

UNIVERSITE SAAD DAHLAB DE BLIDA1

Faculté de Technologie

Département de Mécanique

THESE DE DOCTORAT

en Génie Mécanique

Spécialité : Construction Mécanique

CONTRIBUTION À LA PLANIFICATION ET À L'OPTIMISATION MULTI-
OBJECTIFS DES TRAJECTOIRES POUR MANIPULATEURS ROBOTIQUES

Par

BENDALI Nadir

Devant le jury composé de :

Mustapha. TEMMAR	Professeur., U. Blida1	Président
Salam. ABUDURA	Professeur., U. Médéa	Examineur
El Hadj. AILAM	Professeur., U. Khemis-Miliana	Examineur
Abderrezak. ALLALI	Professeur., U. Blida1	Examineur
Ali. MAHIEDDINE	MCA., U. Khemis-Miliana	Examineur
Mohammed. OUALI	Professeur., U. Blida1	Directeur de thèse

Blida 2018

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

REMERCIEMENTS

Je tiens à remercier en premier lieu DIEU le tout puissant qui m'a donné la force et le courage pour achever ce travail.

Je tiens à remercier Le Professeur OUALI Mohammed et lui exprimer toute ma reconnaissance et ma profonde gratitude pour avoir encadré et dirigé ce travail. Je le remercie vivement, de ses conseils, ses orientations et le soutien que j'ai trouvé auprès de lui, et qu'il n'a jamais cessé de me témoigner durant de longues années.

J'exprime ma gratitude envers le docteur TEMMAR Mustapha, Professeur et Chef de Département de Mécanique à l'université Blida 1, de m'avoir fait l'honneur d'accepter de présider le jury.

J'adresse également mes vifs remerciements aux docteurs : ABUDURA Salam, Professeur à l'université de Médéa, El Hadj AILAM El Hadj, Professeur à l'université de Khemis Miliana, ALLALI Abderrezak, Professeur à l'université de Blida 1, MAHIEDDINE Ali, maître de conférence à l'université de Khemis Miliana, d'avoir accepté de juger ce travail et de faire partie du jury. Qu'ils trouvent ici, l'expression de toute ma gratitude.

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.

Je ne pourrais clôturer ces remerciements sans me tourner vers les êtres qui me sont les plus chers, qui ont eu un rôle essentiel pendant plusieurs années d'études, et qui sans eux aucune réussite n'aurait été possible.

ملخص

يهدف العمل الحالي إلى حل مشكلة التخطيط والمعايير المتعددة الأهداف لمسارات الأذرع الروبوتية، في تنفيذ المهام المفروضة والحررة، مع الأخذ بعين الاعتبار النموذج الديناميكي للروبوت والقيود منها الهندسية، الحركية والديناميكية. أولاً، اقترحنا نهجاً يقوم على تحديد شعاع موجه للفاصل الزمني لحل مشكلة المهام المفروضة، يتم إدخال هذا الشعاع مباشرة إلى الدوال الوضعية ومشتقاتها التي يتم تصميمها باستخدام الدوال التكميبيية. أما المهام الحرة فقد اعتمدنا على طريقة وضعت سابقاً للروبوتات المتنقلة، ومن ثم تم تطبيقها على الأذرع الروبوتية مع الأخذ بعين الاعتبار خصائص هذه الآليات الروبوتية، والخطوة الأولى هي توحيد مقياس الوقت. ثم تجزئ المسار إلى قسمين، والتي يتم تصميمها أيضاً باستخدام الدوال التكميبيية وباحترام خصائص كل قسم. يتم استخدام تقنية عشوائية من الخوارزميات الحينية لتحديد حل الوظائف المتعددة الأهداف المقترحة لكل مهمة. مختلف نتائج عديدة بينت جدوى المنهجيات المقترحة في هذا العمل.

كلمات مفتاحية: الأذرع الروبوتية، التخطيط المسارات، الدوال التكميبيية، الدوال متعددة الأهداف، الخوارزميات الحينية

RESUME

Le présent travail a pour but la résolution du problème de planification et l'optimisation multi-objectifs des trajectoires pour les manipulateurs robotiques, dans l'exécution des tâches imposées et libre. On considère le modèle dynamique du robot en tenant compte des contraintes géométriques, cinématiques et dynamiques. En premier lieu, on propose une approche basée sur la détermination d'un vecteur d'intervalle du temps pour résoudre le problème des tâches imposées. Ce vecteur est introduit directement dans la fonction des positions articulaires et de ses dérivées qui sont modélisées en utilisant les splines cubiques. Pour les tâches libres, on adopte une méthode déjà développée pour les robots mobiles, et on l'applique pour les robots manipulateurs en tenant compte des diverses propriétés des mécanismes robotisés, la démarche consiste en un premier temps à normaliser l'échelle de temps puis décomposer la trajectoire en deux fonctions. Ces dernières sont modélisées par les splines cubiques selon les conditions aux limites de chaque fonction. Une technique stochastique basée sur les algorithmes génétiques est employée afin de déterminer la solution optimale des fonctions multi-objectifs proposées pour chaque tâche. Divers résultats numériques illustrent la faisabilité des méthodologies proposées.

Mots clés : Manipulateurs Robotiques, Planification de Trajectoires, Splines Cubiques, Fonction Multi-Objectif, Algorithmes Génétiques.

ABSTRACT

The present work aims to solve the problem of planning and multi-objective optimization of trajectories for robotic manipulators, in the execution of imposed and free tasks. We consider the dynamic model of the robot and taking into account geometric, kinematic and dynamic constraints. First, an approach based on the determination of a time interval vector is proposed to solve the problem of imposed tasks. This vector is introduced directly into the function of the joints positions and its derivatives which are modeled using cubic splines function. For free tasks, we adopt a method already developed for mobile robots, and it has applied for robot manipulators considering the various properties of these mechanisms, the first step is to normalize the time scale then decompose the trajectory into two functions. These are modeled by cubic splines according to the boundary conditions of each function. A stochastic technique of genetic algorithms is used to determine the optimal solution of the proposed multi-objective functions for each task. Various numerical results illustrate the feasibility of the proposed methodologies.

Keywords: Manipulator Robots, Trajectories Planning, Cubic Spline, Stochastic Multi-Objective Functions, Genetic Algorithms.

TABLE DES MATIERES

REMERCIEMENTS.....	I
RESUME.....	II
TABLE DES MATIERES	IV
LISTE DES FIGURES.....	VII
LISTE DES TABLEAUX	IX
LISTE DES SYMBOLES ET ABREVIATIONS	X
INTRODUCTION GENERALE	1
Historique :.....	3
Contexte de la thèse :.....	6
Motivation :.....	7
Objectifs et Contribution :.....	8
Organisation du manuscrit :.....	9
Résultats publiés / Listes de publications :	10
1. GENERALITES SUR LES ROBOTS MANIPULATEURS	11
1.1. Introduction :.....	11
1.2. Coordonnées articulaires :.....	11
1.3. Coordonnées opérationnelles :.....	12
1.4. Redondance :.....	12
1.5. Configurations singulières :	12
1.6. Modélisation des bras manipulateurs :.....	13
1.6.1. Modélisation géométrique :.....	14
1.6.2. Modélisation cinématique :.....	19
1.6.3. Modélisation dynamique :	20
1.7. Prise en compte des frottements :.....	22
1.8. Les contraintes :.....	23
1.8.1. Contraintes liées à la tâche :.....	23
1.8.2. Contraintes sur les débattements :.....	23
1.8.3. Contraintes cinématiques :.....	24
1.8.4. Contraintes sur les couples moteurs :.....	24

1.8.5. Contraintes dues aux obstacles dans l'espace opérationnel :.....	25
1.9. Conclusion :.....	27
2. PLANIFICATION DES TRAJECTOIRES ET CRITERES DE PERFORMANCE	28
2.1. Introduction :.....	28
2.2. Planification des trajectoires :.....	28
2.2.1. Planification des trajectoires, des chemins et des mouvements :	29
2.2.2. Planification globale et planification locale :.....	30
2.3. Critères de performance dans la recherche de trajectoires :	31
2.3.1. Optimisation de la durée de parcours :	32
2.3.2. Optimisation du temps-efforts quadratiques/Energies :.....	35
2.3.3. Optimisation du temps-jerk :.....	38
2.3.4. Optimisation multi- objectifs :	40
2.4. Planification des trajectoires optimales dans les tâches imposées :	42
2.4.1. Modélisation de la trajectoire avec des splines cubiques :.....	42
2.4.2. Formulation de contraintes liées aux capacités limites du système : .	45
2.4.3. Initialisation du processus de génération des trajectoires :	45
2.4.4. Optimisation par algorithmes génétiques dans les tâches imposées :	47
2.5. Planification des trajectoires optimales dans les tâches libres :	49
2.5.1. Présentation de l'approche :.....	49
2.5.2. Optimisation par les algorithmes génétiques dans les tâches libres :	59
2.6. Conclusion :.....	60
3. RESULTATS ET ANALYSE.....	62
3.1. Introduction :.....	62
3.2. Trajectoires optimales dans les tâches imposées :	62
3.2.1. Minimisation du temps d'exécution :	62
3.2.2. Minimisation temps/jerk du Robot à 3 D.D.L :	71
3.2.3. Minimisation du temps/accélérations/jerk du Robot manipulateur de type PUMA 560 avec 6 D.D.L :	77
3.3. Trajectoires optimales dans les tâches libres :	84
3.3.1. Minimisation du temps/jerk du robot planaire à 3 D.D.L :.....	84
3.3.2. Minimisation du temps/couple/énergie du robot manipulateur Stanford à 6 D.D.L :	87

3.4. Conclusion :.....	92
CONCLUSION	94
ANNEXE	96
REFERENCES BIBLIOGRAPHIQUES	101

LISTE DES FIGURES

Figure 1 : Premier robot Unimates en fonderie sous pression en 1961 chez GM ...	4
Figure 2 : (a) Robot Stanford (1969), (b) ASEA Robot IRB-6 (1974), le premier tout-électrique & (c) PUMA500 (1978), le robot travaille en coopération avec des opérateurs.....	5
Figure 3: (a) Robot ABB IRB1400, (b) Robot SCARA EPSON LS3, (c) Robot Kuka KR 500 570 PA/1.....	5
Figure 1.1: Repérage utilisé en robotique	14
Figure 1.2: Paramétrage D.H d'une structure ouverte simple	16
Figure 2.1: Procédure standard pour la planification de trajectoires dans les robots industriels [22]......	29
Figure 2.2: Trajectoire du robot manipulateur modélisée par les Splines Cubiques	43
Figure 2.3: Représentation des nœuds intermédiaires et supplémentaires de la trajectoire modélisée par les fonctions Splines Cubique.	46
Figure 2.4: Organigramme des Algorithmes Génétiques dans les tâches imposées	48
Figure 2.5: Représentation du profil du chemin avec N_c+2 points de contrôle	51
Figure 2.6: Représentation du profil de mouvement avec N_m+2 points de contrôle	52
Figure 2.7: Organigramme des Algorithmes Génétiques utilisé dans les tâches libres	60
Figure 3.1: Représentation du Robot SCARA 2R planaire.....	64
Figure 3.2: Résultats obtenues du robot SCARA 2R	66
Figure 3.3: Trajectoire optimale du robot Scara 2R dans l'espace Cartésien	67
Figure 3.4: Résultats des générations durant le processus d'optimisation du temps d'exécution de la tâche du robot planaire 2R	67
Figure 3.5: Résultats du robot manipulateur 6R sous des contraintes cinématiques	69

Figure 3.6: Trajectoire optimale du robot manipulateur 6R sous les contraintes cinématiques: (a) entre le nœud 1 et 3, (b) entre le nœud 1 et 4, (c) entre les nœuds 1 et 6	70
Figure 3.7: Résultats des générations durant le processus d'optimisation du temps d'exécution de la tâche du robot manipulateur 6R	71
Figure 3.8: Résultats du Robot 3R, selon la tâche assignée pour (a) $K_J / K_T = 0.02$ et (b) pour $K_J / K_T = 0.3$	74
Figure 3.9: Résultats historique du processus d'optimisation de la fonction objectif pour le robot manipulateur 3R pour (a): $K_J / K_T = 0.02$ et (b): $K_J / K_T = 0.3$	75
Figure 3.10: Allure du mouvement optimisé du robot industriel à 3R.	76
Figure 3.11: Trajectoires optimales des positions articulaires pour le robot PUMA560- 6R.....	79
Figure 3.12: Vitesses, accélérations et jerks optimaux pour le robot PUMA 560-6R pour ($\alpha_1=0.8$, $\alpha_2=0.1$ et $\alpha_3=0.1$)	80
Figure 3.13: Vitesses, accélérations et jerks optimaux pour le robot PUMA 560-6R pour ($\alpha_1=0.1$, $\alpha_2=0.8$ et $\alpha_3=0.1$)	81
Figure 3.14: Résultats historique du processus d'optimisation de la fonction objectif pour le robot manipulateur PUMA560-6R.....	84
Figure 3.15: Résultats des trajectoires optimisées pour le robot planaire 3R pour $\alpha = 0.5$ nous avons trouvé $T = 4.52s$ et $F_{obj} = 2.26$	85
Figure 3.16: Aspects optimal du mouvement pour le robot planaire 3R avec $\alpha=0.5$	86
Figure 3.17: Robot manipulateur Stanford [15]	87
Figure 3.18: Trajectoires optimales du robot Stanford avec contraintes kinodynamiques pour ($\alpha =0.75$, $\beta =0.25$).....	89
Figure 3.19: Trajectoires optimales du robot Stanford avec contraintes dynamiques pour ($\alpha =0.5$, $\beta =0.5$)	90
Figure 3.20: Mouvement Optimal du robot Stanford pour ($\alpha =0.5$, $\beta =0.5$).....	92

LISTE DES TABLEAUX

Tableau 2.1: Intervalles admissibles de T liés aux contraintes couples du système	57
Tableau 3.1: Paramètres géométriques, inertiels et contraintes dynamique du robot 2R	64
Tableau 3.2: Tâche imposée au robot 2R à travers des nœuds en coordonnées articulaires en [Rad] et ses équivalences en coordonnées opérationnelles en [m].....	65
Tableau 3.3: Positions des nœuds (points de passage) pour chaque articulation du robot 6R	68
Tableau 3.4: Limites technologiques du robot 6R PUMA 560.....	68
Tableau 3.5: Positions des nœuds (points de passage) pour chaque articulation du robot 3R	72
Tableau 3.6: Limites technologiques du robot 3R	72
Tableau 3.7: Résultats des intervalles de temps optimaux et de jerks maximaux comparés à ceux de [83].....	76
Tableau 3.8:Nœuds de passage pour chaque articulation du robot PMUA560 [14]	77
Tableau 3.9: Résultats d'optimisation pour le robot PUMA560-6R	82
Tableau 3.10: Résultats des paramètres cinématiques trouvés après le processus d'optimisation pour ($\alpha_1=0.8$, $\alpha_2=0.1$ et $\alpha_3=0.1$)	83
Tableau 3.11: Paramètres géométriques et inertiels du robot planaire 3R	84
Tableau 3.12: Capacités technologiques du robot planaire 3R	85
Tableau 3.13: Résultats des valeurs maximales des contraintes kinodynamiques obtenues par notre procédure d'optimisation	86
Tableau 3.14: Tâche assignée par le robot Stanford [70]	87
Tableau 3.15: Limites technologiques du robot Stanford [70]	88
Tableau 3.16: Résultats numériques du robot Stanford 6 D.D.L	91

LISTE DES SYMBOLES ET ABREVIATIONS

E_c : Energie cinétique totale.

E_p : Energie potentielle totale.

F_s : Vecteur des coefficients de frottement sec.

F_v : Vecteur des coefficients de frottement visqueux.

M : Nombre des intervalles de temps.

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 $\lambda(\xi)$.

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

P_i : Position initiale.

P_f : Position finale.

PMP : Principe de Maximum de Pontriaguine.

$Q(t), V(t), W(t)$ et $J(t)$: Positions, vitesses, accélérations et jerks articulaires pour les tâches imposées.

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

SQP : Programmation Quadratique Séquentielle.

T : Le temps final de transfert.

${}^i T_{i-1}$: La matrice de transformation homogène du repère R_{i-1} au repère R_i .

X, Y, Z : Coordonnées opérationnel du robot.

h_i : Vecteur d'intervalle du temps.

k_T, k_J : Facteurs de pondération du temps et du jerk dans les tâches imposées.

q^{init} : Configuration initiale.

q^{fin} : Configuration finale.

$q(t), \dot{q}(t), \ddot{q}(t)$ et $\ddot{\ddot{q}}(t)$: Positions, vitesses, accélérations et jerks articulaires pour les tâches libres.

α, β : Facteurs de pondération.

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

τ : Vecteur des couples moteurs (les commandes).

INTRODUCTION GENERALE

Dans le monde industriel d'aujourd'hui, le but du manipulateur robotique consiste à développer la qualité des produits, à améliorer la productivité, la précision, la rapidité et la flexibilité. Les exigences des applications industrielles sont très complexes et difficiles à réaliser. Les robots peuvent être employés dans des environnements hostiles et effectuer des tâches que les humains ne peuvent pas ou ne veulent pas faire (Tâches dangereuses, ennuyeuses ou répétitives continues). La recherche d'une fabrication plus efficace et d'un travail de meilleure qualité dans l'industrie est une considération industrielle essentielle, qui a conduit à l'amélioration des systèmes de manipulation robotique moderne et détrempant. Cependant, ces habiles manipulateurs robotiques sont peu autonomes et nécessitent des actions initiales (telles que l'étalonnage, la planification des trajectoires de manipulateurs) afin de répondre aux exigences de certaines tâches. Par conséquent, il est extrêmement intéressant de contrôler correctement leur action dans l'environnement de travail pour exécuter avec succès la tâche donnée. La capacité d'exécuter des missions précises et de suivre une trajectoire de mouvement requise est au cœur de la recherche robotique [1]. Les principaux domaines de recherche du domaine robotique peuvent être résumés comme suit :

- Conception de manipulateur robotique.
- Planification de la trajectoire.
- Intelligence artificielle.

Parmi ces domaines de recherche scientifique, la question de la planification de la trajectoire occupe une place cruciale dans le domaine de la robotique. Lorsque le robot doit être déplacé de la position initiale à la position finale, les problèmes de planification de chemin et de trajectoire sont abordés et pris en

compte par de nombreux chercheurs. Les problèmes de planification de chemin et de la trajectoire sont considérés comme deux parties de recherches distinctes du domaine de la robotique pour lequel de nombreuses recherches sont dédiées. Il existe une nette différence entre la planification du chemin et de la trajectoire pour les systèmes robotiques. La planification des chemins envisage d'obtenir un chemin de configuration du robot entre une position initiale et une position finale qui fournissent une trajectoire sans collision. Attendu que la planification optimale de la trajectoire peut être identifiée comme la procédure de choix d'un mouvement et des contrôles optimaux associés à partir de l'ensemble des mouvements et contrôles admissibles tout en satisfaisant toutes les contraintes cinématiques et dynamiques et en minimisant une fonction de coût requise telle que le temps requis et la consommation d'énergie [2]. Par conséquent, les problèmes de planification de chemin sont un sous-ensemble des problèmes de planification de la trajectoire. La planification optimale de la trajectoire est extrêmement vitale pour le bon fonctionnement d'une application particulière.

Dans de nombreuses tâches d'automatisation industrielle, un grand nombre de manipulateurs robotiques sont utilisés dans la ligne de production. Ces robots effectuent des tâches telles que le soudage, les opérations de sélection et de mise en place et bien d'autres tâches. L'industrie du manipulateur robotique a connu une croissance rapide, ce qui améliore la productivité ainsi que le nombre de manipulateurs robotiques utilisés chaque année. En général, les manipulateurs robotiques industriels sont conduits électriquement et exécutent leurs travaux de façon continue tous les jours. Dans le secteur industriel, les applications industrielles ont des processus répétitifs pour un travail prévu et une grande quantité d'énergie est demandée lors de l'application donnée [3]. Si une installation robotique comporte plusieurs lignes d'installation ou plusieurs installations, la consommation totale d'énergie augmentera considérablement au cours des processus donnés.

Afin d'augmenter la productivité dans les processus requis, le temps d'exécution minimal est largement utilisé comme critère d'optimisation de la trajectoire afin de diminuer le temps de production et de maximiser la rentabilité [4]. Bien que le facteur de temps d'exécution minimum soit un critère inhérent d'optimisation de la trajectoire et une considération industrielle importante, il n'est pas commode de

considérer une trajectoire lisse pour le mouvement robotique. Ce type de trajectoire robotique peut entraîner des chocs et des vibrations indésirables, une consommation d'énergie accrue et une perte de précision pour un chemin donné. En outre, tous les actionneurs du manipulateur fonctionnent à leur rendement maximal. Cela a un impact sur les organes (engrenages) du robot, les actionneurs et la structure du manipulateur [4]. Alors que générer la trajectoire du robot en prenant en compte les critères énergétiques fournit des avantages pour les applications prévues. Il peut fournir des performances de suivi en douceur, moins de vibrations sur les actionneurs robotiques et ne fluctue pas autant que dans le mouvement optimal.

Ce travail présente différentes stratégies pour déterminer la solution optimale de la trajectoire des robots manipulateurs qui exécutent des mouvements libres ou des tâches imposées, en optimisant les indices de performance tels que le temps, le couple de l'actionneur, l'énergie, l'accélération et le jerk. La méthode développée prend en compte à la fois la cinématique du mécanisme et les conditions de contraintes dynamiques. Divers résultats numériques illustrent la faisabilité des méthodologies proposées.

Historique :

Le mot robot dérive du terme Tchèque *Robota* qui signifie « travail compulsif ». Le mot robot apparut pour la première fois en 1921 dans la pièce de théâtre « R.U.R. » (*Rossum's Universal Robots*) mis en scène par le dramaturge tchèque *Karel Capek*. La pièce décrivait des robots humanoïdes détruisant leurs créateurs humains, thème repris depuis dans la science-fiction moderne.

Il est certain que depuis fort longtemps, les réalisateurs d'automates ont cherché à pouvoir insuffler à leurs machines des comportements adaptés aux circonstances. Malheureusement, jusqu'au vingtième siècle, les techniques étaient trop primitives et simplistes pour permettre de telles réalisations. Il a fallu attendre les fins des années cinquante pour que 'George Devol' invente le premier robot industriel qui a rejoint les lignes d'assemblage de General Motors. Depuis lors, la technologie robotique a commencé à fournir de nombreuses applications à l'industrie :

fabrication de circuits intégrés, voitures, nourriture et d'autres lignes automatisées qui collectent, traitent, mouvements et produits d'emballage.

La première compagnie de fabrication de robot, *Unimation Inc.*, fut créée aux états unis par *Joseph Engelberger* en 1956. Le premier robot fut mis en application dans l'usine *Ewing Township* de General Motors en 1961 pour extraire des pièces d'une cellule de fonderie sous pression (Figure 1). La plupart des robots hydrauliques *Unimates* vendue les années suivantes trouvèrent des applications dans la manutention de pièces et le soudage de caisses de véhicules.



Figure 1 : Premier robot *Unimates* en fonderie sous pression en 1961 chez GM

Plus tard, Le Stanford Arm a été conçu en 1969 par *Victor Scheinman* comme un prototype pour la recherche. La conception de robots a été par la suite fortement influencée par les concepts de *Scheinman*. En 1974, la société ASEA (ABB aujourd'hui) a présenté le premier robot commandé par un microprocesseur, l'IRB-6, ce qui a permis le mouvement en trajectoire continue, une condition préalable pour le soudage à l'arc ou l'usinage.

À l'origine, les robots étaient prévus pour remplacer les ouvriers humains. C'était particulièrement dans l'industrie automobile qui souffrait des divers problèmes. Alors, on a pensé que les robots seraient des machines universelles, capables d'être rapidement reprogrammées pour une grande variété de tâches. Ces idées ont motivé le développement des robots PUMA (*Programmable Universal Machine for Assembly*) en 1978 dans un projet de recherche pour General Motors. Le robot a été étroitement modelé sur le bras humain et pouvait soulever un poids à peu près identique (figure 2).



Figure 2 : (a) Robot Stanford (1969), (b) ASEA Robot IRB-6 (1974), le premier tout-électrique & (c) PUMA500 (1978), le robot travaille en coopération avec des opérateurs

Actuellement, nous trouvons des robots manipulateurs industriels lourds, ABB IRB 1400, SCARA (*Selective Compliance Assembly Robot Arm*), EPSON LS3 et Kuka 500 570 PA/1 respectivement, qui sont très sophistiqués et utilisés pour les fabrications qui demandent des efforts importants (figure 3).



Figure 3: (a) Robot ABB IRB1400, (b) Robot SCARA EPSON LS3, (c) Robot Kuka KR 500 570 PA/1

Selon l'étude robotique de la Fédération Internationale de Robotique (IFR) en 2017, il y a au moins 1 946 000 robots industriels opérationnels fin 2016 dans le monde.

Grâce aux avancées des technologies, la robotisation des fabrications industrielles s'est élargie ces dernières années. Toujours d'après (IFR, 2017), de 2017 à 2019 les installations de robots devraient augmenter d'au moins 13% en moyenne par année : 8% dans les Amériques et en Europe et 15% en Asie / Australie. Les ventes mondiales totales atteindront environ 413 000 unités en 2019. Entre 2016 et 2019, on estime que plus de 1,4 million de nouveaux robots industriels seront installés dans des usines à travers le monde.

Contexte de la thèse :

La compréhension de la complexité des robots et de leurs applications requiert la connaissance de diverses branches de l'ingénierie, de l'informatique et des mathématiques.

De tous les systèmes mécaniques robotisés, les manipulateurs en série ont mérité une attention particulière en raison de leur grande pertinence dans l'industrie. D'ailleurs, ils constituent le plus simple de tous les systèmes mécaniques robotiques et, par conséquent, apparaissent comme éléments d'autres systèmes plus complexes.

Un manipulateur, en général, est un dispositif qui aide l'homme à effectuer des tâches de manipulation générale. Sa structure se compose d'un ensemble de liens disposés en série et reliés par des articulations. Ces articulations peuvent être de type rotoïde (rotations) ou prismatiques (translations). Les actionneurs, généralement des moteurs, sont responsables du changement des angles et/ou de la longueur des articulations.

Les travaux de cette thèse se focalisent sur la contribution à la planification des trajectoires optimales pour des robots manipulateurs. Ils abordent deux catégories de trajectoires vis-à-vis des tâches assignées à l'organe terminal : l'une libre appelée aussi mouvement point à point tels que les opérations de la manutention d'objets et l'autre imposée, par exemple, les travaux de découpage ou de soudure en continu.

Ce problème de planification de trajectoires optimales est un problème complexe d'optimisation non linéaire sous contraintes. Dans lequel le but est de minimiser une quantité physique représentée par une fonction objectif 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.

La trajectoire est chargée de spécifier une séquence de mouvements nécessaires au robot pour exécuter une certaine tâche.

En raison de la possibilité d'obtenir plusieurs trajectoires pour l'exécution d'une certaine tâche, une question qui se pose naturellement lors de la programmation du robot se rapporte à l'obtention de la meilleure trajectoire.

Il n'existe pas de solution définitive, puisque la réponse à cette question dépend directement de la fonction objectif qui doit être améliorée. Un aspect qui peut être pris en compte par une analyse minutieuse est la détermination d'une trajectoire sans obstacle. Dans ce contexte, la meilleure trajectoire est celle qui correspond à la fonction objectif minimale qui vérifie efficacement toutes les contraintes imposées au robot manipulateur quel que soit la tâche assignée libre ou bien imposée.

Motivation :

Afin de résoudre les problèmes d'optimisation de trajectoire, divers algorithmes d'optimisation de trajectoire ont été développés et mis en œuvre par de nombreux chercheurs. Toutes ces procédures d'optimisation de trajectoire développées ont leurs propres forces, faiblesses et caractéristiques [1].

Idéalement, afin de déterminer la solution optimale avec succès, des procédures d'optimisation efficaces du point de vue calcul sont préférées pour une tâche donnée. En effet, les algorithmes d'optimisation de trajectoire utilisés peuvent être un problème extrêmement complexe et difficile à résoudre en raison du grand nombre de paramètres d'optimisation et de diverses contraintes qui doivent être traités efficacement pendant le processus d'optimisation [5].

Le processus d'optimisation est constitué des calculs de la fonction objectif qui impliquent l'exécution du modèle dynamique inverse à calculs intensifs du robot manipulateur en considérant d'une part le nombre de degrés de liberté et d'autre part les points de passage (nœuds) assignés à l'organe terminal à savoir des mouvements libres ou imposés. Dans ce cas, la procédure d'optimisation sera un problème chronophage pour une trajectoire donnée afin de déterminer les solutions optimales réalisables.

Par conséquent, pour le succès de toute approche pour la résolution des problèmes de planification de trajectoires multi-objectifs dans les tâches imposées et libres, l'algorithme d'optimisation de trajectoire doit être facilement utilisé sur les différentes morphologies des robots manipulateurs ainsi que sur les différents types de contraintes (géométriques, cinématiques et dynamiques) qui doivent être traités efficacement pendant la procédure d'optimisation de trajectoire, ceci représente un facteur de motivation crucial et un défi intéressant pour cette thèse.

Objectifs et Contribution :

Cette thèse contribue au domaine de la planification et de l'optimisation multi-objectifs des trajectoires pour les manipulateurs robotiques en considérant les deux tâches, imposée et libre assignées à l'organe terminal.

Le but de ce travail se résume comme suit :

- Proposer une méthode efficace pour traiter le problème de planification de trajectoires optimales dans les tâches imposées afin de minimiser des paramètres significatifs à l'exploitation des manipulateurs robotiques tels que le temps, l'accélération, le jerk, le couple moteur et la consommation d'énergie qui sera liée à l'ampleur des couples d'articulation requis pendant le mouvement désiré.
- Exploiter l'approche de normalisation de l'échelle du temps pour transformer le temps final inconnu à un paramètre d'optimisation dépendant afin d'obtenir une nouvelle formulation de problème dans les tâches libres.
- Introduire une méthode originale basée sur la normalisation de l'échelle de temps pour traiter la fonction objectif représentée par le temps, le couple, l'énergie et le jerk dans les tâches libres.

- Exploiter les caractéristiques de base de l'interpolation par des polynômes de type splines cubiques pour améliorer l'efficacité des algorithmes de solution adoptés.
- Utiliser une technique d'optimisation stochastique qui sera souple et efficace afin d'obtenir de meilleurs résultats plausibles, dans des temps de calculs raisonnables, qui peut traiter divers cas de problèmes de planification de trajectoires pour des robots manipulateurs plans ou spatiaux, redondants ou non, avec ou sans évitement d'obstacles.

Pour atteindre ces objectifs, une étude de simulation théorique sera d'abord réalisée pour tester et examiner l'efficacité des éventuelles techniques en utilisant des manipulateurs robotiques comme le robot Scara 2R, le robot planaire à 3 d.d.l, le robot 6R PUMA 560 et le robot manipulateur Stanford.

