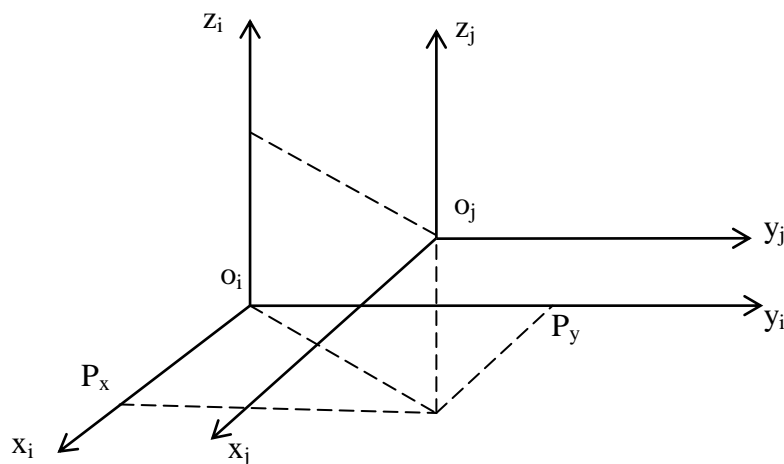
$$\begin{aligned}
 {}^i T_j &= \begin{vmatrix} \vec{x}_i \vec{x}_j & \vec{x}_i \vec{y}_j & \vec{x}_i \vec{z}_j & P_x \\ \vec{y}_i \vec{x}_j & \vec{y}_i \vec{y}_j & \vec{y}_i \vec{z}_j & P_y \\ \vec{z}_i \vec{x}_j & \vec{z}_i \vec{y}_j & \vec{z}_i \vec{z}_j & P_z \\ 0 & 0 & 0 & 1 \end{vmatrix} = \\
 &= \begin{vmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} {}^i s_j & {}^i n_j & {}^i a_j & {}^i P_j \end{vmatrix}
 \end{aligned}$$

Ou: ${}^i s_j$, ${}^i n_j$ 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 où ${}^i P_j$ est l'origine du repère R_j exprimée dans le repère R_i . Cette matrice peut être notée sous la forme partitionnée suivante:

$${}^i T_j = \begin{vmatrix} & {}^i A_j & {}^i P_j \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Ou: ${}^i A_j$ représente la matrice de passage du repère R_i au repère R_j et ${}^i P_j$ représente l'origine du repère R_j exprimée dans le repère R_i .

Translation pure:



Trois translations sont possibles :

- $T(x, P_x)$
- $T(y, P_y)$
- $T(z, P_z)$

Où : P_x , P_y et P_z représentent les translations respectivement par rapport à x , y et z .

La transformation totale ${}^i T_j$ est le produit des trois translations (l'ordre de multiplication est quelconque).

$${}^i T_j = \begin{vmatrix} 1 & 0 & 0 & P_x \\ 0 & 1 & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$${}^i T_j = T(x, P_x) \cdot T(y, P_y) \cdot T(z, P_z)$$

Rotation pure:

On peut considérer une rotation autour de x d'un angle θ de la manière suivante:

$$Rot(x, \theta) = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & c \theta & -s \theta & 0 \\ 0 & s \theta & c \theta & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} A(x, \theta) & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Par analogie, la rotation autour de y d'un angle θ sera donc:

$$Rot(y, \theta) = \begin{vmatrix} c \theta & 0 & s \theta & 0 \\ 0 & 1 & 0 & 0 \\ -s \theta & 0 & c \theta & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} A(y, \theta) & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Et la rotation autour de z d'un angle θ sera:

$$Rot(z, \theta) = \begin{vmatrix} c \theta & -s \theta & 0 & 0 \\ s \theta & c \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} A(z, \theta) & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Propriétés des matrices de transformations homogènes:

Comme :

$${}^i T_j = \begin{vmatrix} {}^i A_j & {}^i P_j \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Rotation pure:

Pour une rotation pure: ${}^iP_j = 0$

Translation pure:

Pour une translation pure: ${}^iA_j = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$

Matrice inverse

Comme la matrice iA_j est orthogonale: ${}^iA_j^T = {}^iA_j^{-1}$

La matrice inverse est égale à la matrice transposée. Leurs éléments représentent les cosinus directeurs d'orientation.

Transformation inverse:

$${}^iT_j^{-1} = {}^jT_i$$

Rotation inverse:

$$Rot^{-1}(u, \theta) = Rot(-u, \theta) = Rot(u, -\theta)$$

Transformation inverse:

L'inverse d'une matrice de transformation peut être écrit sous la forme suivante:

$${}^iT_j^{-1} = \begin{vmatrix} & & & -s^T P \\ A^T & & & -n^T P \\ & & & -a^T P \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} & & & -A^T P \\ A^T & & & \\ & & & \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Produit des transformations:

Si un repère R_0 a subit k transformations consécutives, la transformation totale sera égale à:

$${}^0T_k = {}^0T_1 \cdot {}^1T_2 \cdot \dots \cdot {}^{k-2}T_{k-1} \cdot {}^{k-1}T_k = \prod_{i=k}^{i=1} {}^{i-1}T_i$$

Transformations composées:

Composition de deux matrices:

$$\text{Si } T_1 = \begin{vmatrix} & A_1 & P_1 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad \text{et} \quad T_2 = \begin{vmatrix} & A_2 & P_2 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$\text{Alors: } T_1 \cdot T_2 = \begin{vmatrix} & A_1 A_2 & A_1 P_2 + P_1 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$\text{Avec: } T_1 \cdot T_2 \neq T_2 \cdot T_1$$

Produit des rotations:

$$Rot(u, \theta_1) \cdot Rot(u, \theta_2) = Rot(u, \theta_1 + \theta_2)$$

2 rotations autour du même axe u .

$$Rot(u, \theta) \cdot Trans(u, d) = Trans(u, d) \cdot Rot(u, \theta)$$

A condition que la rotation et la translation se fassent par rapport au même axe u .

Décomposition de la matrice de transformation:

$$T_1 = \begin{vmatrix} & A & P \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} & I & P \\ 0 & 0 & 0 & 1 \end{vmatrix} \times \begin{vmatrix} & A & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

APPENDICE B

REPERAGE D'UN SOLIDE DANS L'ESPACE

Orientation d'un solide dans l'espace:

Dans l'espace, un solide libre et rigide possède 6 degrés de liberté. A un instant donné, il faut pouvoir le positionner et l'orienter par rapport à un repère fixe qu'on choisit préalablement. En robotique, ces paramètres représentent les coordonnées opérationnelles définies dans un espace opérationnel. La matrice de transformation s'écrit sous la forme:

$$T = \begin{vmatrix} & A & P \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

La matrice A (3x3) représente la matrice d'orientation, la matrice P (3x1) celle de position. On associe le repère R_n à un solide, R_0 étant le repère de référence fixe. Pour les paramètres de positionnement d'un solide dans l'espace, on utilise les coordonnées cartésiennes, cylindro-polaires ou sphériques. Quant aux paramètres d'orientation, on dispose des cosinus directeurs, angles d'EULER, des angles de roulis-tangage-lacet et des paramètres d'EULER (appelés aussi paramètres d'OLINDE-RODRIGUES ou quaternions).

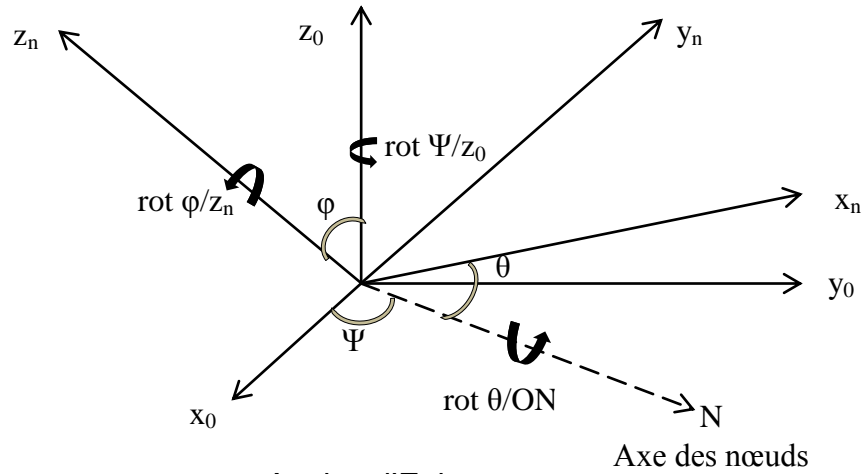
1. Cosinus directeurs:

L'orientation est définie par la donnée de trois vecteurs unitaires s , n et a qui sont les cosinus directeurs. La matrice correspondante s'écrit:

$$A_{CD} = \begin{vmatrix} s_x & n_x & a_x \\ s_y & n_y & a_y \\ s_z & n_z & a_z \end{vmatrix}$$

2. Angles d'EULER:

L'orientation de R_n par rapport à R_0 est donnée par 3 angles ψ , θ et φ .



