



THÈSE

présentée en cotutelle à

**l'Ecole Nationale d'Ingénieurs de Sfax
et l'Université d'Orléans**

en vue de l'obtention du

DOCTORAT

Dans la discipline

Génie Électrique

(Robotique)

par

Yassine Bouteraa

(Ingénieur ENIS)

COMMANDE DISTRIBUÉE ET SYNCHRONISATION DE ROBOTS INDUSTRIELS COOPÉRATIFS

soutenue le 20 février 2012, devant le jury composé de :

M. Abderrazak OUALI (Professeur, Université de Sfax)
Mohamed BENREJEB (Professeur, Université de Tunis El Manar)
Saïd ZEGHLOUL (Professeur, Université de Poitiers)
Gérard POISSON (Professeur, Université d'Orléans)
Nabil DERBEL (Professeur, Université de Sfax)
Gilles MOURIOUX (Maître de Conférences, Université de Limoges)
Jawhar GHOMMAM (Maître de Conférences, Université de Carthage)
Cyril NOVALES (Maître de Conférences, Université d'Orléans)

Président
Rapporteur
Rapporteur
Directeur de thèse
Directeur de thèse
Membre
Membre
Membre

Remerciements

La recherche présentée dans cette thèse est le résultat de mes études de doctorat sur la période allant de septembre 2009 à Février 2012, dirigées, conjointement, sous la direction du Professeur Nabil Derbel de l'université de Sfax et du Professeur Gérard Poisson de l'université d'Orléans.

Au terme de ma thèse, je tiens à remercier :

les professeurs qui m'ont fait l'honneur d'être les rapporteurs de mon travail et qui m'ont fait l'honneur également d'accepter de faire partie de mon jury.

Je tiens à remercier mes directeurs de thèse, M. Nabil Derbel, Professeur à l'université de Sfax et directeur des études de l' ENIS, ainsi que M. Gérard Poisson, Professeur à l'université d'Orléans et directeur de l'IUT de Bourges, pour la confiance qu'ils m'ont accordée tout au long de mon travail de thèse.

Je suis particulièrement reconnaissant au Docteur Jawhar Ghommam, qui a contribué, par ses idées et conseils, d'une manière efficace et importante, à l'avancée de ce travail.

Il me serait impossible d'oublier Monsieur Mohamed Chtourou directeur de l'unité de recherche ICOS pour son amitié, son soutien et ses encouragements.

Avec un grand plaisir Je dédie ce travail à :

Ma mère, Mon père, pour tout le bonheur qu' ils m' apportent. Merci d'être mes parents. De m'avoir supporté et appuyé durant toutes ces années. Il me tient énormément à coeur de vous dire MERCI.

Ma femme et mes deux fils Youssef et Aziz pour avoir accepté tant de sacrifices durant ces dernières années.

Les personnes qui ont joué un grand rôle dans ma vie : ma soeur Sana, ma petite famille, mes docteurs et principalement Sami Makhloufi.

Les personnes qui ont contribué peu ou beaucoup afin de surmonter les moments difficiles que j'ai vécu avec la maladie.

Les nombreuses personnes qui directement, indirectement, humainement, amicalement, amoureuxment et même haineusement, ont contribué à former ma personnalité et mon niveau.

Les amis de tous les jours.

Un travail de thèse est indissociable d'un laboratoire et d'une équipe de recherche. De la qualité des rapports humains qui y règnent, dépend le bon déroulement de celui-ci. Selon la formule consacrée, il est impossible de citer toutes les personnes qui, à différents niveaux, ont contribué à établir une atmosphère à la fois studieuse et amicale. Je les prie de bien vouloir m'en excuser. Donc je remercie tous mes amis de l'ICOS de l'ENIS ainsi que mes amis et mes collègues de PRISME de Bourges.

Table des matières

Table des figures	vii
Liste des tableaux	x
INTRODUCTION GENERALE	1
1 Approche	2
2 Motivation	4
3 Contribution	6
4 Contenu des chapitres	9
5 Publications :	11
5.1 Travaux publiés	11
5.2 Acceptés pour publications	13
5.3 Travaux en cours de révision	13
I État de l’art	14
I.1 Les Systèmes multi-agents coopératifs	15
I.2 Synchronisation	17
I.2.1 La synchronisation des systèmes robotiques	18
I.2.1.1 Synchronisation externe des systèmes multi-robots	20
I.2.1.2 Synchronisation mutuelle des systèmes multi-robots	21
I.3 Algorithmes de consensus.	22

I.4	Concept de stabilité et outils mathématiques	23
I.5	Formalisme de Lagrange	26
I.5.1	Équation d'Euler-Lagrange	26
I.5.1.1	Modèle d'un robot manipulateur	27
I.5.1.2	Propriétés du modèle dynamique	27
I.6	Commande nonlinéaire	28
I.6.1	Commande linéarisante et Dynamique inverse	28
I.6.2	Commande adaptative	30
I.6.3	Technique du backstepping	31
I.6.4	Commande neuronale adaptative	35
II	Synchronisation distribuée de systèmes de Lagrange sans Leader	36
II.1	Introduction	36
II.2	Technique de coordination : cross-coupling	38
II.3	Synchronisation mutuelle par cross-coupling	39
II.4	Synchronisation adaptative	44
II.5	Synchronisation en présence de délais de communication	47
II.6	Simulation	51
II.7	Conclusion	56
III	Synchronisation distribuée de systèmes Lagrange avec leader	57
III.1	Introduction	57
III.2	La théorie des graphes	59
III.2.1	Notions de base	59
III.2.2	Concept en théorie des graphes	60
III.3	Coordination non adaptative par backstepping et théorie des graphes . . .	62
III.4	Coordination adaptative neuronale	67
III.5	Coordination en présence de retard de communication	70

III.6 Simulation	72
III.6.1 Contrôle non adaptatif	73
III.6.2 Contrôle adaptatif	76
III.7 Conclusion	79
IV Synchronisation des systèmes de Lagrange sous-actionnés	80
IV.1 Introduction	80
IV.2 Systèmes multi-robots sous actionnés et groupes coopératifs	81
IV.3 Formulation du problème	83
IV.4 Architecture de la stratégie de contrôle de la synchronisation coopérative .	86
IV.4.1 Transformation de la représentation dynamique sous actionnée en une structure en chaîne	86
IV.4.2 Commande backstepping non linéaire basée sur la passivité	88
IV.5 Simulation	94
IV.6 Conclusion	96
V Mise au point d'une plateforme expérimentale	97
V.1 Introduction	97
V.2 Conception mécanique	97
V.3 Conception du système électronique	100
V.4 Conception logicielle et protocole de communication	101
V.5 Résultats d'expérimentation	101
V.6 Conclusion	104
Annexe 1	105
Annexe 2	110
Conclusions	118

Table des figures

1	Troupeau d'oiseaux.	2
2	Groupe de poissons.	2
3	Manipulation et Pliage Coopératifs.	6
4	Coordination pour Transport et Assemblage.	7
I.1	Coopération et synchronisation des robots industriels.	15
I.2	Coopération des systèmes multi-agents.	16
I.3	Synchronisation externe sous architecture maître-esclaves	20
I.4	Synchronisation mutuelle sous une architecture partiellement maillé	21
I.5	La Topologie de communication : La règle du voisin le plus proche	22
I.6	Schéma d'un réseau de neurones RBF.	35
II.1	La technique de cross-coupling	39
II.2	Synchronisation des robots en absence de toute incertitude	52
II.3	Synchronisation des robots en présence des incertitudes	53
II.4	Les erreurs de position en présence des incertitudes	53
II.5	Les erreurs de synchronisation en présence des incertitudes	54
II.6	Synchronisation des robots en absence de retard	54
II.7	Synchronisation des robots en présence de retard	54
II.8	Simulateur 3D de robots	55
III.1	Graphe non orienté	60
III.2	Graphe orienté	61

III.3 Architecture de réseau leader-suiveurs.	62
III.4 Modélisation du réseau sous forme de graphe.	63
III.5 Topologie leader-suiveur.	72
III.6 Synchronisation et suivi de trajectoire.	74
III.7 Synchronisation des vitesses.	74
III.8 Erreur de suivi.	75
III.9 Erreur de synchronisation.	75
III.10 synchronisation avec un retard de 0.5s	75
III.11 synchronisation avec un retard de 1s	76
III.12 synchronisation avec un retard de 5s	76
III.13 Synchronization adaptative du réseau de robots.	77
III.14 Synchronization adaptative des vitesses.	78
III.15 Erreurs de suivi : cas adaptatif.	78
III.16 Erreurs de synchronisation : cas adaptatif.	78
IV.1 Robot sous-actionné	83
IV.2 Robot polaire	85
IV.3 Configuration des sous systèmes interconnectés en cascade	92
IV.4 Trajectoires des articulations q_{11}^i	95
IV.5 Trajectoires des articulations q_{21}^i	95
V.1 Système vis-écrou	98
V.2 Système poulies-courroie	98
V.3 Roue dentée	98
V.4 Plaque de guidage de la vis	99
V.5 Pince	99
V.6 Robot manipulateur	99
V.7 Module électronique	100
V.8 Interface Homme/Machine	102

V.9	Trajectoire désirée	102
V.10	Trajectoire réalisée	103
V.11	Erreur de trajectoire	103
1	Principe du pont en H	105
2	Circuits intégrés ponts en H	106
3	Module de pont en H	107
4	Diagramme temporel Locked Anti-phase	107
5	Microcontrôleur : PIC18F452	108
6	Plate forme d'expérimentation	109
1	Organigramme du code embarqué	114
2	Organigramme de connexion	115
3	Organigramme d'action	116
4	Organigramme de déconnexion	117
5	Réseau d'expérimentation	117

Liste des tableaux

III.1 Gains de contrôle (cas non adaptatif)	73
III.2 Gains de contrôle (cas adaptatif)	77
1 Les types de trames conçues	111
2 Trame de connexion	111
3 Trame d’acquiescement de validation	111
4 Trame de déconnexion	111
5 Description des champs de trames Robots-PC	113
6 Description des champs de trames PC-Robots	113

INTRODUCTION GENERALE

La robotique est une science vaste et interdisciplinaire comprenant plusieurs domaines de recherche. En effet, c'est une science à la croisée de plusieurs disciplines scientifiques et techniques (mécanique, électronique, informatique ...). Au cours des dernières décennies, la robotique a connu un énorme développement. En particulier, les robots manipulateurs se sont largement développés dans les milieux manufacturiers et poursuivent leurs développements dans d'autres milieux tels que celui de la chirurgie.

Au cours du siècle dernier, la tendance à l'accomplissement de travaux complexes et multitâches s'est augmentée. La volonté de trouver des substituts pour les humains dans des environnements dangereux et pour des activités fatigantes a été l'une des principales motivations pour la recherche de systèmes autonomes et sécurisés. Aujourd'hui, l'exigence sur l'efficacité et la qualité dans les processus de production a entraîné la présence de systèmes de production complexes et intégrés. De tels processus nécessitent une grande manœuvrabilité assurée généralement par des systèmes robotiques de hautes performances pour une variété d'usage pratiquement infinie : assemblage, transport, peinture, soudure... Cependant, l'action d'un seul robot reste toujours limitée, soit du point de vue de l'espace de travail, soit du point de vue de l'espace temporel. De ce fait, l'étude des systèmes multi-robots est devenue une préoccupation majeure pour la communauté de la recherche en robotique. L'impact des systèmes multi-robots se manifeste dans l'accomplissement de tâches que le "seul robot" est incapable de réaliser. Ainsi, il est remarqué que vu leur environnement réel et pratique, les systèmes multi-robots peuvent être considérés différemment des autres systèmes distribués. Dans un contexte plus général, les études des systèmes multi-agents et de leurs contrôles coopératifs ont fait naître un domaine de recherche très prisé. Nous avons pu remarquer que, l'idée de la coopération des systèmes multi-agents a été largement inspirée de la nature. Les figures 1 et 2 illustrent deux exemples de systèmes multi-agents vivants : les nuées d'oiseaux et les groupes de poissons.

D'autre part, depuis quelques décennies, la synchronisation des systèmes mécaniques a pris une grande importance et a attiré beaucoup d'attention, particulièrement dans l'ob-



FIGURE 1 – Troupeau d'oiseaux.



FIGURE 2 – Groupe de poissons.

jectif de faire coopérer deux machines. Néanmoins les systèmes mécaniques ne sont pas les seuls cas d'application pour lesquels la synchronisation joue un rôle important. Dans les systèmes de communication, la synchronisation est fondamentale pour une transmission réussie. La synchronisation chaotique est utilisée pour améliorer l'efficacité au niveau des systèmes émetteurs-récepteurs [9].

1 Approche

L'étude des systèmes multi-robots nous conduit essentiellement vers l'analyse des lois de coordination et de coopération entre les robots. La synchronisation, la coordination et la coopération sont souvent des sujets intimement liés, surtout dans les systèmes mécaniques où elles sont utilisées comme synonymes pour décrire le même genre de comportement. Toutefois, dans la littérature, on note parfois la présence d'un concept de coordination légèrement différent de celui de la coopération des systèmes multi-robots. D'après Rodriguez [51], dans la topologie de coopération, sont considérées les intercon-

nexions entre tous les robots formant le réseau des agents en coopération, et par suite, tous les robots ont une influence directe sur la dynamique combinée du système multi-robots. A la différence avec la topologie de coopération, dans la topologie de coordination, il existe seulement des interconnexions entre le maître et les esclaves. Par conséquent, pour les schémas de coordination, le robot dominant détermine les mouvements synchronisés des autres robots en réseau. La coopération pour les systèmes multi-robots est définie par *Farinelli et al.* [49] comme un groupe de robots qui fonctionnent ensemble pour accomplir une tâche globale, alors que la coordination est une coopération dans laquelle les actions réalisées par chaque agent robotique prennent en compte les actions exécutées par les autres robots de manière que l'ensemble finisse par atteindre un fonctionnement cohérent et de haute performance. En ce qui concerne la topologie, la coordination ou la coopération des robots peuvent se présenter sur deux architectures différentes : une architecture centralisée et une architecture décentralisée.

Dans les systèmes centralisés [51], [69], [80] une unité de base recueille et gère les informations au sein du réseau afin de coordonner et de contrôler les mouvements des robots et c'est à elle de garantir la réalisation correcte de la mission. Donc, dans les approches centralisées, l'unité centrale joue un rôle fondamental en gérant l'ensemble du système : la coordination des informations reçues par les capteurs distribués, la gestion de l'information globale de l'environnement, la prise de toutes les décisions éventuelles et la communication avec tous les robots de l'équipe. Ainsi, cette unité de base devrait être assez puissante pour satisfaire toutes les exigences technologiques. De nombreuses recherches ont été investies dans les approches dites leader-suiveur [27], [39], [45], [47], [48], [52]. En fait, se sont des orientations qui adoptent des topologies de conception de base centralisées.

Par contre, dans les approches décentralisées, les ressources sont réparties entre tous les robots en coopération. Chaque robot utilise ses propres capteurs afin d'extrapoler l'information locale de l'environnement et les positions relatives de ses coéquipiers voisins et par suite prendre ses propres décisions. Ainsi, chaque robot peut communiquer et partager des informations uniquement avec les robots disposés juste à proximité, selon des approches [11], [12], [14], [15], [62] comme il peut aussi communiquer et partager ses données avec tous les robots du réseau selon d'autres approches [10], [51], [69].

Les avantages et les inconvénients des réseaux centralisés et décentralisés ont fait l'objet de plusieurs discussions dans la communauté scientifique. Les réseaux centralisés, par exemple, peuvent gérer l'information globale de l'environnement et optimiser la coordination entre les robots et par suite la réalisation de la mission. D'ailleurs, ils peuvent facilement gérer les fautes de quelques-uns. D'autre part, l'unité centrale peut constituer une faiblesse du système. En fait, elle est sollicitée à la fois par les besoins de temps de calcul et ceux de communication et d'ailleurs sa faute éventuelle conduit à une destruction

de l'ensemble du système. Par contre, les systèmes décentralisés permettent de profiter de tous les avantages de la détection et de l'actionnement répartis, à savoir, permettre d'utiliser des robots moins puissants et d'utiliser des capteurs moins coûteux. Ils permettent d'optimiser l'allocation des ressources et de doter les robots de la même équipe avec un équipage de commande, d'actionnement et de capteurs différents. De plus, la décentralisation du contrôle offre une meilleure tolérance aux pannes qui peuvent affecter un agent du réseau que des mécanismes reposant sur une unité centrale. Dans des travaux récents, afin d'exploiter largement les avantages des deux architectures, on a pu remarquer qu'un nombre importants d'auteurs s'intéressent à avoir des architectures hybrides [4], [11], [13], [23]. En effet, dans la pratique, de nombreux systèmes ne sont pas strictement centralisés ou décentralisés.

Concernant les approches de contrôle, la littérature présente plusieurs méthodes qui traitent la coordination et la coopération des systèmes de Lagrange, telles que le contrôle basé sur le comportement [44], [64] l'approche de la structure virtuelle [81], [82], la stratégie de leader-suiveur [13], [19], [57], [65] la technique de "cross-coupling" [15], [28], [29], [63], [85] et les lois qui se basent sur la théorie des graphes [30], [31], [38], [55], [56], [62]. La philosophie suivie de l'approche à structure virtuelle est de considérer toute la formation comme une seule entité et la trajectoire désirée sera affectée à cette structure qui retrace la trajectoire de chaque agent. Toutefois, cette topologie reste centralisée et pourrait éprouver des difficultés dans certaines applications. Concernant la stratégie basée sur le comportement, il s'agit d'une prescription de plusieurs comportements souhaités et la loi de contrôle dérivera finalement suivant une pondération qui prend en compte l'importance relative de chaque comportement. Cependant, la description de la dynamique du groupe ainsi que la stabilité de système global restent des difficultés majeures qui pourraient être rencontrées en utilisant cette approche [57]. La technique de cross-coupling consiste à coupler les erreurs de position et de synchronisation des robots en communication directe. D'un point de vue pratique, la stratégie leader-followers reste la plus facile à mettre en œuvre.

2 Motivation

La recherche sur les systèmes multi-robots est motivée essentiellement par l'énorme besoin d'avoir un système capable de résoudre des tâches complexes, de surmonter en quelques sortes la limitation du temps et de l'espace, d'achever les missions d'une manière efficace en termes de temps et de qualité et surtout être le moins onéreux possible.

Cependant, un seul robot se trouve incapable, de toute façon, de répondre à toutes ces exigences du fait de ses limites sur des aspects aussi différents qu'espace, temps, flexibilité et capacité. D'autre part et dans des conditions bien définies, s'investir sur un seul robot capable de tout faire, limite extrêmement l'efficacité globale. L'étude de la synchronisation et du contrôle de formation est motivée par les avantages évidemment réalisés par la coordination de plusieurs robots pour atteindre un objectif. Il s'agit notamment d'augmenter la faisabilité, la précision, la robustesse, la flexibilité, l'efficacité énergétique, et surtout la probabilité de succès. Par exemple, la probabilité de succès est très élevée si un groupe d'agents en coordination est sollicité pour remplir une mission au lieu d'un seul agent. En outre, le coût pour assurer les tâches complexes sera de plus en plus réduit. Par exemple, pour les applications d'interférométrie dans les vaisseaux, l'utilisation d'une formation de microspacecraft au lieu d'un engin spatial géant peut réduire le coût de la mission et améliorer la robustesse du système ainsi que sa précision [96]. L'étude des techniques de contrôle non linéaire pour le suivi de trajectoire et la synchronisation des agents en coopération est motivée par le fait que les solutions de contrôle de coordination sont essentiellement limitées aux systèmes modélisés par la dynamique linéaire. Comme la recherche en ingénierie a pour but de repousser les limites de la technologie, il devient nécessaire d'envisager la question de la coordination des systèmes non linéaires. La recherche dans la synchronisation et la coopération des robots concerne diverses applications de haut degré d'importance telles que : l'exploration [87], la reconnaissance [88], la manipulation [69], le contrôle de vol en formation [89], le satellite [90], le contrôle de groupes de véhicules sans pilote [91], les véhicules autonomes sous-marins [92], les engins spatiaux [14], [75], la gestion de trafic de transport [94], le contrôle du trafic aérien [95], etc. Le comportement coopératif donne la souplesse et la maniabilité qui ne peuvent pas être atteintes par un système individuel. Par exemple, les robots à mains multi-doigts, les systèmes multi-robots, les plates-formes multi-actionnées [125], [117] et les systèmes maître-esclave de télé-opération [72], [78]. En médecine, les systèmes de télé-opération, d'architecture maître-esclave, utilisés dans la chirurgie présentent une très bonne prestation, particulièrement au niveau de la précision [73]. Dans les applications aérospatiales, la coordination est utilisée afin de minimiser l'erreur d'attitude dans les formations de satellites [75], [76]. Néanmoins, les systèmes mécaniques ne sont pas les seules applications pour lesquelles la synchronisation joue un rôle important. Dans les systèmes de communication, la synchronisation est utilisée afin d'améliorer l'efficacité des systèmes émetteur-récepteur. De même la synchronisation et le chaos ont été utilisés pour chiffrer les informations et par suite améliorer la sécurité des transmissions [77], [79]. Nous détaillerons, ci-après, deux exemples qui peuvent contribuer à l'industrie et dont la synchronisation est une exigence. La première application citée s'intéresse au pliage symétrique des grandes tôles métal-

liques. Une telle tâche est si complexe qu'il paraît impossible de pouvoir l'exécuter avec un seul robot car la charge à soulever est trop grande et les tôles se plieraient. Deux robots manipulateurs se chargent de la manipulation et du pliage de cette très lourde tôle. Les deux robots, dans une synchronisation parfaite, tournent la tôle en exécutant des mouvements complexes pour l'amener dans une position définie. Les tôles sont pliées, aux quatre coins, respectivement deux fois à 90° . Après chaque opération de pliage, les robots déposent la tôle sur une table tournante afin de la tourner de 90° ou 180° . A la fin de la tâche, les robots posent la cassette en acier sur un poste de réception.



FIGURE 3 – Manipulation et Pliage Coopératifs.

La seconde application concerne le transport coopératif où la synchronisation est considérée comme une exigence indispensable à la réussite de mission. Une telle application a été présentée par la NASA. Deux robots se coordonnent dans l'objectif de soulever ensemble une charge importante qu'aucun des deux ne peut soulever seul. L'algorithme de synchronisation permet aux robots de dialoguer en temps réel pour mener à bien conjointement la tâche de transport. Cette application de transport et d'assemblage fit l'objet d'un projet financé par la NASA (le projet Trestle de l'université Carnegie Mellon) visant à élaborer un mode d'assemblage autonome basé sur des équipes de robots coordonnés entre eux. Ceci constitue une voie de recherche à part entière ouvrant des perspectives larges qui s'intéressent à imaginer des robots capables de travailler ensemble pour construire des stations spatiales sur d'autres planètes et, pourquoi pas, d'autres robots.

3 Contribution

Cette thèse se concentre sur le contrôle de la synchronisation et le suivi de trajectoires des robots industriels qui sont, aujourd'hui, fréquents et importants dans les processus

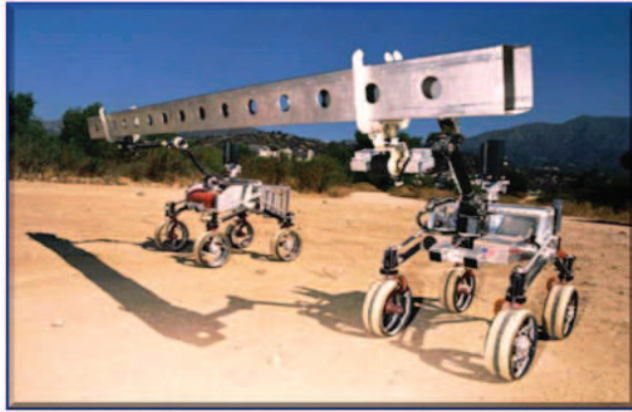


FIGURE 4 – Coordination pour Transport et Assemblage.

de production. Toutefois, la stratégie développée peut être appliquée aux systèmes de Lagrange en général.

Dans une première approche, nous considérons une architecture coopérative totalement décentralisée, modélisée par un graphe non orienté, et qui exige des interactions locales au sein d'un réseau partiellement maillé. Cette topologie ne nécessite pas de toute façon la présence d'un leader. Tout agent est un dominant en lui-même, il contribue d'une manière directe dans la dynamique combinée et par suite dans la décision finale. La stratégie développée vise à synchroniser les positions des robots en coopération en respectant le suivi d'une trajectoire désirée bien définie. Traditionnellement, la boucle de contrôle de chaque robot ne reçoit que les informations locales du robot contrôlé et vise à réaliser la tâche de suivi souhaitée sans tenir compte des états des autres robots. Toutefois, la commande de synchronisation proposée, se base sur deux types de communication : implicite et explicite. En effet, la boucle de contrôle de chaque robot reçoit des informations des propres capteurs de ce robot afin de répondre à un premier objectif : suivre une trajectoire. Elle reçoit aussi les informations provenant des capteurs de ses coéquipiers afin de construire sa base de données et par suite répondre à l'exigence du groupe comme un second objectif à atteindre. Grâce à des interactions locales, cette stratégie de contrôle distribué peut présenter des performances plus efficaces, en particulier en présence de perturbations externes. L'approche proposée repose sur une combinaison de la méthode directe de Lyapunov et les concepts de cross-coupling. La technique de cross-coupling permet à la boucle de contrôle de chaque agent de construire sa commande en se basant sur ses propres données et les données fournies par les autres agents du réseau dans l'objectif d'avoir une meilleure coordination. Par rapport aux approches existantes, les principales contributions de cette approche peuvent être formulées comme suit :

1. La présente stratégie traite des systèmes fortement non linéaires, alors que les réf-

rences [64], [27], [43], pour n'en citer que quelques-unes, traitent uniquement des modèles dynamiques simples.

2. Contrairement aux travaux antérieurs [30], [31], [55], [56], [62] l'approche proposée remplit non seulement la synchronisation globale asymptotique des variables de configuration, mais aussi la convergence asymptotique globale à la trajectoire désirée.

3. La théorie proposée est étendue afin de résoudre les problèmes d'incertitude au niveau des paramètres physiques des systèmes de Lagrange ainsi que les soucis de délais (retard) de communication au sein du réseau.

4. Dans [28], [63], [85], l'erreur de synchronisation de la position de chaque robot est définie comme l'erreur de position différentielle entre ce robot et ses deux robots adjacents. Cependant, la conception proposée permet l'interconnexion entre tous les robots, de telle sorte que tous les robots aient une influence directe dans la dynamique combinée. En conséquence, le comportement du système global est le résultat d'interactions de tous les robots. En outre, il est facile de montrer que le choix particulier de la surface de l'erreur de suivi dans l'approche proposée fournit une convergence exponentielle si nous avons une surface constante de cette erreur de suivi.

Dans une deuxième approche, une stratégie de commande décentralisée basée sur la technique de backstepping est proposée. Comme une contribution majeure, la loi de coordination développée est considérée comme une approche distribuée hybride. En effet, la stratégie proposée combine une topologie leader-suiveur et l'approche décentralisée. En fait, en se basant sur le concept de la théorie des graphes, chaque robot suiveur n'a besoin que d'un échange d'informations avec ses voisins les plus proches et ne suppose pas l'existence du canal de communication direct avec le leader du groupe, ce qui atténue, évidemment, le coût de mise en œuvre, surtout dans le cas des réseaux surchargés. L'objectif de la loi de commande conçue est de synchroniser les positions et les vitesses des suiveurs par rapport au leader. Dans une deuxième étape, la loi de contrôle est étendue pour faire face à l'incertitude des paramètres physiques des robots, une version adaptative est alors développée. Dans une étape plus évoluée, l'approche proposée prend en considération les retards au niveau des canaux de communication. A titre de comparaison avec [10], [51], [69], on note que la contribution proposée réussit à éliminer le couplage maillé et par conséquent permet de diminuer le nombre de variables à gérer dans le réseau. La contribution de ce travail, par rapport aux stratégies leader-suiveur classiques proposées dans [42], [69] plus du fait qu'elle permet les retours de données suiveur-leader, permet aussi à chaque suiveur de se limiter à de simples connections avec ses proches et n'exige pas une liaison directe avec le leader du groupe. Au contraire des travaux qui traitent l'incertitude au niveau des paramètres physiques des agents en réseau [12], [63], la loi proposée ne se base pas dans sa résolution par l'approche adaptative classique mais elle a

recours à l'approche adaptative neuronale. Le choix de la commande neuronale adaptative évite toute une analyse et le calcul des matrices de régression. L'étude de délais dans les systèmes multi-agents reste toujours un challenge à surmonter, une solution est apportée dans notre approche par rapport à d'autres approches très récentes dans la littérature [1], [2], [4], [11], [23], [63]. La synchronisation des robots sous-actionnés est un domaine de recherche d'actualité. D'après nos connaissances, Il n'y a pas encore de résultat concret dans ce contexte. Cette thèse propose une nouvelle stratégie de contrôle de synchronisation pour une classe des robot-manipulateurs sous-actionnés. Les systèmes sous-actionnés ont jusqu'à présent été étudiés cas par cas, en raison de la difficulté à mettre en évidence des propriétés structurelles suffisamment générales et exploitables pour la synthèse. En se basant sur les travaux de [24] et afin de contourner les problèmes causés par le sous actionnement, le modèle dynamique de chaque robot manipulateur sous-actionné est transformé en un système chaîné. Une combinaison de la commande passive et de la technique du backstepping est appliquée sur le système chaîné dans l'objectif d'assurer la synchronisation d'une classe de systèmes sous-actionnés.

4 Contenu des chapitres

Outre cette introduction qui fait office à la fois de présentation de la thématique générale du sujet, des approches présentées dans la littérature, des motivations et des contributions, cette thèse se compose de cinq chapitres organisés comme suit :

Chapitre I Dans ce chapitre, on présente un aperçu sur les travaux antérieurs, dans différents contextes, et auxquels nous avons eu recours : les systèmes multi-robots coopératifs, la synchronisation des systèmes mécaniques, les algorithmes de consensus des systèmes multi-agents. Nous avons introduit quelques concepts du formalisme de Lagrange, la notion de stabilité au sens de Lyapunov et des outils mathématiques comme la commande linéarisante, la dynamique inverse, la commande passive, la commande adaptative, la technique de backstepping et la commande adaptative neuronale.

Chapitre II Ce chapitre discute le principe de la synchronisation sans leader. Une conception d'une loi de commande totalement décentralisée est proposée. L'approche de contrôle se base essentiellement sur la technique de cross-coupling. Sous une topologie de réseau partiellement maillée, la stratégie conçue a pour objectif de synchroniser les mouvements d'un groupe de robots totalement actionnés en suivant une trajectoire commune. Le réseau de robots coopératifs nécessite juste des échanges d'informations entre

ses agents. Dans ce cas, il n'y a pas un leader de groupe. Afin d'avoir un sens plus pratique, la stratégie développée est étendue pour résoudre les incertitudes paramétriques et les problèmes de retards purs dus aux canaux de communications.

Chapitre III Ce chapitre s'intéresse au principe de la synchronisation distribuée avec "leaders". Deux stratégies sont étudiées dans ce contexte. Une première stratégie qui traite l'approche de la synchronisation externe basée sur la technique de "cross-coupling". Dans cette approche, la communication se fait seulement entre leader-suiveurs (aussi appelés "leader-followers") et dans une seule direction. Dans cette topologie, il n'y a pas un retour d'information des suiveurs (aussi appelés "followers") vers le leader, ni un échange de données follower-follower. Le leader est le seul qui connaît la trajectoire désirée, les followers doivent suivre cet agent dominant. Une telle architecture est simple à implanter, diminue le flux d'échange de données et par suite les problèmes de communication au sein du réseau, mais elle n'est pas assez robuste lorsque la topologie du réseau est partiellement maillée. Dans cette architecture, le cas d'incertitudes au niveau des paramètres structurels des robots est aussi étudié. Des lois d'adaptation sont conçues afin d'assurer la coordination.

Une deuxième stratégie de contrôle plus développée est conçue : elle est basée sur la théorie des graphes et l'algorithme de consensus. Une architecture leader-suiveur présentant des canaux de communications non orientés entre leader-suiveur et suiveur-suiveur est proposée. Elle permet, de plus, un retour d'information followers-leader, ainsi qu'une communication non orientée suiveur-suiveur, apportant plus de robustesse au comportement synchronisé du groupe. Le leader est toujours l'agent dominant qui est le seul dans lequel la trajectoire désirée est implantée. La loi de contrôle est conçue par une commande non linéaire se basant sur la technique de backstepping. Une approche neuronale est développée afin de résoudre l'incertitude au niveau des paramètres physiques des robots en réseau. Une étude est traitée pour aboutir à une solution. Une extension du contrôleur a été développée, dans l'objectif de prendre en compte le retard d'information à travers les canaux de communication. Une architecture innovante par rapport à celle présentée au chapitre précédent est proposée : le retard est considéré seulement au niveau des informations arrivant à un robot et provenant des capteurs de ses voisins, tandis que les informations venant de ses propres capteurs sont traitées instantanément. Une telle approche allège le calcul au niveau de l'unité de commande de chaque agent et par conséquent, elle diminue la vitesse d'exécution.

Chapitre IV Ce chapitre s'intéresse à une problématique d'actualité sur laquelle les recherches sont peu nombreuses : la synchronisation des systèmes de Lagrange sous-actionnés. Dans une nouvelle approche, le problème de suivi de trajectoires et de synchronisation d'un réseau de robots sous-actionnés est traité. La synchronisation de ce type de systèmes est basée sur l'utilisation de la passivité comme outil de conception pour résoudre le problème. Le contrôle coopératif est conçu en utilisant une combinaison de la procédure standard du backstepping et la théorie de la passivité et par suite les sorties de synchronisation sont construites de manière récursive, puis partagées au sein du groupe. La technique de conception proposée est basée sur un graphe orienté fortement connecté. Des simulations numériques sont effectuées pour illustrer l'efficacité de l'approche proposée.

Chapitre V Ce chapitre décrit la mise en place d'une plateforme d'expérimentation. En effet, trois robots manipulateurs identiques à quatre degrés de libertés ont été développés afin de remplir un réseau de robots coopératifs. Ce chapitre fournit une description de la conception mécanique, de la conception des cartes électroniques de commande et de puissance, du choix des capteurs et des moteurs et surtout de l'implémentation numérique.

5 Publications :

Ce qui suit est une liste des publications que nous avons rédigées au cours des trois dernières années, qui sont soit publiées, soit acceptées pour publications, soit en cours de révision.

5.1 Travaux publiés

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Nonlinear control and synchronization with time delays of multi-agent robotic systems ; Journal of Control Science and Engineering, 2011.

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Distributed synchronization control to trajectory tracking of multiple robot manipulators ; Journal of Robotics, 2011.

Bouteraa Y., Ghommam J., Poisson G. ; Distributed Backstepping Control for Synchronization of Networked class of Underactuated Systems : A Passivity Approach; 19th IEEE Mediterranean Conference on Control and Automation, Aquis Corfu Holiday Palace, Corfu, Greece, June 20-23 2011.

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; On the trajectory tracking and coordination of multi-robot systems with communication delay; IEEE International Multi-Conference on Systems, Signals and Devices SSD'11, Mars 22-25, 2011, Sousse - Tunisia.

Bouteraa Y., Ghommam J. ; Coordinated backstepping control of multiple robot system of the leader-follower structure; IEEE International Multi-Conference on Systems, Signals and Devices SSD'11, Mars 22-25, 2011, Sousse - Tunisia.

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Adaptive Multi-Robots Synchronization; Proceedings of the 2010 IEEE International Symposium on Industrial Electronics, July 4-7, 2010, Bari, Italy.

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Mutual and external synchronization control of multi-robot systems; IEEE International Multi-Conference on Systems, Signals and Devices SSD'10, June 27-30, 2010, Amman - Jordan.

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Adaptive Synchronization control of multi-robot teams : Cooperative and Coordinated schemes; 18th IEEE Mediterranean Conference on Control and Automation Congress Palace Hotel, Marrakech, Morocco, June 23-25, 2010.

Bouteraa Y., Ghommam J. ; Synchronization Control of Multiple Robots Manipulators; IEEE International Multi-Conference on System, Signals and Devices SSD'09, March 23-26 2009, Tunisia, Djerba.

Bouteraa Y., Derbel N., Poisson G. ; Nonlinear cooperative control and synchronization of multi-robot systems; Journée des Jeunes Chercheurs PRISME, 16 juin 2011, Orléans, France.

5.2 Acceptés pour publications

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Nonlinear adaptive synchronization control of multi-agent robotic systems ; International Journal of Systems, Control and Communications, 2010.

5.3 Travaux en cours de révision

Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Adaptive Backstepping Neural Network for Networked Lagrangian Systems ; submitted in International Journal of Control, Automation and Systems, springer.

Chapitre I

État de l'art

I.1 Les Systèmes multi-agents coopératifs

Les progrès rapides de l'ingénierie des systèmes informatiques, de détection (capteurs) et des technologies de communication ont apporté une contribution majeure dans le développement des systèmes autonomes. En effet, ces progrès ont participé à l'ouverture de nouveaux défis pour concevoir et développer la coopération des systèmes multi-agents. Un système multi-robot peut être défini comme un ensemble de robots fonctionnant dans le même espace de travail. Les systèmes multi-robot peuvent accomplir des tâches qu'un seul robot se trouve incapable de réaliser. En effet, un seul robot, est un agent spatialement limité, qui n'a pas toujours les "compétences" requises pour mener à bien une tâche à haut niveau de complexité. Ainsi, le contrôle des systèmes multi-robots est actuellement une préoccupation majeure de la recherche en robotique. Les études dans la coordination des systèmes multi-agents ont attiré une grande attention de nombreuses équipes de recherche dans les dernières décennies. Les premières études sur les systèmes multi-robots ont été proposées par Fukuda [98]. D'autres efforts de "pionniers" dans les systèmes multi-robots proviennent de l'université de Stanford et du projet particulier GOFER [99]. L'objectif de ce projet était de contrôler les opérations de plusieurs dizaines des robots mobiles dans un environnement intérieur. Ce projet a soulevé de nombreuses questions de recherche : mise en œuvre des systèmes de détection et de résolution des conflits, développement des systèmes de communication et des protocoles utilisés afin d'assurer l'interaction homme-robot et robot-robot, génération de la planification de mouvement et programmation des tâches pour les systèmes multi-robots. En fait, les études des systèmes multi-agents sont passées par deux phases importantes et reconnues. La première étape a été traitée lors des deux dernières décennies du siècle précédent. A ce stade, la recherche était focalisée sur la coopération des robots manipulateurs (robots industriels).

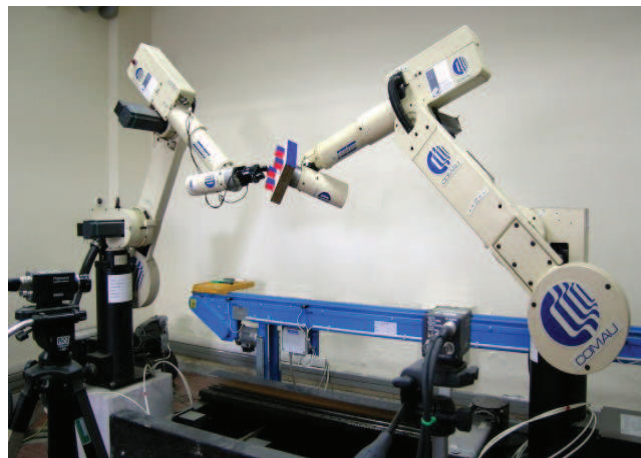


FIGURE I.1 – Coopération et synchronisation des robots industriels.

L'autre voie a été lancée vers le début des années 2000. Elle s'intéresse essentiellement aux contrôles de formation des robots mobiles, aux algorithmes de consensus et aux systèmes distribués. Dans une première phase, ce qui a motivé les études de coordination des robots manipulateurs a été la grande exigence d'avoir des systèmes coopératifs capables de remplir des tâches complexes d'assemblage et de transport dans les applications industrielles et spatiales. Dans ce contexte, trois architectures principales ont été rapportées dans la littérature. La première architecture est celle de la stratégie maître/esclave [121], [122]. La deuxième architecture est celle de la stratégie de contrôle centralisé [107], [108], [109], [110]. La troisième architecture est celle de l'approche décentralisée où chaque agent est contrôlé par son propre contrôleur [111], [112], [114]. Depuis la fin des années 90, des études sur les systèmes multi-agents, présentant plus d'évolutivité, sont devenues nécessaires pour garantir l'exécution avec succès de tâches complexes effectuées dans des milieux dynamiques et incertains. Ces études ont fourni des solutions pour divers problèmes : surtout d'exploration [32], [46], [87], [105], de coopération et de reconnaissance [88], de manipulation [51], [63], [69] de coordination des systèmes de Lagrange [14], [15]. Elles ont apporté des solutions pour du contrôle de vol en formation [89], du contrôle de satellites [90], du contrôle de groupes de véhicules sans pilote [91], des véhicules autonomes sous-marins [92] ou des engins spatiaux [14], [75], de la gestion de trafic de transport [94], du contrôle du trafic aérien [95], des missions de risque [101], [104], des tâches d'assemblage [85], des applications militaires [100], de la navigation des navires autonomes sous-actionnées [8], [20], etc.