Organisation du manuscrit :

Ce document est organisé de la façon suivante :

- Le chapitre 1 fournit une introduction aux manipulateurs robotiques et aux problèmes des techniques de planification de la trajectoire. En outre, le but et les objectifs de la recherche et la nouveauté du travail sont donnés.
- Dans le chapitre 2, nous présentons les rappels théoriques, les notions de base, les concepts essentiels de la cinématique, de la cinématique différentielle et de la dynamique, appliqués en particulier aux robots. Ces concepts permettent la définition de modèles représentatifs pour l'analyse et l'optimisation des critères de performance de chaque manipulateurs robotiques.
- Le chapitre 3, présente les approches de résolution proposées pour les tâches imposées et les tâches point à point et leurs critères de performances ainsi que les formes de trajectoires et la technique d'optimisation choisies.
- Le chapitre 4, regroupe les principales applications numériques menées selon l'approche proposée et apparues dans diverses publications et communications en partant du mouvement imposé jusqu'au mouvement libre. Ces applications ont été faites sur plusieurs manipulateurs robotiques planaires et spatiaux qui subissent différentes contraintes, et en optimisant différentes fonctions objectifs.

- Enfin, une conclusion générale reflète fidèlement les résultats des chapitres suscités avec des analyses et des commentaires, suivi par les perspectives envisagées.

Il convient de noter que toutes les méthodologies proposées tout au long de ce travail peuvent être appliquées à des robots manipulateurs avec diverses configurations cinématiques, c'est-à-dire que des hypothèses restrictives sur des caractéristiques spécifiques d'un manipulateur donné ne sont pas prises en considération pendant le développement de la méthodologie et de son application. En conséquence, lors de la réalisation des diverses applications numériques, nous avons pu travailler avec divers types de robots, avec différentes configurations cinématiques possibles.

Finalement, l'analyse des différentes stratégies d'optimisation a nécessité la mise en œuvre de plusieurs algorithmes.

Pour l'information et la commodité du lecteur, une description des fonctions splines cubiques est donnée en Annexe.

Résultats publiés / Listes de publications :

Une partie du travail effectué dans cette thèse a fait l'objet de plusieurs publications :

- L'approche qui permet d'obtenir la trajectoire optimale dans les tâches imposées proposée dans la section 4.2.1 a été publiée dans [40].
- Les travaux relatifs à la minimisation de la fonction objectif temps et jerk, développés dans la section 4.2.2, ont été principalement publiés dans [61].
- La fonction objectif pour les tâches point à point développée dans la section 4.3.1 a été initialement publiée dans [80].

1. GENERALITES SUR LES ROBOTS MANIPULATEURS

1.1. Introduction :

Ce chapitre consiste à représenter le comportement du robot par des équations algébriques, nous faisons appel à certaines définitions et des notions mathématiques du point de vue position (*Modèle géométrique*) ou vitesse (*modèle cinématique*) ou efforts mis en jeu (*modèle dynamique*). La prise en compte de la dynamique du robot nécessite une formulation du problème basée sur la modélisation dynamique de son comportement. Ce type de modélisation permet d'avoir des mouvements plus compatibles avec la structure polyarticulée. Cependant, il faut aussi tenir compte des contraintes propres au robot, à la tâche et à l'environnement. Ces contraintes sont, à leurs tours, traduites sous forme de conditions et de restrictions sur les variables articulaires et sur les commandes.

1.2. Coordonnées articulaires :

Les coordonnées articulaires appelées aussi coordonnées généralisées sont les variables (angles 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_n, \dots). On choisit généralement pour variables q celles qui sont directement commandées par les actionneurs (rotatif ou translation).

1.3. 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 Dr . Dans le cas d'une tâche de positionnement d'un objet, $Dr \leq 6$.

1.4. 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

1.5. Configurations 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. [6]

Les mouvements d'un manipulateur sont observés dans l'espace cartésien. Ces mouvements sont donnés au manipulateur par les actionneurs, que l'on décrit dans l'espace articulaire. Quand les actionneurs ont des vitesses, on doit pouvoir connaître les vitesses que cela impose à l'effecteur dans toutes les directions. La correspondance entre ces deux mesures doit donc être établie, en vue d'imposer aux actionneurs les vitesses articulaires requises pour amener l'effecteur à se déplacer de la façon prescrite dans l'espace cartésien. [7]

Cette relation prend la forme suivante :

$$A X = B Q$$

Où A et B sont les matrices Jacobienne du manipulateur

Dans certains cas, selon la position du manipulateur, cette relation n'amène plus de solution unique ou dégénère, c'est-à-dire qu'une entrée nulle amène une sortie non nulle ou vice-versa. Ces positions sont appelées les configurations singulières ou singularités du manipulateur. Les lieux de singularité regroupent toutes les configurations singulières d'un mécanisme.

Les singularités des manipulateurs parallèles se présentent sous trois formes distinctes. Les types rencontrés sont décrits par *Gosselin* et *Angeles* (1990). Essentiellement, les singularités sont classées de la manière suivante :

- Type I : La matrice B est singulière. Ce type de singularité correspond généralement aux limites de l'espace atteignable. Dans cette situation, les vitesses cartésiennes peuvent être nulles pour des vitesses articulaires non nulles.
- Type II : La matrice A est singulière. Dans ce cas, des vitesses cartésiennes non nulles peuvent être observées même si les actionneurs présentent des vitesses articulaires nulles.
- Type III : Ce type de singularité peut se produire uniquement dans des cas particuliers, quand le manipulateur possède une architecture particulière.

1.6. 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 dans ce manuscrit.

1.6.1. 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 1.1 on a représenté les principaux repères utilisés en robotique.

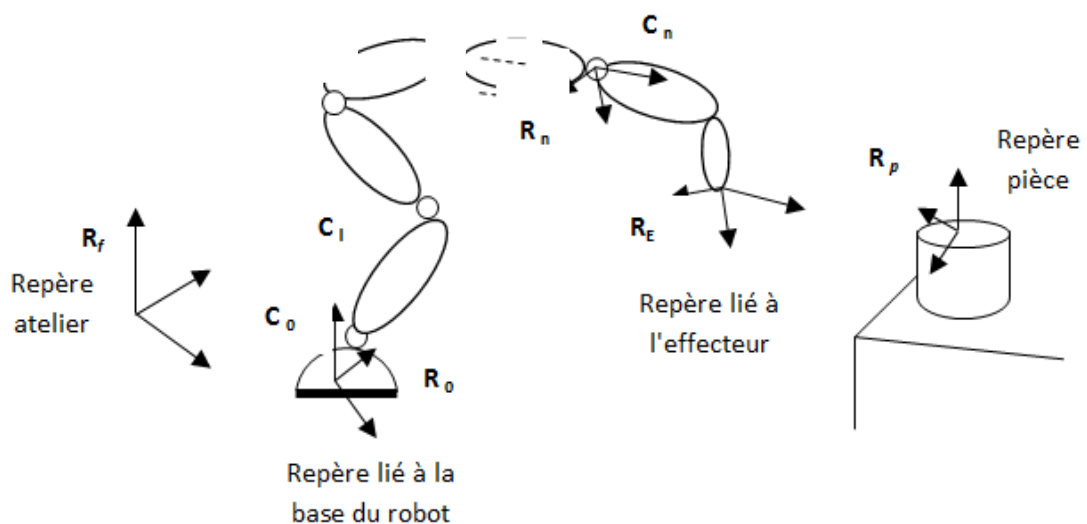


Figure 1.1: Repérage utilisé en robotique

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 (tâche), 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 .

1.6.1.1. Paramétrage des robots manipulateurs :

Plusieurs méthodes de description systématique et notations ont été proposées : le paramétrage des mouvements le plus répandu est celui de *J. Denavit* et *R. S. Hartenberg* [8] modifié plus tard par *W. Khalil* et *J.F. Kleifinger* [9]. Ce paramétrage (figure 1.2) présente l'avantage d'être défini au moyen d'une construction géométrique dont la structure demeure identique quelle que soit la liaison simple, pivot ou glissière. Il permet ainsi l'écriture de procédures itératives qui se prêtent à une détermination automatique du modèle dynamique. Le paramétrage de la structure consiste à décrire, avec un nombre fini de paramètres, la morphologie et la cinématique de celui-ci. Le bon choix de ces paramètres permet une formulation systématique et peu redondante des différents modèles élaborés. La méthode de description repose sur les hypothèses suivantes :

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 1.2).

- 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 .

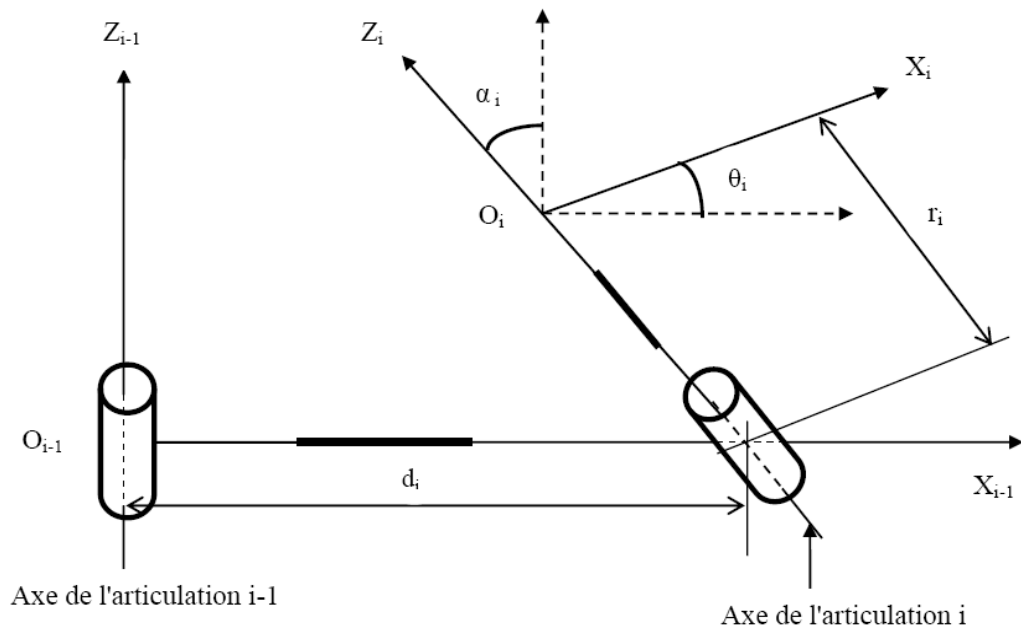


Figure 1.2: Paramétrage D.H d'une structure ouverte simple

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

$$q_i = \begin{cases} \theta_i & A_i \text{ est rotoïde} \\ r_i & A_i \text{ est prismatique} \end{cases}$$

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

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

$$\sigma_i = \begin{cases} 1 & \text{si } A_i \text{ est prismatique} \\ 0 & \text{si } A_i \text{ est rotoïde} \end{cases}$$

En définitive et en utilisant tous les résultats précédents :

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

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

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

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

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

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

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

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

1.6.1.2. Modèle géométrique :

C'est l'ensemble des relations qui permettent d'exprimer la situation de l'organe terminal en fonction des coordonnées articulaires. Ces n variables articulaires sont définies par la matrice unicolonne $q = [q_1, q_2, \dots, q_n]^T$ appartenant à un espace dit articulaire.

La situation de l'organe terminal est définie par la matrice unicolonne formée de m coordonnées opérationnelles telle que : $X = [X_1, X_2, \dots, X_m]^T$ avec $m \leq 6$ cette situation est définie dans un espace dit opérationnel.

Le modèle géométrique direct MGD 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.

Le modèle géométrique inverse *MGI* 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, trois 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 [10] [11]. (Méthode de *PIEPER*, méthode de *PAUL*, etc....).

Devant une tâche définie généralement, pour un emplacement donné du robot par des points de passage ou une trajectoire à suivre, la multiplicité des solutions du *MGI* offre plusieurs possibilités pour la réalisation de cette tâche. Il serait intéressant de profiter de ce choix et prendre celui qui minimise une fonction objectif, synthétisé de façon à accroître le rendement et la fiabilité du robot manipulateur.

Sur le plan pratique, le *MGI* présente un intérêt indéniable aussi bien pour les robots programmés par apprentissage que par langage de haut niveau, il permet

de transcrire les besoins de positionnement et d'orientation de l'effecteur face à une tâche donnée, dans le système de commande.

1.6.2. Modélisation cinématique :

La modélisation cinématique, qui fait-elle aussi abstraction des causes qui ont donné naissance au mouvement, complète la modélisation géométrique en donnant les relations entre les vitesses opérationnelles et les vitesses articulaires. En effet, le modèle cinématique direct *MCD* permet de décrire les variations élémentaires des coordonnées opérationnelles \dot{X} en fonction des variations élémentaires des coordonnées articulaires \dot{q} . La relation de base est la suivante :

$$\dot{X} = J(q)\dot{q} \quad (1.07)$$

$J(q)$ Désigne la matrice jacobienne du robot de dimension $m * n$, elle dépend seulement des paramètres géométriques du mécanisme ; la matrice jacobienne est obtenue selon différentes méthodes, celle qui semble la plus naturelle a priori et qui est utilisée dans nos travaux consiste à réduire des éléments (i, j) à partir du *MGD* selon la relation (1.05):

$$J_{ij} = \frac{\partial f_i(q)}{\partial q_j} = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \frac{\partial f_1}{\partial q_2} & \frac{\partial f_1}{\partial q_3} & \dots & \frac{\partial f_1}{\partial q_n} \\ \frac{\partial f_2}{\partial q_1} & \frac{\partial f_2}{\partial q_2} & \frac{\partial f_2}{\partial q_3} & \dots & \frac{\partial f_2}{\partial q_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial q_1} & \frac{\partial f_n}{\partial q_2} & \frac{\partial f_n}{\partial q_3} & \dots & \frac{\partial f_n}{\partial q_n} \end{bmatrix} \quad i = (1, \dots, m), j = (1, \dots, n)$$

La matrice jacobienne est d'une très grande utilité. En statique, elle facilite le calcul des singularités et la dimension du volume de travail du robot manipulateur, d'autre part elle permet d'établir le lien entre les efforts exercés sur l'organe terminal et les efforts actionneurs.

A partir de la relation (1.07) on peut définir ce qu'on appelle le modèle cinématique du second ordre :

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

Avec :

$$j = \frac{d}{dq} J(q)$$

Ce modèle est souvent utilisé pour la minimisation des couples moteurs des robots redondants et également pour exprimer le modèle dynamique dans l'espace cartésien.

Le modèle cinématique inverse *MCI* a les mêmes objectifs que ceux du modèle géométrique inverse sur le plan des vitesses de l'effecteur. C'est à partir d'une configuration q , qu'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 :

$$[\dot{q}] = [J_{ij}]^{-1} [\dot{X}] \quad (1.09)$$

Pour obtenir le modèle différentiel inverse, on peut soit dériver le modèle géométrique inverse par méthode analytique de *PAUL* soit inverser le modèle différentiel direct d'une manière plus générale. Ce modèle présente un intérêt indéniable lors de l'exploitation, surtout lorsque le robot est appelé à travailler en continu sur des trajectoires fixes avec des contraintes sur les vitesses de déplacement, c'est le cas par exemple des tâches de soudure en continu, où l'épaisseur du cordon de soudure est tributaire de la régularité de la vitesse de déplacement de la torche du robot relativement à la pièce.

1.6.3. Modélisation dynamique :

Le modèle dynamique exprime une équation mathématique qui donne la relation entre les couples et/ou les moments appliqués aux actionneurs en fonction des variables articulaires (q, \dot{q}, \ddot{q}) , des paramètres géométriques et d'inertie du robot, et des efforts extérieurs F appliqués sur les différents corps constituant le robot. On le présente par une relation de la forme :

$$\tau = g(q, \dot{q}, \ddot{q}, F) \quad (1.10)$$

Conventionnellement, on dira que la relation (1.10) représente le modèle dynamique inverse *MDI* ou tout simplement le modèle dynamique. Pour les

besoins des commandes seules les composantes du torseur τ (Couples ou efforts) selon les axes de mouvement des articulations sont exploitées.

Le modèle dynamique direct *MDD* est celui qui exprime les accélérations articulaires en fonction des positions, des vitesses articulaires, et des efforts actionneurs. Il est de la forme :

$$\ddot{q} = h(q, \dot{q}, \tau, F) \quad (1.11)$$

Ce modèle est généralement utilisé juste pour les simulations.

Plusieurs méthodes sont utilisées pour l'obtention du modèle dynamique, les plus connus sont :

- La méthode du lagrangien.
- La méthode Newton-Euler.
- La methode de Kane.
- La méthode d'Alambert's.

Les équations de mouvement issues du modèle de Lagrange s'écrivent, en absence d'efforts extérieurs et de frottement comme suit :

$$\tau_i = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i}; \quad i = (1, \dots, n) \quad (1.12)$$

Où :

τ_i : Couple moteur de la $i^{\text{ème}}$ articulation.

L : Lagrangien du système égale à $E_c - E_p$.

E_c : Energie cinétique totale.

E_p : Energie potentielle totale.

n : nombre total des articulations.

Les équations de mouvement (1.12) se présentent sous forme matricielle de la façon suivante :

$$\tau_i = \sum_{j=1}^n M_{ij}(q)\ddot{q}_j + C_i(q, \dot{q}) + G_i(q) \quad i = 1, \dots, n \quad (1.13)$$

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.

1.7. 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 [12] :

$$\tau_{fi} = F_{si} \text{sign}(\dot{q}_i) + F_{vi}\dot{q}_i \quad i = 1, \dots, n \quad (1.14)$$

Dans cette expression le couple de frottement de l'articulation i est noté τ_{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 :

$$\tau_i = \sum_{j=1}^n M_{ij}(q)\ddot{q}_j + C_i(q, \dot{q}) + G_i(q) + \text{diag}(\text{sign}(\dot{q}_i)) F_{si} + F_{vi} \text{diag}(\dot{q}_i) \quad (1.15)$$

Où :

- $F_v = [F_{v1}, F_{v2}, \dots, F_{vi}]^T$ est le vecteur des coefficients de frottement visqueux.
- $F_c = [F_{c1}, F_{c2}, \dots, F_{ci}]^T$ est le vecteur des coefficients de frottement sec.
- $\text{diag}(\dot{q}_i)$ est une matrice diagonale composée des éléments : $\dot{q}_i, i = 1, \dots, n$;

- $diag(sign(\dot{q}_i))$ est une matrice diagonale composée des éléments :
 $sign(\dot{q}_i), i = 1, \dots, n;$

1.8. 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 :

1.8.1. 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 :

$$q(t = 0) = q^{ini} \quad \text{et} \quad q(t = T) = q^{fin} \quad (1.16)$$

$$\dot{q}(t = 0) = 0 \quad \text{et} \quad \dot{q}(t = T) = 0 \quad (1.17)$$

1.8.2. Contraintes sur les débattements :

Les articulations des robots sont conçues pour permettre une certaine liberté entre les solides rassemblés par cette liaison. Il est naturel que les débattements articulaires des corps soient limités par des butées mécaniques ou même par la conception de la liaison elle-même.

Par conséquent, tout mouvement dépassant les capacités tolérées de l'articulation risque de l'endommager. De ce fait, les mouvements planifiés doivent obligatoirement respecter cette contrainte.

La formulation explicite des contraintes sur les débattements articulaires est donnée comme suit :

$$\forall t \in [0, T] , |q_i(t)| \leq q_i^{max} \quad i = 1, \dots, n \quad (1.18)$$

t : Représente le temps de transfert de la tâche.

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

q_i^{max} : Représente la valeur maximale de la $i^{\text{ème}}$ articulation (valeur intrinsèque du robot).

1.8.3. Contraintes cinématiques :

De même pour les caractéristiques cinématiques des mouvements articulaires, les vitesses et les accélérations permises sont bornées. Ces limitations sont dues principalement à la conception de la liaison et à la technologie des actionneurs motorisant les liaisons actives.

Dans des cas particuliers, des limitations cinématiques supplémentaires peuvent être imposées par la nature de la tâche à exécuter. C'est le cas notamment lorsque les objets manipulés par le robot sont sensibles aux survitesses ou aux chocs de démarrage (accélération) et de freinage (décélération), nous citons l'exemple de transport des contenants déversant.

Les contraintes cinématiques s'écrivent de la manière suivante :

Pour les vitesses :

$$\forall t \in [0, T] , |\dot{q}_i(t)| \leq \dot{q}_i^{max} \quad i = 1, \dots, n \quad (1.19)$$

Pour les accélérations :

$$\forall t \in [0, T] , |\ddot{q}_i(t)| \leq \ddot{q}_i^{max} \quad i = 1, \dots, n \quad (1.20)$$

Pour le jerk :

$$\forall t \in [0, T] , |\dddot{q}_i(t)| \leq \dddot{q}_i^{max} \quad i = 1, \dots, n \quad (1.21)$$

Où \dot{q}_i^{max} , \ddot{q}_i^{max} et \dddot{q}_i^{max} sont imposés par les données technologiques du robot manipulateur.

1.8.4. Contraintes sur les couples moteurs :

Ces contraintes traduisent les capacités maximales des actionneurs. Elles sont imposées par la technologie de chaque actionneur. Pour la $i^{\text{ème}}$ articulation, la contrainte couple s'écrit :

$$\forall t \in [0, T] , |\tau_i(t)| \leq \tau_i^{max} \quad i = 1, \dots, n \quad (1.22)$$

Où τ_i^{max} représente la valeur maximale tolérée.

1.8.5. Contraintes dues aux obstacles dans l'espace opérationnel :

La planification des mouvements doit bien évidemment tenir compte de l'existence des obstacles. Les corps segmentaires du système articulé doivent être maintenus à l'écart des obstacles qui se présentent dans l'espace d'évolution du robot. Le mouvement optimal, dans ce cas, doit nécessairement éviter tout risque de collision.

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

$$g[q(t)] \leq 0 \quad (1.23)$$

Où g est une fonction traduisant, en général, 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.

La recherche d'évitement des collisions peut être montrée comme une zone active dans la littérature robotique et une partie essentielle de la planification de la trajectoire réussie pour les tâches assignées aux robots manipulateurs qu'elles soient imposées ou libres.

Dans le problème des tâches imposées, les auteurs [13] ont utilisé une technique de minimisation séquentielle sans contraintes (SUMT) pour obtenir des trajectoires optimales en minimisant une fonction objectif représentant le temps de transfert et l'énergie mécanique des actionneurs afin d'éviter des obstacles fixes qui se trouvent dans l'espace de travail du robot manipulateurs. Les auteurs de [14,15] ont minimisé la même fonction objectif que [13] avec l'utilisation des algorithmes évolutionnaires multi-objectifs, NSGA-II et MODE. Tandis que dans une étude plus récente les auteurs de [16] ont proposé une nouvelle classe des fonctions d'interpolation polynomiales qui peut être employée pour la planification de trajectoires des manipulateurs robotiques et pour le design industriel, ces fonctions permettent de garantir une continuité de la position, de la vitesse et de l'accélération, afin de s'assurer que la trajectoire en résultant est assez lisse. D'autres auteurs [17] proposent une approche d'optimisation pour la planification

globale des trajectoires pour les robots manipulateurs redondants le long des tâches spécifiées, en évitant les obstacles dans l'espace de travail ainsi que la singularité du robot manipulateur.

Dans les tâches libres, l'auteur de [18] a utilisé un algorithme d'optimisation capable d'obtenir des configurations possibles entre les positions initiale et finale afin de développer une trajectoire pour les manipulateurs robotiques industriels dans les espaces de travail en présence des obstacles. Son travail est composé de deux sections successives, un espace de configuration discret et une trajectoire optimale et réalisable des robots manipulateurs. Dans la première section, les deux configurations possibles du robot manipulateur ont été données pour composer un espace de configuration discret. Dans la deuxième section, la configuration sélectionnée est utilisée pour déterminer une trajectoire optimale sans collision. Ainsi, des trajectoires possibles ont été réalisées dans l'espace articulaire qui minimise le temps d'exécution total et sont dynamiquement compatibles avec les caractéristiques du robot manipulateur. Une autre trajectoire optimale à temps minimum avec un mouvement sans collision pour des robots industriels dans un environnement complexe a été introduite par [19]. Cette recherche était une extension de la recherche précédente menée dans [18]. Un critère de temps minimal optimal est considéré et la dynamique du manipulateur robotisé a été prise en compte pour un algorithme d'optimisation soumis aux contraintes telles que le couple, la puissance, le jerk, l'énergie consommée et l'évitement des collisions.

L'objet du travail [20] est de présenter une nouvelle stratégie pour la planification du mouvement basée sur l'utilisation de fonctions de puissance artificielle. Dans le schéma proposé, les fonctions sont déterminées à partir de la composition additive et multiplicative des fonctions représentant la position finale à atteindre et les obstacles. Un ensemble de procédures est présenté pour la conception des fonctions potentielles afin d'éviter les minimums locaux dans certains scénarios représentatifs.

Les auteurs de [21] ont proposé le modèle de potentiel généralisé pour éviter une collision et résoudre le problème de planification de la trajectoire pour un manipulateur hyper-redondant dans un espace de travail 3D. En utilisant l'information sur l'espace de travail, la force et le couple de répulsion entre le

manipulateur robotique et la surface de l'obstacle ont été déterminés par l'algorithme proposé. En tenant compte de ces forces et de ces couples, la configuration de potentiel minimum peut être déterminée pour le manipulateur robotisé. Ensuite, un chemin sans collision peut être réalisé en modifiant localement cette configuration [21].

1.9. Conclusion :

Dans le présent chapitre nous avons présenté d'une manière générale la terminologie utilisée en robotique et certaines notions de base qui reviennent souvent dans l'étude des structures polyarticulées. Nous avons abordé d'une manière approfondie la modélisation mathématique des robots en tenant compte du comportement cinématique, dynamique et des contraintes propres aux manipulateurs robotiques. Dans le chapitre suivant, nous aborderons d'une manière explicite la planification des trajectoires pour les tâches imposées et libres et les différents critères de performances à optimiser.

2. PLANIFICATION DES TRAJECTOIRES ET CRITERES DE PERFORMANCE

2.1. Introduction :

Les manipulateurs robotiques sont constitués d'une structure articulée dont la base est fixe et dont l'extrémité terminale est généralement munie d'un outil de préhension permettant de manipuler des objets. Ces robots sont souvent destinés à l'exécution de tâches complexes et répétitives dans de vastes espaces de travail. Afin d'améliorer l'efficacité d'un manipulateur robotique lors de l'exécution d'une tâche donnée, il est nécessaire de déterminer la meilleure trajectoire permettant d'exécuter cette tâche. La recherche de cette trajectoire est conditionnée par la satisfaction d'un critère de performance et le respect d'un certain nombre de contraintes. Ce problème peut être défini comme un problème d'optimisation non linéaire sous contraintes.

Dans ce chapitre, nous commencerons de montrer la différence entre les termes utilisés dans la planification de trajectoires et les différentes approches utilisées. Ensuite, nous allons exposer les critères de performances les plus employés et les plus représentatifs. Nous allons aborder par la suite la planification de trajectoires dans les tâches imposées et nous expliquerons notre approche pour résoudre le problème de planification de trajectoire en temps minimal. Enfin, nous allons présenter la planification de trajectoires optimales dans les tâches libres ainsi que notre approche proposée pour remédier au problème de trajectoire libre en temps et jerk minimal.

2.2. Planification des trajectoires :

La planification des trajectoires est un domaine de recherche important en robotique. La recherche dans ce domaine a commencé dans les années 1970 [22], mais dans la littérature il existe certaines confusions entre planification de trajectoires, chemin et mouvement. En réalité, ce sont trois problèmes distincts. Il

est important de définir avec précision les termes de trajectoire, du chemin et du mouvement afin d'éviter toute ambiguïté.

2.2.1. Planification des trajectoires, des chemins et des mouvements :

La planification des trajectoires et la génération des trajectoires sont souvent utilisées de façon synonyme, mais la planification des trajectoires est en fait une partie de la génération de trajectoires. La génération de trajectoires est formée à la fois par la planification des trajectoires et des mouvements (figure 2.1). La trajectoire est une séquence continue de configurations définie en fonction du temps t , reliant une configuration initiale q^{init} à l'instant $t = 0$ à une configuration finale q^{fin} à l'instant $t = T$, ou T est la durée totale de la trajectoire.

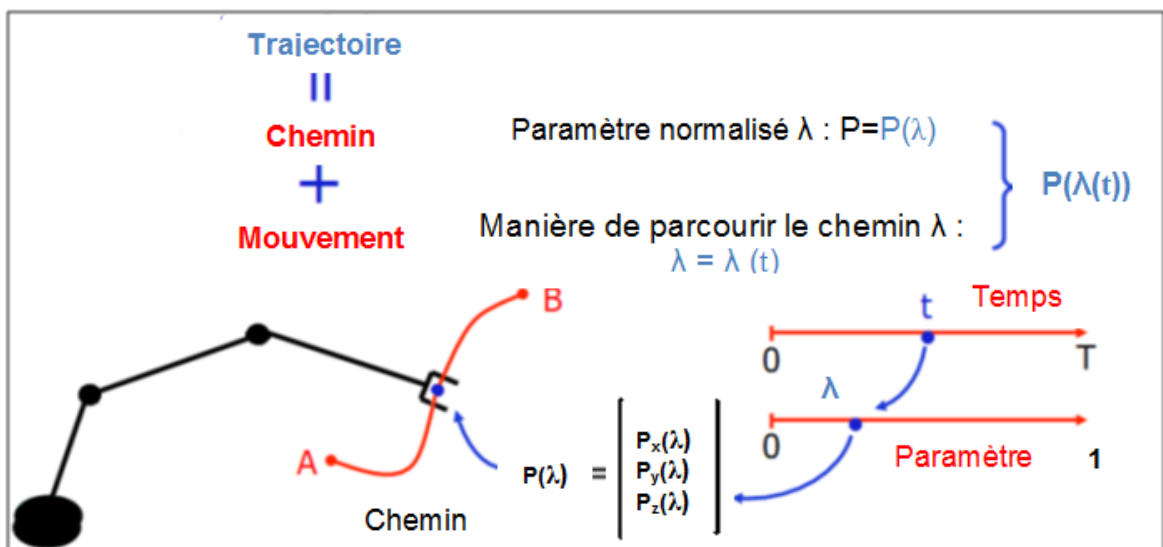


Figure 2.1: Procédure standard pour la planification de trajectoires dans les robots industriels [22].

Par contre, le chemin est une séquence continue de configuration $P(\lambda)$ définie indépendamment du temps en fonction d'un paramètre λ , reliant une configuration initiale q^{init} et une configuration finale q^{fin} . Généralement ce paramètre est normalisé ($\lambda \in [0,1]$). L'abscisse curviligne $P(\lambda)$ est définie en spécifiant les contraintes du chemin et du robot manipulateur telles que les limitations de vitesse, d'accélération et du jerk. Toutefois, on peut toujours construire une trajectoire à partir d'une donnée d'un chemin et d'un mouvement sur ce chemin.

La fonction mouvement est une fonction scalaire $\lambda(t)$ croissante qui permet de définir la manière selon laquelle on parcourt le chemin dans l'espace temporel.