Angles d'Euler

Ψ : rotation autour de z_0 (angle de précession).

θ : rotation autour de l'axe des nœuds \overline{ON} (angle de nutation).

φ : rotation autour de z_n (angle de rotation propre).

La matrice d'orientation s'écrit :

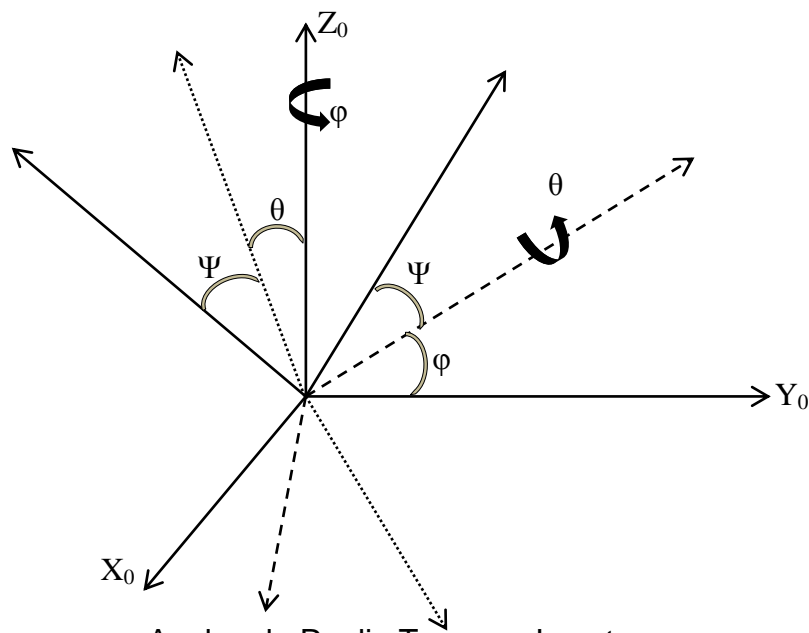
$$A_{EULER} = A(z, \Psi) \cdot A(x, \theta) \cdot A(z, \varphi)$$

Ou:

$$A_{EULER} = \begin{vmatrix} c\Psi c\varphi - s\Psi c\theta s\varphi & -c\Psi s\varphi - s\Psi c\theta c\varphi & s\Psi s\theta \\ s\Psi c\varphi + c\Psi c\theta s\varphi & -s\Psi s\varphi + c\Psi c\theta c\varphi & -c\Psi s\theta \\ s\theta s\varphi & s\theta c\varphi & c\theta \end{vmatrix}$$

3. Angles de Roulis-Tangage-Lacet (RTL):

L'orientation de R_n par rapport à R_0 est donnée par 3 angles φ , θ et ψ désignent respectivement le roulis, le tangage et le lacet:



Angles de Roulis-Tangage-Lacet

- Rotation φ autour de z.
- Rotation θ autour de y.
- Rotation ψ autour de x.

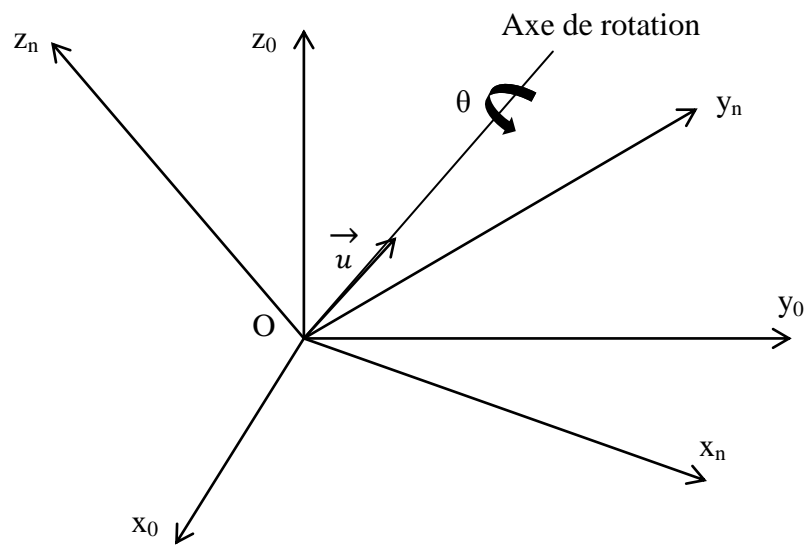
La matrice d'orientation s'écrit:

$$A_{RTL} = A(z, \varphi) \cdot A(y, \theta) \cdot A(x, \Psi)$$

$$A_{RTL} = \begin{vmatrix} c\varphi c\theta & c\varphi s\theta s\Psi - c\varphi c\Psi & c\varphi s\theta c\Psi + s\varphi s\Psi \\ s\varphi c\theta & s\varphi s\theta s\Psi + c\theta c\Psi & s\varphi s\theta c\Psi - c\varphi s\Psi \\ -s\theta & c\theta s\Psi & c\theta c\Psi \end{vmatrix}$$

4. Paramètres d'EULER:

L'orientation est exprimée par 4 paramètres décrivant une rotation unique autour d'un axe de vecteur unitaire u .



Paramètres d'EULER

On définit les paramètres d'EULER comme suit:

$$\lambda_1 = c(\theta/2)$$

$$\lambda_2 = u_x s(\theta/2)$$

$$\lambda_3 = u_y s(\theta/2)$$

$$\lambda_4 = u_z s(\theta/2)$$

La matrice d'orientation s'écrit:

$$A_E = \begin{vmatrix} 2(\lambda_1^2 + \lambda_2^2) - 1 & 2(\lambda_2\lambda_3 - \lambda_1\lambda_4) & 2(\lambda_2\lambda_4 - \lambda_1\lambda_3) \\ 2(\lambda_2\lambda_3 - \lambda_1\lambda_4) & 2(\lambda_1^2 + \lambda_3^2) - 1 & 2(\lambda_2\lambda_4 - \lambda_1\lambda_2) \\ 2(\lambda_2\lambda_4 - \lambda_1\lambda_3) & 2(\lambda_3\lambda_4 - \lambda_1\lambda_2) & 2(\lambda_1^2 + \lambda_4^2) - 1 \end{vmatrix}$$

APPENDICE C

TECHNIQUES STOCHASTIQUES D'OPTIMISATION

Résoudre un problème d'optimisation, c'est trouver parmi un ensemble Ω de solutions possibles S , la solution S^* minimisant (ou maximisant) une fonction objectif $F(S)$ donnée. Les techniques déterministes sont efficaces à condition que $F(S)$ soit continue et comporte très peu d'extremums (minimums locaux) sinon les développements mathématiques deviennent très lourds et difficiles à établir. De plus, elles se basent sur la construction d'une direction de recherche privilégiée dans Ω donc, le risque de manquer l'optimum global S^* est d'autant plus grand quand $F(S)$ comporte plusieurs optimums locaux. Dans ce cas, les techniques stochastiques qui se basent sur une recherche aléatoire de S^* uniformément distribuée dans l'espace Ω sont préférées. Seules des informations sur les valeurs de $F(S)$ sont nécessaires. Les techniques stochastiques les plus répandues sont :

- La technique de Monté-Carlo
- Méthode de Hill-Climbing
- Méthode de Monté-Carlo avec réduction d'intervalle
- Le recuit simulé
- La recherche tabou
- Les algorithmes génétiques

1- Technique de Monté-Carlo:

Elle consiste à générer aléatoirement plusieurs solutions S , de les comparer en retenant à chaque fois la meilleure solution [36]. L'algorithme du cycle de base se présente comme suit :