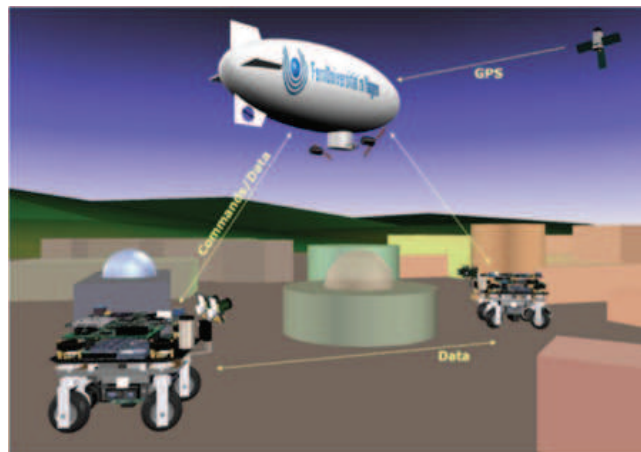


FIGURE I.2 – Coopération des systèmes multi-agents.

I.2 Synchronisation

Selon [115] la synchronisation peut être définie en tant que conformité d'horloge mutuelle de deux processus ou plus. Cette conformité peut être caractérisée par l'apparence de certaines relations entre les fonctionnalités des processus. En se basant sur les types d'interconnexions dans le système et par suite sur les interactions des agents, on peut définir différents genres de synchronisations. Dans la littérature, on parle de la synchronisation naturelle : c'est le cas de la synchronisation des systèmes débranchés, qui présentent un comportement synchronisé. D'autre part, on parle de la synchronisation individuelle : c'est le cas de la synchronisation réalisée par des interconnexions propres au système et sans aucune intervention externe. Un exemple typique de synchronisation naturelle se manifeste dans les mouvements des corps célestes, telle que la rotation des satellites autour des planètes. En revanche, lorsqu'il existe des actions externes (des entrées de contrôle ou des interconnexions artificielles) afin d'achever la synchronisation d'un système multi-agents, une telle synchronisation est dénommée synchronisation contrôlée. A titre illustratif de ce type de synchronisation, nous citons le fait que la plupart des applications pratiques de la théorie de synchronisation telles que les systèmes d'émission réception et les oscillateurs synchronisés dans le domaine de la communication. Pour ce dernier genre de synchronisation, suivant la formulation du contrôleur, on distingue deux classes : une synchronisation interne ce qu'on appelle la synchronisation mutuelle et une synchronisation externe. Dans le cas de la synchronisation mutuelle, tous les multi-agents synchronisés se produisent sur un pied d'égalité dans la dynamique unifiée du système multi-composé. Par conséquent, le mouvement synchrone se produit comme résultat de l'interaction de tous les éléments du système. C'est le cas des oscillateurs synchronisés couplés ou des systèmes coopératifs. En ce qui concerne la synchronisation externe, on considère toujours qu'un agent dans le système multi-composé est plus puissant que les autres et que son mouvement reste indépendant. Cet agent dominant gère le mouvement synchronisé de tout le système et les autres n'ont pas d'impact direct sur la décision finale. C'est le cas des systèmes dits maître-esclaves. D'après Rodriguez [51], le plus intéressant dans le problème du contrôle de la synchronisation c'est de savoir comment on arrive à concevoir un contrôleur ou à remplir les interconnexions convenables capables d'assurer la synchronisation respectant les fonctionnalités désirées. La conception de ces interconnexions se base essentiellement sur le retour d'état des variables ou des signaux qui définissent le mouvement synchronisé désiré.

I.2.1 La synchronisation des systèmes robotiques

Les robots manipulateurs sont très souvent utilisés dans les processus industriels qui exigent une flexibilité et une manœuvrabilité de grande importance.

En effet, l'industrie s'est de plus en plus développée et a connu l'apparition de nouvelles techniques de production (travail en chaîne, production par lot). Une production exigeante du point de vue qualité et quantité a imposé l'appel à des systèmes, avec un espace de travail plus large et qui peuvent achever plusieurs tâches en même temps. Dans une telle situation, un seul robot est incapable de remplir le cahier de charge désirée. Une limitation due à la complexité de la tâche ou à un espace de travail assez réduit crée une grande nécessité d'utiliser des systèmes multi-robots. Le travail en synchronisation est une bonne alternative pour résoudre ces limitations. La synchronisation externe (cas du schéma de coordination maître-esclaves) et la synchronisation mutuelle (cas des robots coopératifs décentralisés) sont deux illustrations importantes pour le même but. L'objectif dans les deux types de synchronisation est d'avoir un système robotique travaillant en synchronisation afin d'achever un cahier de charge commun [117], [125].

De ce cas, la topologie choisie peut être vue comme une solution à un problème de commande, dans lequel on cherche le contrôleur capable de générer les mouvements synchronisés. Dans la synchronisation des systèmes robotiques on parle de certains problèmes fondamentaux qui peuvent être récapitulés dans trois points : premièrement, concevoir les fonctions sur lesquelles se base essentiellement la synchronisation. Pour la synchronisation des robots, ces fonctions peuvent être définies comme la norme de la différence des erreurs couplées : position et vitesse, appelées les variables d'intérêt. Dans les problèmes de suivi de trajectoire, par exemple, on définit deux fonctions essentielles pour la synchronisation : une fonction qui décrit l'erreur de position et l'autre qui indique l'erreur de synchronisation [4], [7], [12], [14], [15]. Ensuite, on développe le contrôleur par retour d'état et les interconnexions assurant la synchronisation des mouvements. Les interconnexions sont définies comme un retour d'état décrivant la différence entre les variables d'intérêt. Finalement, c'est de voir et imposer la condition qui garantit la synchronisation. Dans la littérature [51], [63], le problème de synchronisation pour les robots semble être une extension simple de celui du suivi de la trajectoire. Cependant, le problème de la synchronisation implique des défis qui ne sont pas considérés dans la conception des contrôleurs de suivi de la trajectoire.

Autres que les problèmes fondamentaux, la synchronisation englobe d'autres types de problèmes. Pour avoir une grande performance, le contrôleur de synchronisation doit tenir compte de la perte d'information lors des échanges des données, le retard de transmission [40], [56], de la nature des robots (cas des robots sous actionnés), etc. Pour avoir plus de tolérance et pour être plus flexible et plus pratique, ce contrôleur doit prendre en considération les équipements utilisés pour mesurer les différents paramètres comme la position et la vitesse.

I.2.1.1 Synchronisation externe des systèmes multi-robots

Considérons un système multi-robots formé d'un groupe de robots pour lequel, un robot possède un comportement indépendant des autres. Ce robot est considéré comme un maître ou un leader. Ce robot principal est conduit par un contrôleur déjà conçu et non approprié pour le but de la synchronisation. Dans le cas idéal des problèmes de suivi de trajectoire, le contrôleur assure la convergence des positions angulaires et des vitesses du robot principal (maître) à une trajectoire désirée. Dans les problèmes de synchronisation, l'objectif est de concevoir des interconnexions et par suite un contrôleur décentralisé à implémenter dans chaque robot esclave. Ce contrôleur assure la synchronisation des positions et des vitesses de chaque robot esclave avec les positions et les vitesses du robot maître. Dans ce contexte, Rodriguez *et al.*[69] proposent une architecture de synchronisation externe pour un groupe de robots industriels. C'est une architecture dont la position (aussi bien pour le maître que pour l'esclave) est le seul état accessible pour chaque robot interconnecté. L'information est délivrée dans un seul sens du maître vers tous les esclaves et ne permet aucun retour d'état de n'importe quel esclave vers le maître. L'objectif de l'approche est de concevoir des interconnexions entre les robots esclaves dans le but de suivre la trajectoire du robot maître dans une synchronisation parfaite des points de vue positions et vitesses. Notons que la stratégie proposée a pour objectif de suivre la position et la vitesse du robot dominant et non pas sa trajectoire désirée. Par ailleurs, l'architecture de contrôle ignore la dynamique du maître et se base seulement sur l'hypothèse d'avoir le modèle dynamique des esclaves ; ce qui complique bien le fait de la reconstruction de la vitesse angulaire et l'accélération du robot dominant.

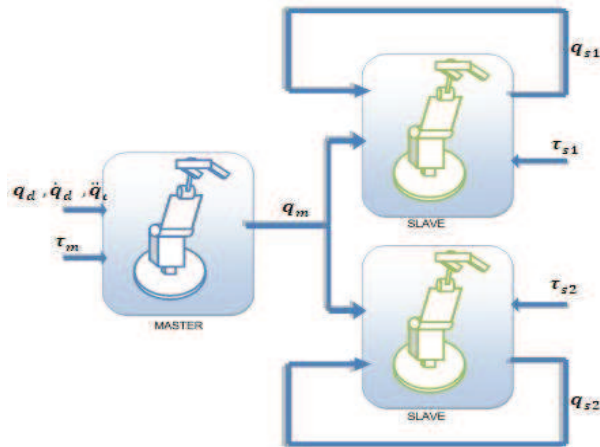


FIGURE I.3 – Synchronisation externe sous architecture maître-esclaves

I.2.1.2 Synchronisation mutuelle des systèmes multi-robots

Vu qu'elle est amenée seulement par le robot dominant, la synchronisation externe restreint le comportement résultant de la synchronisation. Un système multi-robot sera plus polyvalent et plus efficace lorsque toutes les interconnexions entre ses robots seront prises en compte, et non pas seulement celles qui sont entre le robot dominant et les autres robots esclaves. Dans ce cas, le comportement résultant de la synchronisation est le produit des informations explicites et implicites de tous les robots du système. Par conséquent, tout robot du système participe, par les informations qu'il fournit, à la génération de la dynamique combinée.

Une approche remarquable sur la synchronisation mutuelle, en considérant une limitation au niveau des capteurs de vitesse, est développée dans [51]. Cette approche a conduit à la création d'un contrôleur de synchronisation mutuelle qui fait que seules les informations sur les positions des robots coopératifs au sein du réseau sont utiles. Ce contrôleur s'appuie sur toutes les informations venant de tous les robots du système. Un tel contrôleur qui se base sur tous les robots du système afin de déterminer une décision finale offre une grande tolérance de précision au niveau des capteurs.

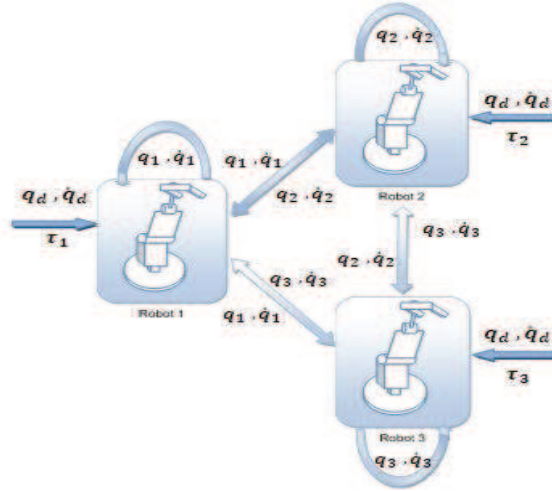


FIGURE I.4 – Synchronisation mutuelle sous une architecture partiellement maillé

D'autres travaux plus récents [12], [15] ont proposé un contrôleur considérant seulement la communication entre chaque robot et ses voisins les plus proches. Un tel réseau se propage. Ces derniers peuvent être considérés plus faciles à mettre en œuvre, tant du point de vue charge de calcul que de celui des signaux exigés. Une approche de synchronisation pour le suivi de trajectoire de plusieurs robots mobiles conservant une formation bien définie au cours du temps est proposée dans [63]. L'idée principale est de contrôler chaque robot pour suivre sa trajectoire désirée tout en synchronisant son mouvement avec les autres

robots dans l'objectif de garder des relations cinématiques exigées par le contrôle de la formation. Une étude sur la synchronisation mutuelle et la convergence exponentielle vers le consensus du groupe de systèmes de Lagrange est discutée dans [12].

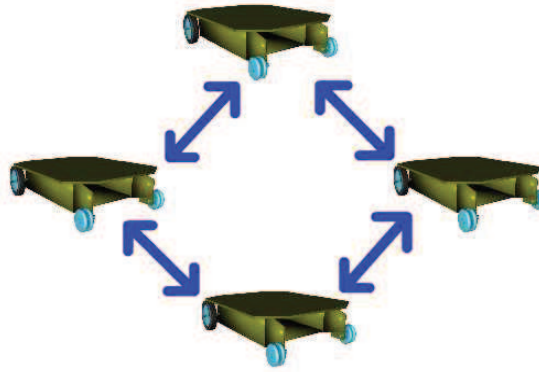


FIGURE I.5 – La Topologie de communication : La règle du voisin le plus proche

I.3 Algorithmes de consensus.

Le contrôle distribué des systèmes multi-agents a reçu une attention considérable ces dernières années [10], [93], [118]. Dans certaines applications de la distribution multi-agents, le partage de l'information joue un rôle primordial et facilite la coordination du groupe. En conséquence, un problème critique pour le contrôle de coordination est de concevoir des protocoles et des algorithmes appropriés tels que le groupe d'agents puissent parvenir à un consensus sur le partage de l'information en présence de l'évolution dynamique de la topologie d'interaction. La convergence vers une valeur commune est appelée un consensus ou un accord. Les problèmes de consensus ont une histoire dans la littérature de la science informatique [118], ils ont attiré récemment l'attention de plusieurs chercheurs [43], [62]. En effet, les algorithmes de consensus cherchent à savoir comment une équipe d'agents, à travers la négociation avec ses voisins, peut parvenir à un accord sur une valeur commune. Dans [62] l'auteur propose des conditions suffisantes afin d'avoir un consensus assurant la coordination d'un groupe d'agents mobiles et autonomes sous une topologie de graphe d'interaction non orienté. Toujours sous une architecture du graphe d'interaction orienté, Ren *et al.* [43] montrent les conditions nécessaires et suffisantes à la détermination d'un consensus d'information pour les systèmes multi-véhicules.

Généralement ces algorithmes sont utilisés afin de développer des stratégies de coordination pour des systèmes multi-agents. Récemment, ils sont largement étudiés dans le cadre du contrôle des systèmes coopératifs [6], [10], [47], [56]. La philosophie des algorithmes de consensus exige seulement un échange d'informations locales entre les agents voisins. L'idée est que, en se basant sur l'information venant de ses voisins et qui est souvent variable au cours du temps, chaque agent essaie de reformuler son statut de façon que l'état final de chaque agent du groupe finisse par converger vers une valeur commune. La plupart des travaux qui traitent les algorithmes de consensus ont été focalisés sur les algorithmes dont la dynamique est de premier ordre [67], [86]. Les extensions de ces algorithmes qui traitent la dynamique de deuxième ordre ont été rapportées dans la littérature dans des applications de contrôle de formation et évitement d'obstacles [10], [31]. Comme extension des algorithmes de second ordre, Ren [10] propose un algorithme qui évolue selon une référence nonlinéaire, seulement connue par quelques agents du groupe.

I.4 Concept de stabilité et outils mathématiques

Quelques définitions de base et la notion de stabilité au sens de Lyapunov sont rappelées brièvement.

Considérons un système nonlinéaire autonome représenté par l'équation différentielle suivante :

$$\dot{x} = f(t, x) \quad (\text{I.1})$$

où $t \in \mathbb{R}_{\geq 0}$ représente le temps, $x \in \mathbb{R}^n$ est l'état et $f : \mathbb{R}_{\geq 0} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$. Nous supposons que l'origine $x = 0$ est un point d'équilibre pour le système (I.1). Pour étudier la stabilité du point d'équilibre $x = 0$, nous introduisons les définitions suivantes :

Définition 3.1 *Le point d'équilibre $x = 0$ de (I.1) est dit **localement stable** s'il existe une constante positive $r > 0$ tel que pour tout $(t_0, x_0(t_0)) \in \mathbb{R}_{\geq 0} \times \mathcal{B}_r$ il existe une fonction $\alpha \in \mathcal{K}$ telle que*

$$\|x(t)\| \leq \alpha(\|x(t_0)\|), \quad \forall t_0 \geq 0, \forall x(t_0) \in \mathcal{B}_r \quad (\text{I.2})$$

*Si l'inégalité (I.2) est vérifiée pour tout $(t_0, x_0(t_0)) \in \mathbb{R}_{\geq 0} \times \mathbb{R}^n$ alors l'origine est **globalement stable**.*

Définition 3.2 Le point d'équilibre $x = 0$ de (I.1) est dit

- **(Localement) asymptotiquement stable** s'il existe une constante $r > 0$ telle que pour toute paire $(t_0, x_0(t_0)) \in \mathbb{R}_{\geq 0} \times \mathcal{B}_r$, il existe une fonction $\beta \in \mathcal{KL}$ telle que

$$\|x(t)\| \leq \beta(\|x(t_0)\|, t - t_0), \quad \forall t \geq t_0 \geq 0, \forall x(t_0) \in \mathcal{B}_r \quad (\text{I.3})$$

- **Semi-globalement asymptotiquement stable** Si pour toute constante $r > 0$ et pour toute paire $(t_0, x_0(t_0)) \in \mathbb{R}_{\geq 0} \times \mathcal{B}_r$, il existe une fonction $\beta \in \mathcal{KL}$ telle que (I.3) est vérifiée.
- **Globalement asymptotiquement stable (GAS)** s'il existe une fonction $\beta \in \mathcal{KL}$ telle que pour toute paire $(t_0, x_0(t_0)) \in \mathbb{R}_{\geq 0} \times \mathbb{R}^n$ l'inégalité (I.3) est vérifiée.

Définition 3.3 Le point d'équilibre $x = 0$ de (I.1) est dit **(localement) exponentiellement stable** s'il est (localement) asymptotiquement stable et (I.3) est vérifiée avec :

$$\beta(r, s) = kre^{-\gamma s}, \quad k > 0, r > 0, \gamma > 0$$

Dans le même contexte, nous pouvons définir pour le point d'équilibre $x = 0$ de (I.1) la notion de **semi globalement exponentiellement stable** et **globalement exponentiellement stable (GES)**

Définition 3.4 Le point d'équilibre $x = 0$ de (I.1) est dit **uniformément stable** s'il existe une constante positive $r > 0$ et une fonction $\alpha \in \mathcal{K}$ indépendantes de t_0 , telle que $\forall t \geq t_0 \geq 0, \forall x(t_0) \in \mathcal{B}_r$

$$\|x(t)\| \leq \alpha(\|x(t_0)\|) \quad (\text{I.4})$$

Si l'inégalité (I.4) est vérifiée pour tout $(t_0, x(t_0)) \in \mathbb{R}_{\geq 0} \times \mathbb{R}^n$, alors l'origine est **globalement uniformément stable**

Définition 3.5 Le point d'équilibre $x = 0$ de (I.1) est dit **(localement) uniformément et exponentiellement stable / semi-globalement et uniformément exponentiellement stable / globalement et uniformément exponentiellement stable (GUES)**, s'il est (localement) uniformément et asymptotiquement stable / semi-globalement et uniformément asymptotiquement stable / globalement et uniformément asymptotiquement stable et (I.4) est satisfaite avec

$$\beta(r, s) = kre^{-\gamma s}, \quad k > 0, r > 0, \gamma > 0$$

Pour prouver les propriétés de stabilité d'un système, de nombreuses méthodes sont basées sur le théorème de Lyapunov [113], qui est énoncé comme suit :

Théorème 3.1 Soit $x = 0$ un point d'équilibre de (I.1) et soit le domaine $D \subset \mathbb{R}^n$ contenant $x = 0$. Soit la fonction $V : D \rightarrow \mathbb{R}^n$ de class $\mathcal{C}^1$, telle que

$$\begin{aligned} V(0) &= 0 \text{ et } V(x) > 0 \text{ dans } D \\ \dot{V}(x) &\leq 0 \text{ dans } D \end{aligned}$$

alors $x = 0$ est stable. De plus, si

$$\dot{V}(x) < 0 \text{ dans } D - \{0\}$$

alors $x = 0$ est asymptotiquement stable. Notons que pour avoir une stabilité asymptotique du point d'équilibre $x = 0$, il est nécessaire de prouver que $\dot{V}(x) < 0$ dans $D - \{0\}$. Cela dit, il existe d'autres théorèmes qui permettent de montrer la stabilité asymptotique quand $\dot{V}(x) \leq 0$. Pour les systèmes autonomes, il est possible de prouver la stabilité asymptotique quand $\dot{V}(x) \leq 0$ en considérant le théorème de LaSalle, alors que pour les systèmes non-autonomes le Lemme de Barbalat est employé pour prouver la stabilité asymptotique.

Théorème 3.2 (Théorème de LaSalle), [113] Le point d'équilibre $\bar{x} = 0$ de (I.1) est asymptotiquement stable s'il existe une fonction $V(x) : D \rightarrow \mathbb{R}$ continûment différentiable ayant les propriétés suivantes :

- D est un ouvert de $\mathbb{R}^n$ et $\bar{x} \in D$;
- $V(x) > V(\bar{x}) \quad \forall x \neq \bar{x} \text{ dans } D$ ($V(x)$ est minimum en $\bar{x}$)
- $\dot{V}(x) \leq 0, \quad \forall x \in D$
- l'ensemble $S \subset D$ tel que $\dot{V}(x) = 0$ ne contient pas de trajectoires du système autre que $x(t) = \bar{x}$.

Lemme 3.1 (Lemme de Barbalat), [113] Soit $\phi : \mathbb{R} \rightarrow \mathbb{R}$ une fonction uniformément continue sur $[0, \infty)$. Supposons que $\lim_{t \rightarrow \infty} \int_0^t \phi(\tau) d\tau$ existe et est finie. Alors $\phi \rightarrow 0$ quand $t \rightarrow \infty$.

Définition 3.6 (L'approche de Lyapunov-Krasovskii) Cette approche est considérée comme une extension de la théorie de Lyapunov appliquée aux systèmes de type retardé. Nous présentons ici brièvement les résultats trouvés par Kharitonov pour construire une fonctionnelle de Lyapunov-Krasovskii et qui sont utilisés dans le chapitre 2 et le chapitre 3. Pour une synthèse de la stabilité des systèmes dynamiques à retard, le lecteur est appelé à consulter cette référence [36].

Théorème 3.4 ([71]) Soit le système dynamique suivant :

$$\dot{x}(t) = Ax(t) + Bx(t - \tau) \quad (\text{I.5})$$

Où A et B sont des matrices de dimensions appropriées et τ est un retard constant. Si le système (I.5) est asymptotiquement stable alors l'expression :

$$\begin{aligned} V(x_t) = & x^T(t)U(0)x(t) + 2x^T(t) \int_{-\tau}^0 U(-\tau - \theta)Ax(t + \theta)d\theta \\ & + \int_{-\tau}^0 x^T(t + \theta_2)A^T \int_{-\tau}^0 U(\theta_1 - \theta_2)Ax(\tau + \theta_1)d\theta_1d\theta_2 \\ & + \int_{-\tau}^0 x^T(t + \theta) [(\tau + \theta)R_1 + \omega_1] x(t + \theta)d\theta \end{aligned} \quad (\text{I.6})$$

est une fonction de de Lyapunov-Krasovskii pour le système (I.5).

Avec

$$u(\tau) = \int_0^\infty K^T(t) [w_0 + w_1 + \tau R_1] K(t + \tau)dt. \quad (\text{I.7})$$

et w_0 , w_1 et R_1 sont définies comme des matrices positives. $K(t)$ est une matrice fondamentale du système, tels que K est une solution de l'équation retardée suivante :

$$\dot{K}(t) = AK(t) + BK(t - \tau) \quad (\text{I.8})$$

dont les conditions initiales sont $K(0) = I_n$ et $K(t) = 0$ pour $t < 0$.

I.5 Formalisme de Lagrange

La thèse est consacrée à étudier les stratégies de synchronisation pour les systèmes de Lagrange. Nous introduisons dans la section (III.2) les systèmes de Lagrange et leurs propriétés principales. Les équations d'Euler Lagrange peuvent être utilisées pour modéliser une classe de systèmes mécaniques, y compris les robots manipulateurs.

I.5.1 Équation d'Euler-Lagrange

Introduisons les modèles dynamiques des robots. Plusieurs moyens sont utilisés afin de développer de tels modèles dynamiques. A titre d'exemple, la formulation lagrangienne ou hamiltonienne de la mécanique classique [129], [130] qui conduisent à une description fondée sur les équations différentielles. Basée sur une représentation en réseaux de systèmes physiques, une approche différente est assurée par la technique de Bond-Graph [127], [128], dont le caractère est très utile en mécatronique. Dans cette thèse, les modèles dynamiques sont basés sur une approche Lagrangienne.

I.5.1.1 Modèle d'un robot manipulateur

Considérons un robot manipulateur à n articulations rigides. Soit $q \in \mathbf{R}^n$ un vecteur colonne représentant les coordonnées des articulations rigides. Le robot est supposé pleinement actionné. L'énergie cinétique de ce robot est donnée par la relation suivante :

$$T(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (\text{I.9})$$

avec $M \in \mathbf{R}^{n \times n}$, une matrice symétrique définie positive, représentant la matrice d'inertie. L'énergie potentielle du système due à la gravité est notée $U(q)$. En appliquant le formalisme d'Euler-Lagrange [124], le modèle dynamique du robot sera présenté comme suit :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \quad (\text{I.10})$$

avec $g(q) = \frac{\partial}{\partial(q)} U(q) \in \mathbf{R}^n$ représente les forces de gravité, $C(q, \dot{q})\dot{q} \in \mathbf{R}^n$ représente les forces de Coriolis et de Centrifuge et τ est un vecteur $\in \mathbf{R}^n$ représentant les couples exercés par les actionneurs. Le modèle dynamique ne tient pas compte des effets dynamiques, tels que générés par le frottement, donc le modèle (I.10) est désigné comme nominal et sans friction.

I.5.1.2 Propriétés du modèle dynamique

Le modèle dynamique (I.10) possède des propriétés qui sont souvent utilisées dans l'étude de la stabilité. Nous présentons dans ce paragraphe, brièvement, ces propriétés que nous exploiterons dans le chapitre 2.

1. La matrice d'inertie $M \in \mathbf{R}^{n \times n}$ est symétrique et positive, définie pour tout $q \in \mathbf{R}^n$.
2. Si la matrice est définie en utilisant les symboles de Christoffel [124], alors la matrice $\dot{M}(q) - 2C(q, \dot{q})$ est une matrice antisymétrique et par suit pour tout $x \in \mathbf{R}^n$:

$$x^T (\dot{M}(q) - 2C(q, \dot{q})) x = 0 \quad (\text{I.11})$$

3. La matrice $C(q, \dot{q})$ s'écrit la forme suivante :

$$C(q, \dot{q}) = \begin{pmatrix} \dot{q}^T C_1(q) \\ \vdots \\ \dot{q}^T C_n(q) \end{pmatrix} \quad (\text{I.12})$$

Tels que $C_i(q, \dot{q}) \in \mathbf{R}^{n \times n}$; $i = 1, \dots, n$ est une matrice symétrique [126] et par suite pour tout $\alpha \in \mathbf{R}$ et pour tout $q, x, y, z \in \mathbf{R}^n$

$$\begin{aligned} C(q, x)y &= C(q, y)x \\ C(q, z + \alpha x)y &= C(q, z)y + \alpha C(q, x)y \end{aligned} \quad (\text{I.13})$$

4. $M(q)$, $C(q, \dot{q})$ et $g(q)$ sont bornées et vérifient les inégalités suivantes :

$$\begin{aligned} 0 < M_m \leq \| M(q) \| &\leq M_M \quad \forall q \in \mathbf{R}^n \\ \| g(q) \| &\leq g_M \quad \forall q \in \mathbf{R}^n \\ xC(q, x)y &= C(q, y)x \quad \forall q, x \in \mathbf{R}^n \end{aligned} \quad (\text{I.14})$$

5. La paramétrisation linéaire des équations dynamiques : l'équation d'Euler-Lagrange (I.10) est linéaire par rapport au paramètres structurels θ , d'où

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) = Y(q_i, \dot{q}_i, \ddot{q}_i)\theta_i \quad (\text{I.15})$$

Avec $\theta_i \in R^a$ est un vecteur contenant les paramètres structurels du robot et a est le nombre des paramètres inconnus.

I.6 Commande nonlinéaire

Nous introduisons brièvement dans ce paragraphe quelques approches de contrôle des systèmes non linéaires étudiées dans cette thèse. La commande par l'approche dynamique inverse a été utilisée pour le contrôle décentralisé d'un groupe de robots sans leaders. L'approche adaptative a été utilisée pour le contrôle de réseau en présence de l'incertitude. Tandis que le backstepping est la technique de conception du contrôleur distribué non linéaire de synchronisation et de suivi de trajectoire pour un réseau de robots avec leader. L'approche neuronale adaptative a été retenue pour la conception du contrôleur adaptatif. On a recours à l'approche de passivité dans la conception de la stratégie de contrôle des groupes de robots sous-actionnés.

I.6.1 Commande linéarisante et Dynamique inverse

La technique la plus classique de stabilisation des systèmes pleinement actionnés consiste à transformer un système nonlinéaire en un système linéaire par un changement

adéquat de coordonnées ainsi que par quelques transformations au niveau de l'entrée de commande. L'idée de base de la linéarisation est simple, le principe est d'éliminer les effets de la nonlinéarité de l'équation originale et par suite avoir un comportement dynamique linéaire. En seconde étape, le concepteur peut concevoir sa commande et tenant compte des nouvelles coordonnées d'états afin de remplir quelques spécifications du type : suivi de trajectoire, rejet de perturbation, synchronisation de mouvement... En fait pour les systèmes étudiés dans cette thèse, l'approche de la linéarisation généralise le concept de dynamique inverse. Dans ce qui suit un exemple de conception d'un contrôleur de suivi de trajectoire est donné. Il se base sur le principe de dynamique inverse. Considérons le modèle dynamique d'un robot manipulateur de n articulations (I.10). L'objet de la dynamique inverse est de chercher une loi de contrôle non linéaire :

$$u = f(q, \dot{q}, t) \quad (\text{I.16})$$

de façon que lorsque l'on substitue (I.16) en (I.10), on obtiendra un système linéaire en boucle fermée. C'est une approche de contrôle qui peut être difficile et même impossible à mettre en œuvre dans certains systèmes non linéaires. Cependant, pour les robots (I.10), le problème est assez facile à résoudre. Un examen rapide de l'équation de la dynamique d'un robot manipulateur motive le choix de contrôle suivant :

$$u = M(q)a_q + C(q, \dot{q})\dot{q} + g(q) \quad (\text{I.17})$$

La combinaison de (I.10) et (I.17) réduit le problème à l'égalité suivante :

$$\ddot{q} = a_q \quad (\text{I.18})$$

De ce fait, a_q est considérée comme étant la nouvelle entrée du système. Il est clair que la combinaison de (I.10) et de (I.17) a donné un système linéaire de second ordre (I.18). Ce système est découplé dans le sens que chaque élément a_{qk} du vecteur de contrôle a_q est considéré comme un contrôle du système scalaire linéaire. Autrement dit, supposons que a_{qk} est fonction de q_k et de ses dérivées seulement, alors a_{qk} affectera q_k indépendamment des autres articulations et de leurs comportements. Puisque a_{qk} est conçu pour contrôler un système linéaire du second ordre, alors le choix le plus simple est :

$$a_q = -k_0 q - k_1 \dot{q} + r \quad (\text{I.19})$$

avec k_0 et k_1 deux matrices diagonales qui représentent les gains de position et les gains de vitesse, respectivement et $r(t)$ est une entrée de référence. Etant donné une trajectoire désirée : $t \rightarrow (q^d(t), \dot{q}^d(t))$. Choisissons une entrée de référence comme suit :

$$r(t) = q^d(t) + k_0 q^d(t) + k_1 \dot{q}^d(t) \quad (\text{I.20})$$

L'erreur de suivi est modélisé par :

$$e = q(t) - q^d(t) \quad (\text{I.21})$$

Une combinaison de (I.20) et (I.21) donne l'équation suivante :

$$\ddot{e}(t) + k_1 \dot{e}(t) + k_0 e(t) = 0 \quad (\text{I.22})$$

Et par suite le système est représenté sous un modèle linéaire du second ordre.

L'approche de la dynamique inverse est extrêmement importante dans la résolution des problèmes de contrôle des robots manipulateurs. On peut recourir à cette approche autrement. Considérons la dynamique d'un robot manipulateur I.10. Puisque la matrice d'inertie $M(q)$ est inversible pour tout $q \in \mathbf{R}^n$, nous pouvons procéder de la manière suivante :

$$\ddot{q} = M^{-1}(u - C(q, \dot{q})q - g(q)) \quad (\text{I.23})$$

La dynamique inverse peut être vue comme une entrée de transformation qui transforme le problème de choix d'un couple en choix d'une accélération comme entrée de commande. Donc supposons que nous avons la possibilité d'avoir l'accélération comme entrée du contrôle du système. Cette hypothèse revient à supposer que nous avons des actionneurs capables de produire directement une accélération et non pas indirectement à travers les couples. L'analyse précédente produit l'équation suivante :

$$\ddot{q}(t) = a_q(t) \quad (\text{I.24})$$

avec $a_q(t)$ le vecteur d'accélération. En réalité, nous n'avons pas la possibilité d'avoir des actionneurs produisant des accélérations. Effectivement, les actionneurs produisent des couples u_i pour chaque articulation i . Mais une simple comparaison entre I.23 et I.24 nous donne la relation suivante qui relie l'accélération au couple :

$$M^{-1}(u(t) - C(q, \dot{q})q - g(q)) = a_q \quad (\text{I.25})$$

La résolution de $u(t)$ produit la loi de contrôle suivante :

$$u(t) = M(q)a_q + C(q, \dot{q})\dot{q} + g(q) \quad (\text{I.26})$$

I.6.2 Commande adaptative

L'approche adaptative [4], [63], [78] est un choix naturel en présence d'incertitudes. Dans cette thèse, on a recours à l'approche adaptative classique pour examiner l'incertitude au niveau des paramètres physiques des robots. En se basant sur la propriété de la linéarisation des paramètres, le modèle dynamique d'un robot i peut s'écrire comme suit :

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) = Y_i(q_i, \dot{q}_i, \ddot{q}_i)\theta_i \quad (\text{I.27})$$

avec Y la matrice de la régression ; $\theta_i \in \mathbf{R}^a$ le vecteur des paramètres structurels du robot et a est le nombre des paramètres inconnus. Une relation similaire à (I.27) se présente en cas de l'incertitude aux niveaux des paramètres physiques du système :

$$\hat{M}_i(q_i)\ddot{q}_i + \hat{C}_i(q_i, \dot{q}_i)\dot{q}_i + \hat{g}_i(q_i) = Y_i(q_i, \dot{q}_i, \ddot{q}_i)\hat{\theta}_i \quad (\text{I.28})$$

où $\hat{M}_i$, $\hat{C}_i$ et $\hat{g}_i$ sont, respectivement, des estimations des matrices M_i , C_i et g_i . L'objectif est de concevoir un contrôleur τ_i afin de faire tendre l'erreur de suivi et l'erreur de synchronisation vers zéro. La loi conçue sur la base de la dynamique inverse prend la forme générale suivante :

$$\tau_i = \hat{C}_i(q_i, \dot{q}_i) + \hat{g}_i(q_i) + \hat{M}_i(u_T + u_s) \quad (\text{I.29})$$

avec u_T un contrôleur de suivi et u_s un contrôleur de synchronisation. Soit e l'erreur globale du système englobant l'erreur de suivi et l'erreur de synchronisation. L'étape la plus importante dans la conception d'une loi adaptative est le choix de la loi d'adaptation. Dans notre cas, on propose la loi suivante :

$$\dot{\hat{\theta}}_i = -\Gamma_i^{-1}(\hat{M}_i^{-1}Y_i)^T \dot{e}_i \quad (\text{I.30})$$

où Γ_i est une matrice diagonale définie positive.

I.6.3 Technique du backstepping

Dans ce paragraphe nous introduisons la technique du backstepping utilisée dans la conception de la stratégie de contrôle présentée dans le chapitre V concernant les réseaux avec leader. La technique du Backstepping est une méthode de calcul récursive dans le but de concevoir une loi de contrôle des systèmes non linéaires. L'idée de base consiste à fragmenter le système en sous systèmes. Les états de ces sous systèmes seront considérés dans une seconde étape comme étant des contrôleurs virtuels imbriqués d'ordre décroissant. La technique exige plusieurs étapes et se base sur un calcul récursif des fonctions de Lyapunov. Chaque étape, en se basant sur la notion de la commande virtuelle et l'ajout des intégrateurs, a pour objectif de stabiliser un sous système et à la dernière étape la loi est vérifiée.

Comparée à la commande par linéarisation, la technique du backstepping ne recourt pas à l'annulation des non linéarités, ce qui améliore bien les réponses aux régimes transitoires. Notons que le système étudié doit s'écrire sous une forme paramétrique. Prenons, comme exemple explicatif, le problème de la stabilisation du système non linéaire sous la forme

triangulaire suivante :

$$\begin{aligned}
 \dot{x}_1 &= x_2 + f_1(x_1) \\
 \dot{x}_2 &= x_3 + f_2(x_1, x_2) \\
 &\bullet \\
 &\bullet \\
 \dot{x}_i &= x_{i+1} + f_i(x_1, x_2, \dots, x_i) \\
 &\bullet \\
 &\bullet \\
 \dot{x}_n &= f_n(x_1, x_2, \dots, x_n) + u
 \end{aligned} \tag{I.31}$$

Donc la procédure commence ici par considérer l'état x_2 comme une commande virtuelle de l'état x_1 . Et par suite l'objectif est de trouver la possibilité de rendre $\dot{x}_2 = -x_1 - f_1(x_1)$ afin de respecter la stabilité de l'état x_1 . Cet objectif pourra être vérifié si nous considérons la fonction de Lyapunov de la forme suivante : $V_1 = \frac{1}{2}x_1^2$. Or, en réalité, x_2 n'est pas la vraie commande de x_1 . D'où nous désignons le changement de variables suivant :

$$z_1 = x_1$$

$$z_2 = x_2 - \alpha_1(x_1)$$

avec $\alpha(x_1) = f_1(x_1)$.

Tenons compte de l'expression de la fonction de Lyapunov $V_1 = \frac{1}{2}Z_1^2$, nous obtenons :

$$\dot{z}_1 = -z_1 + z_2$$

$$\dot{z}_2 = x_3 + f_2(x_1, x_2) - \alpha_1(x_1)(x_2 + f_1(x_1)) = x_3 + \overline{f_2}(z_1, z_2)$$

En procédant récursivement, nous définissons l'erreur z_3 et on augmente la candidate de Lyapunov de la forme suivante :

$$z_3 = x_3 - \alpha_2(z_1, z_2)$$

$$V_2 = V_1 + \frac{1}{2}z_2^2$$

Afin de remplir l'expression de α_2 , on se base sur les expressions des dérivées par rapport au temps de l'erreur z_2 et de la deuxième fonction de lyapounov :

$$\dot{z}_2 = z_3 + \alpha_2(z_1, z_2) + \bar{f}_2(z_1, z_2)$$

$$\dot{V}_2 = -z_1^2 + z_2(z_1 + z_3 + \alpha_2(z_1, z_2) + \bar{f}_2(z_1, z_2))$$

Choisissons

$$\alpha_2(z_1, z_2) = -z_1 - z_2 - \bar{f}_2(z_1, z_2)$$

ce qui résulte :

$$\dot{z}_1 = -z_1 + z_2$$

$$\dot{z}_2 = -z_1 z_2 + z_3$$

$$\dot{V}_2 = -z_1^2 - z_2^2 + z_2 z_3$$

Récursivement, à l'étape i , on définit :

$$z_{i+1} = x_{i+1} - \alpha_i(z_1, \dots, z_i)$$

$$V_i = \frac{1}{2} \sum_{k=1}^i z_k^2$$

Afin de produire :

$$\dot{z}_i = z_{i+1} + \alpha_i(z_1, \dots, z_i) + \bar{f}_i(z_1, \dots, z_i)$$

$$\dot{V}_i = -\sum_{k=1}^{i-1} z_k^2 + z_{i-1} z_i + z_i(z_{i+1} + \alpha_i(z_1, \dots, z_i) + \bar{f}_i(z_1, \dots, z_i))$$

Posons

$$\alpha_i(z_1, \dots, z_i) = -z_{i-1} - z_i - \bar{f}_i(z_1, \dots, z_i)$$

Ce qui résulte :

$$z_i = -z_{i-1} - z_i + z_{i+1}$$

$$v_i = -\sum_{k=1}^i z_k^2 + z_i z_{i+1}$$

Ceci produit à l'étape n :

$$\dot{z}_n = \bar{f}_n(z_1, \dots, z_n) + u$$

Donc, pour un choix de :

$$u = \alpha_n(z_1, \dots, z_n) = -z_{n-1} - z_n - \bar{f}_n(z_1, \dots, z_n)$$

et une fonction de Lyapunov :

$$V_n = \frac{1}{2} \sum_{i=1}^n z_i^2$$

On aura les variables suivantes :

$$\dot{z}_n = -z_{n-1} - z_n$$

$$\dot{v}_n = -\sum_{i=1}^n z_i^2$$

Le développement qui suit la technique du Backstepping, est de trouver une transformation globale (un difféomorphisme) permettant de passer des coordonnées en z aux coordonnées en x . La stabilité du système est prouvée en recourant à de simples fonctions de Lyapunov quadratiques. Il faut aussi noter que la dynamique obtenue en fonction des coordonnées en z est linéaire. L'avantage de la technique du Backstepping est sa flexibilité pour le choix des fonctions stabilisantes α_i qui sont choisies simplement sans éliminer toutes les non-linéarités afin de rendre la fonction $\dot{V}_i$ négative.

