

Université de Batna
Faculté de Technologie
Département d'Electronique

Thèse

présentée pour l'obtention du titre de

Docteur en Sciences

Spécialité : CONTRÔLE-ROBOTIQUE

par **Salima DJEBRANI**

Implémentation Comportementale Multi-Agent dans un milieu Robotique

Soutenue publiquement le 27/01/2013 devant la commission d'examen composée de

<i>Président :</i>	Noureddine GOLEA	Prof.	U. Oum El Bouagui
<i>Rapporteurs :</i>	Foudil ABDESSEMED	Prof.	U. Batna
	Abderraouf BENALI	MC.	ENSI Bourges
<i>Examineurs :</i>	Driss BOUTAT	Prof.	ENSI Bourges
	Djamel-Eddine AMEDDAH	MC.	U. Batna
	Yassine ABDESSEMED	MC.	U. Batna

Remerciements

Mes remerciements sont adressés à Monsieur Foudil ABDESSEMED, Professeur au Département d'Electronique de l'Université de Batna, pour avoir accepté de m'encadrer et d'assurer la direction scientifique de mes travaux. Je le remercie aussi pour sa gentillesse et sa compréhension.

Mes grands remerciements vont particulièrement à Monsieur Abderraouf BENALI, Maître de Conférences à l'Ecole Nationale Supérieure des Ingénieurs de Bourges, France, pour avoir participé à l'encadrement et pour ses orientations et ses judicieux conseils tout au long de l'élaboration de mon travail. Je le remercie encore une fois pour avoir été gentil, compréhensif et bon conseiller.

Mes respects et ma gratitude vont également aux membres de jury qui m'ont fait l'honneur de juger ce travail :

A Monsieur Noureddine GOLEA, Professeur au Département des Sciences et de la Technologie de l'Université de Oum El-Bouagui, pour avoir accepté de présider le jury.

A Monsieur Driss BOUTAT, Professeur des Universités à l'Ecole Nationale Supérieure des Ingénieurs de Bourges, France, pour m'avoir honoré par sa présence au jury.

A Monsieur Djamel-Eddine AMEDDAH, Maître de Conférences au Département d'Electrotechnique de l'Université de Batna, pour avoir accepté de juger ce travail.

A Monsieur Yassine ABDESSEMED, Maître de Conférences au Département d'Electronique de l'Université de Batna, pour avoir accepté de juger ce travail.

Mes remerciements vont également à monsieur Joël ALLAIN, directeur de l'Ecole Nationale Supérieure des Ingénieurs de Bourges, France, de m'avoir accueilli au sein de son école tout au long de mon séjour, et qui m'a offert toutes les conditions favorables à l'accomplissement de mes travaux. Ainsi qu'au personnel de l'école, qui m'a permis d'effectuer cette thèse dans de très bonnes conditions de travail. Je tiens également à adresser mes plus sincères remerciements à tous les personnels de l'ENSI de Bourges et de l'institut PRISME.

Je remercie également Monsieur Christophe ROSENBERGER, professeur des Univer-

sités à l'Ecole Nationale Supérieure des Ingénieurs de Caen, France, pour son aide, sa gentillesse et ses éclaircissements.

Enfin, une pensée émue pour tous les amis de l'ENSI : Hamid, Adel, Karim, Mohamed, Khaled, Oussama, Siham, Yassine, Loubna, Ali. Je n'oublie pas non plus les autres parmi lesquels je remercierai plus particulièrement : Nora, Assia, Soraya, Leila, Ibtissem, Naima, Rabia.

J'adresse un grand merci à toute ma famille qui a toujours été présente lorsque j'en ai eu besoin, en particulier mon père, ma sœur, mes deux frères et mes belles-sœurs.

Finalement je témoigne ma reconnaissance à ma famille, mes amis et mes proches, sans votre soutien et votre confiance rien n'aurait pu être fait.

A la mémoire de ma très chère maman

Résumé

Cette thèse s'inscrit dans le cadre des sujets de coopération du programme intergouvernemental entre l'Algérie et la France. Le sujet de la thèse concerne l'implémentation comportementale multi-agents dans un milieu robotique. Il s'agit d'un sujet complexe en rapport avec la multitude de situations d'interactions qu'un robot peut rencontrer. Le sujet de la thèse a pour objectif d'arriver à contrôler un nombre de robots dans l'exécution de tâches complexes nécessitant coopération et coordination. En effet, le sujet propose une partie de l'approche multi-agents où l'exécution de la tâche dans un environnement complexe repose sur l'intelligence distribuée et l'apport de chaque agent dans la réussite de la tâche. Ainsi, le problème se réduit à de simples comportements qui émergent de nombreuses interactions entre les agents et leur environnement. Nous avons proposé une structure d'un manipulateur mobile qui représente un agent ; étant donné que le manipulateur mobile est composé d'une plate-forme mobile sur laquelle repose un bras manipulateur. Le système constitué d'un ensemble de manipulateurs mobiles, chacun peut communiquer avec les autres afin de réaliser la tâche demandée, cette dernière réside dans la coopération entre deux manipulateurs mobiles ou plus dans une manipulation continue d'un objet. Le choix de la stratégie doit prendre en compte toutes les dynamiques ; dont celle de l'environnement imprévisible ainsi que celle des autres robots.

Nous avons au cours de cette thèse choisi de hiérarchiser les niveaux de la commande en considérant deux niveaux :

- Un niveau haut qui assure la coopération et la coordination en utilisant la négociation entre les robots afin de résoudre le problème de conflit.
- Un niveau bas, où nous avons présenté les différentes étapes à la modélisation cinématique, dynamique, et l'implémentation d'une commande cinématique basée sur les tâches additionnelles avec l'usage de l'impédance. La seconde commande, est une structure de la commande dynamique permettant d'intégrer la théorie de la commande par retour d'état linéarisant, et une commande d'impédance qui prend en compte la présence d'obstacles ou d'autres robots.

Mots clés : Manipulateur mobile holonôme, système multi-agents, modélisation cinématique, modélisation dynamique, tâches additionnels, commande en impédance, commande par retour d'état linéarisant, logique floue, identification.

Publications internationales et nationales

- Salima DJEBRANI, Foudil ABDESSEMED and Abderraouf BENALI; A Multi-Agent Strategy for a Simple Cooperative Behavior; International Journal of Information Acquisition, IJIA 2010, World Scientific Publishing Company, volume. 7, No. 4, pp. 331-345, December, 2010.
- Salima DJEBRANI, Abderraouf BENALI and Foudil ABDESSEMED; Modelling and Control of an Omnidirectional Mobile Manipulator; International Journal of Applied Mathematics and Computer Science, AMCS 2012, volume. 22, No. 3, pp. 601-616, 2012.
- Salima DJEBRANI, Abderraouf BENALI and Foudil ABDESSEMED; Force-Position Control of a Holonomic Mobile Manipulator; The 12th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines, CLAWAR 2009, pp. 1023-1030, Istanbul, Turkey, September, 2009.
- Salima DJEBRANI, Abderraouf BENALI and Foudil ABDESSEMED; Impedance Control of an Omnidirectional Mobile Manipulator; The 14th International Conference on Methods and Models in Automation and Robotics, MMAR 2009, Miedzydroje, Poland, August, 2009.
- Salima DJEBRANI and Foudil ABDESSEMED; Simple Cooperative Behavior-based Multi-Agent Prototyping; 2nd International Conference On Electronics Systems CISE 2009, pp. 80-85, Batna, Algeria, October, 2009.
- Salima DJEBRANI and Foudil ABDESSEMED; Multi-Agent Prototyping for a Cooperative Carrying Task; IEEE International Conference on Robotics and Biomimetics, ROBIO 2009, pp. 1421-1426, Guilin, Guangxi, China, December, 2009.
- Foudil ABDESSEMED and Salima DJEBRANI; Trajectory Generation for Mobile Manipulators; Robot Manipulators Trends and Development, Edited by : Prof Dr Agustin Jimenez and Dr Basil M Al Hadithipp, INTECH 2010, pp. 335-360, March, 2010.
- Salima DJEBRANI, Abderraouf BENALI and Gérard POISSON; Input-State Li-

nearisation of an Omni-Directional Mobile Robot; IEEE International Symposium on Industrial Electronics, ISIE 2010, pp. 1889-1894, Bary, Italy, July, 2010.

- Salima DJEBRANI, Abderraouf BENALI and Foudil ABDESSEMED; Modelling and Feedback Control of an Omni-Directional Mobile Manipulator; IEEE Conference on Automation Science and Engineering, CASE 2011, pp. 785-791, Trieste, Italy, August, 2011.

Table des matières

I	Introduction générale	14
I.1	La naissance de la robotique mobile	14
I.2	Des applications hétéroclites	16
I.2.1	Robotique d'exploration	16
I.2.2	Productions industrielles	17
I.3	Les manipulateurs mobiles	18
I.3.1	Les multipèdes	18
I.3.2	Les manipulateurs mobiles sous-marins	19
I.3.3	Les manipulateurs mobiles à roues	20
I.4	Les systèmes multi-agents	22
I.4.1	Problématique	22
I.4.1.1	Cadre général	22
I.4.1.2	L'intelligence collective	24
I.4.2	Travaux existants	25
I.5	Description de la méthode proposée	25
I.5.1	Description de l'architecture d'un robot	26
I.6	Organisation de la thèse	27
II	Modélisation	30
II.1	Introduction	30
II.2	Modélisation du bras manipulateur	31
II.2.1	Espace articulaire, espace opérationnel et modèle de transformation entre les espaces	31
II.2.1.1	Modèle Géométrique Direct (MGD)	32

II.2.1.2	Modèle Géométrique Inverse (MGI)	32
II.2.1.3	Modèle Cinématique Direct (MCD)	32
II.2.1.4	Modèle Cinématique Inverse (MCI)	33
II.2.2	Redondance et configurations singulières	33
II.2.3	Modélisation dynamique du bras manipulateur	34
II.2.4	Bras manipulateur plan à deux liaisons rotoïdes	34
II.3	Modélisation des plates-formes mobiles à roues	36
II.3.1	Robot mobile à roues	36
II.3.2	Etude des configurations	37
II.3.2.1	Plate-forme de type unicycle	38
II.3.2.2	Plate-forme de type omnidirectionnel	39
II.3.3	Modèles cinématiques des plates-formes mobiles à roues	39
II.3.4	Robot omnidirectionnel <i>ROMNI</i>	40
II.3.4.1	Modélisation cinématique de la plate-forme mobile omni- directionnelle	43
II.3.4.2	Modélisation dynamique de la plate-forme mobile omni- directionnelle	46
II.4	Modélisation des manipulateurs mobiles à roues	49
II.4.1	Description des manipulateurs mobiles à roues	49
II.4.2	Etude géométrique des manipulateurs mobiles à roues	50
II.4.3	Etude cinématique des manipulateurs mobiles à roues	51
II.4.3.1	Modèle cinématique en situation	52
II.4.3.2	Modèle cinématique en configuration	53
II.4.4	Redondance et configurations singulières	53
II.4.5	Modèle cinématique du manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons	54
II.4.6	Modélisation dynamique des manipulateurs mobiles à roues	56
II.4.7	Modélisation dynamique du manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons rotoïdes	59
II.5	Conclusion	62

III	Commande du manipulateur mobile	63
III.1	Introduction	63
III.2	Manipulabilité des bras manipulateurs et des manipulateurs mobiles	64
III.2.1	Manipulabilité des bras manipulateurs	64
III.2.1.1	Mesures de manipulabilité	65
III.2.1.2	Application de la manipulabilité aux bras manipulateurs	66
III.2.2	Manipulabilité des manipulateurs mobiles	67
III.3	La commande du manipulateur mobile	68
III.3.1	Commande séparée des mouvements du manipulateur et de la base mobile omnidirectionnelle	68
III.3.2	Commande à mouvement opérationnel imposé	69
III.3.2.1	Méthodes d'inversion : Cas des bras manipulateurs	70
III.3.2.2	Méthodes d'inversion : Cas des manipulateurs mobiles	71
III.3.2.3	Méthode des tâches additionnelles	71
III.4	Application sur les manipulateurs mobiles omnidirectionnels	72
III.4.1	Manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons	72
III.5	Evitement d'obstacles	74
III.5.1	Méthodes d'évitement d'obstacles	77
III.5.1.1	Méthode des champs de potentiel	77
III.5.1.2	Méthode " <i>Vector Field Histogram</i> "	78
III.5.2	Commande en impédance	80
III.5.2.1	Implémentation de la commande en impédance	80
III.6	Application sur les manipulateurs mobiles omnidirectionnels	82
III.6.1	Manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons	82
III.6.2	Manipulateur mobile omnidirectionnel avec un bras manipulateur à trois liaisons rotoïdes	88
III.7	Conclusion	96
IV	Commande par retour d'état linéarisant	97
IV.1	Introduction	97

IV.2	Rappel sur la commande par retour d'état linéarisant	98
IV.2.1	Théorie de la linéarisation par retour d'état	98
IV.2.1.1	Difféomorphisme	98
IV.2.1.2	Champ de vecteurs	98
IV.2.1.3	Dérivée de Lie	98
IV.2.1.4	Crochet de Lie	99
IV.2.1.5	Involutivité	99
IV.2.2	Linéarisation au sens entrée-état d'un système monovariable	100
IV.2.2.1	Conditions pour une linéarisation au sens entrée-état . . .	101
IV.2.2.2	Etapes à suivre	101
IV.2.3	Linéarisation par retour d'état des systèmes multivariables	102
IV.3	Linéarisation par retour d'état du robot mobile omnidirectionnel <i>ROMNI</i> .	104
IV.3.1	Calcul de loi de commande	104
IV.3.2	Implémentation de la commande en impédance	108
IV.3.3	Résultats de simulation	109
IV.4	Linéarisation par retour d'état du manipulateur mobile omnidirectionnel .	112
IV.4.1	Calcul de loi de commande	112
IV.4.2	Implémentation de la commande en impédance	117
IV.4.3	Résultats de simulation	118
IV.5	Conclusion	122

V	Adaptation des paramètres de l'impédance & identification des paramètres dynamiques	123
V.1	Introduction	123
V.2	Les concepts de la logique floue	124
V.2.1	Histoire de la logique floue	124
V.2.2	Commande et réglage par la logique floue	125
V.2.2.1	Construction d'un contrôleur flou	125
V.2.2.2	Le contrôleur de Takagi-Sugeno	128
V.2.3	Adaptation des paramètres de l'impédance	129
V.2.4	Résultats de simulation	131
V.3	Identification des paramètres dynamiques	131

V.3.1	Identification des paramètres dynamiques du manipulateur mobile omnidirectionnel	133
V.3.2	Résultats de simulation	135
V.4	Conclusion	135
VI	Les systèmes multi-agents	138
VI.1	Introduction	138
VI.2	Systèmes multi-agents et applications de robotique collective	138
VI.2.1	Etude des systèmes multi-agents	139
VI.2.1.1	Définition d'un agent	139
VI.2.1.2	Définition d'un système multi-agents (SMA)	140
VI.2.1.3	Architectures d'agents	140
VI.2.2	Applications à la robotique	144
VI.2.2.1	Application à la robotique modulaire	144
VI.2.2.2	Application à la robotique collective	144
VI.3	Description de la méthode proposée	146
VI.3.1	Objectif de la mission	146
VI.3.2	Comportements du robot	147
VI.3.3	Fixer une cible	148
VI.3.4	Recherche aléatoire d'un objet	149
VI.3.5	Evitement d'obstacles	149
VI.3.5.1	Repositionnement du robot	150
VI.3.5.2	Contournement de l'obstacle	150
VI.3.5.3	Direction vers le point de sûreté	151
VI.3.6	Protocole de communication entre les robots	151
VI.3.7	Protocole de négociation entre les robots	152
VI.3.8	Définition des rôles	153
VI.4	Résultats de simulation	154
VI.5	Conclusion	165
VII	Conclusion générale	166
	Annexes	170

A	Outils mathématiques	171
A.1	Inversion matricielle	171
A.1.1	Matrices inverses à gauche et à droite	171
A.1.2	Matrice inverse généralisée et matrice pseudo-inverse	171
A.1.2.1	Calcul de la matrice pseudo-inverse	172
A.2	Résolution d'un système linéaire	172
A.2.1	Cas où A est de rang maximal	173
A.2.2	Cas où A est de rang déficient	173
A.3	Décomposition en valeurs singulières	173
B	Le fonctionnement interne d'oRis	175
B.1	Langage interprété et orienté objet	175
B.1.1	Programmation orientée objet	175
B.1.2	Programmation structurée	176
B.1.3	Instructions de base	177
B.1.4	Qu'est ce qu'un bloc "execute" ?	177
B.2	Langage permettant la simulation	177
B.2.1	Fonctionnement des threads : agent	177
B.2.2	Le monde multi-agents	178
B.3	Principe de vision : "view" et "viewFirst"	178
	Bibliographie	180

Table des figures

I.1	W. Grey Walter's tortoise (<i>Machinaspeculatrix</i>)	15
I.2	Robot <i>SHAKY</i> du <i>Stanford Research Institute</i>	15
I.3	Robot <i>SEJOURNER</i>	17
I.4	Robot <i>NOMAD</i>	17
I.5	Robot <i>XR4000</i>	18
I.6	Robot <i>PIONNER</i>	18
I.7	Le robot humanoïde <i>QRIO</i>	19
I.8	Le robot humanoïde <i>ASIMO</i>	19
I.9	Le robot <i>VICTOR 6000</i>	20
I.10	Le robot <i>ACHILLE 2000</i>	20
I.11	Le manipulateur mobile <i>SPIRIT</i>	21
I.12	Le manipulateur mobile <i>CARE-O-BOT</i>	21
I.13	Le manipulateur mobile <i>KAMRO</i>	21
I.14	Le manipulateur mobile <i>H2BIS</i>	22
I.15	Le manipulateur mobile <i>ANIS</i>	22
I.16	Architecture d'un robot	26
II.1	Un bras manipulateur plan à deux liaisons rotoïdes	35
II.2	Plate-forme mobile de type unicycle	38
II.3	Plate-forme mobile de type omnidirectionnel	39
II.4	Modèles des deux types de plates-formes	40
II.5	Le robot omnidirectionnel <i>ROMNI</i>	41
II.6	La plate-forme réelle	41
II.7	Disposition des trois essieux	41
II.8	Demi essieu	41

II.9	Essieu à sphères alignées	41
II.10	Essieu à sphères juxtaposées	41
II.11	Représentation d'un essieu	42
II.12	Repère absolu, repère robot et repère essieu	43
II.13	Repère robot et repère essieu	45
II.14	Diagramme dynamique de la base à trois roues	47
II.15	Géométrie des manipulateurs mobiles	51
II.16	Manipulateur mobile plan avec une plate-forme omnidirectionnelle	55
III.1	Ellipsoïde de manipulabilité en $3D$	66
III.2	Ellipses de manipulabilité	67
III.3	Mesure de manipulabilité w_r	67
III.4	Position ξ_1 et vitesse $\dot{\xi}_1$ de l'organe terminal	75
III.5	Position ξ_2 et vitesse $\dot{\xi}_2$ de l'organe terminal	75
III.6	Position x et vitesse \dot{x} de la plate-forme	75
III.7	Position y et vitesse \dot{y} de la plate-forme	75
III.8	Orientation ϑ et vitesse angulaire $\dot{\vartheta}$ de la plate-forme	76
III.9	Orientation q_{r_1} et vitesse angulaire \dot{q}_{r_1} du bras1	76
III.10	Orientation q_{r_2} et vitesse angulaire \dot{q}_{r_2} du bras2	76
III.11	Grille d'occupation locale construite par la méthode "Histogrammic in motion mapping"	79
III.12	Utilisation de l'histogramme des obstacles pour déterminer la direction de déplacement du robot	79
III.13	Le système de perception US du robot	80
III.14	Le schéma bloc pour le système de contrôle	81
III.15	La force répulsive causée par l'obstacle	83
III.16	L'évolution du bras manipulateur et de la plate-forme dans le cas d'un mouvement libre	84
III.17	Trajectoires de l'effecteur et de la plate-forme	84
III.18	Courbe d'évolution de l'orientation ϑ	84
III.19	Courbe d'évolution de l'articulation q_{r_1}	84
III.20	Courbe d'évolution de l'articulation q_{r_2}	85

III.21	L'évolution du bras manipulateur et de la plate-forme dans le cas d'un seul obstacle	85
III.22	Trajectoires de l'effecteur et de la plate-forme	85
III.23	Courbe d'évolution de l'orientation ϑ	85
III.24	Courbe d'évolution de l'articulation q_{r_1}	86
III.25	Courbe d'évolution de l'articulation q_{r_2}	86
III.26	L'évolution du bras manipulateur et de la plate-forme dans le cas de deux obstacles	86
III.27	Trajectoires de l'effecteur et de la plate-forme	86
III.28	Courbe d'évolution de l'orientation ϑ	87
III.29	Courbe d'évolution de l'articulation q_{r_1}	87
III.30	Courbe d'évolution de l'articulation q_{r_2}	87
III.31	Vue 3D de l'évolution du bras manipulateur et de la plate-forme dans le cas d'un mouvement libre	92
III.32	Trajectoires de l'effecteur et de la plate-forme	92
III.33	Courbe d'évolution de l'orientation ϑ	92
III.34	Courbe d'évolution de l'articulation q_{r_1}	92
III.35	Courbe d'évolution de l'articulation q_{r_2}	93
III.36	Courbe d'évolution de l'articulation q_{r_3}	93
III.37	Vue 3D de l'évolution du bras manipulateur et de la plate-forme dans le cas d'un seul obstacle	93
III.38	Trajectoires de l'effecteur et de la plate-forme	93
III.39	Courbe d'évolution de l'orientation ϑ	94
III.40	Courbe d'évolution de l'articulation q_{r_1}	94
III.41	Courbe d'évolution de l'articulation q_{r_2}	94
III.42	Courbe d'évolution de l'articulation q_{r_3}	94
III.43	Vue 3D de l'évolution du bras manipulateur et de la plate-forme dans le cas de deux obstacles	95
III.44	Trajectoires de l'effecteur et de la plate-forme	95
III.45	Courbe d'évolution de l'orientation ϑ	95
III.46	Courbe d'évolution de l'articulation q_{r_1}	95

III.47	Courbe d'évolution de l'articulation q_{r_2}	96
III.48	Courbe d'évolution de l'articulation q_{r_3}	96
IV.1	Schéma du contrôleur proposé pour le robot mobile	109
IV.2	Exemple 1 de navigation du robot mobile dans un milieu encombré d'obstacles	110
IV.3	Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}	110
IV.4	Position x et vitesse \dot{x} du robot mobile	111
IV.5	Position y et vitesse \dot{y} du robot mobile	111
IV.6	Exemple 2 de navigation du robot mobile dans un milieu encombré d'obstacles	111
IV.7	Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}	111
IV.8	Position x et vitesse \dot{x} du robot mobile	112
IV.9	Position y et vitesse \dot{y} du robot mobile	112
IV.10	Schéma du contrôleur proposé pour le manipulateur mobile	117
IV.11	Position x et vitesse \dot{x} de la plate-forme	119
IV.12	Position y et vitesse \dot{y} de la plate-forme	119
IV.13	Orientation q_{r_1} et vitesse angulaire \dot{q}_{r_1} du bras1	119
IV.14	Orientation q_{r_2} et vitesse angulaire \dot{q}_{r_2} du bras2	119
IV.15	Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}	120
IV.16	Exemple de navigation du manipulateur mobile dans un milieu encombré d'obstacles	120
IV.17	Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}	120
IV.18	Position x et vitesse \dot{x} de la plate-forme	121
IV.19	Position y et vitesse \dot{y} de la plate-forme	121
IV.20	Orientation q_{r_1} et vitesse angulaire \dot{q}_{r_1} du bras1	121
IV.21	Orientation q_{r_2} et vitesse angulaire \dot{q}_{r_2} du bras2	121
V.1	Structure d'une commande floue	125
V.2	Schéma fonctionnel du système à contrôler	130
V.3	Fonctions d'appartenance pour la variable F_{obs}	130
V.4	Fonctions d'appartenance pour les variables d_4 et d_5	130

V.5	Exemple de navigation du manipulateur mobile dans un couloir	132
V.6	Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}	132
V.7	Position x et vitesse \dot{x} de la plate-forme	132
V.8	Position y et vitesse \dot{y} de la plate-forme	132
V.9	La variation de la force F_{obs}	133
V.10	La variation de l'impédance Z_d	133
V.11	Les valeurs estimées $\hat{P}e_1$ et $\hat{P}e_2$	136
V.12	Les valeurs estimées $\hat{P}e_3$ et $\hat{P}e_4$	136
V.13	Les valeurs estimées $\hat{P}e_5$ et $\hat{P}e_6$	136
V.14	Les valeurs estimées $\hat{P}e_7$ et $\hat{P}e_8$	136
VI.1	Représentation d'un système multi-agents	141
VI.2	Schéma d'un agent à réflexes simples [1]	142
VI.3	Schéma d'un agent conservant une trace du monde [1]	142
VI.4	Schéma d'un agent ayant des buts [1]	144
VI.5	Schéma d'un agent basé sur l'utilité [1]	144
VI.6	Modélisation d'un robot manipulateur par un système multi-agents	145
VI.7	Navigation dans un environnement encombré d'obstacles	147
VI.8	Organigramme de déplacement pour le robot	148
VI.9	Détermination de l'angle θ	149
VI.10	Déplacement vers le point A	151
VI.11	Contournement de l'obstacle	151
VI.12	Déplacement vers le point C	151
VI.13	Basculement de rôles dans le cas de blocage	153
VI.14	Schéma de l'environnement de simulation	154
VI.15	Robot dans la position initiale	155
VI.16	Recherche aléatoire de l'objet	155
VI.17	Détection de l'objet	156
VI.18	Déplacement vers l'extrémité la plus proche	156
VI.19	Le message envoyé par le robot	156
VI.20	Deux robots dans la position initiale	156
VI.21	Recherche aléatoire de l'objet	157

VI.22	Détection de l'objet et déplacement vers l'extrémité la plus proche	157
VI.23	Déplacement vers l'autre extrémité	157
VI.24	Les messages envoyés par les deux robots	157
VI.25	Robots dans la position initiale	159
VI.26	Recherche aléatoire de l'objet	159
VI.27	Détection de l'objet et déplacement vers l'extrémité la plus proche	159
VI.28	Déplacement vers l'autre extrémité	159
VI.29	Les messages envoyés par les robots	160
VI.30	Robots dans la position initiale	160
VI.31	Recherche aléatoire de l'objet	160
VI.32	Détection du premier objet et déplacement vers l'extrémité la plus proche	160
VI.33	Déplacement vers l'autre extrémité du premier objet	161
VI.34	Détection du deuxième objet et déplacement vers l'extrémité la plus proche	161
VI.35	Déplacement vers l'autre extrémité du deuxième objet	161
VI.36	Les messages envoyés par les robots	161
VI.37	Robot dans la position initiale	162
VI.38	Evitement de l'obstacle (a)	162
VI.39	Evitement de l'obstacle (b)	162
VI.40	Déplacement vers le but	162
VI.41	Robot dans la position initiale	163
VI.42	Evitement des obstacles (a)	163
VI.43	Evitement des obstacles (b)	163
VI.44	Evitement des obstacles (c)	163
VI.45	Déplacement vers le but	164
VI.46	Objet dans la position initiale	164
VI.47	Transport de l'objet	164
VI.48	Objet dans la position initiale	164
VI.49	Les messages envoyés par les robots	164
VI.50	Négociation entre les robots (a)	164
VI.51	Négociation entre les robots (b)	165
VI.52	Le message envoyé par le robot inactif	165

VI.53	Transport de l'objet	165
B.1	Fonctionnement du mécanisme de vision sous oRis	179

Chapitre I

Introduction générale

I.1 La naissance de la robotique mobile

Pour commencer, il est utile d'expliciter la notion de robot mobile. En robotique, nous distinguons les robots en deux principaux types : les robots manipulateurs et les robots mobiles. Les robots manipulateurs ont une base fixe contrairement aux robots mobiles qui peuvent se déplacer. Toute manipulation des robots manipulateurs est limitée par leurs dimensions et par leur nombre de degrés de mobilité. Donc, à la différence des robots manipulateurs industriels qui évoluent dans des environnements protégés et très structurés tels que des chaînes de montage, avec le plus souvent de faibles capacités de travail. Les robots mobiles sont appelés à intervenir dans des environnements de grande taille en perpétuelle évolution, en particulier en raison de la présence humaine. Leur fonction première n'est pas de se substituer à l'être humain, mais de l'assister en réalisant certaines tâches fastidieuses, répétitives ou dangereuses à sa place. Le robot doit donc disposer d'un niveau d'autonomie suffisant pour remplir sa mission. Depuis les origines, les chercheurs progressent dans cette direction et dotent les robots mobiles d'un niveau d'autonomie de plus en plus grand.

Dès 1948, W. Grey Walter met au point des tortues électroniques autonomes dotées de capacités de décision. Ces *Machinaspeculatrix* équipées de capteurs de lumière et capables de pousser ou d'éviter d'éventuels obstacles. Le fonctionnement de ces robots, présentés en figure (I.1), est basé sur les actions-réflexes. Les machines réagissent à partir de leur connaissance partielle de l'environnement, liée à la présence ou à l'absence de lumière.