Générer aléatoirement une solution initiale S_0 dans Ω

Prendre $S^* = S_0$

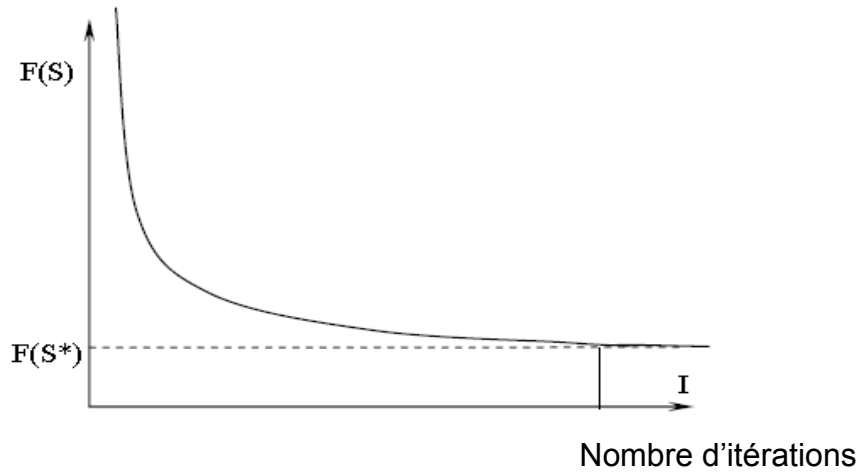
Générer aléatoirement une solution S dans Ω

Evaluer $F(S)$

Si $F(S) < F(S_0)$ alors $S^* = S$

Fin

Répéter

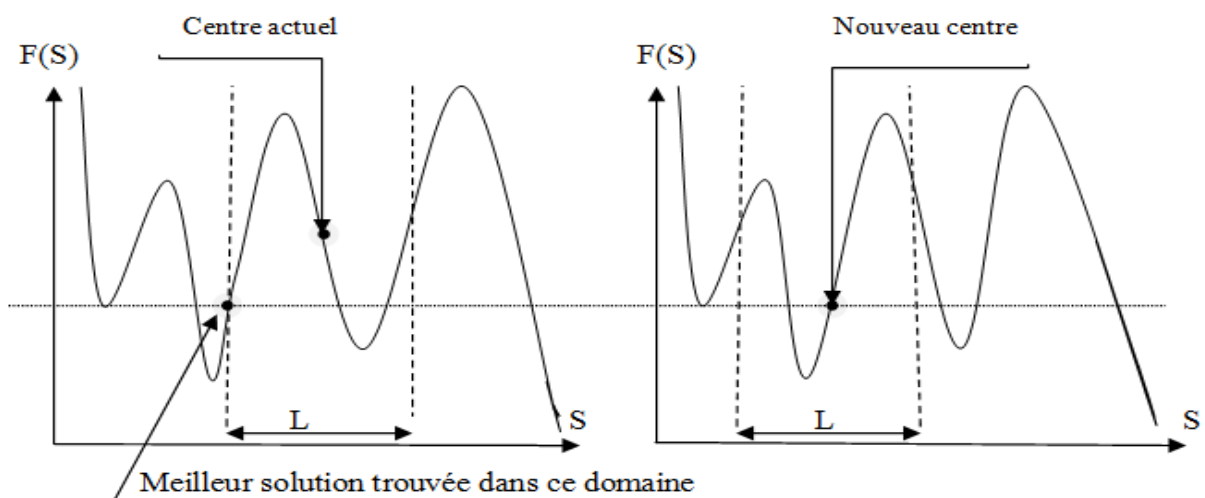


Convergence d'une technique de Monté-Carlo

L'application de cette méthode semble aisée. Il suffit d'évaluer la fonction F à chaque tirage aléatoire de S . Le temps de calcul est infini. Pour des raisons pratiques, le processus de base est arrêté après un nombre d'itérations I suffisant pour que la solution recherchée soit acceptable. Pour réduire l'effort de calcul et accélérer la convergence, on introduit la notion de voisinage $V(S_i)$ défini par son étendue L_i de la meilleure solution S_i obtenue à l'étape i . Cette idée est exploitée dans les techniques de Hill-Climbing, Monté- Carlo, recuit simulé et recherche Tabou.

2- Méthode de Hill-Climbing:

Elle propose d'effectuer une recherche dans un voisinage $V(S)$ d'étendue fixe L centrée autour de la meilleure solution retenue [37].



Changement du centre dans la technique de Hill-Climbing

La seule difficulté réside dans le choix de la taille de L. Un mauvais choix induit à :

- Piégeage de la technique dans un minimum local (L trop petit).
- Augmentation du temps de calcul (L trop grand).

L'algorithme correspondant est comme suit :

Générer aléatoirement une solution initiale S_0
 $S^* = S_0$
Centre de voisinage = S^*
Générer aléatoirement S dans le voisinage $V(S^*) \in \Omega$
Evaluer $F(S)$
Si $F(S) < F(S^*)$ alors $S = S^*$
Centre du voisinage = S^*
Jusqu'à convergence

3- Méthode de Monté-Carlo avec réduction d'intervalle:

C'est une version améliorée de la méthode précédente. Elle introduit un facteur de réduction de l'intervalle L. Lors du tirage initial, L est choisi large. Au fur et à mesure que le nombre d'itérations i augmente, la taille de L est réduite afin d'affiner la solution.

Soit E le nombre d'échecs successifs comptabilisés depuis la dernière amélioration (on considère un échec quand le choix de S fait augmenter $F(S)$). Quand E prend des valeurs significatives et pour rentabiliser le calcul, on réduit l'intervalle L. Pour cela, on se fixe un seuil E_{max} pour une prise de décision.

L'algorithme correspondant sera alors :

Générer une solution initiale S_0 aléatoirement
 $S^* = S_0$
Générer une solution S dans $V(S^*) \in \Omega$, évaluer $F(S^*)$
Si $F(S) < F(S^*)$, alors $S^* = S$
Sinon compter E
Si $E = E_{max}$
Réduire $V(S)$
Jusqu'à convergence

L'efficacité de cette méthode dépend du choix du facteur de réduction de L et de Emax.

4- Recuit simulé :

Le fait, à tout prix, de diminuer la fonction objectif $F(S)$ peut conduire à une situation de piégeage de la recherche autour d'un optimum local. L'idée de cette méthode est de faire échapper $F(S)$ de cette situation (permettre l'augmentation de $F(S)$ pour pouvoir se déplacer vers l'optimum global) [38, 39].

En métallurgie, le recuit est un traitement thermique qui consiste à chauffer un métal à un niveau tel qu'il permet l'équilibre physico-chimique et structurel d'un matériau et de le refroidir par palier afin que les atomes s'organisent de manière à le faire passer (le matériau) d'une configuration de haute énergie à celle d'énergie minimale. Ce passage obéit à la loi de Boltzman :

$$P = \exp [dE/KT]$$

Ou :

K : constante de Boltzman

dE : différence d'énergie entre les niveaux initial et final.

F : température.

P : probabilité de passage d'un groupe d'atomes d'un niveau d'énergie E_1 à un niveau d'énergie supérieure E_2

Par analogie, les configurations stables correspondent aux solutions qui font diminuer la fonction objectif, les configurations instables font augmenter la fonction objectif. La loi de Boltzman adaptée s'écrit alors :

$$P = \exp[\Delta F/T]$$

Où :

$$\Delta F = F(S_{i+1}) - F(S_i)$$

T correspond à un paramètre de contrôle

P représente la probabilité d'acceptation

5- Recherche Tabou:

Cette technique consiste à se déplacer d'une solution à une autre en s'interdisant de revenir à une solution déjà rencontrée. On définit un voisinage $V(S)$ pour chaque

solution S en supposant qu'on dispose d'une liste Tab de toutes les solutions rencontrées depuis le début de l'exécution de la méthode. A partir de la solution courante S_i on passe à une solution $S_{i+1} \in V(S_i)$ minimisant la fonction objectif et en ajoutant S_i à la liste Tab . En pratique, ce processus utilise trop de place mémoire ainsi que le temps de calcul (on doit comparer chaque solution choisie avec tous les éléments de la liste Tab). Pour éviter cet inconvénient, la méthode Tabou préconise de conserver en mémoire la transformation élémentaire qui permet de passer de la solution courante à la suivante en s'interdisant d'appliquer son inverse (la solution courante n'est pas ajoutée à la liste Tab). La recherche tabou se prête bien au problème d'optimisation discret (optimisation dans les réseaux) [40] [41] [43] [43].