I.6.4 Commande neuronale adaptative

Dans le monde de l'ingénierie du contrôle, l'approche neuronale est souvent utilisée pour faire l'approximation des fonctions non linéaires données. Le problème d'approximation des fonctions peut être introduit comme suit :

Définition 3.20 Soit $f(y) : \mathbf{R}^n \longrightarrow \mathbf{R}^m$ une fonction continue et soit $\hat{f}(w, y) : \mathbf{R}^{l \times m} \times \mathbf{R}^n \longrightarrow \mathbf{R}^m$ est une fonction d'approximation. Le problème d'approximation est de déterminer le paramètre optimal w^* tel que la métrique d satisfait la relation suivante :

$$d(f(w, y), f(y)) \leq \varepsilon \quad (\text{I.32})$$

pour $\varepsilon > 0$ suffisamment petit.

Dans notre étude proposée au chapitre III, on a recours à un réseau de neurones "RBF". En fait, c'est une architecture particulière de réseau de neurones qui utilise l nombres de fonctions radial gaussiennes de la forme suivante :

$$a_i(y) = \exp \left(\frac{-(y - u_i)^T (y - u_i)}{\sigma^2} \right) \quad (\text{I.33})$$

où $u_i \in \mathbf{R}^n$ est le vecteur centre et σ^2 est la variance.

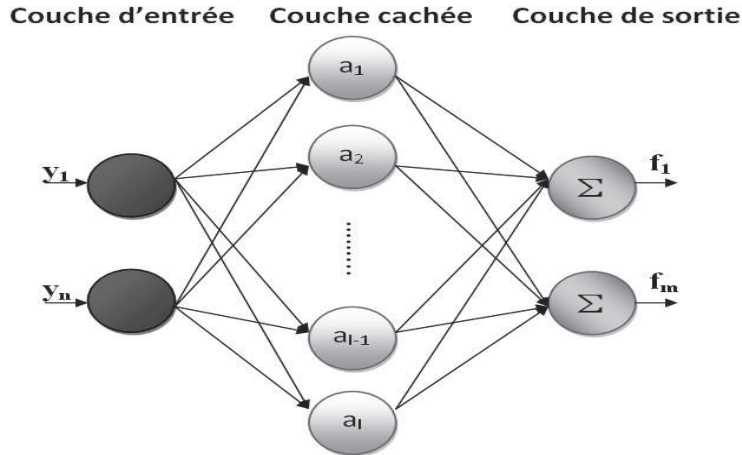


FIGURE I.6 – Schéma d'un réseau de neurones RBF.

Comme il se présente sur la figure FIG. III.4, chaque réseau RBF se compose de trois couches : la couche d'entrée, la couche cachée contenant les fonctions gaussiennes et la couche de sortie. Le réseau de neurone produit en sortie une fonction d'approximation qui se présente sous la forme suivante :

$$\hat{f}(w, y) = w^T a(y) \quad (\text{I.34})$$

avec $a(y) = [a_1(y), a_2(y), \dots, a_l(y)]^T$ le vecteur de fonctions de base.

Chapitre II

Synchronisation distribuée de systèmes de Lagrange sans Leader

Dans ce chapitre, nous proposons une architecture de contrôle totalement décentralisée afin de commander un groupe de systèmes de Lagrange coopératifs. Le but est de faire synchroniser les différents mouvements des robots pleinement actionnés, tout en suivant une trajectoire commune désirée. En effet, la stratégie proposée consiste à synchroniser les positions et les vitesses des articulations de chaque robot par rapport à ses voisins. La topologie proposée est modélisée par un graphe non orienté. La stratégie ne nécessite que des interactions locales entre robots coopératifs et n'exige aucune présence d'un leader du groupe. L'approche de synchronisation développée se base sur la technique de coordination "cross-coupling". L'étude de stabilité du système global interconnecté est assurée par la méthode de Lyapunov. La théorie est étendue dans une deuxième étape afin de traiter les problèmes d'incertitude au niveau des paramètres physiques de robots coopératifs. D'où une loi adaptative est développée. Dans une troisième étape, la stratégie de commande développée s'ouvre sur les systèmes à retard. En effet, la théorie proposée prend en considération les retards de communication dans les canaux de réseau. L'étude de stabilité de réseau est assurée par la méthode de la fonction de Krasovsky.

II.1 Introduction

Des grands efforts de recherche dans les problèmes de synchronisation et les stratégies de contrôle des systèmes coopératifs ont été lancés ces dernières années. Particulièrement les stratégies distribuées dans lesquelles les contrôleurs sont couplés et la décision de chaque robot ne se base pas seulement sur ses propres capteurs mais aussi sur les états de

ses coéquipiers [12], [62]. Les algorithmes de consensus, le contrôle de la synchronisation basé sur les techniques de passivité et la coordination distribuée sont étudiées dans [2] [3] et [10] respectivement. Dans ce contexte, on trouve des travaux récents qui se basent sur la technique de "cross-coupling" dans leurs stratégies de coordination [15], [12], [28], [51], [63]. Dans [12], [63], l'approche a recours à une topologie qui n'exige pour chaque agent que deux canaux de communication avec ses plus proches voisins. Tandis que dans [51], [69], l'approche développée se base sur un réseau partiellement maillé. Néanmoins, les approches conçues à base de la théorie de graphe ont fait des bonnes prestations [5], [55], [56]. Le contrôle adaptatif est une stratégie très efficace utilisée dans les problèmes de synchronisation en présence des incertitudes [63], [85]. Dans la littérature, bien que beaucoup des travaux s'intéressent au consensus et à la coordination des systèmes multi-agents, ces efforts se consacraient principalement sur les dynamiques simples comme les système linéaires, les systèmes du premier ordre ou du second ordre avec des matrices d'inertie non linéaire [27], [43], [86]. D'autre part, la plupart des travaux précédents dans le consensus et la coordination, particulièrement, qui utilisent la théorie des graphes comme technique de coordination [10], [33], [55], [56], [62] ont discuté une synchronisation vers la valeur moyenne pondérée de leurs conditions initiales. En fait, ces travaux cités n'ont pas considéré le cas des systèmes multi-agents qui ont une exigence d'une trajectoire désirée à suivre. L'objectif de la loi de contrôle proposée est de faire synchroniser les mouvements d'un groupe de robots industriels en suivant une trajectoire désirée commune. Donc, bien que la trajectoire commune soit implémentée dans chaque robot, les tâches ne peuvent pas être exécutées en même temps, surtout que les robots n'ont pas les mêmes points initiaux de départ. De ce fait et pour garantir une synchronisation au niveau du temps d'exécution des tâches par les différents robots, chaque robot devrait avoir une image exacte des états de ses coéquipiers. Par conséquent, à travers des interconnexions locales, la stratégie proposée peut réaliser des bonnes performances, en particulier, en présence des perturbations externes. Comparée avec les travaux présentés dans la littérature, les principales contributions de notre approche développée peuvent se résumer dans ce qui suit :

- A.** Au contraire des travaux présentés dans [27], [43], [86] l'approche proposée traite les cas de systèmes de forte non linéarité.
- B.** Comparé aux références [31], [33], [55], [56], [62] l'approche proposée réalise non seulement la synchronisation globale asymptotique des variables de configuration, mais aussi la convergence asymptotique globale à la trajectoire désirée.
- C.** La théorie est étendue afin de traiter les problèmes d'incertitude et les problèmes des délais de communication.
- D.** Dans [28], [63], [85] l'erreur de synchronisation de chaque robot est définie comme

étant la différence des erreurs de position entre ce robot et de ses deux robots adjacents. Cependant, la conception proposée permet l'interconnexion entre tous les robots, de telle sorte que tous les robots ont une influence directe dans la dynamique combinée et par conséquent le comportement synchronisé est le résultat d'interactions entre tous les robots. En outre, il est facile de montrer que le choix particulier de la surface de l'erreur de suivi dans l'approche proposée, en cas d'une surface d'erreur de suivi constante, fournit une convergence exponentielle de q_i vers q_d .

II.2 Technique de coordination : cross-coupling

La commande de synchronisation est idéalement adaptée aux systèmes multi-agents dans l'exécution d'une tâche de groupe telle que l'assemblage et le contrôle de formation. La synchronisation est développée de façon que chaque agent peut remplir sa propre tâche en synchronisant son mouvement avec ses coéquipiers. Afin de répondre à cette exigence de coordination, le système global doit maintenir un ensemble des relations cinématiques. Ce paragraphe présente un aperçu sur la technique de coordination que nous avons utilisée. Dans la majorité des commandes traditionnelles des systèmes multi-agents, la boucle de contrôle de chaque agent se base seulement sur les données de ses propres capteurs et il ne reçoit rien sur les états des autres agents. Autrement dit, le réseau se base seulement sur communication implicite et non pas une communication explicite. Par conséquent, des perturbations dans la boucle de régulation d'un agent peuvent provoquer une erreur au sein de réseau, cette erreur ne peut être corrigée que par la boucle concernée et les autres ne répondent pas, ce qui dégrade essentiellement la synchronisation du système global. La technique de couplage cross-coupling permet à la boucle de contrôle de chaque agent de recevoir un retour des autres agents afin qu'ils participent effectivement dans le comportement dynamique du système global et par suite avoir une coordination robuste. Dans le domaine de la robotique, les applications de la synchronisation ont été lancées par la synchronisation de deux roues motrices dans une application de la technique de couplage dans la commande des robots mobiles [119]. Récemment, la commande de synchronisation a été utilisée dans des applications de coordination des robots manipulateurs. Un contrôleur de synchronisation adaptative pour coordonner les mouvements d'un groupe de robots manipulateurs pour des tâches d'assemblage est développé par Sun et Mills [85]. Dans ce travail Sun et Mills ont défini les erreurs de synchronisation comme étant la différence des erreurs de positions entre chaque deux robots voisins. Rodriguez-Angeles et Nijmeijer [51] ont proposé une approche de la synchronisation mutuelle par retour d'état estimé des robots manipulateurs basé sur la technique de cross-coupling. Dans un travail

référence [14], en utilisant la théorie de cross-coupling, Chung et Slotine ont présenté une loi de commande pour la synchronisation des systèmes de Lagrange coopératifs. Sun [28] a proposé un algorithme de synchronisation en intégrant les techniques de cross-coupling dans l'architecture adaptative de son contrôleur. On présente dans la figure ci-dessous (II.1) un schéma explicatif de la technique de cross-coupling :

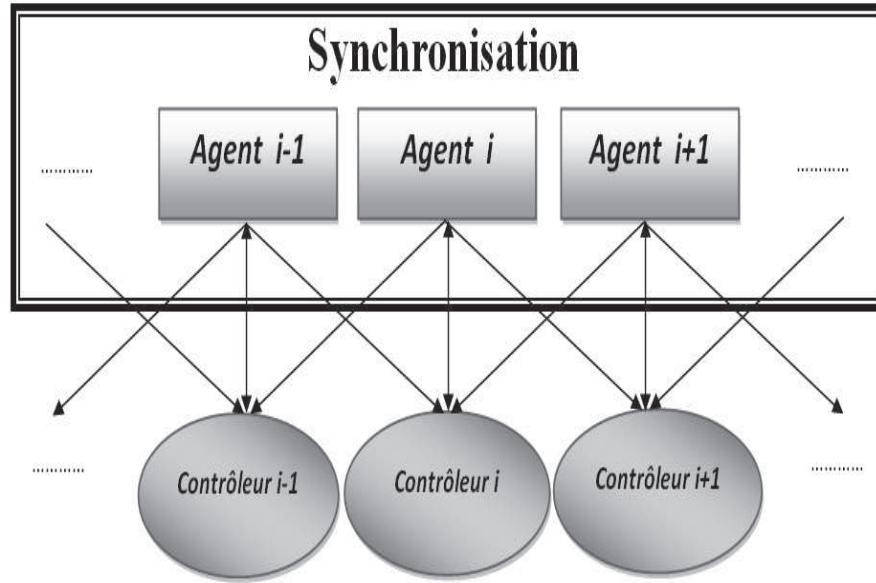


FIGURE II.1 – La technique de cross-coupling

II.3 Synchronisation mutuelle par cross-coupling

Nous concevons des lois de commande décentralisées pour p robots manipulateurs de telle sorte que toutes les positions communes puissent se synchroniser mutuellement et suivre une trajectoire commune désirée. L'objectif de la loi de contrôle proposée est de synchroniser la position et la vitesse de la i^{me} articulation $q_i, \dot{q}_i$ avec les états de chaque manipulateur. De plus le contrôleur devrait assurer le suivi de la trajectoire désirée q_d . Plus précisément, chaque contrôleur de couple est conçu afin de contrôler la convergence vers zéro de l'erreur de position, de vitesse ainsi que de l'erreur de synchronisation. Donc, afin de remplir cet objectif, nous définissons la surface de l'erreur de suivi comme suit :

$$\varepsilon_{1i}(t) = q_i(t) - q_d(t) + \int_{t_0}^t \Lambda_i [q_i(\lambda) - q_d(\lambda)] d\lambda \quad (\text{II.1})$$

où Λ_i est une matrice diagonale définie positive. En fait, les informations du vecteur ε_{1i} donnent une idée sur la convergence de l'état de l'articulation vers l'état désiré. Cependant, il est simple de montrer que ce choix particulier de cette surface de l'erreur de

suivi prévoit une convergence exponentielle de q_i à q_d en cas d'une surface d'erreur de suivi constante. Cependant l'erreur de glissement définie ne garantit que le suivi de la trajectoire. Néanmoins, il est nécessaire de connaître les performances du contrôleur et par suite de savoir comment la trajectoire de chaque robot manipulateur converge par rapport aux autres robots du groupe coopératif. Dans la littérature, Il y a plusieurs façons de choisir l'erreur de synchronisation. Par exemple, dans [51], l'approche proposée consiste à rassembler et inclure les erreurs de position de tous les systèmes impliqués dans la synchronisation. Dans [12], [15] l'erreur de synchronisation est calculée en se basant juste sur les informations provenant des deux plus proches voisins. Notre approche se base sur la technique de "cross-coupling" afin de proposer une erreur de synchronisation calculée à partir de la différence des erreurs de position et définie comme suit :

$$\varepsilon_{2i}(t) = \sum_{j \neq i}^p K_{ij}[(q_i - q_d) - (q_j - q_d)] = \sum_{j \neq i}^p K_{ij}(q_i - q_j) \quad (\text{II.2})$$

où K_{ij} est matrice symétrique et définie positive qui donne un aperçu sur la qualité de communication entre le i^{me} et le j^{ime} robot. Notons que K_{ij} est différent de zéro seulement pour les canaux de communication présentant un échange d'informations entre le robot i et le robot j . Par conséquent, la topologie est partiellement maillée. L'erreur globale pour chaque agent dans le réseau coopératif et qui englobe les deux erreurs de suivi et de la synchronisation est définie comme suit :

$$e_i = \varepsilon_{1i} + \int_{t_0}^t \varepsilon_{2i}(\lambda) d\lambda \quad (\text{II.3})$$

L'objectif est de concevoir une loi de commande telle que des erreurs de position, les erreurs de vitesse et les erreurs de synchronisation convergent toutes vers zéro. Afin de remplir cet objectif, pour chaque robot manipulateur, la loi de commande τ_i est définie comme suit :

$$\begin{aligned} \tau_i = & C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) + M_i(q_i)[\ddot{q}_d - K_{pi}e_i - K_{di}\dot{e}_i \\ & - \Lambda_i(\dot{q}_i - \dot{q}_d)] + M_i(q_i) \left[\sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i \varepsilon_{2i} \right] \end{aligned} \quad (\text{II.4})$$

Avec q_d représente la trajectoire commune à suivre et pour laquelle la première dérivée et la deuxième dérivée existent pour tout $t \geq 0$. K_{pi} et K_{di} sont des matrices symétriques et définies positives.

Théorème 4.1 *Considérons un groupe de robots manipulateurs sous une architecture complètement décentralisée. Si le $K_{di} > 2 \sum_{j \neq i} K_{ij}$, le contrôleur de suivi de trajectoire et de synchronisation proposé (II.4) garantit la convergence asymptotique vers zéro de l'erreur de position, de vitesse et de synchronisation.*

Remarque 1 : Notons que le contrôleur développé pour chaque robot exige seulement une information locale sur la position et la vitesse ainsi que l'information sur la trajectoire désirée. Donc, afin d'assurer un mouvement synchronisé avec les membres du groupe, chaque robot a besoin d'avoir une information sur les positions de ses voisins. Il doit être remarqué que les positions des robots voisins sont regroupées dans la variable de l'erreur ε_{2j} .

Preuve : En substituant (8) dans (1), on obtient :

$$\begin{aligned} M_i(q_i)\ddot{q}_i &= M_i(q_i)[\ddot{q}_d - K_{pi}e_i - K_{di}\dot{e}_i - \Lambda_i(\dot{q}_i - \dot{q}_d)] \\ &\quad + M_i(q_i) \left[\sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i \varepsilon_{2i} \right] \end{aligned} \quad (\text{II.5})$$

La multiplication par M_i^{-1} des deux cotés produit :

$$\begin{aligned} \ddot{q}_i &= \ddot{q}_d - K_{pi}e_i - K_{di}\dot{e}_i - \Lambda_i(\dot{q}_i - \dot{q}_d) \\ &\quad + \sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i \varepsilon_{2i} \end{aligned} \quad (\text{II.6})$$

Ajoutons ε_{2i} des deux cotés

$$\begin{aligned} \ddot{q}_i - \ddot{q}_d + \Lambda_i(\dot{q}_i - \dot{q}_d) + \varepsilon_{2i} &= -K_{pi}e_i - K_{di}\dot{e}_i \\ &\quad + \sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i \varepsilon_{2i} + \varepsilon_{2i} \end{aligned} \quad (\text{II.7})$$

Ce qui en résulte

$$\ddot{e}_i = -K_{pi}e_i - K_{di}\dot{e}_i + \sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i \varepsilon_{2i} + \varepsilon_{2i} \quad (\text{II.8})$$

En utilisant l'expression de l'erreur de la synchronisation ε_{2i} et sa première dérivée, nous obtenons l'équation suivante :

$$\begin{aligned} \ddot{e}_i &= -K_{pi}e_i - K_{di}\dot{e}_i + \sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) \\ &\quad + \sum_{j \neq i} [K_{ij}(\dot{q}_i - \dot{q}_d) - (\dot{q}_j - \dot{q}_d)] \\ &\quad + \sum_{j \neq i} \Lambda_i K_{ij} [(q_i - q_d) - (q_j - q_d)] \end{aligned} \quad (\text{II.9})$$

Tout calcul fait, le résultat est donné par :

$$\ddot{e}_i = -K_{pi}e_i - K_{di}\dot{e}_i + \sum_{j \neq i} K_{ij}(\dot{e}_i - \dot{e}_j) \quad (\text{II.10})$$

L'équation (II.10) représente la dynamique du système synchronisé en boucle fermée du manipulateur i . Dans la suite, afin de prouver que l'erreur de position, l'erreur de vitesse et l'erreur de synchronisation convergent vers zéro, une analyse des propriétés de la stabilité du système globale est fournie. Posons alors l'erreur globale du système multi-robot comme suit : $e^T = [e_1^T, e_2^T, \dots, e_n^T]^T$.

En utilisant (II.10), nous obtenons la dynamique de l'erreur :

$$\ddot{e} = -K_p e - (K_d - K_c)\dot{e} \quad (\text{II.11})$$

Où $K_p = \mathbf{diag}(K_{pi})$, $K_d = \mathbf{diag}(K_{di})$ et K_c est une matrice symétrique semi-définie positive donnée par :

$$\begin{pmatrix} \sum_{j \neq 1} K_{1j} & \cdots & -K_{1j} & \cdots & -K_{1n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -K_{i1} & \cdots & \sum_{j \neq i} K_{ij} & \cdots & -K_{in} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -K_{n1} & \cdots & -K_{nj} & \cdots & \sum_{j \neq n} K_{nj} \end{pmatrix}$$

Notons que la dynamique de l'erreur représentée par l'équation (II.10) est linéaire, invariante dans le temps. Cette dynamique est modélisée par une équation différentielle linéaire du second ordre. L'hypothèse d'avoir K_p et $K_d - K_c$ définies positives est une condition suffisante de la stabilité de la dynamique de l'erreur. Plus précisément, les matrices K_{di} sont des matrices diagonales satisfaisant la relation suivante : $K_{di} > 2 \sum_{i \neq j} K_{ij}$.

Donc, pour étudier la convergence nous posons la fonction de Lyapunov suivante :

$$V = \dot{e}^T \dot{e} + e^T K_p e \quad (\text{II.12})$$

La dérivée par rapport au temps de cette fonction est :

$$\dot{V} = -2\dot{e}^T (K_d - K_c)\dot{e} \leq 0 \quad (\text{II.13})$$

Une application du principe d'invariance de LaSalle montre bien que l'origine $(e, \dot{e}) = (0, 0)$ est globalement asymptotique stable et la $\lim_{t \rightarrow \infty} \dot{e} \rightarrow 0$ pour $t \rightarrow \infty$.

En se référant maintenant à l'équation de l'erreur globale (II.10) :

$$e_i = q_i - q_d + \int_{t_0}^t \Lambda[q_i(\lambda) - q_d(\lambda)]d\lambda + \int_{t_0}^t \sum_{j \neq i} K_{ij}(q_i - q_j) \quad (\text{II.14})$$

Or $\dot{e} = 0$, ce qui produit :

$$\dot{q}_i - \dot{q}_d = -\Lambda_i(q_i - q_d) - \sum_{j \neq i} K_{ij}(q_i - q_j) \quad (\text{II.15})$$

Posons $\varepsilon_i = q_i - q_d$, donc (II.15) peut être représentée par :

$$\dot{\varepsilon}_i = -\Lambda_i \varepsilon_i - \sum_{j \neq i} K_{ij}(\varepsilon_i - \varepsilon_j) \quad (\text{II.16})$$

Notre objectif est maintenant de démontrer que $\lim \varepsilon_i$ converge vers zéro pour $t \mapsto \infty$.

Donc supposons que : $\varepsilon = [\varepsilon_1^T \cdots \varepsilon_i^T \cdots \varepsilon_n^T]^T$ et $\Lambda = \text{diag}(\Lambda_1 \cdots \Lambda_i \cdots \Lambda_n)$.

L'équation (II.16) peut s'écrire comme suit :

$$\dot{\varepsilon} = A \varepsilon \quad (\text{II.17})$$

où la matrice A est donnée par :

$$\begin{pmatrix} -\Lambda_1 - \sum_{j \neq 1} K_{1j} & \cdots & K_{1j} & \cdots & K_{1n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ K_{i1} & \cdots & -\Lambda_i - \sum_{j \neq i} K_{ij} & \cdots & K_{in} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ K_{n1} & \cdots & K_{nj} & \cdots & -\Lambda_n - \sum_{j \neq n} K_{nj} \end{pmatrix}$$

Posons la fonction de Lyapunov suivante

$$v(t) = \varepsilon^T \varepsilon \quad (\text{II.18})$$

La dérivée par rapport au temps de $v(t)$ donne :

$$\begin{aligned} \dot{v} &= 2 \sum_{i=1}^n \varepsilon_i^T \dot{\varepsilon}_i \\ &= 2 \sum_{i=1}^n \varepsilon_i^T (-\Lambda_i \varepsilon_i - \sum_{j \neq i} K_{ij}(\varepsilon_i - \varepsilon_j)) \\ &= -2 \sum_{i=1}^n \varepsilon_i^T \Lambda_i \varepsilon_i - 2 \sum_{i=1}^n \sum_{j \neq i} \varepsilon_i^T K_{ij}(\varepsilon_i - \varepsilon_j) \\ &= -2 \sum_{i=1}^n \varepsilon_i^T \Lambda_i \varepsilon_i - 2 \sum_{i=1}^n \sum_{j \neq i} \varepsilon_i^T K_{ij} \varepsilon_i + 2 \sum_{i=1}^n \sum_{j \neq i} \varepsilon_i^T K_{ij} \varepsilon_j \quad (\text{II.19}) \end{aligned}$$

Sachant que

$$\sum_{i=1}^n \sum_{j \neq i} \varepsilon_i^T K_{ij} \varepsilon_i = \sum_{j=1}^n \sum_{i \neq j} \varepsilon_j^T K_{ji} \varepsilon_j \quad (\text{II.20})$$

D'où,

$$\dot{v} = -2 \sum_{i=1}^n \varepsilon_i^T \Lambda_i \varepsilon_i - \sum_{i=1}^n \sum_{j \neq i}^n (\varepsilon_i - \varepsilon_j)^T K_{ij} (\varepsilon_i - \varepsilon_j) \leq 0. \quad (\text{II.21})$$

Par une application directe du principe d'invariance de LaSalle on peut prouver que l'origine est globalement asymptotiquement stable. Par conséquent, nous obtenons $\lim_{t \rightarrow \infty} \varepsilon_i(t) \rightarrow 0$ pour $t \rightarrow \infty$. D'où $q_i \rightarrow q_d$ et $\dot{q}_i \rightarrow \dot{q}_d$ pour $t \rightarrow \infty$.

En se référant à l'équation (II.15), nous prouvons que $q_i \rightarrow q_j$ pour $t \rightarrow \infty$.

II.4 Synchronisation adaptative

Dans cette section nous nous intéressons à étudier le cas d'incertitude au niveau des paramètres physiques des robots. La stratégie de contrôle est alors étendue afin d'avoir une approche de contrôle adaptative dont le contrôleur prend la forme suivante :

$$\begin{aligned} \tau_i = & \hat{C}_i(q_i, \dot{q}_i) + \hat{g}_i(q_i) + \hat{M}_i(q_i)[\ddot{q}_d - K_{pi}e_i - K_{di}\dot{e}_i \\ & - \Lambda_i(\dot{q}_i - \dot{q}_d)] + \hat{M}_i(q_i) \left[\sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i \varepsilon_{2i} \right] \end{aligned} \quad (\text{II.22})$$

Vu le cas d'incertitude, on a recours à la relation (I.16), similaire à la relation (I.10), dans laquelle le vecteur des paramètres estimés $\hat{\theta}$ remplace le vecteur des paramètres exacts θ . Ce qui produit la forme suivante :

$$\hat{M}_i(q_i)\ddot{q}_i + \hat{C}_i(q_i, \dot{q}_i)\dot{q}_i + \hat{g}_i(q_i) = Y_i(q_i, \dot{q}_i, \ddot{q}_i)\hat{\theta}_i \quad (\text{II.23})$$

Avec $\theta_i \in R^a$ est un vecteur contenant les paramètres structurels du robot et a est le nombre des paramètres inconnus. Le paramètre estimé $\hat{\theta}_i$ qui est en fait le sujet de l'adaptation est défini de la forme suivante :

$$\dot{\hat{\theta}}_i = -\Gamma_i^{-1} \left(\hat{M}_i^{-1} Y_i \right)^T \dot{e}_i \quad (\text{II.24})$$

Où Γ_i est une matrice diagonale définie positive. Vu que la valeur exacte du paramètre θ_i est difficile à connaître dans le cas pratique, on définit ce qu'on appelle la valeur estimée notée $\hat{\theta}_i(t)$. Dans le même contexte, les matrices $\hat{M}_i$, $\hat{C}_i$ et $\hat{g}_i$ sont, respectivement, les estimées de M_i , C_i et g_i . $Y_i(q_i, \dot{q}_i, \ddot{q}_i)$ représente la matrice de régression. Posons $\tilde{\theta}_i = \theta_i - \hat{\theta}_i$ comme un vecteur contenant les erreurs d'estimations. Donc l'équation (II.24) peut s'écrire sous cette forme :

$$\dot{\tilde{\theta}} = \Gamma_i^{-1} (\hat{M}_i^{-1} Y_i)^T \dot{e}_i \quad (\text{II.25})$$

Théorème 4.2 Si $K_{di} > 2 \sum_{j \neq i} K_{ij}$, le contrôleur adaptatif (II.22) -(II.25) assure la convergence asymptotique vers zéro des erreurs de position, de vitesse et de synchronisation.

Preuve : En substituant (II.22) dans le modèle dynamique (II.10), on obtient la dynamique en boucle fermée suivante :

$$\begin{aligned} M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) = \\ \hat{M}_i(q_i)[\ddot{q}_d - K_{pi}e_i - K_{di}\dot{e}_i - \Lambda_i(\dot{q}_i - \dot{q}_d) \\ + \sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i\varepsilon_{2i}] + \hat{C}_i(q_i, \dot{q}_i)\dot{q}_i + \hat{g}_i(q_i) \end{aligned} \quad (II.26)$$

Nous définissons les erreurs d'estimation suivantes :

$$\tilde{M}_i(q_i) = M_i(q_i) - \hat{M}_i(q_i) \quad (II.27)$$

$$\tilde{C}_i(q_i, \dot{q}_i) = C_i(q_i, \dot{q}_i) - \hat{C}_i(q_i, \dot{q}_i) \quad (II.28)$$

$$\tilde{g}_i(q_i) = g_i(q_i) - \hat{g}_i(q_i) \quad (II.29)$$

Tenant compte des équations (II.27), (II.28), (II.29), l'équation (II.26) peut s'écrire sous la forme :

$$\begin{aligned} \tilde{M}_i(q_i)\ddot{q}_i + \tilde{C}_i(q_i, \dot{q}_i)\dot{q}_i + \tilde{g}_i(q_i) + \hat{M}_i(q_i)(\ddot{q}_i - \ddot{q}_d) = \\ \hat{M}_i(q_i)[-K_{pi}e_i - K_{di}\dot{e}_i - \Lambda_i(\dot{q}_i - \dot{q}_d) \\ + \sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i\varepsilon_{2i}] \end{aligned} \quad (II.30)$$

Tenant compte de (II.23), la dynamique de l'équation (II.30) peut s'écrire comme suit :

$$\begin{aligned} Y_i(q_i, \dot{q}_i, \ddot{q}_i)\tilde{\theta}_i + \hat{M}_i(q_i)(\ddot{q}_i - \ddot{q}_d) = \\ \hat{M}_i(q_i)[-K_{pi}e_i - K_{di}\dot{e}_i - \Lambda_i(\dot{q}_i - \dot{q}_d) \\ + \sum_{j \neq i} K_{ij}(\varepsilon_{2i} - \varepsilon_{2j}) + \Lambda_i\varepsilon_{2i}] \end{aligned} \quad (II.31)$$

Et par conséquent

$$\begin{aligned} \hat{M}_i(q_i)\ddot{e}_i + Y_i(q_i, \dot{q}_i, \ddot{q}_i)\tilde{\theta}_i = \\ \hat{M}_i(q_i)[-K_{pi}e_i - K_{di}\dot{e}_i \\ + \sum_{j \neq i} K_{ij}(\dot{e}_i - \dot{e}_j)] \end{aligned} \quad (II.32)$$

Afin de prouver la stabilité pour le système global, on définit : $Y^T = [Y_1^T, \dots, Y_i^T, \dots, Y_n^T]^T$, $\theta^T = [\theta_1^T, \dots, \theta_i^T, \dots, \theta_n^T]^T$, $\Gamma = \text{diag}(\Gamma_1^T, \dots, \Gamma_i^T, \dots, \Gamma_n^T)$

$K_d = \text{diag}(K_{di})$, $K_p = \text{diag}(K_{pi})$, $\hat{M} = \text{diag}(\hat{M}_i)$.

Ecrivons l'équation (II.32) sous forme compacte :

$$\hat{M}[\ddot{e} + K_p e + K_d \dot{e} - K_c \dot{e}] = -Y(q_i, \dot{q}_i, \ddot{q}_i) \tilde{\theta} \quad (\text{II.33})$$

Multiplions par $\hat{M}^{-1}$ des deux cotés :

$$\ddot{e} = -K_p e - K_d \dot{e} + K_c \dot{e} - \hat{M}^{-1} Y(q_i, \dot{q}_i, \ddot{q}_i) \tilde{\theta} \quad (\text{II.34})$$

Soit la fonction de Lyapunov suivante :

$$V = \frac{1}{2} (\dot{e}^T \dot{e} + e^T K_p e + \tilde{\theta}^T \Gamma \tilde{\theta}) \quad (\text{II.35})$$

Dérivons la fonction $V(t)$ par rapport au temps :

$$\dot{V} = \dot{e}^T \dot{e} + e^T K_p \dot{e} + \tilde{\theta}^T \Gamma \dot{\tilde{\theta}} \quad (\text{II.36})$$

En substituant (II.34) dans (II.36)

$$\dot{V} = (-K_p e - K_d \dot{e} + K_c \dot{e} - \hat{M}^{-1} Y \tilde{\theta})^T \dot{e} + e^T K_p \dot{e} + \tilde{\theta}^T \Gamma \dot{\tilde{\theta}} \quad (\text{II.37})$$

Par conséquent,

$$\dot{V} = -\dot{e}^T (K_d - K_c) \dot{e} + \tilde{\theta}^T [\Gamma \dot{\tilde{\theta}} - (\hat{M}^{-1} Y)^T \dot{e}] \quad (\text{II.38})$$

L'utilisation de la loi d'adaptation (II.25) produit :

$$\dot{V} = -\dot{e}^T (K_d - K_c) \dot{e}. \quad (\text{II.39})$$

Puisque on a $K_{di} > \sum_{j \neq i} K_{ij}$, alors $\dot{V}(t) < 0$. Ce qui donne $V(t) < V(0)$. Par suite $e, \dot{e}$ et $\hat{M}^{-1} Y \tilde{\theta}^T$ sont bornées.

Dérivons $V(t)$ par rapport au temps :

$$\ddot{V} = 2(K_p + (K_d - K_c) \dot{e} + \hat{M}^{-1} Y \tilde{\theta})^T (K_d - K_c) \dot{e} \quad (\text{II.40})$$

En utilisant le lemme de Barbalat :

$\ddot{V}$ est borné car $e, \dot{e}$ et $\hat{M}^{-1} Y \tilde{\theta}$ sont bornés. Ce qui implique que $\dot{V}(t) \rightarrow 0$ pour $t \rightarrow \infty$ et donc $\dot{e} \rightarrow 0$ pour $t \rightarrow \infty$. À cette étape, la preuve doit suivre le même raisonnement que celui du paragraphe II.3. Par conséquent, on prouve que les erreurs de position, les erreurs de vitesse et les erreurs de synchronisation convergent asymptotiquement vers 0.

II.5 Synchronisation en présence de délais de communication

Dans ce paragraphe nous étudions les problèmes de coordination dans les systèmes multi-robots en présence de délais de communication. Dans un réseau de n agents modélisé par un graphe orienté, par définition de la théorie des graphes, chaque agent est modélisé i qui représente le $i^{\text{ème}}$ robot du réseau et chaque liaison de communication est modélisée par une arête. Chaque arête noté e_{ij} , représente le canal de communication entre l'agent i et l'agent j . De ceci, le réseau est modélisé par le graphe suivant $\mathbf{G}(V, E)$ dont V est l'ensemble des sommets tandis que E est l'ensemble des arêtes. La dynamique de chaque agent du réseau peut s'écrire comme suit :

$$\dot{x}_i(t) = u_i(t) \quad (\text{II.41})$$

Où x_i représente l'état de l' $i^{\text{ème}}$ agent (position ou vitesse ou les deux ensembles) et u_i est le contrôleur tenant compte des retards. La littérature propose bien plus qu'une stratégie pour traiter les algorithmes distribués des systèmes multi-agents. Une approche proposée [62], [55] traite le problème de la forme suivante :

$$u_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t) - x_i(t)] \quad (\text{II.42})$$

Où N_i représente l'ensemble de robots tenant une architecture de communication avec le $i^{\text{ème}}$ robot et a_{ij} représente le poids de communication entre le robot i et le robot j . C'est une forme simple à implémenter, une forme simple à implémenter, mais qui, du point de vue pratique, n'est pas efficace et reste loin de la réalité. Tenant compte du retard au sein des canaux de communication dans les réseaux des systèmes multi-agents, une autre architecture développée [56] se présente sous la forme suivante :

$$u_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t - \tau_{ij}) - x_i(t - \tau_{ij})] \quad (\text{II.43})$$

où τ_{ij} modélise le retard de l'information au niveau du canal de communication entre les deux robots i et j . Ce modèle est le plus réaliste. Cependant, il exige trop de calcul et par suite plus d'allocation de mémoire et de puissance au niveau de l'unité de contrôle. Une topologie [40] est proposée comme une solution de compromis entre (II.2) et (II.3) :

$$u_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t - \tau_{ij}) - x_i(t)] \quad (\text{II.44})$$

En effet, c'est une loi réduite de (II.3), qui tient en compte du retard et par suite cette loi est réaliste et efficace du côté pratique que (II.2) si elle évite la complexité au niveau

de l'implémentation et le traitement de l'information par laquelle passe la loi de (II.3). Une approche récente résout le problème autrement. Cette dernière stratégie propose l'architecture suivante :

$$u_i(t) = \sum_{j \in N_i} a_{ij} [x_j(t - \tau_{ij}) - x_i(t - \tau)] \quad (\text{II.45})$$

En effet c'est une architecture qui se base sur celle de II.3 en gardant la même philosophie d'émission qui considère un retard variable et qui dépend essentiellement du robot émetteur. Par contre, elle pose un retard constant au niveau de la réception. Cette approche propose de plus un gain k afin d'ajuster la plage du retard dans le but d'établir une synchronisation parfaite. La synchronisation est prouvée pour tout retard $\tau < \frac{\tau_0}{k}$ tels que :

$$\tau_0 = \left[2k \left(\max_{j \in N_i} \sum a_{ij} \right) \right]^{-1} \quad (\text{II.46})$$

Dans notre étude, le retard est justifié par le fait que les informations diffusées par les robots voisins à un robot i seront reçues par ce dernier avec un certain retard. Ce délai peut être dû à certaines contraintes de communication au sein de réseau. De ce fait, nous définissons dans notre étude théorique un certain retard noté τ . Ce terme reflétant le retard est introduit dans l'expression de l'erreur de synchronisation. Par conséquent, l'erreur de synchronisation prend la forme suivante :

$$\varepsilon_{2i}(t) = \sum_{j \neq i} K_{ij} [q_i(t - \tau) - q_j(t - \tau)] \quad (\text{II.47})$$

Ce qui en résulte,

Théorème 4.3 Le contrôleur implémenté dans chaque robot de réseau coopératif prenant l'expression suivante pour tout $(i = 1, \dots, p)$:

$$\begin{aligned} \tau_i = & C_i(q_i, \dot{q}_i) \dot{q}_i + g_i(q_i) + M_i(q_i) [\ddot{q}_d - K_{pi} e_i(t) - K_{di} \dot{e}_i(t) - \Lambda_i(\dot{q}_i - \dot{q}_d)] \\ & + M_i(q_i) \left[\sum_{j \neq i} K_{ij} [\varepsilon_{2i}(t - \tau) - \varepsilon_{2j}(t - \tau)] + \Lambda_i \varepsilon_{2i}(t - \tau) \right] \end{aligned} \quad (\text{II.48})$$

assure la stabilité du système multi-robots en présence de retard de communication.

Preuve : En substituant (II.48) dans (I.10), on obtient :

$$\begin{aligned} M_i(q_i) \ddot{q}_i = & M_i(q_i) [\ddot{q}_d - K_{pi} e_i - K_{di} \dot{e}_i - \Lambda_i(\dot{q}_i - \dot{q}_d)] + \\ & M_i(q_i) \left[\sum_{j \neq i} K_{ij} [\varepsilon_{2i}(t - \tau) - \varepsilon_{2j}(t - \tau)] + \Lambda_i \varepsilon_{2i}(t - \tau) \right] \end{aligned} \quad (\text{II.49})$$

Tout calcul fait, nous obtenons pour expression de la dynamique de l'erreur :

$$\ddot{e} = -K_p e - K_d \dot{e} + K_c \dot{e}(t - \tau) \quad (\text{II.50})$$

où K_p , K_d et K_c sont les mêmes matrices déjà définies dans le paragraphe II.2.

En se basant sur la formule de Leibnitz, nous pouvons écrire :

$$\dot{e} - \dot{e}(t - \tau) = \int_{t-\tau}^t \ddot{e}(\lambda) d\lambda \quad (\text{II.51})$$

La substitution de (II.51) dans (II.50) produit :

$$\ddot{e} = -K_p e - K_d \dot{e} + K_c \left(\dot{e} - \int_{t-\tau}^t \ddot{e}(\lambda) d\lambda \right) \quad (\text{II.52})$$

Posons $\tilde{e} = [e^T, \dot{e}^T]^T$. Donc l'équation (II.52) peut s'écrire comme suit :

$$\dot{\tilde{e}} = \begin{pmatrix} 0 & I \\ -K_p & -K_d + K_c \end{pmatrix} \tilde{e} - \begin{pmatrix} 0 & 0 \\ 0 & K_c \end{pmatrix} \int_{t-\tau}^t \dot{\tilde{e}}(\lambda) d\lambda \quad (\text{II.53})$$

Ce qui produit la forme suivante :

$$\dot{\tilde{e}} = \beta_0 \tilde{e} - \beta_1 \int_{t-\tau}^t \dot{\tilde{e}}(\lambda) d\lambda \quad (\text{II.54})$$

Avec $\beta_0 = \begin{pmatrix} 0 & I \\ -K_p & -K_d + K_c \end{pmatrix}$ et $\beta_1 = \begin{pmatrix} 0 & 0 \\ 0 & K_c \end{pmatrix}$.