FIGURE I.1 : W. Grey Walter's tortoise
(*Machinaspeculatrix*)

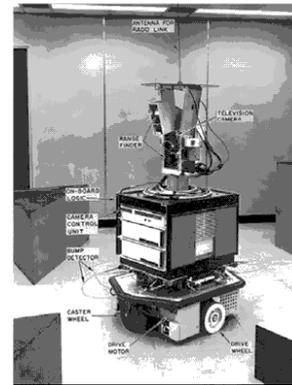


FIGURE I.2 : Robot *SHAKEY* du *Stanford Research Institute*

Ces robots étaient purement électroniques, et leur grande limitation venait de l'environnement perceptible, limité aux sources lumineuses. Ils préfiguraient déjà les petits véhicules utilisés dans le domaine du "multi-agents". Ils ont permis à leur inventeur d'obtenir les premiers résultats de coopération en robotique mobile.

Dans les années soixante, *l'Artificial Intelligence Center* (Université de Stanford) a développé le robot *SHAKEY* (Fig. I.2). Il s'agit d'un robot à deux roues différentielles équipé de caméras et de capteurs de collisions. Il est capable de planifier puis de suivre une trajectoire dans un environnement qu'il appréhende complètement et peut aussi repérer puis pousser les blocs encombrant l'environnement pour se frayer un chemin. Il s'agit du premier robot mobile autonome.

Bien souvent, quand nous parlons de robotique mobile, nous nous entendons robots mobiles à roues. Ce sont en effet les systèmes les plus étudiés, parce qu'ils sont plus simples à réaliser que les autres types de robots mobiles. Vient ensuite la robotique mobile à pattes, avec notamment la robotique humanoïde, mais également des robots avec un nombre de pattes plus élevés qui offrent de bonnes propriétés pour la locomotion en milieu difficile (milieux forestiers et agricoles). Enfin il existe également de nombreux autres types de robots mobiles (robots marins, sous marins, drones volants, micro et nano robots), généralement l'étude de ce type de robots se fait dans des thématiques spécifiques avec des problèmes particuliers à l'application visée.

Dans un robot à roues, les capacités de déplacement dépendent directement des degrés de mobilité affectés à chaque roue en contact avec le sol. Il existe différents types de roues

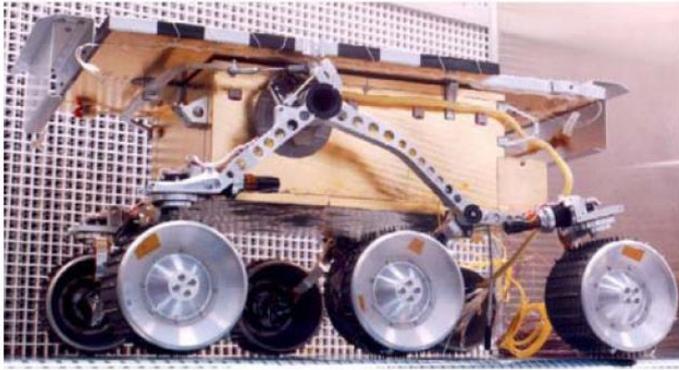
et différentes associations peuvent être réalisées. Un modèle général a été développé par G. Campion [2]. De par leurs architectures mécaniques, les mouvements des robots mobiles à roues sont souvent fortement contraints : si nous considérons l'exemple de la voiture, son orientation dans le plan d'évolution est directement liée à l'angle de braquage de ses roues avant ; les monocycles quant à eux peuvent faire des rotations sur place mais ne peuvent pas effectuer de mouvements latéraux sans introduire de glissement. Ces contraintes sont amenées par l'association de roues, chacune d'entre elles imposant un non-déplacement latéral. La structure d'un robot omnidirectionnel permet d'évoluer sur un plan horizontal et de se déplacer à tout instant dans toutes les directions, avec la possibilité supplémentaire d'effectuer un mouvement de rotation [3].

La conception du véhicule détermine les performances cinématiques et dynamiques du robot. Elle oriente fortement le choix et la conception de la commande. A titre d'exemple, un robot à roues différentielles, est non-holonôme. Son vecteur vitesse est contraint à appartenir à une certaine distribution dans le plan (sa vitesse linéaire et son orientation sont liées). La commande d'un robot non-holonôme est spécifique de sorte que les consignes de vitesse ou d'effort soient cohérentes. Un tel robot ne se commande pas de la même façon qu'un robot à roues directrices ou un robot omnidirectionnel. La commande du robot assure le suivi de la trajectoire, elle est cinématique (consignes de vitesse) ou dynamique (consignes d'accélération). Elle doit s'affranchir des incertitudes sur les modèles, c'est à dire qu'elle doit permettre au robot de suivre au mieux sa trajectoire en dépit de perturbations. La commande cinématique est la plus facile à élaborer. Le modèle cinématique est plus simple. Il ne tient pas compte de limitations telles que la saturation des couples moteurs ou de l'adhérence des roues sur le sol.

I.2 Des applications hétéroclites

I.2.1 Robotique d'exploration

Le robot *SEJOURNER* (Fig. I.3) a été utilisé par la *NASA* pour explorer sur place le sol de la planète Mars. Il s'agit d'un véhicule de petite taille pesant 11 *kg*, télé-opéré à grande distance qui a enregistré des images et analysé des échantillons de sol martien. Toutes les informations recueillies ont été transmises vers la base locale, puis vers la terre.

**FIGURE I.3** : Robot *SEJOURNER***FIGURE I.4** : Robot *NOMAD*

Parmi les robots d'exploration, nous pouvons citer le robot *NOMAD* mis au point par l'université *Carnegie Mellon de Pittsburg*. Il est destiné à l'exploration polaire ou planétaire mais contrairement au précédent c'est un véhicule de grande taille ($2,4\text{ m} \times 2,4\text{ m} \times 2,4\text{ m}$) pesant 550 kg et équipé de quatre roues. Il est capable de définir sa trajectoire de façon autonome ou d'être télé-opéré. Ce robot a connu de grands succès. Il a traversé de façon autonome une partie du désert d'Atacama.

Dans les deux exemples précédents, les objectifs de recherche étaient assez proches. Le premier véhicule a été réalisé pour l'exploration planétaire et le second a servi de plate-forme de test à terre. Aucun d'eux n'a donné lieu à une industrialisation.

I.2.2 Productions industrielles

Certaines entreprises comme *Nomadic Technologie* ou encore *Activ'Média* produisent ou ont produit des véhicules génériques équipés de capteurs et capables de réaliser certains types de travaux à l'intérieur de bâtiments. Seuls des laboratoires de recherche utilisent ces machines. Les robots *PIONNER* (Fig. I.5) d'*Activ'Media* équipent certains laboratoires travaillant sur la coopération de robots. Ils sont moins originaux mais aussi moins coûteux que le *XR4000* de *Nomadic* (Fig. I.6).

FIGURE I.5 : Robot *XR4000*FIGURE I.6 : Robot *PIONNER*

I.3 Les manipulateurs mobiles

Pour les missions en milieu hostile, spatiales, ou qui requièrent tout simplement des capacités de locomotion et de manipulation. Les plates-formes ont dû être dotées d'un bras manipulateur pour devenir des manipulateurs mobiles. Le concept de manipulateur mobile est très simple. Il s'agit d'associer, dans un même système, un ou des moyens de locomotion à un ou des moyens de manipulation. Le(s) moyen(s) de locomotion assure(nt) au système un espace de travail limité principalement par son autonomie énergétique. Le(s) moyen(s) de manipulation assure(nt) des capacités de préhension et de manipulation. Au niveau de l'espace de travail, un bras associé à une plate-forme augmente les possibilités de franchissement d'obstacles (ouverture et franchissement de portes, déplacement des obstacles); la plate-forme permet au bras d'étendre son espace de travail. Au niveau de la nature des tâches pouvant être effectuées, un bras permet de diversifier les travaux possibles, généralement limités au simple transport pour une plate-forme. Le manipulateur mobile, en procurant des degrés de mobilité supplémentaires, augmente les possibilités de manipulation du bras. Parmi les systèmes existant actuellement nous pouvons citer trois grandes familles.

I.3.1 Les multipèdes

Les manipulateurs mobiles "multipèdes" peuvent posséder une à plusieurs jambes. Par le défi à la gravité qu'ils représentent, les humanoïdes sont les plus populaires représentants. Du point de vue de la locomotion, ils posent le problème de la marche. D'un point



FIGURE I.7 : Le robot humanoïde *QRIO* **FIGURE I.8 :** Le robot humanoïde *ASIMO*

de vue de la manipulation, le souhait de bio-mimétisme conduit à les équiper de mains sophistiquées dont la conception et la commande restent des sujets encore très ouverts. Leurs utilisations possibles sont limitées d'un point de vue industriel et leur principal débouché se situe dans la robotique de service (assistance aux personnes âgées, compagnons domestiques) tels le robot *QRIO*¹ de *SONY* (Fig. I.7) et le robot *HONDA ASIMO*² (Fig. I.8).

I.3.2 Les manipulateurs mobiles sous-marins

Les manipulateurs mobiles sous-marins sont aujourd'hui les manipulateurs mobiles les plus utilisés. Souvent télé-opérés, il s'agit alors de ROV (*Remotely Operated Vehicles*), ils permettent l'accès à des zones maritimes non accessibles aux plongeurs et fournissent des capacités de prélèvement, de manipulation, de mesure et d'acquisition de données adaptables aux missions envisagées. Nous pouvons, à titre d'exemple, citer le robot *VICTOR 6000*³ (Fig. I.9) de l'Institut Français de Recherche pour l'Exploitation de la Mer, équipé de deux bras manipulateurs et qui peut intervenir à des profondeurs de 6000 mètres ainsi que le robot *ACHILLE 2000* (Fig. I.10) de la société *COMEX*⁴, lui aussi équipé de deux bras manipulateurs et dont la profondeur principale d'intervention est de 2000 mètres.

1. <http://www.sony.net/SonyInfo/QRIO>

2. <http://world.honda.com/ASIMO/>

3. http://www.ifremer.fr/flotte/systemes_sm/engins/victor.htm

4. <http://www.comex.fr/suite/dom/index2.htm>



FIGURE I.9 : Le robot *VICTOR 6000* FIGURE I.10 : Le robot *ACHILLE 2000*

I.3.3 Les manipulateurs mobiles à roues

Les manipulateurs mobiles à roues prennent aujourd'hui des formes diverses tant du point de vue de leur mode de locomotion que de leur taille ou encore de la fonction attribuée au(x) bras manipulateur(s) dans le système. Les différences majeures entre ces systèmes à roues sont principalement liées au choix du type de locomotion. Ainsi, nous distinguons :

- Les manipulateurs mobiles à roues, tels que le robot *SPIRIT* de la *NASA*⁵ (Fig. I.11), dont la plate-forme mobile peut-être dite tout terrain ;
- Les manipulateurs mobiles à roues, tels que le robot *CARE-O-BOT*⁶ du *Fraunhofer Institut* (Fig. I.12), utilisant des mécanismes de suspension rudimentaires et plus adaptés à des environnements d'intérieur lisses et assez plats ;
- L'Institut de Robotique de Karlsruhe, en Allemagne, a développé *KAMRO*⁷ (Fig. I.13). Il s'agit d'un manipulateur mobile destiné à des tâches autonomes en environnement industriel. Il est constitué d'une plate-forme omnidirectionnelle permettant des mouvements dans des espaces étroits et de deux bras *PUMA 200* de type 6R. *KAMRO* est équipé de capteurs d'efforts six axes pour la manipulation d'un système de vision pour la reconnaissance d'objets et de capteurs à ultrasons pour la navigation [4, 5] ;
- Le LAAS-CNRS (Toulouse, France) dispose du manipulateur mobile *H2BIS*⁸ (Fig.

5. <http://marsrovers.jpl.nasa.gov/home>

6. <http://www.care-o-bot.de>

7. <http://www.honda-p3.com/artificial-intelligence/the-intelligent-mobile-robot-kamro.html>

8. <http://homepages.laas.fr/matthieu/robots/h2bis.shtml>



FIGURE I.11 : Le manipulateur mobile *SPIRIT*



FIGURE I.12 : Le manipulateur mobile *CARE-O-BOT*

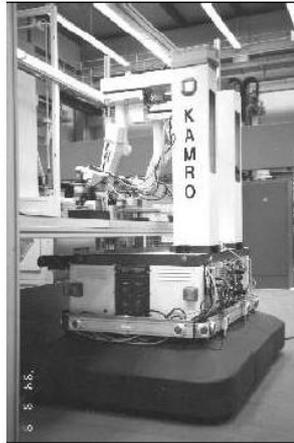


FIGURE I.13 : Le manipulateur mobile *KAMRO*

- I.14) constitué d'une plate-forme non-holonôme de type *HILARE* à deux roues motrices indépendantes portant un bras *ROBOSOFT* de type 6R. Ce robot est notamment utilisé dans l'étude de la coordination plate-forme/bras [6];
- Le robot *ANIS*⁹ développé à l'Institut National de Recherche en Informatique et en Automatique (Sophia-Antipolis, France) dispose d'une caméra sur son organe terminal (Fig. I.15) [7];
 - Nous pouvons citer aussi le robot *ARMAR* [8], *ROBONAUT* [9], *HERMES* [10], *HADALY-2* [11], le robot mobile *HELPER* [12], *SAIKA* [13], etc.

9. <http://www-sop.inria.fr/icare/WEB/Applications/base.html>

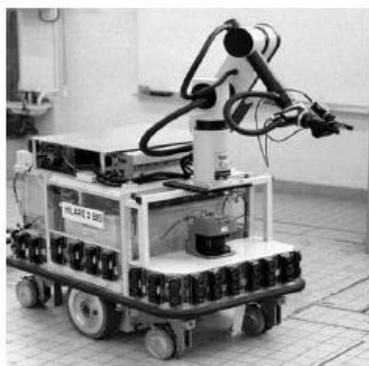


FIGURE I.14 : Le manipulateur mobile *H2BIS*

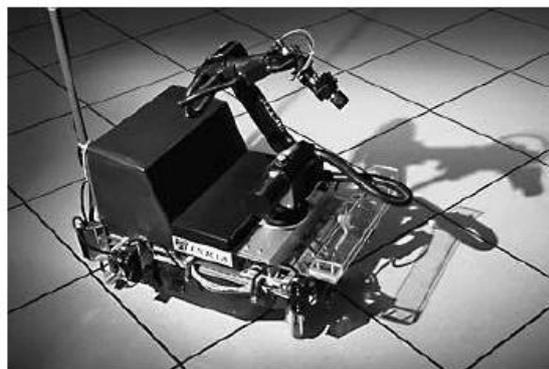


FIGURE I.15 : Le manipulateur mobile *ANIS*

I.4 Les systèmes multi-agents

I.4.1 Problématique

I.4.1.1 Cadre général

Contrairement à l'Intelligence Artificielle (IA) [14], qui modélise le comportement par une centralisation de l'expertise, l'Intelligence Artificielle Distribuée (IAD) s'intéresse à des comportements intelligents qui sont le résultat de la coopération de plusieurs entités appelées agents. L'IAD s'appuie donc sur la distribution de l'intelligence entre agents formant une société (le système multi-agents) dans lequel chacun possède une autonomie et travaille d'une façon concurrente. L'étude des systèmes multi-agents représente alors un projet scientifique dont l'objectif est d'étudier et de concevoir des organisations d'agents.

Cette thèse se situe à la rencontre de deux domaines de recherche : l'IAD et la robotique mobile. Notre étude concerne le problème de la coopération entre plusieurs entités autonomes (ou agents), c'est-à-dire la recherche d'interactions intelligentes permettant aux agents d'effectuer efficacement leurs tâches. C'est un problème central dans le domaine des systèmes multi-agents (SMA) réels ou simulés.

Historiquement, nous pouvons dire que l'approche multi-agents a émergé de celle de l'IA vers le milieu des années 80. Elle concerne l'étude et la conception de systèmes distribués faisant interagir plusieurs agents non supervisés. Il s'agit bien d'un changement radical dans l'approche du traitement des problèmes. La vision mono-agent ou celle d'un

système réparti supervisé, trop monolithique, est rejetée. L'approche multi-agents, qui apparaît sous l'essor de la programmation par objets, du parallélisme et des réseaux de communication s'intéresse aux fonctionnements des systèmes distribués à la fois au niveau global, au niveau des interactions et au niveau de chaque agent.

Cette nouvelle vision informatique des systèmes, de problèmes et de leur résolution couvre un champ très vaste. Les grands domaines d'application sont :

- La résolution de problèmes.
- La simulation multi-agents.
- La robotique distribuée (robotique mobile et robotique cellulaire).
- Les agents communicants (web).

Du côté de la robotique, à la même période, apparaît le besoin de concevoir des systèmes robotisés plus robustes et capables d'une grande adaptabilité face aux variations de l'environnement [15, 16, 17, 18, 19, 20, 21, 22, 23].

Mais au-delà des ces grands domaines d'applications, les systèmes multi-agents traitent des problèmes classiques en intelligence artificielle distribuée :

- Comment concevoir une entité artificielle capable de percevoir et d'agir sur son environnement.
- Comment faire interagir, coopérer, des agents autonomes évoluant dans le même univers.
- Comment leur faire résoudre des problèmes distribués ou dynamiques.
- Comment les faire communiquer.

Il s'agit alors de problèmes distribués nécessitant l'emploi de plusieurs agents situés dans un environnement. Les agents ne sont pas en compétition mais doivent coopérer. Notre étude se concentre sur le problème très général de la coopération entre agents situés (robots) autonomes. Il s'agit d'intégrer cette notion au cœur des agents participants à la résolution de problèmes distribués. La coopération est une des caractéristiques les plus intéressantes des systèmes multi-agents. En effet, un système n'est performant que si les agents sont capables de s'organiser.

Les premiers systèmes coopératifs ont été conçus sur la base de communications de haut niveau. Les échanges de messages effectués par des agents "très experts" ayant une connaissance importante de leur univers, des problèmes à traiter et des méthodes de

résolutions (agents dits cognitifs). En effet, ces agents utilisent des processus de négociation/dialogue pour réaliser une éventuelle coopération ou la résolution d'un conflit. Ces dialogues, nombreux et complexes, nécessitent des capacités cognitives évoluées.

I.4.1.2 L'intelligence collective

A la fin des années 80, une nouvelle voie d'exploration est apparue en IAD. En effet, ce sont des systèmes composés d'agents très simples (réactifs) qui peuvent produire des constructions complexes et des solutions à des problèmes (parcours optimaux, répartition des tâches). Les informations ont repris les principes d'auto-organisation et d'émergence présents dans ces sociétés pour définir ce que l'on nomme l'intelligence collective. La coopération se présente à travers l'auto-organisation des agents. Ce sont des comportements collectifs intelligents qui émergent à partir des nombreuses interactions entre les agents et leur environnement.

La coopération peut être définie comme le comportement intentionnel d'un ou de plusieurs agents s'engageant dans des actions communes. Ce processus est défini en trois phases :

- Identifier un but commun.
- Adopter un but commun.
- Effectuer une action commune.

Ces principes ont été repris pour la conception de robots autonomes. La plupart des recherches en SMA appliquées à la robotique portent sur les applications multi-robots. Cette approche est une approche collective et coopérative de la robotique pour la réalisation d'une mission.

Le transport collectif est une application parmi plusieurs de la robotique collective. La tâche de transport collectif représente également un problème classique des applications multi-robots. L'objectif est de transporter un ou plusieurs objets lourds, un robot ne peut réaliser la mission de façon individuelle ce qui nécessite une collaboration implicite ou explicite entre les différents robots. Ce type d'application pose donc le problème de la coordination collective.

Dans ce travail, nous allons présenter l'étude d'une tâche coopérative par un système multi-agents. Le système constitué d'un ensemble de manipulateurs mobiles, chacun peut

communiquer avec les autres afin de réaliser la tâche demandée, cette dernière réside dans la coopération entre deux manipulateurs mobiles ou plus dans une manipulation continue d'un objet.

I.4.2 Travaux existants

En simulation : O. Khatib [24] (navigation des robots autonomes, planification des mouvements) et R.C. Arkin [15] (coopération émergente pour le transport),

Avec des robots réels : M. Mataric [17] (groupes de robots, application du modèle *schema-based behaviors* à des tâches spatiales et de déplacements en formation), Arkin et Balch [25], Balch et Arkin [26] (coordination pour le maintien en formation d'équipes de robots).

Equipes de robots footballeurs : M. Veloso et P. Stone proposent depuis le milieu des années 90 des architectures de robots footballeurs très performants. En 1998, ils ont introduit un algorithme de sélection d'action permettant aux équipiers non actifs (sans balle) de se positionner pour recevoir une éventuelle passe. Ce processus, basé sur la combinaison d'attractions (du porteur de la balle et du but adverse) et de répulsions (des adversaires et des équipiers), a rendu leur équipe plus performante. Leurs derniers travaux généralisent cette approche par l'emploi de champs de potentiels pour le contrôle de chaque agent et de l'équipe (maintien en formation et éloignement des bords du terrain ou de "lignes arrières" sont combinés avec les contraintes individuelles).

Coopération entre les robots : O. Khatib *et al* [27, 28], H. Osumi [29], K. Kosuge et T. Oosumi [30], Y. Kume *et al* [31], Y. Hirata *et al* [32], J. Desai *et al* [33], M. Abou-Samah [34], M. Fang *et al* [35], H. Su [36], O. Simonin [37, 38], P. Lucidarme *et al* [39, 40], H. Yamaguchi [41], C. Y. Uny *et al* [42], R. Fierro *et al* [43], J. Sequeira [44], etc.

I.5 Description de la méthode proposée

La structure du robot est contrôlée en utilisant une conception d'un système distribué multi-tâches. La conception de la structure du robot doit intégrer de multiples fonctionnalités comme la navigation, la manipulation et l'interaction avec l'environnement. Le robot doit éviter les obstacles statiques/dynamiques ainsi que la collaboration avec

d'autres robots. Cela nécessite différents types de capteurs tels que des capteurs à ultrasons et d'effort. Ces comportements doivent être organisés afin de convenir à une variété de tâches. La structure multi-agents offre une approche appropriée privilégiant à la fois la distribution de décision, l'autonomie et la réactivité des agents, ainsi que la coordination des actions en cas de conflits. Le robot est autonome et peut percevoir l'environnement et communiquer avec les autres afin de synchroniser sa réponse en fonction de la tâche requise. Dans notre proposition un manipulateur mobile représente un agent. Sa structure est organisée comme dans la figure (I.16) [45, 46].

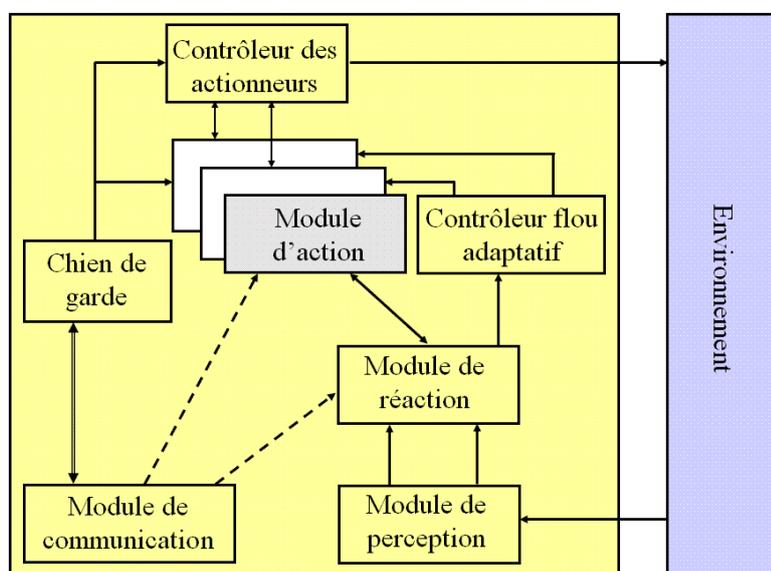


FIGURE I.16 : Architecture d'un robot

I.5.1 Description de l'architecture d'un robot

L'architecture adoptée permet de doter nos différents robots de modules pour la navigation, la prise de décision, l'adaptation, etc. Cette structure est composée de (Fig. I.16) :

- **Module d'action** : Les capacités de raisonnement de l'agent constituent le centre de décision des différentes compétences de l'agent. En fonction du comportement déclenché, un module d'action associé à un comportement donné détermine une réponse adaptée à la situation dans laquelle se trouve l'agent. Le module d'action assure la gestion de la sélection de la loi de commande. Ainsi, nous avons défini

une organisation générique d'une structure multi-couches. Chaque couche fournit un contrôleur spécifique. Au moyen de ce module, l'agent peut choisir des stratégies différentes (navigation libre, évitement d'obstacles, etc.) ou d'alterner les rôles dans le cas d'une coopération de plusieurs agents. Ce module sera détaillé dans les chapitres 3 et 4.

- **Module de communication** : L'objectif de ce module est double, il permet aux agents de communiquer et donner à l'opérateur humain toutes les informations nécessaires pour surveiller l'état de l'agent.
- **Module de perception** : Il rassemble tous les échanges de données avec le module de réaction de l'agent. A un temps d'échantillonnage fixe, le module mesure et transmet les informations nécessaires sur la position, la vitesse, etc, au module de réaction de l'agent. Ces informations sont utilisées ensuite par les autres modules.
- **Contrôleur flou adaptatif** : Ce module est destiné à régler les paramètres de l'impédance utilisés pour contrôler le déplacement de l'agent. Ce module est détaillé dans le chapitre 5.
- **Chien de garde** : Pour assurer la sécurité de l'agent, ce module détecte tous les dysfonctionnements comme les erreurs de communication durant la réalisation d'une tâche. Il alerte/bloque le robot.
- **Module de réaction** : Ce module comprend toutes les informations du module de perception et du module de communication. Il gère la collection des données et leur stockage dans la mémoire du contrôleur.
- **Contrôleur des actionneurs** : Ce module se charge d'envoyer aux actionneurs les vitesses désirées courantes. Il fournit le contrôle de l'agent en fonction du mode choisi désigné par le module d'action.

Tous les modules décrits sont utilisés pour contrôler le comportement du robot.

I.6 Organisation de la thèse

L'objectif principal de cette thèse est de permettre à un groupe de robots mobiles de travailler ensemble pour transporter un objet physique. Pour y arriver nous allons réaliser une architecture de commande pour permettre à ces robots de coopérer dans

un environnement avec obstacles. Puisqu'il n'est pas possible de tout prévoir, nous nous devons de doter nos robots d'une architecture afin que chaque robot puisse apprendre et décider tout seul. Nous avons organisé cette thèse en sept chapitres.

Chapitre 1 - Le premier chapitre est le chapitre en cours. Il constitue une introduction générale du contexte et les objectifs de la thèse.

Chapitre 2 - Modélisation - Nous présentons dans ce chapitre la modélisation géométrique, cinématique et dynamique des manipulateurs mobiles à roues. Nous nous intéressons plus particulièrement au cours de notre travail à l'étude d'une base mobile omnidirectionnelle. Afin de prendre en compte les phénomènes dynamiques, nous établissons le modèle dynamique de la plate-forme omnidirectionnelle *ROMNI* et aussi le modèle dynamique du manipulateur mobile omnidirectionnel formé de la plate-forme *ROMNI* et d'un bras manipulateur plan à deux liaisons rotoïdes. Ces développements sont précédés d'un rappel des notions de mécanique qui permettent de paramétrer rigoureusement le système. Des notions concernant les espaces des configurations et des situations, et des définitions précises aux notions de degrés de liberté et de mobilité.

Chapitre 3 - Commande du manipulateur mobile - Ce chapitre expose le contexte général de calcul d'une commande pour le manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons rotoïdes, quand un mouvement opérationnel est donné en entrée. Nous présentons la méthode basée sur la formulation de tâches additionnelles. Pour comparaison, nous présentons l'exemple d'un manipulateur mobile omnidirectionnel avec un bras manipulateur à trois liaisons rotoïdes. Pour réaliser l'objectif de l'évitement d'obstacles, nous allons proposer la stratégie du contrôle cinématique augmenté d'une boucle externe de contrôle en impédance. Nous présentons à la fin du chapitre les résultats obtenus en simulation avec des tâches simples qui nous permettent, entre autre, de faire suivre à l'organe terminal des trajectoires désirées.

Chapitre 4 - Commande par retour d'état linéarisant - Ce chapitre est essentiellement consacré à la commande dynamique de la plate-forme omnidirectionnelle et du manipulateur mobile omnidirectionnel. Après un rappel de la problématique de la commande par retour d'état, nous présentons le concept de base de notre approche de commande. Nous ajoutons ensuite la stratégie du contrôle dynamique augmenté d'une boucle externe de contrôle en impédance pour l'évitement d'obstacles.

Chapitre 5 - Adaptation des paramètres de l'impédance & identification des paramètres dynamiques - Nous introduisons dans ce chapitre un algorithme adaptatif flou comme une solution de contrôle intelligent pour choisir le comportement désiré de l'impédance. Ce chapitre aussi traite de l'identification des paramètres dynamiques du manipulateur mobile omnidirectionnel. La méthode d'identification par moindres carrés récurrents est présentée.

Chapitre 6 - Les systèmes multi-agents - Ce chapitre est consacré à l'accomplissement d'une tâche coopérative par un système multi-agents. Le système est constitué d'un ensemble de manipulateurs mobiles, chacun peut communiquer avec les autres afin de réaliser la tâche demandée. Cette dernière réside dans la coopération entre deux manipulateurs mobiles ou plus dans une manipulation continue de portage. Le chapitre se décompose en deux sections traitant d'une part des systèmes multi-agents et de la robotique collective, d'autre part de la méthode proposée.