L'algorithme correspondant sera :

Générer aléatoirement une solution initiale S_0
 $S^* = S_0$
 $Tab = \emptyset$
Répéter
Générer une solution aléatoire S dans $V(S^*) \in \Omega$
Si $S \notin Tab$ alors évaluer $F(S)$
Si $F(S) < F(S^*)$ alors $S^* = S$
Sinon ajouter S à la liste Tab
Jusqu'à convergence

6- Algorithmes génétiques:

Les algorithmes génétiques sont une technique de recherche stochastique introduite par "Holland" dans les débuts des années 70, ils sont inspirés par l'évolution biologique des espèces et ils sont souvent utilisés dans l'optimisation globale d'une fonction F sur un espace de recherche S souvent très grand. Vu la facilité relative de la mise en œuvre de ces algorithmes et leur indépendance vis-à-vis du domaine d'application, ces méthodes ont été largement utilisées pour les problèmes d'optimisation en intelligence artificielle et en robotique. [31] [32]

Ils sont basés sur des mécanismes de sélection naturelle et sur la génétique. A partir d'une population de solutions potentielles (chromosomes) initiale arbitrairement choisie, on évalue leurs performances en utilisant des opérateurs simples : la

sélection, le croisement et la mutation et on recommence le cycle [42] [44]. Ces algorithmes diffèrent des méthodes précédentes par les deux points suivants :

- On utilise un codage des paramètres (solutions).
- On travaille sur une population de points au lieu d'un point unique.

L'algorithme sera le suivant :

Générer une population initiale

Répéter

Evaluer les performances de chaque individu

Les sélectionner et les regrouper par paires selon leurs performances

Générer une nouvelle population (appliquer les opérations de croisements et de mutations)

Jusqu'à convergence

APPENDICE D

LA PROGRAMMATION NON LINEAIRE:

LA PROGRAMMATION QUADRATIQUE SEQUENTIELLE (PQS)

Un problème d'optimisation paramétrée se pose généralement de la façon suivante:

$$\begin{aligned} \text{Min } & f(X) \\ & x \in R^n \end{aligned} \tag{1}$$

Soumise aux contraintes suivantes:

$$\begin{cases} G_i(X) = 0, & i = 1, \dots, m_e \\ G_i(X) \leq 0, & i = m_e + 1, \dots, m \\ X_L \leq X \leq X_U \end{cases} \tag{2}$$

Ou :

- X est le vecteur des paramètres de conception ou de commande ($x \in R^n$).
- $f(X)$ est la fonction objectif qui restitue un scalaire pour tout vecteur de conception ($f(X): R^n \rightarrow R$).
- $G(X)$ est une fonction vectorielle qui restitue les valeurs des contraintes type égalité ou inégalité évaluée à X ($G(X): R^n \rightarrow R$).

Dans l'optimisation avec contraintes, l'objectif général est de transformer le problème en un sous problème plus facile à résoudre à l'aide de processus itératifs. Une caractéristique des premières méthodes utilisées pour résoudre ce genre de problèmes est de transformer le problème original en un problème de base sans contraintes en utilisant une fonction de pénalité pour les contraintes. De cette façon le problème avec contraintes est résolu en utilisant une séquence de problèmes d'optimisation paramétrée sans contraintes, qui à la limite converge au problème avec contraintes. Ces méthodes sont considérées maintenant

relativement inefficaces et ont été remplacées par des méthodes gravitant autour des équations de Kuhn-Tucker "KT".

Les équations de KT sont les conditions nécessaires d'optimalité pour un problème d'optimisation avec contraintes. Si le problème est ce qu'on appelle problème de programmation convexe, c'est-à-dire la fonction objectif et les contraintes sont convexes, dans ce cas les conditions de KT deviennent nécessaires et suffisantes pour un optimum global. En se référant au problème général, les conditions de KT peuvent être écrites comme suit:

$$\begin{aligned} \nabla f(X^*) + \sum_{i=1}^m \lambda_i^* \nabla G_i(X) &= 0 \\ \lambda_i G_i(X^*) &= 0 \quad i = 1, \dots, m \\ \lambda_i^* &\geq 0 \quad i = 1, \dots, m \end{aligned} \quad (3)$$

La première équation décrit une dépendance entre les gradients de la fonction objectif et les contraintes actives au niveau de la solution. Les multiplicateurs de Lagrange $(\lambda_i, i = 1, \dots, m)$ sont nécessaires pour équilibrer entre les variations dans les valeurs de la fonction objectif et les gradients des contraintes. Du moment que seulement les contraintes actives sont incluses dans cette équation, les multiplicateurs de Lagrange correspondants aux contraintes inactives sont systématiquement nuls, ceci est implicitement indiqué dans les deux dernières équations de la relation (3).

La solution des équations de KT forme la base de plusieurs algorithmes de la programmation non linéaire. Ces algorithmes tentent de calculer directement les multiplicateurs de Lagrange. Les méthodes quasi-Newtoniennes garantissent une convergence super linéaire par accumulation d'informations du second ordre relatives aux équations de KT, en utilisant une méthode de mise à jour quasi Newtonienne. Ces méthodes sont généralement désignées par le nom de méthodes de Programmation Quadratique Séquentielle PQS (disponibles sous le logiciel Malab), du moment qu'un sous problème de programmation quadratique est résolu en chaque itération principale (ces méthodes sont également connues sous le nom de Programmation Quadratique itérative, Programmation Quadratique Réursive, etc.).

La méthode permet d'utiliser la méthode de Newton pour une optimisation avec contrainte généralement appliquée pour une optimisation sans contraintes. En effet, à chaque itération principale une approximation du Hessien de la fonction Lagrangienne est exécutée en utilisant une procédure de mise à jour. Elle est par la suite utilisée pour générer un sous problème de programmation quadratique dont la solution est utilisée pour former une direction de recherche pour une procédure de recherche linéaire.

Etant donné un problème d'optimisation décrit par les relations (1) et (2), l'idée de base est de formuler un sous problème de programmation quadratique en se basant sur une approximation de la fonction Lagrangienne:

$$L(X, \lambda) = f(X) + \sum_{i=1}^m \lambda_i g_i(X) \quad (4)$$

Le sous problème de programmation quadratique est obtenu en linéarisant les contraintes non linéaires au tour du point X_k .

Sous problème de Programmation Quadratique:

Il est de la forme:

$$\text{Min } \frac{1}{2} d^T H_k d + \nabla f(X_k)^T d \quad (5)$$

Sous les contraintes:

$$\nabla g_i(X_k)^T d + g_i(X_k) = 0 \quad i = 1, \dots, m_e \quad (6.1)$$

$$\nabla g_i(X_k)^T d + g_i(X_k) \leq 0 \quad i = m_e + 1, \dots, m_e \quad (6.2)$$

Ce sous problème peut être résolu en utilisant n'importe quel algorithme de programmation quadratique, la solution est utilisée pour former un nouveau itéré:

$$X_{k+1} = X_k + \alpha_k d_k \quad (7)$$

La longueur du pas α_k est déterminée par une procédure de recherche linéaire appropriée de telle façon une croissance suffisante dans la valeur de la fonction de mérite Ψ soit enregistré.

Cette fonction Ψ est construite comme suit:

$$\Psi(X_k) = f(X_k) + \sum_{i=1}^m r_i g_i(X_k) \quad (8)$$

$$r_i = (r_{k+1})_i = \max \left\{ \lambda_i, \frac{(r_k)_i + \lambda_i}{2} \right\} \quad i = 1, \dots, m$$

L'initialisation de paramètre de pénalité r_i est donnée par:

$$r_i = \frac{\|f_x(X)\|}{\|g_{x_i}(X)\|}$$

La matrice H_k est définie positive et approche la matrice Hessienne de la fonction Lagrangienne (4). Elle est initialisée par une matrice systématique définie positive, entre autres la matrice identité H_k peut être renouvelée par n'importe quelle méthode quasi Newtonienne bien que la méthode BFGS (Broyden, Fletcher, Goldfard, Shanno) soit la plus populaire.