Afin d'analyser la stabilité du système global, nous considérons la fonction de Lyapunov-Krasovskii (FLK) suivante :

$$\begin{aligned} v(t) &= v_1(t) + v_2(t) + v_3(t) \\ v_1(t) &= \tilde{e}^T(t) P \tilde{e}(t) \\ v_2(t) &= \int_{t-\tau}^t \tilde{e}^T(\lambda) R \tilde{e}(\lambda) d\lambda \\ v_3(t) &= \int_{-\tau}^0 \left(\int_{t+s}^t \dot{\tilde{e}}^T(\alpha) Z \dot{\tilde{e}}(\alpha) d\alpha \right) d\lambda \end{aligned} \quad (\text{II.55})$$

Où $P = P^T > 0$; $R = R^T > 0$; $Z = Z^T > 0$ sont des matrices de pondération de dimensions appropriées. La dérivée par rapport au temps de la solution (II.55) est donnée par l'expression suivante :

$$\begin{aligned} \dot{v}(t) &= \delta^T [2N^T P M + N^T R N - Q^T R Q + \tau M^T Z M] \delta \\ &\quad - \int_{t-\tau}^t \dot{\tilde{e}}^T(\lambda) Z \dot{\tilde{e}}(\lambda) d\lambda \end{aligned} \quad (\text{II.56})$$

Où $\delta = [\tilde{e}^T(t), \tilde{e}^T(t - \tau)]^T$; $N = [\beta_0, \beta_1]$; $M = [I, 0]$; $Q = [0, I]$. L'inégalité de Jensen donne une limite appropriée pour l'équation (II.56).

$$\begin{aligned} - \int_{t-\tau}^t \dot{\tilde{e}}^T(\lambda) Z \dot{\tilde{e}}(\lambda) d\lambda &\leq \int_{t-\tau}^t \dot{\tilde{e}}^T(\lambda) d\lambda (Z/\tau) \int_{t-\tau}^t \dot{\tilde{e}}(\lambda) d\lambda \\ &\leq -\tilde{e}^T T^T (Z/\tau) T \tilde{e} \end{aligned} \quad (\text{II.57})$$

Avec $T = [I, -I]$. Donc, la dérivée de la LKF (II.55) peut être bornée telle que $\dot{v}(t) \leq \delta^T \xi \delta$, où :

$$\xi = 2N^T P M + N^T R N - Q^T R Q + \tau M^T Z M - 1/\tau T^T Z T.$$

Donc si l'Inégalité Matricielle Linéaire (IML) $\xi < 0$ est satisfaite, alors la dérivée de la FLK est définie négative. Pour garantir que la matrice ξ soit définie négative, les gains de contrôle devraient être choisis $K_p > K_p^*$ et $K_d - K_c > K^*$, tels que :

$$2N^T P M + N^T R N - Q^T R Q + \tau M^T Z M - \frac{1}{\tau} T^T Z T < 0 \quad (\text{II.58})$$

D'où, FLK est définie négative et par suite l'origine $\tilde{e} = 0$ est asymptotiquement stable. Par conséquent, $e \rightarrow 0$ pour $t \rightarrow \infty$ et $\dot{e} \rightarrow 0$ pour $t \rightarrow \infty$. Prouver la convergence asymptotique de l'erreur globale $\tilde{e}$ reste insuffisant pour démontrer la convergence vers zéro des deux erreurs e_1 et e_2 . Donc, la démonstration continue à suivre le même raisonnement que celui du paragraphe II.3. Comme déduction de (II.15), on obtient l'équation suivante :

$$\dot{\varepsilon}_i = -\Lambda_i \varepsilon_i - \sum_{j \neq i} K_{ij} [\varepsilon_i(t - \tau) - \varepsilon_j(t - \tau)] \quad (\text{II.59})$$

Réécrivons tous les états de l'équation précédente dans une forme compacte et appliquons la Transformée de Laplace :

$$s\varepsilon(s) - \varepsilon(0) = -\Lambda\varepsilon(s) - e^{-\tau s} K_c \varepsilon(s) \quad (\text{II.60})$$

Autrement :

$$\varepsilon(s) = (sI + \Lambda + e^{-\tau s} K_c)^{-1} \varepsilon(0) \quad (\text{II.61})$$

Si l'équation caractéristique a tous ses zéros dans la moitié gauche du plan complexe, alors le système est stable et par suite nous pouvons conclure facilement la convergence de q_i vers q_d . Puisque le système sans retard ($cd, \tau = 0$) est une fonction continue, alors en utilisant la décomposition, la solution minimale positive est représentée comme suit :

$$\tau_i = \frac{1}{\sqrt{K_{ci}^2 - \Lambda_i^2}} \arccos \left(\frac{-\Lambda_i}{K_{ci}} \right)$$

Ce qui donne tous les zéros de l'équation caractéristique à gauche du plan complexe. Donc si nous choisissons $\tau \in [0, \tau_c]$, où $\tau_c = \sup\{\tau_i, 1 \leq i \leq p\}$, alors les solutions de (II.61) convergent vers zéro et par suite $q_i \rightarrow q_d$, $\dot{q}_i \rightarrow \dot{q}_d$, et $q_i \rightarrow q_j$ pour $t \rightarrow \infty$.

II.6 Simulation

Afin de valider la stratégie développée, des simulations ont été effectuées en prenant comme système un groupe de trois robots. Les différentes simulations ont été réalisées avec Matlab/Simulink. Les robots n'ont pas les mêmes conditions initiales de départ de l'opération. En premier lieu, nous supposons que les paramètres des robots sont parfaitement identifiés et que le retard de transmission peut être négligé. La figure (II.2) présente la synchronisation des trois robots suivant une même trajectoire désirée. Ceci montre que les deux objectifs de notre approche sont atteints : le suivi de trajectoire d'une part et la synchronisation d'autre part. Dans un second temps, nous avons supposé que les paramètres physiques de chaque robot en réseau sont mal connus. En conséquence, les matrices d'inertie, de Coriolis et de gravité sont inconnues. La figure (II.3) illustre le comportement du système global avec prise en compte de l'incertitude des paramètres physiques des robots. Ainsi, la synchronisation et le suivi sont aussi validés dans le cas adaptatif. Les figures (II.4) et (II.5), en présentant, respectivement, les erreurs de position et les erreurs de synchronisation, montrent l'efficacité de l'approche proposée. Finalement, nous considérons le problème du retard de communication dans les canaux de transmissions. Les figures (II.6) et (II.7) affichent le comportement du groupe de robots sous l'effet de retard de communication. Ces deux figures montrent la synchronisation des trois robots en suivant une trajectoire commune. Est ainsi illustrée, la résolution du problème de coordination en présence de délais de communication et par suite, la performance de l'approche en termes de tolérance aux contraintes pratiques. Nous remarquons que les robots coopératifs, en absence de toute incertitude, synchronisent leurs mouvement en suivant une trajectoire commune, et ceci très rapidement. Tandis qu'en cas d'incertitudes paramétriques, la synchronisation du mouvement présente un peu de retard ainsi que le suivi de trajectoire marque une petite déviation au départ.

Concernant les simulations des algorithmes en présence et en absence du retard, on remarque pour une trajectoire quelconque (en escalier), avec la négligence de tout retard d'information, une convergence rapide au niveau de la synchronisation du mouvement ainsi qu'au niveau de suivi de trajectoire. Par ailleurs, en présence d'un retard d'information au sein des canaux de communication, on remarque des perturbations de signaux en forme de "pics". Ce problème est résolu après quelques secondes d'adaptation et les robots en coopération retrouvent leur synchronisation ainsi que leur trajectoire commune à suivre.

Dans la situation actuelle, l'utilisation courante de ces types de réseau de communication reste toujours un défi difficile à connaître. De ce fait on a pensé à développer un simulateur qui nous permet de simuler en 3D des groupes de robots dans un environnement

virtuel mais mieux expressifs que les simulations classiques. Le but de la conception de cet environnement interactif est de plonger l'utilisateur dans une immersion totale. En effet, un environnement 3D virtuel est une modélisation d'un monde réel. La simulation reste une solution efficace pour valider des systèmes complexes avant toute implémentation pratique. Afin de concevoir cette application logicielle, on a eu recours à plusieurs outils. Le logiciel "Cinema 4D" a été utilisé pour la conception 3D des robots en dimensions réelles. Pour la réalisation de l'environnement virtuel nous nous sommes basés sur la "toolbox" VRML de matlab. L'implémentation du programme passe par Matlab-Simulink. L'objet 3D crée par Matlab Cinema 4D est extrait en fichier de format VRML97 (.wrl). Ce fichier est édité par le programme V-Real Builder 2.0 juste avant d'être importé dans Matlab-Simulink. L'objectif à moyen terme de cette conception est de préparer une plateforme pour la réalisation d'une application en réalité augmentée. L'environnement virtuel de simulation est présenté à la figure (II.8).

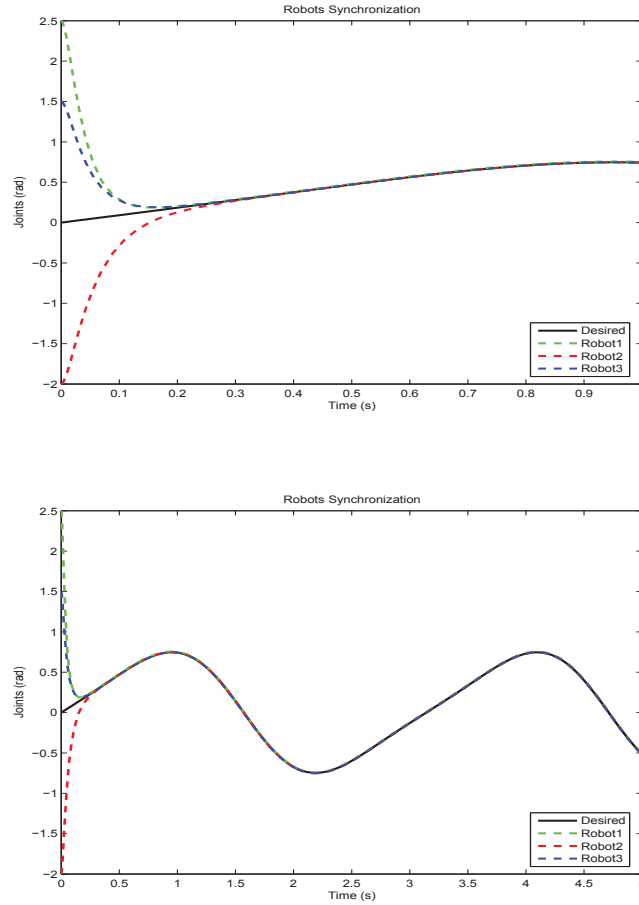


FIGURE II.2 – Synchronisation des robots en absence de toute incertitude

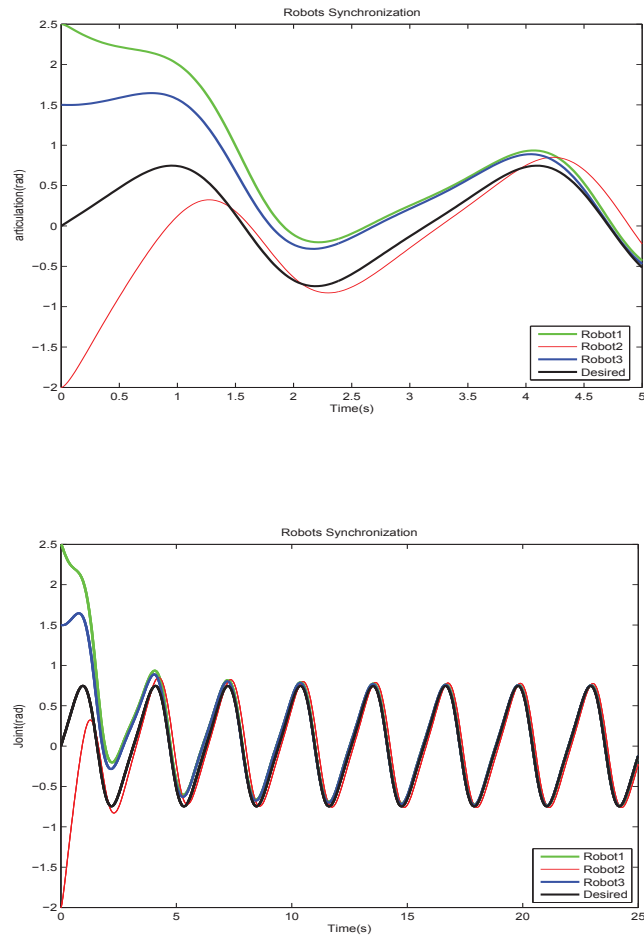


FIGURE II.3 – Synchronisation des robots en présence des incertitudes

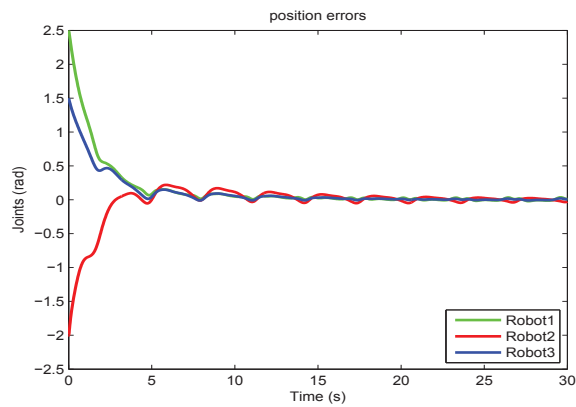


FIGURE II.4 – Les erreurs de position en présence des incertitudes

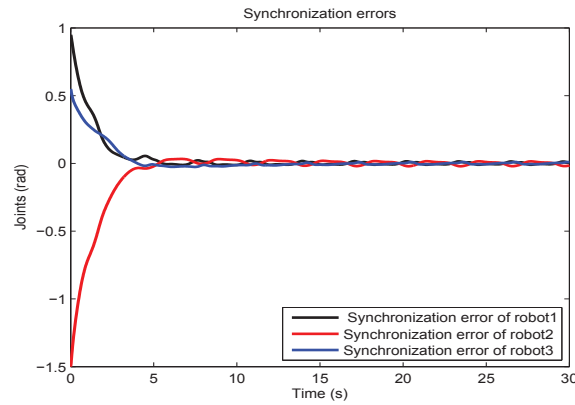


FIGURE II.5 – Les erreurs de synchronisation en présence des incertitudes

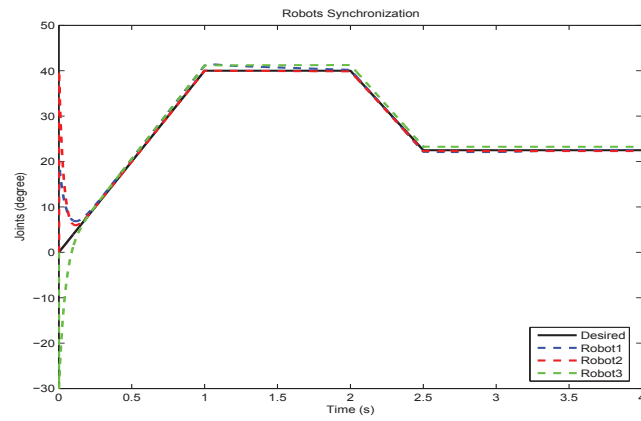


FIGURE II.6 – Synchronisation des robots en absence de retard

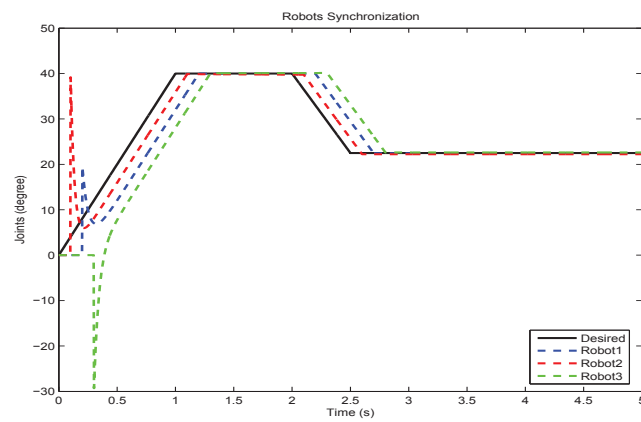


FIGURE II.7 – Synchronisation des robots en présence de retard

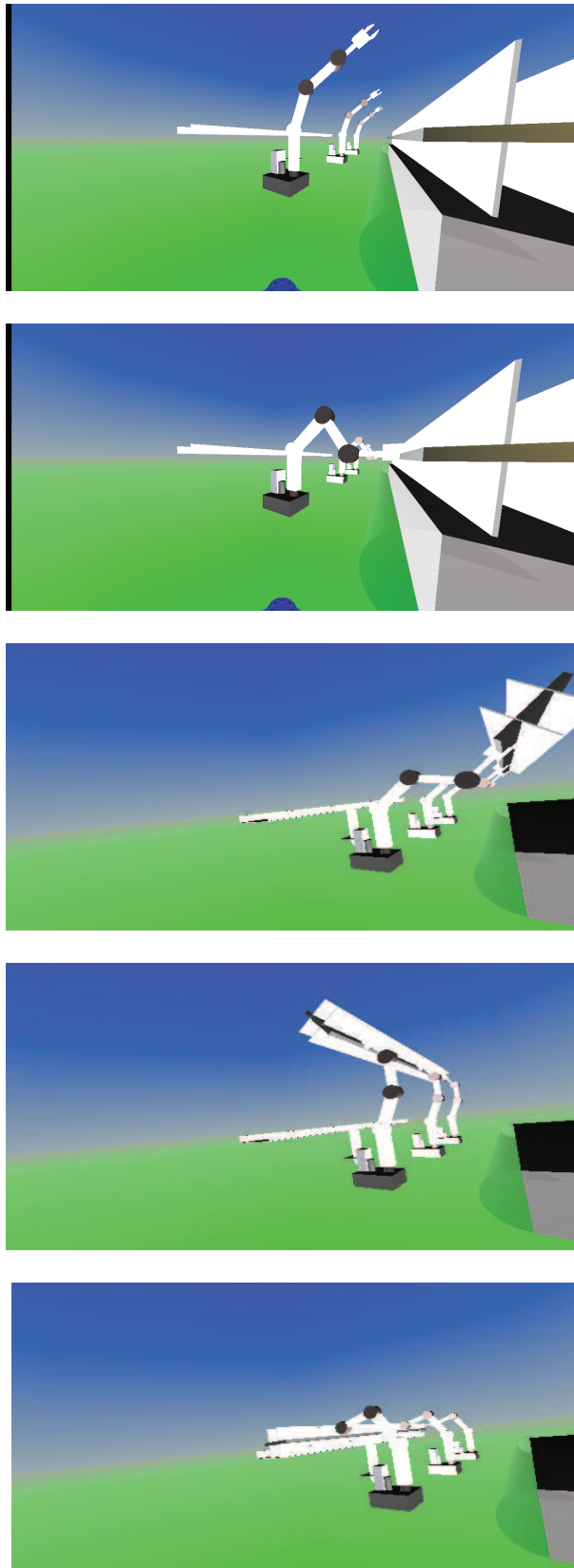


FIGURE II.8 – Simulateur 3D de robots

II.7 Conclusion

Ce chapitre a présenté une stratégie de synchronisation d'un groupe de robots manipulateurs travaillant en coopération. Nous avons proposé une loi de contrôle complètement décentralisée, capable de synchroniser ce groupe de robots sans leader en suivant une trajectoire commune désirée. Chaque robot converge alors vers le consensus final du groupe.

L'objectif était la réduire les erreurs de suivi de trajectoire ainsi que les erreurs de synchronisation pour tout robot du groupe.

Afin d'avoir une approche plus pratique, la loi développée a été étendue à une version adaptative. La proposition prend en considération les incertitudes attachées aux paramètres physiques du groupe ainsi que les retards de communication.

Un challenge dans la conception de systèmes de contrôle au sein des réseaux multi-agents. D'où un autre aspect inspiré des problèmes pratiques de communication est étudié dans cette approche.

Les simulations effectuées sur un groupe de robots manipulateurs nous ont montré l'efficacité de la loi développée dans les différentes cas étudiés : en absence et en présence d'incertitude ainsi qu'en absence et en présence de retard de communication.

Chapitre III

Synchronisation distribuée de systèmes Lagrange avec leader

Dans ce chapitre, une deuxième approche de contrôle de la synchronisation et du suivi de trajectoire est proposée pour des réseaux avec leader. Nous représentons une architecture hybride qui combine la décentralisation et la centralisation du contrôle. En fait la centralisation est claire dans notre architecture vu la présence d'un leader au niveau du groupe. Par ailleurs, la décentralisation est représentée par le fait que l'architecture permet à un robot en coopération de synchroniser son mouvement en suivant la trajectoire commune sans avoir de canaux de communication directs avec le leader. Cette topologie exploite alors les avantages de la centralisation comme elle profite des bienfaits de la décentralisation. En effet, la trajectoire désirée à suivre est implémentée seulement dans le leader et les autres robots du groupe n'ont qu'à suivre ce leader. La topologie développée permet d'avoir des connexions directes suiveur-suiveur ainsi que des retours d'informations des suiveurs au leader. Cette approche distribuée est basée sur la combinaison de la technique du backstepping et la théorie des graphes. Afin d'étendre la stratégie de contrôle développée à synchroniser des robots dont les paramètres physiques sont mal connus, on a recours à l'approche neuronale adaptative. En se basant sur la méthode de Krasovskii. Une étude de la stabilité de système global est lancée afin de résoudre les problèmes de retard dans les canaux de communication au sein du réseau.

III.1 Introduction

Récemment, les problèmes de coordination et de consensus ont été largement étudiés dans le domaine des systèmes distribués. En particulier, la coordination et la synchroni-

sation des systèmes multi-robots ont été largement étudiés dans la littérature avec des applications en robotique mobile [15], [19], en robotique de manipulation [51], [63], sur les véhicules aériens sans pilote [53], les véhicules autonomes sous-marins [92], les avions [95], et les systèmes de Lagrange en général [2], [12]. Les systèmes multi-agents basés sur des systèmes de consensus du premier ordre ont attiré beaucoup d'attention [27], [43]. Toutefois, en raison de la vaste catégorie des agents, ceux-ci doivent être décrits par une dynamique du second ordre. L'algorithme de consensus modélisé par un double intégrateur représente le défi le plus important [10], [26]. Les problèmes de consensus dans les systèmes multi-agents sont en relation directe avec les problèmes de synchronisation qui utilisent la théorie des graphes [56]. Plusieurs travaux de recherche ont concerné les structures de réseau de type leader-followers [27], [39], [45], [47], [48], [52], [62]. Une loi de contrôle de coordination basée sur la méthode des événements est conçue pour un groupe de robots [80]. Toutefois, la stratégie de contrôle utilisée dans [80] est classée comme une approche centralisée. La plupart des travaux antérieurs de coordination des systèmes multi-robots, utilisant l'architecture leader-suiveur, exigent la présence d'un canal de communication reliant directement chaque suiveur au leader du groupe. Autrement, chaque suiveur dans le réseau a un accès direct à l'information de leader [69], [116], [117]. Ces types de topologies sont plus compliquées, plus coûteuses et moins robustes. Des études remarquables prouvent que la conception d'une architecture basée sur la règle des voisins peut assurer la coordination du mouvement d'un groupe de robots malgré l'absence d'un robot leader [43], [62]. La règle des voisins peut être décrite ici comme étant un réseau qui se base sur les échanges d'informations entre les voisins les plus proches seulement. C'est une méthode qui permet la propagation facile des données sur tout le réseau afin d'atteindre un consensus. Des publications récentes s'intéressent à ces types de stratégies [4], [11], [23]. La combinaison de la topologie leader-suiveur et l'approche décentralisée est le modèle le plus efficace. Il bénéficie en même temps de la simplicité, de l'efficacité et de la robustesse. La contribution de la stratégie présentée dans ce chapitre peut essentiellement se résumer dans les points suivants :

1. L'approche développée résout les systèmes complexes de forte non linéarité.
2. Contrairement à plusieurs travaux de coopération des systèmes multi-agents qui utilisent la théorie des graphes [47], [56], [62] cette approche ne se limite pas à faire converger les agents vers la valeur moyenne des conditions initiales en tenant compte des pondérations. En effet, la loi de contrôle proposée, en plus de la synchronisation du mouvement du groupe, pousse les agents à suivre une trajectoire commune désirée.
3. Comparée aux [10], [51] et [69] l'approche développée évite la topologie des réseaux maillés "all to all coupling".
4. Au contraire des structures classiques de leader-suiveur [42], [69] la stratégie proposée

bénéficie de la technique de la règle des plus proches voisins afin de concevoir une architecture de contrôle décentralisée sous une topologie de leader-suiveur et par suite avoir une stratégie moins couteuse et plus robuste.

5. L'approche est étendue pour résoudre les problèmes d'incertitude en se basant sur l'approche neuronale qui évite l'adaptation classique proposée par [12], [63].
6. Distinguée par rapport aux approches proposées dans [1], [2], [4], [11], [23], [63] cette approche traite en plus les problèmes de retard dans les canaux de communication.

III.2 La théorie des graphes

Du point de vue mathématique et science de l'informatique, la théorie des graphes est l'étude des graphes qui sont en fait des outils mathématiques utilisées essentiellement pour modéliser les relations entre les objets dans certaines collections. Dans les dernières années, la théorie des graphes a été étudiée comme une approche pour résoudre différents problèmes tels que ceux posés par le trafic routiers, le transport des grandes charges, l'attribution des tâches, le contrôle de trafic aérien et les applications des robots coopératifs en général. Nous introduisons dans ce qui suit quelques définitions de base dans le contexte de la théorie des graphes.

III.2.1 Notions de base

Définition 3.9 (Graphe non orienté) *Un graphe interconnecté non orienté avec N éléments est défini par un couple d'ensemble $G = (V, E)$ tel que V est l'ensemble des sommets ou nœuds tandis que E représente l'ensemble des connexions entre ces sommets.*

- $V = i; i = 1 \dots N$
- $E = (e_i, e_j) \subseteq V \times V; e_i \neq e_j$

Le graphe non orienté se caractérise par le fait que les éléments de l'ensemble E ne sont pas ordonnés par paires de sommets.

Définition 3.10 (Sous-Graphe) *Soit $G = (V, E)$ un graphe, un sous graphe est défini comme suit :*

Soit $\bar{G} = (\bar{V}, \bar{E})$ tels que $\bar{V} \subseteq V$ et $\bar{E} \subseteq E$.

Définition 3.11 (Chemin dans un graphe) *Un chemin $P(e_0, e_L)$ dans un graphe connectant deux sommets n_0, n_L est défini par une suite finie des nœuds n_k telle que $n_{k-1} \neq n_k; k = 1 \dots L$.*

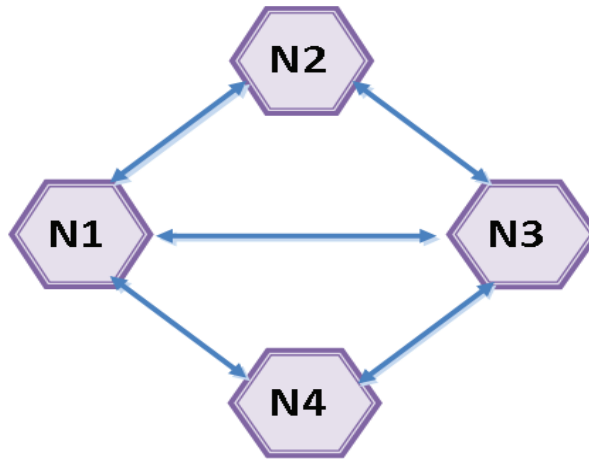


FIGURE III.1 – Graphe non orienté

Définition 3.12 (Graphe connecté) Un graphe $G = (V, E)$ est dit connecté si un chemin $P(n_0, n_L)$ existe pour tout couple de sommets (n_i, n_j) ; $n_i \neq n_j$.

Définition 3.13 (L'ensemble des voisins) Soit un graphe $G = (V, E)$ et un sommet $n_i \in G$, l'ensemble des voisins $N(n_i)$ est défini comme suit : $N(n_i) = \{n_j \in V ; (n_i, n_j) \in E\}$.

Définition 3.14 (Degré d'un nœud) Etant donné un graphe connecté $G = (V, E)$ et un sommet $n_i \in V$, le degré de sommet n_i est donné par le nombre de ses voisins, c'est-à-dire $\deg(n_i) = |N(n_i)|$.

Définition 3.15 (Graphe orienté) Soit un graphe $G = (V, E)$, G est dit orienté si E est défini comme suit : $E = \{(n_i, n_j), (n_j, n_i) \subseteq V \times V | n_i \neq n_j, (n_i, n_j) = -(n_j, n_i)\}$

Définition 3.16 (Graphe pondéré) un graphe pondéré est un graphe auquel on adjoint une fonction de valuation. Un graphe peut être pondéré/valué sur ses sommets comme sur ses arêtes. On note $p(n_i)$ le poids d'un sommet (nœud) i et $p(n_i, n_j)$ le poids d'une arête i, j .

III.2.2 Concept en théorie des graphes

Un moyen intéressant d'étudier les propriétés d'un graphe est de trouver une méthode pour écrire sa structure dans un mode de représentation mathématique.

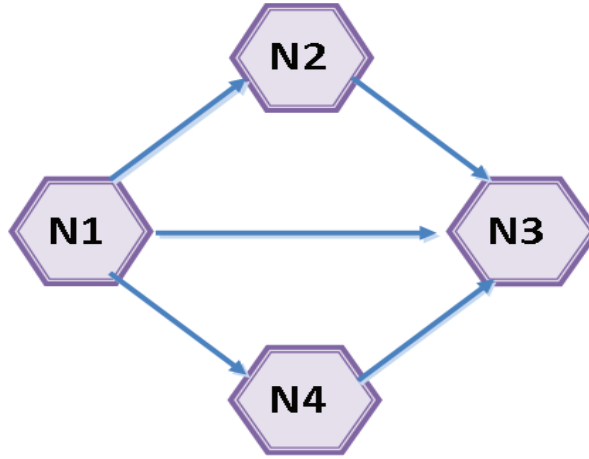


FIGURE III.2 – Graphe orienté

Dans ce paragraphe, les matrices les plus importantes qui définissent un graphe et leurs propriétés spectrales sont exposés.

Définition 3.17 (Matrice d'adjacence) La matrice d'adjacence $A \in \mathbf{R}^{n \times n}$ d'un graphe $G = (V, E)$ est une matrice binaire définie comme suit :

$$\begin{cases} a_{ij} = 1 & \text{si } (n_i, n_j) \in E \\ a_{ij} = 0 & \text{autrement} \end{cases}$$

On remarque que si G est non orienté, sa matrice d'adjacence A est symétrique et ses valeurs propres sont par conséquent réelles.

Définition 3.18 (Degré (matrice)) la matrice des degrés d'un graphe G est une matrice carrée de taille $|V| \times |V|$ telle que :

$$d_{ij} = \begin{cases} \deg_{out}(n_i) & \text{si } (i, j) \in E \\ 0 & \text{autrement} \end{cases}.$$

avec $\deg_{out}(n_i) = \sum_{j=1}^n a_{ij}$.

Définition 3.19 (Matrice de Laplace) La matrice de Laplace $L \in \mathbf{R}_{n \times n}$ d'un graphe G de N sommets est définie comme étant $L = D - A$, plus explicitement, comme suit :

$$L_{ij} = \begin{cases} \deg_{out}(n_i) & \text{si } (i = j) \in E \\ -1 & \text{si } i \neq j; (n_i, n_j) \in E \\ 0 & \text{autrement} \end{cases}.$$

III.3 Coordination non adaptative par backstepping et théorie des graphes

Soit un système multi-agent sous forme d'un réseau de robots industriels. Grâce à des communications explicites et implicites, chaque agent du groupe a une idée complète sur ses états ainsi que les états de ses coéquipiers. Pour simplifier la description de réseau et par suite son étude, nous considérons que chaque robot est représenté par un nœud (ou sommet) et la communication entre ces agents est modélisée par un graphe non orienté $\mathbf{G}$ (voir III.3 et III.4). Les arêtes représentent des canaux de communications bidirectionnels. Le groupe de robot interconnecté compte $m + 1$ robots. Le leader de ce groupe est un agent dominant dont le mouvement ne dépend pas des autres. Ce leader est indexé par l'index 0, tandis que, les suiveurs sont indexés par 1, 2, ..., m . Dans le cas d'absence d'un canal de communication entre le robot leader et un suiveur i du groupe, un zéro est affecté à la valeur de b_i .

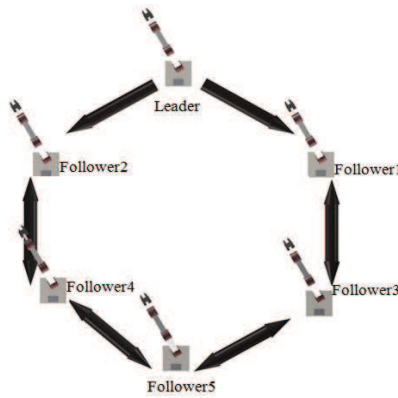


FIGURE III.3 – Architecture de réseau leader-suiveurs.

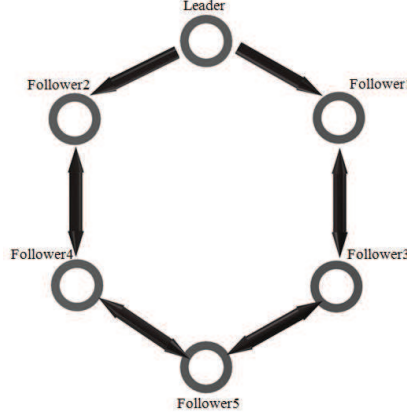


FIGURE III.4 – Modélisation du réseau sous forme de graphe.

Afin de concevoir la loi de contrôle de coordination du groupe de robots interconnectés, nous commençons par une représentation d'état de la dynamique d'un robot i (I.10) au sein de réseau.
$$\begin{cases} \dot{q}_{1i} = q_{2i} \\ M\dot{q}_{2i} = \tau - C(q_1, q_2)q_2 + g(q_1) \end{cases}$$

La philosophie de la méthode du backstepping est de supposer que q_{2i} est un contrôleur virtuel responsable de synchroniser l'état q_{1i} à l'état de leader q_0 . En se basant sur la théorie des systèmes multi-agents, on définit la forme suivante

$$u_i = \sum_{j \neq i} A_{ij}(q_{1j} - q_{1i}) + B_i(q_0 - q_{1i}) \quad (\text{III.1})$$

Où $A_{ij} = \text{diag}(a_{ij})_{m \times m}$; a_{ij} définit la i^{me} ligne et la j^{me} colonne de la matrice d'adjacence A du graphe G ; $B_i = \text{diag}(b_i)_{m \times m}$.

Supposons que q_{2i} est le contrôleur qui va synchroniser l'état q_{1i} avec le leader et k est un scalaire positif. Nous choisissons alors :

$$q_{2i} = ku_i + \dot{q}_0 \quad (\text{III.2})$$

En considérant l'équation (III.1), l'équation (III.2) peut s'écrire sous une forme compacte comme suit :

$$\dot{q}_1 - \dot{q}_0 \mathbf{1} = -k[(L + B) \otimes I_n](q_1 - q_0 \mathbf{1}) \quad (\text{III.3})$$

Ou autrement,

$$q_2 - \dot{q}_0 \mathbf{1} = -k[(L + B) \otimes I_n](q_1 - q_0 \mathbf{1}) \quad (\text{III.4})$$

Avec

$\dot{q}_1 = (\dot{q}_{11}^T, \dots, \dot{q}_{1i}^T, \dots, \dot{q}_{1m}^T)^T \in \mathbb{R}^{m \times n}$; $q_2 = (q_{21}^T, \dots, q_{2i}^T, \dots, q_{2m}^T)^T \in \mathbb{R}^{m \times n}$; $\mathbf{1} = (1, 1, \dots, 1)^T \in \mathbb{R}^n$; $\otimes$ représente le produit de Kronecker et I_n dénote la matrice identité $n \times n$.

L'objectif est de concevoir un contrôleur de couple τ_i qui commande l'état q_{1i} afin de converger vers q_0 pour tout $1 \leq i \leq m$, càd, $\lim q_{1i} \rightarrow q_0$ pour $t \rightarrow \infty$. D'où on définit le signal d'erreur comme suit :

$$e_i = q_{1i} - q_0 \quad (\text{III.5})$$

Puisque q_{2i} n'est pas en fait le contrôleur réel de q_{1i} , nous le considérons alors comme étant un contrôleur virtuel et nous posons la variable suivante comme une mesure de l'erreur.

$$z_i = q_{2i} - \alpha_i \quad (\text{III.6})$$

Avec α_i est devenu le contrôleur virtuel de q_{1i} tel que :

$$\alpha_i = ku_i + \dot{q}_0 \quad (\text{III.7})$$

Concevons un contrôleur de couple pour le i^{me} robot d'un réseau formé de m robots tel que $m \geq 2$:

$$\begin{aligned} \tau_i = & C_i(q_{1i}, q_{2i})q_{2i} + g(q_{1i}) - \frac{1}{2}kz_i - pe_i \\ & + M_i \left(\sum_{j \neq i} \frac{\partial \alpha_i}{\partial q_{ij}}(q_{2i} - q_{2j}) + \frac{\partial \alpha_i}{\partial q_{1i}}q_{2i} + \frac{\partial \alpha_i}{\partial q_0}\dot{q}_0 \right). \end{aligned} \quad (\text{III.8})$$

Où k et p sont des scalaires positifs.

Remarque 1 : Par définition de la matrice de Laplace L , le terme $l_{ij} \neq 0$ si et seulement si il s'agit d'un échange d'informations entre les deux robots i et j . Par ailleurs, pour la matrice diagonale B , le terme $b_{ij} \neq 0$ si et seulement si en cas de communication entre le robot leader et le robot suiveur.

Remarque 2 : Il est à noter que la loi de commande proposée impose de disposer de l'information sur la vitesse $\dot{q}_0$ et l'accélération $\ddot{q}_0$ du leader de groupe. Par conséquent, cette information devrait être accessible à tous les manipulateurs suiveurs dans la phase de conception du contrôleur. C'est aussi une exigence commune dans la littérature existante [25] et [62].

Remarque 3 : À partir de l'expression de la loi du contrôleur proposée, nous pouvons souligner que chaque robot utilise uniquement les informations de ces voisins avec lesquels il est connecté. Cet aspect qui reflète bien la décentralisation du système limite le flux de communication et par suite conduit à avoir un réseau performant et optimisé avec un faible coût.

Nous pouvons énoncer le premier lemme de cette approche comme suit :

Lemme 1 :

Considérons un réseau de robots manipulateurs formé de $(m+1)$ robots sous une topologie de leader-suiveur (III.3). La loi de contrôle proposée en (III.8) implémentée dans chaque

robot i du groupe assure la synchronisation de son mouvement par rapport à celui du leader, c-à-d, $\lim q_{1i} \rightarrow q_0$, $\lim \dot{q}_{1i} \rightarrow \dot{q}_0$ et $\lim \dot{q}_{1i} \rightarrow \dot{q}_{1j \neq i}$ pour $t \rightarrow \infty$.

Preuve: Commençons par l'équation (III.3), d'après [25] $(L + B)$ est une M-matrice non singulière, d'où, pour $P > 0$ et $Q > 0$ on a :

$$Q = P(L + B) + (L + B)^T P \quad (\text{III.9})$$

Afin de prouver que pour chaque $1 \leq i \leq m$, $q_i \rightarrow q_0$, nous considérons le système global et nous posons la fonction de Lyapunov suivante :

$$V = e^T (P \otimes I_n) e \quad (\text{III.10})$$

Où $e = q_1 - q_0 \mathbf{1}$ représente le vecteur d'erreur dans sa forme compacte.

La dérivée de V par rapport au temps est donnée par :

$$\dot{V} = [e^T (p \otimes I_n) \dot{e} + \dot{e}^T (p \otimes I_n) e] \quad (\text{III.11})$$

Tenant compte de l'équation (III.3), l'équation (III.11) s'écrit :

$$\begin{aligned} \dot{V} &= -e^T (p \otimes I_n) [(L + B) \otimes I_n] e - e^T [(L + B) \otimes I_n]^T (P \otimes I_n) e \\ &= -e^T [(p \otimes I_n) [(L + B) \otimes I_n] + [(L + B) \otimes I_n]^T (P \otimes I_n)] e \end{aligned} \quad (\text{III.12})$$

Ce qui produit :

$$\dot{V} = -e^T (Q \otimes I_n) e \quad (\text{III.13})$$

Ceci prouve que V est semi-définie négative et par suite l'origine $e = 0$ est asymptotiquement stable et $e \rightarrow 0$ pour $t \rightarrow \infty$ i.e, $q_i \rightarrow q_0 \forall i \in [1, m]$.