Chapitre 7 - Conclusion générale et perspectives -

Chapitre II

Modélisation

II.1 Introduction

Quel que soit l'objectif que l'on se fixe, concevoir, simuler ou commander un système mécanique, nécessite, de disposer de modèles du comportement. Plusieurs niveaux de modélisation sont possibles : modèles géométriques, cinématiques ou dynamiques.

Dans ce chapitre, nous effectuons la modélisation cinématique et la modélisation dynamique du manipulateur mobile omnidirectionnel. L'exemple choisi pour illustrer la modélisation du manipulateur mobile est celui d'un robot dont la plate-forme est de type omnidirectionnel et qui porte un bras manipulateur de type double pendule horizontal. Nous donnons tout d'abord des rappels sur les modèles que nous avons choisis pour le bras manipulateur et pour la plate-forme mobile à roue.

Pour éviter la confusion entre les grandeurs liées au bras manipulateur et celles liées à la plate-forme mobile, nous introduisons les indices ". r " et ". p " pour désigner leurs grandeurs respectives. Pour la modélisation du système mécanique, nous avons besoin de définir quatre principaux repères.

- un repère fixe du monde, noté $\mathcal{R}(O, \vec{x}, \vec{y}, \vec{z})$, dont l'axe \vec{z} est vertical.
- un repère mobile lié au centre de la plate-forme, noté $\mathcal{R}'(O', \vec{x}', \vec{y}', \vec{z}')$.
- un repère lié à la base du bras manipulateur, noté $\mathcal{R}_{r_0}(O_{r_0}, \vec{x}_{r_0}, \vec{y}_{r_0}, \vec{z}_{r_0})$.
- un repère lié à l'organe terminal, noté $\mathcal{R}_{r_{nr}}(O_{r_{nr}}, \vec{x}_{r_{nr}}, \vec{y}_{r_{nr}}, \vec{z}_{r_{nr}})$.

II.2 Modélisation du bras manipulateur

Soit un bras manipulateur constitué de n_r corps mobiles supposés parfaitement rigides et reliés entre eux par n_r liaisons rotoïdes et/ou prismatiques. La technique la plus répandue pour décrire la géométrie d'un bras manipulateur consiste à utiliser les paramètres de Denavit-Hartenberg modifiés. Nous associons au i -ème corps le repère $\mathcal{R}_{r_i}(O_{r_i}, \vec{x}_{r_i}, \vec{y}_{r_i}, \vec{z}_{r_i})$, avec $i = 0, 1, \dots, n_r$ [47]. Le point $O_{r_{n_r}}$ est le centre de l'OT du bras manipulateur. Les paramètres de Denavit-Hartenberg modifiés permettent de définir le positionnement des corps et liaisons, donc la géométrie du bras manipulateur.

II.2.1 Espace articulaire, espace opérationnel et modèle de transformation entre les espaces

L'espace de configuration articulaire d'un bras manipulateur est l'état du robot représentant la situation de ses différents corps [48]. La configuration est définie par un vecteur q_r de n_r coordonnées indépendantes, appelées coordonnées généralisées. L'espace de ces variables noté $\mathcal{N}_{\mathcal{R}}$ est appelé espace articulaire ou espace des configurations, n_r est appelé indice de mobilité. Les coordonnées généralisées correspondent aux grandeurs caractéristiques des différentes articulations : angles de rotation pour les liaisons rotoïdes, translations pour les liaisons prismatiques. Nous avons donc :

$$q_r = \begin{bmatrix} q_{r_1} \\ q_{r_2} \\ \cdot \\ \cdot \\ q_{r_{n_r}} \end{bmatrix}$$

L'espace opérationnel est celui dans lequel est représentée la situation de l'organe terminal. Les variables permettant de définir la situation de l'OT sont appelées les coordonnées opérationnelles. La situation ξ_r de l'OT du bras manipulateur est alors définie localement par m_r coordonnées opérationnelles, qui donne la position et l'orientation de l'OT dans \mathcal{R}_{r_0} . La situation de l'OT est définie sur un espace $\mathcal{M}_{\mathcal{R}}$, de dimension m_r , appelé espace opérationnel [49]. Nous avons donc :

$$\xi_r = \begin{bmatrix} \xi_{r_1} \\ \xi_{r_2} \\ \vdots \\ \vdots \\ \xi_{r_{m_r}} \end{bmatrix}$$

Le choix le plus commun pour les paramètres de position est celui des coordonnées cartésiennes. L'expression de l'orientation de l'OT dépend du choix des paramètres angulaires [50, 51, 52].

II.2.1.1 Modèle Géométrique Direct (MGD)

Le modèle géométrique direct (MGD) d'un bras manipulateur permet d'exprimer la situation de son OT en fonction de la configuration (exprimer les coordonnées opérationnelles en fonction des coordonnées généralisées). Le MGD est l'application :

$$\begin{aligned} f_r : \mathcal{N}_{\mathcal{R}} &\rightarrow \mathcal{M}_{\mathcal{R}} \\ q_r &\mapsto \xi_r = f_r(q_r) \end{aligned} \quad (\text{II.1})$$

$$\xi_{r_i} = f_{r_i}(q_{r_j}) \quad 1 \leq i \leq m_r \text{ et } 1 \leq j \leq n_r$$

II.2.1.2 Modèle Géométrique Inverse (MGI)

Le modèle géométrique inverse (MGI) d'un bras manipulateur permet d'exprimer la configuration du robot en fonction de la situation de l'OT. C'est l'application inverse de MGD, définie par :

$$\begin{aligned} f_r^{-1} : \mathcal{M}_{\mathcal{R}} &\rightarrow \mathcal{N}_{\mathcal{R}} \\ \xi_r &\mapsto q_r = f_r^{-1}(\xi_r) \end{aligned} \quad (\text{II.2})$$

II.2.1.3 Modèle Cinématique Direct (MCD)

Le modèle cinématique direct (MCD) est le modèle qui permet d'exprimer la différentielle de la situation de l'OT en fonction de la différentielle de configuration (les vitesses opérationnelles en fonction des vitesses généralisées). En différentiant l'expression du modèle géométrique, donnée par l'équation (II.1), nous obtenons :

$$\dot{\xi}_r = \frac{\partial f_r}{\partial q_r} \dot{q}_r \quad (\text{II.3})$$

Nous posons $J_r(q_r) = \frac{\partial f_r}{\partial q_r}$. Cette matrice de dimension $(m_r \times n_r)$ est nommée la matrice jacobienne du bras manipulateur. Le modèle cinématique s'écrit sous la forme suivante :

$$\dot{\xi}_r = J_r(q_r)\dot{q}_r \quad (\text{II.4})$$

$$J_{r_{ij}}(q_r) = \frac{\partial f_{r_i}}{\partial q_{r_j}} \quad 1 \leq i \leq m_r \quad \text{et} \quad 1 \leq j \leq n_r$$

II.2.1.4 Modèle Cinématique Inverse (MCI)

Le modèle cinématique inverse (MCI) est l'application inverse du MCD, donnant \dot{q}_r en fonction de $\dot{\xi}_r$:

$$\dot{q}_r = [J_r(q_r)]^{-1}\dot{\xi}_r \quad (\text{II.5})$$

Selon le robot et la tâche considérée, cette opération admet une solution unique ou une infinité de solutions.

Nous définissons le vecteur des commandes cinématiques appliquées au bras manipulateur par :

$$u_r = \dot{q}_r \quad (\text{II.6})$$

II.2.2 Redondance et configurations singulières

Convenons de noter $d_r(q_r)$ la valeur que prend le rang de $J_r(q_r)$ dans la configuration q_r et d'appeler cette grandeur degré de liberté local de l'OT. D_r est le degré de liberté global de l'OT.

$$\begin{aligned} d_r(q_r) &= \text{rang } J_r(q_r) \\ D_r &= \max_{q_r \in \mathcal{N}_{\mathcal{R}}} d_r(q_r) \end{aligned} \quad (\text{II.7})$$

Définition II.1. *Un bras manipulateur est redondant de degré $n_r - D_r$ lorsque le degré de liberté global D_r de son organe terminal est inférieur strictement à son indice de mobilité n_r , ($D_r < n_r$). Dans ce cas, à une même situation de l'OT est associée une infinité de configurations de dimension $n_r - D_r$. Lorsque $D_r = n_r$, un nombre fini de configurations du bras manipulateur correspond à une même situation de l'OT.*

Définition II.2. *Une configuration q_r est dite singulière si $d_r(q_r) < D_r$. L'ordre de singularité est $D_r - d_r(q_r)$. Une configuration non singulière est dite régulière.*

II.2.3 Modélisation dynamique du bras manipulateur

L'étude dynamique consiste à déterminer les forces et les couples à appliquer aux articulations afin d'obtenir les configurations désirées. En ce qui concerne les bras manipulateurs, des méthodes systématiques de calcul existent et peuvent être trouvées dans la plupart des ouvrages de robotique classique [53, 54, 50]. Deux approches sont souvent utilisées pour la modélisation dynamique des robots manipulateurs. L'approche de Newton-Euler offre la possibilité de calculer directement les forces généralisées sans l'intermédiaire des équations littérales. Une méthode standard pour développer les équations dynamiques d'un système mécanique est l'utilisation des équations d'Euler-Lagrange.

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_r} - \frac{\partial L}{\partial q_r} = \tau_r \quad (\text{II.8})$$

où $q_r = [q_{r_1}, \dots, q_{r_{n_r}}]^T$ est le vecteur des coordonnées généralisées du système, L est le Lagrangien représentant la différence entre l'énergie cinétique K et l'énergie potentielle P et $\tau_r = [\tau_{r_1}, \dots, \tau_{r_{n_r}}]^T$ est le vecteur des forces généralisées agissant sur le système.

L'énergie cinétique est une fonction quadratique de la forme :

$$K = \frac{1}{2} \sum_{i,j}^{n_r} M_r^{(ij)}(q_r) \dot{q}_{r_i} \dot{q}_{r_j} = \frac{1}{2} \dot{q}_r^T M_r(q_r) \dot{q}_r \quad (\text{II.9})$$

La matrice d'inertie $M_r(q_r)$ de dimension $n_r \times n_r$ possède la propriété d'être symétrique et définie positive pour tout $q_r \in \mathcal{R}^{n_r}$. Nous remarquons que pour un robot manipulateur que l'énergie potentielle P est indépendante de \dot{q}_r . Nous déterminons alors les équations d'Euler-Lagrange en formulant L , tel que :

$$L = K - P = \frac{1}{2} \sum_{i,j}^{n_r} M_r^{(ij)}(q_r) \dot{q}_{r_i} \dot{q}_{r_j} - P \quad (\text{II.10})$$

II.2.4 Bras manipulateur plan à deux liaisons rotoïdes

Le système utilisé dans cette thèse est un bras manipulateur constitué de deux corps, de longueurs l_1 et l_2 et de masses m_1 et m_2 (Fig. II.1) articulés autour de deux liaisons rotoïdes d'axes verticaux parallèles. O_{r_0} est placé à la base du bras manipulateur et O_{r_2} est le centre de l'OT. La configuration du bras manipulateur est $q_r = [q_{r_1}, q_{r_2}]^T$ de dimension $n_r = 2$ et la situation de l'organe terminal est définie par la position du point O_{r_2} dans le

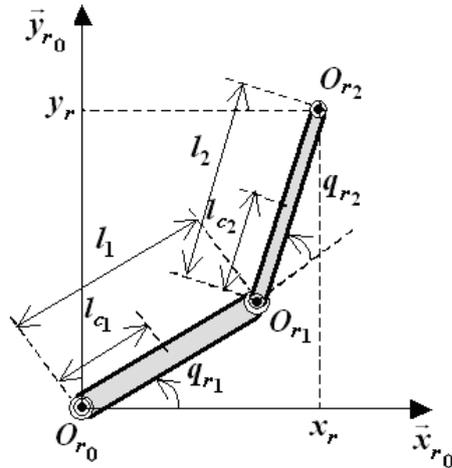


FIGURE II.1 : Un bras manipulateur plan à deux liaisons rotoïdes

repère \mathcal{R}_{r_0} , soit $\xi_r = [x_r, y_r]^T$ de dimension $m_r = 2$. Le modèle géométrique direct (MGD) du bras manipulateur est :

$$\begin{cases} x_r = l_1 \cos(q_{r_1}) + l_2 \cos(q_{r_1} + q_{r_2}) \\ y_r = l_1 \sin(q_{r_1}) + l_2 \sin(q_{r_1} + q_{r_2}) \end{cases} \quad (\text{II.11})$$

En dérivant les équations (II.11), nous trouvons le MCD du bras manipulateur.

$$\begin{cases} \dot{x}_r = -(l_1 \sin(q_{r_1}) + l_2 \sin(q_{r_1} + q_{r_2})) \dot{q}_{r_1} - l_2 \sin(q_{r_1} + q_{r_2}) \dot{q}_{r_2} \\ \dot{y}_r = (l_1 \cos(q_{r_1}) + l_2 \cos(q_{r_1} + q_{r_2})) \dot{q}_{r_1} + l_2 \cos(q_{r_1} + q_{r_2}) \dot{q}_{r_2} \end{cases} \quad (\text{II.12})$$

sous forme matricielle,

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} -(l_1 \sin(q_{r_1}) + l_2 \sin(q_{r_1} + q_{r_2})) & -l_2 \sin(q_{r_1} + q_{r_2}) \\ (l_1 \cos(q_{r_1}) + l_2 \cos(q_{r_1} + q_{r_2})) & l_2 \cos(q_{r_1} + q_{r_2}) \end{bmatrix} \begin{bmatrix} \dot{q}_{r_1} \\ \dot{q}_{r_2} \end{bmatrix} \quad (\text{II.13})$$

Le déterminant de la matrice jacobienne $J_r(q_r)$ vaut :

$$\det J_r(q_r) = l_1 l_2 \sin(q_{r_2})$$

Le degré de liberté local $d_r(q_r) = 2$ sauf pour les configurations singulières $q_{r_2} = 0 \text{ mod } \pi$, valeurs de q_{r_2} pour lesquelles la configuration est singulière. En dehors de ces configurations le degré de liberté local $d_r(q_r) = 2$. Le degré de liberté global $D_r = 2$. Le bras manipulateur, qui possède deux actionneurs, n'est pas redondant vis-à-vis de la tâche en positionnement.

L'équation dynamique du manipulateur utilisant le formalisme de Lagrange est donnée par :

$$M_r(q_r)\dot{\omega}_r + C_{r_1}(q_r, \omega_r)\omega_r = \tau_r \quad (\text{II.14})$$

La matrice d'inertie $M_r(q_r)$ de dimension $n_r \times n_r$ est symétrique et définie positive pour tout $q_r \in R^{n_r}$, $C_{r_1}(q_r, \omega_r)$ est la matrice des effets centrifuges et de Coriolis. Les entrées τ_{r_i} sont les couples appliqués à chaque actionneur du bras manipulateur. L'équation dynamique du robot manipulateur plan est obtenu en utilisant la méthode d'Euler-Lagrange qui donne le résultat suivant :

$$\begin{bmatrix} M_r^{(11)} & M_r^{(12)} \\ M_r^{(21)} & M_r^{(22)} \end{bmatrix} \begin{bmatrix} \dot{\omega}_{r_1} \\ \dot{\omega}_{r_2} \end{bmatrix} + \begin{bmatrix} h\omega_{r_2} & h(\omega_{r_1} + \omega_{r_2}) \\ -h\omega_{r_1} & 0 \end{bmatrix} \begin{bmatrix} \omega_{r_1} \\ \omega_{r_2} \end{bmatrix} = \begin{bmatrix} \tau_{r_1} \\ \tau_{r_2} \end{bmatrix} \quad (\text{II.15})$$

$$\omega_r = [\omega_{r_1}, \omega_{r_2}]^T = \dot{q}_r = [\dot{q}_{r_1}, \dot{q}_{r_2}]^T, \tau_r = [\tau_{r_1}, \tau_{r_2}]^T$$

$$M_r^{(11)} = m_1 l_{c_1}^2 + m_2 (l_1^2 + l_{c_2}^2 + 2l_1 l_{c_2} \cos(q_{r_2})) + I_1 + I_2$$

$$M_r^{(12)} = M_r^{(21)} = m_2 (l_{c_2}^2 + l_1 l_{c_2} \cos(q_{r_2})) + I_2$$

$$M_r^{(22)} = m_2 l_{c_2}^2 + I_2$$

$$h = -m_2 l_1 l_{c_2} \sin(q_{r_2})$$

m_i désigne la masse du corps i , l_i désigne la longueur du corps i , l_{c_i} indique la distance de l'articulation antérieure au centre de gravité du corps i et I_i désigne le moment d'inertie du corps i . Maintenant que nous avons modélisé le bras manipulateur passons à la plate-forme.

II.3 Modélisation des plates-formes mobiles à roues

II.3.1 Robot mobile à roues

Lors du déplacement d'un robot mobile, plusieurs contraintes apparaissent qui ont pour conséquence de limiter le domaine d'évolution de celui-ci. Dans un robot mobile à roues, les capacités de déplacement dépendent directement des degrés de mobilité affectés à chaque roue en contact avec le sol. Il existe différents types de roues et différentes associations peuvent être réalisées. Un modèle général a été développé par G. Campion [2], celui-ci permet d'obtenir un modèle cinématique pour tout robot mobile à roues évoluant dans un plan.

La configuration d'une plate-forme mobile à roues est connue lorsque sa situation dans le repère \mathcal{R} est connue et que la configuration de chacune de ses roues est connue. Nous supposons que la plate-forme mobile évolue sur une surface plane. Notons $\mathcal{R}'(O', \vec{x}', \vec{y}', \vec{z}')$ un repère mobile lié à cette plate-forme. L'état du robot est donné par la position du point de référence O' qui est situé au centre de l'axe des roues motrices et par son orientation ϑ dans un repère fixe $\mathcal{R}(O, \vec{x}, \vec{y}, \vec{z})$. Nous appelons situation de la plate-forme le vecteur :

$$\xi_p = \begin{bmatrix} x \\ y \\ \vartheta \end{bmatrix} \quad (\text{II.16})$$

où x et y sont respectivement l'abscisse et l'ordonnée du point O' dans \mathcal{R} et ϑ est l'angle (\vec{x}, \vec{x}') . La situation de la plate-forme mobile est définie sur l'espace \mathcal{M}_P de dimension $m_p = 3$. Nous définissons aussi la matrice de changement de base $R_p(\vartheta)$,

$$R_p(\vartheta) = \begin{bmatrix} \cos \vartheta & \sin \vartheta & 0 \\ -\sin \vartheta & \cos \vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{II.17})$$

Le choix du point O' est libre mais G. Campion *et al.* montrent dans [2] qu'un choix judicieux peut être fait en fonction du type de plate-forme mobile à roues envisagé.

II.3.2 Etude des configurations

Nous pouvons considérer en général un robot mobile équipé de N roues. Nous utilisons les 4 indices suivants [2] : f pour les roues fixes, co pour les roues centrées orientables, do pour les roues décentrées orientables et sw pour les roues suédoises. Le nombre de roues pour chaque type est N_f , N_{co} , N_{do} et N_{sw} . La configuration du robot est décrite par le vecteur ξ_p (Eq. II.16). Les vecteurs de rotation propre des différentes roues :

$$\varphi = \begin{bmatrix} \varphi_f(t) \\ \varphi_{co}(t) \\ \varphi_{do}(t) \\ \varphi_{sw}(t) \end{bmatrix} \quad (\text{II.18})$$

Les vecteurs d'orientation $\beta_{co}(t)$ pour les angles d'orientation des roues centrées orientables et $\beta_{do}(t)$ pour les angles d'orientation des roues décentrées orientables. La configu-

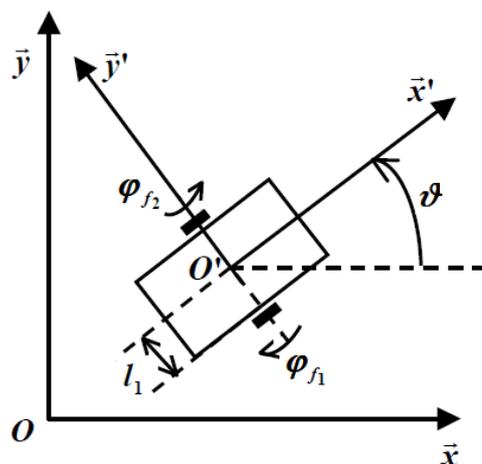


FIGURE II.2 : Plate-forme mobile de type unicycle

ration est donnée par le vecteur :

$$q_p = \begin{bmatrix} \xi_p \\ \varphi \\ \beta_{co} \\ \beta_{do} \end{bmatrix} \quad (\text{II.19})$$

II.3.2.1 Plate-forme de type unicycle

Il s'agit d'une plate-forme mobile actionnée par deux roues indépendantes et possédant un certain nombre de roues folles assurant sa stabilité. Son centre de rotation est situé sur l'axe reliant les deux roues motrices. Le schéma d'un robot de type unicycle est donné par la figure (II.2). φ_{f_1} et φ_{f_2} désignent les angles de rotation des roues droite et gauche. C'est un robot non-holonôme. En effet, il est impossible de le déplacer dans une direction perpendiculaire aux roues de locomotion. Sa commande peut être très simple, il est en effet assez facile de le déplacer d'un point à un autre par une suite de rotations simples et de lignes droites. La configuration de la plate-forme unicycle est donnée par le vecteur :

$$q_p = \begin{bmatrix} x \\ y \\ \vartheta \\ \varphi_{f_1} \\ \varphi_{f_2} \end{bmatrix} \quad (\text{II.20})$$

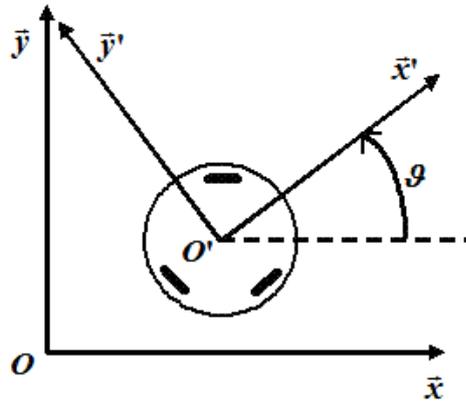


FIGURE II.3 : Plate-forme mobile de type omnidirectionnel

II.3.2.2 Plate-forme de type omnidirectionnel

Une plate-forme est dite omnidirectionnelle si l'on peut agir indépendamment sur les vitesses de translation selon les axes \vec{x} , \vec{y} et de rotation autour de \vec{z} (voir Fig. II.3). Un robot omnidirectionnel est un robot qui peut se déplacer librement dans toutes les directions. Il est en général constitué de trois roues décentrées orientables placées en triangle équilatéral. L'énorme avantage du robot omnidirectionnel est qu'il est holonôme puisqu'il peut se déplacer dans toutes les directions.

II.3.3 Modèles cinématiques des plates-formes mobiles à roues

Le modèle cinématique en situation (MCS) de la plate-forme mobile relie la dérivée de la situation de la plate-forme $\dot{\xi}_p$, dans une configuration donnée, à la commande de mobilité de cette plate-forme η_p . La dimension du vecteur de commande de mobilité de la plate-forme est appelée degré de mobilité δ_{m_p} de la plate-forme [55, 56, 2].

$$\dot{\xi}_p = R_p^T(\vartheta) \sum (\beta_{co}) \eta_p \quad (\text{II.21})$$

Nous remarquons que l'application linéaire $R_p^T(\vartheta) \sum (\beta_{co})$ dépend des paramètres d'orientation ϑ de la plate-forme et β_{co} des roues centrées orientables, lorsqu'il en existe. La commande cinématique de la plate-forme nécessite la synthèse de la commande de mobilité η_p , mais aussi de la commande de dirigeabilité ζ_p . Ces deux éléments constituent la commande de manœuvrabilité u_p de la plate-forme. Le degré de dirigeabilité de la plate-forme est nommé δ_{s_p} . Comme il existe N_{co} roues centrées orientables, si le nous voulons

les commander indépendamment, il faut que $N_{co} = \delta_{s_p}$. Posons :

$$\delta_{M_p} = \delta_{m_p} + \delta_{s_p} = \delta_{m_p} + N_{co} \quad (0 \leq \delta_{M_p} \leq \dim(\xi_p) = 3)$$

δ_{M_p} est appelé degré de manœuvrabilité de la plate-forme mobile. Le modèle cinématique en configuration (MCC) de la plate-forme mobile relie les dérivées de la configuration de la plate-forme à la commande de manœuvrabilité.

$$\dot{q}_p = S_p(\vartheta, \beta_{co}, \beta_{do})u_p \quad \text{avec} \quad u_p = [\eta_p^T \quad \zeta_p^T]^T \quad (\text{II.22})$$

La figure II.4 montre les détails des deux cas des plates-formes unicycle et omnidirectionnel [56, 2].

Plate-forme de type unicycle	Plate-forme de type omnidirectionnel
$\delta_{m_p} = 2$	$\delta_{m_p} = 3$
$\delta_{s_p} = 0$	$\delta_{s_p} = 0$
$\delta_{M_p} = 2$	$\delta_{M_p} = 3$
$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$	$\Sigma(\vartheta) = R_p(\vartheta)$

FIGURE II.4 : Modèles des deux types de plates-formes

II.3.4 Robot omnidirectionnel *ROMNI*

Un robot omnidirectionnel se déplaçant sur le sol est un robot dont la structure mécanique lui permet de changer à tout moment sa direction de déplacement, ceci sans avoir à attendre une reconfiguration de ces éléments roulants. Une des conséquences de l'omnidirectionnalité est que l'orientation propre d'un robot devient indépendante de la trajectoire à réaliser. Pour cela, chacune des roues du robot doit posséder les 3 degrés de mobilité (les 2 translations et la rotation dans \mathcal{R}^2) et permettre la motorisation d'au moins un des 2 degrés de mobilité en translation. La plate-forme *ROMNI* a une structure originale permet d'évoluer sur un plan horizontal et de se déplacer à tout instant dans toutes les directions, avec la possibilité supplémentaire d'effectuer un mouvement de rotation. La plate-forme *ROMNI*, réalisée au laboratoire *PRISME* de Bourges (Fig. II.5),

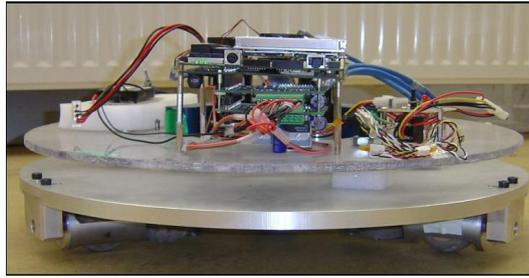
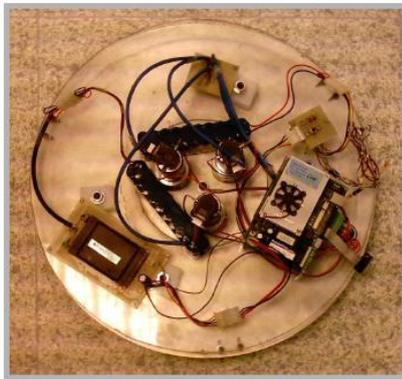
FIGURE II.5 : Le robot omnidirectionnel *ROMNI*

FIGURE II.6 : La plate-forme réelle

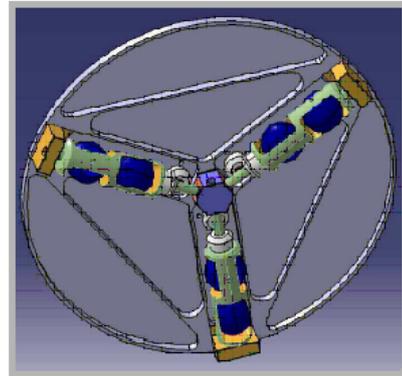


FIGURE II.7 : Disposition des trois essieux

s'inspire dans sa conception de l'utilisation des roues sphériques tronquées [57, 58]. Sur les figures (II.6) et (II.7), nous distinguons les trois moteurs d'articulation, l'informatique et l'électronique de pilotage, les batteries pour l'autonomie énergétique, ainsi que les essieux. La plate-forme *ROMNI* est constituée d'un plateau circulaire en alliage d'aluminium de diamètre de $R = 600 \text{ mm}$. Cette plate-forme circulaire est mue par trois essieux identiques disposés radicalement à 120° (Fig. II.7).

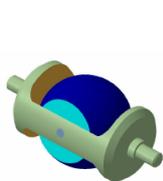


FIGURE II.8 : Demi essieu

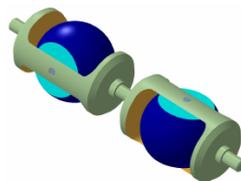


FIGURE II.9 : Essieu à sphères alignées

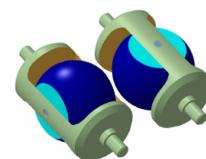


FIGURE II.10 : Essieu à sphères juxtaposées

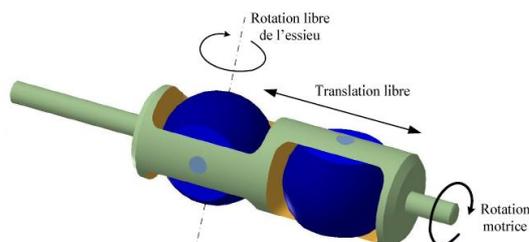


FIGURE II.11 : Représentation d'un essieu

La plate-forme est basée sur le concept d'une roue composée de deux sphères tronquées liées entre elles mécaniquement. Chaque sphère est tronquée de chaque côté par un plan. Un axe lui permet de tourner librement sur elle-même. Cet axe est maintenu par une fourchette sur les deux côtés tronqués, qui elle-même peut être mise en rotation en utilisant un moteur (Fig. II.8). Tant que la partie non tronquée de la sphère touche le sol, cette roue possède 3 degrés de mobilité : la rotation autour de l'axe libre, la rotation autour de la fourchette (pour un quart de tour seulement) et la rotation d'axe vertical autour du point de contact roue/sol. Lorsque la rotation autour de la fourchette dépasse le quart de tour, il n'y a plus de contact avec le sol ; il faut donc relayer cette sphère tronquée par une seconde sphère tronquée dont la fourchette tourne à la même vitesse que la première, avec un décalage d'un quart de tour (Fig. II.9 et Fig. II.10). La roue ainsi composée est en contact avec le sol alternativement avec l'une de ses deux sphères tronquées. Un essieu est constitué d'un corps cylindrique rigide dans lequel sont enchâssées deux sphères tronquées en liaison pivot avec l'essieu (Fig. II.11). Les axes de rotation des deux sphères sont perpendiculaires et concourants avec l'axe longitudinal de l'essieu.