A cet effet, et dans la mesure où la planification de trajectoires prend en compte une dimension supplémentaire, le temps et des contraintes supplémentaires, on peut s'attendre à ce que planifier une trajectoire soit un problème au moins aussi complexe que planifier un chemin.

2.2.2. Planification globale et planification locale :

De nombreux algorithmes de planification de chemin ont été présentés dans la littérature, ils peuvent être divisés en deux catégories [23] : algorithmes globaux et locaux.

Les méthodes globales sont basées sur une connaissance complète de l'environnement dans lequel le robot évolue. L'espace libre est généralement représenté dans l'espace des configurations du robot [24] (appelé aussi *C-Space*) par une série de paramètres donnant la position et l'orientation de l'effecteur. L'avantage de formuler le problème de planification dans *C-Space* est qu'il devient équivalent à la navigation d'un point objet. Afin de pouvoir représenter l'espace des configurations libres, les méthodes globales ont besoin d'une transformation des obstacles à partir de leurs représentations cartésiennes en une représentation dans l'espace de configuration du robot [25]. Il est ainsi possible de déterminer les configurations qui emmènent le robot de la configuration initiale à la configuration finale souhaitée, en évitant celles qui provoquent des collisions. Il faut mentionner que les méthodes globales ne sont pas adaptées aux environnements qui changent lors du déplacement du robot et pour tout changement il faut recalculer l'espace libre accessible par le robot qui est très gourmand en temps de calcul. De plus, le calcul augmente exponentiellement avec le nombre de degrés de liberté.

Pour pallier aux problèmes de temps de calcul des méthodes globales et pour permettre la navigation des robots dans des environnements dynamiques, les méthodes locales ont été proposées. En plus, ces méthodes n'ont besoin que d'une connaissance partielle de l'espace de travail. Elles cherchent une trajectoire au fur et à mesure que le robot avance. La trajectoire initiale est généralement représentée par une ligne droite liant la position initiale à la position finale dans l'espace de configurations [25]. À chaque étape de l'algorithme, le processus

vérifie l'existence d'un éventuel obstacle. Et dans un tel cas, la trajectoire est alors modifiée localement afin d'éviter la collision.

2.3. Critères de performance dans la recherche de trajectoires :

Le critère de performance représente un coût à minimiser lors du transfert et il permet d'engendrer des allures caractéristiques de mouvement. 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 trois formes distinctes : une fonctionnelle de type intégrale, de type terminal ou de type mixte. Bien que formulés différemment, ces trois 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 :

$$F(u, T) = \int_0^T L(x(t), u(t)) dt \quad (2.1)$$

Avec :

L : Lagrangien qui peut prendre différentes formulations selon l'objectif visé par le processus d'optimisation.

$x(t)$: Vecteur des variables d'état qui décrivent l'évolution du système à l'instant $t \in [0, T]$. Dans un problème de planification de trajectoires des robots manipulateurs, le vecteur $x(t)$ représente les variables articulaires de position $q(t)$, de vitesse $\dot{q}(t)$, d'accélération $\ddot{q}(t)$ et du jerk $\dddot{q}(t)$ qui décrivent l'état géométrique et cinématique du manipulateur.

$u(t)$: Vecteur qui représente la commande à optimiser considérée comme une inconnue du problème. Dans un problème de planification de trajectoires, la commande $u(t)$ représente les couples moteurs $\tau(t)$ qui génèrent le mouvement du robot.

T : Temps total d'exécution de la tâche assignée au robot manipulateur.

Dans le cas d'un problème de planification de trajectoires de manipulateurs robotiques, le critère (2.1) s'écrit de la manière suivante :

$$F(\tau, T) = \int_0^T L(q(t), \dot{q}(t), \ddot{q}(t), \tau(t)) dt \quad (2.2)$$

Dans la littérature scientifique, presque toutes les techniques des problèmes de planification de trajectoire sont basées sur l'optimisation de certains paramètres ou certaines fonctions objectifs. Nous citons dans la suite les critères les plus utilisés et les plus représentatifs.

2.3.1. Optimisation de la durée de parcours :

Le critère de performance basé sur la minimisation du temps d'exécution de la tâche a suscité l'intérêt d'une grande partie de chercheurs roboticiens, il reflète exclusivement la rapidité d'exécution de la tâche. La minimisation du temps de parcours est très significative lorsque le robot travaille d'une manière cyclique, où les mêmes tâches se répètent périodiquement. La minimisation du temps dans ce cas se répercute directement sur la rentabilité du processus.

Cependant, la dynamique du robot durant le transfert n'est pas représentée directement dans la formulation du critère. Ceci conduit en général à des commandes optimales discontinues, de type bang-bang (couples saturés).

La formulation de ce critère est la plus simple en comparaison avec les autres qui existent. En effet, la commande $\tau(t)$ n'apparaît pas dans l'expression du lagrangien. Ce dernier prend la forme suivante :

$$L(q(t), \dot{q}(t), \ddot{q}(t), \tau) = 1 \quad (2.3)$$

La fonction objectif dans ce cas peut se mettre sous la forme suivante :

$$F(T) = \int_0^T dt = T \quad (2.4)$$

Il n'est donc pas étonnant qu'on trouve dans la littérature robotique de nombreux articles proposant diverses approches dédiées à ce critère.

Pour les tâches imposées, le critère de temps minimum a été présenté dans plusieurs travaux. Dans ce contexte les travaux de [26] ont été parmi les premiers à aborder le calcul des trajectoires optimales pour les tâches imposées à travers

l'interpolation de l'ensemble de nœuds dans l'espace articulaire. Les nœuds qui forment la trajectoire sont obtenus à partir d'un ensemble de positions discrètes de l'organe terminal. Quatre ans plus tard, les auteurs de [27] ont proposé une formulation des trajectoires pour les robots industriels en utilisant des fonctions B-splines quadratiques. Cependant, leur algorithme et pour la même étude de cas n'a pas pu obtenir un meilleur temps d'exécution que l'algorithme proposé dans [26]. Plus tard, l'algorithme des auteurs [26] a été utilisé par [28], où les trajectoires sont modélisées en utilisant les fonctions B-splines cubiques. Néanmoins, leur procédure n'a pas produit de meilleur temps d'exécution que celle de [26], mais elle a apporté plus de rapidité en termes de temps de calcul. Les auteurs de [29] ont développé une version améliorée de l'algorithme d'optimisation proposé dans [26] pour inclure des contraintes dynamiques dans les calculs, tandis que d'autres utilisaient l'analyse par intervalles pour résoudre le problème de trajectoire temporelle minimum soumis à des contraintes cinématiques [30]. D'un autre côté, Tse et Wang [31] ont résolu l'algorithme de Lin et al. [26] en utilisant des algorithmes génétiques. Dans une étude récente faite par [32] les auteurs ont proposé une approche qui utilise plusieurs populations dans l'algorithme génétique avec une technique de migration, les résultats trouvés de cette approche ont été comparés à ceux présentés par [31].

Une autre méthode exposée par les auteurs de [33] présente en détail la solution de [34], [35] et [36]. Ces auteurs [33] ont développé un planificateur de trajectoire de temps minimum (la méthode du plan de phase) qui tient compte des effets dynamiques, en supposant que le chemin souhaité est donné sous une forme paramétrée. Cette forme est substituée dans les équations dynamiques du robot manipulateur pour obtenir des courbes d'accélération et de décélération maximales admissibles. En utilisant le fait que la solution est de type bang-bang dans l'accélération (par l'application de PMP), la trajectoire optimale est construite en détectant les points de commutation entre deux courbes d'accélération le long du chemin paramétrique. Par la suite [37] et [38] ont amélioré cette technique et ont étudié certains cas particuliers non abordés précédemment. La méthode du plan de phase s'est révélée efficace dans la pratique pour les manipulateurs à six degrés de liberté mais malheureusement, elle ne convient qu'aux problèmes de temps minimum qui conduisent généralement à des profils de contrôle de type bang-bang et qui peuvent endommager le système. Une approche pour obtenir

une trajectoire optimale du robot manipulateur a été détaillée dans [39]. Les fonctions splines cubiques « *Clamped* » ont été utilisées pour modéliser la trajectoire et une méthode dite Harmony Search (HS) pour résoudre le problème d'optimisation, les différentes contraintes cinématiques ont été prises en compte. Les auteurs de [40] ont proposé une approche basée sur la détermination d'un vecteur d'intervalle du temps. Ce dernier est introduit directement dans la fonction des positions articulaires et de ses dérivées qui sont modélisées en utilisant les splines cubiques. Les algorithmes génétiques ont été utilisés pour déterminer le temps optimal des manipulateurs robotiques en tenant compte des contraintes kinodynamiques.

Pour les tâches libres, Les auteurs Kahn et Roth [41] sont parmi les premiers chercheurs qui ont traité le problème de planification de trajectoires à temps minimum pour les manipulateurs robotiques qui exécutent des tâches point à point. Dans leur recherche, un robot sériel à trois bras est étudié et les limitations sur les couples d'actionneurs ont été supposées constantes pendant le mouvement désiré. Pour déplacer l'organe terminal du manipulateur, les couples d'actionneurs maximaux autorisés ont été déterminés par un algorithme d'optimisation. La trajectoire du manipulateur robotique n'a pas été spécifiée, mais seule la destination finale a été donnée. Bien que ce travail fût pratique pour certaines tâches robotiques, de nombreuses implémentations nécessitent souvent une planification de trajectoire essentielle pour spécifier la trajectoire du manipulateur robotique afin d'éviter la collision avec des obstacles.

Pour planifier des trajectoires à temps minimal des robots industriels le long d'une tâche point à point, les auteurs de [42] ont adopté une approche qui a été proposée auparavant pour traiter le problème de l'exploitation optimale des robots mobiles par [43]. En effet, cette approche est basée sur la normalisation de l'échelle de temps ensuite sur la décomposition de la fonction trajectoire en deux fonctions. Ces dernières sont modélisées en utilisant, les splines cubiques *natural* et *clamped*. La minimisation de la fonction objectif s'effectue en utilisant une technique stochastique. Dans des travaux plus récents présentés par [44] un algorithme est proposé pour obtenir des trajectoires de robots industriels travaillant dans des environnements industriels encombrés d'obstacles. Les solutions obtenues sont des trajectoires optimales avec un temps minimum, qui

répondent aux contraintes physiques du robot (les valeurs maximales du couple moteurs, la puissance et le jerk pour chaque actionneur), à l'évitement des collisions et aux contraintes dues aux énergies consommées. Les résultats d'un test effectué sur un robot PUMA560 ont montré l'efficacité de l'algorithme proposé en matière du temps d'exécution de la tâche et du temps de calcul de l'algorithme en comparant avec ceux trouvés précédemment par [45].

2.3.2. Optimisation du temps-efforts quadratiques/Energies :

Comparativement au critère précédent qui implique des discontinuités lors des transferts de charges et qui est 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) [10] [46]. Le lagrangien prend la forme suivante :

$$L(q(t), \dot{q}(t), \ddot{q}(t), \tau) = \mu T + \frac{1-\mu}{2} \sum_{i=1}^n \left(\frac{\tau_i(t)}{\tau_i^{max}} \right)^2 \quad ; \quad \mu \in]0,1[\quad (2.5)$$

τ_i^{max} 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.

Pour ce critère de performance, il est possible de trouver une vaste littérature scientifique consacrée à la présentation d'algorithmes de planification de trajectoires minimisant le critère de performance donné par le temps d'exécution et l'effort quadratique [46-58].

Pour les tâches imposées :

Les auteurs [46] [47] proposent un critère mixte minimisant à la fois le temps de parcours et les efforts actionneurs, ce critère permet d'approcher la solution en un 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 gênant pour le bon fonctionnement du robot manipulateur. Un manipulateur robotique planaire à deux degrés de liberté ainsi que le bras robotique à entraînement direct UCLS ont été utilisés pour des études théoriques et expérimentales afin de vérifier l'efficacité de la méthode d'optimisation proposée.

Shiller [48] a traité le problème d'optimisation énergétique en utilisant (PMP). Les trajectoires optimales ont été obtenues en résolvant un problème de valeur limite à deux points de quatrième ordre et les contrôles correspondant trouvés sont moins brusques.

Une autre approche présentée par les auteurs de [49] qui ont proposé une solution au problème du déplacement d'un robot manipulateur le long d'un chemin géométrique spécifié en modélisant les trajectoires par les fonctions paramétriques des splines cubiques. Une technique de minimisation séquentielle non-contrainte (SUMT) est utilisée pour optimiser une fonction objectif qui représente le temps de la tâche et l'énergie des articulations sous des contraintes kinodynamiques. Le résultat optimal a été observé en modifiant les deux termes de la fonction objectif qui prend en compte le modèle dynamique du manipulateur et l'évitement des collisions. Deux robots à trois et six degrés de liberté ont été utilisés pour vérifier l'efficacité de la méthode proposée.

Une autre technique pour la planification de trajectoire optimale des manipulateurs robotiques a été décrite dans [50]. La technique est basée sur la minimisation d'une fonction objectif qui prend en compte à la fois le temps d'exécution et l'énergie totale tout au long d'une trajectoire spécifiée. En outre, les contraintes cinématiques dynamiques, exprimées au moyen des limites supérieures de la vitesse, de l'accélération, du jerk et de la force / couple sont prises en compte. Ici, la trajectoire du robot est composée par des paires de points de passage et modélisée en utilisant les splines cubiques. Un optimum global peut être atteint en utilisant une technique d'optimisation proposée.

Cependant, toute méthode d'optimisation de trajectoire avec une contrainte non linéaire, telle que l'algorithme de programmation quadratique séquentielle (SQP), peut aboutir à un minimum local. Pour surmonter ce problème, *Ramabalan* et *Saravanan* [14] ont présenté une technique pour planifier des trajectoires optimales basées sur les algorithmes génétiques. La trajectoire optimisée qui en résulte a été assurée de manière suffisamment souple et rapide par la technique

d'optimisation proposée, en plus de cela, une solution optimale globale était réalisable. Les contraintes géométriques et kinodynamiques ont été prises en compte lors du processus d'optimisation de la trajectoire des robots manipulateurs (robot cartésien (PP) avec deux articulations prismatiques et robot PUMA 560).

Pour les tâches libres :

Les auteurs [51] 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 dynamiques de robot manipulateur, par contre dans [52] les auteurs proposent de résoudre le même problème avec l'influence du nombre de nœuds sur le calcul de la trajectoire et des coefficients de pondération sur la fonction objectif tout en minimisant l'énergie des actionneurs. Pour [53], la trajectoire est modélisée par des fonctions B-splines cubiques ; les limites physiques des articulations s'ajoutent aux contraintes kinodynamiques ont été prises en considération. La fonction objectif inclut également un terme supplémentaire (fonction de pénalité), afin d'éviter les obstacles mobiles qui se trouvent dans l'espace de travail du robot manipulateur de type STANFORD.

Les auteurs de [54] ont considéré le couple, l'accélération et la vitesse des articulations comme des contraintes et le temps et l'énergie optimaux comme le but. Ensuite, le problème de contrôle optimal générique a été transféré à un problème d'optimisation non-linéaire sous contraintes, traité par une méthode de programmation quadratique séquentielle. (SQP).

L'approche proposée par [42] a été entendue dans [55, 56] pour prendre en compte d'une part les contraintes telles que les contraintes sur les débattements, les vitesses, les accélérations et le jerk et d'autre part des manipulateurs robotiques avec un nombre de degrés de liberté important.

Les auteurs de [57] ont proposé une combinaison d'une méthode directe pour l'estimation des variables adjointes et une méthode indirecte basée sur les fonctions Hamiltoniennes afin d'obtenir des trajectoires de temps minimum et d'énergie minimale pour le robot industriel Manutec-r3. L'objectif de cette combinaison est d'assurer un vaste domaine de convergence et des solutions très précises. Le problème d'optimisation est traité à l'aide d'une méthode de programmation quadratique séquentielle (SQP). Dans [58], la même approche a été employée pour calculer les trajectoires optimales d'un robot industriel (ABB

IRB 6400). Des problèmes pratiques inhérents à la nature des contrôleurs de robots sont également rapportés.

2.3.3. Optimisation du temps-jerk :

Le jerk est aujourd'hui la dénomination connue et acceptée de la dérivée temporelle de l'accélération, i.e. la dérivée 3^{ème} de la position. Les dérivées d'ordre supérieur sont respectivement nommées, de façon non officielle, le snap, le crackle et le pop. Au-delà de la dérivée 6^{ème} de la position, il n'existe plus de nom particulier. Ces dérivées se rencontrent presque exclusivement dans le domaine de l'étude et/ou de la reproduction de mouvements humains [59, 60]. En mécanique, le jerk, traduisant le taux de variation de l'accélération par unité de temps, est utilisé comme critère caractérisant la douceur d'une trajectoire (en langue anglaise : *smoothness*).

La diminution du temps d'exécution de la tâche est importante pour augmenter la productivité des robots manipulateurs. En outre, la limitation des secousses est très importante, car les valeurs élevées de jerk peuvent épuiser la structure du robot et exciter fortement ses fréquences de résonance. Les vibrations induites par des trajectoires non lisses peuvent endommager les actionneurs du robot et introduire de grosses erreurs pendant que le robot exécute ses tâches telles que le suivi de la trajectoire, de plus les trajectoires de faible jerk peuvent être exécutées plus précisément et avec exactitude [61].

La fonction objectif qui minimise deux termes composés d'une part par le terme proportionnel au temps d'exécution total de la tâche et d'autre part le terme proportionnel au jerk total est représentée par l'expression mathématique suivante:

$$F_{obj} = \min \left(k_T \sum_{i=1}^N T_i + k_J \sum_{i=1}^N \max_{h_i} |J_i(t)| \right) \quad (2.6)$$

Où T_i est le temps d'exécution total de la tâche ; k_T et k_J sont les coefficients de pondération selon les besoins de l'utilisateur qui peut favoriser soit le temps d'exécution de la tâche soit le jerk.

En effet, de nombreux travaux dans le domaine de la robotique ont été consacrés à l'étude du problème de la planification du mouvement en considérant l'ensemble du temps d'exécution et le jerk comme des paramètres importants à minimiser.

Dans ce contexte nous citons :

Pour les tâches imposées :

Une approche originale [62] traite le problème de la planification de trajectoire du robot manipulateur dans des tâches imposées en considérant seulement les contraintes cinématiques. La fonction d'optimisation est scindée en un premier terme représentant la durée du mouvement et en un second terme proportionnel au carré du jerk par axe. Cette fonction permet de pondérer le compromis entre durée du mouvement et souplesse de la trajectoire, ainsi elle est minimisée en utilisant la programmation quadratique séquentielle (SQP). Un an plus tard, les mêmes auteurs de [62] ont résolu la même fonction objectif en utilisant les fonctions splines cubique pour la construction de la trajectoire [63]. Ces travaux ont été étendus plus tard dans [64], où une validation expérimentale a été effectuée par un algorithme de planification de trajectoire optimale en minimisant un critère hybride temps-jerk. Les profils de trajectoires ont été modélisés et comparés avec les fonctions splines cubiques et les B-splines du cinquième ordre. Dans leur algorithme d'optimisation de trajectoire, l'exigence d'une exécution rapide de la tâche où l'exigence d'une trajectoire lisse du robot manipulateur peut être réalisée en modifiant les valeurs des deux facteurs de pondération de la fonction objectif hybride.

Dans [65] et [66], un algorithme basé sur l'analyse par intervalles est présenté. Cette méthode minimise globalement la valeur absolue maximale du jerk suivant une trajectoire dont le temps d'exécution est imposé a priori. Par conséquent, il s'agit d'une approche minimax limitée au temps d'exécution de la trajectoire. Les trajectoires sont exprimées au moyen de splines cubiques et les intervalles entre les nœuds intermédiaires ont été calculés de manière à produire le pic du jerk le plus bas possible. Cependant, ces techniques considèrent le temps d'exécution comme connu (et défini a priori); de plus, il n'est pas possible de définir une contrainte cinématique quelconque sur la trajectoire, car elle n'est pas prise en compte.

Pour les tâches libres :

Dans [67], les auteurs ont obtenu des solutions analytiques d'un problème de planification de trajectoire optimale pour un mouvement point à point. Les solutions ont été basées sur l'optimisation du jerk minimal avec des valeurs de vitesse et d'accélération imposées nulles. L'optimisation, a été réalisée en appliquant le principe de *Pontryagin*, elle concerne spécifiquement deux fonctions objectifs l'une concerne la valeur absolue maximale du jerk (approche minimax) et l'autre concerne l'intégrale du jerk au carré.

2.3.4. Optimisation multi- objectifs :

Afin de réaliser les tâches prévues dans l'industrie, les capacités des robots industriels ont été restreintes en raison des limites de profils d'accélération et de décélération maximales le long de la courbe de solution, plusieurs points indésirables apparaissent tels que les points singuliers, les points critiques pour la configuration du robot et les obstacles. Toutes ces restrictions dans la pratique conduisent à un échec dans la recherche de la solution optimale. La détermination de la solution optimale exige un temps de calcul trop long et une capacité de mémoire de machine très élevée.

Pour les tâches imposées :

Dans [13], les auteurs ont présenté deux stratégies différentes pour résoudre le problème de trajectoire optimale pour les manipulateurs robotiques en présence d'obstacles fixes. La première stratégie correspond au cas où des points intermédiaires permettent d'éviter les obstacles. La seconde traite le cas où seules les configurations limites sont spécifiées. Les trajectoires sont modélisées à l'aide de fonctions splines paramétriques et le problème d'optimisation est résolu en utilisant une technique de minimisation séquentielle non contrainte (SMUT) tout en considérant une fonction multi-objectifs qui comporte le temps de la tâche, le couple moteur et la fonction anticollision. Dans ce même contexte, des travaux plus récents de [68] ont présenté une application typique traitant le placement approprié dans le site de travail ainsi que les trajectoires optimales pour un robot manipulateur de type SCARA. Une fonction multi-objectifs proposée dans ce travail considère la minimisation de l'ensemble du temps d'exécution, de l'énergie totale et le jerk en tenant compte de l'équation dynamique du robot ainsi que les limites technologiques sur les débattements, les vitesses, les accélérations, le jerk

et les couples moteurs de chaque articulation. L'interpolation linéaire sphérique du quaternion ainsi que les polynômes des splines d'ordre cinq ont été utilisés pour construire les trajectoires du robot. Des solutions optimales avant et après le réglage du placement du robot ont été comparées et analysées. D'après les résultats obtenus, il est constaté que le placement du robot a un grand impact sur les résultats optimaux, ce qui montre le placement approprié du robot permet d'améliorer la performance en termes de fonction objectif et définir beaucoup mieux la trajectoire optimale.

Pour les tâches libres :

Les auteurs de [69] 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. Différentes contraintes cinématiques, dynamiques, et des obstacles ont été imposées sur le mécanisme.

Plus tard, une procédure d'optimisation de trajectoire sous des contraintes non linéaires a été proposée dans les travaux de [71]. Cette dernière est basée sur un algorithme évolutif (NSGA-II) et un algorithme d'évolution différentielle (DE) pour les obstacles fixes et mobiles. La fonction multi-objectif combinée a été formulée au moyen de six critères d'optimisation qui sont la durée de la tâche, la moyenne quadratique des efforts d'actionneur, la pénalité pour le mouvement sans collision, l'évitement de la singularité, le jerk et les accélérations articulaires. L'objectif est de trouver une trajectoire optimale sous différentes contraintes formulée par des fonctions B-splines. Avec cet algorithme d'optimisation de trajectoire, une solution optimale globale est possible et la méthode peut être considérée comme applicable pour tous les types de robots sériels. Ce même algorithme d'optimisation de trajectoire a été proposé aussi par [70] où les auteurs s'intéressent cette fois-ci à la minimisation d'une fonction objectif dont les paramètres sont le temps de la tâche, le couple moteur et la puissance quadratique de chaque actionneur. Toutefois des contraintes géométriques, kinodynamiques et les équations dynamiques de mouvement du robot manipulateur PUMA560 sont considérés. L'approche d'optimisation basée sur (NSGA-II et DE) converge rapidement et montre sa nature supérieure en donnant de meilleurs résultats que la méthode d'optimisation conventionnelle SQP utilisée dans [69].

D'autres auteurs [51] 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 du robot manipulateur.

Les auteurs de [72] ont étendu les travaux de [55] et ont développé un algorithme qui permet de minimiser une fonction objectif qui représente une pondération entre le temps de transfert, les efforts des actionneurs et l'énergie mécanique consommée ; avec la satisfaction des contraintes géométriques, cinématiques et dynamiques. Une application consacrée à un système de robot planaire à trois d.d.l illustre la méthodologie développée.

2.4. Planification des trajectoires optimales dans les tâches imposées :

Selon la tâche que doit accomplir le robot manipulateur, il peut être nécessaire de spécifier la trajectoire de l'effecteur dans l'espace opérationnel. C'est le cas notamment lorsque l'outil fixé en fin de chaîne agit sans interruption selon un parcours déterminé. Un tel mouvement peut, par exemple, correspondre à des tâches de coupe, de soudage ou de projection. L'optimisation dans ce cas porte sur les modalités de parcours de la trajectoire.

2.4.1. Modélisation de la trajectoire avec des splines cubiques :

Les splines cubiques sont des fonctions polynomiales par morceaux du 3^{ème} degré (Figure 2.2). Les vitesses et accélérations des conditions initiales et finales (v_1, v_m, a_1 et a_m) sont spécifiées nulles. Ces conditions font que deux équations de l'algorithme spline deviennent nulles et que le modèle ne peut pas être résolu. Par conséquent, deux nœuds supplémentaires (valeurs de position à l'instant t_2 et t_{m-1}) sont ajoutés et leurs valeurs de position ne sont pas spécifiées.

Dans notre problème de planification des trajectoires, nous supposons que le robot manipulateur possède N articulations et que la trajectoire de chaque articulation a $M + 1$ nœuds ou points de contrôle, y compris le premier et le dernier nœud. Ainsi, pour chaque articulation, il existe M intervalles de temps et nous choisissons le deuxième et l'avant-dernier point de contrôle comme nœuds supplémentaires pour représenter la trajectoire du robot [40,61].

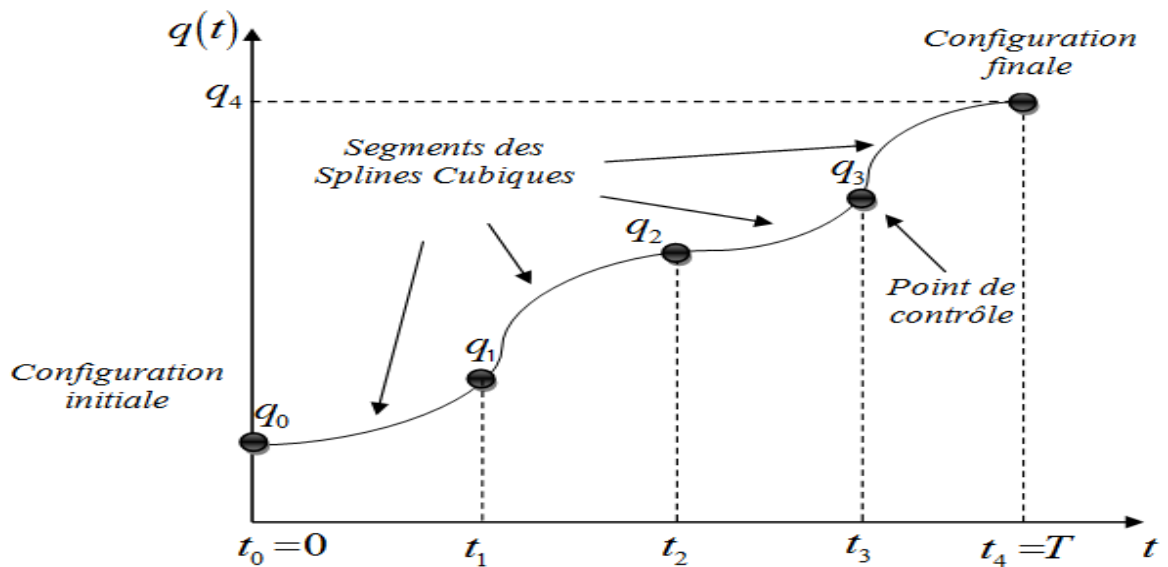


Figure 2.2: Trajectoire du robot manipulateur modélisée par les Splines Cubiques

Soit $Q_i(t)$ le polynôme cubique de l'articulation i dans l'intervalle de temps $[t_i, t_{i+1}]$. La dérivée seconde de $Q_i(t)$ est une interpolation linéaire qui peut s'écrire sous la forme suivante [64] :

$$W_i(t) = \frac{t_{i+1}-t}{h_i} W_i(t_i) + \frac{t-t_i}{h_i} W_i(t_{i+1}) \quad (2.7)$$

Où $i = 1, 2, \dots, n-1$ et $h_i = t_{i+1} - t_i$

En intégrant l'équation (2.7) deux fois et en imposant les conditions $Q_i(t_i)=q_i$ et $Q_i(t_{i+1})=q_{i+1}$, on obtient les fonctions des vitesses articulaires et des positions articulaires suivantes :

$$V_i(t) = -\frac{W_i(t_i)}{2h_i} (t_{i+1}-t)^2 + \frac{W_i(t_{i+1})}{2h_i} (t-t_i)^2 + \left[\frac{q_{i+1}}{h_i} - \frac{h_i W_i(t_{i+1})}{6} \right] - \left[\frac{q_i}{h_i} - \frac{h_i W_i(t_i)}{6} \right] \quad (2.8)$$

Et:

$$Q_i(t) = \frac{W_i(t_i)}{6h_i} (t_{i+1}-t)^3 + \frac{W_i(t_{i+1})}{6h_i} (t-t_i)^3 + \left[\frac{q_{i+1}}{h_i} - \frac{h_i W_i(t_{i+1})}{6} \right] (t-t_i) + \left[\frac{q_i}{h_i} - \frac{h_i W_i(t_i)}{6} \right] (t_{i+1}-t) \quad (2.9)$$

Ensuite, les valeurs des positions pour les deux nœuds supplémentaires q_2 et q_{m-1} ne sont pas fixes et elles sont utilisées pour ajouter deux nouvelles équations au système de manière à ce qu'il puisse être résolu. Les déplacements articulaires de ces deux nœuds s'écrivent comme suit :

En dérivant l'équation (2.7) et nous tirons le jerk de la trajectoire pour $i^{\text{ème}}$ articulation, qui s'écrit comme suit :

$$J_i(t) = -\frac{1}{h_i}W_i(t_i) + \frac{1}{h_i}J_i(t_{i+1}); \quad i = 1, 2, \dots, m-1 \quad (2.13)$$

2.4.2. Formulation de contraintes liées aux capacités limites du système :

Les contraintes dues aux vitesses du problème d'optimisation sont formulées en valeurs absolues maximales des vitesses aux points extrêmes t_i et t_{i+1} ou t_i avec $V_{ji}^{\otimes} = V_{ji}(t_i) = 0$ dans chaque intervalle. La vitesse est calculée en utilisant l'équation (2.8). Les contraintes liées aux capacités limites en vitesse du système deviennent :

$$\max \left\{ |V_{ji}(t_i)|, |V_{ji}(t_{i+1})|, |V_{ji}^{\otimes}| \right\} \leq V_j^{\max}; \quad j = 1, \dots, N; \quad \text{et } i = 1, \dots, m-1 \quad (2.14)$$

L'accélération est la solution du système d'équation (2.12). Les contraintes sont formulées à partir de la fonction d'accélération linéaire et la valeur absolue maximale existe aux points extrêmes t_i au t_{i+1} . Les contraintes liées aux capacités limites en accélération du système deviennent :

$$\max \left\{ |W_{j,1}|, \dots, |W_{j,n}| \right\} \leq W_j^{\max}; \quad j = 1, \dots, N \quad (2.15)$$

Le jerk est calculé en utilisant l'équation (2.7). Les contraintes sont formulées à partir de la fonction linéaire du jerk et la valeur absolue maximale existe aux points extrêmes t_i et t_{i+1} . Les contraintes liées aux capacités limites en jerk du système deviennent :

$$\max \left\{ |J_{j,1}|, \dots, |J_{j,n}| \right\} \leq J_j^{\max}; \quad j = 1, \dots, N \quad (2.16)$$

2.4.3. Initialisation du processus de génération des trajectoires :

Soit le vecteur d'intervalle du temps $h_i = [h_1, h_2, \dots, h_{m-1}]$. Pour initialiser le processus d'optimisation, nous considérons que :

$$h^{(0)} = \max_{j=1, \dots, m-1} \left[\left(\frac{|q_{j2} - q_{j1}|}{V_j^{\max}} \right), \left(\frac{|q_{j3} - q_{j2}|}{V_j^{\max}} \right), \dots, \left(\frac{|q_{jm} - q_{j,m-1}|}{V_j^{\max}} \right) \right] \quad (2.17)$$

Comme les deux nœuds supplémentaires sont nécessaires, ils sont initialement pris comme suit :

$$q_{j2} = (q_{j1} + q_{j3}) / 2 \text{ et } q_{j,m-1} = (q_{j,m-2} + q_{jm}) / 2 \quad (2.18)$$

Au cours du processus d'optimisation, tous les nœuds intermédiaires et les deux nœuds supplémentaires ne généreront qu'horizontalement comme le montre la figure 2.3, par conséquent la trajectoire change et se déplace également horizontalement en minimisant la fonction objectif et on obtient le vecteur d'intervalle de temps optimal h^{opt} , soit :

$$h^{opt} = [h_1^{opt}, h_2^{opt}, \dots, h_{m-1}^{opt}] \quad (2.19)$$

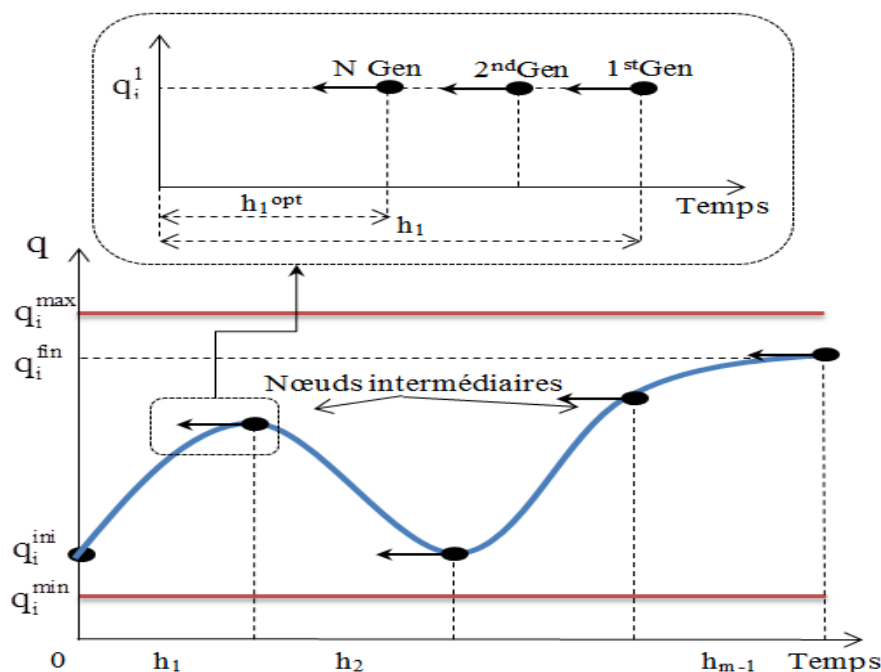


Figure 2.3: Représentation des nœuds intermédiaires et supplémentaires de la trajectoire modélisée par les fonctions Splines Cubiques.