Comme q_2 n'est pas le contrôleur réel de q_1 , on le considère comme étant un contrôleur virtuel et on pose la variable z déjà définie dans (III.6) comme étant la variable d'erreur. Dans sa forme compacte l'équation (III.6) peut être écrite comme suit :

$$z = q_2 - \alpha \quad (\text{III.14})$$

Avec $z = [z_1^T, \dots, z_m^T]^T$, $q_2 = [q_{21}^T, \dots, q_{2m}^T]^T$, $\alpha = [\alpha_1^T, \dots, \alpha_m^T]^T$.

Dans une forme matricielle α est représentée de cette forme :

$$\alpha = ku + \dot{q}_0 \mathbf{1} \quad (\text{III.15})$$

Afin de concevoir le contrôleur réel de notre approche, la procédure de backstepping passe par deux étapes essentielles :

Première étape : Considérons la première fonction de Lyapunov définie comme suit :

$$V_1 = e^T P e \quad (\text{III.16})$$

La dérivée de V_1 par rapport au temps est décrite par :

$$\dot{V}_1 = 2e^T Pz + [e^T P(\alpha - \dot{q}_0) + (\alpha - \dot{q}_0)^T P e] \quad (\text{III.17})$$

Un simple calcul donne :

$$\dot{V}_1 = 2e^T Pz - e^T kQ \otimes I_n e \quad (\text{III.18})$$

Le premier terme de $\dot{V}_1$ sera traité dans la prochaine étape de la procédure de backstepping.

Deuxième étape : La dérivée par rapport au temps de z donne :

$$\dot{z} = \dot{q}_2 - \dot{\alpha} \quad (\text{III.19})$$

Multiplions par M des deux côtés.

$$M\dot{z} = M\dot{q}_2 - M\dot{\alpha}$$

Tenons en compte de la représentation d'état de la dynamique de la i^{me} robot, l'équation précédente peut s'écrire comme suit :

$$\begin{aligned} M\dot{z} = & \tau - C(q_1, q_2)q_2 - g(q_1) \\ & - M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} \dot{q}_{1ij} + \frac{\partial \alpha}{\partial q_{1i}} \dot{q}_{1i} + \frac{\partial \alpha}{\partial q_0} \dot{q}_0 \right) \end{aligned} \quad (\text{III.20})$$

Tout calcul fait,

$$\begin{aligned} M\dot{z} = & \tau - C(q_1, q_2)q_2 - g(q_1) \\ & - M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} (q_{2i} - q_{2j}) + \frac{\partial \alpha}{\partial q_{1i}} q_2 + \frac{\partial \alpha}{\partial q_0} \dot{q}_0 \right) \end{aligned} \quad (\text{III.21})$$

Sachant le contrôleur de couple τ_i déjà défini (III.8), on pose la fonction de lyapunov suivante :

$$V_2 = V_1 + z^T Mz \quad (\text{III.22})$$

La dérivée de V_2 par rapport au temps donne :

$$\dot{V}_2 = -e^T Q e + 2e^T Pz + 2z^T M\dot{z}$$

$$\begin{aligned}
&= -e^T Q e - z^T k z + 2e^T P z - 2z^T p e \\
&= -e^T Q e - z^T k z
\end{aligned} \tag{III.23}$$

Ce qui prouve bien que V_2 est définie négative. Par conséquent, nous prouvons la convergence asymptotique vers zéro des deux erreurs e et z .

III.4 Coordination adaptative neuronale

La première étude proposée est basée sur la connaissance des paramètres physiques des robots en réseau. Avec cette hypothèse, les matrices $M(q)$, $C(q, \dot{q})$ et $g(q)$ sont censées être connues. Toutefois, les applications pratiques de systèmes robotiques sont confrontées par des contraintes d'incertitude paramétrique. Ainsi, si ces matrices structurelles sont inconnues, la loi proposée dans (III.8) n'est plus valable. Par conséquent, dans ce paragraphe, nous discuterons les défis de l'incertitude dans les paramètres physiques des robots. Pour faire face à cette hypothèse, nous pouvons procéder de deux façons. La première approche consiste à utiliser l'approche adaptative classique (standard). Alors que la deuxième approche consiste à utiliser l'approche adaptative neuronale en utilisant la méthode d'approximation par des fonctions à base radiales (Radial Basis Functions). Ce qui est motivant pour la seconde approche est qu'elle empêche "l'ennui" de l'analyse et le calcul des matrices de régression dans lequel la complexité dépend principalement de la complexité du système étudié.

Considérons l'architecture de réseau de neurone multicouche dont la sortie du perceptron est la suivante :

$$Y_i = \sum_{j=1..N_o}^{N_h} \left[w_{ij} \sigma \left(\sum_{k=1}^{N_i} v_{jk} x_k + \theta_{vj} \right) + \theta_{wi} \right] \tag{III.24}$$

Où $\sigma(\cdot)$ est la fonction d'activation ; N_h est le nombre de neurones de la couche cachée. N_i et N_o sont considérés, successivement, comme étant le nombre de neurones de la couche d'entrée et le nombre de neurones de la couche de sortie. v_{jk} représente les poids d'interconnexion de la première à la deuxième couche ; w_{ij} représente les poids d'interconnexion de la deuxième à la troisième couche ; x_k est le k^{me} entrée du réseau de neurones. θ_{vj} et θ_{wi} désignent les décalages de seuil. En collectant les valeurs v_{jk} et w_{ij} dans des matrices V^T et W^T . Alors l'équation (III.24) peut être écrite en terme de vecteur tel que :

$$Y = W^T \sigma(V^T x) \tag{III.25}$$

$Y(x)$ est considérée comme une fonction de $\mathbb{R}^{N_i}$ dans $\mathbb{R}^{N_o}$ qui représente les incertitudes non structurées. Elle peut être écrite comme suit :

$$Y = W^T \phi(x) + \epsilon \tag{III.26}$$

Où ϕ est la fonction de base et ϵ est le vecteur d'erreur tel que $\|\epsilon\| < \epsilon_M \forall x$.

L'objectif est de développer une loi adaptative neuronale de contrôle de synchronisation et de suivi de trajectoire d'un groupe de robots industriels. Ainsi, supposons que la fonction :

$$H = C(q_1, q_2)q_2 + g(q_1) + M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} \dot{q}_{1ij} + \frac{\partial \alpha}{\partial q_{1i}} \dot{q}_{1i} + \frac{\partial \alpha}{\partial q_0} \dot{q}_0 \right) \quad (\text{III.27})$$

est inconnue. Selon les propriétés d'approximation définies pour les réseaux à fonction de base radiale, H s'écrit :

$$H = \theta^T \varphi + \epsilon \quad (\text{III.28})$$

L'estimation de cette fonction non linéaire est donnée par :

$$\hat{H} = \hat{\theta}^T \varphi(q_1, q_2, q_0, q_{ij}) \quad (\text{III.29})$$

Où $\hat{\theta}$ est l'estimation actuelle des poids du réseau de neurones pour lesquels nous proposons de déterminer une loi d'adaptation.

Par conséquent, en tenant compte des équations (III.27), (III.28) et (III.29) le contrôleur développé (III.8) se présente dans sa version adaptative neuronale :

$$\tau = \hat{\theta}^T \varphi - Pe - \frac{K_z}{2} z \quad (\text{III.30})$$

La loi d'adaptation est conçue comme suit :

$$\dot{\hat{\theta}} = -\Gamma(2\varphi z^T - k\hat{\theta}) \quad (\text{III.31})$$

D'où, nous pouvons maintenant énoncer notre théorème principal :

Théorème : La loi de contrôle (III.30-III.31) assure la synchronisation et le suivi de trajectoire, sous une topologie leader-suiveurs, d'un groupe de robots manipulateurs en présence de l'incertitude au niveau de leurs paramètres physiques.

Preuve : Considérons une nouvelle fonction de Lyapunov de la forme suivante :

$$V_3 = V_2 + \frac{1}{2} \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \tilde{\theta}\} \quad (\text{III.32})$$

avec $\tilde{\theta} = \theta - \hat{\theta}$; Γ est une matrice diagonale définie positive.

La dérivée par rapport au temps de V_3 est :

$$\begin{aligned} \dot{V}_3 &= -e^T Q e + 2e^T P z + 2z^T M \dot{z} + \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}}\} \\ &= 2z^T \left[\tau - C(q_1, q_2)q_2 - g(q_1) - M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} (q_{2i} - q_{2j}) + \frac{\partial \alpha}{\partial q_{1i}} q_2 + \frac{\partial \alpha}{\partial q_0} \right) \right] \\ &\quad - e^T Q e + 2e^T P z + \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}}\} \end{aligned}$$

$$\begin{aligned}
&= 2z^T(\tau - H) - e^T Qe + 2e^T Pz + \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}}\} \\
&= 2z^T(\tau - \theta^T \varphi(q_1, q_2, q_0, q_{ij}) - \epsilon) - e^T Qe + 2e^T Pz + \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}}\} \quad (\text{III.33})
\end{aligned}$$

Posons

$$\tau = \hat{\theta}^T \varphi - Pe - \frac{K_z}{2} z \quad (\text{III.34})$$

Donc,

$$\begin{aligned}
\dot{V}_3 &= 2z^T(-\tilde{\theta}^T \varphi - Pe - \epsilon) - e^T Qe + 2e^T Pz + \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}}\} - z^T k_z z \\
&= -e^T Qe - 2z^T \epsilon - 2z^T \tilde{\theta}^T \varphi - z^T k_z z + \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}}\} \quad (\text{III.35})
\end{aligned}$$

D'où, nous pouvons écrire :

$$\dot{V}_3 \leq -e^T Qe - 2z^T \epsilon - z^T k_z + \text{trace}\{\tilde{\theta}^T (\Gamma^{-1} \dot{\tilde{\theta}} + 2\varphi z^T)\} \quad (\text{III.36})$$

Choisissons

$$\dot{\tilde{\theta}} = -\Gamma(2\varphi z^T - k\hat{\theta}) \quad (\text{III.37})$$

Ce qui produit :

$$\dot{V}_3 \leq -e^T Qe - 2z^T \epsilon - z^T k_z + \text{trace}\left\{\tilde{\theta}^T \left[\Gamma^{-1}(-2\Gamma\varphi z^T + k\Gamma\hat{\theta}) + 2\varphi z^T\right]\right\} \quad (\text{III.38})$$

Ce qui donne :

$$\dot{V}_3 \leq -e^T Qe - 2z^T \epsilon - z^T k_z + k \text{trace}\{\tilde{\theta}^T (\theta - \hat{\theta})\} \quad (\text{III.39})$$

Pour $\epsilon > 0$, l'erreur de l'approximation est bornée par $\|\epsilon\| \leq \epsilon_M$.

Par suite,

$$\dot{V}_3 \leq -\lambda_{\min}(Q) \|e\|^2 - 2\|z\| \epsilon_M + k\theta_M \|\tilde{\theta}\|_F - k \|\tilde{\theta}\|_F^2 - \lambda_{\min}(k_z) \|z\|^2 \quad (\text{III.40})$$

Où $\|\tilde{\theta}\|_F = \sqrt{\text{trace}\{\tilde{\theta}^T \tilde{\theta}\}}$.

Dans sa forme compacte l'équation (III.40) peut s'écrire :

$$\dot{V}_3 \leq - \begin{pmatrix} \|e\| \\ \|\tilde{\theta}\|_F \end{pmatrix}^T \begin{pmatrix} \frac{\lambda_{\min}(Q)}{2} & -\frac{1}{2}k\theta_M \\ -\frac{1}{2}k\theta_M & \frac{\lambda_{\min}(Q)}{2} \end{pmatrix} \begin{pmatrix} \|e\| \\ \|\tilde{\theta}\|_F \end{pmatrix} - \lambda_{\min}(k_z) \|z\|^2 - 2\|z\| \epsilon_M \quad (\text{III.41})$$

III.5 Coordination en présence de retard de communication

Dans la majorité des applications de contrôle dans les domaines d'ingénierie, le retard, lors de la communication entre les systèmes interconnectés, génère un effet prépondérant sur les performances du système global. Ce phénomène de retard de la réception des données dégrade la réflexion et par suite la rapidité de la synchronisation, une contrainte qui ne peut pas être négligée. Par conséquent, pour que l'approche développée soit plus pratique, nous nous intéressons dans ce paragraphe à résoudre le problème de l'information tardive qui peut se produire lors de la communication entre robots. Dans la littérature, il y a plus d'une approche qui prend en considération le retard dans les systèmes multi-agents. Dans [56], Olfati Saber a utilisé le protocole de consensus avec le modèle linéaire suivant :

$$u_i(t) = \sum_{j \neq i} a_{ij} [x_j(t - \tau_{ij}) - x_i(t - \tau_{ij})] \quad (\text{III.42})$$

Où $a_{ij} > 0$ est l'élément de la i^{me} ligne et la j^{me} colonne de la matrice d'adjacence et qui reflète une idée sur la qualité de communication dans le canal; $x_i(t)$ et $x_j(t)$ sont, respectivement, les états des robots i et j ; $u_i(t)$ désigne le vecteur de commande et τ_{ij} est le retard de communication qui correspond au canal e_{ij} . Cependant, [60] Wang et Slotine ont proposé le modèle dynamique des systèmes multi-agents à retard suivant :

$$u_i(t) = \sum_{j \neq i} a_{ij} [x_j(t - \tau_{ij}) - x_i(t)] \quad (\text{III.43})$$

Dans notre approche, nous nous appuyons sur l'algorithme de consensus pour les systèmes multi-agents de simple dynamique proposée dans [60] et nous l'avons appliqué pour les systèmes de forte non-linéarité. D'où la nouvelle expression du contrôleur virtuel :

$$\alpha_i = -k \left(\sum_{j \neq i} A_{ij} [q_{1i}(t) - q_{1j}(t - \tau)] + B_i [q_i(t) - q_0(t - \tau)] \right) + \dot{q}_0 \quad (\text{III.44})$$

Donc on énonce le lemme suivant :

Lemme Considérons $(m + 1)$ robots manipulateurs pleinement actionnés. Ce groupe de robots, sous une architecture de leader-suiveur, forme un réseau d'agents coopératifs. La communication au sein de ce réseau se fait avec un certain retard τ . Malgré la présence de ce retard de communication, la loi de contrôle proposée (III.8-III.44) garantit la synchronisation du mouvement du groupe en suivant une trajectoire commune désirée.

Preuve En tenant compte de l'expression de l'erreur, l'équation (III.44) s'écrit :

$$\alpha_i = -k \left(\sum_{j \neq i} A_{ij} [e_i(t) - e_j(t - \tau)] - \sum_{j \neq i} A_{ij} [q_0(t) - q_0(t - \tau)] \right) + \dot{q}_0$$

$$-k(-B_i e_i(t) - B_i[q_0(t) - q_0(t - \tau)]) \quad (\text{III.45})$$

Sachant que $q_0(t) - q_0(t - \tau) = 0$, l'équation (III.45) peut être simplifiée :

$$\alpha_i = -k \left(\sum_{j \neq i} A_{ij} [e_i(t) - e_j(t - \tau)] - B_i e_i(t) \right) + \dot{q}_0 \quad (\text{III.46})$$

Considérons que α_i est le vrai contrôleur de q_{1i} , donc nous pouvons écrire :

$$\dot{e}_i = -k \left(\sum_{j \neq i} A_{ij} [e_i(t) - e_j(t - \tau)] - B_i e_i(t) \right) \quad (\text{III.47})$$

Récrivons tous les états dans une représentation compacte :

$$\dot{e} = -k(D + B)e(t) + kAe(t - \tau) \quad (\text{III.48})$$

Considérons maintenant la fonction de Lyapunov suivante :

$$V(t) = e^T e + \int_{t-\tau}^0 e^T(s) k(D + B)e(s) ds \quad (\text{III.49})$$

La dérivée de $V(t)$ par rapport au temps donne :

$$\begin{aligned} \dot{V}(t) &= 2e^T \dot{e} - e^T(t - \tau)(D + B)e(t - \tau) \\ &= 2e^T (-k(D + B)e(t) + kAe(t - \tau)) - ke^T(t - \tau)(D + B)e(t - \tau) \\ &= -2ke^T(D + B)e(t) + 2e^T kAe(t - \tau) - ke^T(t - \tau)(D + B)e(t - \tau) \\ &= -ke^T(D + B)e(t) + (-ke^T(D + B)e(t) + 2e^T kAe(t - \tau) - ke^T(t - \tau)(D + B)e(t - \tau)) \end{aligned} \quad (\text{III.50})$$

Donc, posons

$$V = w_1 + w_2$$

où

$$w_1 = -ke^T(D + B)e(t)$$

et

$$w_2 = (-ke^T(D + B)e(t) + 2e^T kAe(t - \tau) - ke^T(t - \tau)(D + B)e(t - \tau))$$

Il est clair que $w_1 \leq 0$. Il nous reste à montrer maintenant que $w_2 < 0$. Or, w_2 peut prendre la forme suivante :

$$w_2 = \begin{pmatrix} e^T(t) & e^T(t - \tau) \end{pmatrix} \begin{pmatrix} -k(D + B) & kA \\ kA & -k(D + B) \end{pmatrix} \begin{pmatrix} e(t) \\ e(t - \tau) \end{pmatrix}$$

Par conséquent, selon le théorème de cercle Gershgorin [61] on peut montrer que la matrice ($w_2 \leq 0$) et par suite on prouve que $V \leq 0$.

III.6 Simulation

Afin de montrer la faisabilité et les performances de l'approche développée. Des simulations ont été effectuées sur un groupe de robots manipulateurs. La modélisation largement utilisée dans la littérature de la dynamique des manipulateurs est donnée par :

$$\begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \begin{pmatrix} \ddot{q}_{11} \\ \ddot{q}_{12} \end{pmatrix} + \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} \begin{pmatrix} \dot{q}_{11} \\ \dot{q}_{12} \end{pmatrix} + \begin{pmatrix} g_1 \\ g_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} \quad (\text{III.51})$$

avec

$$M_{11} = m_2 l_2^2 + m_2 l_1 l_2 \cos q_2$$

$$M_{12} = (m_1 + m_2) l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 \cos(q_2)$$

$$M_{21} = M_{12}$$

$$M_{22} = m_2 l_2^2$$

Les paramètres physiques de robot sont choisis :

$l_1 = 1m$, longueur du premier bras ;

$l_2 = 0.8m$, longueur du deuxième bras ;

$m_1 = 1kg$, la masse du premier bras ;

$m_2 = 0.8kg$, la masse du second bras.

Nous supposons que le groupe de robots est homogène. La simulation est effectuée pour un groupe de 4 robots sous une topologie de leader-suiveur, comme elle est montrée dans la figure (III.5) :

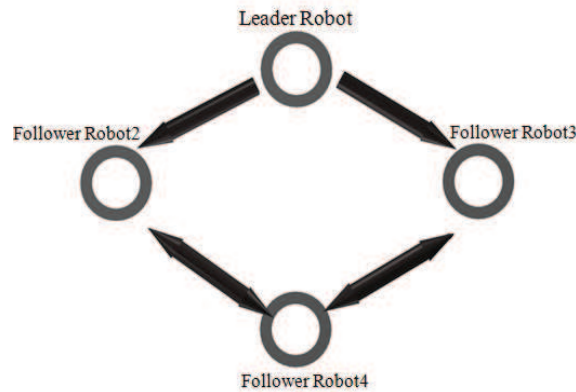


FIGURE III.5 – Topologie leader-suiveur.

Les conditions initiales des robots suiveurs sont définies comme suit :

robot1 : $(q_{11}, q_{12}) = (20^\circ, 20^\circ)$,

robot2 : $(q_{21}, q_{22}) = (10^\circ, 10^\circ)$,

robot3 : $(q_{31}, q_{32}) = (-10^\circ, -10^\circ)$.

Avec (q_{11}, q_{12}) , (q_{21}, q_{22}) et (q_{31}, q_{32}) les coordonnées des articulations de chaque robot.

Soit la trajectoire désirée définie par l'équation suivante :

$$q_d = \sin(t)\sin(2\cos(t)).$$

Afin de modéliser les connexions au sein du réseau, nous définissons la matrice d'adjacence suivante :

$$A = \begin{pmatrix} 0 & 0 & 0.8 \\ 0 & 0 & 0.7 \\ 0.8 & 0.7 & 0 \end{pmatrix} \quad (\text{III.52})$$

L'échange des données au sein de réseau se traduit par la matrice de Laplace suivante :

$$L = \begin{pmatrix} 1 & 0 & -0.8 \\ 0 & 1 & -0.7 \\ -0.8 & -0.7 & 2 \end{pmatrix} \quad (\text{III.53})$$

III.6.1 Contrôle non adaptatif

On présente d'abord les résultats de simulation de l'approche non adaptative. Les gains de contrôle utilisés dans les tests de simulation sont présentés dans le tableau ci dessous :

TABLE III.1 – Gains de contrôle (cas non adaptatif)

Gains de contrôle	suiveur1	suiveur2	suiveur3
k	50	40	40
p	100	100	200
b	0.5	0.4	0

La figure (III.6) illustre les trajectoires de 3 robots suiveurs ainsi que la trajectoire du robot leader. Les trois robots suiveurs synchronisent rapidement leurs mouvements et suivent la trajectoire de leader. Ce qui prouve l'efficacité de la loi de contrôle qui respecte ses deux objectifs. On remarque bien que les robots n'ont pas les mêmes points de départ. D'où le fait que la synchronisation des vitesses soit une exigence de performance de la stratégie. Dans ce contexte, la figure (III.7) illustre les différentes vitesses des robots en

groupe. L'erreur de suivi ainsi que l'erreur de synchronisation sont présentées, respectivement, dans les figures (III.8) et (III.9).

Nous remarquons une synchronisation rapide et efficace des mouvements des trois robots. En effet, les trois robots qui partent en réalité de trois positions différentes ne prennent que quelques secondes pour synchroniser leurs mouvements et atteindre l'objectif désiré. En ce qui concerne les résultats de simulations en présence de retard d'information lors de la communication entre robots (voir figure III.10, figure III.11, figure III.12), nous observons que la perturbation, que ce soit au niveau de la synchronisation ou au niveau du suivi de trajectoire, est proportionnelle au temps de retard. Toutefois, cette perturbation ne peut pas influencer globalement l'objectif principal du groupe coopératif.

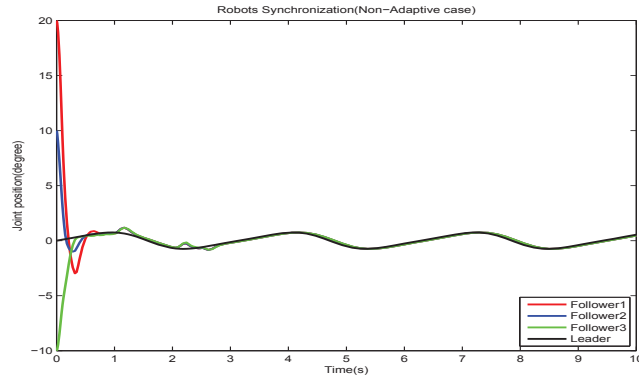


FIGURE III.6 – Synchronisation et suivi de trajectoire.

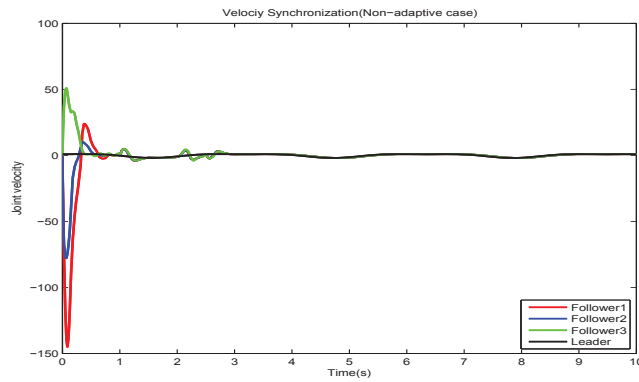


FIGURE III.7 – Synchronisation des vitesses.

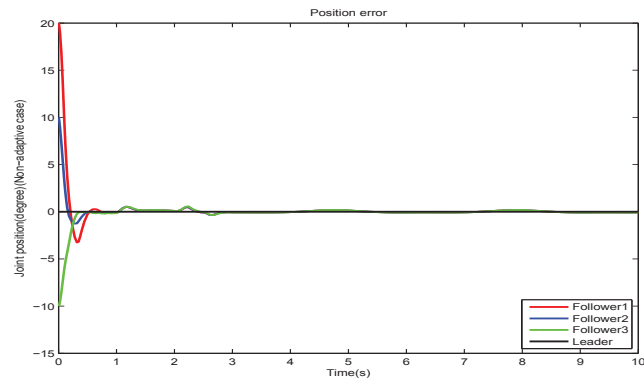


FIGURE III.8 – Erreur de suivi.

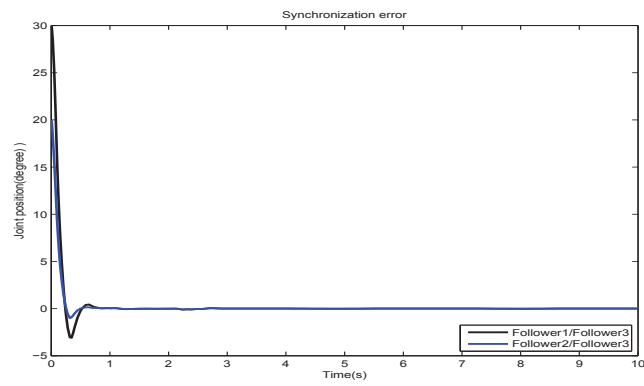


FIGURE III.9 – Erreur de synchronisation.

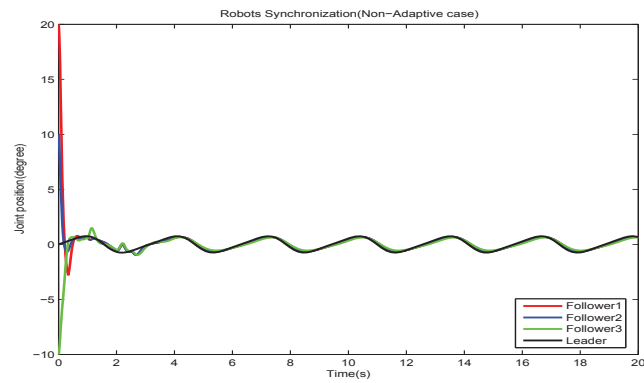


FIGURE III.10 – synchronisation avec un retard de 0.5s

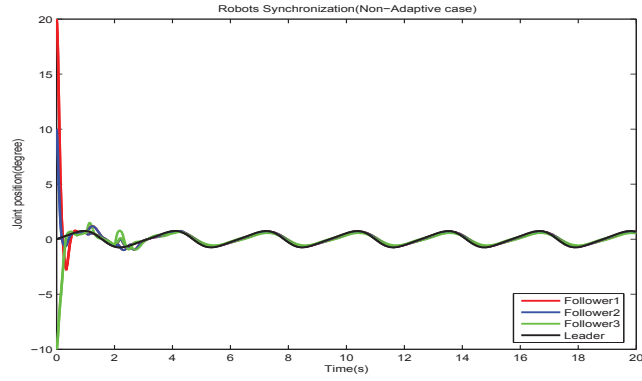


FIGURE III.11 – synchronisation avec un retard de 1s

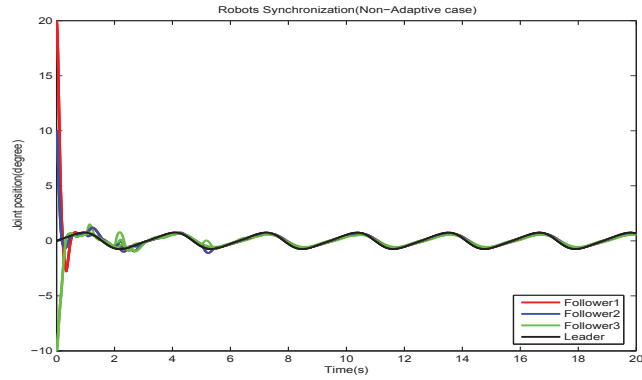


FIGURE III.12 – synchronisation avec un retard de 5s

III.6.2 Contrôle adaptatif

Dans ces simulations, nous supposons que les paramètres physiques des robots ne sont pas connus. Ainsi, une adaptation neuronale est utilisée pour estimer en ligne l'incertitude des paramètres. Les gains de contrôle utilisés pour la simulation sont présentés ci-dessous :

L'architecture utilisée pour la fonction gaussienne est de la forme suivante :

$$\phi_i = \exp \left[\frac{-\|x - c_i\|^2}{2\gamma_i^2} \right]$$

Où, $x = [q_i^T, \dot{q}_i^T, q_0^T]^T$ est le vecteur d'entrée de la Fonction Radiale de Base (FRB) ; $c_i \in \mathbb{R}^2$ définit le centre et $\gamma^2 \in \mathbb{R}$ est la variance. Les valeurs de c_i et de γ^2 sont fixées,

TABLE III.2 – Gains de contrôle (cas adaptatif)

Gains de contrôle	suiveur1	suiveur2	suiveur3
k	50	40	40
p	200	200	250
b	0.5	0.4	0
Γ	0.002	0.002	0.002

respectivement à $[100, 100, 100]^T$ et 50.

L'évolution de comportement de chaque robot est illustré dans la figure (III.13). Un aperçu sur le graphique montre bien une synchronisation parfaite des trois robots en suivant la même trajectoire désirée. Ce qui prouve l'efficacité de la stratégie développée, en présence des incertitudes paramétriques. La figure (III.14) affiche la synchronisation des vitesses. Les erreurs de position et les erreurs de synchronisations sont affichées, respectivement, dans la figure (III.15) et dans la figure (III.16).

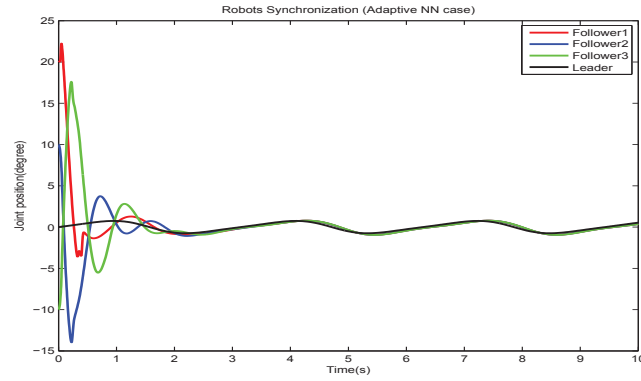


FIGURE III.13 – Synchronization adaptative du réseau de robots.

En comparaison avec le contexte non adaptatif, nous signalons la présence de quelques perturbations au niveau de la synchronisation ainsi qu'au niveau de suivi de trajectoire au départ de l'application. Ce phénomène ne prend que quelques secondes avant que les trois robots reprennent la trajectoire désirée en synchronisant leurs mouvements.

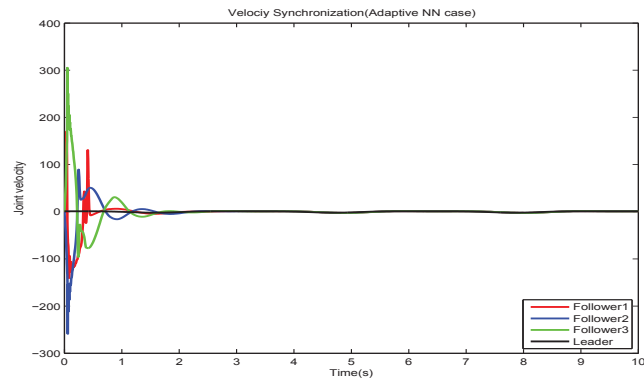


FIGURE III.14 – Synchronization adaptative des vitesses.

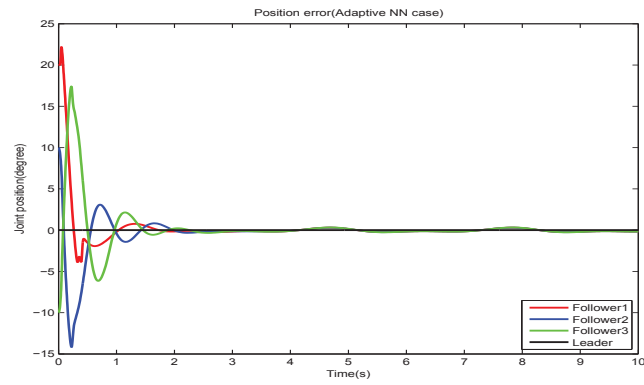


FIGURE III.15 – Erreurs de suivi : cas adaptatif.

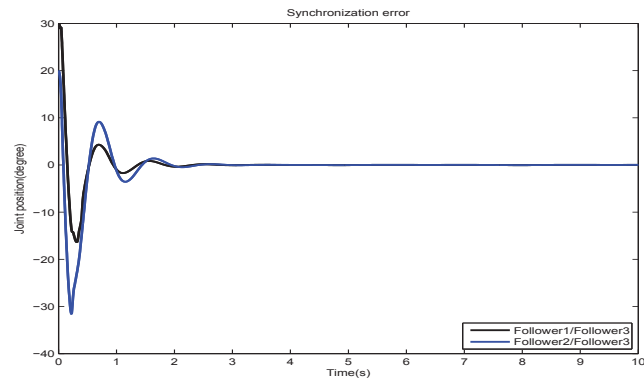


FIGURE III.16 – Erreurs de synchronisation : cas adaptatif.

III.7 Conclusion

Tout d'abord, en utilisant une combinaison de la technique du backstepping et la théorie des graphes, l'approche proposée vise à synchroniser les positions et les vitesses d'un groupe de robots en suivant une trajectoire commune. L'architecture étudiée est une topologie leader-suiveur. L'approche de communication permet la communication suiveur-suiveur ainsi que le retour d'information du suiveur au leader. La stratégie proposée est hybride, dans le sens qu'elle n'impose pas que chaque agent soit connecté directement au leader. Dans une deuxième étape, la loi de contrôle développée est étendue pour traiter les incertitudes paramétriques. Afin de traiter ces incertitudes, on a recours à l'approche adaptative neuronale par appel aux réseaux de neurones à Fonction de Base Radiale. L'approche présentée offre aussi l'opportunité de traiter les retards de communication dans ces groupes de robots. D'où une extension de résolution des délais est proposée dans notre étude. Les simulations ont montré des bonnes performances, en appliquant cette stratégie de contrôle à un groupe de robot manipulateurs sous une topologie de leader-suiveur.

Chapitre IV

Synchronisation des systèmes de Lagrange sous-actionnés

Ce chapitre traite le problème de synchronisation des systèmes de Lagrange sous-actionnés. Nous considérons ici un réseau d'une classe particulière de systèmes sous-actionnés. Les systèmes sous actionnés sont caractérisés essentiellement par le fait que le nombre des actionneurs est inférieur au nombre de leurs degrés de liberté. Afin de satisfaire aux exigences de la productivité moderne : faible consommation et grande vitesse, les robots sous actionnés sont considérés comme une solution très performante. Nous proposons une nouvelle stratégie pour la synchronisation de ce type de systèmes basée sur l'utilisation de la passivité comme outil de conception afin de résoudre le problème en main. D'abord le système sous-actionné est transformé en une classe de systèmes non linéaires sous une structure en chaîne. En effet, en utilisant un retour d'état statique non linéaire convenable, le système sous actionné étudié peut être représenté sous une forme interconnectée. Le contrôle coopératif est conçu en utilisant une combinaison de la procédure standard de backstepping et la théorie de passivité. La technique de passivité permet la construction récursive des sorties de synchronisation qui seront partagées au sein du groupe. La technique de conception proposée est basée sur une structure de communication modélisée par un graphe orienté fortement connecté. Les résultats de simulations numériques illustrent bien l'efficacité de l'approche proposée.

IV.1 Introduction

Au cours du siècle passé, il y a eu un intérêt croissant et une recherche importante sur les systèmes sous-actionnés. Ces types de systèmes sont caractérisés par un nombre d'ac-

tionneurs inférieur au nombre de degrés de liberté. Le sous actionnement de ces systèmes pourrait être une propriété intrinsèque de leurs dynamiques. Il peut être la conséquence de la conception mécanique faite afin de réduire le coût du système. Il pourrait aussi être dû aux raisons purement technologiques ou liés essentiellement à des applications particulières. Ce type de systèmes est fréquemment rencontré dans les applications des robots marcheurs [54], [59], les véhicules aérospatiaux [83], [84] et les systèmes non holonomes en général [68]. En effet, la commande de robots sous actionnés a attiré beaucoup d'attention comme étant un domaine de recherche très actif [58], [68], [83]. Des travaux récents traitent la commande en mode glissant [60], la commande intelligente [97] et la commande hybride de systèmes non-linéaires sous actionnés [66]. Wang *et al.* [60] présentent une loi de commande en mode glissant afin de contrôler un système sous actionné du second ordre. Kobayashi et Yoshikawa [106] ont étudié le contrôle d'un robot manipulateur à une seule articulation passive. Les effets de la dynamique couplée des robots sous actionnés ont été largement discutés [123]. Un travail très récent qui s'intéresse aux robots sous actionnés flexibles a été proposé [16]. A titre d'exemple, ces systèmes comprennent des manipulateurs planaires avec articulations passives, des robots type pendule, des robots rigides avec des articulations élastiques ou des bras flexibles, des robots sous-marins, des navires de surface, etc. La dynamique des propriétés pour ce genre de systèmes présente habituellement des contraintes non-holonômes qui sont généralement non-intégrables et qui ne remplissent pas les conditions nécessaires de la Brockett [132]. Ce qui complique de plus le problème du contrôle.

IV.2 Systèmes multi-robots sous actionnés et groupes coopératifs

Actuellement, l'attention des chercheurs est détournée vers la conception des théories de contrôle de groupes de systèmes coopératifs. En raison des nombreux avantages qu'apporte la coopération des systèmes dans l'exécution de certaines tâches complexes, la recherche moderne s'intéresse à développer des stratégies de contrôle coopératif. La coopération d'un groupe de systèmes assure le remplissage d'une tâche complexe, comme le transport d'une grande charge ou la manipulation d'un objet lourd, d'une manière plus facile, plus efficace et moins coûteuse qu'un seul système qui se trouve souvent incapable de parvenir à respecter l'objectif. Très récemment, Rodriguez et Nijmeijer [134], [135] ont abordé le problème de la synchronisation mutuelle et externe des positions et des vitesses articulaires des robots rigides et flexibles. Le problème de synchronisation de robots de différents types (réseaux de robots hétérogènes) a été abordé sous différents angles [12]. En

particulier, on cite la conception d'une stratégie de contrôle en absence de l'information sur les vitesses articulaires [134]. L'objectif était d'assurer la synchronisation coopérative dans un réseau de robots manipulateurs hétérogènes en utilisant les observateurs. D'autre part, Pettersen *et al.* [35] ont proposé une synchronisation mutuelle de deux robots manipulateurs dans l'espace de fonctionnement et non pas dans l'espace articulaire (le cas contraire de l'étude présentée par Rodriguez [134]). Saber [83] a développé une forme en cascade de robots sous actionnés fondée sur la symétrie mécanique. Arai et Taehi [120] utilisent une commande non linéaire afin de contrôler les robots sous actionnés dans l'espace articulaire. Dans [103] une commande optimale pour les systèmes sous actionnés est proposée. Dans une autre référence [12], les auteurs proposent un contrôleur de synchronisation mutuelle plus simple que celui de Rodriguez [134]. Ce contrôleur ne nécessite que le couplage des robots "les plus" voisins afin d'assurer une synchronisation mutuelle. Dans ce travail [12], le contrôleur proposé se base sur la théorie de contraction qui permet d'obtenir non seulement la synchronisation exponentielle mais aussi une convergence exponentielle vers la trajectoire désirée. Une conception d'une loi de contrôle de synchronisation des réseaux de systèmes lagrangiens incertains a été proposée [10]. Cette conception englobe l'algorithme de consensus pour les systèmes à double intégrateur. L'approche développée élève le problème de la perte de la symétrie dans les graphes de communications orientés. Comme elle permet aussi l'utilisation des graphes orientés contenant l'arbre couvrant. Dans un contexte de télé-opération, les références [37] et [41] proposent une stratégie de contrôle de synchronisation qui se base sur la théorie de passivité. La robustesse de la synchronisation est prouvée même en présence des retards constants et arbitraires dans le réseau de communication. Dans une autre référence [24], l'auteur a souligné l'utilité de la propriété qui prend le système global formé de sous systèmes passifs interconnectés comme étant un système passif. Cette propriété a été utilisée dans la conception d'une loi de contrôle des systèmes distribués. Plus tard, en appliquant la technique du backstepping et en se basant sur le travail de Chopra [37], les références [17] et [18] ont proposé un nouveau contrôleur passif de synchronisation. Dans la plupart des recherches citées ci-dessus, la synchronisation des systèmes multi-agents à une dynamique linéaire ou à une dynamique non linéaire et surtout la synchronisation des robots pleinement actionnés sont étudiées. Cependant, il y a peu de résultats sur la synchronisation des systèmes sous actionnés. Dans ce travail, on propose une loi de synchronisation d'une classe de robots sous actionnés coopératifs. En se basant sur la recherche de Sun [70], afin de contourner les problèmes causés par le phénomène de sous actionnement, le système dynamique de chaque robot manipulateur est converti à un système à structure chaînée. Cette transformation n'est qu'une préparation d'une structure qui permet l'application de la théorie de passivité combinée avec la technique du backstepping. Cette combinaison de la théorie de

passivité et de la procédure alternative de backstepping propose la conception d'une loi de synchronisation pour une classe de systèmes sous actionnés en réseau. La stratégie proposée est une extension de la méthodologie du protocole de consensus passif développée pour des systèmes qui pourront être linéarisés par retour d'état. Cette méthodologie est conçue afin d'assurer la synchronisation du groupe et le suivi de trajectoire d'un leader [24].