Nous disposons donc d'une "roue" motorisable dans une direction et libre dans les deux autres, et qui n'interdit aucun mouvement sur le sol. Une disposition de ces trois roues sur les sommets d'un triangle équilatéral, et dont chaque axe moteur est aligné avec le centre du triangle, permet de répartir de manière homogène les trois mouvements de la plate-forme.

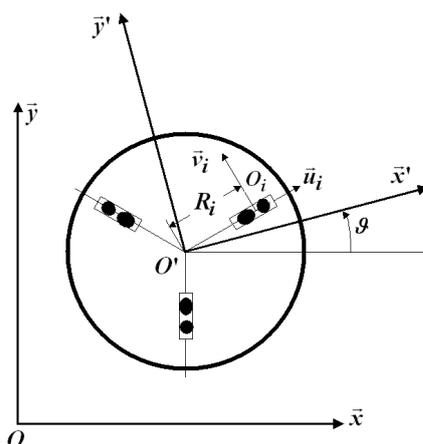


FIGURE II.12 : Repère absolu, repère robot et repère essieu

II.3.4.1 Modélisation cinématique de la plate-forme mobile omnidirectionnelle

Afin de déterminer le modèle cinématique de la plate-forme, nous calculons la vitesse \vec{V}_{O_i} du point de contact O_i dans le mouvement de l'essieu par rapport au repère absolu $\mathcal{R}(O, \vec{x}, \vec{y})$ (Fig. II.12) :

$$\vec{V}_{O_i} = \vec{V}_{O'} + \vec{\Omega} \wedge \overrightarrow{O'O_i} \quad (\text{II.23})$$

Nous utilisons les notations suivantes, avec les indices i caractérisent respectivement les essieux de 1 à 3 :

- \vec{u}_i : vecteur de direction longitudinale de l'essieu ;
- \vec{v}_i : vecteur directement perpendiculaire à \vec{u}_i ;
- O_i : centre de la sphère en contact avec le sol ;
- \vec{V}_{O_i} : vecteur vitesse du point O_i ;
- $\vec{V}_{O'}$: vecteur vitesse du point O' ;

- $\vec{\Omega}$: vitesse angulaire du point O' ;
- r : rayon d'une sphère ;
- φ_i : l'angle de rotation propre de l'essieu i ;
- $\dot{\varphi}_i$: vitesse de rotation propre de l'essieu i ;
- ϑ : l'angle compris entre \vec{x} et \vec{x}' , il caractérise l'orientation absolue du robot ;
- α_i : les angles compris entre \vec{x}' et \vec{u}_i , ils sont respectivement : $\alpha_1 = 0$, $\alpha_2 = 2\pi/3$ et $\alpha_3 = 4\pi/3$;

Pour chaque essieu, nous allons noter R_i , la distance $O'O_i$. Cette distance est variable selon la sphère en contact avec le sol. Sur un même essieu, lors d'une rotation, le point de contact avec le sol passe d'une sphère à l'autre : c'est le saut de roues. Ce dernier provoque une discontinuité des rayons R_i chaque $\frac{1}{4}$ de tour :

$$R_i = \begin{cases} R_{i_{min}} + \frac{\Delta R}{2} / 0 \leq \varphi_i < \frac{\pi}{2} \text{ et } \pi \leq \varphi_i < \frac{3\pi}{2} \\ R_{i_{max}} - \frac{\Delta R}{2} / \frac{\pi}{2} \leq \varphi_i < \pi \text{ et } \frac{3\pi}{2} \leq \varphi_i < 2\pi \end{cases} \quad (\text{II.24})$$

$\Delta R = R_{i_{max}} - R_{i_{min}}$ correspond à l'écart entre les deux sphères tronquées d'un même essieu. $R_{i_{max}}$ et $R_{i_{min}}$ caractérisant la disposition des roues sous l'essieu. Nous pouvons écrire :

$$\vec{V}_{O_i} = \dot{x}\vec{x} + \dot{y}\vec{y} + \dot{\vartheta}R_i\vec{v}_i \quad (\text{II.25})$$

\dot{x} , \dot{y} , $\dot{\vartheta}$: représentent les paramètres cinématiques du mouvement au centre O' de la plateforme, exprimés dans le repère absolu lié à l'environnement $\mathcal{R}(O, \vec{x}, \vec{y})$.

L'expression des vecteurs \vec{x} et \vec{y} dans le repère lié à l'essieu est :

$$\begin{cases} \vec{x} = \cos(\vartheta + \alpha_i)\vec{u}_i - \sin(\vartheta + \alpha_i)\vec{v}_i \\ \vec{y} = \sin(\vartheta + \alpha_i)\vec{u}_i + \cos(\vartheta + \alpha_i)\vec{v}_i \end{cases} \quad (\text{II.26})$$

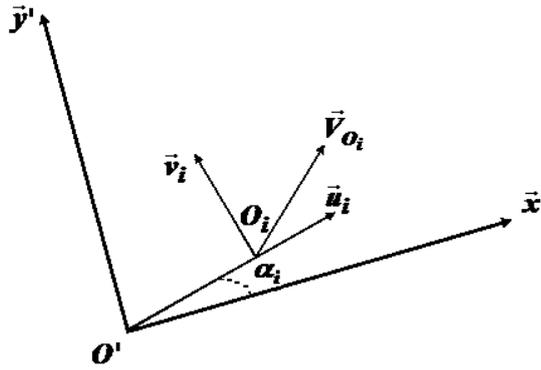


FIGURE II.13 : Repère robot et repère essieu

L'équation (II.25) peut s'écrire :

$$\vec{V}_{O_i} = (\dot{x} \cos(\vartheta + \alpha_i) + \dot{y} \sin(\vartheta + \alpha_i)) \vec{u}_i + (-\dot{x} \sin(\vartheta + \alpha_i) + \dot{y} \cos(\vartheta + \alpha_i)) \vec{v}_i + \dot{\vartheta} R_i \vec{v}_i \quad (\text{II.27})$$

Avec une roue sphérique classique et l'hypothèse de roulement sans glissement, nous obtenons que \vec{V}_{O_i} est colinéaire à \vec{v}_i : $\vec{V}_{O_i} = -r\dot{\varphi}_i \vec{v}_i$. Dans notre cas (Fig. II.13), la vitesse dans la direction \vec{u}_i est libre et non commandée par le mouvement de l'essieu, $\vec{V}_{O_i} \vec{u}_i$ est quelconque, seul $\vec{V}_{O_i} \vec{v}_i$ est commandée par le moteur. Nous pouvons écrire pour chacun des trois essieux la contrainte cinématique suivante :

$$\vec{V}_{O_i} \vec{v}_i = -r\dot{\varphi}_i \quad \forall i \in \{1, 2, 3\} \quad (\text{II.28})$$

La combinaison des équations (II.27) et (II.28) donne :

$$\begin{cases} -\dot{x} \sin(\vartheta + \alpha_1) + \dot{y} \cos(\vartheta + \alpha_1) + \dot{\vartheta} R_1 = -r\dot{\varphi}_1 \\ -\dot{x} \sin(\vartheta + \alpha_2) + \dot{y} \cos(\vartheta + \alpha_2) + \dot{\vartheta} R_2 = -r\dot{\varphi}_2 \\ -\dot{x} \sin(\vartheta + \alpha_3) + \dot{y} \cos(\vartheta + \alpha_3) + \dot{\vartheta} R_3 = -r\dot{\varphi}_3 \end{cases} \quad (\text{II.29})$$

Premièrement, considérons le vecteur ξ_p de dimension $m_p = 3$ décrivant le robot :

$$\xi_p = \begin{bmatrix} x \\ y \\ \vartheta \end{bmatrix} \quad (\text{II.30})$$

avec x et y sont les coordonnées relatives à la référence O' dans le repère fixe.

Deuxièmement, nous pouvons écrire :

$$\omega_v = \begin{bmatrix} \omega_{v_1} \\ \omega_{v_2} \\ \omega_{v_3} \end{bmatrix} = \begin{bmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \end{bmatrix} \quad (\text{II.31})$$

ω_{v_1} , ω_{v_2} et ω_{v_3} sont les vitesses angulaires du robot. Le vecteur ω_v est de dimension $m_v = 3$.

La jacobienne peut être obtenue, en fournissant une relation directe entre les paramètres cinématiques et les vitesses angulaires des roues [59, 60, 61, 57, 58].

$$\omega_v = \frac{1}{r} J_p \dot{\xi}_p \quad (\text{II.32})$$

avec $J_p = A_p R_p(\vartheta)$,

$$A_p = \begin{bmatrix} \sin \alpha_1 & -\cos \alpha_1 & -R_1 \\ \sin \alpha_2 & -\cos \alpha_2 & -R_2 \\ \sin \alpha_3 & -\cos \alpha_3 & -R_3 \end{bmatrix} = \begin{bmatrix} 0 & -1 & -R_1 \\ \sqrt{3}/2 & 1/2 & -R_2 \\ -\sqrt{3}/2 & 1/2 & -R_3 \end{bmatrix} \quad (\text{II.33})$$

et

$$R_p(\vartheta) = \begin{bmatrix} \cos \vartheta & \sin \vartheta & 0 \\ -\sin \vartheta & \cos \vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{II.34})$$

II.3.4.2 Modélisation dynamique de la plate-forme mobile omnidirectionnelle

Le couple moteur est déterminé par le calcul de toutes les forces agissant sur chaque roue. En raison de l'inertie I_R et de la masse m_R de la base, la force globale F_p doit être appliquée pour accélérer et ralentir la base. La force nécessaire peut être calculée en utilisant la deuxième loi de Newton [62].

$$F_p = D_p \ddot{\xi}_p \quad (\text{II.35})$$

où D_p est une matrice diagonale de la forme :

$$D_p = \begin{bmatrix} m_R & 0 & 0 \\ 0 & m_R & 0 \\ 0 & 0 & I_R \end{bmatrix} \quad (\text{II.36})$$

Utilisons l'équation (II.32) avec $J_p = A_p R_p(\vartheta)$, nous obtenons :

$$\ddot{\xi}_p = r \dot{R}_p^T A_p^{-1} \omega_v + r R_p^T A_p^{-1} \dot{\omega}_v \quad (\text{II.37})$$

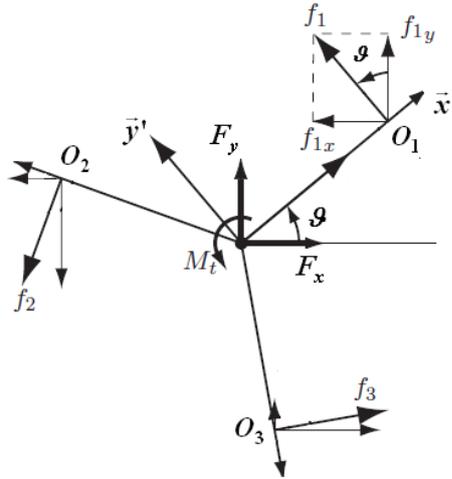


FIGURE II.14 : Diagramme dynamique de la base à trois roues

Les grandeurs des forces F_x , F_y et du moment M_t peuvent être écrites en utilisant la masse de la base m_R et le moment d'inertie I_R comme suit :

$$\begin{bmatrix} F_x \\ F_y \\ M_t \end{bmatrix} = \begin{bmatrix} m_R & 0 & 0 \\ 0 & m_R & 0 \\ 0 & 0 & I_R \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\vartheta} \end{bmatrix} \quad (\text{II.38})$$

Il est également possible d'obtenir la résultante des forces exercées par chacune des roues f_1 , f_2 et f_3 . Les forces F_x et F_y peuvent être remplacées par la somme des composantes des forces dans les directions x et y respectivement. Le moment M_t est donné par la somme des produits des forces f_1 , f_2 et f_3 et des rayons R_1 , R_2 et R_3 (Fig. II.14) [63, 64, 65] :

$$\left\{ \begin{array}{l} F_x = f_{1x} + f_{2x} + f_{3x} = \\ -f_1 \sin(\vartheta + \alpha_1) - f_2 \sin(\vartheta + \alpha_2) - f_3 \sin(\vartheta + \alpha_3) \\ F_y = f_{1y} + f_{2y} + f_{3y} = \\ +f_1 \cos(\vartheta + \alpha_1) + f_2 \cos(\vartheta + \alpha_2) + f_3 \cos(\vartheta + \alpha_3) \\ M_t = f_1 R_1 + f_2 R_2 + f_3 R_3 \end{array} \right. \quad (\text{II.39})$$

sous une forme plus compacte :

$$F_p = \begin{bmatrix} m_R \ddot{x} \\ m_R \ddot{y} \\ I_R \ddot{\vartheta} \end{bmatrix}^T = J_p^T [f_1, f_2, f_3]^T \quad (\text{II.40})$$

nous pouvons écrire :

$$[f_1, f_2, f_3]^T = J_p^{-T} \begin{bmatrix} m_R \ddot{x} \\ m_R \ddot{y} \\ I_R \ddot{\vartheta} \end{bmatrix}^T \quad (\text{II.41})$$

f_i est la force générée pour chaque roue. En remplaçant (II.37) dans (II.41), nous obtenons :

$$f = A_p^{-T} R_p D_p \left[r \dot{R}_p^T A_p^{-1} \omega_v + r R_p^T A_p^{-1} \dot{\omega}_v \right] \quad (\text{II.42})$$

La dynamique de chaque roue entraînée par un moteur à courant continu peut être décrite comme suit :

$$I_m \dot{\omega}_{m_i} + \left(\frac{C_m C_e}{R_a} + b_m \right) \omega_{m_i} - \frac{C_m}{R_a} \tau_{v_i} = -\frac{r}{n_m} f_i \quad (\text{II.43})$$

avec I_m est le moment d'inertie du moteur, ω_{m_i} est la vitesse angulaire de chaque moteur, R_a est la résistance interne du moteur, C_e est la constante de force électromotrice (EMF), C_m est la constante de couple du moteur, b_m est le coefficient du frottement visqueux du moteur, n_m est le rapport d'engrenage du moteur et τ_{v_i} est la tension appliquée [66].

Dans la forme matricielle, nous devons considérer la relation $\omega_{m_i} = n_m \omega_{v_i}$ entre la vitesse angulaire de moteur et la vitesse angulaire du robot :

$$f = -P_v \dot{\omega}_v - Q_v \omega_v + E_v \tau_v \quad (\text{II.44})$$

où

$$P_v = \frac{n_m^2}{r} I_m \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad Q_v = \frac{n_m^2}{r} \left(\frac{C_m C_e}{R_a} + b_m \right) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{et} \quad E_v = \frac{n_m C_m}{r R_a} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

En remplaçant (II.42) dans (II.44), nous pouvons écrire l'équation dynamique du robot mobile comme :

$$M_{v_1}(\xi_p) \dot{\omega}_v + C_{v_1}(\xi_p, \omega_v) \omega_v = E_v \tau_v \quad (\text{II.45})$$

avec $M_{v_1}(\xi_p) = (r A_p^{-T} R_p D_p R_p^T A_p^{-1} + P_v)$, $C_{v_1}(\xi_p, \omega_v) = (r A_p^{-T} R_p D_p \dot{R}_p^T A_p^{-1} + Q_v)$ sont les matrices d'inertie et du couplage respectivement.

Les éléments de la matrice d'inertie $M_{v_1}(\xi_p)$ sont :

$$\begin{aligned} M_{v_1}^{(11)} &= M_{v_1}^{(22)} = M_{v_1}^{(33)} = 4r I_R + \frac{4m_R r}{9} + \frac{I_m n_m^2}{r} \\ M_{v_1}^{(12)} &= M_{v_1}^{(13)} = M_{v_1}^{(21)} = M_{v_1}^{(23)} = M_{v_1}^{(31)} = M_{v_1}^{(32)} = 4r I_R - \frac{2m_R r}{9} \end{aligned}$$

Les éléments de la matrice $C_{v_1}(\xi_p, \omega_v)$ sont :

$$\begin{aligned} C_{v_1}^{(11)} &= C_{v_1}^{(22)} = C_{v_1}^{(33)} = \frac{C_m C_e n_m^2}{r R_a} + \frac{b_m n_m^2}{r} \\ C_{v_1}^{(12)} &= C_{v_1}^{(23)} = C_{v_1}^{(31)} = \frac{-2r m_R \dot{\vartheta}}{3\sqrt{3}} \\ C_{v_1}^{(13)} &= C_{v_1}^{(21)} = C_{v_1}^{(32)} = \frac{2r m_R \dot{\vartheta}}{3\sqrt{3}} \end{aligned}$$

Les éléments de la matrice E_v sont :

$$\begin{aligned} E_v^{(11)} &= E_v^{(22)} = E_v^{(33)} = \frac{C_m n_m}{r R_a} \\ E_v^{(12)} &= E_v^{(13)} = E_v^{(21)} = E_v^{(23)} = E_v^{(31)} = E_v^{(32)} = 0 \end{aligned}$$

II.4 Modélisation des manipulateurs mobiles à roues

II.4.1 Description des manipulateurs mobiles à roues

De nombreux travaux ont déjà traité de la modélisation géométrique et cinématique des manipulateurs mobiles, systèmes composés d'une plate-forme mobile porteuse d'un bras manipulateur [67, 68, 56]. Nous considérons un manipulateur mobile constitué d'une plate-forme mobile équipée d'un bras manipulateur, Le point O_{r_0} , qui est l'origine du repère \mathcal{R}_{r_0} lié à la base du bras manipulateur, a pour coordonnées $[a_1, a_2, 0]^T$ dans le repère \mathcal{R}' . La configuration d'un manipulateur mobile à roues est connue dès lors que les configurations de la plate-forme et du bras manipulateur sont connues. Elle est définie sur l'espace des configurations \mathcal{N} de dimension $n_{rp} = n_r + n_p$, où n_r et n_p sont respectivement les dimensions des espaces généralisés du bras manipulateur et de la plate-forme. La configuration est donnée par le vecteur q_{rp} des n_{rp} coordonnées généralisées :

$$q_{rp} = \begin{bmatrix} q_{rp_1} \\ q_{rp_2} \\ \cdot \\ \cdot \\ q_{rp_{n_{rp}}} \end{bmatrix} = \begin{bmatrix} q_r \\ q_p \end{bmatrix}$$

La situation de l'OT du manipulateur mobile donne la position du point $O_{r_{n_r}}$, centre de l'OT et l'orientation du repère $\mathcal{R}_{r_{n_r}}$, qui lui est lié, dans \mathcal{R} . La situation, définie sur l'espace \mathcal{M} , est donnée par le vecteur ξ des m_{rp} coordonnées opérationnelles :

$$\xi = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \cdot \\ \cdot \\ \xi_{m_{rp}} \end{bmatrix}$$

De manière générale la situation de l'OT est de dimension six : trois coordonnées donnent la position de l'OT et trois son orientation. Il peut y avoir quelques exceptions à cette règle. Dans le cas du manipulateur mobile avec un bras plan à deux liaisons rotoïdes de la figure (II.1), $m_{rp} = 3$, car nous ne tenons compte que de la position et de l'orientation dans le plan horizontal : $\xi = [\xi_1 \ \xi_2 \ \xi_3]^T$.

II.4.2 Etude géométrique des manipulateurs mobiles à roues

Le modèle géométrique direct (MGD) d'un manipulateur mobile à roues est, de manière similaire aux autres systèmes robotiques, défini comme le modèle qui permet d'exprimer la situation de l'OT en fonction de la configuration du manipulateur mobile. Le MGD est l'application :

$$\begin{aligned} f : \mathcal{N} &\rightarrow \mathcal{M} \\ q_{rp} &\mapsto \xi = f(q_{rp}) \end{aligned} \quad (\text{II.46})$$

Le modèle géométrique inverse (MGI) du manipulateur mobile est un modèle qui permet d'exprimer la configuration du manipulateur mobile en fonction de la situation de l'OT. C'est l'application inverse de MGD, définie par :

$$\begin{aligned} f^{-1} : \mathcal{M} &\rightarrow \mathcal{N} \\ \xi &\mapsto q_{rp} = f^{-1}(\xi) \end{aligned} \quad (\text{II.47})$$

Nous présentons sur la figure (II.15) les différents repères et grandeurs intervenant dans la description géométrique d'un manipulateur mobile.

Quelle que soit la plate-forme mobile à roues de situation $\xi_p = [x \ y \ \vartheta]^T$ portant un bras manipulateur de configuration q_r et dont la situation ξ_r de l'organe terminal s'exprime dans le repère $\mathcal{R}_{r_0}(O_{r_0}, \vec{x}_{r_0}, \vec{y}_{r_0}, \vec{z}_{r_0})$ via le modèle géométrique du bras manipulateur.

Les coordonnées du point O_{r_0} exprimées dans le repère \mathcal{R}' s'écrivent $O_{r_0}^{\mathcal{R}'} = [a_1, a_2, 0]^T$ et s'expriment dans \mathcal{R} via la transformation suivante :

$$\begin{bmatrix} O_{r_0}^{\mathcal{R}} \\ 1 \end{bmatrix} = \begin{bmatrix} & x \\ R_p^T(\vartheta) & y \\ & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} O_{r_0}^{\mathcal{R}'} \\ 1 \end{bmatrix} \quad (\text{II.48})$$

Les relations (II.1) et (II.48) permettent d'écrire, indépendamment du type de plate-forme mobile utilisée, le modèle géométrique direct d'un manipulateur mobile :

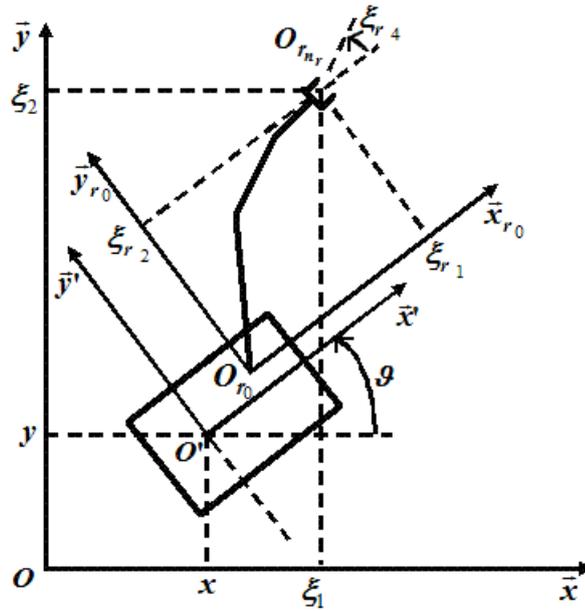


FIGURE II.15 : Géométrie des manipulateurs mobiles

$$\begin{aligned}
 f : \mathcal{N}_{\mathcal{R}} \times \mathcal{M}_{\mathcal{P}} &\rightarrow \mathcal{M} \\
 (q_r, \xi_p) &\mapsto \xi = f(q_r, \xi_p)
 \end{aligned}
 \tag{II.49}$$

La forme (II.49) du MGD indique qu'il peut être écrit en fonction de la configuration q_r du bras manipulateur et de la situation ξ_p de la plate-forme mobile dans \mathcal{R} uniquement. En détaillant chaque terme, nous obtenons :

$$\begin{aligned}
 \xi_1 &= x + (a_1 + \xi_{r_1}) \cos(\vartheta) - (a_2 + \xi_{r_2}) \sin(\vartheta) \\
 \xi_2 &= y + (a_1 + \xi_{r_1}) \sin(\vartheta) + (a_2 + \xi_{r_2}) \cos(\vartheta) \\
 \xi_3 &= \vartheta + \xi_{r_4}
 \end{aligned}
 \tag{II.50}$$

où ξ_1 et ξ_2 donnent la position de l'OT dans \mathcal{R} et ξ_3 donne l'orientation de l'OT dans \mathcal{R} .

II.4.3 Etude cinématique des manipulateurs mobiles à roues

Le modèle cinématique direct (MCD) du système est le modèle qui permet d'exprimer les différentielles des coordonnées opérationnelles de l'OT, pour une configuration donnée, en fonction des différentielles des coordonnées généralisées du système, il s'écrit :

$$\dot{\xi} = J(q_{rp}) \dot{q}_{rp}
 \tag{II.51}$$

où $J(q_{rp})$ est la matrice jacobienne de l'application $f(q_r, \xi_p)$:

$$J(q_{rp}) = \frac{\partial f}{\partial q_{rp}} \text{ soit } J_{ij}(q_{rp}) = \frac{\partial f_i}{\partial q_{rpj}} \quad 1 \leq i \leq m_{rp} \text{ et } 1 \leq j \leq n_{rp}$$

Le modèle cinématique inverse (MCI) est l'application inverse du MCD, donnant \dot{q}_{rp} en fonction de $\dot{\xi}$:

$$\dot{q}_{rp} = [J(q_{rp})]^{-1} \dot{\xi} \quad (\text{II.52})$$

La configuration partielle du manipulateur mobile peut être définie par ses coordonnées généralisées partielles [69] :

$$\tilde{q}_{rp} = \begin{bmatrix} q_r \\ \xi_p \end{bmatrix} = \begin{bmatrix} q_r \\ x \\ y \\ \vartheta \end{bmatrix}$$

Avec le choix de \tilde{q}_{rp} comme vecteur de configuration, le MCD du manipulateur mobile s'en déduit :

$$\dot{\xi} = J(\tilde{q}_{rp}) \dot{\tilde{q}}_{rp} \quad (\text{II.53})$$

où $J(\tilde{q}_{rp}) = \frac{\partial \xi}{\partial \tilde{q}_{rp}}$. Le vecteur $\dot{\tilde{q}}_{rp}$ donne les vitesses généralisées partielles.

II.4.3.1 Modèle cinématique en situation

Pour obtenir les vitesses opérationnelles du manipulateur mobile, nous dérivons par rapport au temps l'équation (II.49) du MGD du manipulateur mobile. Il vient :

$$\dot{\xi} = \frac{\partial f}{\partial q_r}(q_r, \vartheta) \dot{q}_r + \frac{\partial f}{\partial \xi_p}(q_r, \vartheta) \dot{\xi}_p \quad (\text{II.54})$$

D'après (II.6), (II.21) et (II.54) s'écrit :

$$\dot{\xi} = \frac{\partial f}{\partial q_r}(q_r, \vartheta) u_r + \frac{\partial f}{\partial \xi_p}(q_r, \vartheta) R_p^T(\vartheta) \sum (\beta_{co}) \eta_p \quad (\text{II.55})$$

Sous forme matricielle :

$$\dot{\xi} = \bar{J}(q_r, \vartheta, \beta_{co}) \eta \quad (\text{II.56})$$

où

$$\bar{J}(q_r, \vartheta, \beta_{co}) = \left[\frac{\partial f}{\partial q_r}(q_r, \vartheta) \quad \frac{\partial f}{\partial \xi_p}(q_r, \vartheta) R_p^T(\vartheta) \sum (\beta_{co}) \right]$$

L'application \bar{J} est de dimension $m_{rp} \times (n_r + \delta_{m_p})$ dépend de la configuration q_r du bras manipulateur et des paramètres d'orientation ϑ de la plate-forme et β_{co} des roues centrées orientables, lorsqu'il en existe, et :

$$\eta = \begin{bmatrix} u_r \\ \eta_p \end{bmatrix}$$

L'écriture (II.56) traduit le modèle cinématique en situation (MCS) du manipulateur mobile. η est appelée commande de mobilité du manipulateur mobile. De manière similaire à la modélisation des plates-formes mobiles, nous avons deux vecteurs : η , la commande de mobilité du manipulateur mobile de dimension $\delta_m = n_r + \delta_{m_p}$ et ζ_p la commande de dirigeabilité de la plate-forme mobile. La commande cinématique du manipulateur mobile nécessite la synthèse de la commande de mobilité η , mais aussi de la commande de dirigeabilité ζ_p . Ces deux éléments constituent la commande de manœuvrabilité u du manipulateur mobile de dimension $\delta_M = n_r + \delta_{M_p}$.

II.4.3.2 Modèle cinématique en configuration

Le modèle cinématique en configuration (MCC) du manipulateur mobile est donné par :

$$\dot{q}_{rp} = S(q_{rp})u \quad \text{avec} \quad u = [u_r^T \quad u_p^T]^T \quad \text{et} \quad \dot{q}_{rp} = [\dot{q}_r^T \quad \dot{q}_p^T]^T \quad (\text{II.57})$$

Le modèle cinématique en configuration du manipulateur mobile est obtenu par analogie du modèle cinématique en configuration de la plate-forme. Une fois le MCC de la plate-forme est connu, d'après (II.22), l'obtention du MCC du manipulateur mobile est immédiate.