2.4.4. Optimisation par algorithmes génétiques dans les tâches imposées :

Afin de rendre cette approche exploitable, il est nécessaire d'utiliser une technique d'optimisation appropriée pour minimiser la fonction objectif, nous proposons d'utiliser les algorithmes génétiques (GAs).

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 où leur nombre est défini selon le nombre des points des contrôles utilisés pour générer la fonction de positions articulaires. 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 contrôle intermédiaires. La figure 2.4 montre l'organigramme utilisé dans le prochain chapitre pour optimiser les fonctions objectifs dans les tâches imposées.

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

Pour créer la prochaine génération, trois opérateurs génétiques sont appliqués :

- La reproduction : habituellement la stratégie générale de reproduction est que les chromosomes (parents) avec meilleure fonction de coût ont plus la possibilité de se 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 nouveaux 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. [73] [74] [75] [76] [77]

Ce cycle est répété jusqu'à ce qu'une solution satisfaisante soit trouvée. Il y a quatre points principaux qui font la différence fondamentale entre ces algorithmes et d'autres méthodes :

- Les GAs utilisent un codage des paramètres, pas les paramètres eux-mêmes.
- Les GAs travaillent sur une population de points, au lieu d'un seul point.
- Les GAs utilisent seulement les valeurs de la fonction objectif étudiée, pas sa dérivée, ou une autre connaissance auxiliaire.
- Les GAs sélectionnent la population suivante en utilisant des calculs probabilistes, non déterministes.

Dans les exemples, nous allons fixer le taux ou la probabilité de croisement à 65% et la probabilité de mutation à 4%.

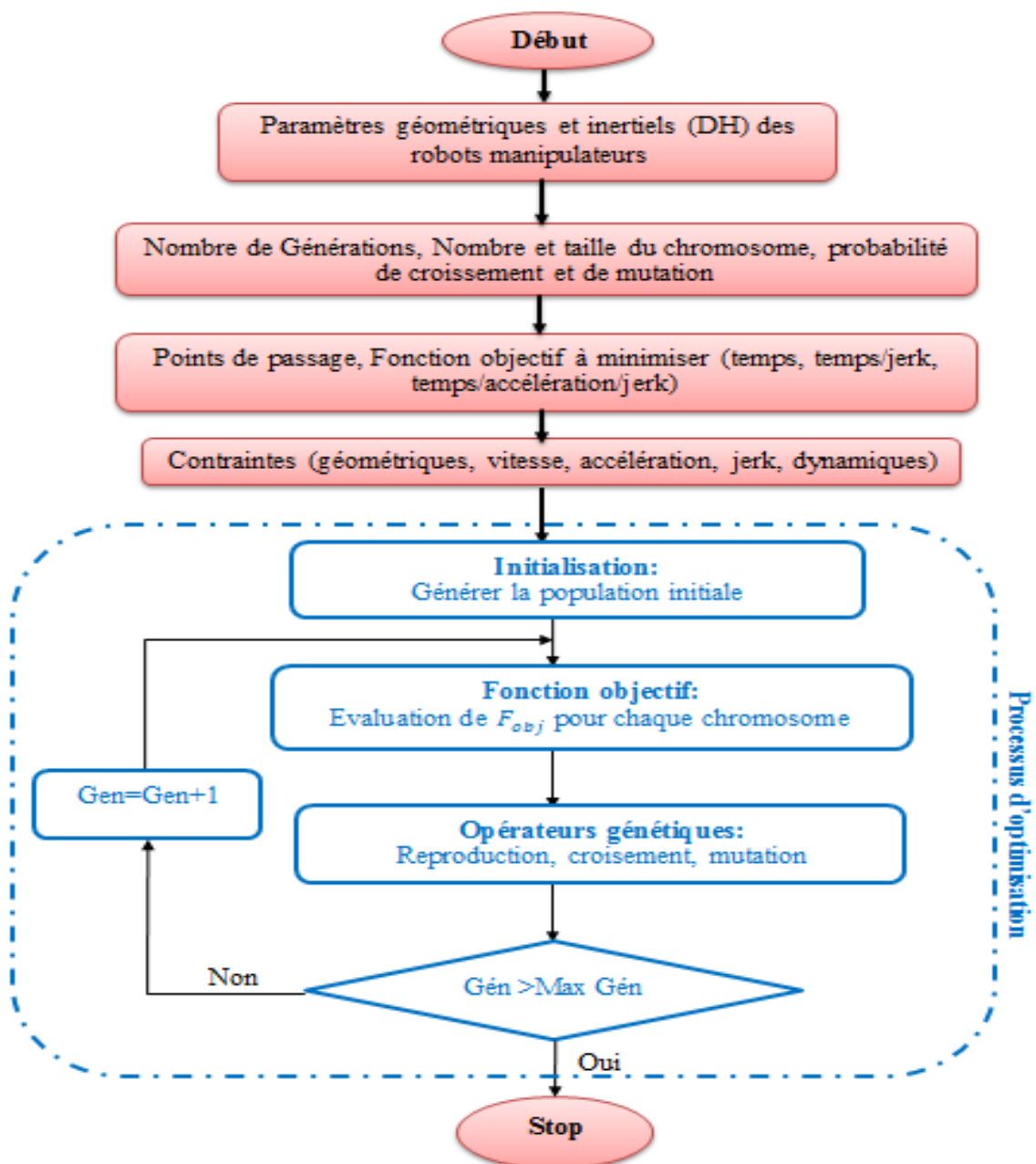


Figure 2.4: Organigramme des Algorithmes Génétiques dans les tâches imposées

2.5. Planification des trajectoires optimales dans les tâches libres :

Appelée aussi mouvement point à point ou mouvement de transfert. Dans ce genre de mouvement, seules les configurations initiales et finales 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.

2.5.1. Présentation de l'approche :

Dans ce genre de tâche nous allons présenter l'approche de résolution qu'on a adoptée pour résoudre le problème de planification de trajectoire libre des robots manipulateurs.

En effet, cette approche a été proposée par [78] pour traiter ce problème pour les robots manipulateurs à base fixe, elle consiste à transformer le problème de temps final inconnu (difficile à résoudre) en un problème équivalent de bornes définies (plus facile à résoudre), cette approche a par la suite évoluée et développée dans les travaux de [43] pour prendre en compte d'autres 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, 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 [43] en exploitant les avantages qu'elle offre pour trouver des meilleures solutions à des problèmes de planification de trajectoires pour les robots manipulateurs en minimisant d'autres fonctions objectifs et en utilisant d'autres fonctions d'interpolation plus une technique d'optimisation adéquate.

Les inconnus d'un problème de planification de trajectoire dans les tâches point à point 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 inconnus 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.

3.5.1.1. Normalisation de l'échelle du temps :

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 bornes indéfinies $[0, T^{best}]$ vers un autre équivalent dont les bornes de recherche sont complètement connues $[0, 1]$ et qui sera alors plus simple à résoudre.

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 :

$$q(t) = Q(\xi(t)) = Q(\xi) \circ \xi(t) \quad (2.20)$$

Où $\xi(t) = t / T$ 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 appliquant la normalisation de l'échelle du temps (2.20) pour une trajectoire quelconque $q(t)$, les équations des vitesses articulaires $\dot{q}(t)$, des accélérations articulaires $\ddot{q}(t)$ et du jerk $\dddot{q}(t)$ peuvent être écrites comme suit :

$$\begin{cases} \dot{q}(t) = \frac{1}{T} Q'(\xi), \\ \ddot{q}(t) = \frac{1}{T^2} Q''(\xi), \\ \dddot{q}(t) = \frac{1}{T^3} Q'''(\xi) \end{cases} \quad (2.21)$$

3.5.1.2. Décomposition & modélisation des fonctions :

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 :

$$Q(\xi) = P(\lambda) \circ \lambda(\xi) \quad (2.22)$$

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 $\lambda = 1$.

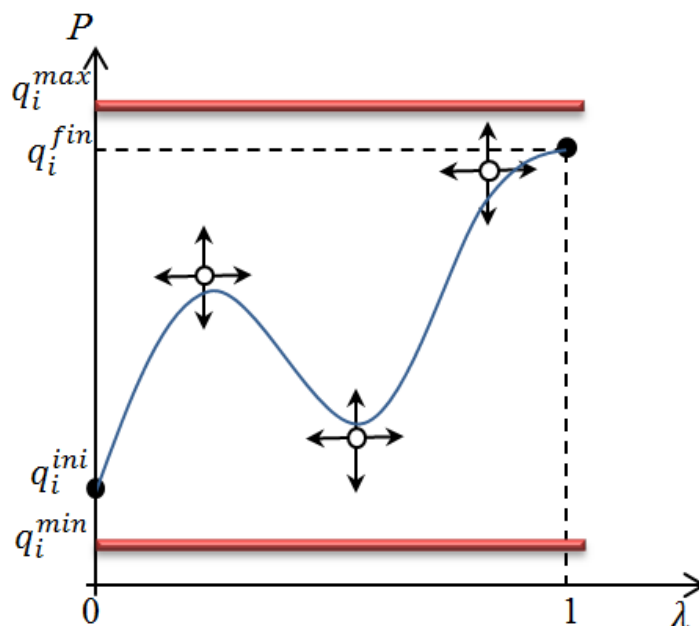


Figure 2.5: Représentation du profil du chemin avec N_c+2 points de contrôle

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 :

$$P(\lambda = 0) = q^{ini} \quad \text{et} \quad P(\lambda = 1) = q^{fin} \quad (2.23a)$$

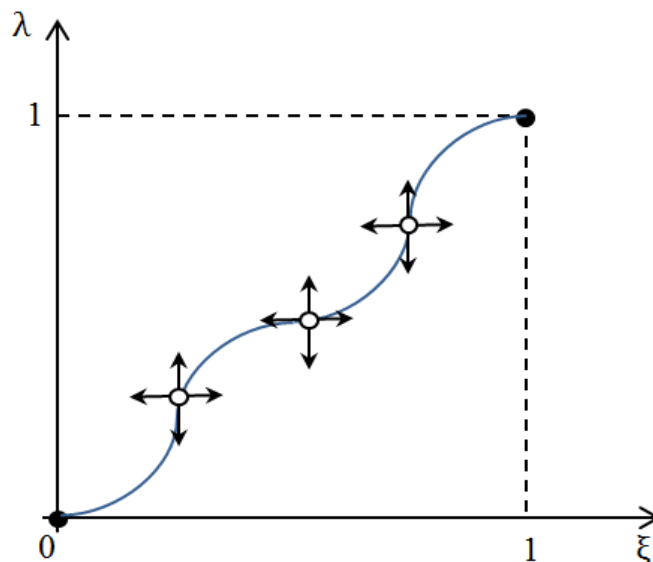


Figure 2.6: Représentation du profil de mouvement avec N_m+2 points de contrôle

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 (2.23a). Tandis que, la fonction mouvement $\lambda(\xi)$ doit satisfaire, d'une part les conditions suivantes :

$$\lambda(\xi = 0) = 0 \quad \text{et} \quad \lambda(\xi = 1) = 1 \quad (2.23b)$$

$$\lambda'(\xi) \geq 0 \quad (2.23c)$$

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

$$\lambda'(\xi = 0) = \lambda'(\xi = 1) = 0 \quad (2.23d)$$

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

3.5.1.3. Traitement des contraintes :

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 (2.20).

3.5.1.3.1. Les contraintes cinématiques :

En appliquant la normalisation de l'échelle du temps (2.20) 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 :

$$\dot{q}(t) = \frac{\partial \xi}{\partial t} \frac{\partial Q(\xi)}{\partial \xi} = \frac{1}{T} Q'(\xi) \quad (2.24a)$$

$$\ddot{q}(t) = \frac{\partial \xi}{\partial t} \frac{\partial Q'(\xi)}{\partial \xi} = \frac{1}{T^2} Q''(\xi) \quad (2.24b)$$

$$\ddot{\ddot{q}}(t) = \frac{\partial \xi}{\partial t} \frac{\partial Q''(\xi)}{\partial \xi} = \frac{1}{T^3} Q'''(\xi) \quad (2.24c)$$

Ce qui permet de reformuler les contraintes (1.19), (1.20) et (1.21) 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 :
$$T \geq T_v \quad (2.25a)$$

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 :

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

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

Soit :
$$T \geq T_a \quad (2.25b)$$

Avec :

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

- Contraintes sur le jerk :

$$\begin{aligned} \forall t \in [0, T] \quad |\dddot{q}_i(t)| &\leq \dddot{q}_i^{max}, \quad i = 1, \dots, n \\ \Rightarrow \forall \xi \in [0, 1] \quad \frac{1}{T^3} |Q'''_i(\xi)| &\leq \dddot{q}_i^{max}, \quad i = 1, \dots, n \end{aligned}$$

$$\Rightarrow T \geq \max_{i=1,\dots,n} \left[\max_{\xi \in [0,1]} \left(\frac{|Q'''_i(\xi)|}{\dddot{q}_i^{max}} \right)^{1/3} \right]$$

Soit :
$$T \geq T_j \quad (2.25c)$$

Avec :

$$T_j = \max_{i=1, \dots, n} \left[\max_{\xi \in [0,1]} \left(\frac{|Q_i'''(\xi)|}{\ddot{q}_i^{max}} \right)^{1/3} \right]$$

Les trois 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 :

$$T_Q \geq T^* \quad (2.26)$$

Avec :

$$T^* = \max_{i=1, \dots, n} \left(\max_{\xi \in [0,1]} \left[\left(\frac{|Q_i'(\xi)|}{\dot{q}_i^{max}} \right), \left(\frac{|Q_i''(\xi)|}{\ddot{q}_i^{max}} \right)^{1/2}, \left(\frac{|Q_i'''(\xi)|}{\ddot{q}_i^{max}} \right)^{1/3} \right] \right)$$

3.5.1.3.2. 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 [79] :

$$\tau_i(t) = \sum_{j=1}^n M_{ij}(q(t)) \ddot{q}_j(t) + C_i(q(t), \dot{q}(t)) + G_i(q(t)) \quad (2.27)$$

En utilisant l'équation (2.20) et les équations des vitesses et des accélérations généralisées (2.24) dans les équations du modèle dynamique, on obtient :

$$\tau_i(\xi) = \frac{1}{T^2} \sum_{j=1}^n M_{ij}(Q(\xi)) Q''_j(\xi) + \frac{1}{T^2} C_i(Q(\xi), Q'(\xi)) + G_i(Q(\xi)) \quad (2.28)$$

Si on pose :

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

L'équation (2.28) devient :

$$\tau_i(\xi) = \frac{1}{T^2} w_i(\xi) + G_i(Q(\xi)) \quad (2.29)$$

Pour simplifier encore les calculs numériques, on divise l'équation (2.29) par τ_i^{max} , et on pose :

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

L'équation (2.29) devient :

$$\xi \in [0, 1] : \bar{\tau}_i(\xi) = \frac{1}{T^2} \bar{w}_i(\xi) + \bar{G}_i(\xi) \quad (2.30)$$

Ce qui permet de réécrire les contraintes (1.22), liées aux capacités en couple des actionneurs de la manière suivante :

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

Cette double inéquation peut être réécrite comme suit :

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

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 :

$$T = [T_{L_i}, T_{R_i}] \quad (2.31)$$

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 $\bar{w}_i(\xi)$, A et B , sont résumées dans le tableau 2.1.

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 en respectant les contraintes dynamiques dues aux couples :

$$T \in [T_g, T_d] \quad (2.32)$$

Avec :

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

Tableau 2.1: Intervalles admissibles de T liés aux contraintes couples du système

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

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

$$I_{ad} = [T^*, \infty[\cap [T_g, T_d] = [I_{ad}^{inf}, I_{ad}^{sup}] \quad (2.33)$$

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.

3.5.1.4. 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.

3.5.1.4.1. Fonction objectifs temps & jerk [80]:

La fonction objectif dans notre cas représente une pondération entre le temps de transfert de la tâche et le Jerk, sa formule s'écrit :

$$J_Q = \alpha T_f + (1-\alpha) \int_0^{T_f} \sum_{i=1}^n (\ddot{q}_i(t))^2 dt \quad (2.34)$$

α est un facteur de pondération compris entre 0 et 1 selon les besoins de l'utilisateur qui peut favoriser soit le temps d'exécution de la tâche soit le jerk. En

introduisant l'équation (2.24c) dans (2.34) nous obtiendrons pour un profil de trajectoire Q donné, l'expression de la fonction objectif J_Q sous la forme suivante :

$$J_Q = \alpha T_f + (1-\alpha) \int_0^1 \sum_{i=1}^n \left(\frac{1}{T_f^3} Q_i'''(\xi) \right)^2 d\xi \quad (2.35)$$

Avec :

$$Q_i'(\xi) = \frac{dQ}{d\lambda} \frac{d\lambda}{d\xi}$$

$$Q_i''(\xi) = \frac{d^2Q}{d\lambda^2} \left(\frac{d\lambda}{d\xi} \right)^2 + \frac{dQ}{d\lambda} \frac{d^2\lambda}{d\xi^2}$$

$$Q_i'''(\xi) = \frac{d^3Q}{d\lambda^3} \left(\frac{d\lambda}{d\xi} \right)^3 + 3 \frac{d^2Q}{d\lambda^2} \frac{d^2\lambda}{d\xi^2} \frac{d\lambda}{d\xi} + \frac{dQ}{d\lambda} \frac{d^3\lambda}{d\xi^3}$$

Le temps de déplacement T_m qui minimise la fonction objectif J_Q pour un profil de trajectoire $Q_i(\xi)$ est :

$$T_m = \sqrt[7]{6 \frac{S_2}{S_1}} \quad (2.36)$$

$$\text{Avec : } S_1 = \alpha; \quad \text{et : } S_2 = (1-\alpha) \int_0^1 \sum_{i=1}^n (Q_i'''(\xi))^2 d\xi$$

3.5.1.4.2. Fonction objectifs temps, couple moteur et énergie [43]:

La fonction objectif dans ce cas représente une pondération entre le temps de transfert, les efforts des actionneurs et l'énergie consommée, elle peut s'écrire sous la formule suivante :

$$J(T) = T \left(\alpha + (1-\alpha) \left(\beta \sum_{i=1}^n \left(\frac{\tau_i}{\tau_{i,\max}} \right)^2 + (1-\beta) \sum_{i=1}^n \left(\frac{\tau_i \cdot \dot{q}}{\tau_{i,\max} \cdot \dot{q}_{\max}} \right)^2 \right) \right) \quad (2.37)$$

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

Pour obtenir l'expression de la fonction objectif J_Q , uniquement en fonction du temps pour un profil Q donné, on remplace les équations (2.24) et (2.30) dans (2.37) et nous obtiendrons la formule suivante :

$$J_Q(T) = T \left(\alpha + (1-\alpha) \left(\beta \left(E_0 + \frac{E_1}{T^2} + \frac{E_2}{T^4} \right) + (1-\beta) \left(\frac{W_0}{T^2} + \frac{W_1}{T^4} + \frac{W_2}{T^6} \right) \right) \right) \quad (2.38)$$

Avec :

$$E_0 = \int_0^1 \sum_{i=1}^n \bar{G}_i^2 d\xi; E_1 = 2 \int_0^1 \sum_{i=1}^n \bar{G}_i \bar{h}_i d\xi; E_2 = \int_0^1 \sum_{i=1}^n \bar{h}_i^2 d\xi$$

$$W_0 = \int_0^1 \sum_{i=1}^n \bar{Q}'_i \bar{G}_i^2 d\xi; W_1 = 2 \int_0^1 \sum_{i=1}^n \bar{Q}'_i \bar{G}_i \bar{h}_i d\xi; W_2 = \int_0^1 \sum_{i=1}^n \bar{Q}'_i \bar{h}_i^2 d\xi \text{ et: } \bar{Q}'_i = \frac{Q'_i}{\dot{q}_i^{\max}}$$

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 de la fonction considérée.

Si on annule la dérivée de la fonction (2.38) on obtient l'équation non linéaire suivante :

$$A_3 T^6 - A_2 T^4 - A_1 T^2 - A_0 = 0 \quad (2.39)$$

Avec:

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

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

A_0, A_1, A_2 et A_3 sont des coefficients réels indépendants du temps de transfert T .

2.5.2. Optimisation par les algorithmes génétiques dans les tâches libres :

Pour chercher la trajectoire optimale, on doit générer aléatoirement selon une technique d'optimisation des algorithmes génétiques (voir section 3.4.4) un profil de chemin et un profil de mouvement, qui vont nous donner par la suite un profil de trajectoire candidat. Ce dernier sera par la suite évalué et comparé avec d'autres profils. Cette opération est répétée pour tous les chromosomes introduits et le meilleur résultat 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 cinématiques et/ou dynamiques seront automatiquement rejetés.

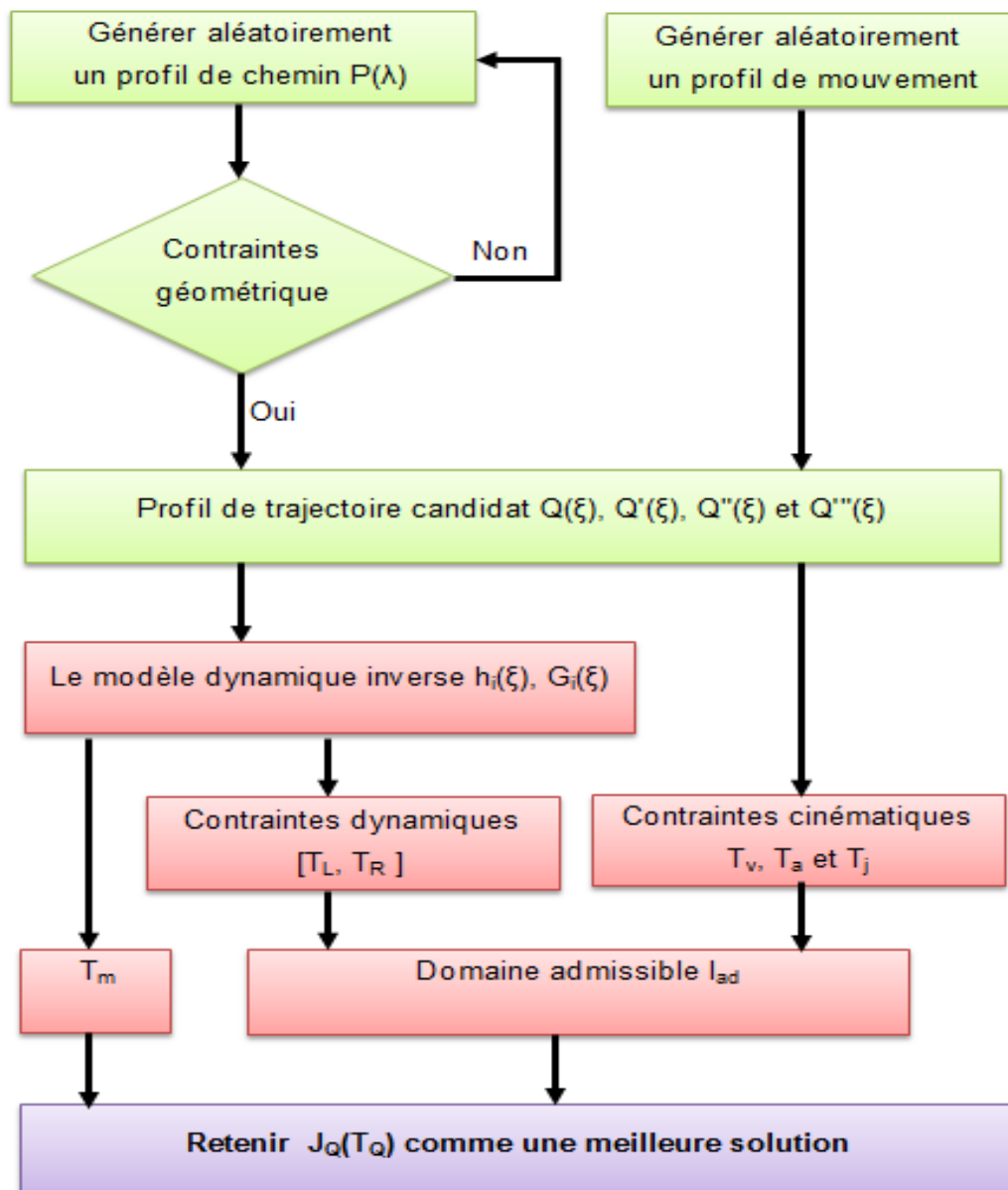


Figure 2.7: Organigramme des Algorithmes Génétiques utilisé dans les tâches libres

2.6. Conclusion :

Dans ce chapitre, nous avons présenté deux approches différentes pour la planification des trajectoires optimales, l'une pour les tâches imposées et l'autre pour les tâches point à point (libre). La première approche est basée sur la

résolution du problème en utilisant un vecteur des intervalles de temps qui est introduit dans la fonction objectif afin de la minimiser sous les différentes contraintes imposées sur la structure polyarticulée. La deuxième approche repose sur la normalisation de l'échelle du temps, ensuite la décomposition de la fonction de la trajectoire en deux fonctions (chemin & mouvement) pour mieux traiter les contraintes selon leurs difficultés. Ces deux approches nécessitent la modélisation des profils pour cela nous avons opté pour les fonctions splines cubiques qui offrent un avantage indéniable par leurs propriétés pour traiter ce problème. Finalement, pour que ces approches soient finalisées elles nécessitent 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.

Dans le prochain chapitre, nous allons essayer d'appliquer les étapes décrites précédemment pour trouver les trajectoires optimales pour des robots manipulateurs qui exécutent des tâches imposées et libres.

3. RESULTATS ET ANALYSE

3.1. Introduction :

Ce chapitre concerne l'application des méthodes de planification de trajectoires prescrites précédemment dans les tâches imposées et les tâches point à point sur des manipulateurs robotiques en tenant compte de la fonction objectif et/ou de la fonction multi-objectifs à minimiser. Plusieurs exemples sont discutés et comparés pour divers manipulateurs robotiques dont les caractéristiques mécaniques sont celles rapportées dans les références.

3.2. Trajectoires optimales dans les tâches imposées :

3.2.1. Minimisation du temps d'exécution :

Le robot planaire à deux degrés de liberté rotoïde et le robot manipulateur à six degrés de liberté sont considérés, la tâche est de déplacer les robots dans un espace de travail le long d'un chemin spécifié tout en minimisant une fonction objectif. Le modèle dynamique des robots est obtenu en utilisant le formalisme d'énergie de Lagrange. Les équations du mouvement comprennent les termes d'inertie des actionneurs, les vecteurs de force de Coriolis et centrifuge et les efforts gravitationnels.