IV.3 Formulation du problème

Tout d'abord, nous commençons par donner une définition formelle des systèmes mécaniques sous-actionnés (IV.1).

Définition 4.1 ([22]) Soit le système mécanique affine en la commande, décrit par l'équation suivante :

$$\ddot{q} = f(q, \dot{q}) + G(q)u \quad (\text{IV.1})$$

avec q le vecteur des positions généralisées, $f(\cdot)$ est le champ de vecteur traduisant la dynamique du système, $\dot{q}$ est le vecteur des vitesses généralisées, G est la matrice d'entrée et u est le vecteur des entrées généralisées. Le système (IV.1) est dit sous-actionné si les entrées généralisées ne peuvent pas commander les accélérations instantanées dans toutes les directions de l'espace des configurations. Ceci s'explique quand $\text{rang}(G) < \dim(q)$, où $\dim(q)$ est généralement définie comme étant le nombre de degrés de libertés de (IV.1).

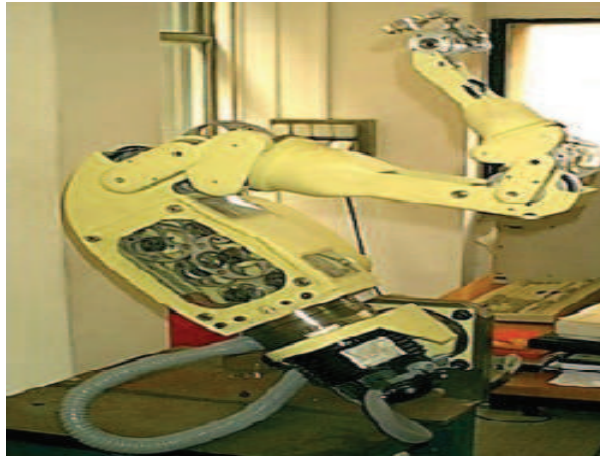


FIGURE IV.1 – Robot sous-actionné

La notion sur le principe de passivité est présentée en détails dans [102]. Pour la clarté du développement, on va brièvement rappeler quelques propriétés essentielles qui seront utilisées dans les paragraphes suivants. Soit un système non linéaire dont la dynamique est décrite comme suit :

$$\begin{aligned}\dot{x} &= f(x, u) \\ y &= h(x), \quad x \in \mathbb{R}^n, \quad u, y \in \mathbb{R}^m\end{aligned}\tag{IV.2}$$

Le système non linéaire (IV.2) est dit passif s'il existe une fonction de stockage ou énergie V qui soit continue et différentiable sur $V(x) \geq 0$ est une fonction semi-définie positive $W(x)$ telle que :

$$\dot{V}(x) \leq -W(x) + u^\top y\tag{IV.3}$$

La dynamique d'un robot manipulateur peut s'écrire sous la forme suivante :

$$M(q)\ddot{q} + N(q, \dot{q}) = G(q)\tau\tag{IV.4}$$

Où q est un vecteur de dimension $n \times 1$ et τ est le contrôleur du système de dimension $m \times 1$; $M(q)$ est une matrice d'inertie symétrique et définie positive et $N(q, \dot{q})$ est un vecteur reflétant les forces de Coriolis et centrifuges et l'énergie potentielle du système. Dans le cas où $n = m$, le robot manipulateur est dit pleinement actionné. Cependant, il est décrit sous actionné dans le cas où $m < n$. Dans ce cas, le robot a un nombre d'actionneurs (entrées de contrôle) moins grand que son nombre de degrés de liberté. Par ailleurs, si la matrice G est de rang plein, alors à travers un changement de variables d'état et de commande, l'équation dynamique du robot manipulateur sous actionné peut se présenter sous une structure partitionnée :

$$\begin{aligned}M_1(q)\ddot{q}_1 + M_2(q)\ddot{q}_2 + N_1(q, \dot{q}) &= 0 \\ M_3(q)\ddot{q}_1 + M_4(q)\ddot{q}_2 + N_2(q, \dot{q}) &= G_a(q)\tau\end{aligned}\tag{IV.5}$$

Où q_1 est un vecteur représentant les articulations sous actionnées et q_2 est un vecteur qui représente les articulations motorisées. La matrice $G_a(q)$ est une matrice inversible pour tout vecteur q . Une caractéristique très importante des systèmes sous actionnés est leur susceptibilité de transformation à des systèmes en cascade. En effet, comme il était prouvé dans un travail de "pionnier" [136], en utilisant un retour partiel d'état, la représentation dynamique sous actionnée peut être transformée en une structure en cascade. Ce qui produit une représentation dynamique fortement couplée. D'où une entrée de contrôle conçue par une stratégie où on doit contrôler plusieurs degrés de liberté dans un même temps. Ce défi, en quelque sorte, va durcir le contrôle et la stabilisation de tous les degrés

de liberté du système. La référence [70] propose, à travers l'appel d'un retour d'état non régulier qui permet de mettre le système sous forme triangulaire, une transformation du système dynamique sous actionné à un système dynamique en chaîne de second ordre. Alors, la procédure de backstepping peut être facilement appliquée à cette structure en chaîne. Pour une représentation claire, on va considérer dans notre étude une classe de robots polaires avec deux articulations flexibles [74].

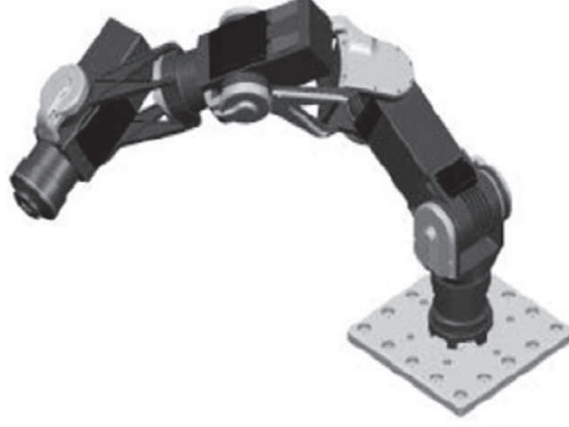


FIGURE IV.2 – Robot polaire

Cette classe de systèmes est représentée par la dynamique suivante :

$$\begin{aligned} M_{1i}(q_{1i})\ddot{q}_{1i} + M_{2i}\ddot{q}_{2i} + C_{1i}(q_{1i}, \dot{q}_{1i})\dot{q}_{1i} \\ + K_{ei}(q_{1i} - q_{2i}) + \Delta_{1i} = 0 \\ M_{2i}^\top \ddot{q}_{1i} + M_{3i}\ddot{q}_{2i} + K_{ei}(q_{1i} - q_{2i}) + \Delta_{2i} = \tau_i \end{aligned} \quad (\text{IV.6})$$

Où i décrit le *ime* robot dans le groupe coopératif. Le vecteur $(q_{1i} = (q_{1,1}^i, q_{1,2}^i))$ et le vecteur $(q_{2i} = (q_{2,1}^i, q_{2,2}^i))$ représentent, respectivement, les positions et les vitesses des deux articulations. $C_{1i}(q_{1i}, \dot{q}_{1i})$ est la matrice de Coriolis ; K_{ei} est une matrice diagonale de rigidité. Δ_{1i} et Δ_{2i} englobent les forces gravitationnelles et les forces de frottement. M_{2i} est une matrice strictement triangulaire supérieure, donnée par l'expression suivante :

$$M_{2i} = \begin{pmatrix} 0 & m_{1,2}^i \\ 0 & 0 \end{pmatrix}.$$

Finalement, τ est le vecteur des couples. Supposons un groupe de N robots sous actionnés dont chacun est décrit par l'équation dynamique (IV.6). Chaque robot doit communiquer avec ses voisins afin d'échanger les informations concernant leurs positions et leurs vitesses et par suite synchroniser leurs mouvements. En se basant sur les concepts de la théorie des graphes, on suppose que la communication au sein du réseau de robots coopératifs peut être représentée par un graphe fortement connexe. L'objectif est de développer un

contrôleur distribué. De son implémentation dans la dynamique (IV.6) résultera la synchronisation du groupe de robots. La synchronisation du groupe n'est que le résultat de la synchronisation des articulations de chaque robot manipulateur avec celles des robots voisins, qui se traduit effectivement par les objectifs mathématiques suivants :

$$\begin{aligned}\lim_{t \rightarrow \infty} |q_1^i - q_1^j| &= 0, \quad \forall i \neq j \\ \lim_{t \rightarrow \infty} |q_2^i - q_2^j| &= 0, \quad \forall i \neq j\end{aligned}\tag{IV.7}$$

Dans ce travail, nous proposons une combinaison de la technique du backstepping et de la théorie des graphes afin de concevoir une loi de commande de synchronisation d'un groupe de robots sous actionnés coopératifs. Cependant, le système dynamique représenté par l'équation (IV.6) n'est pas adopté pour appliquer notre stratégie de commande. L'idée était de transformer cette équation à une structure en chaîne. Une telle forme qui permettra la construction et l'application d'une théorie de contrôle basée sur la procédure du backstepping et de la technique de passivité.

IV.4 Architecture de la stratégie de contrôle de la synchronisation coopérative

La conception du nouveau contrôleur distribué de synchronisation coopérative passe par deux étapes essentielles. Au début un retour d'état non régulier est utilisé afin de transformer le système dynamique (IV.6) en un système non linéaire en chaîne. Dans une deuxième étape, chaque portion de cette structure en chaîne est décomposée en deux sous systèmes passifs interconnectés. En effet, cette décomposition est assurée par un contrôleur virtuel distribué et approprié. L'itération de cette procédure en utilisant la technique du backstepping conduit à déterminer le contrôleur distribué de synchronisation et de suivi de trajectoire.

IV.4.1 Transformation de la représentation dynamique sous actionnée en une structure en chaîne

La première étape de la conception de la loi de synchronisation était de transformer le système dynamique (IV.6) sous une forme de retour d'état dynamique rendant le système

partiellement linéaire [136]. Posons la loi de contrôle suivante :

$$\begin{aligned}\tau_i = & -M_{2i}^\top M_{1i}^{-1}(q_{1i}) \left(C_{1i}(q_{1i}, \dot{q}_{1i}) \dot{q}_{1i} + K_{ei}(q_{1i} - q_{2i}) \right. \\ & \left. + \Delta_{1i} \right) + K_{ei}(q_{2i} - q_{1i}) + \Delta_{2i} \\ & + \left(M_{3i} - M_{2i}^\top M_{1i}^{-1}(q_{1i}) M_{2i} \right) u_i\end{aligned}\quad (\text{IV.8})$$

Avec $u_i = [u_{1i}^\top, u_{2i}^\top]^\top$ est une entrée implémentée dans l'expression du couple et qui va englober après l'expression du contrôleur de synchronisation. Tenons compte de l'expression (IV.8), l'équation (IV.6) peut s'écrire sous la forme suivante :

$$\begin{aligned}\ddot{q}_{1i} &= H(q_{1i}, \dot{q}_{1i}) + M_{1i}^{-1}(q_{1i}) \left(K_{ei} q_{2i} - M_{2i} u_i \right) \\ \ddot{q}_{2i} &= u_i\end{aligned}\quad (\text{IV.9})$$

Où $H(q_{1i}, \dot{q}_{1i}) = [H_1^i, H_2^i]^\top$ est une fonction non linéaire. Vu la forme triangulaire de la matrice M_{2i} , seulement le contrôleur u_{2i} peut être choisi comme un retour d'état linéarisant $q_{1,1}^i$ afin d'avoir une forme dynamique en chaîne. Choisissons alors le contrôleur u_i sous la forme suivante :

$$\begin{aligned}u_{2i} &= \frac{1}{m_{1,2} \delta_{1i}(q_{1i})} \left(H_1^i - q_{1,2}^i + [\delta_{1i}(q_{1i}), \delta_{2i}(q_{1i})] K_{ei} q_{2i} \right) \\ u_{1i} &= v_i\end{aligned}\quad (\text{IV.10})$$

où $\delta_{ki}, k = 1, 2$ sont les éléments de la première ligne de l'inverse de la matrice $M_{1i}(q_{1i})$. Le système dynamique (IV.9) et l'expression du contrôleur u_{2i} (IV.10) en boucle fermée peuvent produire la forme dynamique en chaîne :

$$\begin{aligned}\ddot{q}_{1,1}^i &= q_{1,2} \\ \ddot{q}_{1,2}^i &= f_{1i}(q_{1i}, \dot{q}_{1i}) + g_{1i}(q_{1i}) \dot{q}_{2,2}^i \\ \ddot{q}_{2,2}^i &= f_{2i}(q_{1i}, \dot{q}_{1i}, q_{2,2}^i) + g_{2i} \dot{q}_{2,1}^i \\ \ddot{q}_{2,1}^i &= v_i\end{aligned}\quad (\text{IV.11})$$

Avec $f_{1i}(q_{1i}, \dot{q}_{1i})$, $f_{2i}(q_{1i}, \dot{q}_{1i}, q_{2,2}^i)$, $g_{1i}(q_{1i})$ et g_{2i} des fonctions non linéaires. La structure en chaîne (IV.11) obtenue permet d'appliquer la procédure du backstepping pour la stabilisation du système global comme il était prouvé dans [70]. Toutefois, dans notre travail, en plus de suivi de trajectoire et de la stabilisation, nous sommes intéressés à la synchronisation du mouvement. La conception va se baser sur la technique de backstepping et sur l'approche de passivité afin d'atteindre l'objectif de la synchronisation. Donc la stratégie de contrôle va exploiter une combinaison de ces deux techniques de commande afin de concevoir une loi capable d'assurer la synchronisation du mouvement et le suivi de trajectoire pour un réseau de robots coopératifs.

IV.4.2 Commande backstepping non linéaire basée sur la passivité

L'objectif maintenant est d'adopter le système sous actionné à l'application de la procédure du backstepping. Commençons par définir une erreur de la forme suivante :

$$z_{1i} = \dot{q}_{1,1}^i - v_1(t)$$

Avec $v_1(t)$ est la vitesse désirée à suivre par $q_{1,1}^i$. On reprend l'écriture de l'équation (IV.11) en tenant compte des variables récemment définies et qui seront intercalées dans la structure dynamique pour avoir une forme d'intégrateurs en chaîne :

$$\begin{aligned} \dot{z}_{1i} &= \dot{q}_{1,2}^i - \dot{v}_1(t) \\ \dot{q}_{1,2}^i &= x_{1,2}^i \\ \dot{x}_{1,2}^i &= f_{1i}(q_{1i}, \dot{q}_{1i}) + g_{1i}(q_{1i})q_{2,2}^i \\ \dot{q}_{2,2}^i &= x_{2,2}^i \\ \dot{x}_{2,2}^i &= f_{2i}(q_{1i}, \dot{q}_{1i}, q_{2,2}^i) + g_{2i}q_{2,1}^i \\ \dot{q}_{2,1}^i &= x_{2,1}^i \\ \dot{x}_{2,1}^i &= v_i \end{aligned} \tag{IV.12}$$

La procédure backstepping commence par le premier sous système de la dynamique (IV.12) :

$$\dot{z}_{1i} = \dot{q}_{1,2}^i - \dot{v}_1(t) \tag{IV.13}$$

Si nous considérons que $q_{1,2}^i$ est le contrôleur du système (IV.13), il sera facile de concevoir $q_{1,2}^i$ de façon que le système soit passif avec z_{1i} comme sortie, u_i comme entrée et une fonction quadratique de stockage d'énergie $V_{1i} = \frac{1}{2}z_{1i}^2$. Afin de prouver cette méthodologie, choisissons alors $q_{1,2}^i = -k_{1i}z_{1i} + \dot{v}_1(t) + \tau_{1i}$. Tenons compte de l'équation (IV.13), la dérivée de la fonction de Lyapunov V_{1i} produit la forme suivante :

$$\begin{aligned} \dot{V}_{1i} &= -k_{1i}z_{1i}^2 + z_{1i}\tau_{1i} \\ &\leq z_{1i}\tau_{1i} \end{aligned} \tag{IV.14}$$

Ce qui prouve que l'équation dynamique (IV.13) est passive de z_{1i} à τ_{1i} . Afin d'assurer une synchronisation des articulations $q_{1,1}^i \forall i = 1, \dots, N$, nous posons une entrée auxiliaire τ_{1i} inspirée de l'algorithme de consensus de la forme suivante :

$$\tau_{1i} = - \sum_{j=1}^N (q_{1,1}^i - q_{1,1}^j), \quad i = 1, \dots, N \tag{IV.15}$$

L'expression de τ_{1i} est reconnue comme étant une modélisation du protocole de consensus pour un groupe de N agents. Dans une écriture matricielle et afin de prendre en considération le système global, ce protocole de consensus s'écrit sous la forme suivante :

$$U_1 = -L\mathbf{q}_{1,1} \quad (\text{IV.16})$$

Où $U_1 = [\tau_{11}, \tau_{12}, \dots, \tau_{1N}]^\top$, $\mathbf{q}_{1,1} = [q_{1,1}^1, q_{1,1}^2, \dots, q_{1,1}^N]^\top$ et L est la matrice Laplacienne du graphe qui modélise la communication et l'échange d'informations entre agents au sein du réseau. Nous supposons que la communication est bidirectionnelle ce qui donne une matrice Laplacienne symétrique et semi définie positive avec un zéro comme une somme sur les lignes ainsi que sur les colonnes. Cependant, le contrôleur $q_{1,2}$ n'est pas le contrôleur réel du système (IV.12). Ce qui nous permet de poser le contrôleur virtuel dont l'expression est :

$$\alpha_{1i} = -k_{1i}z_{1i} + \dot{v}_1(t) + \tau_{1i}$$

et d'introduire l'erreur suivante :

$$z_{2i} = q_{1,2}^i - \alpha_{1i}.$$

Par suite, les systèmes (IV.13) et (IV.14) se présentent sous l'écriture suivante :

$$\begin{aligned} \dot{V}_{1i} &= -k_{1i}z_{1i}^2 + z_{1i}\tau_{1i} + z_{1i}z_{2i} \\ \dot{z}_{1i} &= -k_{1i}z_{1i} + \tau_{1i} + z_{2i} \end{aligned} \quad (\text{IV.17})$$

L'étape suivante est de considérer un état intermédiaire $x_{1,2}^i$ qui jouera le rôle d'un contrôleur afin d'assurer la passivité du système. Définissons un contrôleur virtuel α_{2i} pour les sous systèmes d'état $\dot{q}_{1,2}^i$ et posons la variable d'erreur $z_{3i} = x_{1,2}^i - \alpha_{2i}$ et la fonction de Lyapunov suivante $V_{2i} = V_{1i} + \frac{1}{2}z_{2i}^2$. La dérivée de cette fonction de Lyapunov prend la forme suivante :

$$\dot{V}_{2i} = -k_{1i}z_{1i}^2 + z_{1i}\tau_{1i} + z_{2i}(z_{3i} + \alpha_{2i} + z_{1i} - \dot{\alpha}_{1i}) \quad (\text{IV.18})$$

Afin de rendre ce sous système de $\dot{q}_{1,2}^i$ un système passif de z_{2i} à z_{3i} , nous choisissons :

$$\alpha_{2i} = -k_{2i}z_{2i} - z_{1i} + \dot{\alpha}_{1i}.$$

Ce qui produit les résultats suivants :

$$\begin{aligned} \dot{V}_{2i} &= -k_{1i}z_{1i}^2 - k_{2i}z_{2i}^2 + z_{1i}\tau_{1i} + z_{2i}z_{3i} \\ \dot{z}_{2i} &= -k_{2i}z_{2i} - z_{1i} + z_{3i} \end{aligned} \quad (\text{IV.19})$$

Ensuite, nous procédons étape par étape, en introduisant à chaque fois les erreurs suivantes :

$$z_{li} = \zeta_i - \alpha_{(l-1)i}(\zeta_i, z_{3i}, z_{4i}, \dots, z_{(l-1)i}), \quad l = 3, \dots, 7 \quad (\text{IV.20})$$

Avec ζ_i représente l'état intermédiaire (le cas de $x_{1,2}^i, q_{2,2}^i, \dots$ de l'équation (IV.12)).

La fonction de Lyapunov est développée sous la forme suivante :

$$V_{(l)i} = V_{(l-1)i} + \frac{1}{2}z_{(l-1)i}^2, \quad l = 3, \dots, 6 \quad (\text{IV.21})$$

Posons les contrôleurs virtuels α_{li} selon les expressions ci-dessous :

$$\begin{aligned} \alpha_{3i} &= \frac{1}{g_{1i}}(-k_{3i}z_{3i} - z_{2i} - f_{1i} + \dot{\alpha}_{2i}) \\ \alpha_{4i} &= -k_{4i}z_{4i} - g_{1i}z_{3i} + \dot{\alpha}_{3i} \\ \alpha_{5i} &= \frac{1}{g_{2i}}(-k_{5i}z_{5i} - z_{4i} - f_{2i} + \dot{\alpha}_{4i}) \\ \alpha_{6i} &= -k_{6i}z_{6i} - g_{2i}z_{5i} + \dot{\alpha}_{5i} \end{aligned} \quad (\text{IV.22})$$

Ce qui produit la formule suivante :

$$\dot{V}_{(l)i} = - \sum_{l=1}^6 k_{li}z_{li}^2 + z_{1i}\tau_{1i} + z_{7i}z_{6i} \quad (\text{IV.23})$$

Considérons maintenant la variable Z_{7i} comme étant une sortie virtuelle. Alors, l'entrée réelle v_i va apparaître en dérivant par rapport au temps cette sortie virtuelle. On propose $V_{7i} = V_{6i} + \frac{1}{2}z_{7i}^2$ comme étant une fonction de Lyapunov, ce qui produit la dérivée par rapport au temps suivante :

$$\dot{V}_{(l)i} = - \sum_{l=1}^6 k_{li}z_{li}^2 + z_{1i}\tau_{1i} + z_{7i}(z_{6i} + v_i - \dot{\alpha}_{6i}) \quad (\text{IV.24})$$

Afin de remplir la synchronisation des articulations $q_{2,1}^i$ ainsi que d'assurer la passivité du système (IV.12), nous choisissons les expressions suivantes :

$$\begin{aligned} v_i &= -k_{7i}z_{7i} - z_{6i} + \dot{\alpha}_{6i} + \tau_{2i} \\ \tau_{2i} &= - \sum_{j=1}^N (q_{2,1}^i - q_{2,1}^j), \quad i = 1, \dots, N \end{aligned} \quad (\text{IV.25})$$

Avec τ_{2i} est une modélisation du protocole de consensus. En tenant compte de (IV.25), l'équation (IV.26) peut s'écrire sous la forme :

$$\dot{V}_{(l)i} = - \sum_{l=1}^6 k_{li}z_{li}^2 - k_{7i}z_{7i}^2 + z_{1i}\tau_{1i} + z_{7i}\tau_{2i} \quad (\text{IV.26})$$

Posons les variables Y_{1i} comme étant des sorties pour chaque sous système. La dynamique présentée dans l'équation (IV.12) peut se reformuler sous une structure de sous systèmes passifs interconnectés, comme il est présenté dans la forme suivante :

$$\begin{aligned}
 \mathcal{H}_{1i} : & \begin{cases} \dot{z}_{1i} = -k_{1i}z_{1i} + \tau_{1i} + z_{2i} \\ y_{1i} = z_{1i} \end{cases} \\
 \mathcal{H}_{2i} : & \begin{cases} \dot{z}_{2i} = -k_{2i}z_{2i} - z_{1i} + z_{3i} \\ y_{2i} = z_{2i} \end{cases} \\
 \mathcal{H}_{3i} : & \begin{cases} \dot{z}_{3i} = -k_{3i}z_{3i} - z_{2i} + g_{1i}z_{4i} \\ y_{3i} = z_{3i} \end{cases} \\
 \mathcal{H}_{4i} : & \begin{cases} \dot{z}_{4i} = -k_{4i}z_{4i} - g_{1i}z_{3i} + z_{5i} \\ y_{4i} = z_{4i} \end{cases} \\
 \mathcal{H}_{5i} : & \begin{cases} \dot{z}_{5i} = -k_{5i}z_{5i} - z_{4i} + g_{2i}z_{6i} \\ y_{5i} = z_{5i} \end{cases} \\
 \mathcal{H}_{6i} : & \begin{cases} \dot{z}_{6i} = -k_{6i}z_{6i} - g_{2i}z_{5i} + z_{7i} \\ y_{6i} = z_{6i} \end{cases} \\
 \mathcal{H}_{7i} : & \begin{cases} \dot{z}_{7i} = -k_{7i}z_{7i} + \tau_{2i} - z_{6i} \\ y_{7i} = z_{7i} \end{cases}
 \end{aligned} \tag{IV.27}$$

La structure (IV.27) montre bien une interconnexion en cascade des différents sous systèmes. Ce qui motive bien le recours à la technique de passivité afin d'analyser et prouver la stabilité asymptotique du système global. Donc, l'étude développée dans ce chapitre mène à un résultat principal qui se traduit dans le théorème suivant :

Théorème 4.1 : Considérons un groupe de robots sous actionnés dont la dynamique peut être représentée sous une forme triangulaire (IV.12). L'interconnexion des robots sous le couplage linéaire $L \otimes I_N$ (voir Figure IV.3) respecte une synchronisation asymptotique.

Preuve D'abord, notons $U_2 = [\tau_{21}, \tau_{22}, \dots, \tau_{2N}]^\top$, $\mathbf{q}_{2,1} = [q_{2,1}^1, q_{2,1}^2, \dots, q_{2,1}^N]^\top$, $\mathbf{y}_1 = -U_1$ et $\mathbf{y}_2 = -U_2$. Puis, considérons une fonction d'énergie $V_{ff}^{\mathcal{H}_1}$ pour le système $\mathcal{H}_1$ telle que :

$$V_{ff}^{\mathcal{H}_1} = \frac{1}{2} \mathbf{q}_{1,1}^\top (L \otimes I_N) \mathbf{q}_{1,1} \tag{IV.28}$$

La dérivée de cette fonction par rapport au temps produit l'expression suivante :

$$\begin{aligned}
 \dot{V}_{ff}^{\mathcal{H}_1} &= \left((L \otimes I_N) \mathbf{q}_{1,1} \right)^\top \dot{\mathbf{q}}_{1,1} \\
 &= (-U_1)^\top \dot{\mathbf{q}}_{1,1}
 \end{aligned} \tag{IV.29}$$

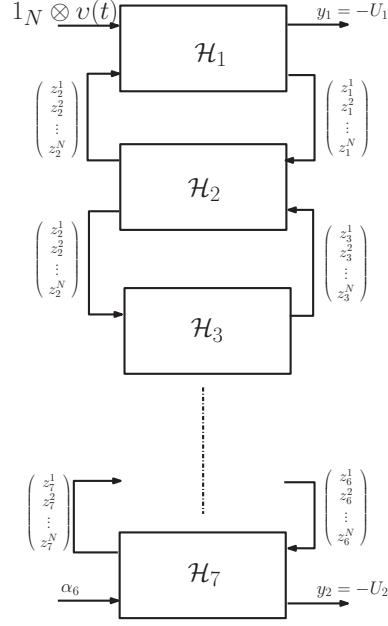


FIGURE IV.3 – Configuration des sous systèmes interconnectés en cascade .

Le résultat montre bien que le système $\mathcal{H}_1$ avec $\dot{\mathbf{q}}_{1,1}$ comme entrée et $-U_1$ comme sortie est un système passif. Tenant compte de $\dot{\mathbf{q}}_{1,1} = Z_1 + (1_N \otimes v(t))$ et sachant que $(L^\top \otimes I_N)(1_N \otimes v(t)) = 0$, alors l'équation (IV.29) peut s'écrire sous la forme

$$\dot{V}_{ff}^{\mathcal{H}_1} = (-U_1)^\top Z_1$$

Cette expression prouve que le système $\mathcal{H}_1$ avec Z_1 comme entrée et $-U_1$ comme sortie est aussi passif. Etudions maintenant le système $\mathcal{H}_2$ et considérons la fonction $V_{ff}^{\mathcal{H}_2}$ comme fonction d'énergie définie pour ce système comme suit :

$$V_{ff}^{\mathcal{H}_2} = \frac{1}{2} \mathbf{q}_{2,1}^\top (L \otimes I_N) \mathbf{q}_{2,1} \quad (\text{IV.30})$$

Comme le cas du système $\mathcal{H}_1$, nous pouvons montrer que $\mathcal{H}_2$ d'entrée $\dot{\mathbf{q}}_{2,1}$ et de sortie $-U_2$ est un système passif. Néanmoins, puisque $\dot{\mathbf{q}}_{2,1} = Z_7 + \alpha_6$ alors $\dot{V}_{ff}^{\mathcal{H}_2}$ prend la forme suivante :

$$\dot{V}_{ff}^{\mathcal{H}_2} = (-U_2)^\top Z_7 + (-U_2)^\top \alpha_6$$

Ce qui prouve que pour les entrées Z_7 et α_6 et la sortie $-U_2$, $\mathcal{H}_2$ est un système passif. Maintenant, afin de prouver que le retour d'information à partir de l'entrée $\underline{U} = [U_1^\top, U_2^\top]^\top$ jusqu'à la sortie $\underline{Z} = [Z_1^\top, Z_7^\top]^\top$ est passif, nous procédons par démontrer que les sous systèmes sont passifs à partir de $\mathcal{H}_1$ jusqu'à $\mathcal{H}_7$.

Considérons pour ce chemin de retour la fonction de stockage suivante :

$$V_{fb} = \sum_{i=1}^N \sum_{l=1}^7 V_{(l)i} \quad (\text{IV.31})$$

La dérivée par rapport au temps de cette fonction est donnée par :

$$\begin{aligned} \dot{V}_{fb} &= - \sum_{i=1}^N \sum_{l=1}^7 k_{li} z_{li}^2 + U_1^\top Z_1 + U_2^\top Z_2 \\ &= - \sum_{i=1}^N \sum_{l=1}^7 k_{li} z_{li}^2 + \underline{U}^\top \underline{Z} \end{aligned} \quad (\text{IV.32})$$

D'où le système étudié d'entrée/sortie $(\underline{U}/\underline{Z})$ est passif. Posons $e = [(1_N \otimes v(t))^\top, \alpha_6^\top]^\top et $y = [\mathbf{y}_1^\top, \mathbf{y}_2^\top]^\top$. Maintenant, prouvons que le système global d'entrée e et de sortie y est passif. On définit la fonction de stockage du système global en boucle fermée comme suit :$

$$V_{cl} = V_{ff}^{\mathcal{H}_1} + V_{ff}^{\mathcal{H}_2} + V_{bf} \quad (\text{IV.33})$$

Ce qui résulte une dérivée par rapport au temps de la forme :

$$\begin{aligned} \dot{V}_{cl} &= (-\underline{U}^\top) \underline{Z} + \underline{U}^\top \underline{Z} - \sum_{i=1}^N \sum_{l=1}^7 k_{li} z_{li}^2 + (-U_2^\top) \alpha_6 \\ &= - \sum_{i=1}^N \sum_{l=1}^7 k_{li} z_{li}^2 + (-U_2^\top) \alpha_6 + (-U_1^\top) (1_N \otimes v(t)) \\ &= - \sum_{i=1}^N \sum_{l=1}^7 k_{li} z_{li}^2 + \mathbf{y}^\top \mathbf{e} \end{aligned} \quad (\text{IV.34})$$

Encore une fois on s'est basé dans notre développement sur la relation $(L^\top \otimes I_N)(1_N \otimes v(t)) = 0$. Une équation qui découle de l'hypothèse qui dit que la matrice Laplacienne de notre graphe de communication est symétrique avec une somme nulle sur les lignes ainsi que sur les colonnes. Donc l'équation (IV.34) prouve que le système d'entrée-sortie $(\mathbf{e}, \mathbf{y})$ en boucle fermée est un système passif. L'expression de (IV.34) peut être majorée comme suit :

$$\begin{aligned} \dot{V}_{cl} &\leq - \sum_{i=1}^N \sum_{l=1}^7 k_{li} z_{li}^2 - \kappa U_2^\top U_2 - \frac{1}{4\kappa} \alpha_6^\top \alpha_6 \\ &\leq - \sum_{i=1}^N \sum_{l=1}^7 k_{li} z_{li}^2 - \kappa U_2^\top U_2 \end{aligned} \quad (\text{IV.35})$$

De ceci, nous pouvons conclure que le système est globalement stable. Considérons l'ensemble $E = \{z_{(l)i} \in \mathbb{R}, i = 1, \dots, N | \dot{V}_{cl} = 0\}$. L'ensemble E est caractérisé par toutes

les trajectoires telles que $\{\sum_{l=1}^7 k_{li} z_{li}^2, U_2 = 0, \forall i = 1, \dots, N, \forall i \neq j\}$. L'application du principe de Lasalle résulte $\lim_{t \rightarrow \infty} z_{(l)i} = 0, \forall l = 1, \dots, 7$ and $\lim_{t \rightarrow \infty} (L^\top \otimes I_N) \mathbf{q}_{2,1} = \mathbf{0}$. Donc, puisque $\lim_{t \rightarrow \infty} z_{1i} = \lim_{t \rightarrow \infty} z_{2i} = 0, \forall i = 1, \dots, N$. Ceci implique, en se basant sur l'équation (IV.27), que $\lim_{t \rightarrow \infty} (L^\top \otimes I_N) \mathbf{q}_{1,1} = \mathbf{0}$.

IV.5 Simulation

Afin de prouver l'efficacité de l'approche de synchronisation développée, nous proposons des simulations numériques sous la plate forme logicielle Matlab 7.0 pour un groupe de trois robots manipulateurs. Chaque robot en réseau est présenté par le modèle (IV.6) inspiré de l'équation dynamique (IV.5). Nous proposons les paramètres physiques suivants :

$M_3 = \begin{pmatrix} \delta_5 & 0 \\ 0 & \delta_6 \end{pmatrix}$ et

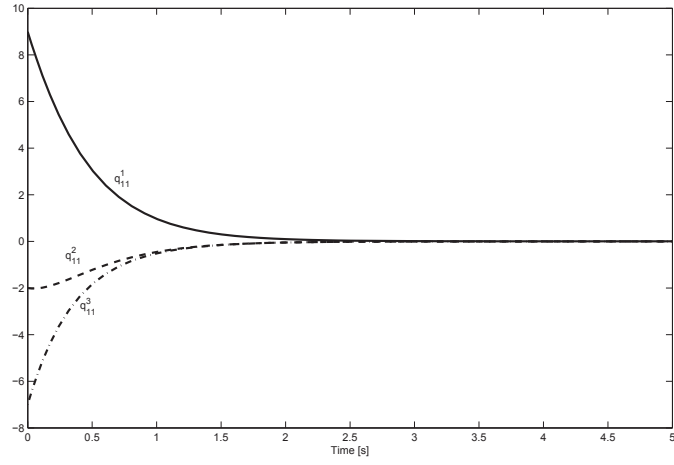
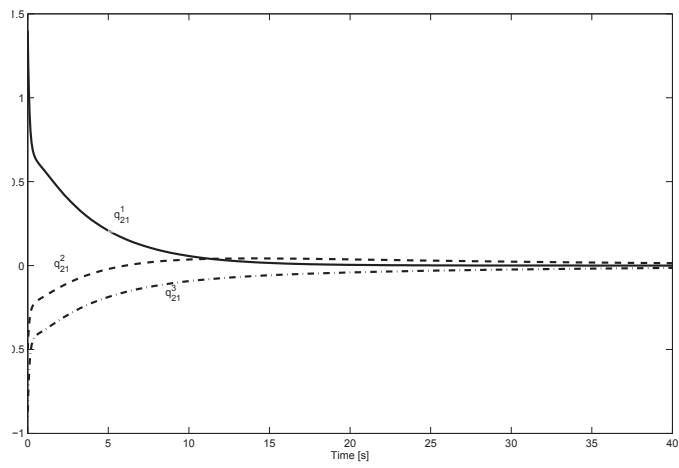
$$M_1(q_1) = \begin{pmatrix} \delta_1 + \delta_2 \cos^2 q_{1,2} & 0 \\ 0 & \delta_3 \end{pmatrix}, M_2 = \begin{pmatrix} 0 & \delta_4 \\ 0 & 0 \end{pmatrix}$$

$$C_1(q_1, \dot{q}_1) = \begin{pmatrix} -0.5\delta_2 \dot{q}_{1,2} \sin 2q_{1,2} & -0.5\delta_2 \dot{q}_{1,1} \sin 2q_{1,2} \\ 0.5\delta_2 \dot{q}_{1,1} \sin 2q_{1,2} & 0 \end{pmatrix}$$

et

$$\Delta_1 = \begin{pmatrix} 0 \\ \delta_7 \cos q_{1,2} \end{pmatrix}, \Delta_2 = 0$$

Où $\delta_k, k = 1, \dots, 7$ sont des constantes positives. Après une mise en forme triangulaire, chaque robot du groupe coopératif prendra le modèle dynamique (IV.11). Dans cette simulation numérique, nous considérons que les robots sont homogènes. Les paramètres du modèle sont choisis tels que : $[\delta_1, \delta_2, \delta_3, \delta_4, \delta_5, \delta_6, \delta_7] = [1.5, 0.5, 2, 0.2, 1, 1, 50] kgm^2$, $k_1 = 100 Nm^{-1}$ et $k_2 = 200 Nm^{-1}$. Les figures IV.4 et IV.5 présentent les trajectoires de position des deux articulations pour les trois robots. Il est bien remarquable que les trois courbes pour les deux articulations convergent vers une trajectoire commune. Initialement, les trois robots partent de positions différentes puis ils réagissent rapidement et leurs mouvements se synchronisent en suivant une trajectoire commune. On remarque que les articulations motorisées représentées dans la figure (IV.4) remplissent un consensus plus rapide que celles des articulations non-motorisées représentées dans la figure (IV.5). Ce qui nous paraît naturel. Ainsi, ces simulations prouvent l'efficacité de l'approche de synchronisation implémentée.

FIGURE IV.4 – Trajectoires des articulations q_{11}^i .FIGURE IV.5 – Trajectoires des articulations q_{21}^i .

IV.6 Conclusion

Dans ce chapitre, une nouvelle approche de contrôle de synchronisation et de suivi de trajectoire pour les robots sous actionnés est développée. La stratégie proposée se base sur un retour d'état statique non régulier capable de transformer le modèle dynamique d'un système mécanique sous actionné à une structure en chaîne. Une telle transformation qui prépare une structure favorable pour l'application de la procédure de backstepping et la technique de passivité afin de concevoir une loi de contrôle de synchronisation. L'application de la procédure de backstepping au système dynamique en chaîne produit une représentation dynamique des sous systèmes interconnectés en cascade. Cette forme d'interconnexion en cascade justifie bien l'utilisation de la technique de passivité afin d'analyser la stabilité asymptotique du système global. Pour plus de robustesse de la stratégie proposée pour les robots sous actionnés, le retard de l'information reste un défi à relever dans ce contexte de problèmes de communication.

Chapitre V

Mise au point d'une plateforme expérimentale

V.1 Introduction

Ce chapitre décrit la mise en place de la plate forme expérimentale que nous avons développée afin de valider nos propositions de stratégies de contrôle de synchronisation coopérative. Nous avons réalisé un groupe de trois robots manipulateurs identiques. La conception mécanique du robot a été développée sur la plate forme logicielle “Cinema 4D”, tandis que les simulations et les tests des différents modules électroniques ont été réalisés sur le logiciel électronique de CAO “Proteus”. Le travail a surtout concerné la conception et le développement d'un réseau filaire ou sans fil afin d'assurer une communication explicite entre les robots en coopération. Pour des raisons financières, nous avons retenu l'utilisation d'un réseau filaire. Le protocole de communication développé est basé sur le principe de “Broad Cast”. Ce protocole assure un bon partage de l'information comme il offre une flexibilité d'échange de données avec n'importe quel autre type de réseaux (Bus, étoile...).

V.2 Conception mécanique

Le système mécanique conçu est composé essentiellement de deux sous systèmes. Un sous système responsable de la rotation de chaque articulation et un autre pour assurer la manipulation de la pince. La rotation d'une articulation utilise deux types de transformation de mouvement :

A. Une transformation du mouvement de la rotation du moteur en une translation grâce à un système vis-écrou. Le système vis-écrou (liaison hélicoïdale) possède un degré de liberté sous la forme de deux mouvements élémentaires liés.

B. Une transformation de translation d'écrou en une rotation, grâce à un système Poulie/Courroie.

L'ensemble mécanique réalisé est visible à la figure (V.4). La conception du système globale est présentée dans la figure (V.6).