Mise à jour de la matrice Hessienne:

A chaque itération principale, une approximation quasi-Newtonienne définie positive de la matrice Hessienne de la fonction Lagrangienne H , est calculée en utilisant la méthode BFGS selon le schéma suivant:

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k} \quad (9)$$

Avec :

$$s_k = X_{k+1} - X_k$$

$$q_k = \left(\nabla f(X_{k+1}) + \sum_{i=1}^m \lambda_i \nabla g_i(X_{k+1}) \right) - \left(\nabla f(X_k) + \sum_{i=1}^m \lambda_i \nabla g_i(X_k) \right)$$

L'algorithme associé est le suivant :

Algorithme de Broyden, Fletcher, Goldfard ET Shanno: [46] [47]

1. Choisir X et H_0 définie positive quelconque (par exemple $H_0 = I$)
2. A l'itération k , calculer la direction de déplacement

$$d_k = -H_k^{-1} \nabla f(X_k),$$

Déterminer le pas optimal α_k et poser:

$$X_{k+1} = X_k + \alpha_k d_k,$$

3. Poser $s_k = \alpha_k d_k$ et $q_k = \nabla f(X_{k+1}) - \nabla f(X_k)$ puis calculer

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k}$$

4. Faire $k \leftarrow k + 1$. Retourner en 2 sauf si le critère d'arrêt est vérifié.

APPENDICE E

LE PRINCIPE DU MAXIMUM DE PONTRIAGUINE (PMP)

Il permet une prise en compte directe et exacte des contraintes de bornes sur toute variable de commande qui est, dans la plupart des travaux utilisant ce principe, l'effort actionneur. En effet, l'une des conditions de maximalité de ce principe assure l'appartenance de la commande optimale au domaine des commandes admissibles. Dans la suite, nous détaillerons la formulation mathématique de base de cette approche afin de déceler ses points forts et ses points faibles.

1. Le modèle d'état:

La forme générale des équations du mouvement du système dynamique est de la forme :

$$\dot{X}_i = f_i(x_1, x_2, x_3, \dots, x_n, u_1, u_2, u_3, \dots, u_r), i = 1, 2, \dots, n \quad (2.01)$$

Ou:

- X: Le vecteur des variables d'état constitué des paramètres de positions et de vitesses articulaires.
- u: Le vecteur des commandes (couples articulaires).

L'équation (2.01) constitue un système de 2n équations différentielles non linéaires du 1^{er} ordre. Le travail du système est évalué par la fonctionnelle :

$$L = \int_0^T f_0(x_1, x_2, x_3, \dots, x_n, u_1, u_2, u_3, \dots, u_r) dt \quad (2.02)$$

La commande u sera déterminée dans la limite des contraintes imposées pour que la fonctionnelle (2.02) soit minimisée.

$u \in \theta$ (θ est l'intervalle des commandes admissibles).

On désigne par l la valeur minimale de (2.02) et on introduit les notations suivantes :

$$\Psi_0 = -1, \quad \Psi_i = -\frac{\partial l}{\partial x_i} \quad i = 1, 2, \dots, n, \quad H = \sum_{i=0}^n \Psi_i f_i$$

La fonction H représente le hamiltonien. A chaque instant, sur la trajectoire optimale, la commande optimale maximise H ($H_{max} = 0$). Il est impossible d'utiliser directement ce résultat étant donné que les fonctions ψ_j ($j = 1, 2, \dots, n$) sont inconnues. [6]

2. Enoncé du principe:

$$\max H = \sum_{i=0}^n \Psi_i f_i = 0, \quad u \in \theta \quad (2.03)$$

Les composantes ψ_j vérifient les équations suivantes:

$$\Psi_0 = -1 \quad (2.04)$$

$$\frac{\partial \Psi_i}{\partial t} = - \sum_{j=0}^n \Psi_j \frac{\partial f_j}{\partial x_i} \quad (2.05)$$

3. Conditions aux limites:

Si les équations du mouvement ont la forme (2.01), il faut faire passer le système de l'état initial:

$$x_i(0) = x_{i0} \quad ; \quad i = 1, 2, 3, \dots, n \quad (2.06)$$

A l'état final:

$$x_i(T) = x_{i1} \quad ; \quad i = 1, 2, 3, \dots, n \quad (2.07)$$

De telle manière que la fonctionnelle (2.02) ait une valeur maximale. La durée T du processus n'étant pas fixée. Parmi les inconvénients de cette méthode, on peut évoquer :

- La formulation du modèle dynamique d'un système peut être très complexe quand le nddl dépasse 3.

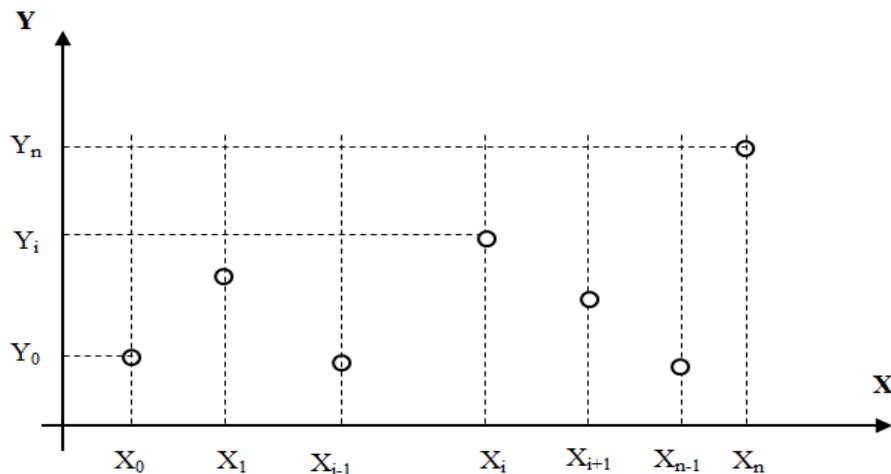
- Le problème se compliquera davantage si on doit ajouter à cela les dérivées que nécessitent les conditions d'optimalité.
- En présence d'obstacle, par exemple, la prise en compte des contraintes sur l'état du système pose un problème particulièrement ardu d'où, la nécessité d'introduire des artifices afin de traiter le problème dans sa totalité.

Si on opte pour une résolution numérique, le système différentiel nécessite une procédure d'initialisation d'une grande précision pour garantir une convergence vers un optimum global. [6]

APPENDICE F

INTERPOLATION POLYNOMIALE PAR MORCEAUX

"SPLINES CUBIQUES"



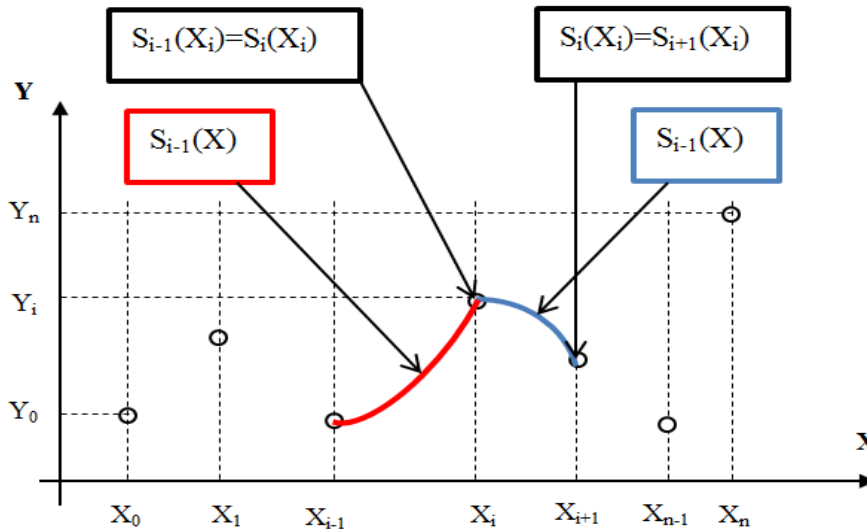
Une interpolation polynomiale [48] entre les points $[X_0, Y_0]$ et $[X_n, Y_n]$ peut être effectuée de la manière suivante :

- Utiliser un polynôme de degré n s'écrit:

$$f(x) = \sum_{i=0}^n a_i x^i$$

Qui passe par tous les points intermédiaires $[X_i, Y_i]$, du fait qu'il possède $(n-1)$ extremums, la courbe $f(x)$ va présenter des oscillations si n est trop élevé. Cette dernière ne sera pas régulière.

- Utiliser un polynôme de degré faible pour éviter ces inconvénients en effectuant une interpolation par morceaux. On préconise des fonctions splines qui peuvent être linéaires, quadratiques ou cubiques. Pour leur caractère de stabilité, ces dernières présentent des avantages au niveau des nœuds de raccordement entre les morceaux (bon lissage de la courbe totale) à l'inverse des splines linéaires qui présentent des points anguleux. Il suffit d'assurer la continuité de la courbe totale.



$S(x)$ représente la courbe totale reliant tous les points $[X_i, Y_i]$.

$$S(x) = \{S_i(x)\} \text{ avec } x \in [x_0, x_n]$$

$S_i(x)$ représente le morceau qui relie les points $[X_i, Y_i]$ et $[X_{i+1}, Y_{i+1}]$.