La fonction objectif présente l'intervalle de temps h_i entre deux nœuds de la trajectoire et le problème d'optimisation est définie comme suit :

$$\text{Minimiser :} \quad F_{obj} = \sum_{i=1}^{n-1} h_i \quad (3.1)$$

Tel que :

- Vitesses articulaire : $|V_{ji}(t)| \leq V_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$

- Accélérations articulaires : $|W_{ji}(t)| \leq W_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$
- Jerks : $|J_{ji}(t)| \leq J_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$
- Vitesses linéaire : $|cv_{ji}(t)| \leq cv_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$
- Vitesses angulaire : $|av_{ji}(t)| \leq av_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$
- Coupes moteur : $|T_{ji}(t)| \leq T_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$

Avec V_j^{max} , W_j^{max} , J_j^{max} , cv_j^{max} et av_j^{max} représentent respectivement les contraintes cinématiques des vitesses, accélérations, jerks, vitesse linéaire et vitesse angulaire du lien j -th défini à partir des données technologiques et de conception des robots.

4.2.1.1. Exemple1 : robot SCARA 2R planaire :

Le robot considéré dans cet exemple est planaire à deux degrés de liberté (Figure 3.1). Il s'agit de planifier des trajectoires sous des contraintes de couples et de vitesses dans l'espace cartésien ; les paramètres inertiels et dynamiques sont résumés dans le tableau 3.1 et les performances cinématiques limites sont les vitesses linéaires et angulaires de l'effecteur, définies par les relations (3.2) et (3.3) successivement. Le robot est invité de se déplacer à travers des points de passage imposés dans l'espace cartésien. En résolvant le modèle cinématique inverse du robot, les points de passage cartésiens sont convertis en points intermédiaires articulaires qui sont rapportés dans les deux premières colonnes du tableau 3.2. De plus, nous considérons le bras au repos dans les positions initiale et finale ($V_i^{init} = V_i^{fin} = 0$; $W_i^{init} = W_i^{fin} = 0$).

Notons que ce robot 2R planaire a été principalement suggéré pour comparer les résultats obtenus par notre technique avec ceux obtenus par l'algorithme génétique / intervalle utilisé dans [81]. Le but est de prouver que notre approche peut être utilement utilisée pour des problèmes complexes et qui présentent des contraintes kinodynamiques.

$$\|cv\| = \sqrt{l_1^2 V_1^2 + l_2^2 (V_1 + V_2)^2 + 2l_1 l_2 V_1 (V_1 + V_2) \cos(Q_2)} \leq 0.7 \text{ m/s} \quad (3.2)$$

$$av_z = V_1 + V_2 \leq 1.5 \text{ rd/s} \quad (3.3)$$

En appliquant la méthode du Lagrangien sur le robot et sous l'hypothèse de masses concentrées à l'extrémité distale de chaque lien, les équations du modèle

dynamique inverse du robot Scara planaire à deux degrés de liberté s'écrivent comme suit :

$$\tau_1 = m_1 l_2^2 (W_1 + W_2) + m_2 l_1 l_2 \cos(Q_2) (2W_1 + W_2) + (m_1 + m_2) l_1^2 W_1 - m_2 l_1 l_2 \sin(Q_2) V_2^2 - 2m_2 l_1 l_2 \sin(Q_2) V_1 V_2 + m_2 l_2 g \cos(Q_1 + Q_2) + (m_1 + m_2) l_1 g \cos(Q_1) \quad (3.4)$$

$$\tau_2 = m_2 l_1 l_2 \cos(Q_2) W_1 + m_2 l_1 l_2 \sin(Q_2) V_1^2 + m_2 l_2 g \cos(Q_1 + Q_2) + m_2 l_2^2 (W_1 + W_2)$$

Tableau 3.1: Paramètres géométriques, inertiels et contraintes dynamique du robot 2R

	Lien 1	Lien 2
m (kg)	15.0	7.0
l (m)	1.0	0.50
Couples (N. m)	260.0	50.0

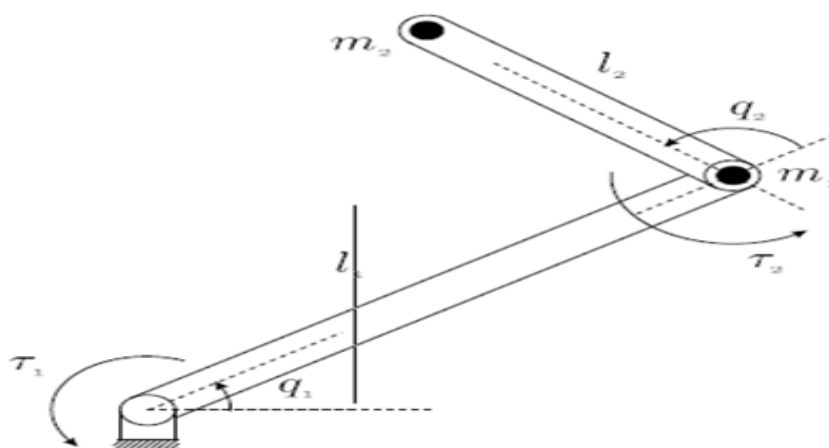


Figure 3.1: Représentation du Robot SCARA 2R planaire

Les résultats des profils de trajectoires modélisées avec les fonctions splines cubiques et leurs dérivées, les vitesses articulaires, les accélérations articulaires et les couples sont montrés dans la figure 3.2-a-d. La tâche imposée a été donnée par dix points de passage représentés par des cercles et deux nœuds supplémentaires représentés par des croix. Ces deux nœuds sont calculés à partir des équations (2.10) et (2.11). Les contraintes dynamiques dues aux couples

moteurs des deux articulations ont été respectées et le couple moteurs du premier bras présente une zone de saturation en fin de tâche mais sans qu'il y ait une violation des commandes imposées.

Tableau 3.2: Tâche imposée au robot 2R à travers des nœuds en coordonnées articulaires en [Rad] et ses équivalences en coordonnées opérationnelles en [m]

	Q₁ [Rad]	Q₂ [Rad]	X[m]	Y[m]	
Nœuds Intermédiaires	1	0.000	-1.5708	1.00	-0.500
	2	Nœuds supplémentaires			
	3	0.1253	-1.6804	1.00	-0.375
	4	0.2517	-1.7594	1.00	-0.250
	5	0.3789	-1.8074	1.00	-0.125
	6	0.5054	-1.8235	1.00	0.000
	7	0.5837	-1.7087	1.05	0.100
	8	0.6119	-1.4581	1.15	0.200
	9	0.4263	-1.1040	1.30	0.100
	10	0.3903	-1.1124	1.30	0.050
	11	Nœuds supplémentaires			
	12	0.3526	-1.1152	1.30	0.000

De plus, la figure 3.2-d montre les vitesses linéaires et angulaires de l'effecteur, les contraintes cinématiques appliquées sur l'effecteur (équation 4.2, 4.3) et les contraintes dynamiques (équation 4.4) appliquées sur les articulations ont été respectées. Il convient de mentionner que cette approche traite toutes les contraintes de manière séquentielle et par ordre de difficulté, les contraintes cinématiques et les contraintes dynamiques en incluant les équations du modèle dynamique du robot, toute trajectoire qui violerait l'une des contraintes kinodynamiques sera automatiquement rejetée. La figure 3.3 montre la trajectoire optimale dans l'espace cartésien (XY) du robot Scara 2R qui effectue une tâche selon les nœuds intermédiaires imposés. Tandis que la figure 3.4 montre le processus de minimisation de la fonction objectif F_{obj} en fonction du nombre de générations considéré du robot 2R.

La minimisation de la fonction objectif évaluée par notre algorithme donne un vecteur d'intervalles de temps $hi = [0.0031, 0.3724, 0.1838, 0.1869, 0.1835, 0.1656, 0.2275, 0.3629, 0.0995, 0.2304, 0.0200]$ qui signifie un temps de

déplacement minimum global estimé $\sum_{i=1}^{11} h_i=2.0356\text{sec}$, la solution trouvée par l'algorithme génétique / intervalle utilisé dans [81] donne un intervalle de temps minimum globale estimé $\sum_{i=1}^{11} h_i = 2.05009\text{sec}$.

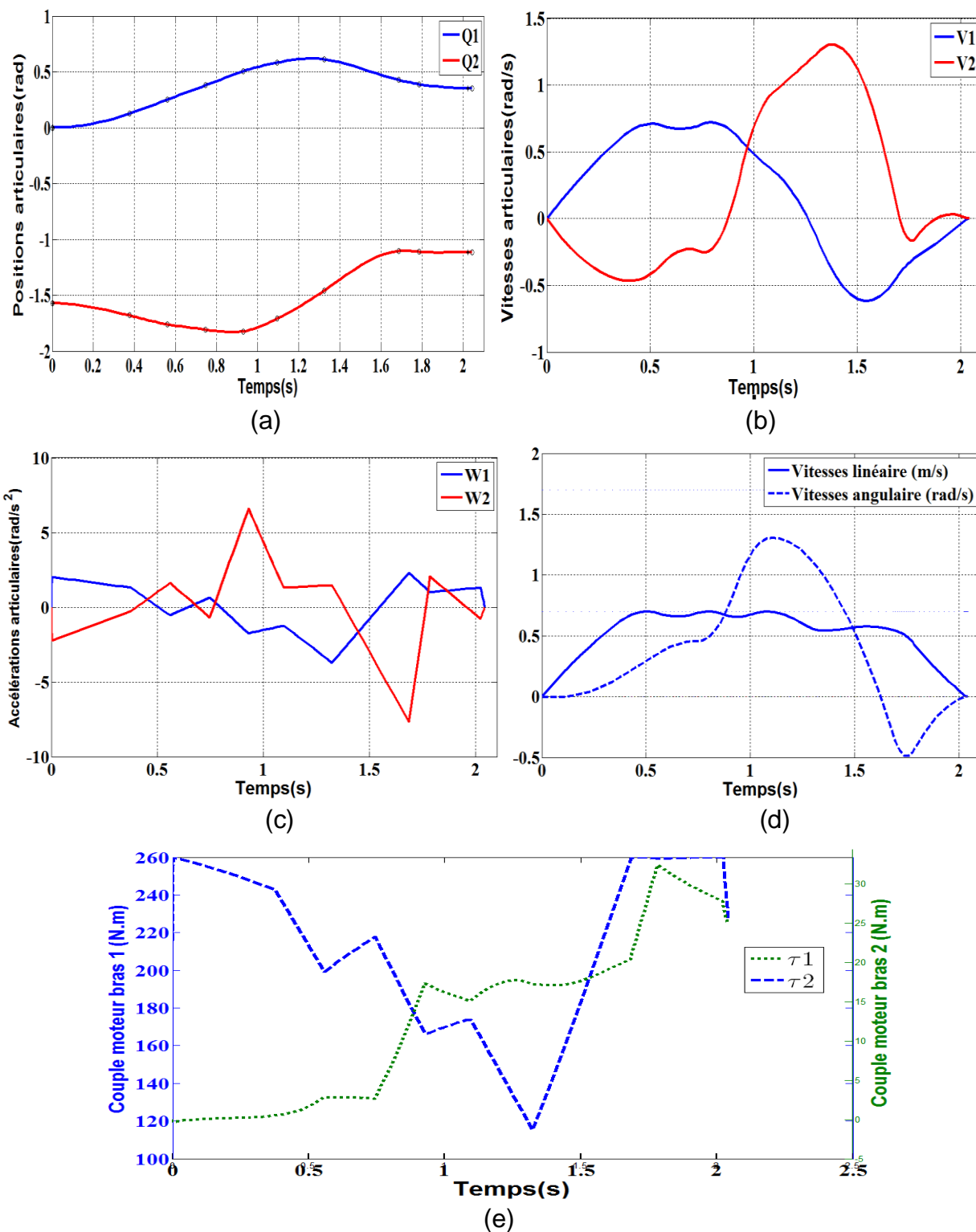


Figure 3.2: Résultats obtenus du robot SCARA 2R

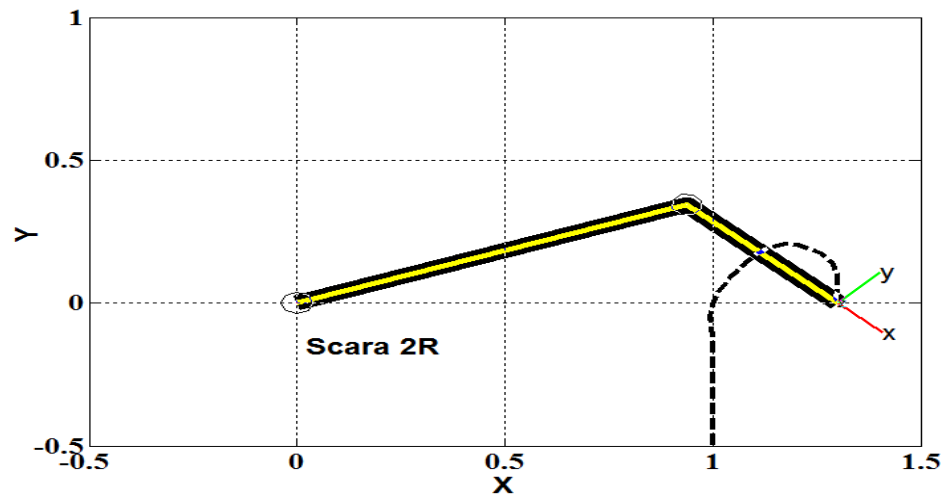


Figure 3.3: Trajectoire optimale du robot Scara 2R dans l'espace Cartésien

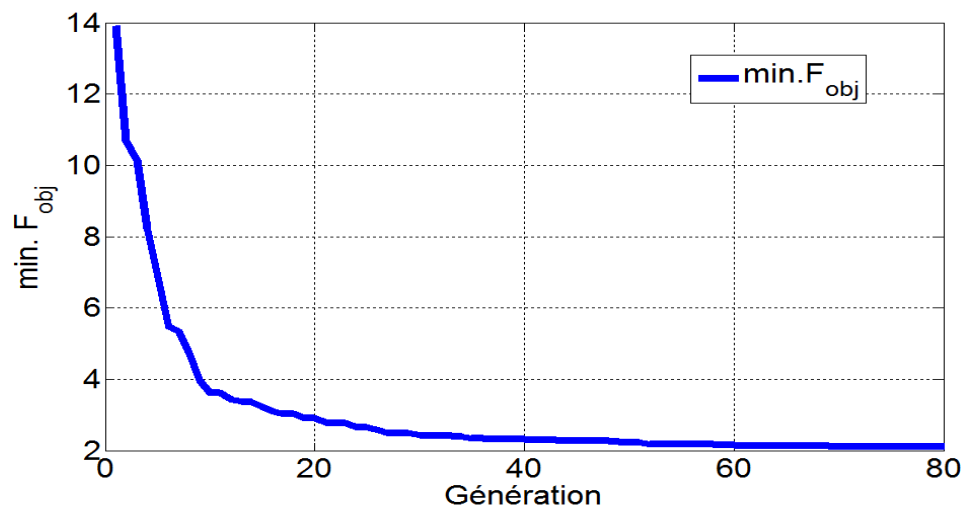


Figure 3.4: Résultats des générations durant le processus d'optimisation du temps d'exécution de la tâche du robot planaire 2R

4.2.1.2. Exemple2 : robot manipulateur 6 D.D.L :

Dans cet exemple, nous traiterons un robot manipulateur à six degrés de liberté dont la structure contient six articulations rotoïdes étudié dans [82]. Ce robot manipulateur est chargé de faire un mouvement entre les positions des nœuds (points de passage) représentés dans le tableau 3.3, cette tâche doit être exécutée sans violation des contraintes cinématiques du système mécanique (tableau 3.4). La fonction objectif prise dans ces simulations est donnée par l'équation (3.1).

Les résultats que nous avons obtenus sont comparés avec deux résultats décrochés l'une par la méthode (SQP) utilisée dans [63] et l'autre par la méthode (HNSA) utilisée dans [82], le but est de prouver que la procédure proposée peut

être utilement utilisée avec des problèmes complexes en présence de toutes les contraintes cinématiques et avec un système polyarticulé qui possède un nombre important de degrés de liberté.

Tableau 3.3: Positions des nœuds (points de passage) pour chaque articulation du robot 6R

Articulation	Points de passage (degrés)					
	1	2	3	4	5	6
1	-10		60	20		55
2	20		50	120		35
3	15	Nœuds	100	-10	Nœuds	30
4	150	supplémentaires	100	40	supplémentaires	10
5	30		110	90		70
6	120		60	100		25

Les résultats trouvés pour les positions articulaires, vitesses articulaires, accélérations articulaires et le jerk après l'exécution du code de calcul ont été représentés dans la figure 3.5. La trajectoire optimale avec un minimum de temps a été donnée par quatre positions de nœuds ou avec cinq intervalles de temps pour l'ensemble des six articulations du robot manipulateur. En appliquant notre approche, nous observons que les limitations technologiques dues aux contraintes cinématiques considérées dans le processus d'optimisation ont été respectées et de manière séquentielle et avec ordre de difficultés : les contraintes de vitesses, d'accélération et de jerk.

Tableau 3.4: Limites technologiques du robot 6R PUMA 560

Articulation	1	2	3	4	5	6
Vitesse (deg/s)	100	95	100	150	130	110
Accélération (deg/s²)	60	60	75	70	90	80
Jerk (deg/s³)	60	66	85	70	75	70

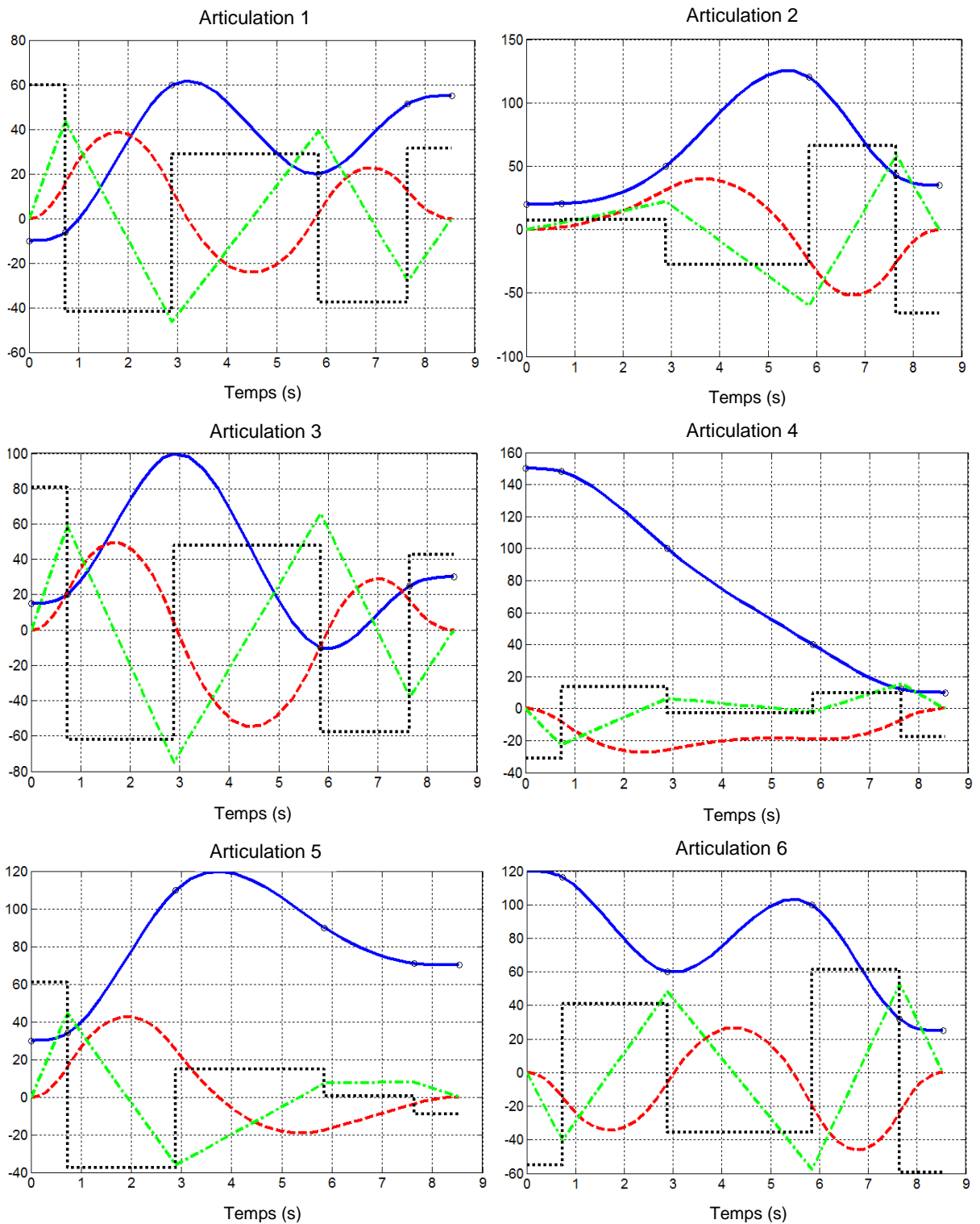


Figure 3.5: Résultats du robot manipulateur 6R sous des contraintes cinématiques

Avec: **Q(deg)** **V(deg/s)** **W(deg/s²)** **J(deg/s³)** représentent respectivement position, vitesse, accélérations et jerk. Les cercles indiquent les positions des nœuds et les croix indiquent les deux nœuds supplémentaires.

La figure 3.6 montre la simulation du robot manipulateur 6R dans l'espace des coordonnées cartésiennes 3D (XYZ), où la trajectoire optimale trouvée pour cette tâche imposée est décrite entre les points de passage 1-3-4-6. Notons qu'au cours du processus d'optimisation, nous avons fixé le nombre de génération égale à 80.

Les résultats de la génération sont présentés dans la figure 3.7, ces résultats montrent que notre approche converge rapidement vers la solution optimale de la fonction objectif.

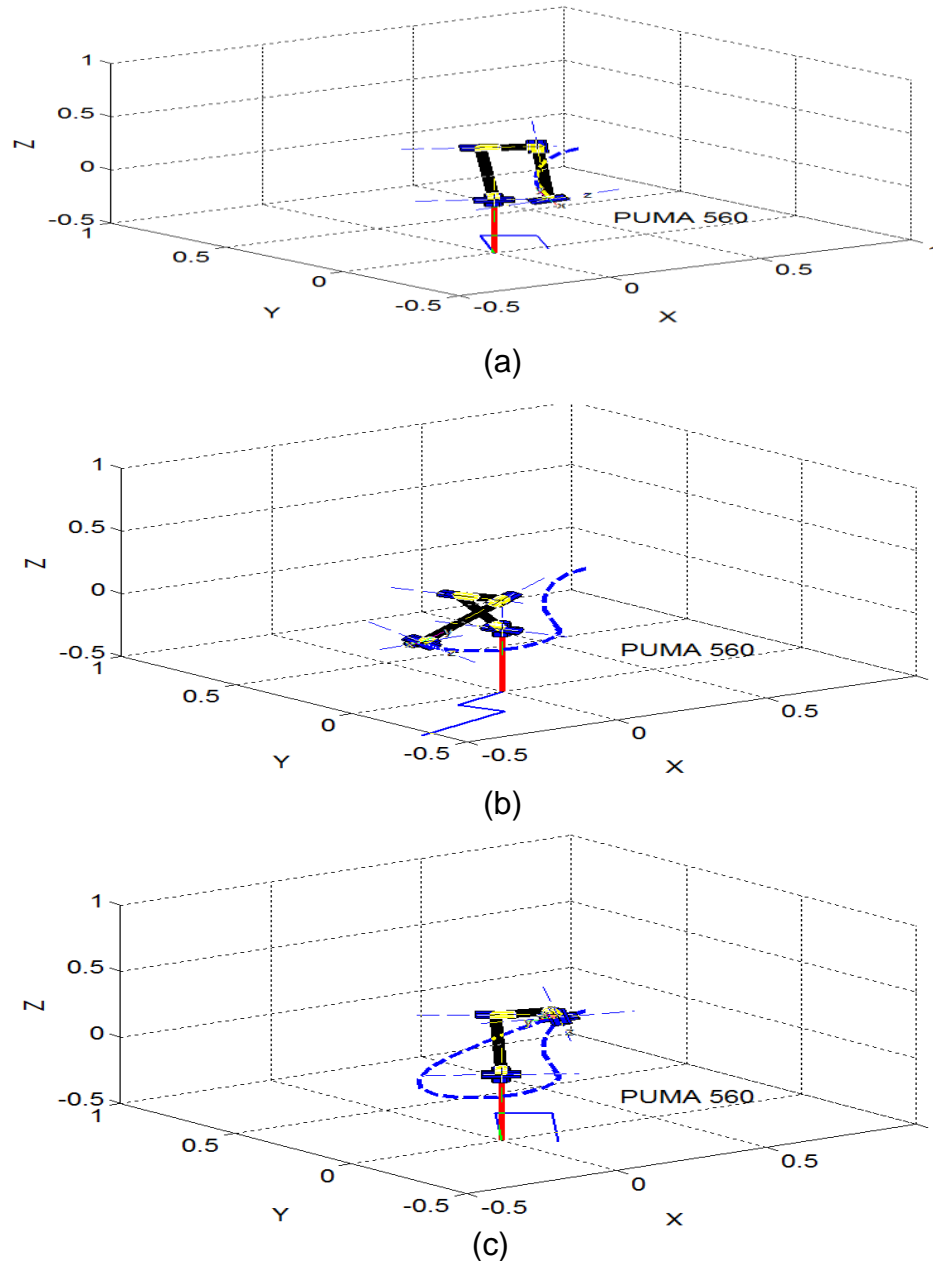


Figure 3.6: Trajectoire optimale du robot manipulateur 6R sous les contraintes cinématiques : (a) entre le nœud 1 et 3, (b) entre le nœud 1 et 4, (c) entre les nœuds 1 et 6

La minimisation de la fonction objectif (3.1) évaluée par notre algorithme donne un vecteur d'intervalle du temps $h = [0.7289, 2.1588, 2.9611, 1.7954, 0.8856]$ ce qui égal à un temps d'exécution de la tâche minimum $\sum_{i=1}^5 h_i = 8.5298 \text{sec}$.

Le temps minimal est la donnée la plus significative pour comparer les techniques utilisées dans la planification des trajectoires optimales des robots industriels pour les tâches spécifiées. Les résultats montrent que l'approche décrite dans ce paragraphe peut obtenir une meilleure solution par rapport aux autres techniques utilisées précédemment, y compris les résultats des auteurs de [63] qui ont utilisé une technique de programmation quadratique séquentielle (SQP) et qui ont obtenu une fonction objectif égale à $\sum_{i=1}^5 h_i = 8.5726 \text{sec}$ et les auteurs de [82] qui ont utilisé un algorithme de recherche dite d'harmonie hybride (HHSA) et qui ont obtenu une fonction objectif égale à $\sum_{i=1}^5 h_i = 8.5310 \text{sec}$.

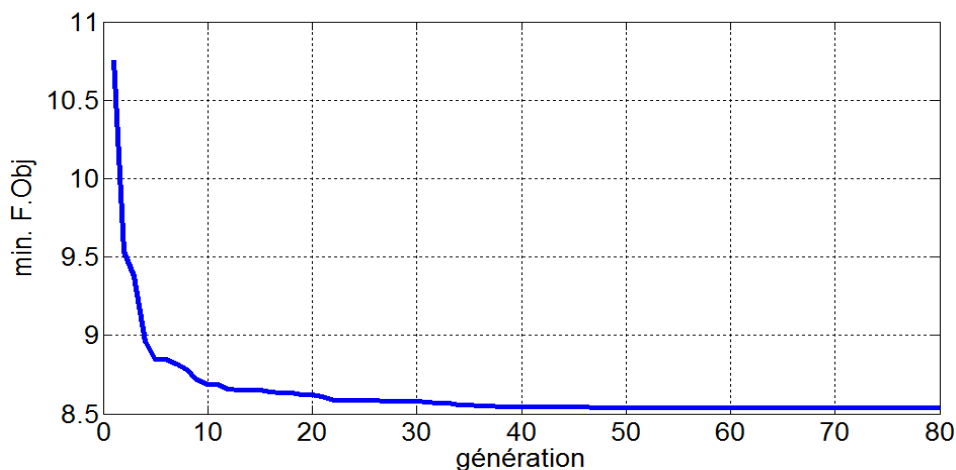


Figure 3.7: Résultats des générations durant le processus d'optimisation du temps d'exécution de la tâche du robot manipulateur 6R

3.2.2. Minimisation temps/jerk du Robot à 3 D.D.L :

L'application numérique a été implémentée sur un robot industriel à trois degrés de liberté de type rotoïde qui est chargé d'une manutention des substrats en verre selon une tâche imposée [83]. Pour trouver la solution quasi-optimale d'une trajectoire optimale en minimisant la fonction objectif (3.5) nous allons utiliser notre méthode proposée.

Le robot manipulateur doit accomplir une tâche en parcourant une trajectoire qui est composée par quatre nœuds ou cinq intervalles de temps ($M = 5$) pour les trois articulations. Le tableau 3.5 montre les positions des points intermédiaires de chaque articulation tandis que le tableau 3.6 montre les contraintes cinématiques liées aux capacités maximales en vitesses, accélération du système polyarticulé.

La fonction objectif combinée F_{obj} est définie comme suit :

$$F_{obj} = \min \left(k_T \sum_{i=1}^N T_i + k_J \sum_{i=1}^N \max_{h_i} |J_i(t)| \right) \quad (3.5)$$

Vérifiant pour tout $t \geq 0$ les contraintes dues aux :

- Vitesses articulaires : $|V_{ji}(t)| \leq V_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$ (3.6a)

- Accélérations articulaires : $|W_{ji}(t)| \leq W_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$ (3.6b)

Tableau 3.5: Positions des nœuds (points de passage) pour chaque articulation du robot 3R

Articulation	Point de passage (degrés)					
	1	2	3	4	5	6
1	120	Nœuds supplémentai res	90	45	Nœuds supplémentai res	0
2	-10		60	40		100
3	0		-20	30		70

Tableau 3.6: Limites technologiques du robot 3R

Articulation	Vitesse (deg/s)	Accélération (deg/s ²)
1	100	70
2	95	75
3	100	75

Pour représenter la trajectoire et résoudre la position des nœuds supplémentaires inconnus indiqués dans le tableau 3.5, nous utilisons les deux équations 3.10 et 3.11. Nous avons calculé le jerk J_{ij} de la trajectoire à partir de

l'équation (3.7), ensuite nous résolvons le problème d'optimisation de la fonction objectif temps-jerk minimum (3.5). Au départ, nous avons créé la population initiale (chromosomes), et pour rechercher la trajectoire optimale, nous devons générer par hasard selon les techniques des algorithmes génétiques des trajectoires. Chaque chromosome candidat sera évalué puis comparé aux autres chromosomes, cette opération est répétée pour tous les chromosomes introduits. Le meilleur résultat est celui qui satisfait la fonction objectif donnée. Il est à noter que toute trajectoire qui violerait l'une des contraintes de vitesse ou d'accélération indiquées dans le tableau 3.6 sera automatiquement rejetée par les équations 4.6a et 4.6b respectivement. Nos résultats ont été comparés et validés par les résultats de la référence [83].