II.4.4 Redondance et configurations singulières

Connaissant le MGD (II.49) d'un manipulateur mobile, nous définissons son indice de mobilité par $M_{rp} = n_r + m_p = n_r + 3$. Par ailleurs, nous définissons le degré de mobilité d'un manipulateur mobile, d'après son MCS (II.56), par $\delta_m = n_r + \delta_{m_p}$. Il est ainsi égale au nombre de paramètres de commande de mobilité du manipulateur mobile.

Nous définissons le degré de liberté local de l'OT du manipulateur mobile par :

$$d_{rp}(q_{rp}) = d_{rp}(q_r, \vartheta, \beta_{co}) = \text{rang} \bar{J}(q_r, \vartheta, \beta_{co}) \quad (\text{II.58})$$

$$D_{rp} = \max_{q_{rp} \in \mathcal{N}} d_{rp}(q_{rp})$$

D_{rp} est le degré de liberté global de l'OT.

Définition II.3. *Un manipulateur mobile est redondant de degré $\delta_m - D_{rp}$ lorsque D_{rp} est inférieur strictement au degré de mobilité δ_m du système. Dans ce cas, à une même vitesse $\dot{\xi}$ de l'OT est associée une infinité de vitesses généralisées \dot{q}_{rp} de dimension $\delta_m - D_{rp}$. Dans ce cas, une infinité de commandes de mobilité η du manipulateur mobile correspond à une même vitesse opérationnelle $\dot{\xi}$. Lorsque $\delta_m = D_{rp}$, un nombre fini de commandes de mobilité η du manipulateur mobile correspond à une même vitesse $\dot{\xi}$ de l'OT.*

Définition II.4. *Une configuration q_{rp} est dite singulière si $d_{rp}(q_{rp}) < D_{rp}$. L'ordre de singularité est $D_{rp} - d_{rp}(q_{rp})$. Une configuration q_{rp} du manipulateur mobile est donc singulière si et seulement si $\bar{J}(q_r, \vartheta, \beta_{co})$ n'est pas de rang δ_m .*

II.4.5 Modèle cinématique du manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons

Ce manipulateur mobile est composé d'une plate-forme de type omnidirectionnel (paragraphe II.3.2.2) et un bras manipulateur plan à deux liaisons (paragraphe II.2.4) (Fig. II.16). Dans le repère \mathcal{R} , la position du point O_{r_2} est donnée par ses coordonnées cartésiennes ξ_1 et ξ_2 et l'orientation de l'OT par l'angle ξ_3 . La configuration est donnée par :

$$q_{rp} = \begin{bmatrix} q_r \\ q_p \end{bmatrix} = \begin{bmatrix} q_{r1} \\ q_{r2} \\ x \\ y \\ \vartheta \\ \varphi_1 \\ \varphi_2 \\ \varphi_3 \end{bmatrix}$$

L'espace des configurations est de dimension $n_{rp} = n_r + n_p = 2 + 6 = 8$. L'indice de mobilité vaut $M_{rp} = n_r + m_p = 2 + 3 = 5$. Dans le cas de la plate-forme de type omnidirectionnel, les commandes de mobilité et de manœuvrabilité sont identiques : $u_p = \eta_p$. Etant donné que :

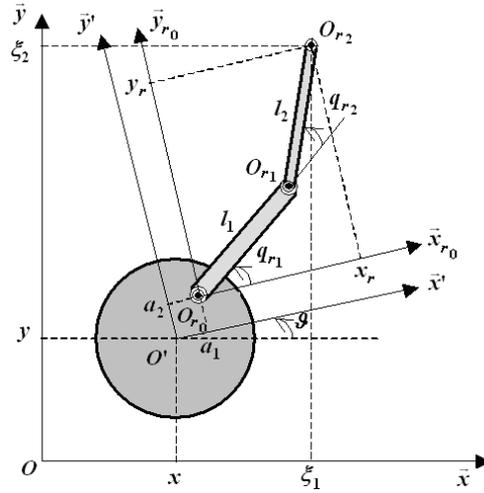


FIGURE II.16 : Manipulateur mobile plan avec une plate-forme omnidirectionnelle

$$u_p = \eta_p = \begin{bmatrix} \eta_{p1} \\ \eta_{p2} \\ \eta_{p3} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\vartheta} \end{bmatrix}$$

La commande de manœuvrabilité du manipulateur mobile est :

$$u = \begin{bmatrix} u_r \\ u_p \end{bmatrix} = \begin{bmatrix} \dot{q}_{r1} \\ \dot{q}_{r2} \\ \dot{x} \\ \dot{y} \\ \dot{\vartheta} \end{bmatrix}$$

avec

$$\delta_m = n_r + \delta_{m_p} = 2 + 3 = 5$$

$$\delta_{M_p} = \delta_{m_p} + \delta_{s_p} = 3 + 0 = 3$$

$$\delta_M = n_r + \delta_{M_p} = 2 + 3 = 5$$

Le MGD de ce manipulateur mobile est :

$$\begin{cases} \xi_1 = x + (a_1 + l_1 \cos(q_{r1}) + l_2 \cos(q_{r1} + q_{r2})) \cos(\vartheta) - \\ \quad (a_2 + l_1 \sin(q_{r1}) + l_2 \sin(q_{r1} + q_{r2})) \sin(\vartheta) \\ \xi_2 = y + (a_1 + l_1 \cos(q_{r1}) + l_2 \cos(q_{r1} + q_{r2})) \sin(\vartheta) + \\ \quad (a_2 + l_1 \sin(q_{r1}) + l_2 \sin(q_{r1} + q_{r2})) \cos(\vartheta) \\ \xi_3 = q_{r1} + q_{r2} + \vartheta \end{cases} \quad (\text{II.59})$$

D'après les équations (II.55), (II.56) et (II.59), nous obtenons le modèle cinématique en situation :

$$\dot{\xi} = \begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \end{bmatrix} = \bar{J}(q_{r_1}, q_{r_2}, \vartheta) \begin{bmatrix} \dot{q}_{r_1} \\ \dot{q}_{r_2} \\ \dot{x} \\ \dot{y} \\ \dot{\vartheta} \end{bmatrix} \quad (\text{II.60})$$

$$\bar{J}(q_{r_1}, q_{r_2}, \vartheta) = \begin{bmatrix} -l_1 S_{1\vartheta} - l_2 S_{12\vartheta} & -l_2 S_{12\vartheta} & 1 & 0 \\ +l_1 C_{1\vartheta} + l_2 C_{12\vartheta} & l_2 C_{12\vartheta} & 0 & 1 \\ 1 & 1 & 0 & 0 \\ -a_1 S_\vartheta - a_2 C_\vartheta - l_1 S_{1\vartheta} - l_2 S_{12\vartheta} \\ +a_1 C_\vartheta - a_2 S_\vartheta + l_1 C_{1\vartheta} + l_2 C_{12\vartheta} \\ 1 \end{bmatrix}$$

Pour la redondance, le degré de mobilité de la plate-forme de type omnidirectionnel vaut $\delta_{m_p} = 3$ et le degré de mobilité du manipulateur mobile $\delta_m = n_r + \delta_{m_p} = 2 + 3 = 5$.

- Pour une tâche en position et orientation, l'espace des configurations est de dimension $m_{rp} = 3$. D'après l'expression de $\bar{J}(q_{r_1}, q_{r_2}, \vartheta)$, le degré de liberté global $D_{rp} = 3$. Le manipulateur mobile est redondant de degré 2, $\delta_m - D_{rp} = 2$.
- Pour une tâche en position, l'espace des configurations est de dimension $m_{rp} = 2$. D'après l'expression de $\bar{J}(q_{r_1}, q_{r_2}, \vartheta)$, le degré de liberté global $D_{rp} = 2$. Le manipulateur mobile est redondant de degré 3, $\delta_m - D_{rp} = 3$.

II.4.6 Modélisation dynamique des manipulateurs mobiles à roues

De manière similaire au cas des plates-formes à roues, le problème est souvent traité autour d'un exemple particulier. C'est le cas des travaux de Y. Yamamoto dans [70] qui, en utilisant le formalisme de Lagrange, établit le modèle dynamique d'un manipulateur mobile composé d'une plate-forme unicycle. Il calcule notamment les termes de couplage dynamique entre les deux sous-systèmes et montre, en simulation, que du point de vue de la commande, seuls les effets de couplage de la plate-forme sur le bras manipulateur influencent réellement les performances. O. Khatib *et al* utilisent dans [27] la modélisation

dynamique de manipulateurs mobiles à roues employés pour un travail de transport collaboratif. Les robots utilisés sont omnidirectionnels et la méthode d'obtention des modèles n'est pas abordée.

Nous proposons pour notre cas de développer le modèle dynamique d'un manipulateur mobile à roues en utilisant le formalisme de Lagrange. Ce modèle est écrit sous une forme directement exploitable pour la commande dynamique du système et il permet la prise en compte d'éventuels efforts d'interaction. L'analyse de sa structure permet de faire apparaître explicitement les termes de couplage dynamique.

Pour décrire le modèle dynamique d'un manipulateur mobile, nous allons adopter la modélisation décrite dans [70, 71]. Cette modélisation présente le comportement individuel de chaque robot ainsi que les interactions entre les dynamiques explicites de la base mobile et du manipulateur [55, 72]. Par conséquent, les équations du mouvement du robot manipulateur soumis aux déplacements d'un véhicule porteur sont données par :

$$\begin{cases} M_r(q_r)\ddot{q}_r + C_{r_1}(q_r, \dot{q}_r)\dot{q}_r + C_{r_2}(q_r, \dot{q}_r, \dot{q}_v) = \tau_r - R_r(q_r, q_v)\ddot{q}_v \\ C_r(q_r, \dot{q}_r, \dot{q}_v) = C_{r_1}(q_r, \dot{q}_r)\dot{q}_r + C_{r_2}(q_r, \dot{q}_r, \dot{q}_v) \end{cases} \quad (\text{II.61})$$

avec τ_r le couple des actionneurs du robot manipulateur, q_r le vecteur articulaire du robot manipulateur, M_r la matrice d'inertie du robot manipulateur, C_{r_1} la matrice des effets centrifuges et de Coriolis du robot manipulateur, C_{r_2} la matrice des effets centrifuges et de Coriolis causés par le mouvement angulaire de la plate-forme, R_r les termes inertiels des couplages dus aux mouvements de la plate-forme mobile. Nous notons que $C_{r_2}(q_r, \dot{q}_r, \dot{q}_v)$ et $R_r(q_r, q_v)\ddot{q}_v$ sont les termes ajoutés à l'équation du mouvement du manipulateur. Ces termes représentent l'interaction dynamique provoquée par le mouvement de la plate-forme mobile.

Supposons que la configuration de la plate-forme est uniquement déterminée par m_v variables dépendantes, $q_v = [q_{v_1}, q_{v_2}, \dots, q_{v_{m_v}}]$. La matrice de transformation homogène définissant le repère \mathcal{R}_{r_0} , repère lié à la base du bras manipulateur, dans le repère \mathcal{R} , repère fixe du monde, est désignée par $T_v(q_v)$. La matrice de transformation définissant le repère \mathcal{R}_{r_i} lié au corps i du bras manipulateur qui est monté sur la plate-forme dans le repère \mathcal{R} est désignée par T_i .

$$T_i = T_v A_1^0 A_2^1 \dots A_i^{i-1} \quad i = 1, \dots, n_r \quad (\text{II.62})$$

avec

$$\mathbf{T}_v = \begin{bmatrix} & & & x \\ & R_p^T(\vartheta) & & y \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Les expressions de C_{r_2} et R_r sont données par :

$$C_{r_2}^{(i)} = 2 \sum_{j=1}^{m_v} \sum_{k=1}^{n_r} \sum_{h=\max(i,k)}^{n_r} \text{trace} \left[\frac{\partial \mathbf{T}_h}{\partial q_{r,i}} I_h \frac{\partial \mathbf{T}_h^T}{\partial q_{v,j} \partial q_{r,k}} \right] \dot{q}_{v,j} \dot{q}_{r,k} + \sum_{j=1}^{m_v} \sum_{k=1}^{m_v} \sum_{h=i}^{n_r} \text{trace} \left[\frac{\partial \mathbf{T}_h}{\partial q_{r,i}} I_h \frac{\partial \mathbf{T}_h^T}{\partial q_{v,j} \partial q_{v,k}} \right] \dot{q}_{v,j} \dot{q}_{v,k} \quad (\text{II.63})$$

$$R_r^{(ij)} = \sum_{k=i}^{n_r} \text{trace} \left[\frac{\partial \mathbf{T}_k}{\partial q_{r,i}} I_k \frac{\partial \mathbf{T}_k^T}{\partial q_{v,j}} \right] \quad 1 \leq i \leq n_r, \quad 1 \leq k \leq n_r, \quad 1 \leq j \leq m_v \quad (\text{II.64})$$

Le premier terme dans l'équation (II.63) caractérise l'effet de Coriolis sur le corps i du bras manipulateur en raison de l'accouplement des vitesses du corps k du bras manipulateur et la variable $q_{v,j}$ de la plate-forme. Le deuxième terme représente la totalité de forces centrifuges exercées sur le corps j par $\dot{q}_{v,j}$ de la plate-forme si $j = k$, et les forces de Coriolis exercées sur le corps i en raison de l'accouplement des vitesses de deux coordonnées de la plate-forme $q_{v,j}$ et $q_{v,k}$ lorsque $j \neq k$.

Les équations du mouvement de la plate-forme mobile prennent la forme suivante :

$$\begin{cases} M_{v_1}(q_v) \ddot{q}_v + C_{v_1}(q_v, \dot{q}_v) \dot{q}_v + C_{v_2}(q_r, q_v, \dot{q}_r, \dot{q}_v) = E_v \tau_v - \\ \quad M_{v_2}(q_r, q_v) \ddot{q}_v - R_v(q_r, q_v) \ddot{q}_r \\ C_v(q_r, q_v, \dot{q}_r, \dot{q}_v) = C_{v_1}(q_v, \dot{q}_v) \dot{q}_v + C_{v_2}(q_r, q_v, \dot{q}_r, \dot{q}_v) \\ M_v(q_r, q_v) = M_{v_1}(q_v) + M_{v_2}(q_r, q_v) \end{cases} \quad (\text{II.65})$$

avec τ_v le vecteur des entrées de commande de la plate-forme mobile, E_v est une matrice constante, M_{v_1} la matrice d'inertie de la plate-forme, C_{v_1} la matrice des termes dépendants de la vitesse de la plate-forme, M_{v_2} et C_{v_2} représentent les termes inertiels et les termes de centrifuges et de Coriolis en raison de la présence du bras manipulateur, R_v représente la matrice d'inertie qui reflète l'effet dynamique du mouvement du bras manipulateur sur le véhicule.

Les expressions de M_{v_2} , C_{v_2} et R_v sont données par :

$$M_{v_2}^{(ij)} = \sum_{k=1}^{n_r} \text{trace} \left[\frac{\partial \mathbf{T}_k}{\partial q_{v,i}} I_k \frac{\partial \mathbf{T}_k^T}{\partial q_{v,j}} \right] \quad 1 \leq i, j \leq m_v, \quad 1 \leq k \leq m_v \quad (\text{II.66})$$

$$C_{v_2}^{(i)} = 2 \sum_{j=1}^{n_r} \sum_{k=1}^{m_v} \sum_{h=j}^{n_r} \text{trace} \left[\frac{\partial \mathbf{T}_h}{\partial q_{v,i}} I_h \frac{\partial \mathbf{T}_h^T}{\partial q_{r,j} \partial q_{v,k}} \right] \dot{q}_{r,j} \dot{q}_{v,k} + \sum_{j=1}^{n_r} \sum_{k=1}^{n_r} \sum_{h=\max(j,k)}^{n_r} \text{trace} \left[\frac{\partial \mathbf{T}_h}{\partial q_{v,i}} I_h \frac{\partial \mathbf{T}_h^T}{\partial q_{r,j} \partial q_{r,k}} \right] \dot{q}_{r,j} \dot{q}_{r,k} \quad (\text{II.67})$$

$$R_v^{(ij)} = \sum_{k=j}^{n_r} \text{trace} \left[\frac{\partial \Gamma_k}{\partial q_{v,i}} I_k \frac{\partial \Gamma_k^T}{\partial q_{r,j}} \right] \quad 1 \leq i \leq m_v, \quad 1 \leq k \leq m_v, \quad 1 \leq j \leq n_r \quad (\text{II.68})$$

Le premier terme dans l'équation (II.67) caractérise l'effet de Coriolis de la coordonnée de la plate-forme $q_{v,i}$ en raison de l'accouplement des vitesses du corps j du bras manipulateur et la variable $q_{v,k}$ de la plate-forme. Le deuxième terme représente la totalité de forces centrifuges exercées sur la coordonnée de la plate-forme $q_{v,i}$ par le corps j du bras manipulateur si $j = k$, et les forces de Coriolis exercées sur la coordonnée de la plate-forme $q_{v,k}$ en raison de l'accouplement des vitesses des deux corps différents du bras manipulateur.

II.4.7 Modélisation dynamique du manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons rotoïdes

Considérons le manipulateur mobile de la figure (II.16) dont nous avons déterminé indépendamment les modèles dynamiques du bras manipulateur plan à deux liaisons rotoïdes (paragraphe II.2.4) et de la plate-forme mobile omnidirectionnelle (paragraphe II.3.4.2). Le point O_{r_0} , qui est l'origine du repère \mathcal{R}_{r_0} lié à la base du bras manipulateur, il a pour coordonnées $[a_1, a_2, 0]^T$ dans le repère \mathcal{R}' . La situation de la plate-forme est donnée par le vecteur $\xi_p = [x, y, \vartheta]^T$ où x et y sont respectivement l'abscisse et l'ordonnée du point O' dans le repère \mathcal{R} et ϑ l'angle (\vec{x}, \vec{x}') . La position du point O_{r_2} dans le repère \mathcal{R} est donnée par les coordonnées ξ_1 et ξ_2 . L'équation dynamique du manipulateur mobile est obtenue en utilisant le formalisme de Lagrange [70, 71].

$$M(q)\dot{\omega} + C(q, \omega) = \tau \quad (\text{II.69})$$

$$\begin{aligned} q_r &= [q_{r_1}, q_{r_2}]^T, \quad q_v = [\varphi_1, \varphi_2, \varphi_3]^T, \quad \omega_r = [\omega_{r_1}, \omega_{r_2}]^T = \dot{q}_r = [\dot{q}_{r_1}, \dot{q}_{r_2}]^T, \quad \omega_v = [\omega_{v_1}, \omega_{v_2}, \omega_{v_3}]^T = \\ &= [\dot{\varphi}_1, \dot{\varphi}_2, \dot{\varphi}_3]^T = \dot{q}_v = [\dot{q}_{v_1}, \dot{q}_{v_2}, \dot{q}_{v_3}]^T, \quad q = [q_r^T, \xi_p^T]^T, \quad \omega = [\omega_r^T, \omega_v^T]^T, \quad \dot{\omega} = [\dot{\omega}_r^T, \dot{\omega}_v^T]^T, \\ \tau_r &= [\tau_{r_1}, \tau_{r_2}]^T, \quad \tau_v = [\tau_{v_1}, \tau_{v_2}, \tau_{v_3}]^T, \quad \tau = [\tau_r^T, [E_v \tau_v]^T]^T. \end{aligned}$$

Les équations du mouvement du robot manipulateur soumis aux déplacements d'un

véhicule porteur sont données par :

$$\begin{cases} M_r(q_r)\dot{\omega}_r + C_{r_1}(q_r, \omega_r)\omega_r + C_{r_2}(q_r, \omega_r, \omega_v) = \tau_r - R_r(q_r, \xi_p)\dot{\omega}_v \\ C_r(q_r, \omega_r, \omega_v) = C_{r_1}(q_r, \omega_r)\omega_r + C_{r_2}(q_r, \omega_r, \omega_v) \end{cases} \quad (\text{II.70})$$

$$\begin{bmatrix} M_r^{(11)} & M_r^{(12)} \\ M_r^{(21)} & M_r^{(22)} \end{bmatrix} \begin{bmatrix} \dot{\omega}_{r_1} \\ \dot{\omega}_{r_2} \end{bmatrix} + \begin{bmatrix} C_{r_1}^{(11)} & C_{r_1}^{(12)} \\ C_{r_1}^{(21)} & C_{r_1}^{(22)} \end{bmatrix} \begin{bmatrix} \omega_{r_1} \\ \omega_{r_2} \end{bmatrix} + \begin{bmatrix} C_{r_2}^{(1)} \\ C_{r_2}^{(2)} \end{bmatrix} = \\ \begin{bmatrix} \tau_{r_1} \\ \tau_{r_2} \end{bmatrix} - \begin{bmatrix} R_r^{(11)} & R_r^{(12)} & R_r^{(13)} \\ R_r^{(21)} & R_r^{(22)} & R_r^{(23)} \end{bmatrix} \begin{bmatrix} \dot{\omega}_{v_1} \\ \dot{\omega}_{v_2} \\ \dot{\omega}_{v_3} \end{bmatrix} \quad (\text{II.71})$$

Les équations du mouvement de la plate-forme mobile prennent la forme suivante :

$$\begin{cases} M_{v_1}(\xi_p)\dot{\omega}_v + C_{v_1}(\xi_p, \omega_v)\omega_v + C_{v_2}(q_r, \xi_p, \omega_r, \omega_v) = E_v\tau_v - M_{v_2}(q_r, \xi_p)\dot{\omega}_v - R_v(q_r, \xi_p)\dot{\omega}_r \\ C_v(q_r, \xi_p, \omega_r, \omega_v) = C_{v_1}(\xi_p, \omega_v)\omega_v + C_{v_2}(q_r, \xi_p, \omega_r, \omega_v) \\ M_v(q_r, \xi_p) = M_{v_1}(\xi_p) + M_{v_2}(q_r, \xi_p) \end{cases} \quad (\text{II.72})$$

$$\begin{bmatrix} M_{v_1}^{(11)} & M_{v_1}^{(12)} & M_{v_1}^{(13)} \\ M_{v_1}^{(21)} & M_{v_1}^{(22)} & M_{v_1}^{(23)} \\ M_{v_1}^{(31)} & M_{v_1}^{(32)} & M_{v_1}^{(33)} \end{bmatrix} \begin{bmatrix} \dot{\omega}_{v_1} \\ \dot{\omega}_{v_2} \\ \dot{\omega}_{v_3} \end{bmatrix} + \begin{bmatrix} C_{v_1}^{(11)} & C_{v_1}^{(12)} & C_{v_1}^{(13)} \\ C_{v_1}^{(21)} & C_{v_1}^{(22)} & C_{v_1}^{(23)} \\ C_{v_1}^{(31)} & C_{v_1}^{(32)} & C_{v_1}^{(33)} \end{bmatrix} \begin{bmatrix} \omega_{v_1} \\ \omega_{v_2} \\ \omega_{v_3} \end{bmatrix} + \begin{bmatrix} C_{v_2}^{(1)} \\ C_{v_2}^{(2)} \\ C_{v_2}^{(3)} \end{bmatrix} = \\ \begin{bmatrix} E_v^{(11)} & E_v^{(12)} & E_v^{(13)} \\ E_v^{(21)} & E_v^{(22)} & E_v^{(23)} \\ E_v^{(31)} & E_v^{(32)} & E_v^{(33)} \end{bmatrix} \begin{bmatrix} \tau_{v_1} \\ \tau_{v_2} \\ \tau_{v_3} \end{bmatrix} - \begin{bmatrix} M_{v_2}^{(11)} & M_{v_2}^{(12)} & M_{v_2}^{(13)} \\ M_{v_2}^{(21)} & M_{v_2}^{(22)} & M_{v_2}^{(23)} \\ M_{v_2}^{(31)} & M_{v_2}^{(32)} & M_{v_2}^{(33)} \end{bmatrix} \begin{bmatrix} \dot{\omega}_{v_1} \\ \dot{\omega}_{v_2} \\ \dot{\omega}_{v_3} \end{bmatrix} - \\ \begin{bmatrix} R_v^{(11)} & R_v^{(12)} \\ R_v^{(21)} & R_v^{(22)} \\ R_v^{(31)} & R_v^{(32)} \end{bmatrix} \begin{bmatrix} \dot{\omega}_{r_1} \\ \dot{\omega}_{r_2} \end{bmatrix} \quad (\text{II.73})$$

Le modèle dynamique (Eq. II.69) du manipulateur mobile est décrit par :

$$\begin{bmatrix} M_r & R_r \\ R_v & M_v \end{bmatrix} \begin{bmatrix} \dot{\omega}_r \\ \dot{\omega}_v \end{bmatrix} + \begin{bmatrix} C_r \\ C_v \end{bmatrix} = \begin{bmatrix} \tau_r \\ E_v\tau_v \end{bmatrix} \quad (\text{II.74})$$

Les expressions de C_{r_1} , C_{r_2} , R_r , M_{v_1} , C_{v_1} , M_{v_2} , C_{v_2} , E_v et R_v sont données par :

$$\begin{aligned}
C_{r_1}^{(11)} &= -m_2 l_1 l_{c_2} \sin(q_{r_2}) \omega_{r_2} \\
C_{r_1}^{(12)} &= -m_2 l_1 l_{c_2} \sin(q_{r_2}) (\omega_{r_1} + \omega_{r_2}) \\
C_{r_1}^{(21)} &= +m_2 l_1 l_{c_2} \sin(q_{r_2}) \omega_{r_1} \\
C_{r_1}^{(22)} &= 0 \\
C_{r_2}^{(1)} &= (-2I_2 l_1 l_2 \sin(q_{r_2})) \omega_{r_2} \omega_{v_3} \\
C_{r_2}^{(2)} &= (2I_2 l_1 l_2 \sin(q_{r_2})) \omega_{r_1} \omega_{v_3} + (I_2 l_1 l_2 \sin(q_{r_2})) \omega_{v_3}^2
\end{aligned}$$

$$\begin{aligned}
R_r^{(11)} &= [-l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) - l_1 \sin(q_{r_1} + \vartheta)] I_2 - I_1 l_1 \sin(q_{r_1} + \vartheta) \\
R_r^{(12)} &= [+l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + l_1 \cos(q_{r_1} + \vartheta)] I_2 + I_1 l_1 \cos(q_{r_1} + \vartheta) \\
R_r^{(13)} &= [(-l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) - l_1 \sin(q_{r_1} + \vartheta))^2 + (l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + \\
&\quad l_1 \cos(q_{r_1} + \vartheta))^2 + 2] I_2 + I_1 (l_1^2 + 2) \\
R_r^{(21)} &= -I_2 l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) \\
R_r^{(22)} &= +I_2 l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) \\
R_r^{(23)} &= I_2 [l_1 l_2 \cos(q_{r_2}) + l_2^2 + 2]
\end{aligned}$$

$$\begin{aligned}
M_{v_1}^{(11)} &= M_{v_1}^{(22)} = M_{v_1}^{(33)} = 4r J_R + \frac{4m_R r}{9} + \frac{J_m n_m^2}{r} \\
M_{v_1}^{(12)} &= M_{v_1}^{(13)} = M_{v_1}^{(21)} = M_{v_1}^{(23)} = M_{v_1}^{(31)} = M_{v_1}^{(32)} = 4r J_R - \frac{2m_R r}{9}
\end{aligned}$$

$$\begin{aligned}
C_{v_1}^{(11)} &= C_{v_1}^{(22)} = C_{v_1}^{(33)} = \frac{C_m C_e n_m^2}{r R_a} + \frac{b_m n_m^2}{r} \\
C_{v_1}^{(12)} &= C_{v_1}^{(23)} = C_{v_1}^{(31)} = \frac{-2rm_R \dot{\vartheta}}{3\sqrt{3}} \\
C_{v_1}^{(13)} &= C_{v_1}^{(21)} = C_{v_1}^{(32)} = \frac{2rm_R \dot{\vartheta}}{3\sqrt{3}} \\
C_{v_2}^{(1)} &= -2I_1 l_1 \cos(q_{r_1} + \vartheta) \omega_{r_1} \omega_{v_3} - 2I_2 [l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + l_1 \cos(q_{r_1} + \vartheta)] \omega_{r_1} \omega_{v_3} - \\
&\quad 2I_2 l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_2} \omega_{v_3} - I_1 l_1 \cos(q_{r_1} + \vartheta) \omega_{r_1}^2 - I_2 [l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + \\
&\quad l_1 \cos(q_{r_1} + \vartheta)] \omega_{r_1}^2 - I_2 l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_1} \omega_{r_2} - I_2 l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_2} \omega_{r_1} - \\
&\quad I_2 l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_2}^2 \\
C_{v_2}^{(2)} &= -2I_1 l_1 \sin(q_{r_1} + \vartheta) \omega_{r_1} \omega_{v_3} - 2I_2 [l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) + l_1 \sin(q_{r_1} + \vartheta)] \omega_{r_1} \omega_{v_3} - \\
&\quad 2I_2 l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_2} \omega_{v_3} - I_1 l_1 \sin(q_{r_1} + \vartheta) \omega_{r_1}^2 - I_2 [l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) + \\
&\quad l_1 \sin(q_{r_1} + \vartheta)] \omega_{r_1}^2 - I_2 l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_1} \omega_{r_2} - I_2 l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_2} \omega_{r_1} - \\
&\quad I_2 l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) \omega_{r_2}^2 \\
C_{v_2}^{(3)} &= -2I_2 l_1 l_2 \sin(q_{r_2}) \omega_{r_2} \omega_{v_3} - I_2 l_1 l_2 \sin(q_{r_2}) \omega_{r_1} \omega_{r_2} - I_2 l_1 l_2 \sin(q_{r_2}) \omega_{r_2} \omega_{r_1} - \\
&\quad I_2 l_1 l_2 \sin(q_{r_2}) \omega_{r_2}
\end{aligned}$$

$$E_v^{(11)} = E_v^{(22)} = E_v^{(33)} = \frac{C_m n_m}{r R_a}$$

$$E_v^{(12)} = E_v^{(13)} = E_v^{(21)} = E_v^{(23)} = E_v^{(31)} = E_v^{(32)} = 0$$

$$M_{v_2}^{(11)} = M_{v_2}^{(22)} = I_1 + I_2$$

$$M_{v_2}^{(12)} = M_{v_2}^{(21)} = 0$$

$$M_{v_2}^{(13)} = [-l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) - l_1 \sin(q_{r_1} + \vartheta)] I_2 - I_1 l_1 \sin(q_{r_1} + \vartheta)$$

$$M_{v_2}^{(23)} = [l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + l_1 \cos(q_{r_1} + \vartheta)] I_2 + I_1 l_1 \cos(q_{r_1} + \vartheta)$$

$$M_{v_2}^{(31)} = [-l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) - l_1 \sin(q_{r_1} + \vartheta)] I_2 - I_1 l_1 \sin(q_{r_1} + \vartheta)$$

$$M_{v_2}^{(32)} = [+l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + l_1 \cos(q_{r_1} + \vartheta)] I_2 + I_1 l_1 \cos(q_{r_1} + \vartheta)$$

$$M_{v_2}^{(33)} = [(-l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) - l_1 \sin(q_{r_1} + \vartheta))^2 + (l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + l_1 \cos(q_{r_1} + \vartheta))^2 + 2] I_2 + I_1 (l_1^2 + 2)$$

$$R_v^{(11)} = [-l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) - l_1 \sin(q_{r_1} + \vartheta)] I_2 - I_1 l_1 \sin(q_{r_1} + \vartheta)$$

$$R_v^{(12)} = -I_2 l_2 \sin(q_{r_1} + q_{r_2} + \vartheta)$$

$$R_v^{(21)} = [+l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + l_1 \cos(q_{r_1} + \vartheta)] I_2 + I_1 l_1 \cos(q_{r_1} + \vartheta)$$

$$R_v^{(22)} = +I_2 l_2 \cos(q_{r_1} + q_{r_2} + \vartheta)$$

$$R_v^{(31)} = [(-l_2 \sin(q_{r_1} + q_{r_2} + \vartheta) - l_1 \sin(q_{r_1} + \vartheta))^2 + (l_2 \cos(q_{r_1} + q_{r_2} + \vartheta) + l_1 \cos(q_{r_1} + \vartheta))^2 + 2] I_2 + I_1 (l_1^2 + 2)$$

$$R_v^{(32)} = I_2 [l_1 l_2 \cos(q_{r_2}) + l_2^2 + 2]$$

II.5 Conclusion

Ce chapitre a présenté les modèles géométriques, cinématiques et dynamiques du bras manipulateur plan à deux liaisons rotoïdes, de la plate-forme mobile omnidirectionnelle *ROMNI* et du manipulateur mobile composé de la plate-forme et du bras manipulateur. Ces modèles seront utilisés dans la suite de cette thèse pour le développement d'une commande cinématique et dynamique.