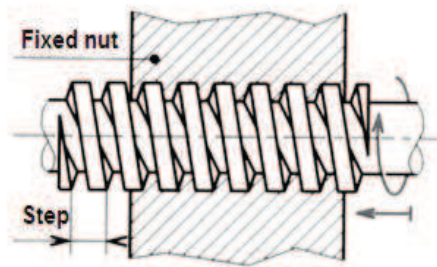


FIGURE V.1 – Système vis-écrou



FIGURE V.2 – Système poulies-courroie



FIGURE V.3 – Roue dentée

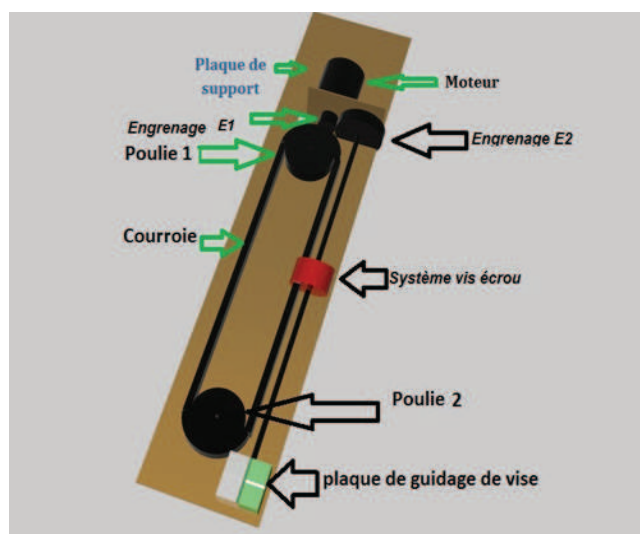


FIGURE V.4 – Plaque de guidage de la vis



FIGURE V.5 – Pince



FIGURE V.6 – Robot manipulateur

V.3 Conception du système électronique

Dans un système mécatronique, l'interfaçage des multiples capteurs et actionneurs est un point critique. Dans le but de simplifier cet interfaçage, on a développé un module du contrôle du moteur à courant continu (DC) (FIG. V.7).

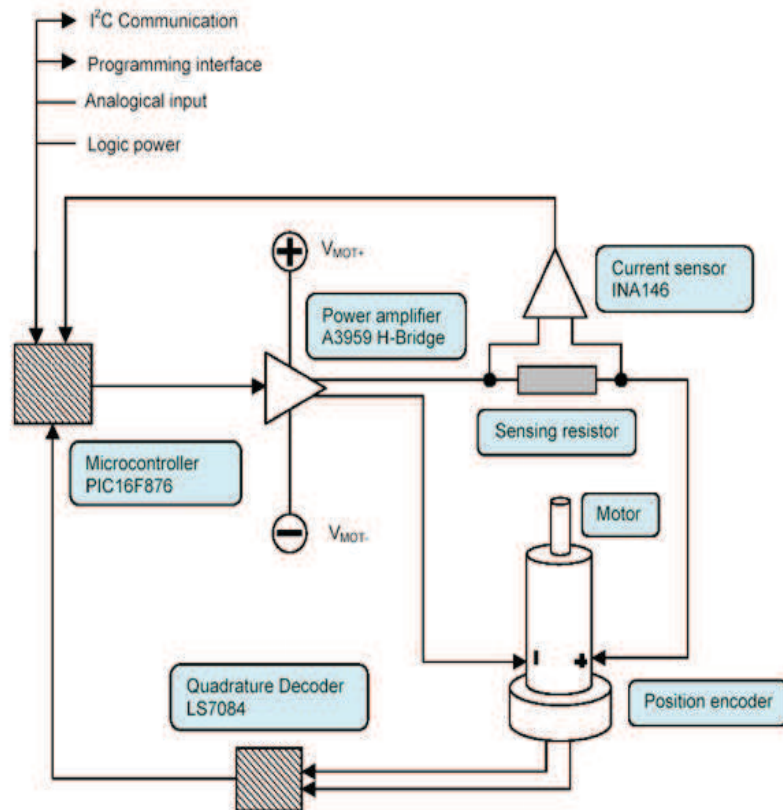


FIGURE V.7 – Module électronique

Le module développé intègre diverses fonctions capables d'alimenter le moteur, de varier sa vitesse et de commander son sens de rotation. Ce module est équipé aussi des capteurs implantés dans l'axe de rotation de chaque articulation afin d'assurer un retour d'information. Le module se base essentiellement sur un microcontrôleur afin d'assurer le contrôle de contrôle du moteur et plusieurs autres sous-systèmes annexes pour les mesures, la communication et toute adaptation et conversion des signaux (CAN ou CNA).

L'annexe 1 décrit plus en détails comment nous avons conçu et réalisé cette partie électronique.

V.4 Conception logicielle et protocole de communication

Dans le système embarqué, le langage C est utilisé comme langage de programmation du micro-contrôleur. Dans le code C embarqué, on a implémenté tout ce qui est gestion des ports, interruptions internes et externes, conversions Analogiques/Numériques et surtout traitement des trames et gestion de réseau.

L'annexe 2 décrit plus en détails la conception logicielle et le protocole de communication.

V.5 Résultats d'expérimentation

Afin d'assurer le contrôle/ commande du groupe de robots coopératifs, une Interface Homme/Machine (IHM) est développée sous l'environnement Matlab (V.8). Les IHM (Interfaces Homme Machine), sont appelées GUI (Graphical User Interfaces) dans Matlab. Elles permettent à l'utilisateur, grâce à des objets graphiques (boutons, menus, cases à cocher...) d'interagir avec un programme informatique. Dans ce qui suit, un aperçu rapide sur l'interface Homme-Machine conçue est présenté. Il s'agit d'une interface de communication PC-Robot contenant divers champs assurant la commande et la supervision. Cette interface offre la possibilité de configuration des ports de communication et du choix du nombre de robots en coopération. En termes de commande, l'interface présente des champs assurant le choix de la trajectoire, le lancement de la connexion et le déclenchement de l'application. La visualisation des résultats de la manipulation est assurée par des champs qui offrent la supervision de la trajectoire réalisée ainsi que la vitesse et l'erreur par rapport à la trajectoire désirée. Les figures 9, 10, 11 montrent quelques résultats expérimentaux.

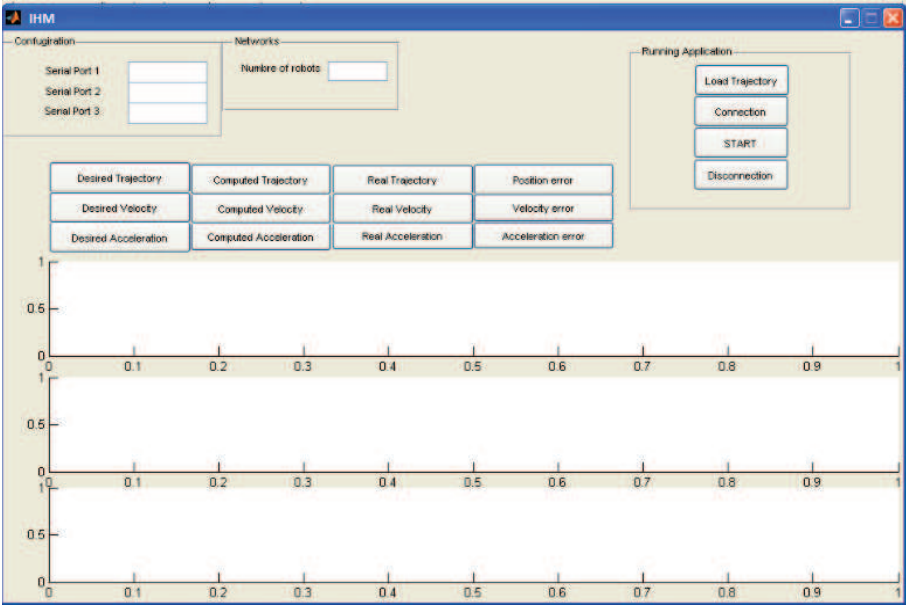


FIGURE V.8 – Interface Homme/Machine

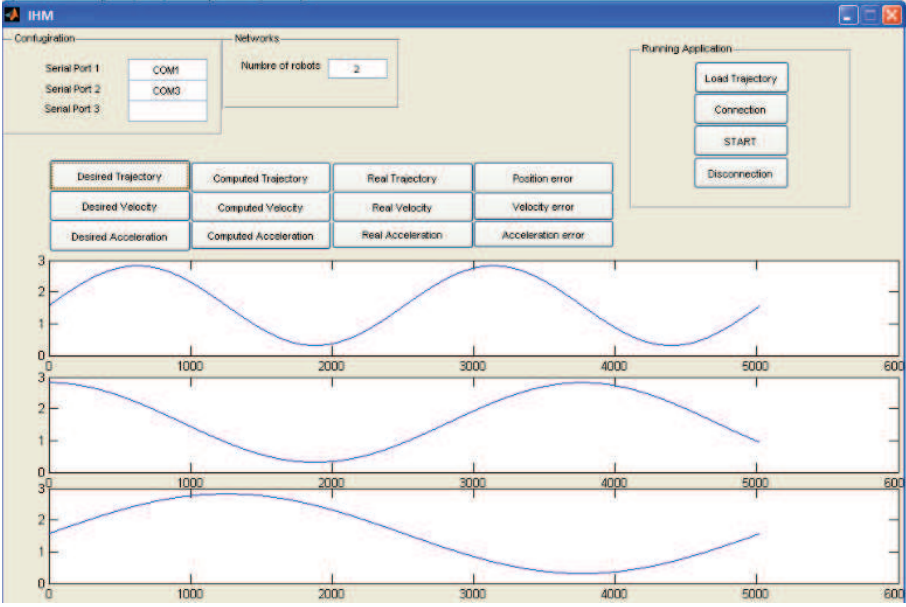


FIGURE V.9 – Trajectoire désirée

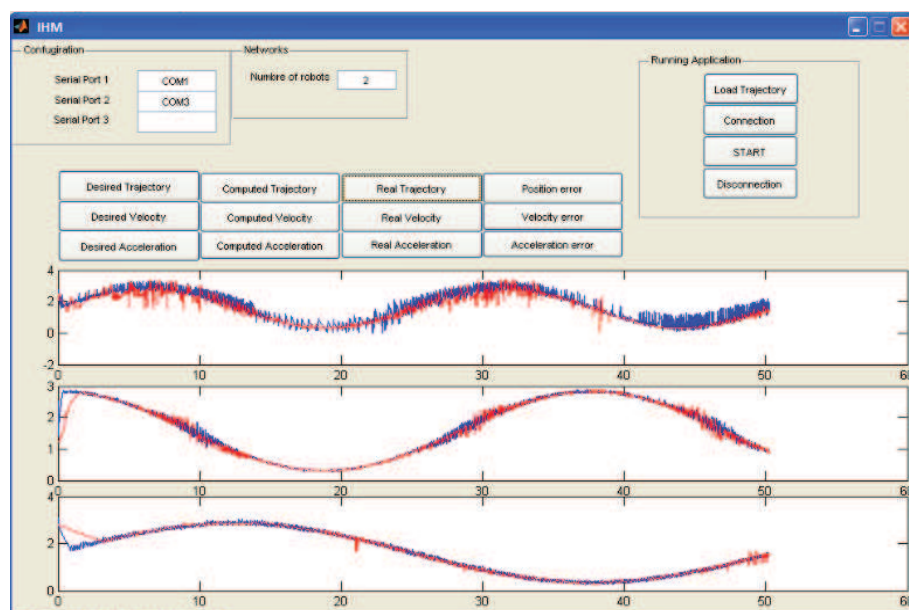


FIGURE V.10 – Trajectoire réalisée

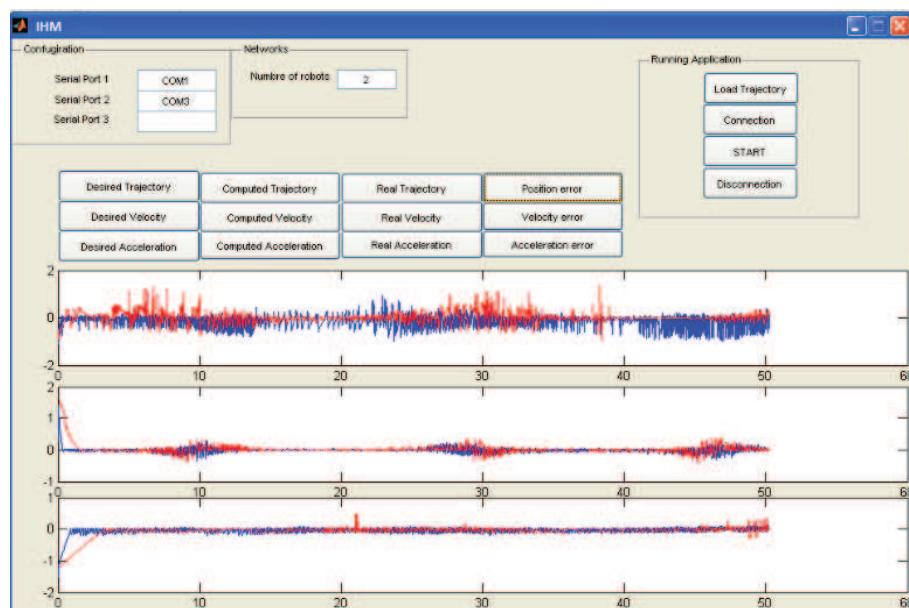


FIGURE V.11 – Erreur de trajectoire

V.6 Conclusion

Ce chapitre présente la conception et la réalisation de la plate-forme d'expérimentation pour la robotique coopérative que nous avons conçue et réalisée. Dans une première étape, la conception mécanique du robot manipulateur a été réalisée. Ensuite, nos efforts se sont orientés vers l'intégration mécatronique d'un groupe de trois robots. Dans une autre étape, nous avons développé les modules de contrôle-commande pour chaque robot manipulateur. La partie commande est basée essentiellement sur un microcontrôleur de la famille PIC 18F embarqué sur chaque robot. L'unité de commande de chaque robot est capable de gérer les flux de communication, d'envoyer les ordres d'actions, de recevoir les mesures et de garantir les retours d'information. Une interface homme-machine a également été développée dans le but d'assurer une commande facile et un contrôle efficace. Cette interface permet à l'utilisateur de choisir sa trajectoire désirée ainsi que le nombre de robots en coopération. Elle lui permet aussi d'établir la connexion et de lancer l'application. Elle joue le rôle de supervision et permet l'affichage de l'état du système et des informations provenant des différents capteurs. Les résultats obtenus montrent l'efficacité du système conçu sur les points de vue matériel et logiciel.

Annexe 1

Le plus important de ces sous-systèmes est le pont en H (FIG. 1) qui sert à alimenter le moteur et fournir une puissance capable de gérer la vitesse et le sens de rotation. Néanmoins, les circuits intégrés de systèmes pont en H trouvés sur le marché ne peuvent

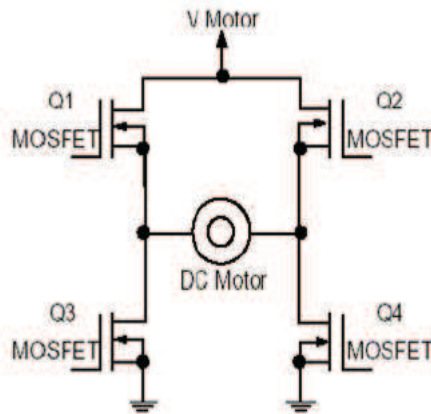


FIGURE 1 – Principe du pont en H

pas supporter un courant dépassant 5A comme il est indiqué dans le tableau ci-dessous (2). L'idée était de réaliser une carte de puissance pont en H qui accepte jusqu'au 7A (figure 9). Le fonctionnement de base est relativement simple, il permet d'alimenter le moteur dans un sens ou dans l'autre, rendant ainsi possible son changement de sens de rotation. En effet, si Q2 et Q3 conduisent, la borne moins du moteur sera connectée à la masse et la borne plus à la tension d'alimentation faisant ainsi tourner le moteur dans son sens de rotation normal. Si maintenant ce sont Q1 et Q4 qui conduisent, la borne moins du moteur sera connectée à la tension d'alimentation et la borne plus à la masse faisant alors tourner le moteur en sens inverse. Les diodes D1 à D4 sont appelées diodes de roue libre et sont nécessaires au bon fonctionnement du pont en H. Lorsque le moteur est alimenté (c.à.d. Q2 et Q3 conduisent ou Q1 et Q4 conduisent) ces diodes n'ont aucune effet. Elles servent uniquement lorsque le moteur passe d'un état alimenté à un état non alimenté. En effet, dans ce cas, les transistors vont arrêter brusquement de conduire

Numéro	Fabriqueur	Courant (A)	Tension d'alimentation (V)	Dimensions (mm)	Images
A3953	Allegro	1,3	5-50	7,5x10x2,5	
A3958	Allegro	2	20-50	7,5x15x2,5	
A3959	Allegro	3	9,5-50	7,5x15x2,5	
SI9986	Vishay	1	3,8-13,2	4x5x1,5	
LMD18200	National Semiconductor	3	12-60	13x31x5	
MPC17510	Motorola	1	2-15	6,5x8x1,5	
MC33887	Motorola	5	5-40	10x16x3,5	

FIGURE 2 – Circuits intégrés ponts en H

le courant et celui-ci pourra alors passer par les diodes de roue libre. Cela permet de limiter les pics de tension inverse aux bornes du moteur et ainsi d'éviter la destruction des composants électroniques. On remarque que les transistors Q1 et Q2 ou Q3 et Q4 ne doivent jamais conduire simultanément sous peine de court-circuit franc et de destruction des transistors. Le pont en H peut être commandé de plusieurs façons différentes suivant la manière dont les commutations des transistors sont enchainées. Dans le module développé, on a recours au mode "Locked Anti-phase". Dans ce mode, aucune distinction n'est faite quant à la direction de rotation du moteur, les deux directions étant contrôlées directement par le signal PWM. Ce mode est aussi divisé en deux phases. Dans la première phase, un des transistors de la partie haute du pont et le transistor opposé de la partie basse conduisent et alimentent le moteur dans un certain sens. Dans la deuxième phase, ce sont les deux autres transistors qui conduisent et qui alimentent le moteur dans le sens inverse. Le moteur sera donc connecté en permanence à une source de faible impédance. Il est à remarquer que pour mettre le moteur à l'arrêt, un rapport cyclique de 50% doit être utilisé. Afin de surmonter les problèmes de la différence d'alimentation entre la carte de commande et la carte de puissance et par suite les méfaits technologiques de retour du courant, on a recours à un dispositif d'isolation galvanique. Pour faire varier la vitesse d'un moteur on peut faire varier la tension d'alimentation à ses bornes, mais dans ce cas une partie importante de l'énergie est consommée par le dispositif d'alimentation. On préfère l'alimenter de façon discontinue avec un hacheur et faire ainsi varier la tension moyenne à ses bornes. On parle alors de Modulation par Largeur d'Impulsions (MLI), ou Pulse Width Modulation (PWM).

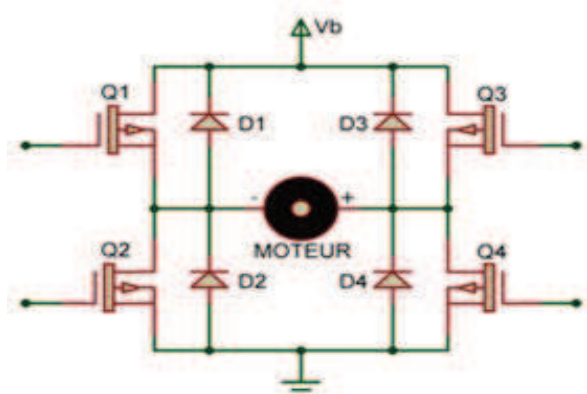


FIGURE 3 – Module de pont en H

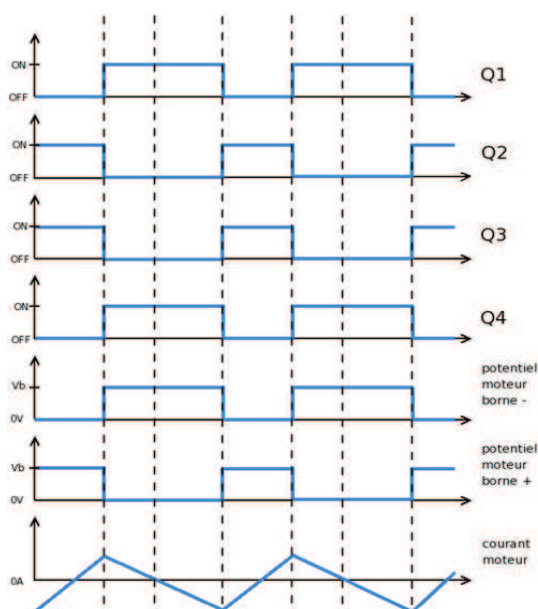


FIGURE 4 – Diagramme temporel Locked Anti-phase

La partie commande est basée sur un PIC18F452 de la famille PIC18. La famille PIC18 a un jeu d'instruction plus complet puisqu'elle comprend 75 instructions. Cette palette d'instructions étendue lui permet de faire fonctionner du code C compilé de manière nettement plus efficace que d'autres familles. Sur les dernières versions (sous-famille "K"), on peut les utiliser avec un quartz fonctionnant jusqu'à 64 MHz. Cette famille propose une multitude de dérivés intégrant l'USB, l'ETHERNET, le bus CAN, des canaux de MLI dédiés au contrôle moteur.

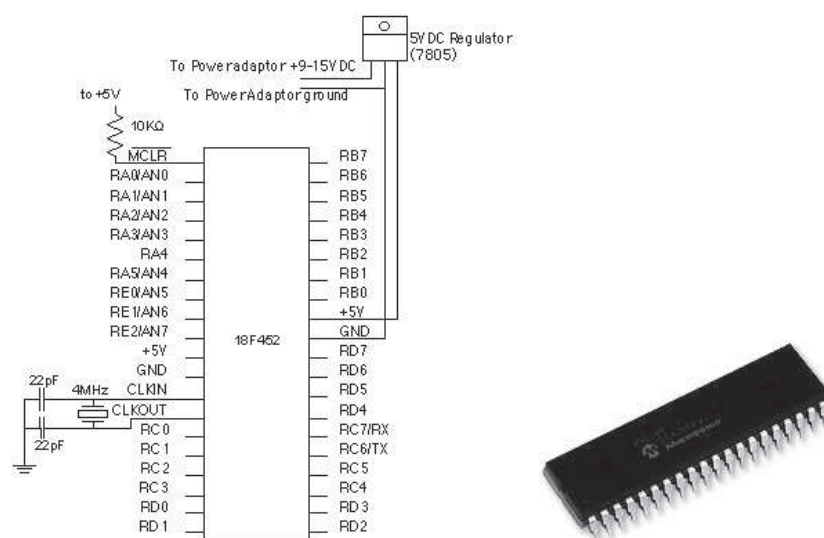


FIGURE 5 – Microcontrôleur : PIC18F452

Sur la plate forme développée, des potentiomètres sont implantés sur chaque axe des différentes articulations. Ces potentiomètres servent à avoir une mesure absolue des angles des articulations. Ainsi, avec l'aide de ces derniers, une initialisation des encodeurs est possible et il n'y a pas besoin, lors de la mise en route du bras, de le bouger pour aller à une position de référence. Pour la lecture du potentiomètre par le microcontrôleur, un code, déjà implémenté, permet d'utiliser le potentiomètre comme encodeur. Ce fichier contient également des fonctions pour déterminer la vitesse et le sens de rotation de l'articulation. Afin d'initialiser les encodeurs avec la position des potentiomètres, il faut transformer la position 8 bits en position 32 bits. La position de l'encodeur est donnée par un compteur 32 bits qui est incrémenté à chaque pas de l'encodeur. La position qui est utilisée pour le contrôle est un nombre 8 bits qui est tiré du compteur 32 bits en fonction de la précision désirée. Par exemple, dans le cas d'un moteur simple avec un encodeur, comme la position utilisée est un nombre 8 bits signé, on aura une plage de mesure de -128 à 127. A supposer que cette plage couvre un tour complet de l'axe du moteur, on aura 256 incréments par tour. Si maintenant on fixe un réducteur à la sortie, la plage de mesure de l'encodeur couvrira un angle moins grand après l'étage de réduction. On aura donc une résolution plus grande, mais une plage de mesure réduite.



FIGURE 6 – Plate forme d'expérimentation

Annexe 2

Dans ce qui suit, nous présentons un organigramme descriptif du code implémenté (FIG. 1). Initialement, le robot est dans un statut déconnecté. Chaque caractère reçu sur le port de connexion est ajouté dans un buffer jusqu'à la réception d'un caractère d'induction de trame. Le microcontrôleur est chargé de vérifier la taille de la trame reçue. Si le test est validé, il va traiter le contenu de la trame, si non il supprime le buffer. Le traitement de la trame se fait selon l'état de robot. Trois étapes de traitement sont implémentées :

- Si le robot est en état de déconnexion et s'il reçoit une trame de connexion alors il va charger l'adresse de cette trame dans son buffer d'adresse. Ensuite, il va faire la lecture du capteur avant de construire sa trame de position. Puis, il va actualiser son statut en état "en cours de connexion" et envoyer la trame.
- Si le robot est en cours de connexion et il reçoit une trame d'acquiescement de validation d'adresse, il va alors se préparer vers le changement de statut de "déconnecté" vers "connecté".
- Si à la réception de la trame, le robot était connecté, il va vérifier si cette trame est dédiée pour lui ou non. Si le test est positif, il ouvre alors ses ports et traite la trame selon son exigence (action, déconnexion...).

Afin d'avoir un partage efficace de l'information, la stratégie développée se base sur un réseau "Broad Cast". Cette technique possède une grande flexibilité d'échange d'information avec n'importe quel type de réseau (bus, étoile...). La trame diffusée est de 23 octets contenant les informations nécessaires et suffisantes pour la gestion de la communication au sein de réseau. Dans le réseau conçu, nous pouvons parler de deux types de trame :

- Les trames de configuration : sont des trames dédiées à l'organisation et à la disposition des robots au sein du réseau dans le but de gérer la transmission de données.

- Le protocole conçu assure la distinction de la nature de la trame envoyée et reçue. En effet une indication sur le dernier octet de la trame distingue bien le type de la trame. Le tableau ci- dessous explique bien les différents types de trames utilisées. Pour tout genre

TABLE 1 – Les types de trames conçues

Indice	Description (suivant le sens de la transmission)
T	Désigne une trame de connexion ou la réponse à cette trame.
G	Message d'acquittement pour confirmer l'adresse attribuée.
P	C'est une trame qui décrit la position désirée ou la position capturées.
E	Une trame de déconnexion.

de trame, les trois premiers octets sont réservés pour l'adressage du robot. Une trame de configuration peut être effectivement une trame de connexion ou une trame d'acquittement de validation ou une trame de déconnexion. De ce fait, cette trame peut avoir les trois formes suivantes (voir Tables 2.3 et 4), selon la nature de sa fonction réelle.

TABLE 2 – Trame de connexion

[illegible]

TABLE 3 – Trame d’acquiescement de validation

[illegible]

TABLE 4 – Trame de déconnexion

[illegible]

Alors, lors de fonctionnement du réseau, le protocole de communication conçu exige le passage par les trois étapes suivantes :

étape de connexion : c'est la première étape nécessaire pour lancer l'application. Cette étape commence par l'envoi d'une trame de 23 octet contenant une adresse réseau non allouée. La trame de connexion se distingue par la présence de l'indice 'T' comme contenu de son dernier octet. Lors de la réception de la trame, chaque robot non connecté change son adresse et envoie immédiatement les positions de chacune de ses articulations. A cette phase le robot doit changer son statut et migre de "non connecté" vers un statut "en cours de connexion". La trame envoyée de la part du robot sera reçu par l'unité de commande centrale qui va traiter les informations. Après le traitement de ces données, les positions des différentes articulations du robot seront enregistrées et une trame d'acquittement d'indice 'G' est envoyée à la même adresse.

étape d'action : l'unité de commande traite les algorithmes de synchronisation coopérative et envoie une trame d'action sur le support de réseau. Le robot concerné ouvre ses ports pour traiter cette trame de données. Par suite, il va extraire les actions à réaliser et passe directement à l'exécution. L'action se termine par un retour d'information vers l'unité centrale sur les positions actuels du robot à travers une trame d'indice 'P'.

étape de déconnexion : afin de libérer le support et bien gérer les flux de communication, le protocole conçu utilise une trame de déconnexion. Il s'agit d'une trame de donnée d'indice 'E' envoyée par l'unité de commande centrale. Le robot qui reçoit ce message se déconnecte et par suite il aura un changement de statut de "connecté" vers "non connecté". L'agent qui vient d'être déconnecté reste en écoute du réseau en attendant une autre tentative de connexion.

Les organigrammes explicatifs de ces trois étapes sont présentés dans les figures ci-dessous (voir 2, 3, 4) :

Cependant une trame de transmission de données porte 23 octets d'informations. Cette trame se distingue selon le sens de transmission. Les deux tableaux suivants présentent une description de la disposition de chaque trame.

TABLE 5 – Description des champs de trames Robots-PC

Champs (octets)	Descriptions
1, 2, 3	Adresse de robot dans le réseau
4, 5 , 6, 7 et 8	Position de la première articulation
9	Octet de séparation entre les données
10,11,12,13 et 14	Position de la deuxième articulation
15	Octet de séparation
16, 17, 18,19 et 20	Position de la troisième articulation
21	Octet de séparation (espacement)
22	Information sur l'état de la pince

TABLE 6 – Description des champs de trames PC-Robots

Champs(octets)	Fonctionnement
1, 2, 3	Adresse de robot dans le réseau
4	L'activation et le sens de rotation de l'articulation 1
5	L'activation et le sens de rotation de l'articulation 2
6	L'activation et le sens de rotation de l'articulation 3
7	L'ouverture ou la fermeture de la pince
8 et 9	Durée de l'état haut de l'articulation 1
10 et 11	Durée de l'état haut de l'articulation2
12 et 13	Durée de l'état haut de l'articulation3
14 et 15	Durée de l'état bas de l'articulation1
16 et 17	Durée de l'état bas de l'articulation2
18 et 19	Durée de l'état bas de l'articulation3
20	Nombre de réputation de la boucle de l'artculation1
21	Nombre de réputation de la boucle de l'artculation2
22	Nombre de réputation de la boucle de l'artculation3
23	Indexation de la trame

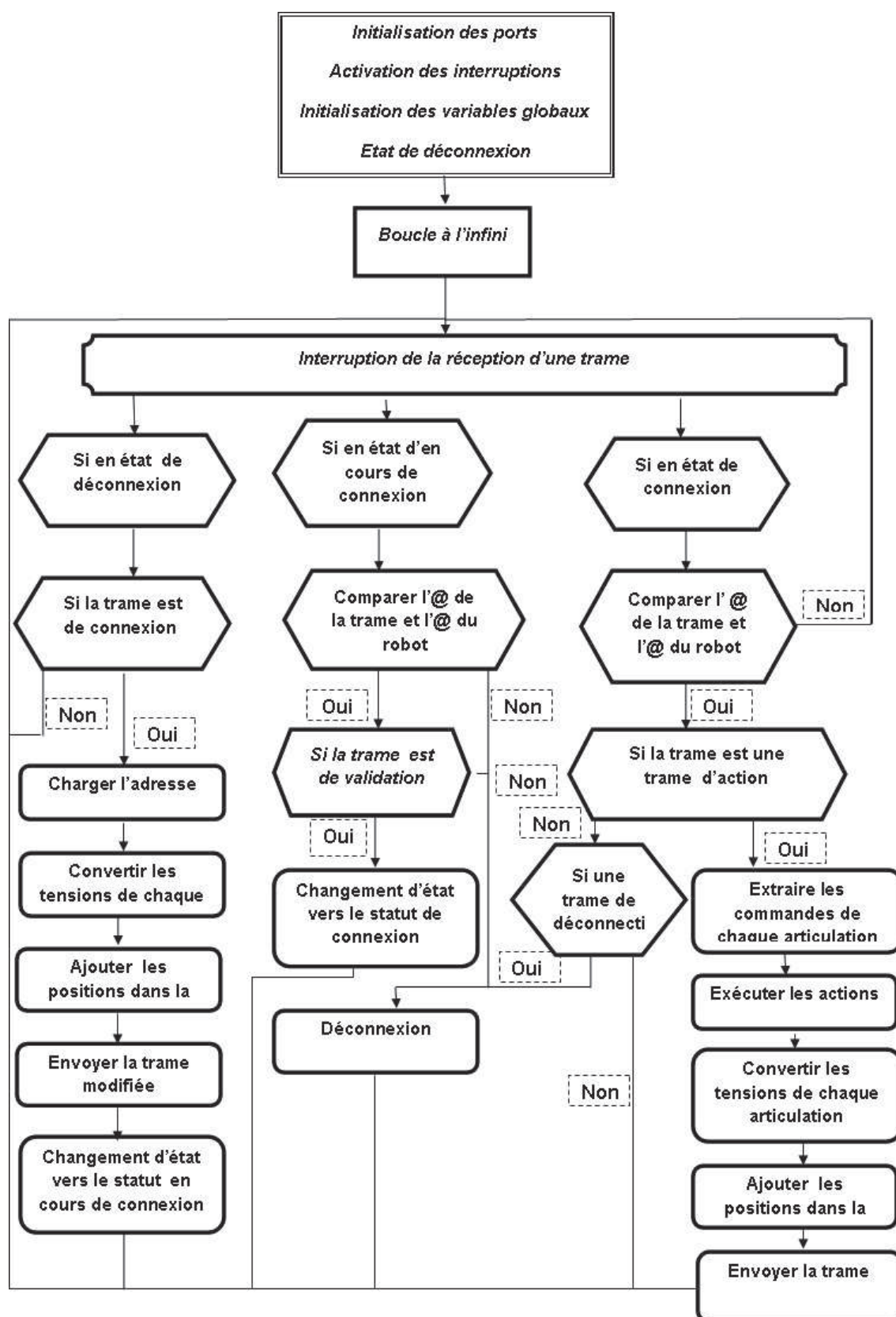


FIGURE 1 – Organigramme du code embarqué

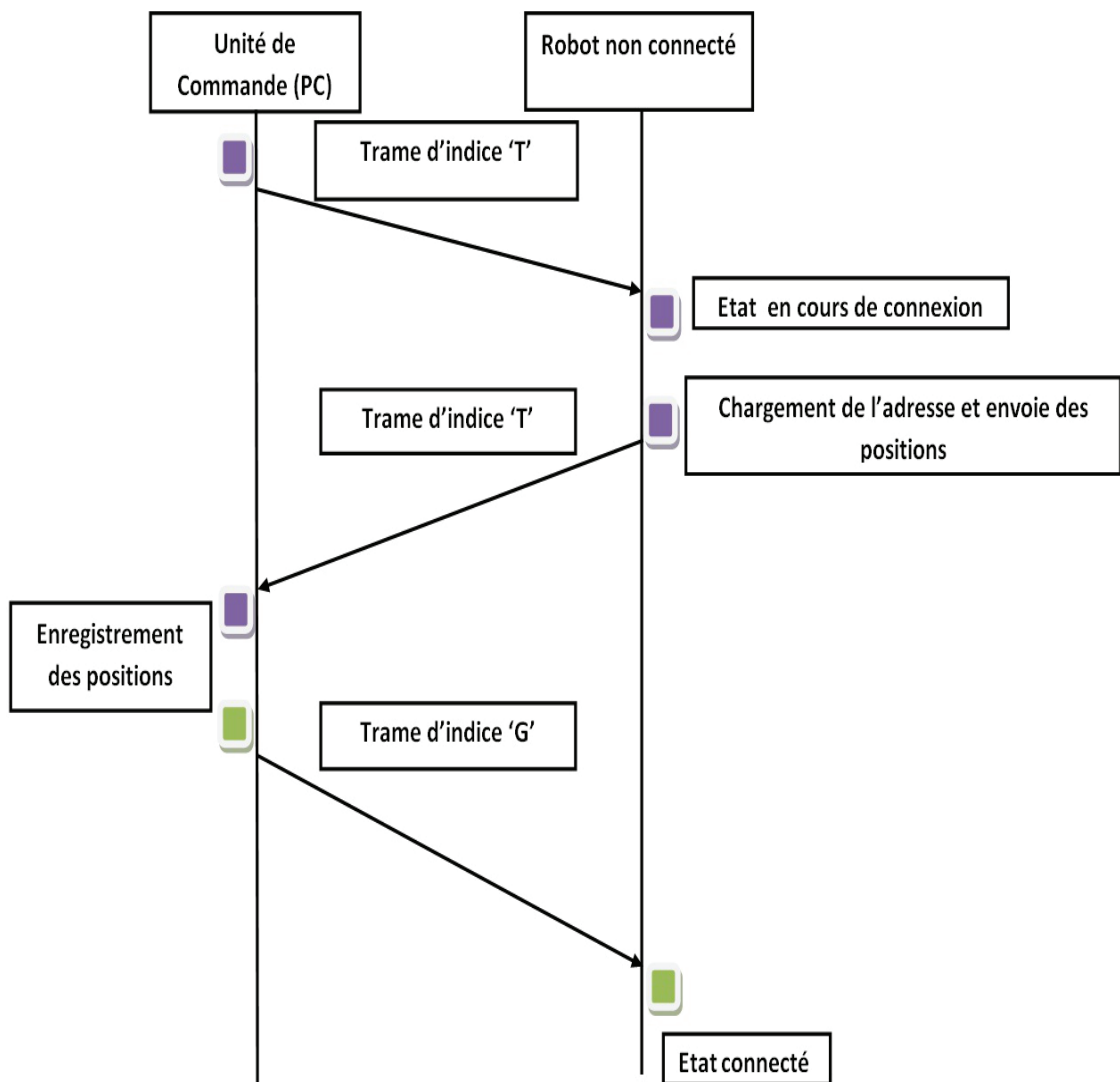


FIGURE 2 – Organigramme de connexion

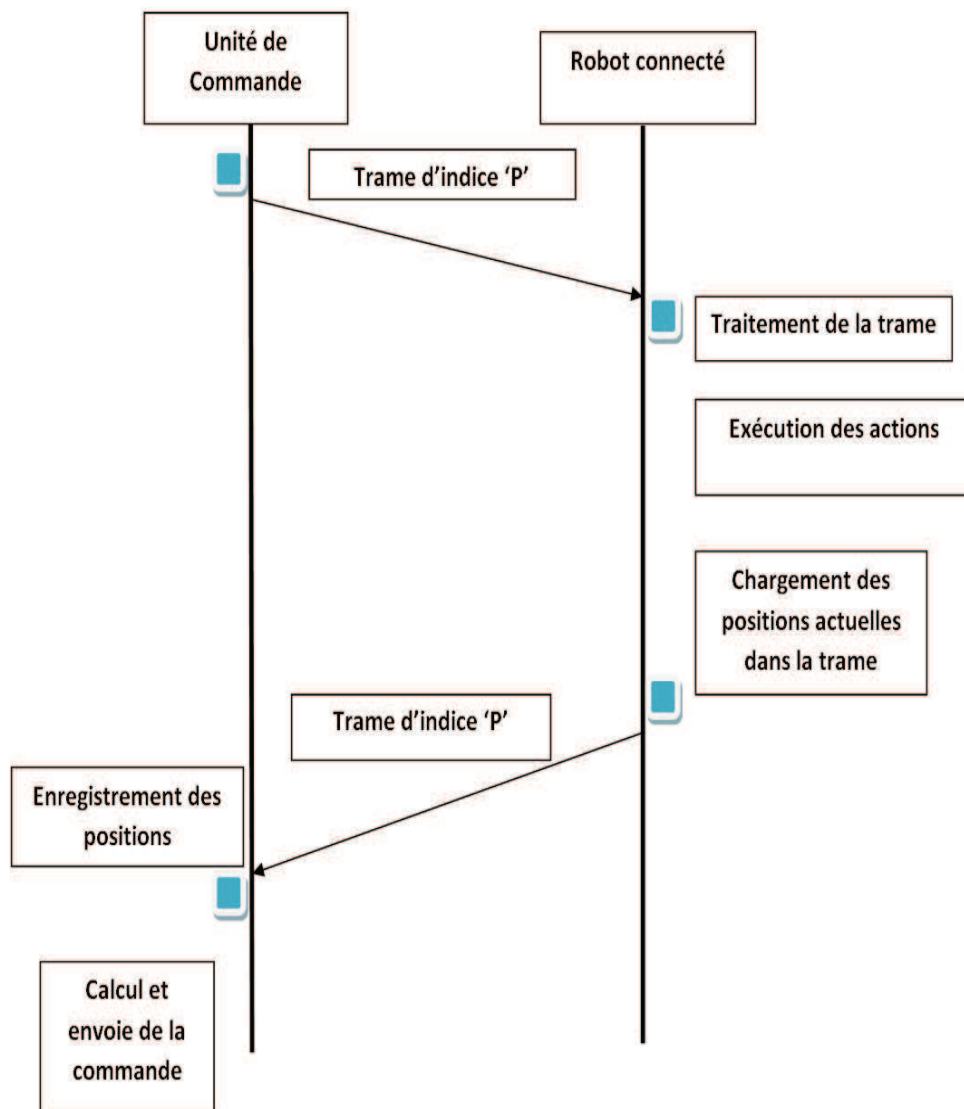


FIGURE 3 – Organigramme d'action

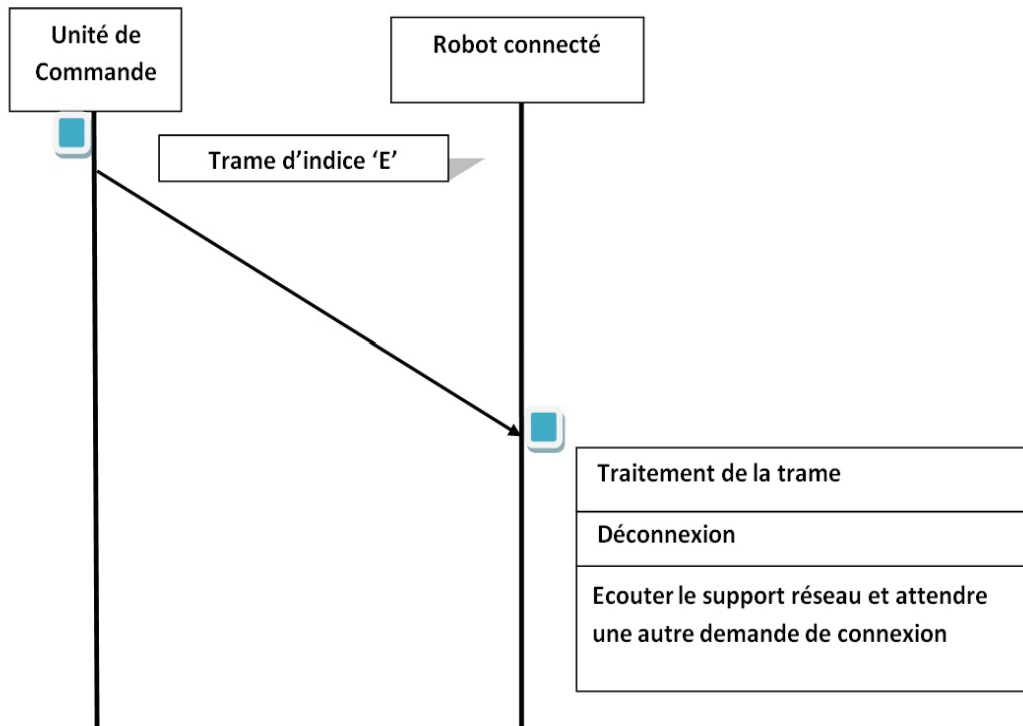


FIGURE 4 – Organigramme de déconnexion

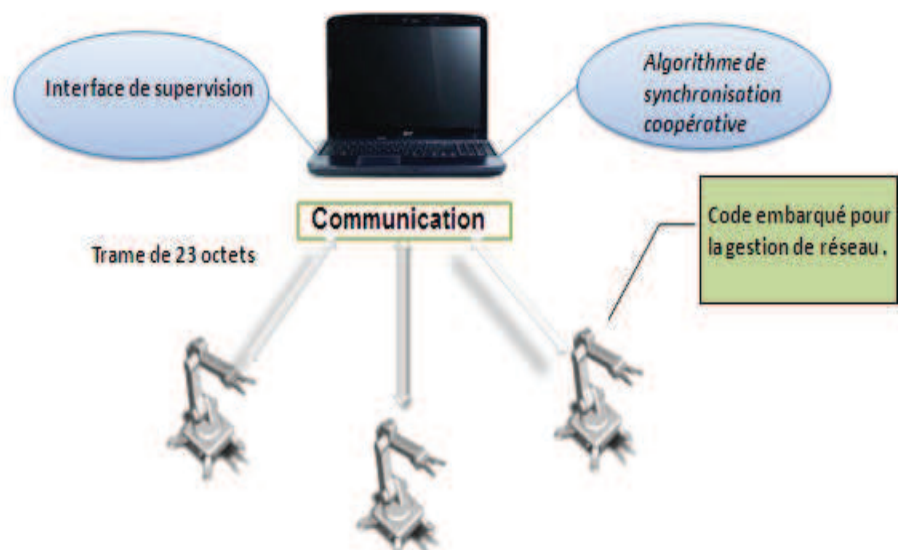


FIGURE 5 – Réseau d'expérimentation

Conclusions

Dans cette thèse nous avons développé les lois de coordination pour des systèmes de Lagrange.