Les trajectoires modélisées par les fonctions splines cubiques incluant leurs dérivées (vitesses, accélérations et jerk) pour $K_J / K_T = 0.02$ et $K_J / K_T = 0.3$ sont illustrées sur la figure 3.8-a et la figure 3.8-b respectivement. Pour le premier cas, nous avons obtenu un vecteur d'intervalles de temps $h_i = [0.6017, 1.6992, 1.4349,$

$1.2623, 1.0895]$ égal au temps d'exécution total de la tâche soit $\sum_{i=1}^5 h_i = 6.0876 \text{sec}$,

pour le second cas nous avons obtenu un vecteur d'intervalles de temps $h_i = [1.0301, 2.9256, 2.3687, 1.9771, 1.4761]$ égal au temps d'exécution total de la

tâche $\sum_{i=1}^5 h_i = 9.777 \text{sec}$. Ces résultats montrent que l'approche décrite dans ce

paragraphe donne une meilleure solution que [83] où le temps d'exécution total de

la tâche égale $\sum_{i=1}^5 h_i = 6.72 \text{sec}$ pour le premier cas et $\sum_{i=1}^5 h_i = 10.262 \text{sec}$ pour le

deuxième cas en utilisant l'algorithme génétique de Fuzzy « Fuzzy genetic algorithm ».

L'historique des résultats du processus d'optimisation de la fonction objectif pour le robot manipulateur 3R chargé de la manutention des substrats en verre pour $K_J / K_T = 0.02$ et $K_J / K_T = 0.3$ est montré sur les figures 3.9-a et 3.9-b successivement. Pour le premier cas, nous avons obtenu une fonction objectif $F_{obj} = 7.1025$ et pour le second cas la fonction objectif $F_{obj} = 12.846$.

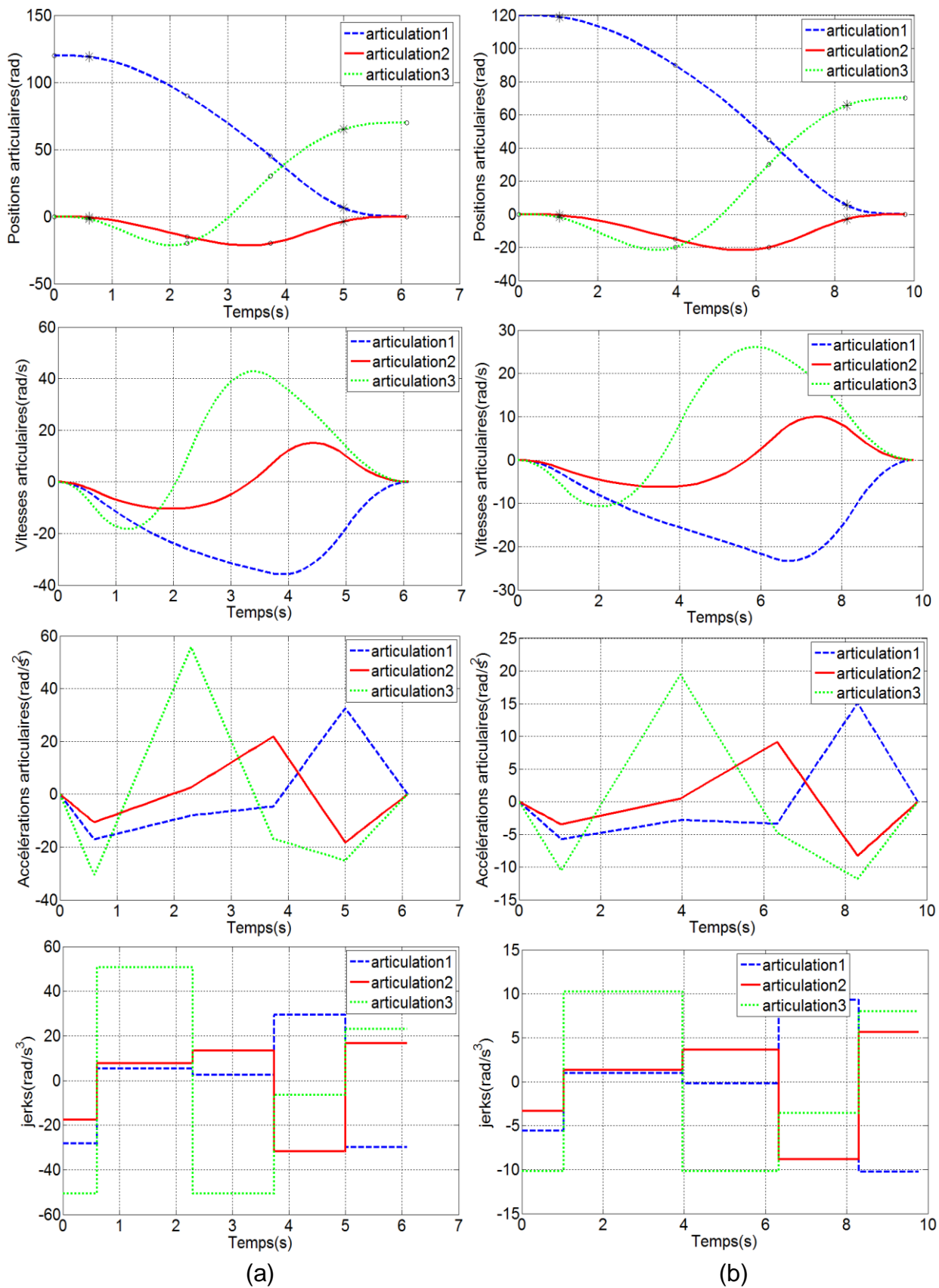


Figure 3.8: Résultats du Robot 3R, selon la tâche assignée pour (a) $K_J / K_T = 0.02$

et (b) pour $K_J / K_T = 0.3$

Ces résultats ont été obtenus avec un nombre fixe de générations égal à 50 ce qui correspondant au $\frac{1}{10}$ du nombre de générations utilisé dans la référence [83] ; ce qui reflète que notre méthode proposée converge rapidement et avec une vitesse de calcul plus rapide.

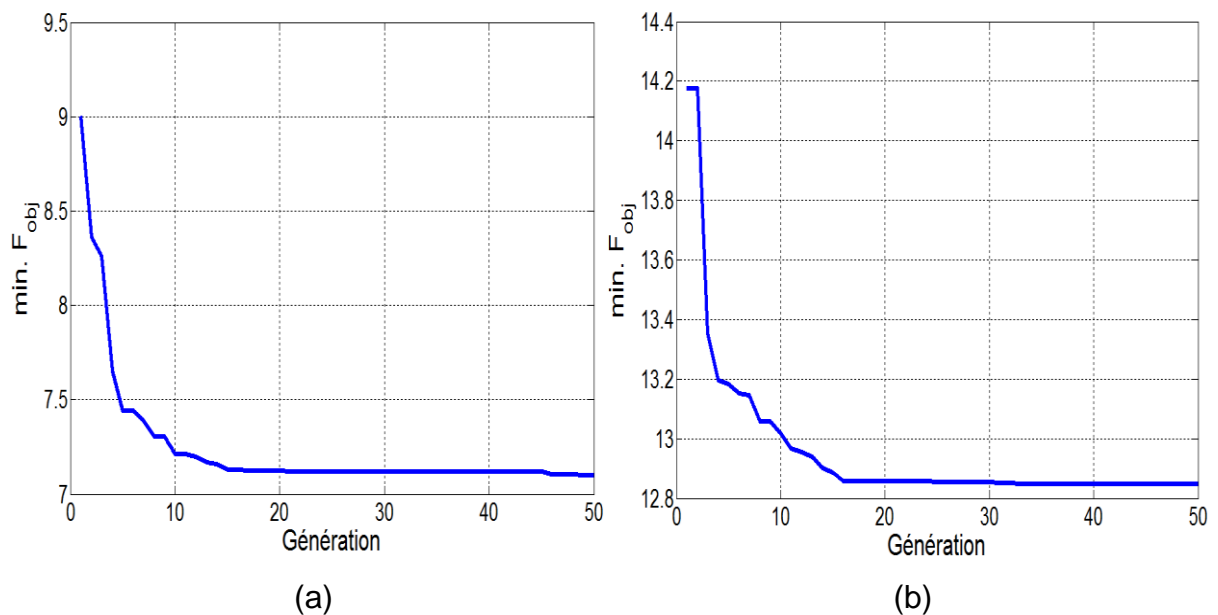


Figure 3.9: Résultats historique du processus d'optimisation de la fonction objectif pour le robot manipulateur 3R pour (a) : $K_J / K_T = 0.02$ et (b) : $K_J / K_T = 0.3$

Le tableau 3.7 montre les valeurs maximales du jerk trouvées par notre approche et avec différentes valeurs de coefficients de pondération K_J / K_T . Ces valeurs sont comparées à celles obtenues par la méthode « *Fuzzy genetic algorithm* » proposée dans la référence [83]. En considérant la valeur de K_J / K_T si elle est supérieure à 0,3 le robot aura plus de temps pour l'exécution de la tâche et si la valeur est inférieure à 0,02 le robot manipulateur subira plus de vibrations dans ses articulations et ceci influent sur les objets transportés au cours de la tâche; donc les valeurs optimales du rapport K_J / K_T sont comprises entre l'intervalle $[0.02, 0.3]$. On peut remarquer que les résultats de notre méthode décrite dans ce paragraphe sont bien comparables à ceux fournis par la méthode de la référence [83] en respectant bien évidemment toutes les contraintes imposées

au système polyarticulé. Les aspects optimaux pour le mouvement du robot rotoïde 3R sont représentés dans la figure 3.10.

Tableau 3.7: Résultats des intervalles de temps optimaux et de jerks maximaux comparés à ceux de [83]

$\frac{K_J}{K_T}$	Méthode Proposée				Méthode de Cong [83]			
	h_i (s)	J_1^{\max} ($^\circ/s^3$)	J_2^{\max} ($^\circ/s^3$)	J_3^{\max} ($^\circ/s^3$)	h_i (s)	J_1^{\max} ($^\circ/s^3$)	J_2^{\max} ($^\circ/s^3$)	J_3^{\max} ($^\circ/s^3$)
0.5	11.483	6.33	5.56	6.32	11.484	5.235	6.467	11.60
0.4	10.284	8.71	7.85	8.74	10.976	5.764	7.152	13.83
0.3	9.777	10.22	8.82	10.23	10.262	6.714	8.399	17.85
0.2	8.665	14.78	12.75	14.80	9.371	8.088	10.21	25.94
0.15	8.389	15.99	14.63	16.11	8.939	9.486	11.97	30.31
0.1	7.433	23.86	18.21	23.86	8.111	12.60	15.97	43.40
0.05	6.884	29.66	23.15	30.08	7.13	17.78	25.46	70.65
0.02	6.087	29.94	31.77	50.74	6.72	27.72	46.01	79.28
0	4.896	79.76	58.89	99.86	6.29	99.21	135.6	103.0

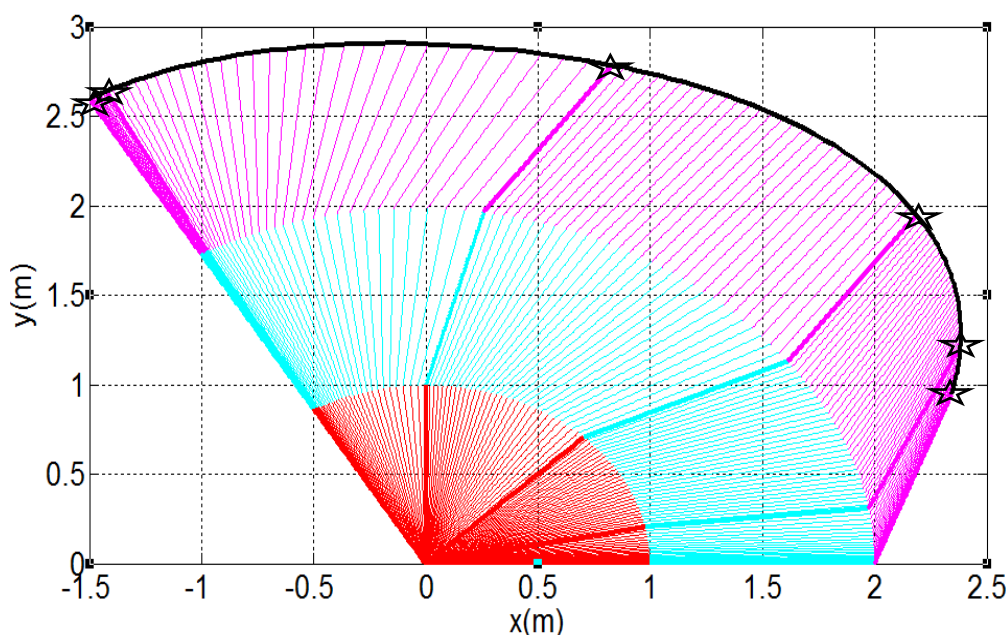


Figure 3.10: Allure du mouvement optimisé du robot industriel à 3R.
(Les étoiles indiquent les différents points de passage de la tâche imposée).

3.2.3. Minimisation du temps/accélérations/jerk du Robot manipulateur de type PUMA 560 avec 6 D.D.L :

Dans cette application nous allons considérer un robot manipulateur avec six articulations rotoïdes de type PUMA 560. Les paramètres géométriques, inertiels et les contraintes dues aux vitesses, accélérations et jerks sont présentées dans [14]. Le robot est initialement au repos et s'arrête complètement à la fin de la trajectoire avec ($V_i^{init} = V_i^{fn} = 0$; $W_i^{init} = W_i^{fn} = 0$), il doit accomplir une tâche imposée à travers dix nœuds de passage entre le point initial et le point final comme indiqué dans le tableau 3.8.

Tableau 3.8:Nœuds de passage pour chaque articulation du robot PMUA560 [14]

Nœuds	Articulations					
	1	2	3	4	5	6
1	10°	10°	10°	5°	10°	6°
2	Nœuds supplémentaires					
3	-130°	-120°	70°	110°	-110°	70°
4	90°	95°	-55°	200°	85°	140°
5	130°	120°	60°	110°	-110°	70°
6	110°	-55°	-55°	20°	100°	-100°
7	100°	-70°	60°	60°	50°	100°
8	-90°	-10°	110°	-100°	-40°	10°
9	180°	-30°	100°	60°	-40°	180°
10	180°	-30°	130°	60°	-40°	180°
11	Nœuds supplémentaires					
12	-100°	100°	120°	-60°	50°	120°

La fonction objectif dans cet exemple représente trois facteurs qui peuvent influencer le fonctionnement du robot manipulateur et son environnement à savoir le temps d'exécution de la tâche, l'accélération et le jerk. A cet effet le problème de planification des trajectoires devient un problème d'optimisation multi-objectifs plus compliqué à résoudre en tenant compte les contraintes technologiques imposées sur le système polyarticulé. Notons que, si les fonctions objectifs sont simplement pondérées et ajoutées pour produire une forme physique simple, la

fonction avec la gamme la plus étendue dominerait l'évolution. Pour éviter cela, toutes les fonctions objectifs sont normalisées pour avoir une même gamme c'est à dire que tous les termes de la fonction multi-objectif sont traités de manière à avoir le même nombre de décimales. Pour notre problème, la fonction objectif combinée F_{obj} est définie comme suit :

$$F_{obj} = \alpha_1 N \sum_{i=1}^{m-1} h_i + \alpha_2 Z \sum_{i=1}^N \sum_{i-1}^{n-1} \left[\frac{(W_{j,i+1} - W_{j,i})^2}{h_i} \right] + \alpha_3 Z \sum_{i=1}^N \sum_{i-1}^{n-1} \left[\frac{(J_{j,i+1} - J_{j,i})^2}{h_i} \right] \quad (3.6)$$

Tel que :

- Vitesses articulaires : $|V_{ji}(t)| \leq V_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$ (3.7a)
- Accélérations articulaires : $|W_{ji}(t)| \leq W_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m$ (3.7b)
- Jerks : $|J_{ji}(t)| \leq J_j^{max}$ pour $j=1, \dots, n$ et $i=1, \dots, m-1$ (3.7c)

α_1 , α_2 et α_3 sont des facteurs de pondération donnés à la fonction objectif selon les besoin de l'utilisateur avec : $\alpha_1 + \alpha_2 + \alpha_3 = 1$. Le paramètre de normalisation de la fonction objectif pour cet exemple est $Z=1/1000$.

Six cas avec différentes valeurs des facteurs de pondération (α_1 , α_2 et α_3) sont considérés pour montrer l'influence de ces paramètres sur le problème de planification des trajectoires optimales en employant l'équation (3.6) sous les contraintes cinématiques (3.7). La figure 3.11 montre les résultats de la planification des trajectoires optimales des positions articulaires. A cet effet deux cas qui concernent la valeur des facteurs de pondération sont présentés, le premier basé sur la minimisation du temps de la tâche avec ($\alpha_1=0.8$, $\alpha_2=0.1$ et $\alpha_3=0.1$) et l'autre sur la minimisation de l'accélération (un temps de transfert maximal) avec ($\alpha_1=0.1$, $\alpha_2=0.8$ et $\alpha_3=0.1$). Ces résultats montrent que l'interpolation par les fonctions splines cubiques permet d'obtenir des allures de trajectoires lisses et stables. Tandis que les résultats des vitesses, accélérations et jerks optimales pour les deux cas précédents sont présentés dans la figure 3.12 et 3.13 respectivement.

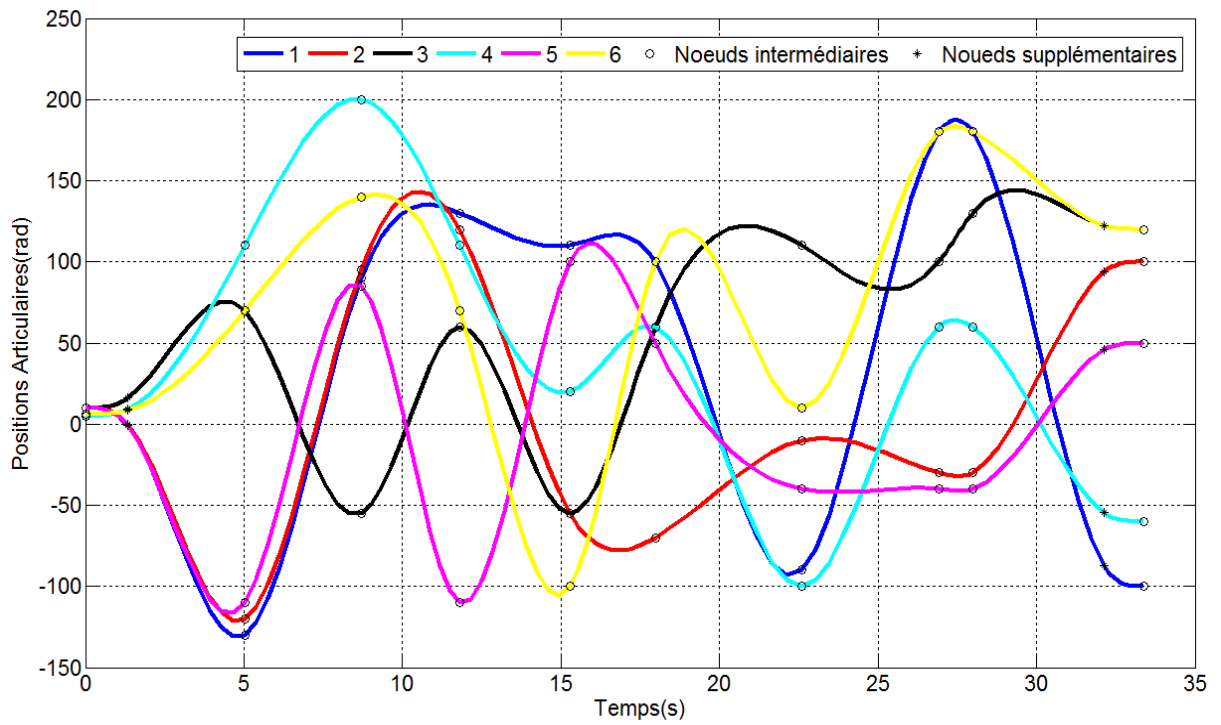
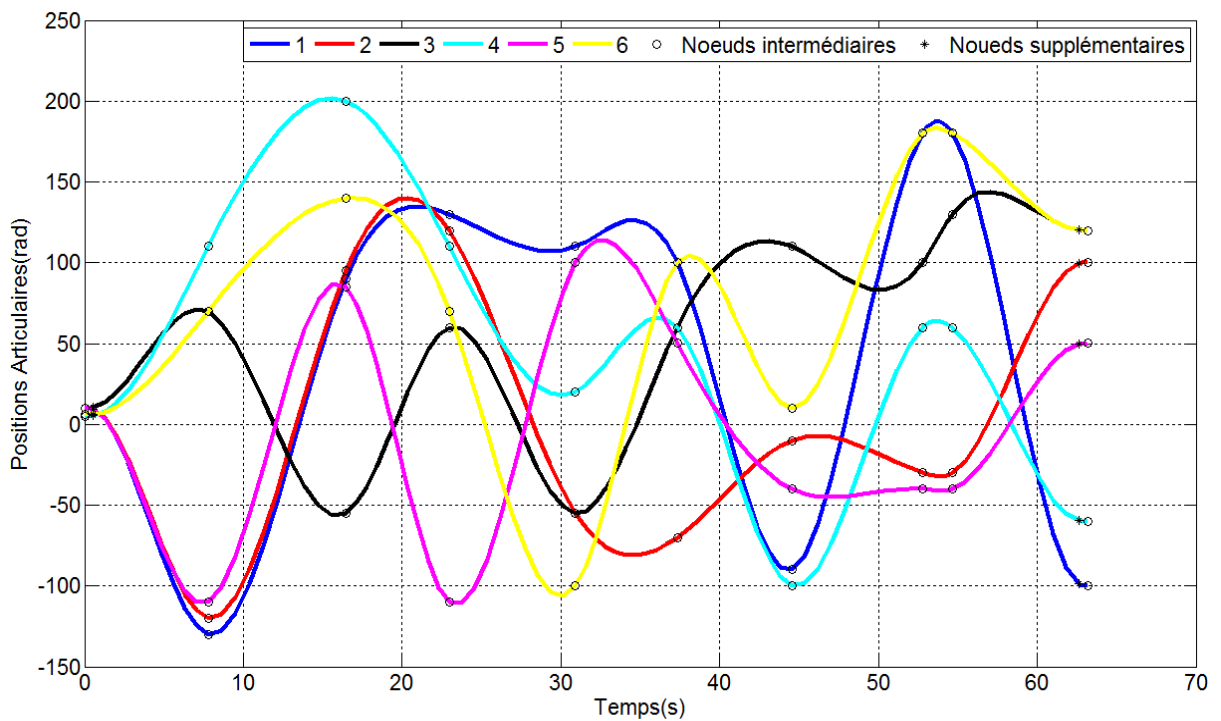
Cas 1 : ($\alpha_1=0.8$, $\alpha_2=0.1$ et $\alpha_3=0.1$)Cas 2 : ($\alpha_1=0.1$, $\alpha_2=0.8$ et $\alpha_3=0.1$)

Figure 3.11: Trajectoires optimales des positions articulaires pour le robot PUMA560- 6R.

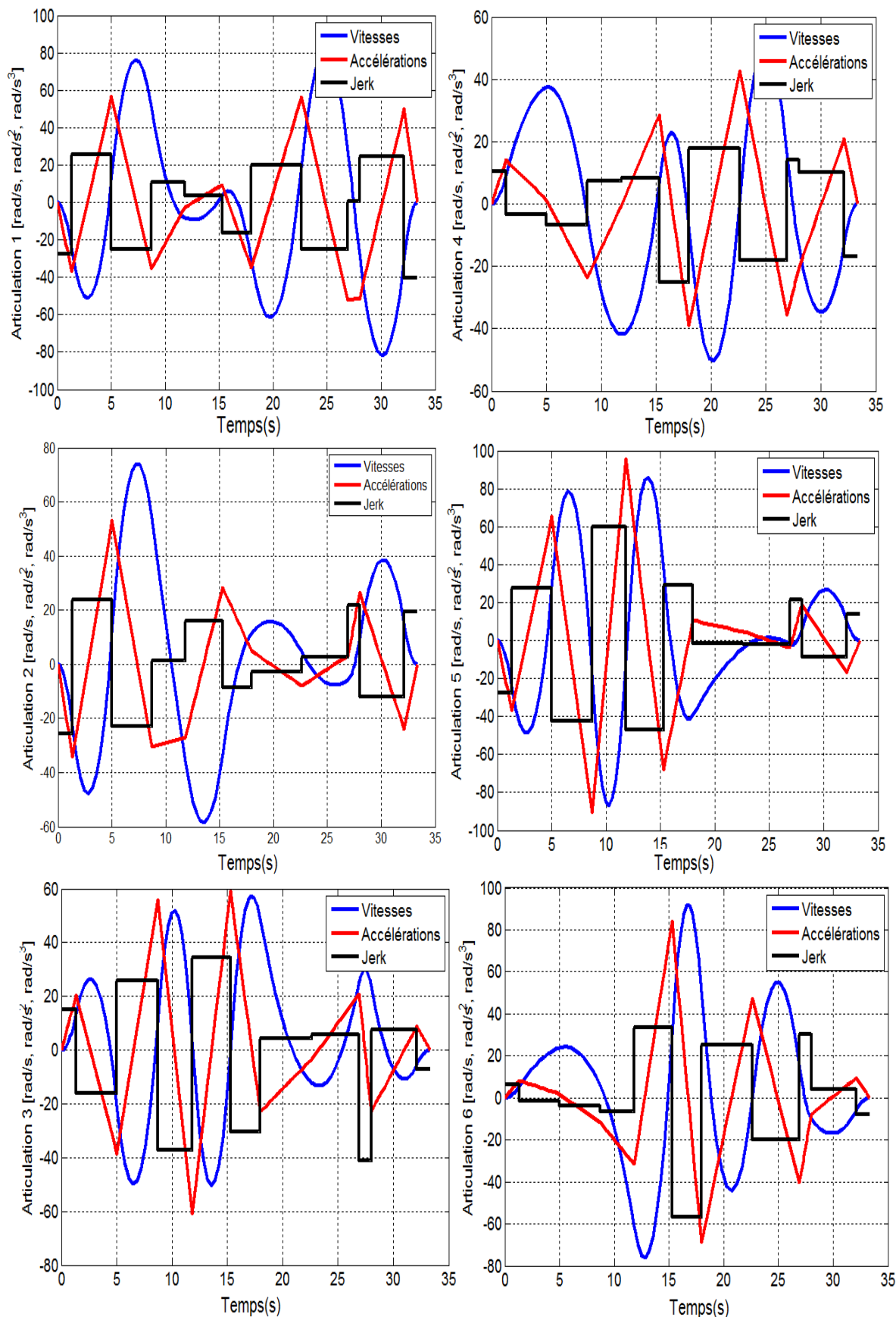


Figure 3.12: Vitesses, accélérations et jerks optimaux pour le robot PUMA 560-6R pour ($\alpha_1=0.8$, $\alpha_2=0.1$ et $\alpha_3=0.1$)

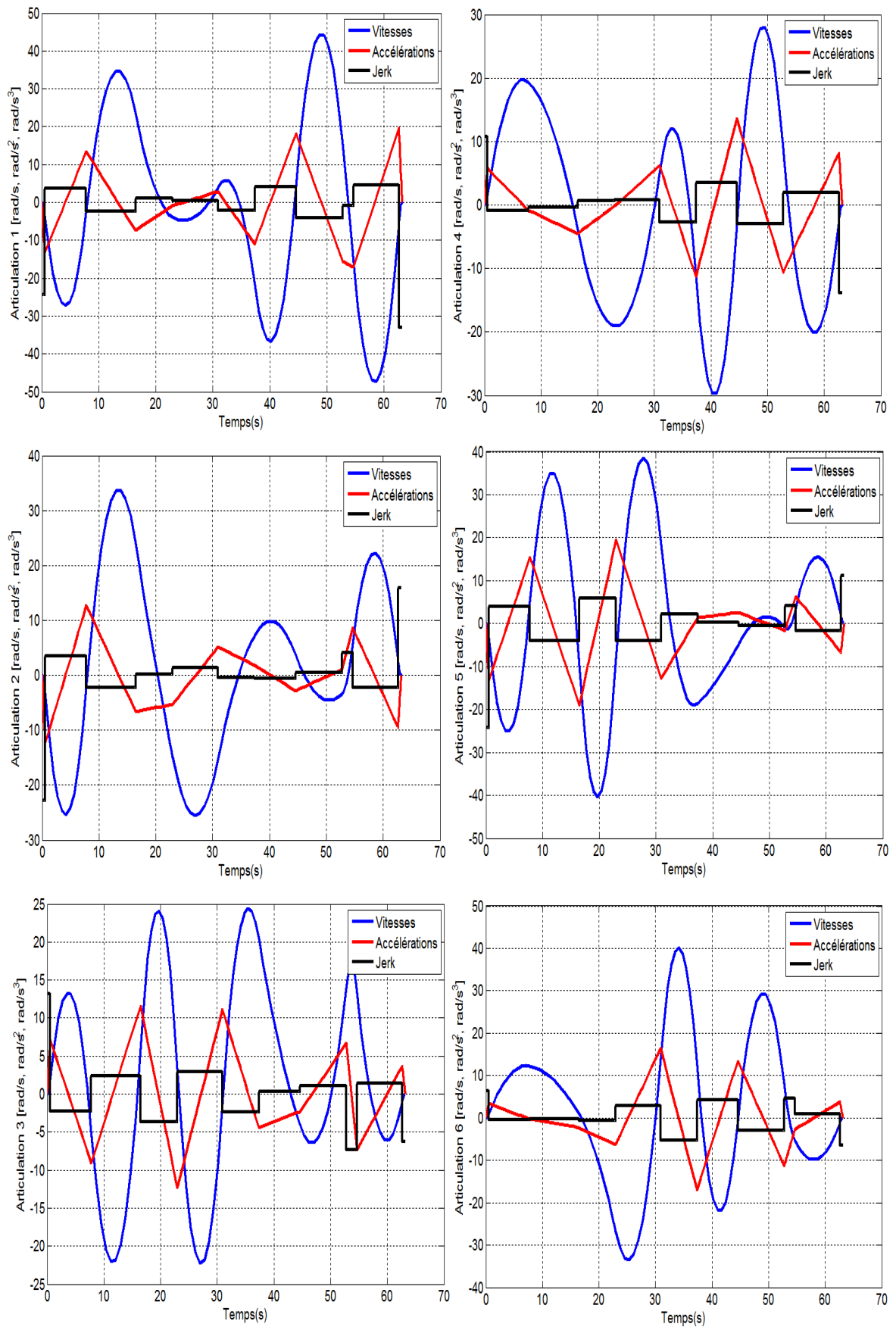


Figure 3.13: Vitesses, accélérations et jerks optimaux pour le robot PUMA 560-6R pour ($\alpha_1=0.1$, $\alpha_2=0.8$ et $\alpha_3=0.1$)

En comparant le cas 1 au cas 2, nous avons trouvé un temps de transfert de la tâche inférieur par contre les allures des accélérations et des jerks sont plus importantes pour toutes les articulations du robot, ceci est dû principalement à l'influence et aux valeurs accordées aux différents facteurs de pondérations. Si le robot manipulateur fonctionne dans un tel état pendant une période de temps excessive, sa durée de vie sera réduite.