Chapitre III

Commande du manipulateur mobile

III.1 Introduction

Le problème de commande du bras manipulateur se ramène dans certains cas à un problème de génération de trajectoires. Pour ce faire, nous distinguons dans un premier temps la planification du mouvement, qui consiste à trouver la trajectoire adéquate dans l'espace opérationnel de l'organe terminal. Dans un deuxième temps, cette trajectoire sera traduite dans l'espace articulaire pour être utilisée comme entrée de référence des actionneurs.

Plusieurs facteurs doivent être pris en compte lors du choix du mécanisme et de la taille du robot manipulateur durant la phase de conception, ou lors de la détermination de la posture du manipulateur dans l'espace de travail pour effectuer une tâche opérationnelle. Lors de sa réalisation d'une tâche, le robot manipulateur doit se présenter dans les meilleures conditions afin de transmettre de manière optimale la vitesse et l'effort à l'objet manipulé. Un facteur important est la facilité de changement arbitraire de la position et de l'orientation de l'organe terminal. C'est la notion de manipulabilité. Cette notion introduite dans [73, 74, 75] permet d'analyser la cinématique des bras manipulateurs. Yoshikawa a introduit le concept de l'ellipsoïde de manipulabilité à partir duquel il a défini quelques mesures pour caractériser la capacité de manipulation.

La localisation (position et orientation) de l'effecteur est de suivre une trajectoire opérationnelle imposée, c'est à dire dont la tâche opérationnelle est définie par une trajectoire de l'organe terminal, outre la redondance qui caractérise généralement ces systèmes. La

coordination est alors basée sur la résolution de cette redondance. Dans la littérature, les contributions les plus importantes peuvent être divisées en deux sous-classes. Dans la première, la génération du parcours généralisé est obtenue en ajoutant des tâches en vue de résoudre le problème de la redondance [76, 77, 69]. Dans la deuxième, la génération du parcours généralisé est obtenue en minimisant un critère quadratique [77]. Dans notre travail, nous allons utiliser la méthode des tâches additionnelles. C'est une technique qui permet d'étendre la dimension de l'espace de la tâche originale en imposant un nombre de contraintes décrites en fonction de variables de configuration.

Dans ce chapitre, nous présentons tout d'abord les éléments de la théorie de la manipulabilité nécessaires et l'application de cette notion sur les bras manipulateurs et les manipulateurs mobiles. Nous allons décrire ensuite les méthodes proposées comme solutions au problème d'exécution de la tâche dans un espace opérationnel. Pour résoudre le problème de la redondance, dans le cas d'un mouvement à trajectoire opérationnelle imposée, nous allons utiliser le concept des tâches additionnelles. Et pour réaliser l'objectif de l'évitement d'obstacles, nous allons proposer la stratégie du contrôle cinématique augmenté d'une boucle externe de contrôle en impédance.

III.2 Manipulabilité des bras manipulateurs et des manipulateurs mobiles

III.2.1 Manipulabilité des bras manipulateurs

La manipulabilité est une quantité qui représente la capacité du manipulateur à se déplacer autour d'une configuration donnée. La manipulabilité est utilisée dans diverses applications : lors de la conception des mécanismes, dans la planification de trajectoires ou la commande des systèmes mécaniques comme les manipulateurs redondants. Quand le manipulateur est redondant, il existe une infinité de solutions de la cinématique inverse. Sciavicco et Siciliano [54] utilisent l'indice de manipulabilité comme un critère à maximiser pour mettre le bras dans une configuration loin de ses configurations singulières. Nakamura [78] et Yoshikawa [73, 75] prolongent l'exploitation de la redondance avec des critères exprimés dans l'espace cartésien : une tâche cartésienne secondaire qui peut être satisfaite

sans affecter la tâche principale. C'est la résolution simultanée de deux tâches avec des priorités différentes.

Définition III.1. *Considérons l'ensemble des vitesses opérationnelles réalisables $\dot{\xi}_r$ par les vitesses articulaires \dot{q}_r telle que la norme Euclidienne $\|\dot{q}_r\|^2 \leq 1$ est un ellipsoïde, appelé ellipsoïde de manipulabilité, dans l'espace Euclidien de dimension m_r .*

Définition III.2. *Soit $J_r = U_r \Sigma_r V_r^T$ la décomposition en valeurs singulières (SVD) de la matrice J_r , où $U_r \in \mathfrak{R}^{m_r \times m_r}$ et $V_r \in \mathfrak{R}^{n_r \times n_r}$ des matrices orthogonales, et :*

$$\Sigma_r = \begin{bmatrix} \sigma_{r_1} & 0 & \dots & 0 & 0 \\ 0 & \sigma_{r_2} & \dots & 0 & 0 \\ \cdot & \cdot & \dots & \cdot & \cdot & 0 \\ \cdot & \cdot & \dots & \cdot & \cdot \\ \cdot & \cdot & \dots & \sigma_{r_{m_r}} & 0 \end{bmatrix} \in \mathfrak{R}^{m_r \times n_r} \quad (\text{III.1})$$

où les scalaires $\sigma_{r_1} \geq \sigma_{r_2} \geq \dots \geq \sigma_{r_{m_r}}$ sont appelés les valeurs singulières de J_r .

Les axes principaux de l'ellipsoïde de manipulabilité sont donnés par les vecteurs $\sigma_{r_1} U_{r_1}, \sigma_{r_2} U_{r_2}, \dots, \sigma_{r_{m_r}} U_{r_{m_r}}$, le vecteur U_{r_k} est la k -ième colonne de la matrice U_r .

La forme de l'ellipse si l'espace opérationnel est de dimension deux ou de l'ellipsoïde si l'espace opérationnel est de dimension trois donne des indications sur la capacité de l'organe terminal à transmettre une vitesse ou effort dans l'espace opérationnel. Dans la direction de l'axe majeur de l'ellipsoïde, l'organe terminal peut se déplacer à une grande vitesse. Dans la direction de l'axe mineur de l'ellipsoïde de manipulabilité, l'organe terminal peut se déplacer à faible vitesse. Si l'ellipsoïde tend vers une sphère, l'organe terminal peut se déplacer dans toutes les directions uniformément. Puisque l'ellipsoïde représente la capacité de manipulation, il est appelé ellipsoïde de manipulabilité (Fig. III.1).

Des mesures de manipulabilité tirées du concept de manipulabilité sont définies pour caractériser la capacité de manipulation des robots manipulateurs.

III.2.1.1 Mesures de manipulabilité

Une des mesures de manipulabilité la plus courante, dérivée de l'ellipsoïde de manipulabilité, est :

$$w_r = \sqrt{\det(J_r(q_r) J_r^T(q_r))} \quad (\text{III.2})$$

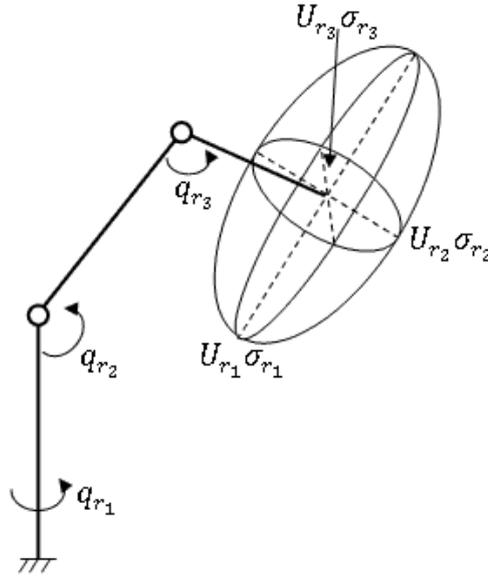


FIGURE III.1 : Ellipsoïde de manipulabilité en 3D

qui s'écrit en fonction des valeurs singulières de J_r sous la forme :

$$w_r = \sigma_{r_1} \sigma_{r_2} \dots \sigma_{r_{m_r}} \quad (\text{III.3})$$

Cette mesure est proportionnelle au volume de l'ellipsoïde de manipulabilité. Dans le cas des manipulateurs non redondants ($n_r = m_r$), l'expression de la mesure de manipulabilité w_r , (Eq .III.2), se simplifie à :

$$w_r = |\det(J_r(q_r))| \quad (\text{III.4})$$

III.2.1.2 Application de la manipulabilité aux bras manipulateurs

Une des applications principales de la manipulabilité est l'analyse de la cinématique des mécanismes. Pour illustrer les différentes notions décrites ci-dessus, nous considérons le bras manipulateur plan à deux liaisons rotoïdes, donné par la figure (II.1), pour lequel $l_1 = l_2 = 1 \text{ m}$. Cet exemple est choisi par rapport à sa simplicité [54, 50]. La configuration du bras manipulateur plan est donnée par les angles de rotation q_{r_1} et q_{r_2} des deux liaisons. Le modèle géométrique direct (MGD) du bras manipulateur est donné par l'équation (II.11) et le modèle cinématique direct (MCD) par l'équation (II.13). Nous examinons l'évolution de la manipulabilité quand la position de l'OT suit une ligne droite depuis une configuration repliée ($q_{r_1} = \frac{\pi}{2}$, $q_{r_2} = -\pi \text{ rad}$) à une configuration étendue ($q_{r_1} = q_{r_2} =$

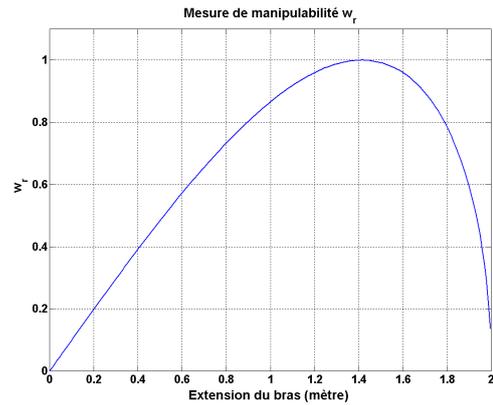
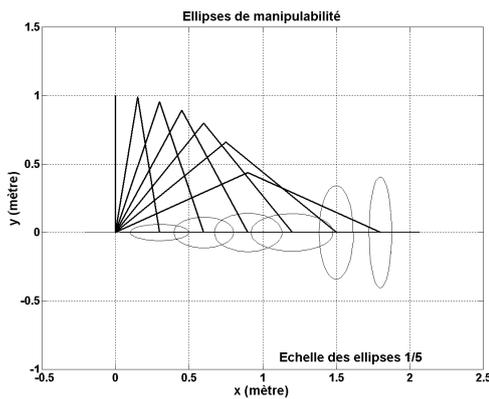


FIGURE III.2 : Ellipses de manipulabilité **FIGURE III.3 :** Mesure de manipulabilité

w_r

0 rad) [56, 73]. La figure (III.2) représente l'évolution des ellipses de manipulabilité. La forme de l'ellipse donne des informations sur la distribution des vitesses. Les configurations singulières correspondent à des ellipses plates. La forme de l'ellipse de manipulabilité tend vers un cercle lorsque la capacité de déplacement est uniforme dans toutes les directions du plan du cercle. La figure (III.3) montre la mesure de manipulabilité en fonction de l'extension du bras. La mesure w_r vaut : $w_r = |\det(J_r(q_r))| = |l_1 l_2 \sin(q_{r_2})|$, elle s'annule pour les deux configurations $q_{r_2} = 0$ ou $-\pi$ (bras étendu ou replié). w_r est maximal pour $q_{r_2} = \pm\pi/2$.

III.2.2 Manipulabilité des manipulateurs mobiles

La manipulabilité des manipulateurs mobiles a été étudiée par un petit nombre de groupes de recherches. Seraji [69] a étudié l'effet de la mobilité de la plate-forme sur l'indice de manipulabilité. La mesure de la manipulabilité est utilisée dans une commande cinématique du manipulateur mobile. Seraji a développé une méthode de commande cinématique qui définit la relation mathématique entre les vitesses généralisées cherchées et les vitesses opérationnelles désirées [69]. La méthode est basée sur la formulation des tâches additionnelles pour rendre le système non redondant. Yamamoto et Yun [79] ont traité la manipulation et la locomotion ensemble dans un même cadre du point de vue espace de la tâche. Le système étudié par les auteurs est constitué de deux manipulateurs embarqués sur une plate-forme mobile et qui manipulent un objet commun. Gardner et Valinsky

[80] ont utilisé la manipulabilité du manipulateur mobile dans le but de la conception de robots. Les auteurs ont introduit des comparaisons numériques qui permettent de choisir la position de la base du bras manipulateur à trois liaisons rotoïdes sur une plate-forme mobile de type unicycle. Leur système est destiné à réaliser une tâche spécifiée qui consiste à tracer un marquage en ligne droite sur une autoroute.

La définition de la manipulabilité aux cas des manipulateurs mobiles non-holonômes décrit par le modèle cinématique en situation (II.56) est par analogie aux cas des bras manipulateurs en remplaçant respectivement ξ_r , u_r et $J_r(q_r)$ par ξ , η et $\bar{J}(q_r, \vartheta, \beta_{co})$. La manipulabilité caractérise un sous ensemble $\dot{\xi}$ de vitesses réalisables telles que les vitesses de commande η correspondantes vérifient la relation $\|\eta\|^2 \leq 1$. Par conséquent, plusieurs mesures définies pour les bras manipulateurs peuvent être réutilisées dans le cas du manipulateur mobile en utilisant les valeurs singulières de la matrice \bar{J} . La mesure de manipulabilité pour le manipulateur mobile est comme suit :

$$w = \sqrt{\bar{J}(q_r, \vartheta, \beta_{co})\bar{J}^T(q_r, \vartheta, \beta_{co})} \quad (\text{III.5})$$

III.3 La commande du manipulateur mobile

Nous décrivons ci-après les deux modes de fonctionnement pour le manipulateur mobile à partir des différentes tâches allouées à ce système. Les deux modes sont définis comme suit :

- Travail alterné : La base mobile est utilisée pour transporter le bras manipulateur vers son espace de travail et puis le bras manipule.
- Travail continu : La base mobile et le bras manipulateur travaillent en coopération pour l'exécution d'une tâche continue dans l'espace de travail.

III.3.1 Commande séparée des mouvements du manipulateur et de la base mobile omnidirectionnelle

La commande séparée du bras manipulateur et de la plate-forme mobile est utilisée pour permettre la réalisation de tâches ponctuelles. La manipulation de l'objet pendant le déplacement n'étant pas envisagée ce qui allège la charge de calcul. Dans ce mode de fonctionnement un seul sous système est activé : c'est la commande décentralisée.

La base mobile se déplace d'une position vers une autre dans le but de configurer le bras manipulateur dans son espace de travail. Pendant le déplacement de la base mobile, le bras est mis dans une configuration optimale. Par contre, dès lors qu'il s'agit d'actionner le bras pour exécuter une tâche précise, la base mobile rendue stationnaire. Dans ce mode, le suivi de trajectoire se fait en deux étapes : d'abord la base mobile se déplace vers l'espace de travail du manipulateur puis se met en position stationnaire suivant une certaine orientation. Le bras manipulateur effectue dans l'espace de la tâche suivant les trajectoires de référence qui lui sont indiquées au niveau des articulations par la cinématique inverse.

La configuration optimale d'un bras manipulateur plan à deux liaisons rotoïdes est garantie par la maximisation de la manipulabilité :

$$w_r = |\det J_r(q_{r_1}, q_{r_2})| = |l_1 l_2 \sin(q_{r_2})| \quad (\text{III.6})$$

où $J_r(q_{r_1}, q_{r_2})$ la matrice jacobienne du bras manipulateur.

III.3.2 Commande à mouvement opérationnel imposé

Cette partie introduit une structure de commande pour la réalisation de tâches assignées à des manipulateurs mobiles à roues. Dans le cas du contrôle de mouvement, la problématique du contrôle opérationnel se pose comme suit : soit $\xi^*(t)$, la situation de consigne pour l'OT, quelles consignes articulaires envoyer aux actionneurs du système afin de faire tendre la situation réelle $\xi(t)$ vers la situation désirée. Il s'agit en fait de réguler à zéro l'erreur de situation opérationnelle $e(t) = \xi^*(t) - \xi(t)$.

Le mouvement opérationnel d'un manipulateur mobile peut être défini par :

$$\begin{aligned} \xi &: [t_0, t_f] \rightarrow \mathcal{M} \\ t &\mapsto \xi(t) \end{aligned} \quad (\text{III.7})$$

où t désigne le temps. Le mouvement généralisé est lui défini par la fonction :

$$\begin{aligned} q_{rp} &: [t_0, t_f] \rightarrow \mathcal{N} \\ t &\mapsto q_{rp}(t) \end{aligned} \quad (\text{III.8})$$

La trajectoire opérationnelle de mouvement désirée est définie par :

$$\begin{aligned} \xi^* &: [t_0, t_f] \rightarrow \mathcal{M} \\ t &\mapsto \xi^*(t) \end{aligned} \quad (\text{III.9})$$

Résoudre le problème de la commande à mouvement opérationnel imposé consiste à produire, à chaque instant, la commande de manœuvrabilité $u(t)$ du manipulateur mobile lui permettant de réaliser un mouvement opérationnel de consigne.

Nous trouvons dans la littérature de nombreuses méthodes permettant de résoudre le problème de la commande des bras manipulateurs à mouvement opérationnel imposé [54] ; elles sont appelées algorithmes de cinématique inverse. Dans ce travail, nous nous sommes particulièrement intéressés par les méthodes dites d'inversion au cas des manipulateurs mobiles [56, 69].

III.3.2.1 Méthodes d'inversion : Cas des bras manipulateurs

Les méthodes d'inversion pour la commande à mouvement opérationnel imposé sont nombreuses [54, 78]. Elles consistent à calculer les vitesses généralisées nécessaires pour obtenir les vitesses opérationnelles spécifiées. En boucle ouverte, pour un bras manipulateur décrit par son MCD sous la forme (II.4), nous choisissons une loi du type :

$$u_r(t) = D_{J_r}(t)\dot{\xi}_r^*(t) \quad (\text{III.10})$$

où $D_{J_r}(t)$ est une inverse à droite de $J_r(t)$ et $\dot{\xi}_r^*(t)$ le mouvement opérationnel de consigne. En pré-multipliant l'équation (III.10) par $J_r(t)$,

$$\begin{aligned} J_r(t)u_r(t) &= J_r(t)D_{J_r}(t)\dot{\xi}_r^*(t) \\ \dot{\xi}_r(t) &= \dot{\xi}_r^*(t) \end{aligned} \quad (\text{III.11})$$

Nous constatons que cette loi conduit à une erreur $e_r(t) = \xi_r^*(t) - \xi_r(t)$ nulle sur la situation de l'OT. Nous choisissons une loi du type [56] :

$$u_r(t) = D_{J_r}(t)(\dot{\xi}_r^*(t) + W_r(\xi_r^*(t) - \xi_r(t))) \quad (\text{III.12})$$

où W_r est une matrice de pondération définie positive. En pré-multipliant l'équation (III.12) par $J_r(t)$,

$$\begin{aligned} J_r(t)u_r(t) &= J_r(t)D_{J_r}(t)(\dot{\xi}_r^*(t) + W_r(\xi_r^*(t) - \xi_r(t))) \\ \dot{\xi}_r(t) &= \dot{\xi}_r^*(t) + W_r(\xi_r^*(t) - \xi_r(t)) \end{aligned} \quad (\text{III.13})$$

Nous constatons que cette loi conduit à une décroissance asymptotique de l'erreur $e_r(t) : \dot{e}_r(t) + W_r e_r(t) = 0$.

Dans le cas où le bras manipulateur n'est pas redondant vis-à-vis de la tâche, au lieu d'inverser la matrice jacobienne puis de multiplier par la vitesse opérationnelle de consigne, nous calculons directement le modèle cinématique inverse du bras manipulateur. Dans le cas où le bras manipulateur est redondant vis-à-vis de la tâche, nous calculons la matrice inverse à droite de $J_r(t)$ puis nous multiplions par la vitesse opérationnelle de consigne.

III.3.2.2 Méthodes d'inversion : Cas des manipulateurs mobiles

III.3.2.3 Méthode des tâches additionnelles

Comme le manipulateur mobile est un mécanisme qui est caractérisé par une certaine redondance, le modèle cinématique en situation (MCS) est un système linéaire rectangulaire qui possède plus d'inconnues que d'équations ($m_{rp} < \delta_m$) (II.56). Le système augmenté d'un nombre m_{add} de tâches additionnelles sont aussi exprimées sous forme de relations linéaires, elles sont décrites par la relation suivante [56, 69] :

$$\dot{\xi}_{add} = J_{add}(q_{rp})\dot{q}_{rp} \quad (\text{III.14})$$

Les tâches additionnelles sont des fonctions de la configuration du bras manipulateur, de la situation de la plate-forme et de l'orientation d'éventuelles roues centrées orientables. Dans le cas simple, le degré de dirigeabilité est égale à zéro $\delta_{s_p} = 0$. Dans ce cas, les degrés de mobilité et de manœuvrabilité sont égaux $\delta_m = \delta_M$ et la commande de manœuvrabilité est réduite à la commande de mobilité $u = \eta$. D'après le MCC ($\dot{q}_{rp} = S(q_{rp})u$), l'équation des m_{add} tâches additionnelles (III.14) peut être réécrite sous la forme :

$$\dot{\xi}_{add} = \bar{J}_{add}(q_{rp})\eta \quad (\text{III.15})$$

avec $\bar{J}_{add}(q_{rp}) = J_{add}(q_{rp})S(q_{rp})$ est une matrice de dimension $m_{add} \times (n_r + \delta_{m_p})$. Le système linéaire augmenté en regroupant (III.15) et (II.56) s'écrit :

$$\dot{\xi}_t = \bar{J}_t(q_{rp})\eta \quad (\text{III.16})$$

$$\text{avec : } \bar{J}_t(q_{rp}) = \begin{bmatrix} \bar{J}(q_r, \vartheta, \beta_{co}) \\ \bar{J}_{add}(q_{rp}) \end{bmatrix} \text{ et } \xi_t = \begin{bmatrix} \xi \\ \xi_{add} \end{bmatrix}$$

Cherchons à réguler la valeur actuelle ξ_t à $\xi_t^* = [\xi^{*T}, \xi_{add}^{*T}]^T$, avec $e_t = [e^T, e_{add}^T]^T = \xi_t^* - \xi_t$ désigne l'erreur sur la situation $e = \xi^* - \xi$ et l'erreur sur les tâches additionnelles

$e_{add} = \xi_{add}^* - \xi_{add}$. La matrice $\bar{J}_t(q_{rp})$ est de dimension $(m_{rp} + m_{add}) \times (n_r + \delta_{m_p})$ et de rang $\rho \leq \min(m_{rp} + m_{add}, n_r + \delta_{m_p})$. Nous supposons le cas où $\bar{J}_t(q_{rp})$ est de rang maximal, $\rho = \min(m_{rp} + m_{add}, n_r + \delta_{m_p})$. La loi du contrôle η est calculée en résolvant le système linéaire (III.15) [81, 56].

- Cas où $\rho = \delta_m \leq m_{rp} + m_{add}$

$$\eta = L(\dot{\xi}^* + W(\xi^* - \xi)) + (I_{\delta_m} - L\bar{J})\bar{J}_t^+ \dot{\xi}_t^* \quad (\text{III.17})$$

L est une matrice inverse à droite de \bar{J} , W est une matrice de pondération définie positive de dimension m_{rp} , $\bar{J}_t^+ = (\bar{J}_t^T \bar{J}_t)^{-1} \bar{J}_t^T$ est la matrice inverse à gauche de \bar{J}_t . ξ^* est la trajectoire désirée du manipulateur mobile.

- Cas où $\rho = m_{rp} + m_{add} \leq \delta_m$

$$\eta = \bar{J}_t^+(\dot{\xi}_t^* + W_t(\xi_t^* - \xi_t)) + (I_{\delta_m} - \bar{J}_t^+ \bar{J}_t)g \quad \forall g \in \mathcal{R}^{\delta_m} \quad (\text{III.18})$$

W_t est une matrice de pondération définie positive de dimension $m_{rp} + m_{add}$, $\bar{J}_t^+ = \bar{J}_t^T (\bar{J}_t \bar{J}_t^T)^{-1}$ est la matrice inverse à droite de \bar{J}_t , g est un vecteur quelconque.

- Cas où $\rho = m_{rp} + m_{add} = \delta_m$

$$\eta = \bar{J}_t^{-1}(\dot{\xi}_t^* + W_t(\xi_t^* - \xi_t)) \quad (\text{III.19})$$

\bar{J}_t^{-1} est la matrice inverse de \bar{J}_t .

Maintenant que nous avons défini les lois de commande, nous allons utiliser cette méthode et nous allons l'appliquer au manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons.