$S_i(x_i)$ représente la valeur de $S_i(x)$ au point x_i .

Soient $(n+1)$ points tels que $a = x_0 < x_1 < \dots < x_{n-1} < x_n = b$.

La fonction $S(x)$ est appelée spline cubique s'il existe N fonctions polynomiales $S_i(x)$ qui vérifient les propriétés suivantes:

$$1. S(x) = S_i(x) = a_{i,0} + a_{i,1}(x - x_i) + a_{i,2}(x - x_i)^2 + a_{i,3}(x - x_i)^3 \quad (\text{F. 1})$$

Pour tout $x \in [x_i, x_{i+1}]$ et pour $i=0, 1, \dots, n-1$.

$$2. S(x_i) = y_i \quad \text{pour } i = 0, 1, \dots, n. \quad (\text{F. 2})$$

Qui implique que la courbe passe par tous les points.

$$3. S_i(x_{i+1}) = S_{i+1}(x_{i+1}) \quad \text{pour } i = 0, 1, \dots, n-2. \quad (\text{F. 3})$$

Qui implique la continuité de la courbe totale $S(x)$.

$$4. S'_i(x_{i+1}) = S'_{i+1}(x_{i+1}) \quad \text{pour } i = 0, 1, \dots, n-2. \quad (\text{F. 4})$$

Qui implique que la fonction totale est lisse au niveau des nœuds de raccordement.

$$5. S''_i(x_{i+1}) = S''_{i+1}(x_{i+1}) \quad \text{pour } i = 0, 1, \dots, n - 2. \quad (\text{F. 5})$$

Qui implique que les dérivées secondes sont continuées.

Les inconnues des problèmes:

L'expression (E.1) contient 4 inconnues ($a_{i,0}, a_{i,1}, a_{i,2}, a_{i,3}$) pour l'intervalle $[X_i, X_{i+1}]$. Comme ces derniers sont au nombre de N ((N+1) points) alors, au total on doit déterminer 4N inconnues.

Equations disponibles:

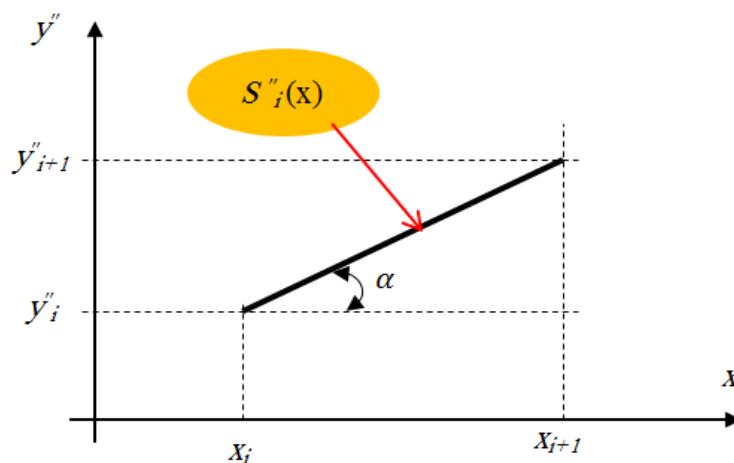
En principe, on peut établir (N+1) équations avec l'expression (E.2), (N-1) équations avec chacune des expressions (E.2), (E.2) et (E.2), ce qui fait au total $(N+1)+3(N-1)=4N-2$. Deux équations manquantes qu'on peut écrire aux extrémités de l'intervalle en faisant intervenir l'une des dérivés (première ou deuxième).

Construction des courbes:

Dans l'intervalle $[X_i, X_{i+1}]$ on peut toujours écrire que:

$$S''(x) = S''_i(x)$$

$$y''_{i+1} = S''_i(x_{i+1}) \quad \text{et} \quad y''_i = S''_i(x_i)$$



La linéarité de la dérivée seconde est telle que:

$$S''_i(x) - S''_i(x_i) = (x - x_i) \operatorname{tg} \alpha$$

Ou :

$$\operatorname{tg} \alpha = \frac{S''_i(x) - S''_i(x_i)}{x_{i+1} - x_i}$$

Ce qui donne après calcul:

$$S''(x) = S''_i(x) = S''(x_i) \frac{x - x_{i+1}}{x_i - x_{i+1}} + S''(x_{i+1}) \frac{x - x_i}{x_{i+1} - x_i} \quad (\text{F. 6})$$

On pose: $m_i = S''(x_i)$; $m_{i+1} = S''(x_{i+1})$; $h_i = (x_{i+1} - x_i)$

L'expression (E.6) devient:

$$S''_i(x) = \frac{m_i}{h_i} (x_{i+1} - x) + \frac{m_{i+1}}{h_i} (x - x_i) \quad (\text{F. 7})$$

On intègre 02 fois (E.7); ainsi on obtient:

$$S_i(x) = \frac{m_i}{6h_i} (x_{i+1} - x)^3 + \frac{m_{i+1}}{6h_i} (x - x_i)^3 + p_i(x_{i+1} - x) + q_i(x - x_i) \quad (\text{F. 8})$$

Ou p_i et q_i représentent des constantes d'intégration.

On remplace successivement dans l'expression (E.8) x par x_i et par x_{i+1} :

$$y_i = S_i(x_i) = \frac{m_i}{6} h_i^2 + p_i h_i$$

$$y_{i+1} = S_i(x_{i+1}) = \frac{m_{i+1}}{6} h_i^2 + q_i h_i$$

D'où :

$$p_i = \frac{y_i}{h_i} - \frac{m_i h_i}{6} \quad \text{et} \quad q_i = \frac{y_{i+1}}{h_i} - \frac{m_{i+1} h_i}{6}$$

Les seules inconnues de l'expression (E.8) sont les termes $\{m_i\}$ qu'on peut trouver en dérivant la même expression. Ainsi, on a:

$$S'_i(x) = -\frac{m_i}{2h_i} (x_{i+1} - x)^2 + \frac{m_{i+1}}{2h_i} (x - x_i)^2 + p_i + q_i \quad (\text{F. 9})$$

Cette dernière expression est valable quand $x = x_i$. Alors:

$$S'_i(x_i) = -\frac{m_i}{3}h_i - \frac{m_{i+1}}{6}h_i + \frac{y_{i+1} - y_i}{h_i} \quad (\text{F. 10})$$

Elle est aussi quand $i = i - 1$ donc:

$$S'_{i-1}(x_i) = -\frac{m_i}{3}h_{i-1} + \frac{m_{i-1}}{6}h_{i-1} + \frac{y_i - y_{i-1}}{h_{i-1}} \quad (\text{F. 11})$$

En utilisant la propriété 4 et les expressions (E.10) et (E.11), on obtient une relation qui lie m_{i-1} , m_i et m_{i+1} . Pour $i = 1, \dots, n - 1$.

$$h_{i-1} \cdot m_{i-1} + 2(h_{i-1} + h_i) + h_i \cdot m_{i+1} = 6 \left(\frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_i} \right) \quad (\text{F. 12})$$

L'équation (E.12) représente un système de (N-1) équations linéaires à (N+1) inconnues. Les deux équations manquantes sont obtenues aux extrémités. Une fois les valeurs de $\{m_i\}$ sont déterminées, les coefficients de la fonction spline $S_i(x)$ seront calculés comme suit:

$$a_{i,0} = y_i \quad , \quad a_{i,1} = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i(2m_i + m_{i+1})}{6}$$

$$a_{i,2} = \frac{m_i}{2} \quad , \quad a_{i,3} = \frac{m_{i+1} - m_i}{6h_i}$$

Chaque polynôme peut se mettre sous la forme:

$$S_i(x) = \{[a_{i,3}(x - x_i) + a_{i,2}](x - x_i) + a_{i,1}\}(x - x_i) + a_{i,0} \quad (\text{F. 13})$$

Ou: $x_i \leq x \leq x_{i+1}$

APPENDICE G

FONCTION B-SPLINE

1- Définition:[49]

Soit une suite non décroissante de réels positifs $\{noeud_0 \leq \dots \leq noeud_M\}$, appelée suite de nœuds, qui contient l'ensemble des points de raccordement. Les B-splines d'ordre $d=1$ sont définies par:

$$\forall j = \{0, \dots, M - 1\},$$

$$B_{j,1}(t) = \begin{cases} 1 & \text{si } noeud_j < t < noeud_{j+1} \\ 0 & \text{sinon} \end{cases}$$

Et les B-splines d'ordre $d \geq 2$ ($d \in \mathbb{N}$) par la relation de récurrence:

$$\forall j = \{0, \dots, M - d\},$$

$$B_{j,d}(t) = \frac{t - noeud_j}{noeud_{j+d+1} - noeud_j} B_{j,d-1}(t) + \frac{noeud_{j+d} - t}{noeud_{j+d} - noeud_{j+1}} B_{j+1,d-1}(t)$$

Lorsque plusieurs nœuds t_j sont confondus on pose $\frac{0}{0} = 0$. Quand les nœuds sont équidistants, les B-splines sont dites uniformes.