Tableau 3.9: Résultats d'optimisation pour le robot PUMA560-6R

Paramètres	Résultats numériques pour $Z_1=1/1000$, $Z_2=1/1000$						
	Initiales	$\alpha_1=0.8$,	$\alpha_1=0.5$,	$\alpha_1=0.5$,	$\alpha_1=0.3$,	$\alpha_1=0.1$,	$\alpha_1=0.1$,
		$\alpha_2=0.1$,	$\alpha_2=0.4$,	$\alpha_2=0.1$,	$\alpha_2=0.3$,	$\alpha_2=0.8$,	$\alpha_2=0.1$,
	$\alpha_3=0.1$	$\alpha_3=0.1$	$\alpha_3=0.4$	$\alpha_3=0.4$	$\alpha_3=0.1$	$\alpha_3=0.8$	
$h_1(s)$	10	1.33	0.68	1.21	1.06	0.53	1.72
$h_2(s)$	10	3.69	4.08	3.90	4.82	7.24	5.76
$h_3(s)$	10	3.67	5.06	4.87	5.85	8.66	7.27
$h_4(s)$	10	3.11	3.80	3.77	4.54	6.52	5.59
$h_5(s)$	10	3.46	4.66	4.50	5.41	7.94	6.66
$h_6(s)$	10	2.68	3.81	3.85	4.50	6.42	5.62
$h_7(s)$	10	4.62	4.20	4.23	4.94	7.22	6.08
$h_8(s)$	10	4.33	4.76	4.45	5.48	8.22	6.75
$h_9(s)$	10	1.06	1.24	1.43	1.57	1.89	2.12
$h_{10}(s)$	10	4.15	4.52	4.22	4.97	7.93	6.01
$h_{11}(s)$	10	1.23	0.83	1.16	1.23	0.59	1.82
$\sum_{i=1}^{m-1} h_i$	110	33.37	37.69	37.65	44.43	63.27	55.71
$\sum_{i=1}^N \sum_{j=i}^{n-1} \left[\frac{(W_{j,i+1} - W_{j,i})^2}{h_i} \right]$	4561	119730	71488	74350	44346	14739	23234
$\sum_{i=1}^N \sum_{j=i}^{n-1} \left[\frac{(J_{j,i+1} - J_{j,i})^2}{h_i} \right]$	397	97987	48145	45218	20165	4828	6532
min F_{obj}	/	181.97	146.48	138.49	101.35	50.19	40.81

Les résultats trouvés en considérant les six cas de facteurs de pondération introduits dans les fonctions objectifs (3.6) sont résumés dans le tableau 3.9. On

remarque que le temps d'exécution augmente progressivement quand α_1 diminue progressivement de 1 à 0. Toutefois nous constatons que le cas avec ($\alpha_1=0.3$, $\alpha_2=0.3$ et $\alpha_3=0.4$) est idéal pour un fonctionnement optimal du robot manipulateur et il n'y a aucune influence sur la structure ni son environnement.

Cependant dans le cas où on donne une grande importance au temps par rapport à l'accélération et le jerk ($\alpha_1=0.8$, $\alpha_2=0.1$ et $\alpha_3=0.1$) le temps d'exécution de la tâche trouvé par [14] est égale 28.51 secondes en utilisant la méthode NSGA-II et 26.35 secondes en utilisant la méthode MODE. Malheureusement les résultats fournis par cette référence ne permettent pas d'effectuer une comparaison complète. En effet, la valeur de la fonction multi-objectif ainsi que les valeurs maximales des vitesses articulaires et de leurs dérivées ne sont pas données. La seule indication disponible est le temps de transfert. Pour notre part, nous avons trouvé une durée de transfert égale à 33.37 secondes ce qui reste acceptable, si on considère que cette tâche est composée de 12 nœuds. Il est à noter aussi que les allures de vitesses, accélérations et jerk sont à l'intérieur de leurs valeurs limites (tableau 3.10).

Tableau 3.10: Résultats des paramètres cinématiques trouvés après le processus d'optimisation pour ($\alpha_1=0.8$, $\alpha_2=0.1$ et $\alpha_3=0.1$)

Paramètres cinématique	Articulations					
	1	2	3	4	5	6
V_j^{\max}	81.88	73.98	57.28	51.06	86.11	91.52
W_j^{\max}	56.71	53.27	59.10	42.67	95.41	84.08
J_j^{\max}	25.33	23.73	34.56	17.69	59.68	33.35
V_j^{\min}	-81.96	-58.47	-50.47	-50.36	-86.71	-75.95
W_j^{\min}	-52.16	-34.38	-60.68	-39.06	-90.67	-68.88
J_j^{\min}	-40.41	-25.78	-41.14	-25.17	-47.09	-56.89

Il est observé aussi dans la figure 3.14 que la méthode proposée converge rapidement vers une solution adéquate et à un état stable, malgré le nombre de degrés de liberté du robot polyarticulé ainsi que la présence des contraintes cinématiques. Nous tenons compte bien évidemment des valeurs allouées aux

facteurs de pondération pour déterminer la fonction objectif dont les valeurs trouvées sont $F_{obj}=181.17$ et $F_{obj}=50.19$ pour les deux cas traités auparavant après 60 générations de la population par le code de calcul.

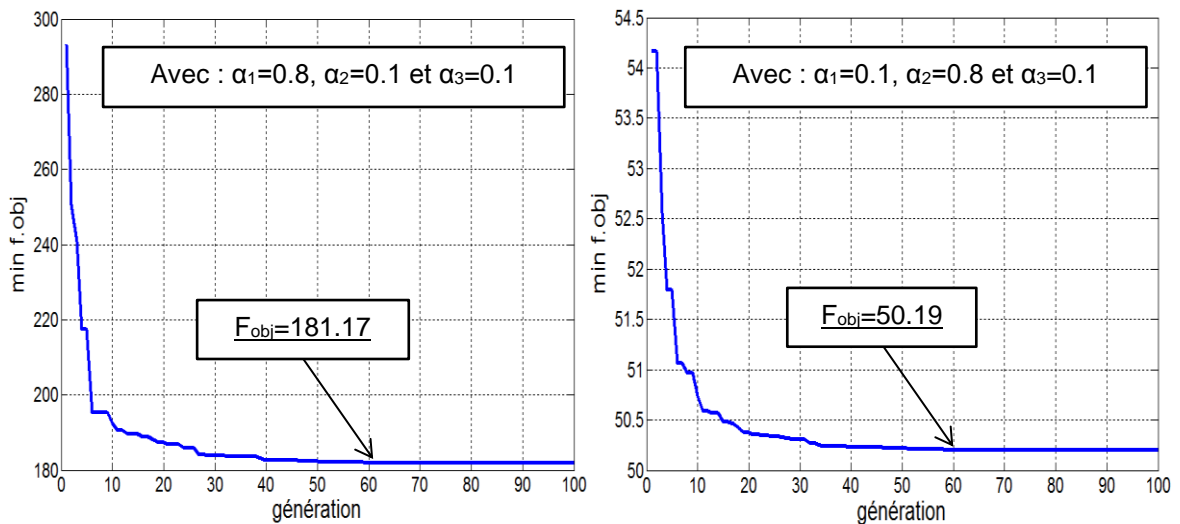


Figure 3.14: Résultats historiques du processus d'optimisation de la fonction objectif pour le robot manipulateur PUMA560-6R

3.3. Trajectoires optimales dans les tâches libres :

3.3.1. Minimisation du temps/jerk du robot planaire à 3 D.D.L :

Nous allons considérer un robot planaire à trois articulations de types rotoïde (3R), ces paramètres D-H et inertiels sont regroupés dans tableau 3.11. Ce robot se 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, Les capacités technologiques sont regroupées dans le tableau 3.12.

Tableau 3.11: Paramètres géométriques et inertiels du robot planaire 3R

Segment <i>i</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

Tableau 3.12: Capacités technologiques du robot planaire 3R

Articulation	q_i^{max}	\dot{q}_i^{max}	\ddot{q}_i^{max}	\dddot{q}_i^{max}	τ_i^{max}
i	[rad]	[rad/s]	[rad/s ²]	[rad/s ³]	[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 temps et jerk représenté par l'équation (2.35), les résultats obtenus et les aspects optimaux pour le mouvement d'un robot planaire 3R sont représentés respectivement sur la figure 3.15 et la figure 3.16.

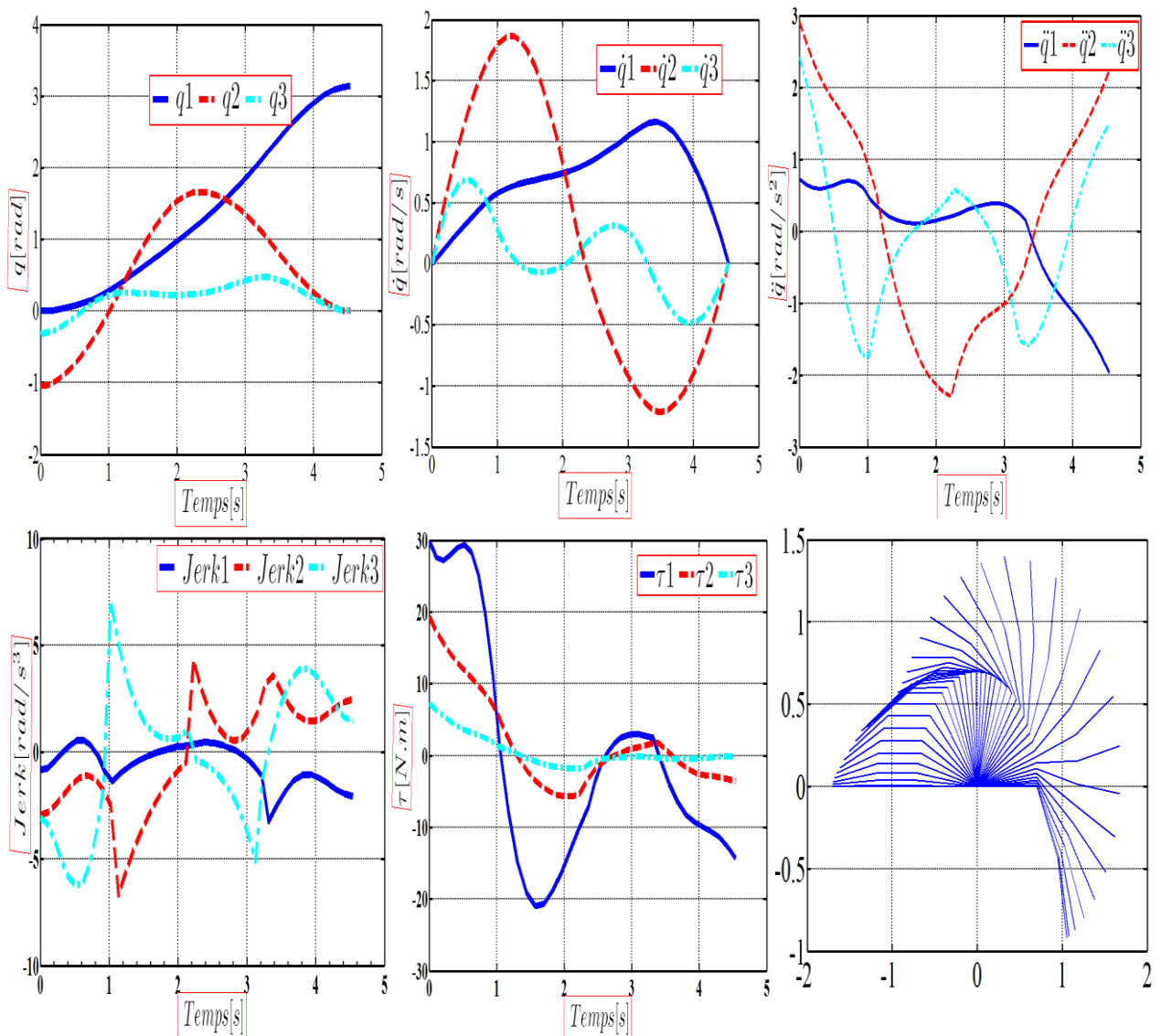


Figure 3.15: Résultats des trajectoires optimisées pour le robot planaire 3R pour

$\alpha = 0.5$ nous avons trouvé $T = 4.52s$ et $F_{obj} = 2.26$

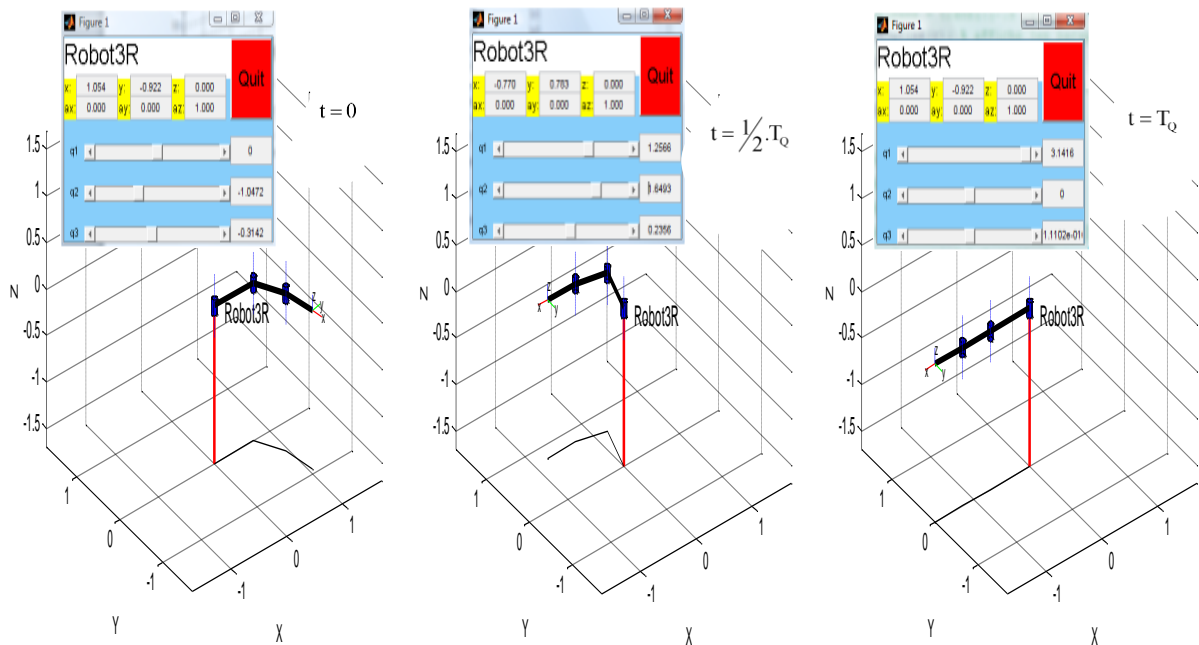
Figure 3.16: Aspects optimaux du mouvement pour le robot planaire 3R avec $\alpha=0.5$

Tableau 3.13: Résultats des valeurs maximales des contraintes kinodynamiques obtenues par notre procédure d'optimisation

Contraintes	Articulation		
	1	2	3
$q_i^{max} [rad]$	3.141	1.682	0.313
$\dot{q}_i^{max} [rad/s]$	1.174	1.879	0.688
$\ddot{q}_i^{max} [rad/s^2]$	-1.985	2.856	2.397
$\dddot{q}_i^{max} [rad/s^3]$	-3.285	-6.721	7.132
$\tau_i^{max} [N.m]$	30	19.224	7.483

A travers cet exemple, nous avons pu montrer la possibilité de notre approche de donner des résultats sub-optimaux pour les problèmes de planification de trajectoire des robots manipulateurs en minimisant une fonction objectif qui contient les deux facteurs le temps et le Jerk sous des contraintes kinodynamiques qui ont été imposées sur la structure polyarticulée. Nous avons pu traiter un problème de planification de trajectoire pour des opérations point à point (*pick and place*) en utilisant des fonctions splines cubiques qui permettent de garantir le lissage de la trajectoire et en même temps la continuité des vitesses,

des accélérations et des jerks pour un robot manipulateur planaire 3R. Notons que l'exécution du temps de calcul du code nécessite 34 secondes en utilisant 100 chromosomes dans des algorithmes génétiques et sur un PC de 2Ghz.

3.3.2. Minimisation du temps/couple/énergie du robot manipulateur Stanford à 6 D.D.L :

Considérant maintenant le robot manipulateur Stanford à six degrés de liberté de type 2RP3R (figure 3.17), ses paramètres de *Denavit-Hartenberg*, géométriques et inertiels sont reportés dans [15].

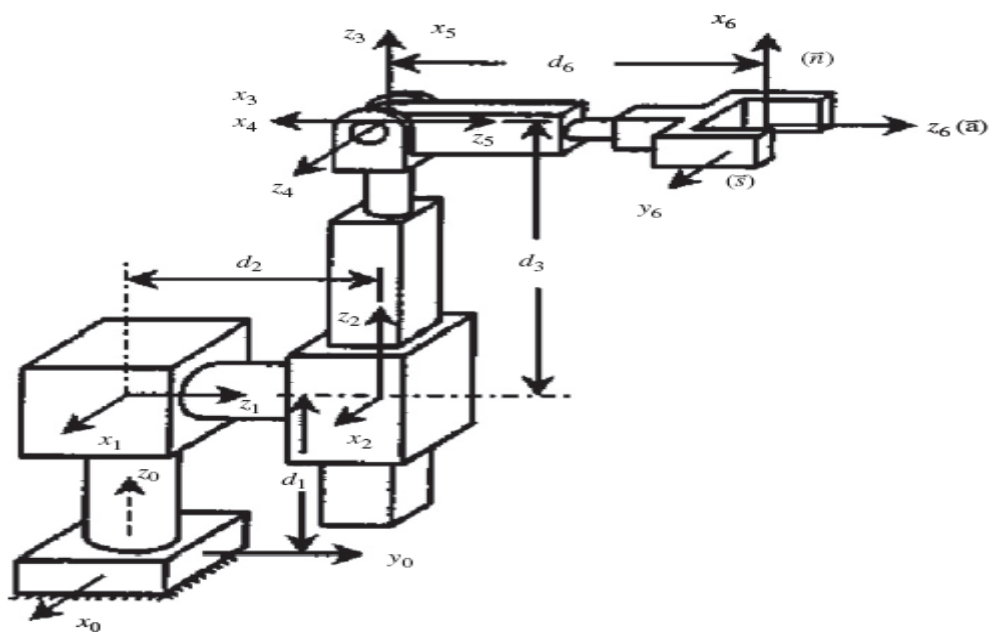


Figure 3.17: Robot manipulateur Stanford [15]

Le robot est invité à effectuer une tâche libre entre deux configurations définies dans le tableau 3.14. Les vitesses initiales et finales des articulations sont nulles.

Tableau 3.14: Tâche assignée par le robot Stanford [70]

Configuration	Articulation					
	1 (rad)	2 (rad)	3 (m)	4 (rad)	5 (rad)	6 (rad)
Initiale	0.1745	0.145	0.8	0.0873	0.1745	0.1047
Finale	-1.745	1.396	1.2	-0.853	1.078	-1.3894

Le transfert de la tâche doit être fait sans violation des contraintes représentées dans le tableau 3.15 et d'une manière souple et lisse en utilisant les fonctions de splines cubiques avec trois nœuds intermédiaires entre la position initiale et la position finale.

Tableau 3.15: Limites technologiques du robot Stanford [70]

Contraintes	Articulation						Contraintes	
	1-2 et 4-6	1	2	3	4	5		6
$q_i^{max}(rad)$		3.1	3.1	1.5	3.1	3.1	3.1	$q_i^{max}(m)$
$\dot{q}_i^{max}(rad/s)$		2.5	2.5	2.5	2.5	2.5	2.5	$\dot{q}_i^{max}(m/s)$
$\ddot{q}_i^{max}(rad/s^2)$		9.5	9.5	9.5	9.5	9.5	9.5	$\ddot{q}_i^{max}(m/s^2)$
$\dddot{q}_i^{max}(rad/s^3)$		50	50	50	50	50	50	$\dddot{q}_i^{max}(m/s^3)$
$\tau_i^{max}(N.m)$		50	80	100	10	10	10	$\tau_i^{max}(N)$

Nous allons utiliser un critère mixte avec puissance quadratiques (Eq 3.37) et avec différentes valeurs de coefficients de pondération α et β . Les résultats numériques obtenus sont récapitulés dans le tableau 3.16.

Les figures (3.18 et 3.19) montrent des profils de positions articulaires (rad ou m), des vitesses (rad/s ou m/s), des accélérations (rad/s^2 ou m/s^2) et des jerks (rad/s^3 ou m/s^3) du robot manipulateur Stanford. L'observation de ces résultats (Tableau 3.16) montre que lorsqu'on a attribué une égalité entre le temps de transfert et la moyenne quadratique des efforts actionneurs soit $\alpha=0.5$ et $\beta=0.5$ sous des contraintes dynamiques, on obtient une fonction objectif $F_{obj}=1.427$ qui correspond à un temps de transfert optimal $T=2.755s$. Tandis que lorsque nous stabilisons les facteurs de pondérations $\alpha=0.75$ et $\beta=0.25$ et avec prise en compte les contraintes kinodynamiques on obtient une fonction objectif $F_{obj}=1.889$ et un temps de transfert $T=2.373s$.

En effet, le transfert entre les configurations initiales et finales est exécuté d'une manière douce et sans violation des contraintes imposées sur la structure polyarticulé. La variation des coefficients de pondération mène à différents types de mouvements et par conséquent elle mène aussi à différents temps de transfert et fonctions objectifs.

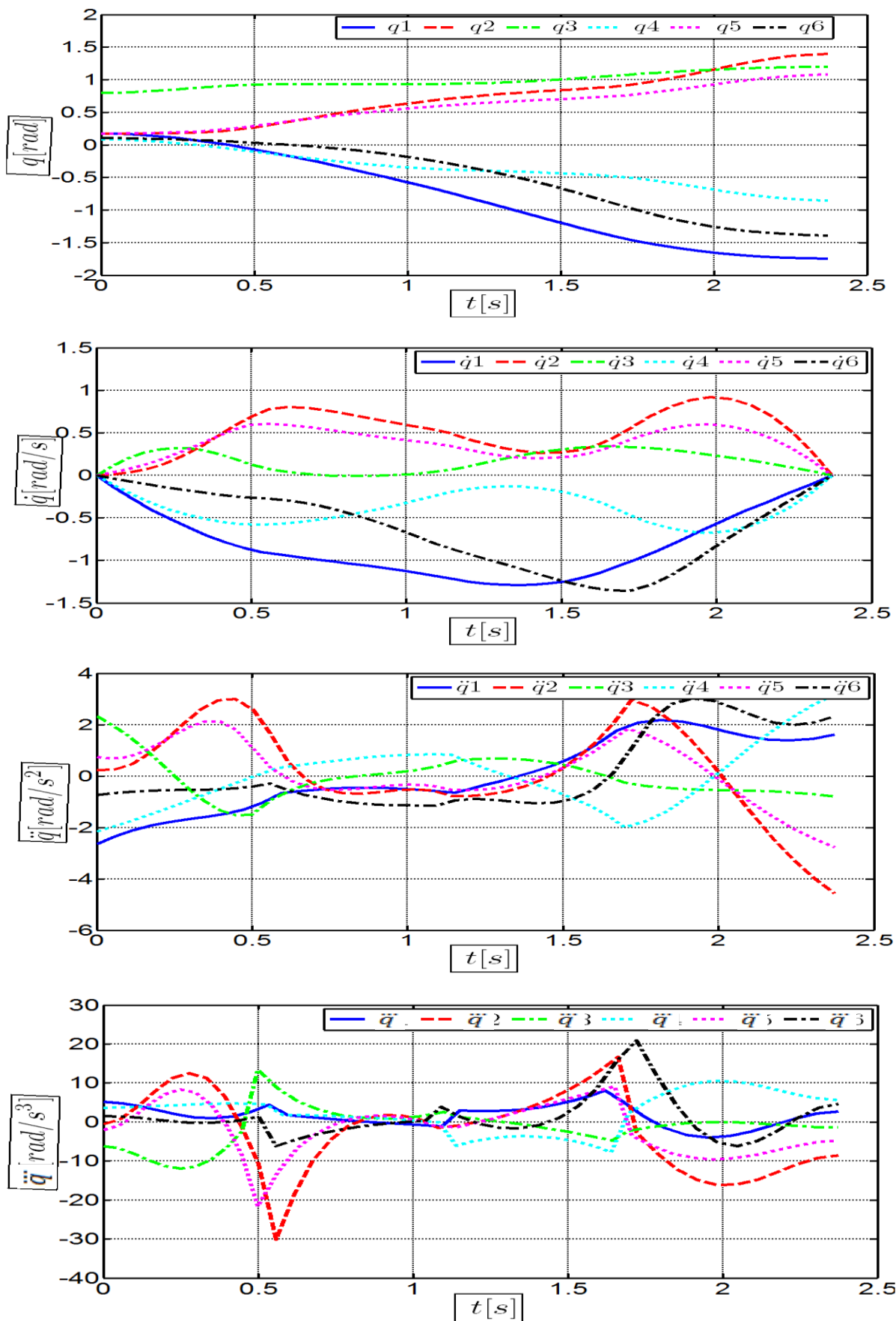


Figure 3.18: Trajectoires optimales du robot Stanford avec contraintes kinodynamiques pour $(\alpha = 0.75, \beta = 0.25)$

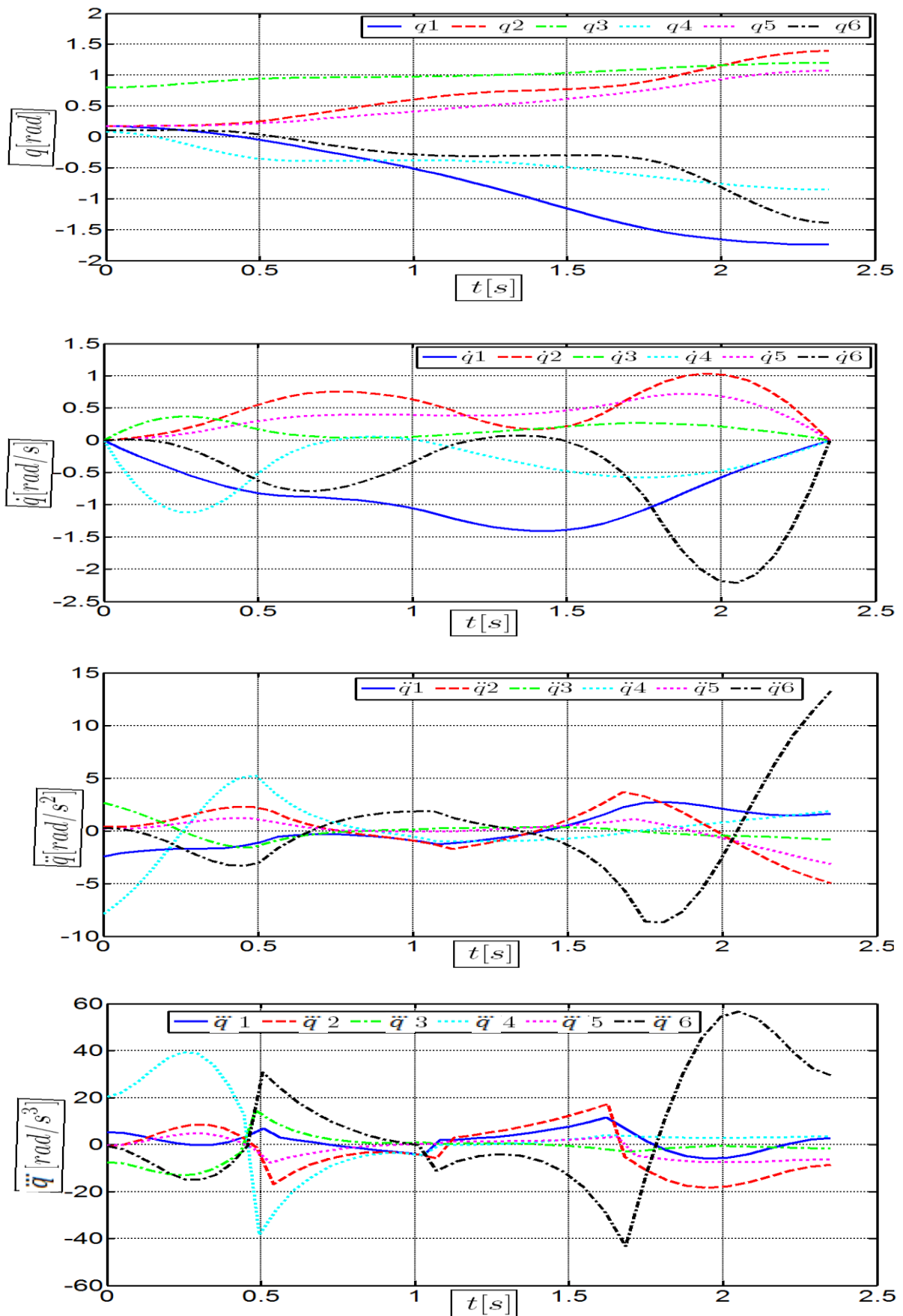


Figure 3.19: Trajectoires optimales du robot Stanford avec contraintes dynamiques pour ($\alpha = 0.5$, $\beta = 0.5$)

Généralement, un coefficient de pondération α fixe et β décroissant mènent à abaisser le temps de transfert mais les couples et les vitesses sont plus élevés. D'autre part, quand nous réduisons la valeur du coefficient de pondération α et nous exposons des contraintes cinématiques nous parvenons à des temps de transfert plus grands et par conséquent des mouvements plus lents.

Finalement, la simulation de ce robot manipulateur en vue 3-D (figure 3.20), montre le mouvement optimal de ce robot qui commence à partir de la configuration initiale et se termine à la configuration finale.