III.4 Application sur les manipulateurs mobiles omnidirectionnels

III.4.1 Manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons

Notre manipulateur mobile est composé d'une plate-forme de type omnidirectionnel et un bras manipulateur plan à deux liaisons. Nous allons restreindre l'espace opérationnel

en ne considérant que la position de l'organe terminal $\xi = [\xi_1, \xi_2]^T$, soit $m_{rp} = 2$. Dans ce cas le MGD de ce manipulateur mobile est :

$$\begin{cases} \xi_1 = x + (a_1 + l_1 \cos(q_{r_1}) + l_2 \cos(q_{r_1} + q_{r_2})) \cos(\vartheta) - \\ \quad (a_2 + l_1 \sin(q_{r_1}) + l_2 \sin(q_{r_1} + q_{r_2})) \sin(\vartheta) \\ \xi_2 = y + (a_1 + l_1 \cos(q_{r_1}) + l_2 \cos(q_{r_1} + q_{r_2})) \sin(\vartheta) + \\ \quad (a_2 + l_1 \sin(q_{r_1}) + l_2 \sin(q_{r_1} + q_{r_2})) \cos(\vartheta) \end{cases}$$

Nous prenons le cas où $a_1 = a_2 = 0$. La matrice jacobienne est donnée par :

$$\bar{J}(q_{r_1}, q_{r_2}, \vartheta) = \begin{bmatrix} -l_1 S_{1\vartheta} - l_2 S_{12\vartheta} & -l_2 S_{12\vartheta} & 1 & 0 & -l_1 S_{1\vartheta} - l_2 S_{12\vartheta} \\ +l_1 C_{1\vartheta} + l_2 C_{12\vartheta} & l_2 C_{12\vartheta} & 0 & 1 & +l_1 C_{1\vartheta} + l_2 C_{12\vartheta} \end{bmatrix}$$

Nous prenons l'exemple pour lequel $m_{rp} + m_{add} = \delta_m$. Donc, nous imposons m_{add} tâches additionnelles :

$$\xi_{add} = \begin{bmatrix} \xi_{add_1} \\ \xi_{add_2} \\ \xi_{add_3} \end{bmatrix} = \begin{bmatrix} x \\ y \\ \vartheta \end{bmatrix}$$

Nous avons donc :

$$\dot{\xi}_{add} = \begin{bmatrix} \dot{\xi}_{add_1} \\ \dot{\xi}_{add_2} \\ \dot{\xi}_{add_3} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\vartheta} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_{J_{add}} \begin{bmatrix} \dot{q}_{r_1} \\ \dot{q}_{r_2} \\ \dot{x} \\ \dot{y} \\ \dot{\vartheta} \\ \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \end{bmatrix}$$

$$\bar{J}_{add}(\vartheta) = J_{add}(\vartheta)S(\vartheta) = J_{add}(\vartheta) \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{1}{r} \sin(\alpha_1 + \vartheta) & -\frac{1}{r} \cos(\alpha_1 + \vartheta) & -R_1/r \\ 0 & 0 & \frac{1}{r} \sin(\alpha_2 + \vartheta) & -\frac{1}{r} \cos(\alpha_2 + \vartheta) & -R_2/r \\ 0 & 0 & \frac{1}{r} \sin(\alpha_3 + \vartheta) & -\frac{1}{r} \cos(\alpha_3 + \vartheta) & -R_3/r \end{bmatrix}$$

soit :

$$\bar{J}_{add}(\vartheta) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

et :

$$\bar{J}_t(q_{r_1}, q_{r_2}, \vartheta) = \begin{bmatrix} \bar{J}(q_{r_1}, q_{r_2}, \vartheta) \\ \bar{J}_{add}(\vartheta) \end{bmatrix} = \begin{bmatrix} -l_1 S_{1\vartheta} - l_2 S_{12\vartheta} & -l_2 S_{12\vartheta} & 1 & 0 & -l_1 S_{1\vartheta} - l_2 S_{12\vartheta} \\ +l_1 C_{1\vartheta} + l_2 C_{12\vartheta} & l_2 C_{12\vartheta} & 0 & 1 & +l_1 C_{1\vartheta} + l_2 C_{12\vartheta} \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Nous choisissons un exemple numérique $l_1 = l_2 = 0.5 \text{ m}$ et où la configuration initiale du manipulateur mobile est telle que : $\xi_p = [0.1, 0.1, \pi/6]^T$ et $q_r = [\pi/4, -\pi/2]^T$. Pour calculer la commande sous la forme (III.19), il faut calculer les valeurs numériques $\bar{J}(q_{r_1}, q_{r_2}, \vartheta)$ et $\bar{J}_t^{-1}(q_{r_1}, q_{r_2}, \vartheta)$. Le mouvement opérationnel imposé, à partir de $t_0 = 0$, est $\xi^*(t) = [\xi_1^*(t), \xi_2^*(t)]^T = [0.1t, 0.1t]^T$ et la vitesse opérationnelle $\dot{\xi}^*(t) = [\dot{\xi}_1^*(t), \dot{\xi}_2^*(t)]^T = [0.1, 0.1]^T$. Nous imposons les tâches additionnelles suivantes $\dot{x}^* = \dot{y}^* = \dot{\vartheta}^* = 0$. Autrement dit, nous désirons limiter le plus possible les mouvements de la plate-forme. Puis nous calculons la commande du manipulateur mobile d'après (III.19) :

$$u = \eta = \bar{J}_t^{-1}(\dot{\xi}_t^* + W_t(\xi_t^* - \xi_t)) \quad (\text{III.20})$$

La position de l'organe terminal ξ ainsi que la position de la plate-forme (x, y) sont respectivement reportées sur les figures (III.4), (III.5), (III.6) et (III.7). L'orientation ϑ de la plate-forme ainsi que les variations des différents angles d'articulation q_{r_1} et q_{r_2} du bras manipulateur sont respectivement reportées sur les figures (III.8), (III.9) et (III.10) [60, 61].

III.5 Evitement d'obstacles

L'évitement d'obstacles est un comportement de base présent dans quasiment tous les robots mobiles. Il est indispensable pour permettre au robot de fonctionner dans un

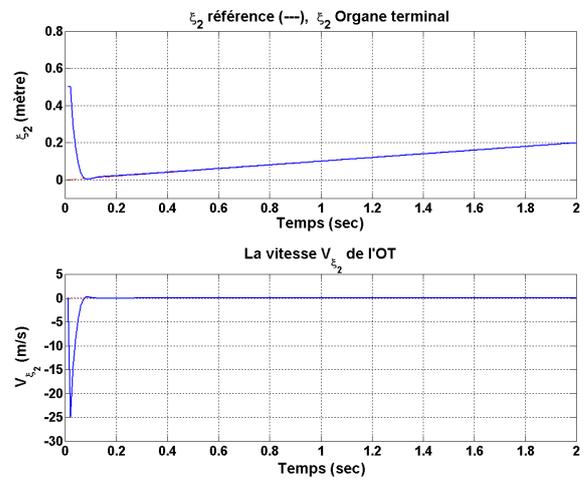
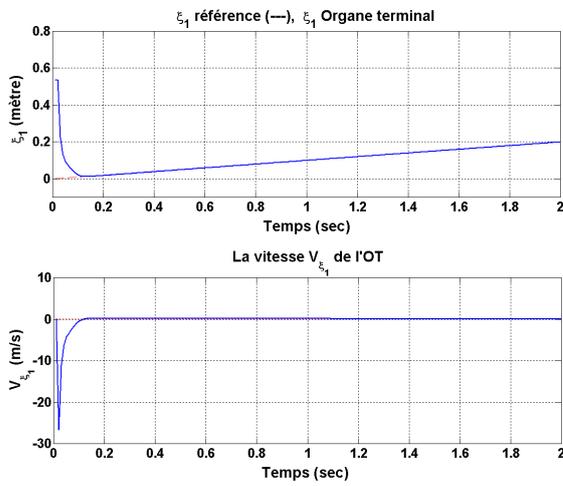


FIGURE III.4 : Position ξ_1 et vitesse $\dot{\xi}_1$ de l'organe terminal

FIGURE III.5 : Position ξ_2 et vitesse $\dot{\xi}_2$ de l'organe terminal

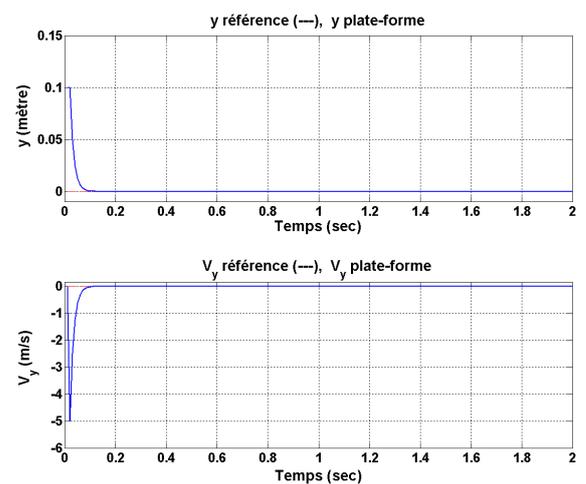
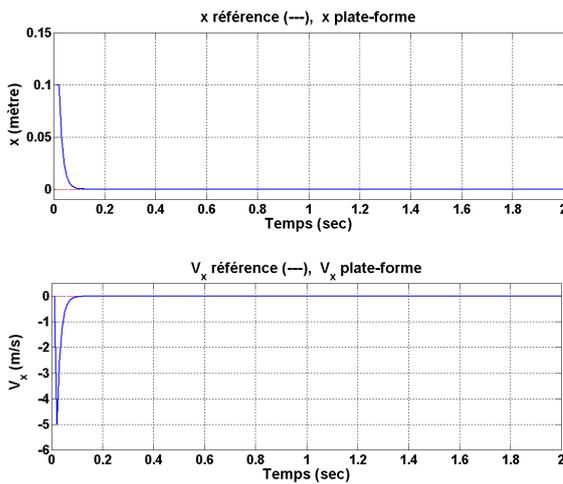


FIGURE III.6 : Position x et vitesse \dot{x} de la plate-forme

FIGURE III.7 : Position y et vitesse \dot{y} de la plate-forme

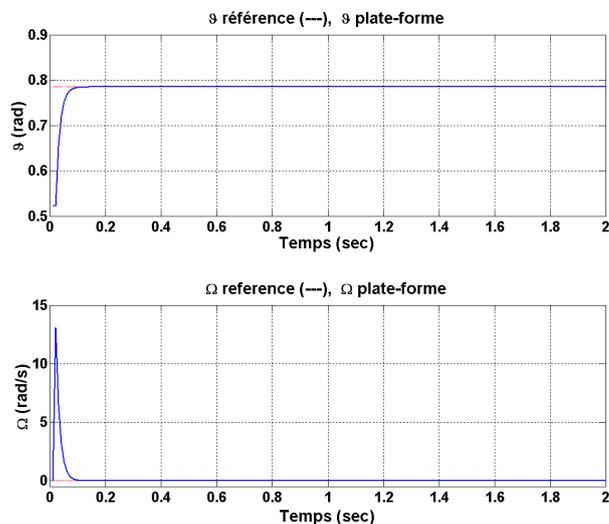


FIGURE III.8 : Orientation ϑ et vitesse angulaire $\dot{\vartheta}$ de la plate-forme

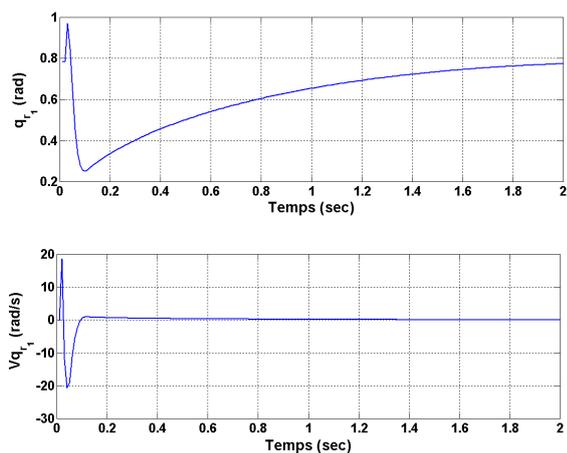


FIGURE III.9 : Orientation q_{r_1} et vitesse angulaire \dot{q}_{r_1} du bras1

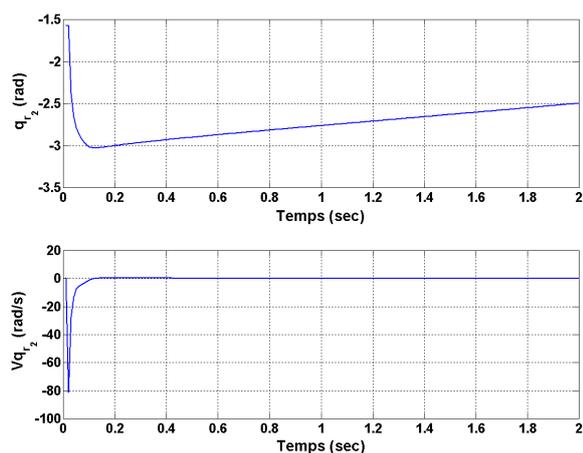


FIGURE III.10 : Orientation q_{r_2} et vitesse angulaire \dot{q}_{r_2} du bras2

environnement dynamique et pour gérer les écarts entre le modèle interne et le monde réel. Nous présentons dans la section suivante quelques méthodes qui sont efficaces à condition d'avoir une perception correcte de l'environnement.

III.5.1 Méthodes d'évitement d'obstacles

III.5.1.1 Méthode des champs de potentiel

La technique consiste à établir un champ de potentiel à partir d'une carte des obstacles donnée a priori. Chaque obstacle génère une force répulsive, le but génère une force attractive et la résultante de ces forces permet de construire une carte vectorielle. Ainsi, si l'animat sait déterminer à chaque instant sa position dans la carte, il peut déduire la suite de déplacements élémentaires le conduisant au but en évitant les obstacles. Khatib [24] est le premier à avoir appliqué cette approche à la planification de chemins et à l'évitement d'obstacles. Le robot est considéré comme une particule qui se déplace dans un champ de potentiel généré par le but et par les obstacles présents dans l'environnement. L'objectif génère un potentiel attractif alors que chaque obstacle génère un potentiel répulsif. La fonction de potentiel est définie de façon à ce que le potentiel soit élevé quand le robot est proche des obstacles ou du but et qu'il diminue quand il s'en éloigne. Le calcul du potentiel en chaque point du robot s'effectue en fonction de la distance de ce point aux obstacles ou au but.

Soit q représente la position du robot, le champ de potentiel où le robot se déplace est une fonction scalaire $U_{tot}(q)$ généré par la superposition de potentiels attractif et répulsif.

$$U_{tot}(q) = U_{att}(q) + U_{rep}(q) \quad (\text{III.21})$$

Le potentiel répulsif résultant de la superposition des potentiels répulsifs générés par les obstacles, et ainsi (III.21) peut être écrite comme :

$$U_{tot}(q) = U_{att}(q) + \sum_i U_{rep_i}(q) \quad (\text{III.22})$$

où $U_{rep_i}(q)$ représente le potentiel généré par l'obstacle i .

En utilisant le formalisme de Lagrange,

$$F_{tot}(q) = F_{att}(q) + F_{rep}(q) = -\nabla U_{att}(q) - \nabla U_{rep}(q) \quad (\text{III.23})$$

Le potentiel attractif est donné par :

$$U_{att}(q) = \frac{1}{2}k_{att}d_{but}^2(q) \quad (\text{III.24})$$

avec $d_{but}(q) = \|q - q_{but}\|$ est la distance Euclidienne entre le robot et le but, k_{att} est un facteur scalaire. Le gradient est donné par :

$$\nabla U_{att}(q) = k_{att}(q - q_{but}) \quad (\text{III.25})$$

La force attractive est considérée comme,

$$F_{att}(q) = -\nabla U_{att}(q) = -k_{att}(q - q_{but}) \quad (\text{III.26})$$

Pour le potentiel répulsif

$$U_{rep}(q) = \sum_i U_{rep_i}(q) \quad (\text{III.27})$$

avec

$$U_{rep_i}(q) = \begin{cases} \frac{1}{2}k_{obst_i} \left(\frac{1}{d_{obst_i}(q)} - \frac{1}{d_0} \right)^2 & si \quad d_{obst_i}(q) < d_0 \\ 0 & si \quad d_{obst_i}(q) \geq d_0 \end{cases} \quad (\text{III.28})$$

avec $d_{obst_i}(q)$ est la distance entre le robot et l'obstacle, k_{obst_i} est une constante et d_0 est la distance au delà de laquelle il n'y a plus répulsion.

La force répulsive, $F_{rep_i}(q) = -\nabla U_{rep_i}(q)$, est donnée par :

$$F_{rep_i}(q) = \begin{cases} k_{obst_i} \left(\frac{1}{d_{obst_i}(q)} - \frac{1}{d_0} \right) \frac{1}{d_{obst_i}^2(q)} \frac{q - q_{obst}}{d_{obst_i}} & si \quad d_{obst_i}(q) < d_0 \\ 0 & si \quad d_{obst_i}(q) \geq d_0 \end{cases} \quad (\text{III.29})$$

III.5.1.2 Méthode "Vector Field Histogram"

La méthode "Vector Field Histogram" introduite par Borenstein et Koren [82] est une méthode pour l'évitement d'obstacles en temps réel. Cette méthode a été conçue spécifiquement pour utiliser une grille d'occupation locale construite à partir de capteurs à ultrasons. Cette grille est construite de manière très rapide par la méthode "Histogrammic in motion mapping" [83] qui produit une grille dont chaque cellule contient un nombre d'autant plus élevé qu'elle a souvent été perçue comme contenant un obstacle (Fig. III.11). La grille est construite dans le référentiel du robot : un compteur est incrémenté pour chaque cellule appartenant au secteur angulaire dans lequel un obstacle a été détecté et les valeurs sont déplacées d'une cellule à l'autre en fonction des déplacements

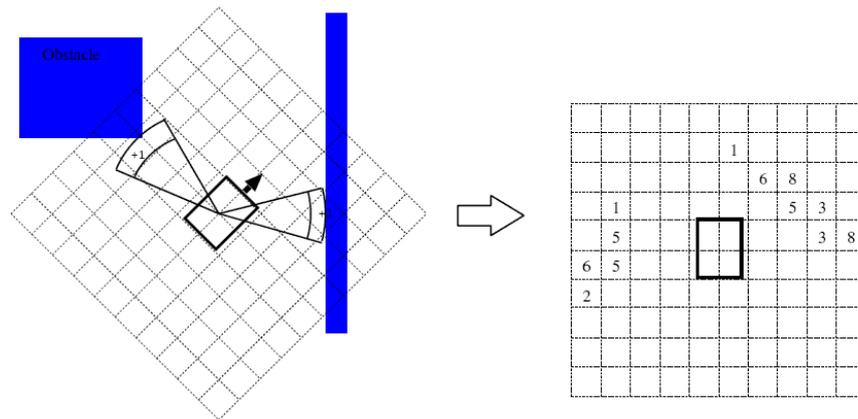


FIGURE III.11 : Grille d'occupation locale construite par la méthode "Histogrammic in motion mapping"

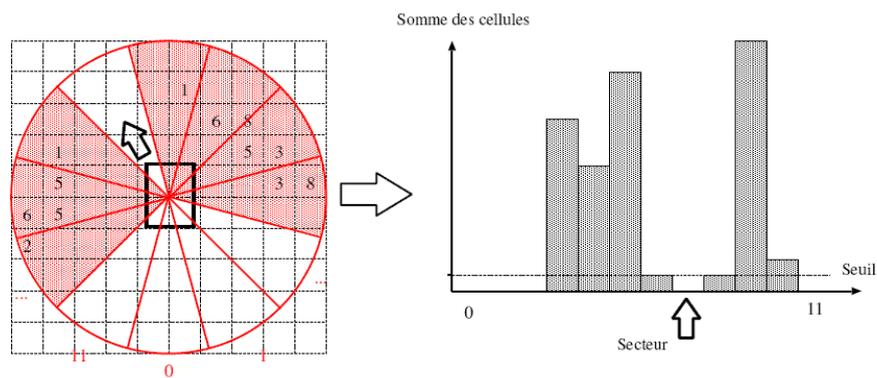


FIGURE III.12 : Utilisation de l'histogramme des obstacles pour déterminer la direction de déplacement du robot

du robot. Un histogramme représentant l'occupation de l'environnement autour du robot est ensuite construit à partir de cette grille d'occupation locale. Pour cela, l'environnement est discrétisé en secteurs angulaires pour lesquels la somme des valeurs des cellules est calculée (Fig. III.12). Un seuil permettant de tolérer un certain bruit et ensuite utilisé pour déterminer les directions possibles pour le robot : toutes les directions dont la valeur est inférieure au seuil sont considérées. Le choix de la direction est finalement réalisé parmi les directions possibles en fonction de contraintes externes (par exemple la direction la plus proche de la direction du but).

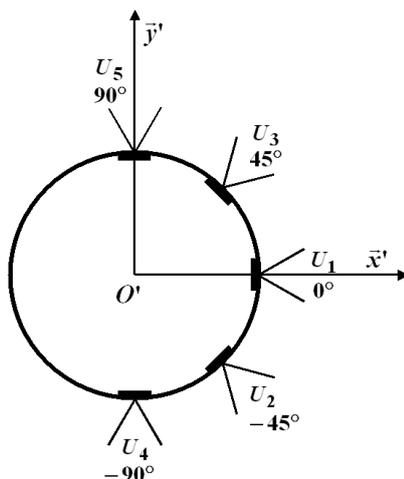


FIGURE III.13 : Le système de perception US du robot

III.5.2 Commande en impédance

L'objectif du contrôle exige des demandes multiples, recherche des buts, évitement d'obstacles et coopération avec d'autres robots. Afin de résoudre les problèmes évoqués plus haut, nous proposons en plus de la commande cinématique une boucle externe de contrôle basée sur la technique du contrôle en impédance. Le concept de contrôle de l'impédance a été introduit initialement dans les travaux de Hogan [84], puis modifié dans [85]. Ce concept est une généralisation des techniques classiques connues sous le nom de commande de la raideur, ou de la viscosité. Il consiste à calculer l'accroissement nécessaire pour corriger la trajectoire désirée. Ce dernier est fonction du rapport de l'effort mesuré et de l'impédance désirée. Plusieurs propositions ont été faites pour inclure le modèle dynamique dans la commande, nous citons les travaux dans [86][87][88][89].

III.5.2.1 Implémentation de la commande en impédance

Lorsque l'on divise l'espace du robot mobile en plusieurs sous-espaces, nous pouvons analyser puis résoudre le problème de navigation du robot vers le but final dans un environnement pourvu d'obstacles. En effet, ceci est possible en utilisant un système de perception qui permet une analyse parfaite des différentes configurations des obstacles. Dans notre cas, cinq capteurs à ultrasons arrangés en ceinture sur l'avant et sur les cotés et distant de 45 degrés sont utilisés comme le montre la figure (Fig.III.13). La portée maximale de chaque capteur est un peu plus de 3 m.

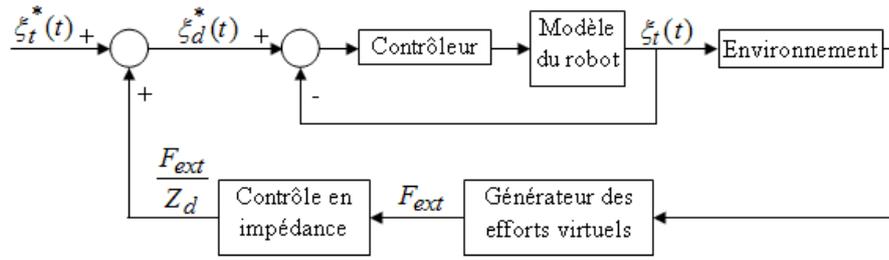


FIGURE III.14 : Le schéma bloc pour le système de contrôle

Cette partie est dédiée à l'écriture de la loi de commande assurant la relation d'impédance en boucle fermée. Soient ξ_t , $\dot{\xi}_t$, $\ddot{\xi}_t$ la position, vitesse et accélération du robot, ξ_t^* , $\dot{\xi}_t^*$, $\ddot{\xi}_t^*$ l'équivalent des mêmes variables désirées.

La figure (III.14) représente le schéma bloc proposé pour le système de contrôle, avec ξ_t^* le vecteur de la trajectoire désirée et ξ_d^* le vecteur de la trajectoire modifiée. Nous remarquons que la boucle externe fonctionne principalement pour prendre en compte les obstacles.

Autour du point ξ_t^* , la relation d'impédance à vérifier est de la forme [90, 85] :

$$M_d(\ddot{\xi}_t^* - \ddot{\xi}_t) + B_d(\dot{\xi}_t^* - \dot{\xi}_t) + K_d(\xi_t^* - \xi_t) = -F_{ext} \quad (\text{III.30})$$

avec $F_{ext} = \sum_i F_{obs_i}$ représente toutes les forces répulsives exercées sur le robot, F_{obs_i} est la force répulsive due à la présence de l'obstacle i . M_d , B_d , K_d les matrices d'inertie, de frottement et de raideur de l'impédance. Ces matrices sont souvent choisies diagonales et définies positives en fonction du comportement désiré.

Selon la relation (III.30), nous pouvons constater qu'en l'absence des obstacles $F_{ext} = 0$, ce qui implique que $\xi_t^* = \xi_t$. En revanche, en présence des obstacles, la trajectoire désirée ξ_t^* doit être régulée pour vérifier la même relation (III.30). Cet avantage nous permet d'utiliser cette technique aussi bien en mouvements libres qu'en mouvements contraints. La relation (III.30) peut être réécrite en regroupant les termes de l'impédance désirée Z_d sous la forme :

$$(\xi_t^* - \xi_t) + \frac{F_{ext}}{Z_d} = e_t + e_1 = 0 \quad (\text{III.31})$$

avec $Z_d = M_d p^2 + B_d p + K_d$ est l'impédance désirée, $p \equiv \frac{d}{dt}$ la variable de Laplace, $e_t = (\xi_t^* - \xi_t)$, $e_1 = \frac{F_{ext}}{Z_d}$. Il est clair que si nous remplaçons la relation (III.31) par le

nouveau vecteur d'erreur e_d , nous obtenons :

$$e_d = (\xi_t^* - \xi_t) + \frac{F_{ext}}{Z_d} \quad (\text{III.32})$$

Il suffit alors, pour vérifier la relation d'impédance, d'imposer à e_d de tendre vers zéro. Ainsi, nous pouvons utiliser le signal d'erreur e_d dans le cas libre ($F_{ext} = 0$) ou le cas contraint (présence des obstacles) ($F_{ext} \neq 0$). La nouvelle trajectoire désirée ξ_d^* peut être écrite comme la somme de la trajectoire désirée ξ_t^* et le terme $\frac{F_{ext}}{Z_d}$, $\xi_d^*(t) = \xi_t^*(t) + \frac{F_{ext}}{Z_d}$.

III.6 Application sur les manipulateurs mobiles omnidirectionnels

III.6.1 Manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons

La loi de commande cinématique donnée par l'équation (III.20) ne prend pas en compte la présence des obstacles. Le mouvement désiré modifié est donné par $\xi_d^*(t) = \xi_t^*(t) + \frac{F_{ext}}{Z_d}$. La nouvelle loi de commande est donc donnée par :

$$u = \bar{J}_t^{-1}(\dot{\xi}_d^* + W_t(\xi_d^* - \xi_t)) \quad (\text{III.33})$$

de telle sorte que si le mouvement est libre $F_{ext} = 0$ et dans la présence des obstacles $F_{ext} = \sum_i F_{obs_i}$. A partir de (III.15) et (III.33), l'équation en boucle fermée est donnée par $\dot{e}_d + W_t e_d = 0$. Il suffit de choisir W_t définie positive pour s'assurer la stabilité de la commande.

La force F_{obs} est calculée comme [91, 82] :

$$F_{obs} = a_{F_{obs}} - b_{F_{obs}}(d_{obs}(t) - d_{min})^2 \quad (\text{III.34})$$

où $a_{F_{obs}}$ et $b_{F_{obs}}$ sont des constantes positives qui vérifient la condition suivante :

$$a_{F_{obs}} = b_{F_{obs}}(d_{max} - d_{min})^2 \quad (\text{III.35})$$

d_{max} est la distance maximale entre le robot et l'obstacle détecté qui cause une force répulsive non nulle, $F_{obs} = 0$ au dehors de la zone répulsive ; d_{min} représente la distance

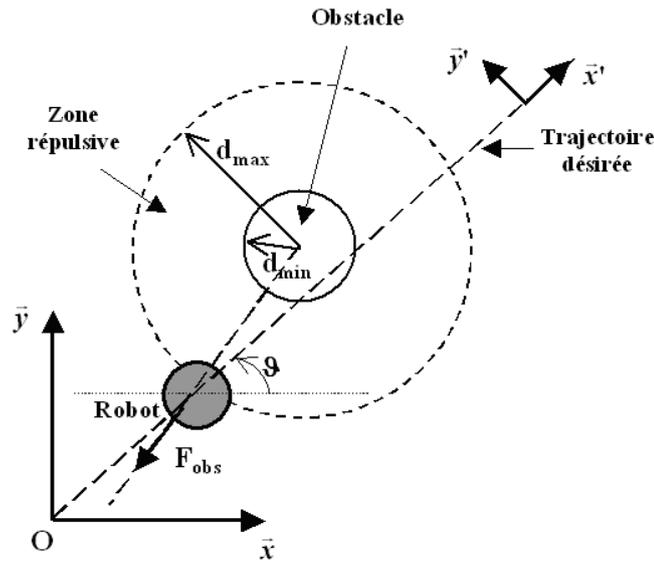


FIGURE III.15 : La force répulsive causée par l'obstacle

minimale pour un non-contact entre le robot et l'obstacle ; $d_{obs}(t)$ est la distance mesurée entre le robot et l'obstacle $d_{min} < d_{obs}(t) < d_{max}$ (Fig. III.15).

Dans une deuxième simulation, la configuration initiale du manipulateur mobile est telle que : $\xi_p = [0.1, 0.1, \pi/6]^T$ et $q_r = [\pi/4, -\pi/2]^T$. Le mouvement opérationnel imposé est $\xi^*(t) = [\xi_1^*(t), \xi_2^*(t)]^T = [0.1t, 0.1t]^T$ et la vitesse opérationnelle $\dot{\xi}^*(t) = [\dot{\xi}_1^*(t), \dot{\xi}_2^*(t)]^T = [0.1, 0.1]^T$. Nous imposons les tâches additionnelles suivantes $[x^*, y^*, \vartheta^*]^T = [0.1t, 0.1t, \pi/4]^T$. Les figures (III.16) et (III.17) montrent la trajectoire du manipulateur mobile dans le cas d'un mouvement libre. L'orientation ϑ de la plate-forme ainsi que les variations des différents angles d'articulation q_{r_1} et q_{r_2} du bras manipulateur sont respectivement reportées sur les figures (III.18), (III.19) et (III.20). Dans un deuxième temps, lorsque nous plaçons un obstacle sur le chemin de la plate-forme nous remarquons, d'après la figure (III.21), que la plate-forme réussit à contourner l'obstacle tout en maintenant le point terminal sur la trajectoire de référence. La figure (III.22) montre la nouvelle trajectoire de la plate-forme. L'orientation ϑ de la plate-forme ainsi que les variations des différents angles d'articulation q_{r_1} et q_{r_2} du bras manipulateur sont respectivement reportées sur les figures (III.23), (III.24) et (III.25). Les figures (III.26) à (III.30) montrent le cas de deux obstacles, [60, 61].