2- Propriétés des B-splines:

Avec les notations ci-dessous, on montre les propriétés suivantes:

1. $B_{j,d}(t) = 0, \forall t \notin [noeud_j, noeud_{j+i}]$. De plus $B_{j,d}(t) \geq 0$ sur $[noeud_j, noeud_{j+d}]$.
2. Les B-splines d'ordre d sont des fonctions polynomiales par morceaux de degré au plus $d-1$.
Si les nœuds sont tous distincts, ce sont des fonctions de classe C^{d-2} .
Lorsqu'un nœud est de multiplicité $1 \leq r \leq d - 1$, les B-splines n'est que $d - r - 1$ fois continument dérivable en ce point.

Si un nœud est de multiplicité d , il apparait alors une discontinuité dans le graphe de la B-spline.

A partir des relations de récurrence, on a:

$$\dot{B}_{j,d}(t) = (d - 1) \left(\frac{B_{j,d-1}(t)}{\text{noeud}_{j+d-1} - \text{noeud}_j} - \frac{B_{j+1,d-1}(t)}{\text{noeud}_{j+d} - \text{noeud}_{j+1}} \right)$$

Qui permet d'obtenir les dérivées successives des B-splines en tenant compte de la multiplicité des nœuds.

LISTE DES SYMBOLES ET DES ABREVIATIONS

PMP : Principe de Maximum de Pontriaguine.

SQP : Programmation Quadratique Séquentielle.

$q(t)$: Trajectoire articulaire.

$Q(\xi), Q'(\xi), Q''(\xi), Q'''(\xi)$: Forme de la trajectoire, de la vitesse, de l'accélération et des jerks.

$P(\lambda)$: Fonction de chemin.

$\lambda(\xi)$: Fonction mouvement.

α, β : Facteurs de pondération.

T : Le temps final de transfert.

τ : Vecteur des couples moteurs (les commandes).

N_c : Nombre de nœuds de raccordement des fonctions Splines Cubiques constituant $P(\lambda)$.

N_m : Nombre de nœuds de raccordement des fonctions Splines Cubiques constituant $\lambda(\xi)$.

J_Q : Fonction objectif.

F^s : Coefficients du frottement sec.

F^v : Coefficients du frottement visqueux.

P_i : Position initiale.

P_f : Position finale.

q^{init} : Configuration initiale.

q^{fin} : Configuration finale.

REFERENCES

1. W. KHALIL, E.DOMBRE « Modélisation identification et commande des robots », Hermès Sciences Publications, Paris, 1999.
2. P.COIFFET « La robotique : principes et applications », Hermès, Paris, 1986.
3. T. CHETTIBI « Contribution à l'exploitation optimale des bras manipulateurs », Mémoire de Magister, EMP, Bordj el Bahri, Juillet 2000.
4. M Giordano, J Lottin « cours de robotique description et fonctionnement des robots industriels» Edition Armand Colin, Paris, 1990.
5. A. BOUGUERRA « Contribution à la planification optimale des robots coopérants », Mémoire de Magister, Département de Mécanique, Université de Saad Dahlab de Blida, Janvier 2005.
6. J. J. Graig « Introduction to robotics, mechanics and control » Addison-Wesley 1989.
7. J. P. Lallemand, S. Zegloul « Robotique, aspects fondamentaux » Edition Masson 1994.
8. A. A. Kobrinski, A. E. Kobrinski « Bras manipulateurs des robots, architecture et théorie » Edition MIR Moscou 1989.
9. V. Arnold « Les méthodes mathématiques de la mécanique classique » Edition MIR Moscou 1976.
10. P. Fisette, H. Buyse, J.G. Samin «Meca 2732 : Introduction à la robotique» Nov 2004
11. T. Laliberté, C. M .Gosselin « Efficient algorithm for the trajectory planning of redundant manipulators with obstacles avoidance » IEEE international conference on robotics and automation, vol. 3, p. 2044-2049. May 1994.
12. Z. Shiller « Time energy optimal control of articulated systems with geometric path constraints » IEEE int. conf. on rob. and aut. Vol. 4 p. 2680-2685. May 1994.
13. T. Chettibi, H. E. Lehtihet « A new approach for point to point optimal motion planning problems of robotic manipulators » ESDA 2002. APM-10. ASME conf. 2002.

14. S. Rebiai « Contribution à la planification optimal des mouvements libres des robots manipulateurs » Mémoire de Magister, EMP, 2002.
15. R. V. Ma yorga « A framework for the path planning of robot manipulator » Lasted third int. conf. on rob. and manufacturing, p. 61-66. June 1995.
16. N. R. Shahri, I. Troch « A new on line methode to avoid collisions with links of redundant articulated robots » Robotica N° 14, p. 611-619. 1996.
17. F. Pfeiffer and J. Rainer « A concept for manipulator trajectory planning » IEEE journal for robotics automation, vol. Ra-3 N° 2, April 1987.
18. A. Shiller and H. H. Lu « Computation of path constrained time optimal motions with dynamic singularities » Trans of the ESME, vol. 114, p. 34-40. March 1992.
19. J. E. Bobrow, S. Dubowshy and J. S. Gibson « Time-optimal control of robotic manipulators along specified paths » The international journal of robotics research, vol. 4 N° 3. 1985.
20. I. Duleba « The method of minimal neighbourhood: a new and most effective iterative for minimum cost trajectory planning in robot manipulators » Robotica vol. 13 p. 297-304. 1992.
21. W. Jack, I. Deluba and P. Rogalinski « A graph- searching approach to trajectory planning of robots » Robotica vol. 10 p. 531-537. 1992.
22. Z. Shiller « On singular time-optimal control along specified paths » IEEE tran. on rob. and aut. 10(4) p. 561-566. August 1994.
23. L. Pantriaguine, V. Boltianski, R. Gamkrelidje, E. Michtchenko « Théorie mathématique des processus optimaux » Edition Mir 1965.
24. H.H.Tan, R.B.Potts «A discret trajectory planner for robotic arms with six degrees of freedom » IEEE Transactions on Robotics and Automation, vol. 5 N° 5, October 1989.
25. C.P.Neuman, V.D.Tourassis « Discret Dynamic Robot Models » IEEE Tran. Sys. Man Cybern, Vol.SMC-15, N°2, pp 193-204, Mar/Apr. 1985.
26. V.D.Tourassis « Inverse Dynamics Applications of Robot Models » IEEE Tran. Sys. Man Cybern, Vol.SMC-15, N°6, pp 798-803, Nov/Dec. 1985.
27. H.H.Tan, R.B.Potts « Minimum time trajectory planner for the discrete ynamic robot model with dynamic constraints » IEEE J. of Rob. Aut, Vol.4, N°2, pp 174-185, April 1988.
28. T. Chettibi « Research of optimal free motions of manipulators robots by nonlinear optimisation » SIGMA. Avril 2002.