Tableau 3.16: Résultats numériques du robot Stanford 6 D.D.L

α	β	Contraintes				T	F _{obj}
		Couple	Vitesse	Accélération	Jerk		
0.75	0.25	o				2.316	1.845
		o	o			2.321	1.858
		o	o	o		2.360	1.881
		o	o	o	o	2.373	1.889
0.75	1	o				2.330	2.174
		o	o			2.339	2.167
		o	o	o		2.363	2.197
		o	o	o	o	2.396	2.200
0.50	0	o				2.334	1.169
		o	o			2.359	1.182
		o	o	o		2.390	1.197
		o	o	o	o	2.405	1.224
0.50	0.50	o				2.350	1.599
		o	o			2.363	1.605
		o	o	o		2.395	1.631
		o	o	o	o	2.411	1.685
0.25	0.75	o				2.344	1.492
		o	o			2.360	1.411
		o	o	o		2.399	1.512
		o	o	o	o	2.539	1.705

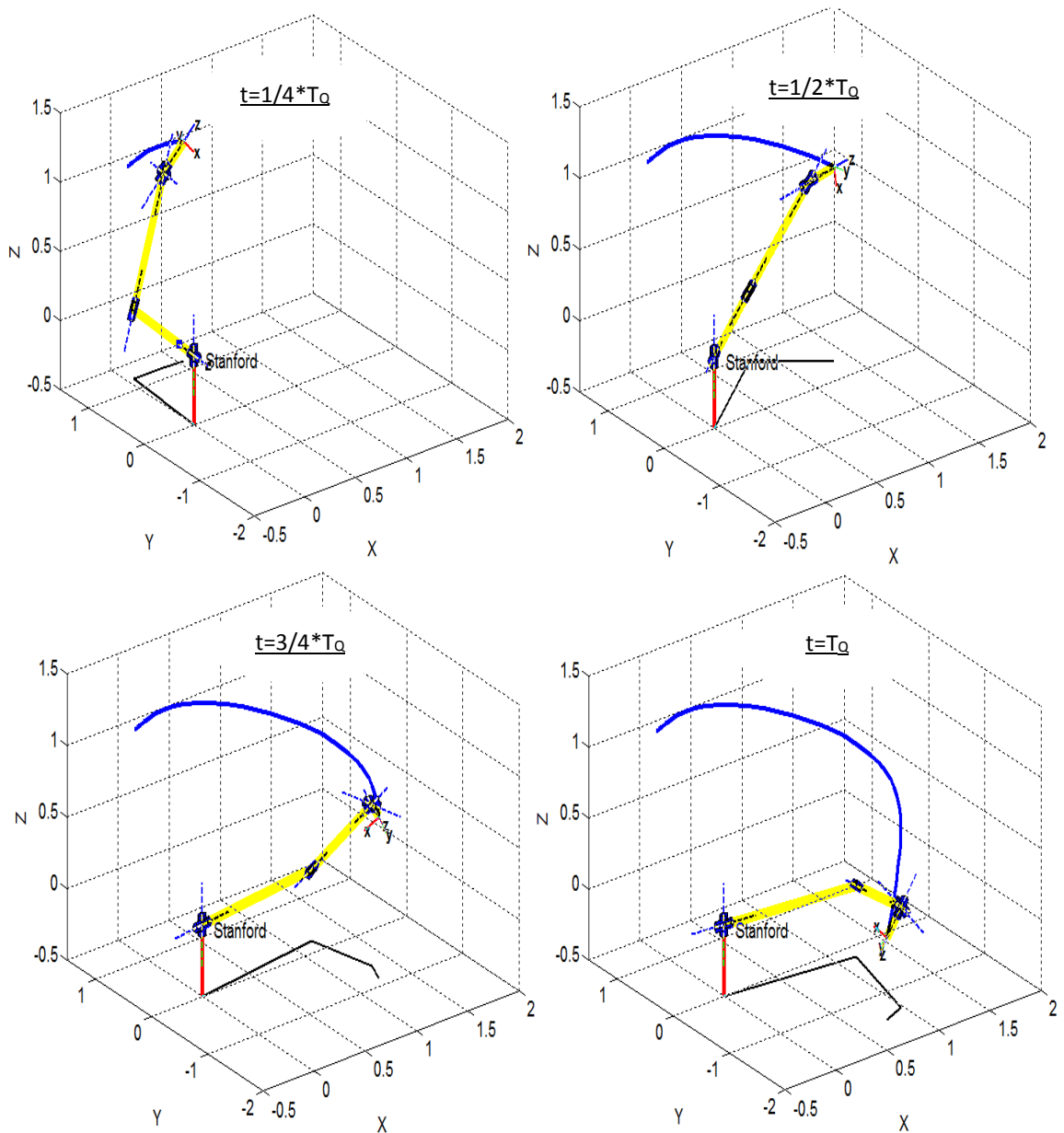


Figure 3.20: Mouvement Optimal du robot Stanford pour ($\alpha = 0.5$, $\beta = 0.5$)

3.4. Conclusion :

Dans ce chapitre, différents cas du problème de planification de trajectoire des robots manipulateurs ont été discutés. Les deux premiers exemples ont porté sur la comparaison des résultats obtenus par notre approche avec ceux obtenus d'une part par l'algorithme génétique / intervalle [81], et d'autre part avec une technique de programmation quadratique séquentielle (SQP) [63] et un algorithme de recherche dit d'harmonie hybride (HHSA) [82]. Le problème qui a été considéré est celui de la minimisation du temps de déplacement sous des contraintes

cinématiques et dynamiques des tâches imposées. Les résultats obtenus dans ces deux exemples montrent que notre approche demeure très compétitive par rapport à d'autres méthodes. Les résultats obtenus dans le troisième exemple ont été comparés avec ceux obtenus par l'algorithme génétique de Fuzzy « *Fuzzy genetic algorithm* » [83] et ceci montre que notre approche peut s'adapter facilement à des problèmes dont la fonction objectif combine entre la minimisation du temps de la tâche et/ou le jerk. Un autre exemple a traité aussi le cas des tâches imposées où un robot manipulateur avec *6ddl* est considéré pour exécuter une tâche avec 12 points de passage et en minimisant une fonction multi-objectifs qui comporte le temps, l'accélération et le jerk sous des contraintes cinématiques les résultats obtenus montrent que notre approche converge rapidement vers une solution optimale, un autre avantage indéniable de cette technique, d'ailleurs c'est l'objectif pour lequel elle est mise en œuvre, c'est son habilité à traiter les problèmes de grande dimension par exemple un grand nombre de d.d.l et de nœuds de raccordement, ce qui n'est pas le cas pour la plupart des techniques dont la mise en œuvre devient une tâche fastidieuse.

Le problème qui a été considéré pour les deux derniers exemples est celui de la planification de trajectoires optimales dans les tâches point à point. Pour cela, deux fonctions multi-objectifs ont été minimisées et qui représentent d'une part le temps et le jerk d'un robot à *3ddl* et d'autre part le temps, le couple moteur et l'énergie d'un robot manipulateur à *6ddl* de type Stanford. Des contraintes géométriques et kinodynamiques ont été présent en comptes pour les deux robots. Les résultats obtenus de ces deux exemples montrent que la méthode converge vers des solutions de bonne qualité dans des temps de calcul raisonnables.

CONCLUSION

Le travail mené dans cette thèse est considéré comme une contribution à la planification et à l'optimisation multi-objectifs des trajectoires pour les manipulateurs robotiques. Il constitue en particulier une tentative de proposer d'autres méthodes efficaces pour le traitement des trajectoires optimales dans les tâches imposées et libres en minimisant certains critères significatifs pour améliorer la productivité des robots industriels qui font généralement des tâches répétitives.

Sur la base de cette représentation, deux voies ont été investiguées.

En premier lieu, nous nous sommes intéressés à la planification des trajectoires dans les tâches imposées. En l'occurrence, nous avons proposé la méthode du vecteur d'intervalle de temps. Ceci constitue une contribution originale dans l'étude de planification des trajectoires imposées dans le sens où nous avons pu minimiser une fonction objectif représentant le temps ensuite le temps/le jerk et finalement le temps/l'accélération/le jerk et en introduisant des facteurs de pondérations. La simplicité de cette méthode a permis de générer des trajectoires optimales dans les espaces cartésiens et articulaires, même avec un nombre de points intermédiaires important. Les résultats ainsi que leurs analyses ont été présentés en détail pour des systèmes polyarticulés avec 2, 3 et 6 degrés de liberté sous des contraintes cinématiques et dynamiques imposées sur leurs structures. Les résultats obtenus par la programmation de codes sous le logiciel MATLAB®, sont confrontés à ceux de la littérature afin de pouvoir vérifier leur exactitude.

En deuxième lieu, nous avons développé une méthodologie générale pour générer des trajectoires optimales dans les tâches libres. En normalisant l'échelle temporelle du robot, le problème original de dimension infinie a été défini comme

un problème d'optimisation de dimension finie et les contraintes kinodynamiques ont été converties sous forme d'intervalles admissibles sur le temps de transfert. Ensuite, la trajectoire du robot est discrétisée pour simplifier encore le traitement de diverses contraintes du problème, alors que les critères de performances sont réécrits sous formes d'une fonction à variable unique simple à optimiser. Pour cela, deux fonctions multi-objectifs ont été minimisées d'une part une fonction qui représente le temps/le jerk et d'autre part une fonction qui représente le temps/le couple/l'énergie. Les résultats obtenus par des codes de calcul sous le logiciel MATLAB® ont montré l'efficacité de la méthode qui peut approcher des solutions de type sub-optimal en des temps de calcul acceptables, ce qui la rend très intéressante pour les applications industrielles.

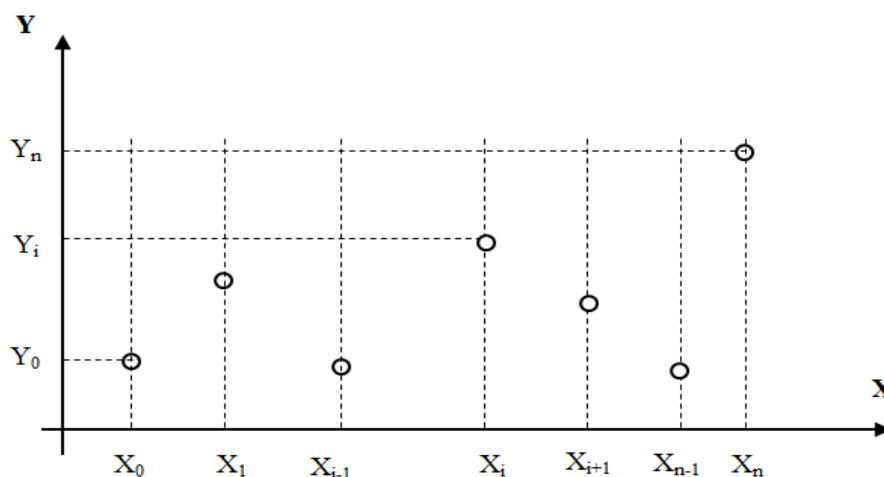
Comme perspectives, nous envisageons les thèmes suivants :

- 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.
- L'utilisation d'autres techniques d'optimisation hybride avec des fonctions multi-objectifs.
- L'optimisation multi-objectifs dans les tâches imposées ou bien les tâches libres car les résultats très encourageants obtenus incitent à faire des améliorations pour son application à la planification de trajectoires en tenant compte de plusieurs paramètres tels que : les limites physiques du système mécanique robotisé, les spécifications exigées de la tâche à réaliser et les paramètres liés à l'environnement d'évolution du robot ;
- L'extension de l'approche à la planification de trajectoire pour d'autres systèmes robotisés tels que les robots et les plateformes mobiles, les robots flexibles.

ANNEXE

INTERPOLATION POLYNOMIALE PAR MORCEAUX

"SPLINES CUBIQUES"



Une interpolation polynomiale [84] entre les points $[X_0, Y_0]$ et $[X_n, Y_n]$ peut être effectuée de la manière suivante :

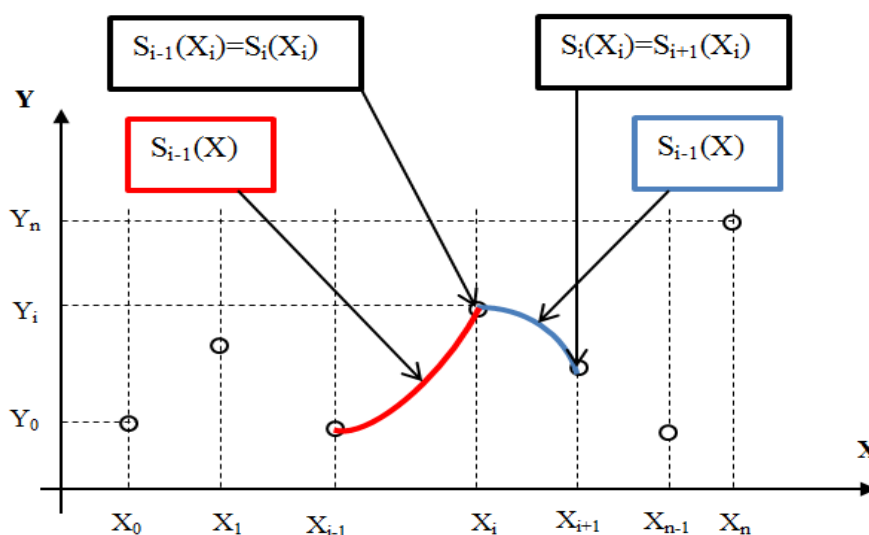
- Utiliser un polynôme de degré n s'écrit :

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

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 :

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

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

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

Qui implique que la courbe passe par tous les points.

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

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

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

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

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

Qui implique que les dérivées secondes sont continuées.

Les inconnues des problèmes :

L'expression (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 :

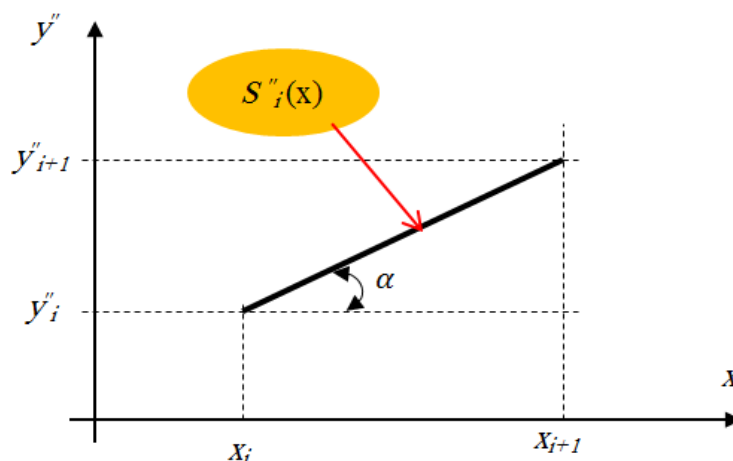
En principe, on peut établir $(N+1)$ équations avec l'expression (2), $(N-1)$ équations avec chacune des expressions (2), (3) et (4), ce qui fait au total $(N+1)+3(N-1)=4N-2$. Deux équations manquantes qu'on peut écrire aux extrémités de l'intervalle en faisant intervenir l'une des dérivés (première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

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

Où :

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

Ce qui donne après calcul :

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

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

L'expression (6) devient :

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

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

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

Ou p_i et q_i représentent des constantes d'intégration.

On remplace successivement dans l'expression (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 (8) sont les termes $\{m_i\}$ qu'on peut trouver en dérivant la même expression. Ainsi, on a :

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

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

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

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

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

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 $i = 1, \dots, n - 1$.

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

L'équation (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 :

$$\begin{aligned} a_{i,0} = y_i & \quad , & \quad a_{i,1} = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i(2m_i + m_{i+1})}{6} \\ a_{i,2} = \frac{m_i}{2} & \quad , & \quad a_{i,3} = \frac{m_{i+1} - m_i}{6h_i} \end{aligned}$$

Chaque polynôme peut se mettre sous la forme :

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

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

REFERENCES BIBLIOGRAPHIQUES

1. Robert, M and Rayner, C., "The optimisation of a high speed servomechanism", Ph.D, University of Bath, (2010).
2. Saeed, B. N. and Sun, F., "Introduction to robotics analysis, systems, applications", Edit, Prentice Hall, (2001).
3. Craig, J. J., "Introduction to robotics: mechanics and control", Upper Saddle River: Pearson Prentice Hall, (2005).
4. Zoller, Z. and Zentay, P., "Constant kinetic energy robot trajectory planning", Periodica Polytechnica, Mechanical Engineering, 43(2), (1999), 213–228.
5. Ayten, K. K. Iravani, P. and Sahinkaya ,M. N., "Optimum trajectory planning for industrial robots through inverse dynamics", In Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics (ICINCO2011), (2011).
6. COIFFET, P., "La robotique : principes et applications", Hermès, Paris, (1986).
7. SELIG, J.M., "Introductory robotics", Prentice Hall International, (1992).
8. Denavit, J., and R. S. Hartenberg., "Kinematic modelling for robot calibration", Trans. ASME Journal of Applied Mechanics, 22, (1955), 215-221.
9. Wisama, K. Kleinfinger, J. and Gautier, Maxime., "Reducing the computational burden of the dynamic models of robots", Robotics and Automation. Proceedings, IEEE International Conference, V. 3, (1986).
10. Craig, J. J., "Introduction to robotics, mechanics and control", Addison-Wesley, (1989).
11. Lallemand, J. P. and Zegloul, S., "Robotique, aspects fondamentaux", Edition Masson, (1994).
12. Khalil, W. and Dombre, E., "Modélisation, identification et commande des robots", 2^{ème} Edition Hermes, (1999).

13. Saramago, S.F.P. and Steffen, V., "Trajectory modeling of robot manipulators in the presence of obstacles", *Journal of Optimization Theory and Applications*, 110 (1), (2001), 17–34.
14. Ramabalan, S., Saravanan, R. and Balamurugan, C., "Multi-objective dynamic optimal trajectory planning of robot manipulators in the presence of obstacles", *Int J Adv Manuf Technol*, 41, (2009), 580–594.
15. Saravanan, R., Ramabalan, S. and Balamurugan, C., "Evolutionary multi-criteria trajectory modeling of industrial robots in the presence of obstacles", *Engineering Applications of Artificial Intelligence*, 22, (2009), 329-342.
16. Sheng, M., Su, B. and Zou, L., "A Class of Variable Degree Trigonometric Polynomial Spline and Its Applications", *Edutainment 2016, Lecture Notes in Computer Science*, Springer, Cham, (2016), 148–162.
17. Lin C-J., "Motion planning of redundant robots by perturbation method", *Mechatronics*, 14(3), (2004), 281–297.
18. Valero, F., Mata, V. and Besa, A., "Trajectory planning in workspaces with obstacles taking into account the dynamic robot behaviour", *Mechanism and machine theory*, 41(5), (2006), 525–536.
19. Rubio, F., Valero, F., Sunyer, J. and Cuadrado, J., "Optimal time trajectories for industrial robots with torque, power, jerk and energy consumed constraints", *Industrial Robot: An International Journal*, 39(1), (2012), 92–100.
20. KIM, D. H. and SHIN, S., "Local path planning using a new artificial potential function composition and its analytical design guidelines", *Advanced Robotics*, V. 20, n°1, (2006), 115–135.
21. Lin, C. C. and Chuang, J. H., "A potential-based path planning algorithm for hyper-redundant manipulators", *Journal of the Chinese Institute of Engineers*, 33(3), (2010), 415–427.
22. Homsí, S., "Online generation of time-optimal trajectories for industrial robots in dynamic environments", PhD diss, Grenoble Alpes, (2016).
23. Choset, H., Lynch, K. M., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L. E. and Thrun, S., "Principles of Robot Motion: Theory, Algorithms, and Implementations", MIT Press, (2005).
24. Lozano-Pérez, T. and Wesley, M.A., "An algorithm for planning collision-free paths among polyhedral obstacles", *Communications of the ACM*, 22(10), (1979), 560-570.

25. Helguera Arellano, C., "Contribution à la résolution du problème des minima locaux dans une méthode de planification de trajectoires sans collision pour robots manipulateurs", Thèse de Doctorat, université de Poitiers, (2001).
26. Lin, C.S., Chang, P. R. and Luh, J.Y.S "Formulation and optimization of cubic polynomial joint trajectories for industrial robots", IEEE Trans Autom Control, 28(12), (1983), 1066–1074.
27. Thompson, S.E. and Patel, R.V., "Formulation of joint trajectories for industrial robots using b-splines", IEEE Trans Ind Electron, IE-34(2), (1987), 192–199.
28. Wang, C.H. and Horng, J.G., "Constrained minimum-time path planning for robot manipulators via virtual knots of the cubic b-spline functions", IEEE Trans Autom Control, 35(5), (1990), 573–577.
29. Jamhour, E. and André, P., "Planning smooth trajectories along parametric paths", Math Comput Simul, 41(5), (1996), 615–626.
30. Piazzzi, A. and Visioli, A., "Global minimum-time trajectory planning of mechanical manipulators using interval analysis", Int J Control, 71(4), (1998), 631–652.
31. Tse, K.M. and Wang, C.H., "Evolutionary optimization of cubic polynomial joint trajectories for industrial robots", In: Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, V. 4, (1998), 3272–3276.
32. Abu dakka, F. J., Assad, I. F., Alkhdour, R. M. and Abderahim, M., "Statistical evaluation of an evolutionary algorithm for minimum time trajectory planning problem for industrial robots", Int J Adv Manuf Technol, V. 89, (2017), 389-406.
33. Bobrow, J. E., Dubowsky, S. and Gibson, J. S., "Time-optimal control of robotic manipulators along specified paths", The International Journal of Robotics Research, 4(3), (1985), 3–17.
34. Bobrow, J. E., "Optimal control of robotic manipulators", PhD thesis, University of California, Los Angeles, (1982).
35. Bobrow, J. E., Dubowsky, S. and Gibson, J. S., "On the optimal control of robotic manipulators with actuator constraints", In American Control Conference, (1983), 782–787.
36. Shin, K. and McKay, N., "Minimum-time control of robotic manipulators with geometric path constraints", IEEE Transactions on Automatic Control, 30(6), (1985), 531-541.

37. Pfeiffer, F. and Johanni, R., "A concept for manipulator trajectory planning", *IEEE Journal on Robotics and Automation*, V 3, (1987), 115-123.
38. Slotine, J.J.E. and Li, W., "Composite adaptive control of robot manipulators", *Automatica*, V 25, (1989), 509-519.
39. Tangpattanakul, P. and Artrit, P., "Minimum-Time Trajectory of Robot Manipulator Using Harmony Search Algorithm", In: 6th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Chonburi, Thailand, (2009).
40. Bendali, N. and Ouali, M., "Optimization the Trajectories of Robot Manipulators Along Specified Task", *International Journal of Intelligent Engineering and Systems*, V.11, n°.1, (2018), 11-19.
41. Kahn, M. E. and Roth, B., "The near-minimum-time control of open-loop articulated kinematic chains", *Journal of Dynamic Systems Measurement and Control*, (1971), 93-164.
42. Bendali, N. Ouali, M. Kifouche, A. and Boughedaoui. R., "Planification des Trajectoires Libres en Temps Optimal des Robots Industriels sous les Contraintes Dynamiques", *Congrès Algérien de Mécanique*, Mascara, Algeria, (2013).
43. Haddad, M., "Exploitation optimale des plateformes mobiles et des manipulateurs mobiles à roues", *Thèse de Doctorat, EMP*, (2007).
44. Rubio, F., Llopis-Albert, C., Valero, F. and Suñera, J.L., "Industrial robot efficient trajectory generation without collision through the evolution of the optimal trajectory", *J. Rob. Autonom. Syst*, 86, (2016), 106–112.
45. Rubio, F. J., Valero, F. J., Suñer, J. L. and Garrido, A., "The simultaneous algorithm and the best interpolation function for trajectory planning", *Industrial Robot*, (ISSN: 0143-991X). 37, 5, (2010), 441-451.
46. Shiller, Z., "Time energy optimal control of articulated systems with geometric path constraints", *IEEE int. conf. on rob. and aut*, V. 4, (May 1994), 2680-2685.
47. Duleba, I., "The method of minimal neighbourhood: a new and most effective iterative for minimum cost trajectory planning in robot manipulators", *Robotica*, V. 13, (1995), 297-304.

48. Shiller, Z., "Time-Energy Optimal Control of Articulated Systems With Geometric Path Constraints", *J. Dyn. Sys., Meas., Control*, 118(1), (1996), 139-143.
49. Saramago, S.F.P. and Steffen, V. Jr., "Optimization of the trajectory planning of robot manipulators taking into account the dynamics of the system", *Mech. Mach. Theory*, 33(7), (1998), 883-894.
50. Xu, H., Zhuang, J., Wang, S. and Zhu, Z., "Global Time-Energy Optimal Planning of Robot Trajectories", In: *Proceedings of the International Conference on Mechatronics and Automation*, (2009), 4034-4039.
51. Antonelli, G., Curatella, C. and Marino, A., "Constrained motion planning for open-chain industrial robots", *Robotica*, (2010), 1-18.
52. Saramago, S. F. P. and Ceccarelli, M., "Effect of basic numerical parameters on a path planning of robots taking into account actuating energy", *Mechanism and Machine*, 39, (2004), 247-260.
53. Saramago, S.F.P. and Steffen, V. Jr., "Optimal trajectory planning of robot manipulators in the presence of moving obstacles", *Mech. Mach. Theory*, 35(7), (2000), 1079-1094.
54. Chettibi, T., "Synthesis of dynamic motions for robotic manipulators geometric path constraints", *Mechatronics*, 16(9), (2006), 547-563.
55. Bendali, N. Ouali, M. Kifouche, A. and Hamli Ben Zahar, H., "Optimal Trajectory of the Robots Manipulators in the Point-to-point Tasks under Dynamic Constraints", *International Conference on the electrical engineering CIGE'2013*, Bechar, Algeria, (2013).
56. Bendali, N. Ouali, M. and Ghellal, K., "Optimal Trajectory Planning under Kinodynamics Constraints for a 6-DOF PUMA 560", *Proceedings of the The International Conference on Engineering & MIS 2015*, Istanbul, Turkey, (2015).
57. Von Stryk, O. and Schlemmer, M., "Optimal control of the industrial robot Manutec r3", in: R. Bulirsch, D. Kraft (Eds.), *Computational Optimal Control*, International Series of Numerical Mathematics, vol. 115, Basel, Birkhauser, (1994), 367-382.
58. Heim, A. and Stryk, O. V., "Trajectory optimization of industrial robots with application to computer-aided robotics and robot controllers", *Journal of Mathematical Programming and Operations Research*, V. 47, (2007), 407-420.

59. Kawamura, S. and Svinin, M., "Advances in Robot Control: From Everyday Physics to Human-Like Movements", Springer Science & Business Media, ISBN 978-3-540-37346-9, (2007).
60. Zatsiorsky, V.Z., "Kinematics of Human Motion, Human Kinetics", ISBN 9780880116763, (1998).
61. Bendali, N. and Ouali, M., "On-line trajectory planning of time-jerk optimal for robotic arms", *Mediterranean Journal of Modeling and Simulation*, 6(1), (2016), 034-044.
62. Gasparetto, A. and Zanatto, V., "A New Method for Smooth Trajectory Planning of Robot Manipulators", *Mechanism and Machine Theory* 42, (2007), 455-471.
63. Gasparetto, A. and Zanotto, V., "A technique for time-jerk optimal planning of robot trajectories", *Robot Comput Integr Manuf*, 24(3), (2008), 415-426.
64. Hl, L., "A fast and unified method to find a minimum-jerk robot joint trajectory using particle swarm optimization", *J Intell Robot Syst*, 75(3-4), (2014), 379-392.
65. Piazzzi, A. and Visioli, A., "An interval algorithm for minimum-jerk trajectory planning of robot manipulators", In: *Proceedings of the 36th Conference on Decision and Control*, (1997), 1924-1927.
66. Piazzzi, A. and Visioli, A., "Global minimum-jerk trajectory planning of robot manipulators", *IEEE Trans. Ind. Electron*, 47(1), (2000), 140-149.
67. Kyriakopoulos, K.J., Saridis, G.N., "Minimum jerk path generation", In: *Proceedings of the IEEE International Conference on Robotics and Automation*, (1998), 364-369.
68. Zhu, Z., Xu, W., Meng, Z and Sun, Y., "Optimal trajectory and placement of a SCARA robot for natural yarns handling in the lattice distortion modification processing", *International Journal of Mechatronics and Manufacturing Systems*, 8(3-4), (2015), 85-101.
69. Chettibi, T., Lehtihet, H.E., Haddad, M. and Hanchi, S., "Minimum cost trajectory planning for industrial robots", *Eur Jour of Mech A/Solids* 23, (2004), 703-715.
70. Saravanan, R. and Ramabalan, S., "Evolutionary Minimum Cost Trajectory Planning for Industrial Robots", *J Intell Robot Syst*, V. 52, (2008), 45-77.

71. Saravanan, R., Ramabalan, S. and Balamurugan, C., "Evolutionary collision-free optimal trajectory planning for intelligent robots", *The International Journal of Advanced Manufacturing Technology*, 36(11), (2008), 1234–1251.
72. Bendali, N., Ouali, M., Kifouche, A. and Benlamnour, M. F., "Planification des Trajectoires des Robots Manipulateurs Sous les Contraintes en Utilisant les Splines Cubiques", *International Conference on the electrical engineering CIGE'2013*, Bechar, Algeria, (2013).
73. Larios, J., "Le fil d'Ariane Une méthode de planification générale application à la planification", *Thèse de Doctorat, INP de Grenoble*, (1994).
74. Mekhnacha, K., "Méthode probabilistes Bayésiennes pour la prise en compte des incertitudes", *Thèse de Doctorat, INP de Grenoble*, (1999).
75. Berro, A., "Optimisation multiobjectif et stratégies d'évolution en environnement dynamique", *Thèse de Doctorat, Université de Toulouse I*, (2001).
76. Bergamaschi, P. R., Saramago, S. D. F. P. and Coelho, L. D. S., "Comparative study of SQP and metaheuristics for robotic manipulator design", *Applied Numerical Mathematics* 58, (2008), 1396-1412.
77. Jonson, C. G. and Marsh, D., "Modelling robot manipulators with multivariate B-splines", *Robotica*, V. 17, (1999), 239-247.
78. Chettibi, T. and Lehtihet, H. E., "A new approach for point to point optimal motion planning problems of robotic manipulators", *ESDA 2002, APM-10, ASME conf*, (2002).
79. Hollerbach, J.M., "Dynamic scaling of manipulator trajectories", *Journal of Dynamic Systems Measurement and Control*, 106(1), (1984), 102-106.
80. Bendali, N., Ouali, M. and Kifouche, A., "A New Method for Time-Jerk Optimal Trajectory Planning Under Kino-dynamic Constraint of Robot Manipulators in Pick-and-Place Operations", *IAES International Journal of Robotics and Automation*, 3(3), (2014), 184-190.
81. Guarino Lo Bianco, C. and Piazzi, A., "A semi-infinite optimization approach to optimal spline trajectory planning of mechanical manipulators", in: M.A. Goberna, M.A. Lopez (Eds.), *Semi-infinite Programming: Recent Advances*, Springer-Verlag, Berlin, (2001), 271-297.
82. Tangpattanakul, P., Meesomboon, A. and P. Artrit., "Optimal Trajectory of Robot Manipulator Using Harmony Search Algorithm", *Recent Advances in*

- Harmony Search Algorithm, the series Studies in Computational Intelligence, Springer-Verlag, Berlin, V.270, (2010), 23-36.
83. Cong, M., Xu, X., Xu, P., "Time-jerk synthetic optimal trajectory planning of robot based on fuzzy genetic algorithm", Int. J. Intelligent Systems Technologies and Applications, V. 8, (2010), 1-4.
84. Mathews, J. H., "Numerical Methods for Mathematics, Science and Engineering", Prentice-Hall of India. Second Edition, (1994).