En premier lieu nous avons proposé une stratégie complètement décentralisée pour la commande d'un groupe de robots, appelé réseau, qui synchronisent leurs mouvements en suivant une trajectoire désirée. Le groupe coopératif étudié dans cette approche n'impose pas la présence d'un leader. La trajectoire commune est implémentée dans chaque robot du groupe (ou agent) sachant que les robots n'ont pas les mêmes points de départ. En se basant sur la technique de cross-coupling chaque agent échange ses données avec les autres agents. Les informations recueillies par chaque agent, implicitement et explicitement, le conduisent à synchroniser son mouvement avec ses coéquipiers en suivant une trajectoire désirée. La technique de couplage et par suite la coordination entre ces types de robots sont très importantes, surtout dans les cas où il y a des perturbations externes ou dans les cas où les systèmes ne sont pas homogènes.

Dans une deuxième étape, nous nous sommes proposés d'étendre cette stratégie pour faire face à l'incertitude paramétrique des robots souvent rencontrée dans la pratique. L'approche adaptative a alors été développée afin d'avoir une loi de contrôle capable de synchroniser un groupe de robots coopératifs en présence d'incertitudes paramétriques. La stratégie de contrôle de synchronisation coopérative basée sur la technique de cross-coupling est étendue afin de tenir en compte les retards de transmission au niveau des canaux de communication. Ces retards sont fréquemment rencontrés dans les applications de communication. Une loi de contrôle de synchronisation et de suivi de trajectoire des systèmes à retard a été proposée.

Dans une autre étape de cette thèse, une deuxième architecture a été proposée. Cette architecture se concentre sur les réseaux à leader. La topologie considérée est hybride : bien qu'elle soit sous une structure de leader-suiveur elle est décentralisée et n'exige pas à

chaque agent d'avoir un lien de communication direct avec le leader. Dans cette approche, nous avons eu recours à la technique du backstepping afin de définir la loi de contrôle.

Une extension adaptative a été développée pour traiter les cas d'incertitude paramétrique. L'adaptation a été faite par des réseaux de neurones à fonctions de bases radiales. Les retards de communication restant toujours un grand défi à résoudre dans les réseaux industriels, nous avons pris en considération les délais de réception des données dans la stratégie de commande.

Un troisième volet porte sur la synchronisation coopérative des robots sous actionnés. En particulier, nous avons considéré un réseau d'une classe de robots polaires avec deux articulations flexibles. En se basant sur la notion du système en chaîne, le concept de la passivité et la technique de backstepping, une nouvelle approche de coordination des systèmes sous actionnés a été développée. L'idée de base dans la conception de cette nouvelle approche a été de transformer le système sous actionné en un système en chaîne, ce qui facilite la tâche de contrôle et par suite l'application des techniques de commande. La topologie proposée est basée sur une structure de communication modélisée par un graphe orienté fortement connecté.

Afin de valider nos travaux sur les algorithmes de synchronisation des robots coopératifs, nous avons souhaité mener des expérimentations. Ne disposant pas d'un groupe de robots industriels comme nous aurions pu le souhaiter, nous avons développé une plateforme expérimentale pour la robotique coopérative. Ce travail a exigé des développements mécaniques, électroniques et informatiques. La plate forme que nous avons conçue et réalisée est un groupe de trois robots poly-articulés. La conception mécanique a été menée sur le logiciel "Cinéma 4 D" et les simulations électroniques ont été effectuées sur la plate forme logicielle "Proteus".

Les expérimentations que nous avons réalisées montrent l'efficacité des algorithmes implantés. Nous avons toutefois besoin d'améliorer la qualité mécatronique globale de notre plate forme expérimentale afin de mettre mieux en valeur les intérêts de nos approches pour la commande synchronisée. Nous espérons par ailleurs pouvoir travailler avec un industriel ou un laboratoire qui mettrait à disposition un groupe de robots industriels du commerce.

A l'issue de ce travail de thèse, plusieurs problèmes demeurent ouverts et d'autres investigations sont à développer. Nous présentons ici ce qui nous semble être le cadre de travaux où des avancées importantes sont envisageables.

L'extension de la synchronisation coopérative pour les systèmes sous actionnés est un défi à relever. En effet, les algorithmes de commande synchronisée doivent prendre en considération les incertitudes paramétriques des systèmes sous actionnés et par suite conduire à une version adaptative du contrôle. Par ailleurs, le retard de transmission d'information au niveau des canaux de communication reste aussi un problème clé à résoudre.

Le contrôle de la synchronisation de robots complètement actionnés ou sous actionnés en coopération a été étudié dans cette thèse. Toutefois, cette étude considère toujours une topologie de communication fixe. Or, dans certaines applications, cette hypothèse n'est pas pleinement vérifiée, plus particulièrement lorsque la communication entre les agents dépend de leurs distances relatives ou de leurs attitudes en général. L'étude de topologies évolutives s'avère être nécessaire.

Un environnement 3D a été développé dans le chapitre II pour la simulation des robots coopératifs. Le but de cet environnement interactif est de plonger l'utilisateur dans une immersion totale. Dans cet objectif, nous avons proposé de mettre en œuvre une technique de réalité virtuelle. Combinant cette pratique avec l'usage de la plateforme réelle, le concept de réalité augmentée apparaît, pour une utilisation à distance, confortable et efficace, de la plate-forme coopérative.

Bibliographie

- [1] Chen G., Lewis F. L. ; Distributed Adaptive Tracking Control for Synchronization of Unknown Networked Lagrangian Systems ; IEEE Transactions on Systems, Man, and Cybernetics, Part B : Cybernetics, Vol 41, No. 3, pp. 805-816, June 2011.
- [2] J. Mei, W. Ren, and G. Ma ; Distributed coordinated tracking with a dynamic leader for multiple Euler-Lagrange systems ; IEEE Transactions on Automatic Control, No. 99, January 2011.
- [3] H. Yu, P. Antsaklis ; Passivity-based output synchronization of networked euler-lagrange systems subject to nonholonomic constraints ; American Control Conference (ACC) pp. 208-213, July 2010.
- [4] Long Cheng, Zeng-Guang Hou, Min Tan, Yingzi Lin, Wenjun Zhang ; Neural-Network-Based Adaptive Leader-Following Control for Multiagent Systems With Uncertainties ; IEEE Transactions on Neural Networks, Vol. 21, No. 8, pp : 1351-1358, August 2010.
- [5] Y.-C. Liu, N. Chopra ; Synchronization of networked robotic systems on strongly connected graphs ; 49th IEEE Conference on Decision and Control (CDC), pp. 3194-3199, 2010.
- [6] Wei Ren ; Consensus Tracking Under Directed Interaction Topologies : Algorithms and Experiments IEEE Transactions on Control Systems Technology, Vol. 18, No. 1, pp. 230-237, January 2010.
- [7] Bouteraa Y., Ghommam J., Derbel N., Poisson G. ; Adaptive Synchronization control of multi-robot teams : Cooperative and Coordinated schemes ; 18th IEEE Mediterranean Conference on Control and Automation Congress Palace Hotel, Marrakech, Morocco, June 23-25, 2010.
- [8] Filippo Arrichiello, Jnaneshwar Das, Hordur Heidarsson, Arvind Pereira, Stefano Chiaverini and Gaurav S. Sukhatme ; Multi-Robot Collaboration with Range-Limited Communication : Experiments with Two Underactuated ASVs ; Springer Tracts in Advanced Robotics, Vol. 62, pp. 443-453, 2010.

-
- [9] G Kaddoum, D Roviras, P Chargé and Danièle Fournier-Prunaret ; Robust synchronization for asynchronous multi-user chaos-based DS-CDMA Signal Processing ; Elsevier, vol.89, No. 5, pp. 807-818, May 2009.
 - [10] Wei Ren ; Distributed leaderless consensus algorithms for networked Euler-Lagrange systems ; International Journal of Control, Vol. 82, No. 11, pp. 2137-2149, 2009.
 - [11] Z. G. Hou, L. Cheng, and M. Tan ; Decentralized robust adaptive control for the multiagent system consensus problem using neural networks ; IEEE Trans. Syst., Man, Cybern. B, Cybern., Vol. 39, No. 3, pp. 636-647, June 2009.
 - [12] S. Chung and J. J. E. Slotine ; Cooperative robot control and concurrent synchronization of Lagrangian systems, IEEE Trans. Robot., Vol. 25, No. 3, pp. 686-700, Jun. 2009.
 - [13] Jing Yao, Hou-Neng Wang, Zhi-Hong Guan, Hua O. Wang ; Synchronization of leader-follower networks with coupling delays via variable structure control Vol. 11, No. 4, pp. 407-410, July 2009.
 - [14] S.-J. Chung, U. Ahsun, and J.-J. E. Slotine ; Application of Synchronization to Formation Flying Spacecraft : Lagrangian Approach ; Journal of Guidance, Control and Dynamics, Vol. 32, No. 2, pp. 512-526, March 2009.
 - [15] Dong Sun, Can Wang, Wen Shang and Gang Feng ; A Synchronization Approach to Trajectory Tracking of Multiple Mobile Robots While Maintaining Time-Varying Formations, IEEE Transactions on Robotics , Vol. 25, No. 5, October 2009.
 - [16] Wei Chen, Yueqing Yu, Xinhua Zhao and Lianyu Zhao ; Dynamic Control of the Underactuated Flexible Robot in the Operational Space ; Proceedings of the IEEE International Conference on Mechatronics and Automation, Vol. 9, No. 12, Changchun, China, August 2009.
 - [17] Kim D. Listmann and Craig A. Woolsey ; Output Synchronization of Systems in Chained Form ; 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference Shanghai, P.R. China, Vol. 16, No. 18, December 2009.
 - [18] Kim D. Listmann, Craig A. Woolsey and Jürgen Adamy ; Passivity-based coordination of multi-agent systems : A backstepping approach ; Proceedings of the European Control Conference, 2009.
 - [19] J. Chen, D. Sun, J. Yang, and H. Chen ; A leader-follower formation control of multiple nonholonomic mobile robots incorporating recedinghorizon scheme ; Int. J. Robot. Res., Vol. 28, 2009.
 - [20] Ghommam, J. Mnif, F. ; Coordinated Path-Following Control for a Group of Underactuated Surface Vessels ; IEEE Transactions on Industrial Electronics, Vol. 56, No. 10, pp. 3951-3963, Oct. 2009.

-
- [21] X. Xin, J-H. Sheb, T. Yamasakia, and Y. Liu; Swing-up control based on virtual composite links for n-link underactuated robot with passive first joint; *Automatica* Vol. 45, pp. 1986-1994, 2009.
 - [22] Ghommam, J.; *Commande Nonlinéaire et Navigation des Véhicules Marins Sous-actionnés*; Thèse présentée à l'Université d'Orléans pour l'obtention du grade de Doctorat, janvier 2008.
 - [23] Long Cheng Zeng-Guang Hou Min Tan; Decentralized adaptive leader-follower control of multi-manipulator system with uncertain dynamics, the 34th Annual Conference of IEEE Industrial Electronics, Nov. 2008.
 - [24] M. Arcak; Passivity as a Design Tool for Group Coordination; *IEEE Transaction On Automatic Control*, Vol. 52, No. 8, August 2007.
 - [25] J. Hu and Y. Hong; Leader following coordination of multi agent systems with coupling time delays; *Physica A*, vol. 374, No. 2, pp. 853-863, 2007.
 - [26] Wei Ren; Second-order Consensus Algorithm with Extensions to Switching Topologies and Reference Models *American Control Conference*, Vol. 9, No. 13, pp. 1431-1436, 2007.
 - [27] Reza Olfati-Saber; Consensus and Cooperation in Networked Multi- Agent Systems; invited paper, Vol. 95, No. 1, *Proceedings of the IEEE*, January 2007.
 - [28] Dong Sun; Xiaoyin Shao; Gang Feng; A Model-Free Cross-Coupled Control for Position Synchronization of Multi-Axis Motions : Theory and Experiments; *IEEE Transactions on Control Systems Technology*, Vol. 15, No. 2, pp. 306-314, march 2007.
 - [29] Dong Sun, Can Wang; Controlling Swarms of Mobile Robots for Switching between Formations Using Synchronization Concept; *IEEE International Conference on Robotics and Automation*, 2007.
 - [30] W. Ren; Multi-vehicle consensus with a time-varying reference state; *Systems & Control Letters*, Vol. 56, No. 7-8, pp. 474-483, July 2007.
 - [31] W. Ren, R. W. Beard, and E. Atkins; Information consensus in multivehicle cooperative control : collective group behavior through local interaction; *IEEE Control Systems Magazine*, Vol. 27, No. 2, pp. 71-82, Apr. 2007.
 - [32] M. N. Rooker and A. Birk; Multi-robot exploration under the constraints of wireless networking; *Control Engineering Practice*, Vol. 15, No. 4, pp. 435-445, Apr. 2007.
 - [33] W. Ren; Multi-vehicle consensus with a time-varying reference state; *Systems & Control Letters*, Vol. 56, No. 7-8, pp. 474-483, July 2007.
 - [34] W. N., Warren, M. Foss, and X. Guo; A Direct Lyapunov Approach for Stabilization of Underactuated Mechanical Systems; *Proceedings of the American Control Conference*, New York, Vol. 11, No. 13, , pp. 4817-4822, July 2007.

-
- [35] E. Kyrkjebø and K.Y. Pettersen; Operational space synchronization of two robot manipulators through a virtual velocity estimate; in Proc. 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, Vol. 12, No. 14, pp. 1052-1057, Dec. 2007
 - [36] Nima Yeganefar; Definitions et Analyse de stabilité pour les systèmes à retard non-linéaires; Thèse de Nima Yeganefar, Lille1, 2006.
 - [37] N. Chopra and M. W. Spong; Passivity-Based Control of Multi-Agent Systems, From Everyday Physics to Human-Like Movements; Advances in Robot Control, Springer Berlin Heidelberg, 2006.
 - [38] D. Lee and M. W. Spong; Stable flocking of multiple inertial agents on balanced graph; Proc. of the 2006 American Control Conf., Vol. 52, pp. 1469-1475, June 2006.
 - [39] Y. Hong, J. Hu, and L. Gao; Tracking control for multi-agent consensus with an active leader and variable topology; Automatica, Vol. 42, No. 7, pp. 1177-1182, 2006.
 - [40] Wei Wang, Slotine, J.-J.E., Contraction analysis of time-delayed communications and group cooperation, IEEE Transactions on Automatic Control, Vol. 51, No. 4, pp. 712-717, April 2006.
 - [41] M. W. Spong and N. Chopra; Synchronization of Networked Lagrange Systems, Lagrange and Hamiltonian methods for nonlinear control; Proceedings from the 3rd IFACWorkshop, Nagoya, Japan, July 2006.
 - [42] Xiaohai LI, Jizhong XIAO; Robot Formation Control in Leader-Follower Motion Using Direct Lyapunov Method; International Journal of Intelligent Control and Systems vol. 10, No. 3, 244-250, september 2005.
 - [43] W. Ren and R. W. Beard; Consensus seeking in multiagent systems under dynamically changing interaction topologies, IEEE Transactions on Automatic Control, Vol. 50, pp. 655-661, 2005.
 - [44] M. Long, A. Gage, R. Murphy, and K. Valavanis; Application of the distributed field robot architecture to a simulated deming task; in Proc. IEEE Int. Conf. Robot. Autom., Barcelona, Spain, pp. 3204-3211, Apr. 2005.
 - [45] W. Wang and J.J.E. Slotine; On partial contraction analysis for coupled nonlinear oscillators; Biological Cybernetics, Vol. 92, No. 1, pp. 38-53, 2005.
 - [46] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Coordinated multirobot exploration," Robotics, IEEE Transactions on, vol. 21, no. 3, pp. 376-386, 2005.
 - [47] W. Ren, R.W. Beard, and E.M. Atkins; survey of consensus problems in multi-agent coordination; Proceedings of American Control Conference, Portland, Oregon, pp. 1859-1864, 2005.

-
- [48] Z. Lin, B. Francis, and M. Maggiore ; Necessary and sufficient graphical conditions for formation control of unicycles ; IEEE Transactions on Automatic Control, Vol. 50, No. 1, pp. 121-127, 2005.
 - [49] A. Farinelli, L. Iocchi, and D. Nardi ; Multirobot systems : a classification focused on coordination ; Systems, Man and Cybernetics, Part B, IEEE Transactions, Vol. 34, No.5, pp. 2015-2028, 2004.
 - [50] H. Takahashi, H. Nishi, and K. Ohnishi ; Autonomous decentralized control for formation of multiple mobile robots considering ability of robot ; IEEE Trans. Ind. Electron., Vol. 51, No. 6, pp. 1272-1279, Dec. 2004.
 - [51] Alejandro Rodriguez-Angeles and Henk Nijmeijer ; Mutual Synchronization of Robots via Estimated : State Feedback : A Cooperative Approach ; IEEE Transactions On Control Systems Technology, Vol. 12, No. 4, July 2004.
 - [52] J.A. Fax and R.M. Murray ; Information flow and cooperative control of vehicle formations", IEEE Transactions on Automatic Control, Vol. 49, No. 9, pp. 1465-1476, 2004.
 - [53] De Sousa, J.B., Girard, A.R., Hedrick, J.K. ; Elemental maneuvers and coordination structures for unmanned air vehicles ; IEEE Conference on Decision and Control, Nassau, December 2004.
 - [54] Westervelt, E.R., Buche, G., and Grizzle, J. W. ; Experimental Validation of a Framework for the Design of Controllers that Induce Stable Walking in Planar Bipedes ; The International Journal of Robotics Research, Vol. 23, No. 6, pp. 559-582, 2004.
 - [55] Lin, Z., Broucke,M., and Francis, B. ; Local Control Strategies for Groups of Mobile Autonomous Agents ; IEEE Trans. on Automatic Control, 2004.
 - [56] Reza Olfati-Saber, R., and Murray, R.M. ; Consensus Problems in Networks of Agents with Switching Topology and Time-Delays ; in the special issue of the IEEE Transactions on Automatic Control on Networked Control Systems, Vol. 49, No. 9, Sept. 2004.
 - [57] H. Takahashi, H. Nishi, and K. Ohnishi ; Autonomous decentralized control for formation of multiple mobile robots considering ability of robot ; IEEE Trans. Ind. Electron., Vol. 51, No. 6, pp. 1272-1279, Dec. 2004.
 - [58] Bullo, F., and Lewis, A.D. ;Geometric Control of Mechanical Systems- Modeling Analysis, and Design for Simple Mechanical Control Systems ; Texts in Applied Mathematics, Springer- Verlag, 2004.
 - [59] Tedrake, R. L. ; Applied Optimal Control for Dynamically Stable Legged Locomotion ; Ph.D. thesis, Massachusetts Institute of Technology, 2004.

-
- [60] Wang, W., Yi, J., Zhao, D., and Liu, D. ; Design of a Stable Sliding-Mode Controller for a Class of Second-Order Underactuated Systems ; IEE Proc.-Control Theory Appl., Vol. 151, No. 6, November 2004.
 - [61] R. S. VARGA, Gerschgorin and His Circles, Springer-Verlag, Berlin, Germany, 2004.
 - [62] A. Jadbabaie, J. Lin, and A. S. Morse ; Coordination of groups of mobile autonomous agents using nearest neighbor rules ; IEEE Transactions on Automatic Control, Vol. 48, No. 6, pp. 988-1001, June 2003.
 - [63] Dong Sun ; Position synchronization of multiple motion axes with adaptative coupling control ; Automatica, Vol. 39, No.6, pp. 997-1005, 2003.
 - [64] J. R. T. Lawton, R.W. Beard, and B. J. Young ; A decentralized approach to formation maneuvers ; IEEE Trans. Robot. Autom., Vol. 19, No. 6, pp. 933-941, Dec. 2003.
 - [65] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos ; Nonholonomic navigation and control of multiple mobile manipulators ; IEEE Trans. Robot. Autom., Vol. 19, No. 1, pp. 53-64, Feb. 2003.
 - [66] Zhang, M., and Tarn, T.-J. ; A Hybrid Switching Control Strategy for Nonlinear and Underactuated Mechanical Systems ; IEEE Trans. on Automatic Control, Vol. 48, No. 10, October 2003.
 - [67] R. Olfati-Saber and R. M. Murray ; Flocking with obstacle avoidance : Cooperation with limited communication in mobile networks ; in Proceedings of the IEEE Conference on Decision and Control, Maui,Hawaii pp. 2022-2028, December 2003.
 - [68] Bloch, A.M, Baillieul, J., Crouch, P., and Marsden, J. ; Nonholonomic Mechanics and Control ; Springer-Verlag, New York, 2003.
 - [69] A. Rodriguez-Angeles, H. Nijmeijer ; Coordination of two robot manipulators based on position measurements only ; International Journal of Control, Vol. 74, No. 13, pp. 1311-1323, 2001.
 - [70] Z. Sun, S. S. Ge and T. H. Lee, Stabilization of underactuated mechanical systems : a nonregular backstepping approach, International Journal of Control, vol. 74, No. 11, pp. 1045-1051, 2001.
 - [71] Kharitonov, V., Zhabko, A. ; Approach to the robust stability analysis of time-delay systems. Automatica, Vol. 39, No. 1, pp. 15-20.
 - [72] Dubey, R. V., T. F. Chan and S. E. Everett ; Variable damping impedance control of a bilateral telerobotics system ; IEEE Control and Systems 17, 37-45.
 - [73] Guthart, G. S. and J. K. Salisbury Jr. The intuitive TM telesurgery system : overview and application. Conference on Robotics and Automation, Vol. 1, pp. 618-621, 2000.

-
- [74] P. Tomei ; A simple PD controller for robots with elastic joints ; IEEE Transactions on Automatic Control, Vol. 36, No. 10, pp. 1208-1213.
- [75] Wang, P.K.C., A. Sparks and S. Banda ; Coordination and control of multiple microspacecraft moving in formation ; The Journal of the Astronautical Sciences Vol. 44, pp. 315-355, 1996.
- [76] Kang, W. and H. Yeh ; Coordinated attitude control of multi-satellite systems ; International Journal of Robust and Nonlinear Control Vol. 12, 185-205, 2002.
- [77] Kocarev, L., K. S. Halle, K. Eckert and L. O. Chua ; Experimental demonstration of secure communication via chaotic synchronization ; International Journal of Bifurcations and Chaos Vol. 2, pp. 709-713, 1992.
- [78] Lee, H. K. and M. J. Chung ; Adaptive controller of a master-slave system for transparent teleoperation ; Journal of Robotic Systems, Vol. 15, No. 8, pp. 465-475, August 1998.
- [79] Cuomo, K. M., A. V. Oppenheim and S. H. Strogatz ; Synchronization of Lorenz-based chaotic circuits with application to communications ; IEEE Transactions on Circuit Systems II : Analog and Digital Signal Processing Vol. 40, pp. 626-633, 1993.
- [80] J. Tarn, A.K. Bejczy ; Intelligent planning and control for multirobot coordination : An event-based approach ; IEEE Transactions on Robotics and Automation, Vol. 12, No. 3, pp. 439-452, 1996.
- [81] M. A. Lewis and K. H. Tan ; High precision formation control of mobile robots using virtual structures ; Auton. Robot., Vol. 4, pp. 387-403, 1997.
- [82] W. Kang, N. Xi, and A. Sparks ; Formation control of autonomous agents in 3D workspace ; in Proc. IEEE Int. Conf. Robot. Autom., San Francisco, CA, pp. 1755-1760, Apr. 2000.
- [83] Olfati-Saber, R. ; Nonlinear Control of Underactuated Mechanical Systems with Application to Robotics and Aerospace Vehicles ; Ph.D. thesis, Department of EECS, MIT, February 2001.
- [84] Frazzoli, Emilio ; Robust Hybrid Control for Autonomous Vehicle Motion Planning ; Ph.D. Thesis, Department of Aeronautics and Astronautics, MIT, May 2001.
- [85] Dong Sun ; Mills, J.K. ; Adaptive synchronized control for coordination of multirobot assembly tasks ; IEEE Transaction on Robotics and Automation. Vol.18, No. 4, pp. 498 - 510, Aug 2002.
- [86] J. R. Lawton and R. W. Beard ; Synchronized multiple spacecraft rotations ; Automatica, Vol. 38, No. 8, pp. 1359 -1364, 2002.

-
- [87] D. Fox, W. Burgard, H. Kruppa, and S. Thrun; A probabilistic approach to collaborative multi-robot localization; *Auton. Robots*, Vol. 8, No. 3, pp. 325-344, Jun. 2000.
 - [88] T. Balch and R. Arkin; Behavior-based formation control for multi-robot systems; *IEEE Trans. Robot. Autom.*, Vol. 14, No. 6, pp. 926-939, Dec. 1998.
 - [89] M. Mesbahi and F. Hadaegh; Formation flying of multiple spacecraft via graphs, matrix inequalities, and switching; *AIAA J. Guid., Control, Dyn.*, Vol. 24, pp. 369-377, Mar. 2001.
 - [90] C. R. McInnes; Autonomous ring formation for a planar constellation of satellites; *AIAA J. Guid., Control, Dyn.*, Vol. 18, No. 5, pp. 1215-1217, 1995.
 - [91] F. Giulietti, L. Pollini, and M. Innocenti; Autonomous formation flight; *IEEE Control Syst. Mag.*, Vol. 20, No. 6, pp. 34-44, Jun. 2000.
 - [92] D. J. Stilwell and B. E. Bishop; Platoons of underwater vehicles; *IEEE Control Systems Magazine*, Vol. 20, pp. 45-52, December 2000.
 - [93] R.W. Beard and V. Stepanyan; Synchronization of information in distributed multiple vehicle coordinated control; in *Proceedings of the IEEE Conference on Decision and Control*, pp. 2029-2034, December 2003.
 - [94] N. Miyata, J. Ota, Y. Aiyama, H. Asama, and T. Arai; Cooperative transport in unknown environment : Application of real-time task assignment; *IEEE Transactions on Robotics and Automation*, Vol.18 No. 5, pp. 769 -780, October 2002
 - [95] G. Inalhan, D. M. Stipanovic, and C. J. Tomlin; Decentralized optimization, with application to multiple aircraft coordination; in *Proceedings of the IEEE Conference on Decision and Control*, pp. 1147-1155, December 2002.
 - [96] F. Y. Hadaegh, W.-M. Lu, and P. K. C. Wang; Adaptive control of formation flying spacecraft for interferometry; *IFAC*, 1998.
 - [97] Brown, S.C., Passino, K.M.; Intelligent Control for an Acrobot; *Journal of Intelligent and Robotic Systems*, Vol. 18, pp. 209-248, 1997.
 - [98] T. Fukuda, S. Nakagawa, Y. Kawauchi, and M. Buss; Structure decision method for self organising robots based on cell structures-cebot; *Proceedings. IEEE International Conference on Robotics and Automation*, Vol.2, pp. 695-700, 1989.
 - [99] M. Mataric; Minimizing complexity in controlling a mobile robot population; *Proceedings., IEEE International Conference on Robotics and Automation*, Vol.1, pp. 830-835, 1992.
 - [100] Weisbin C, Blitch J, Lavery D, Krotkov E, Shoemaker C, Matthies L, Rodriguez G "Miniature Robots for Space and Military Missions," *IEEE Robotics and Automation Magazine*, IEEE Press, Vol 6. No. 3, pp. 9-18 September 1999.

-
- [101] A. Davids, ;Urban search and rescue robots : from tragedy to technology ; Intelligent Systems, IEEE, Vol. 17, No. 2, pp. 81-83, 2002.
 - [102] R. Lozano, B. Brogliato, O. Egeland, and B. Maschke ; Dissipative System Analysis and Control : Theory and Applications ; Springer-Verlag, London, 2000.
 - [103] M.Bergerman and M. H.Terra ; Opt imal Control of Underactuated Manipulators via Actuation Redundancy ; Proc. of IEEE International Confe rence on Robotics and Automation, Vol.2, pp.2114-2119, May 2001.
 - [104] ; Decentralized coordination for multirobot exploration ; Robotics and Autonomous Systems, Vol. 29, No. 2-3, pp. 111-118, Nov. 1999.
 - [105] C. Weisbin and G. Rodriguez ; Nasa robotics research for planetary surface exploration ; Robotics and Automation Magazine, IEEE, Vol. 7, No. 4, pp. 25-34, 2000.
 - [106] K.Kobayashi, T.Yoshikawa. ; Controllability of Under-Actuated Planar Manipulators with One Unactuated Joint ; Proceedings of IEEE/RSJ Conf erence on Intelligent Robotics and Systems, Vol.I, pp.133-138, October 2000.
 - [107] Koivo, A. J., and Unseren, M. A. ;Reduced order model and decoupled control architecture for two manipulators holding a rigid object ; ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 113, No. 4, pp. 646-654, 1991.
 - [108] Tarn, T. J., Bejczy, A. K., and Yun, X. ; Coordinated control of two arms ; IEEE International Conference on Robotics and Automation, San Francisco, 1193-1202, 1986.
 - [109] Wen, J. T., and Delgado, K. K. ; Motion and force control of multiple robotic manipulators ; Automatica, Vol. 28, No. 4, pp. 729-743, 1992.
 - [110] Yun, X., Sarkar, N., and Kumar, V. J. ; Dynamic control of 3-D rolling contacts in two arm manipulation ; IEEE Transactions on Robotics and Automation, Vol. 13, No. 3, pp. 364-376, 1997.
 - [111] Kosuge, K. et al. ; Decentralized coordinated motion control of multirobots ; RSJ Annual Conference, pp. 705-706, 1993.
 - [112] Liu, Y. H., Arimoto, S., and Ogasawara, T. ; Decentralized cooperation control : non-communication object handling ; IEEE International Conference on Robotics and Automation, Minneapolis, pp. 2414-2419, 1996.
 - [113] Khalil, H. K. ; Nonlinear Systems. Prentice Hall. Upper Saddle River, New Jersey, 1996.
 - [114] Xi, N., Tarn, T. J., and Bejczy, A. K ; Intelligent planning and control for multi-robot coordination : an event-based approach ; IEEE Transactions on Robotics and Automation, Vol. 12, No. 3, pp. 439-452, 1996.

-
- [115] Blekhman, I. I., P. S. Landa and M. G. Rosenblum; Synchronization and chaotization in interacting dynamical systems; ASME Applied Mechanical Review Vol. 48, pp. 733-752, 1995.
- [116] WONG LEE, K., and KHALIL, H. K.; Adaptive output feedback control of robot manipulators using high-gain observer; Int. J. of Control, Vol. 67, pp.869-886, 1997.
- [117] LIU, Y.-H. , XU, Y., and BERGERMAN, M.; Cooperation control of multiple manipulators with passive joints; IEEE Trans. Robotics Automation, Vol. 15, pp. 258-267, 1999.
- [118] N. A. Lynch, Distributed Algorithms. San Francisco, California : Morgan Kaufmann Publishers, Inc., 1996.
- [119] Feng, L., Koren, Y., and Borenstein, J.; Cross-coupling motion controller for mobile robots; IEEE Control Systems Magazine, Vol. 13, No. 6, pp. 35-43.
- [120] H.Arai and S.Taehi; Position Control of a Manipulator with Passive Joints Using Dynamic Coupling; IEEE Transactions on Robotics and Automation, Vol.7, No.4, pp. 528-534, August 1991.
- [121] Arimoto, S., Miyazaki, F., and Kawamura, S.; Cooperative motion control of multiple robot arms or fingers; IEEE International Conference on Robotics and Automation, Raleigh, pp. 1407-1412, 1987.
- [122] Luh, J. Y. S., and Zheng, Y. F; Constrained relations between two coordinated industrial robots for motion control. International Journal of Robotics Research, Vol. 6, No. 3, pp. 60-70, 1987.
- [123] M.Bergerman, C. Lee, and Y.S.Xu; Dynamic Coupling of Underactuated Manipulators; Proc. of the IEEE Conference on Control Applications, vol.I2, No. 10, pp.500-505, October 1995.
- [124] Spong, M. W. and M. Vidyasagar (1989). Robot Dynamics and Control. Wiley. New York.
- [125] Brunt, M; Coordination of Redundant Systems. PhD thesis. Technical University Delft, The Netherlands, 1998.
- [126] Craig, J.J., P. Hsu and S. Sastry; Adaptive control of mechanical manipulators; Proceedings of IEEE Conference on Robotics and Automation, San Francisco, 1986.
- [127] Stramigioli, S.; From Differentiable Manifolds to Interactive Robot Control; PhD thesis, Technical University Delft, The Netherlands, 1998.
- [128] Breedveld, P. C., R. C. Rosenberg and T. Zhou; Bibliography of bond graph theory and application; Journal of the Franklin Institute, Vol. 328, pp. 1067-1109, 1991.

-
- [129] Book, W. J. ; Recursive lagrangian dynamics of flexible manipulator arms ; International Journal of Robotics Research Vol. 3, pp. 87-101, 1984.
 - [130] H. Goldstein ; Classical Mechanics (Second Edition) ; Addison-Wesley Publishing Company, Reading, Massachusetts. ISBN 0-201-02969-3, (1980)
 - [131] J. R. Lawton and R. W. Beard ; Synchronized multiple spacecraft rotations ; Automatica, Vol. 38, No. 8, pp. 1359-1364, 2002.
 - [132] R. W., Brockett ; Asymptotic stability and feedback stabilization. Differential geometric control theory. Boston : Birkhauser, 1983.
 - [133] R. Olfati-Saber and A. Megretski ; Control for a Class of Underactuated Nonlinear Systems ; Proceedings of the IEEE Conference on Decision and Control, pp. 4182-4187, 1998.
 - [134] A. Rodriguez-Angeles ; Synchronization of Mechanical Systems ; PhD thesis. Technische Universiteit Eindhoven, Netherlands, 2002.
 - [135] H. Nijmeijer and A. Rodriguez-Angeles ; Synchronization of mechanical systems ; World Scientific Publishing Company, Nov. 2003.
 - [136] M. W. Spong ; Partial feedback linearization of underactuated mechanical systems ; In Proc. of the IEEE/RSJ/GI Intelligent Robots and Systems. Advanced Robotic Systems and the Real World, pp 314-321, Munich , Germany, 1994.

Yassine BOUTERAA

Commande distribuée et synchronisation de robots industriels coopératifs

Résumé : Cette thèse développe les lois de coordination de systèmes de Lagrange. Elle propose en premier lieu une stratégie complètement décentralisée qui se base sur la technique de *cross-coupling* pour la commande d'un groupe de robots, appelé réseau, qui synchronisent leurs mouvements en suivant une trajectoire désirée. Cette stratégie est étendue pour faire face à l'incertitude paramétrique des robots ainsi qu'aux retards fréquemment rencontrés dans les applications pratiques de réseaux de communication. Une deuxième architecture basée sur la théorie des graphes est proposée pour les réseaux à leader. L'approche développée est considérée hybride. Une extension adaptative à base de réseaux de neurones est développée pour traiter les cas d'incertitude paramétrique. La stratégie conçue prend en considération les délais dans la réception des données. En se basant sur la notion de système en chaîne, la théorie des graphes, le concept de la passivité et la technique du *backstepping*, une nouvelle méthodologie de la conception de contrôleur de synchronisation pour une classe de systèmes sous-actionnés est développée. Afin d'avoir la possibilité d'implémenter ces stratégies de contrôle, on a développé une plate-forme d'expérimentation pour la robotique industrielle coopérative.

Mots clés : coopération de robots, synchronisation, techniques de coordination, système sous actionné, commande distribuée, commande à retard, commande non linéaire.

Distributed Control and Synchronization of Cooperative Robot Manipulators

Abstract : This thesis investigates the issue of designing decentralized control laws to cooperatively control a team of robot manipulators. The purpose is to synchronize their movements while tracking common desired trajectory. Based on a combination of Lyapunov direct method and cross-coupling technique, To account for unmatched uncertainties, the proposed decentralized control laws are extended to an adaptive synchronization tracking controllers. Moreover, due to communication imperfection, time delay communication problems are considered in the performance analysis of the controllers. Another relevant problem for distributed synchronized systems is the leader-follower control problem. In this strategy, a decentralized control laws based on the backstepping scheme is proposed to deal with a leader-follower multiple robots structure. Based on graph theory, the coordination strategy combines the leader follower control with the decentralized control. The thesis, also considers the cooperative movement of under- actuated manipulators tracking reference trajectories defined by the user. The control problem for a network of class of under-actuated systems is considered. The approach we adopted in this thesis consists in decomposing the under-actuated manipulators into a cascade of passive subsystems that synchronize with he other neighbors subsystems. The resulting synchronized control law is basically a combination of non-regular backstepping procedure aided with some concepts from graph theory. The proposed controllers are validated numerically, assuming that the underlying communication graph is strongly connected. To implement these control strategies, we developed an experimental platform made of three robot manipulators.

Keywords : Multi-robot, cooperative synchronization, coordination, Lagrangian system, under-actuated system, time-delay systems, distributed control, non-linear control.



Equipe Système Robotiques Interactifs (SRI),
Institut PRISME de recherche, Université
d'Orléans, 18020 Bourges cedex (France)