29. O. Khatib « Real time obstacle avoidance for manipulators and mobile robots » International journal of robotics research, vol. 5 N° 1 O. 90-98. 1986.
30. M. Haddad « Exploitation optimale des plateformes mobiles et des manipulateur mobiles à roues » Thèse de Doctorat, EMP, 2007.
31. J. Larios «Le fil d'Ariane Une méthode de planification générale application à la planification» Thèse de Doctorat, INP de Grenoble, 1994.
32. K Mekhnacha « Méthode probabilistes Bayésiennes pour la prise en compte des incertitudes » Thèse de Doctorat, INP de Grenoble, 1999.
33. A.Berro « Optimisation multiobjectif et stratégies d'évolution en environnement dynamique » Thèse de Doctorat, Université de Toulouse I, 2001.
34. P. R. Bergamaschi, S. D. F. P. Saramago, L. D. S. Coelho « Comparative study of SQP and metaheuristics for robotic manipulator design » Applied Numerical Mathematics 58 (2008) 1396-1412.
35. C. G. Jonson, D. Marsh « Modelling robot manipulators with multivariate B-splines » Robotica (1999) Volume 17, pp. 239-247.
36. P. Athanasios « Probability, random variables and stochastic processes » WCB 1991.
37. S. Kirkpatrick, C. D. Gelatt and P. M. Vecchi « Optimization by simulated annealing » Science 220 p. 671-680. 1983.
38. J. Lam and J.-M. Delosme « Logic minimization using simulated annealing » Proc. International conference on computer aided design ICCAD 86, Santa- Clara CA p. 348-351. Nov. 1986.
39. J. Lam and J.-M. Delosme « Simulated annealing: a fast heuristic for some generic layout problems » Proc. International conference on computer aided design ICCAD 88, Santa- Clara CA p. 510-513. Nov. 1988.
40. R. Battiti, G. Tecchiolli « The reactive Tabu search » ORSA journal on computing p. 126-140. 1994.
41. F. Glover « Tabu search » Part 1 ORSA J. on computing 1(3) p. 190-206. 1989.
42. F. Glover « Tabu search » Part 2 ORSA J. on computing 2(1) p. 04-32. 1990.
43. T. Crainic, M. Gendreau, P. Soriano and M. Toulouse « A tabu search procedure for multicommodity location/allocation with balancing requirement » Annals of operations research 42 (1-4) p. 359-383. 1993.

44. R. Dorsey and W. Mayer « Genetic algorithms for estimation problems with multiple optima, nondifferentiability and other irregular features» Journal of business and economic statistics 13(1) p. 53-60. 1995.
45. K. Krishnakumar and D. Goldberg « Control system optimization using genetic algorithm » Journal of guidance, control and dynamics 15(3) p. 735-740. 1992.
46. S. Mottelet « Optimisation non-linéaire », Université de Technologie de Compiègne, cours 2003.
47. Matlab optimization Help.
48. J. H. Mathews «Numerical Methods for Mathematics, Science and Engineering» Prentice-Hall of India. Second Edition 1994.
49. M. Defoort « Contributions à la planification et à la commande pour les robots mobiles coopérants», Thèse de Doctorat, Ecole Centrale de Lille, Septembre 2007.
50. J. Angeles « Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms, 3rd Edition » Edition - Springer 2007.
51. L. Landau, E. Lifchitz « Mécanique » Edition MIR Moscou 1988.
52. R. L. Fox « Optimization methods for engineering design» Addison-Wesley 1971.
53. Glass K, Colbaugh R, Lim D, Seradji H, « Real time collision avoidance for redundant manipulators,» IEEE Transaction on rob. And aut. N°11 Vol3, pp 448-457, 1995.
54. Mayorga R.V, « A framework for the path planning of robot manipulators,» IASTED third Int. Conf. on Rob. and Manufacturing, pp 61-66 June 1995.
55. Mitsi S. Bouzakis, K. D, Mansour G. « Optimization of robot links motion in inverse kinematics solution considering collision avoidance and joints limits,» Mach. And mec. Theory N°30 vol 5 pp 653-663, 1995.
56. Laliberté T, Gosselin C.M, «Efficient algorithm for the trajectory planning of redundant manipulators with obstacles avoidance » IEEE International conference on robotics and automation, Vol3 pp2044-2049, May1994.
57. Cao B, Dodds G.I, Irwin G.W «An approach to time optimal, smooth and collision free path planning in a two robot arm environment» Robotica, Vol14 pp 61-70 1996.

58. J. E. Bobrow, S. Dubowsky and J. S. Gibson « Time-optimal control of robotic manipulators along specified paths » The international journal of robotics research, vol. 4 N° 3. 1985.
59. F. Pfeiffer, J. Rainer « A concept for manipulator trajectory planning » IEEE, Journal of robotics automation, Vol. Ra-3, NO. 2, April 1987.
60. Z. Shiller « Time energy optimal control of articulated systems with geometric path constraints » IEEE int. conf. on rob. and aut. Vol. 4 p. 2680-2685. May 1994.
61. I. Duleba « The method of minimal neighbourhood: a new and most effective iterative for minimum cost trajectory planning in robot manipulators » Robotica vol. 13 p. 297-304. 1995.
62. P. Tangpattanakul, A. Meesomboon, P. Artrit «Optimal Trajectory of Robot Manipulator Using Harmony Search Algorithms», SCI 270, pp. 23-36, Springer, 2010.
63. A. Gasparetto, V. Zanotto « A new method for smooth trajectory planning of robot manipulators» Mechanism and Machine Theory 42 (2007) 455-471.
64. A. Gasparetto, V. Zanotto « Optimal Trajectory Planning For Industrial Robots» Advances in Engineering Software 41 (2010) 548-556.3
65. T. Chettibi « Synthesis of dynamic motions for robotic manipulators with geometric path constraints» Mechatronics 16 (2006) 547-563.
66. T. Laliberté, C. M. Gosselin « Efficient algorithm for the trajectory planning of redundant manipulators with obstacles avoidance» IEEE International Conference on robotics and Automation, Vol3 pp2044-2049, May 1994.
67. C. W. Warren, J. C. Danos, B. W. Mooring « An approach to manipulator path planning» Int Jour of Robotics Reas, pp 87-95, 1989.
68. T. Laliberté «Planification de trajectoire d'un manipulateur serieel redondant dans un environnement encombré » Mémoire M.Sc, Université Laval, 1994.
69. M. Pasquier « Planification de trajectoires pour un robot manipulateur » Thèse de Doctorat, INP Grenoble, 1989.
70. M. Kirkanski, O. Timcenko « A geometric approach to manipulator path planning in 3d space in the presence of obstacles», Robotica, Vol 10 pp 321-328, 1992.
71. S. Yue, D. Henrich, W. L. Xu, S. K. Tso « Point-to-Point trajectory planning of flexible redundant robot manipulators using genetic algorithms», Robotica, Vol 20, pp 269-280, 2002.

72. G. Antonelli, C. Curatella, A. Marino « Constrained motion planning for open-chain industrial robots», *Robotica*, pp 1-18, 2010.
73. S. F. P. Saramago, M. Ceccarelli « Effect of basic numerical parameters on a path planning of robots taking into account actuating energy », *Mechanism and Machine* 39, 247-260, 2004.
74. R. Saravanan, S. Ramabalan, C. Balamurugan « Multiobjective trajectory planner for industrial robots with payload constraints », *Robotica*, Vol26, pp753-765, 2008.
75. S. F. P. Saramago, V. Steffen « Trajectory Modeling of Robot Manipulators in the Presence of Obstacles», *Jour of Opt The and App*, Vol 110, No1, pp 17-34, July 2001
76. A. Hentout, H. E. Lehtihet, T. Chettibi, B. Bouzouia « Roadmap-Based Collision-Free Trajectory Planning for Manipulator Robots», *Jour of Mode and Sim of Syst*, Vol 1, pp40-49, 2010.
77. R. R. D. Santos, V. Syeffen, S.F.P. Saramago « Robot path planning in a constrained workspace by using optimal control techniques », *Mult Syst Dyn*, Vol19, pp159-177, 2008.
78. R. Saravanan, S. Ramabalan « Evolutionary Minimum Cost Trajectory Planning for Industrial Robots», *J Intell Robot Syst*, Vol52, pp 45-77, 2008.
79. T. Chettibi, H.E. Lehtihet, M. Haddad, S. Hanchi « Minimum cost trajectory planning for industrial robots », *Eur Jour of Mech A/Solids* 23, 703-715, 2004.
80. A. D. Jutard-Malinge, G. Bessonnet «Optimal path planning of manipulatory systems subjected to non-autonomous motion laws», *Robotica*, Vol 15, pp251-261, 1997.
81. A. Dominique, J. Malinge, G. Bessonnet « Optimal Motion Planning of Robotic Manipulators Removing Mobile Objects Grasped in Motion», *Jour of Intell and Rob Syst* 29, pp 233-255, 2000.
82. Y. Chen, A. Desrochers « A proof of the structure of the minimum time control of robotic manipulators using Hamiltonian formulation », *IEEE Trans on Rob and Aut*. Vol 6 NO 3 pp 388-393 June 1990.
83. F. Dannes « Critère et contraintes pour la synthèse optimale des mouvements de robots manipulateurs. Application à l'évitement d'obstacles», *Thèse de Doctorat, Université de Poitier*, 1998.