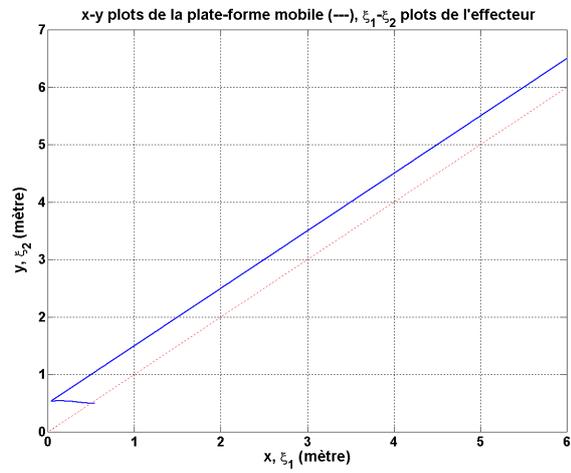
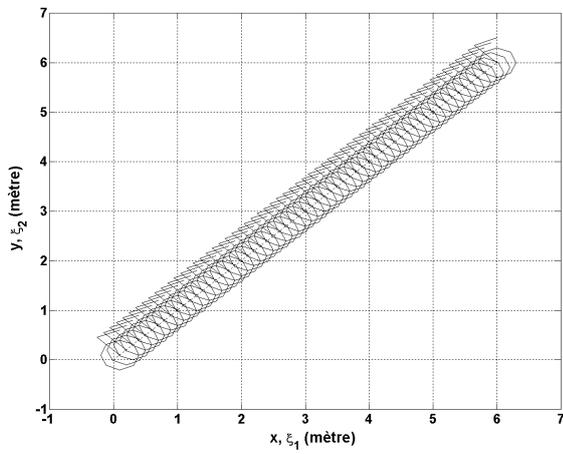


FIGURE III.16 : L'évolution du bras ma- **FIGURE III.17** : Trajectoires de l'effecteur
nipulateur et de la plate-forme dans le cas et de la plate-forme
d'un mouvement libre

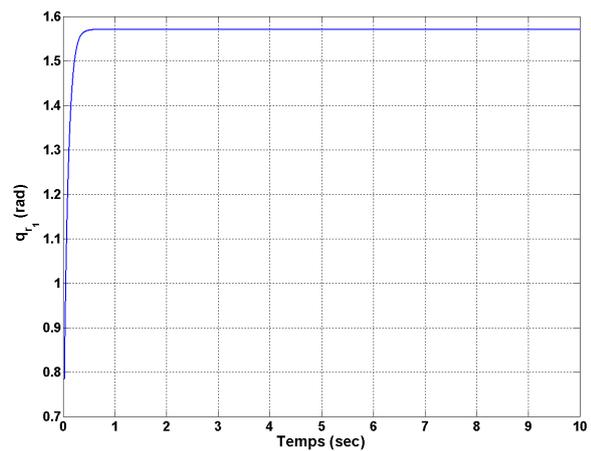
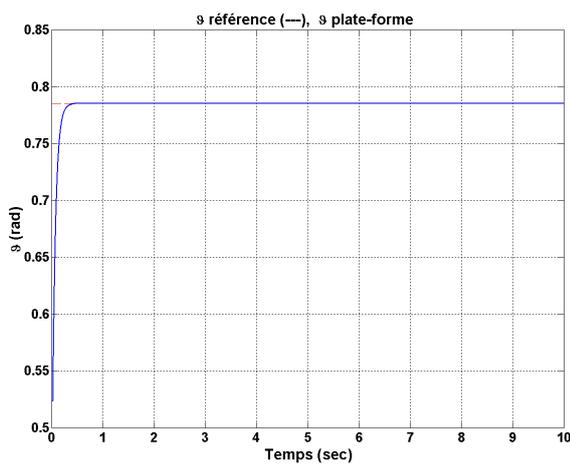


FIGURE III.18 : Courbe d'évolution de **FIGURE III.19** : Courbe d'évolution de
l'orientation ϑ l'articulation q_{r1}

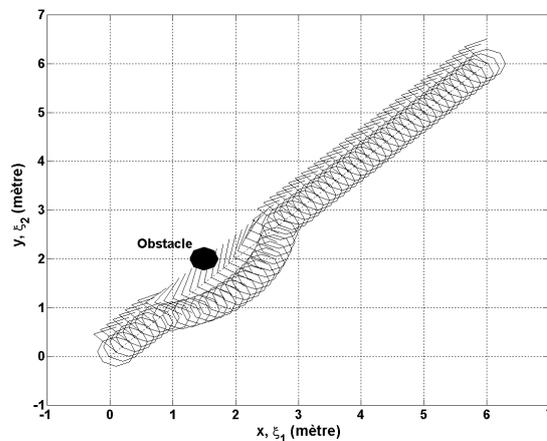
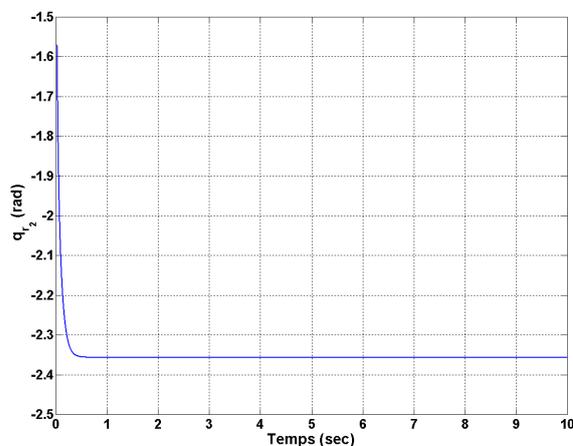


FIGURE III.20 : Courbe d'évolution de l'articulation q_{r_2}

FIGURE III.21 : L'évolution du bras manipulateur et de la plate-forme dans le cas d'un seul obstacle

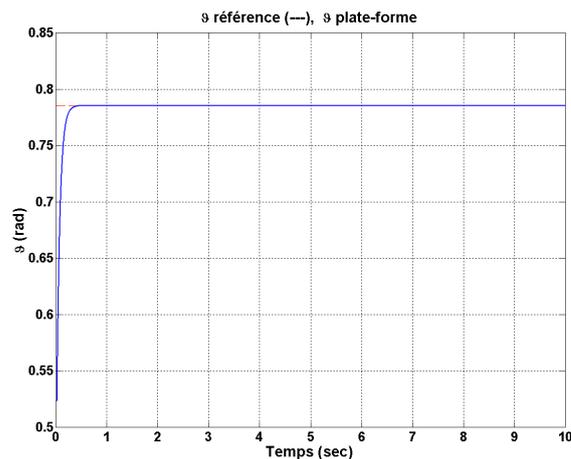
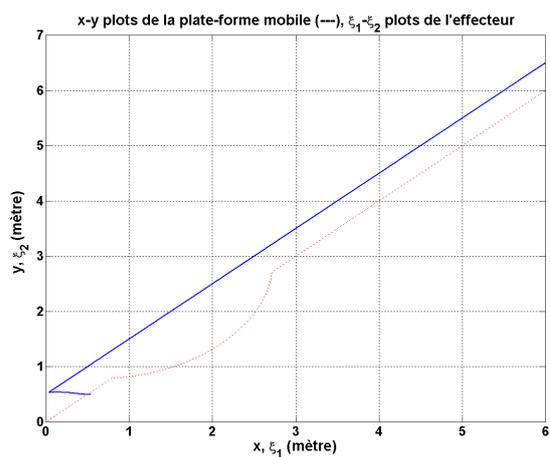


FIGURE III.22 : Trajectoires de l'effecteur et de la plate-forme

FIGURE III.23 : Courbe d'évolution de l'orientation ϑ

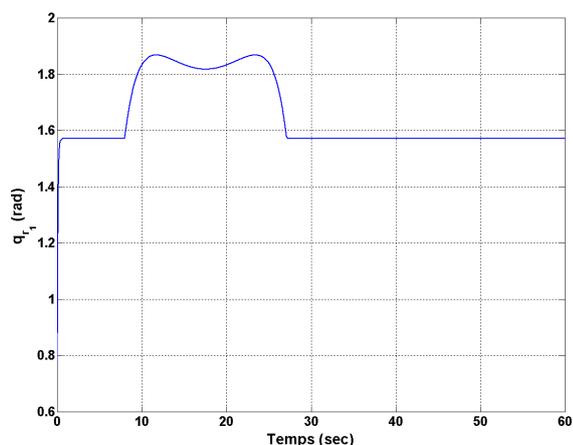


FIGURE III.24 : Courbe d'évolution de l'articulation q_{r1}

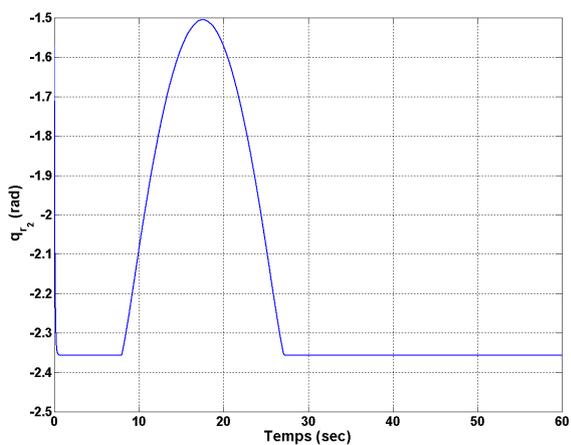


FIGURE III.25 : Courbe d'évolution de l'articulation q_{r2}

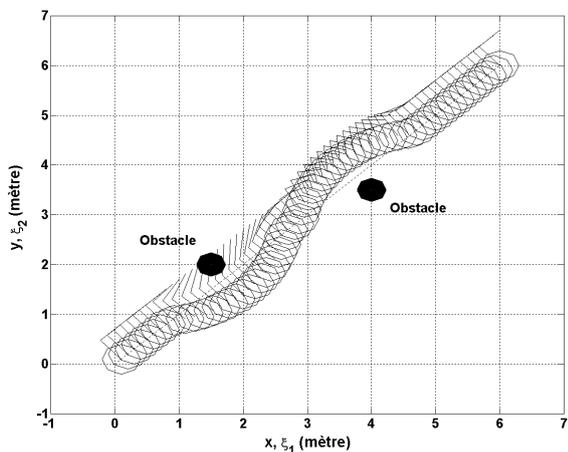


FIGURE III.26 : L'évolution du bras manipulateur et de la plate-forme dans le cas de deux obstacles

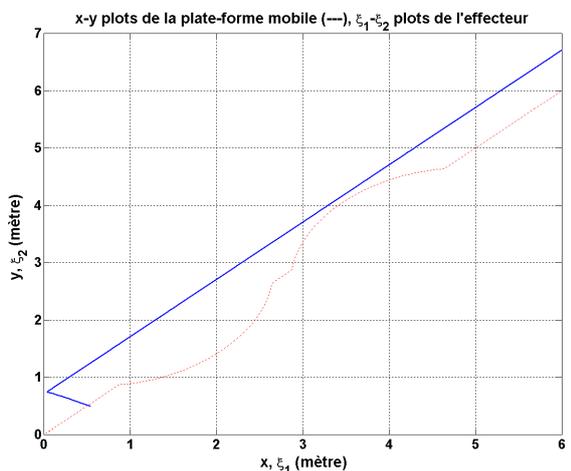


FIGURE III.27 : Trajectoires de l'effecteur

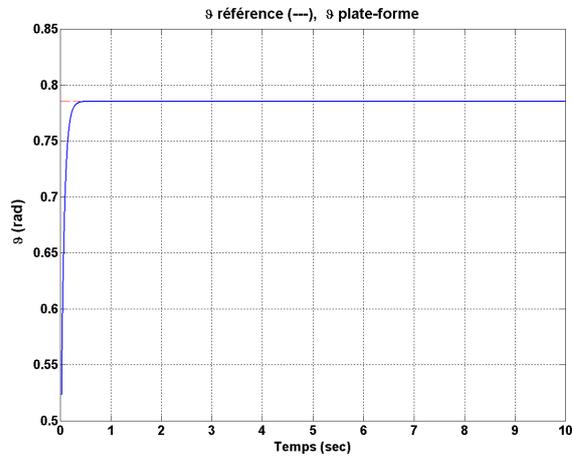


FIGURE III.28 : Courbe d'évolution de l'orientation ϑ

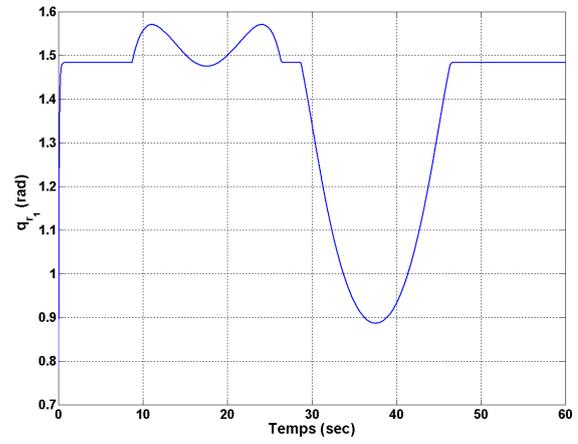


FIGURE III.29 : Courbe d'évolution de l'articulation q_{r_1}

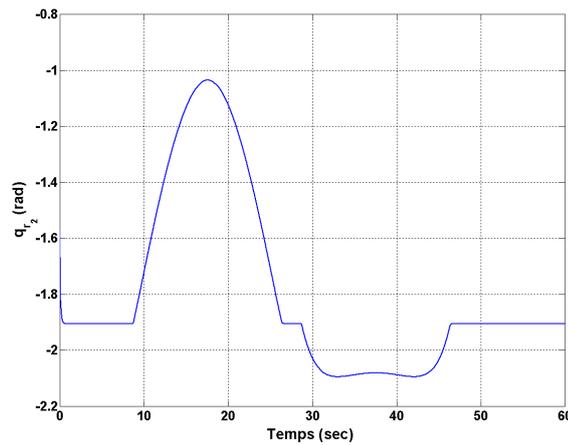


FIGURE III.30 : Courbe d'évolution de l'articulation q_{r_2}

III.6.2 Manipulateur mobile omnidirectionnel avec un bras manipulateur à trois liaisons rotoïdes

Pour comparer nous avons traité le cas du manipulateur mobile composé d'une plateforme de type omnidirectionnel et un bras manipulateur à trois liaisons rotoïdes. Le bras manipulateur est constitué de trois corps, de longueurs l_1 , l_2 et l_3 articulés autour de trois liaisons rotoïdes. Le premier corps effectue une rotation autour de l'axe \vec{z}_{r_0} , et les deux autres effectuent leurs rotations autour de \vec{z}_{r_1} et \vec{z}_{r_2} respectivement. O_{r_0} est placé à la base du bras manipulateur et O_{r_3} est le centre de l'OT. La configuration du bras manipulateur est donnée par les angles de rotation q_{r_1} , q_{r_2} et q_{r_3} des trois liaisons, donc $q_r = [q_{r_1}, q_{r_2}, q_{r_3}]^T$ de dimension $n_r = 3$. La situation de l'organe terminal est définie par la position du point O_{r_3} dans le repère \mathcal{R}_{r_0} , soit $\xi_r = [x_r, y_r, z_r]^T$ de dimension $m_r = 3$. Le modèle géométrique direct (MGD) du bras manipulateur est :

$$\begin{cases} x_r = \cos(q_{r_1})(l_2 \cos(q_{r_2}) + l_3 \cos(q_{r_2} + q_{r_3})) \\ y_r = \sin(q_{r_1})(l_2 \cos(q_{r_2}) + l_3 \cos(q_{r_2} + q_{r_3})) \\ z_r = l_1 - l_2 \sin(q_{r_2}) - l_3 \sin(q_{r_2} + q_{r_3}) \end{cases} \quad (\text{III.36})$$

En dérivant les équations (III.36), nous trouvons le MCD du bras manipulateur.

$$\begin{cases} \dot{x}_r = -(l_2 \sin(q_{r_1}) \cos(q_{r_2}) + l_3 \sin(q_{r_1}) \cos(q_{r_2} + q_{r_3})) \dot{q}_{r_1} \\ \quad - (l_2 \cos(q_{r_1}) \sin(q_{r_2}) + l_3 \cos(q_{r_1}) \sin(q_{r_2} + q_{r_3})) \dot{q}_{r_2} \\ \quad - (l_3 \cos(q_{r_1}) \sin(q_{r_2} + q_{r_3})) \dot{q}_{r_3} \\ \dot{y}_r = + (l_2 \cos(q_{r_1}) \cos(q_{r_2}) + l_3 \cos(q_{r_1}) \cos(q_{r_2} + q_{r_3})) \dot{q}_{r_1} \\ \quad - (l_2 \sin(q_{r_1}) \sin(q_{r_2}) + l_3 \sin(q_{r_1}) \sin(q_{r_2} + q_{r_3})) \dot{q}_{r_2} \\ \quad - (l_3 \sin(q_{r_1}) \sin(q_{r_2} + q_{r_3})) \dot{q}_{r_3} \\ \dot{z}_r = - (l_2 \cos(q_{r_2}) + l_3 \cos(q_{r_2} + q_{r_3})) \dot{q}_{r_2} - (l_3 \cos(q_{r_2} + q_{r_3})) \dot{q}_{r_3} \end{cases} \quad (\text{III.37})$$

sous forme matricielle,

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{z}_r \end{bmatrix} = \begin{bmatrix} -l_2 S_1 C_2 - l_3 S_1 C_{23} & -l_2 C_1 S_2 - l_3 C_1 S_{23} & -l_3 C_1 S_{23} \\ +l_2 C_1 C_2 + l_3 C_1 C_{23} & -l_2 S_1 S_2 - l_3 S_1 S_{23} & -l_3 S_1 S_{23} \\ 0 & -l_2 C_2 - l_3 C_{23} & -l_3 C_{23} \end{bmatrix} \begin{bmatrix} \dot{q}_{r_1} \\ \dot{q}_{r_2} \\ \dot{q}_{r_3} \end{bmatrix} \quad (\text{III.38})$$

avec $C_i = \cos(q_{r_i})$, $S_i = \sin(q_{r_i})$, $C_{ij} = \cos(q_{r_i} + q_{r_j})$ et $S_{ij} = \sin(q_{r_i} + q_{r_j}) \forall i, j = 1, 2, 3$.

Le déterminant de la matrice jacobienne $J_r(q_r)$ vaut :

$$\det J_r(q_r) = l_2 l_3 S_3 (l_2 C_2 + l_3 C_{23})$$

Le degré de liberté local $d_r(q_r) = 3$ sauf pour les configurations singulières $q_{r_3} = 0 \text{ mod } \pi$ ou bien $l_2 C_2 + l_3 C_{23} = 0$, valeurs pour lesquelles la configuration est singulière. En dehors de ces configurations le degré de liberté local $d_r(q_r) = 3$. Le degré de liberté global $D_r = 3$. Le bras manipulateur, qui possède trois actionneurs, n'est pas redondant vis-à-vis de la tâche en positionnement.

Nous allons restreindre l'espace opérationnel en ne considérant que la position de l'organe terminal $\xi = [\xi_1, \xi_2, \xi_3]^T$, soit $m_{rp} = 3$. Dans ce cas le MGD de ce manipulateur mobile est :

$$\left\{ \begin{array}{l} \xi_1 = x + (a_1 + \cos(q_{r_1})(l_2 \cos(q_{r_2}) + l_3 \cos(q_{r_2} + q_{r_3}))) \cos(\vartheta) - \\ \quad (a_2 + \sin(q_{r_1})(l_2 \cos(q_{r_2}) + l_3 \cos(q_{r_2} + q_{r_3}))) \sin(\vartheta) \\ \xi_2 = y + (a_1 + \cos(q_{r_1})(l_2 \cos(q_{r_2}) + l_3 \cos(q_{r_2} + q_{r_3}))) \sin(\vartheta) + \\ \quad (a_2 + \sin(q_{r_1})(l_2 \cos(q_{r_2}) + l_3 \cos(q_{r_2} + q_{r_3}))) \cos(\vartheta) \\ \xi_3 = z + l_1 - l_2 \sin(q_{r_2}) - l_3 \sin(q_{r_2} + q_{r_3}) \end{array} \right.$$

Nous prenons le cas où $a_1 = a_2 = 0$. La matrice jacobienne est donnée par :

$$\bar{J}(q_{r_1}, q_{r_2}, q_{r_3}, \vartheta) = \begin{bmatrix} -l_2 S_{1\vartheta} C_2 - l_3 S_{1\vartheta} C_{23} & -l_2 C_{1\vartheta} S_2 - l_3 C_{1\vartheta} S_{23} & -l_3 C_{1\vartheta} S_{23} \\ +l_2 C_{1\vartheta} C_2 + l_3 C_{1\vartheta} C_{23} & -l_2 S_{1\vartheta} S_2 - l_3 S_{1\vartheta} S_{23} & -l_3 S_{1\vartheta} S_{23} \\ 0 & -l_2 C_2 - l_3 C_{23} & -l_3 C_{23} \\ 1 & 0 & -l_2 S_{1\vartheta} C_2 - l_3 S_{1\vartheta} C_{23} \\ 0 & 1 & +l_2 C_{1\vartheta} C_2 + l_3 C_{1\vartheta} C_{23} \\ 0 & 0 & 0 \end{bmatrix}$$

Nous prenons l'exemple pour lequel $m_{rp} + m_{add} = \delta_m$. Donc, nous imposons m_{add} tâches additionnelles :

$$\xi_{add} = \begin{bmatrix} \xi_{add_1} \\ \xi_{add_2} \\ \xi_{add_3} \end{bmatrix} = \begin{bmatrix} x \\ y \\ \vartheta \end{bmatrix}$$

Nous avons donc :

$$\dot{\xi}_{add} = \begin{bmatrix} \dot{\xi}_{add_1} \\ \dot{\xi}_{add_2} \\ \dot{\xi}_{add_3} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\vartheta} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_{J_{add}} \begin{bmatrix} \dot{q}_{r_1} \\ \dot{q}_{r_2} \\ \dot{q}_{r_3} \\ \dot{x} \\ \dot{y} \\ \dot{\vartheta} \\ \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \end{bmatrix}$$

$$\bar{J}_{add}(\vartheta) = J_{add}(\vartheta)S(\vartheta) = J_{add}(\vartheta) \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{1}{r} \sin(\alpha_1 + \vartheta) & -\frac{1}{r} \cos(\alpha_1 + \vartheta) & -R_1/r \\ 0 & 0 & 0 & \frac{1}{r} \sin(\alpha_2 + \vartheta) & -\frac{1}{r} \cos(\alpha_2 + \vartheta) & -R_2/r \\ 0 & 0 & 0 & \frac{1}{r} \sin(\alpha_3 + \vartheta) & -\frac{1}{r} \cos(\alpha_3 + \vartheta) & -R_3/r \end{bmatrix}$$

soit :

$$\bar{J}_{add}(\vartheta) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

et :

$$\bar{J}_t(q_{r1}, q_{r2}, q_{r3}, \vartheta) = \begin{bmatrix} \bar{J}(q_{r1}, q_{r2}, q_{r3}, \vartheta) \\ \bar{J}_{add}(\vartheta) \end{bmatrix} = \begin{bmatrix} -l_2 S_{1\vartheta} C_2 - l_3 S_{1\vartheta} C_{23} & -l_2 C_{1\vartheta} S_2 - l_3 C_{1\vartheta} S_{23} & -l_3 C_{1\vartheta} S_{23} & 1 & 0 & -l_2 S_{1\vartheta} C_2 - l_3 S_{1\vartheta} C_{23} \\ +l_2 C_{1\vartheta} C_2 + l_3 C_{1\vartheta} C_{23} & -l_2 S_{1\vartheta} S_2 - l_3 S_{1\vartheta} S_{23} & -l_3 S_{1\vartheta} S_{23} & 0 & 1 & +l_2 C_{1\vartheta} C_2 + l_3 C_{1\vartheta} C_{23} \\ 0 & -l_2 C_2 - l_3 C_{23} & -l_3 C_{23} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Nous choisissons un exemple numérique $l_1 = 0.6 \text{ m}$, $l_2 = 0.4 \text{ m}$, $l_3 = 0.3 \text{ m}$ et où la configuration initiale du manipulateur mobile est telle que : $\xi_p = [0.1, 0.1, \pi/6]^T$ et $q_r = [\pi/4, -\pi/2, \pi/4]^T$. Pour calculer la commande sous la forme (III.19), il faut calculer les valeurs numériques $\bar{J}(q_{r1}, q_{r2}, \vartheta)$ et $\bar{J}_t^{-1}(q_{r1}, q_{r2}, \vartheta)$. Le mouvement opérationnel imposé est $\xi^*(t) = [\xi_1^*(t), \xi_2^*(t), \xi_3^*(t)]^T = [0.1t, 0.1t, 0.7 + 0.01t]^T$ et la vitesse opérationnelle $\dot{\xi}^*(t) = [\dot{\xi}_1^*(t), \dot{\xi}_2^*(t), \dot{\xi}_3^*(t)]^T = [0.1, 0.1, 0.01]^T$. Nous imposons les tâches additionnelles suivantes $[x^*, y^*, \vartheta^*]^T = [0.1t, 0.1t, \pi/4]^T$. Puis nous calculons la commande du manipulateur mobile d'après (III.19) :

$$u = \eta = \bar{J}_t^{-1}(\dot{\xi}_t^* + W_t(\xi_t^* - \xi_t))$$

Les figures (III.31) et (III.32) montrent la trajectoire du manipulateur mobile dans le cas d'un mouvement libre. L'orientation ϑ de la plate-forme ainsi que les variations des différents angles d'articulation q_{r1} , q_{r2} et q_{r3} du bras manipulateur sont respectivement reportées sur les figures (III.33), (III.34), (III.35) et (III.36). Dans un deuxième temps, lorsque nous plaçons un obstacle sur le chemin de la plate-forme nous remarquons, d'après la figure (III.37), que la plate-forme réussit à contourner l'obstacle tout en maintenant le point terminal sur la trajectoire de référence. La figure (III.38) montre la nouvelle trajectoire de la plate-forme. L'orientation ϑ de la plate-forme ainsi que les variations des différents angles d'articulation q_{r1} , q_{r2} et q_{r3} du bras manipulateur sont respectivement reportées sur les figures (III.39), (III.40), (III.41) et (III.42). Les figures (III.43) à (III.48) montrent le cas de deux obstacles [92].

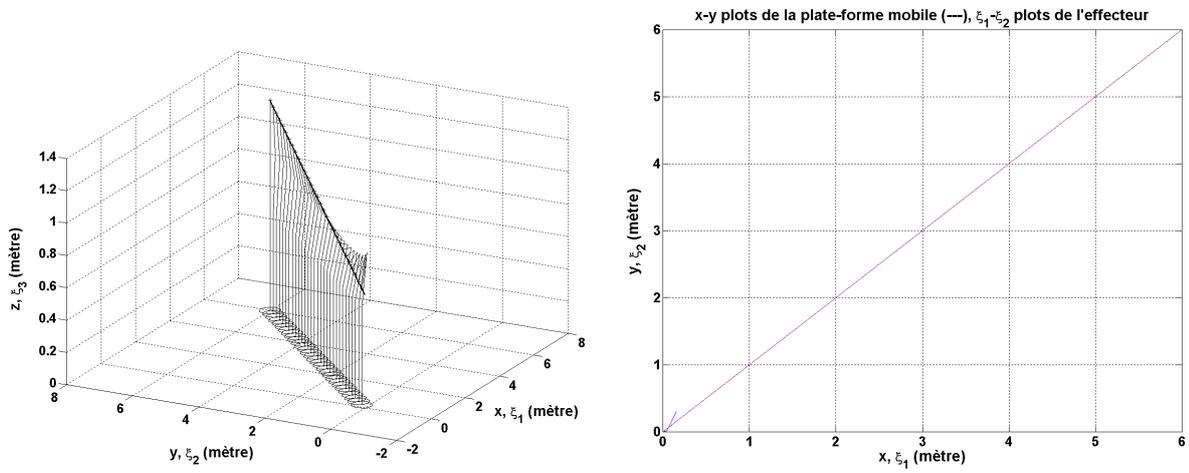


FIGURE III.31 : Vue 3D de l'évolution du bras manipulateur et de la plate-forme dans le cas d'un mouvement libre

FIGURE III.32 : Trajectoires de l'effecteur et de la plate-forme

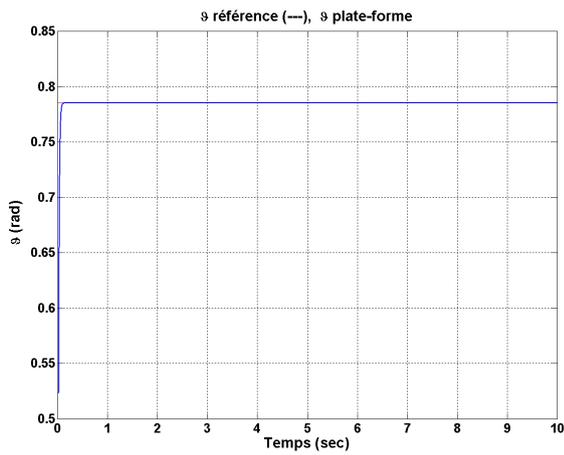


FIGURE III.33 : Courbe d'évolution de l'orientation ϑ

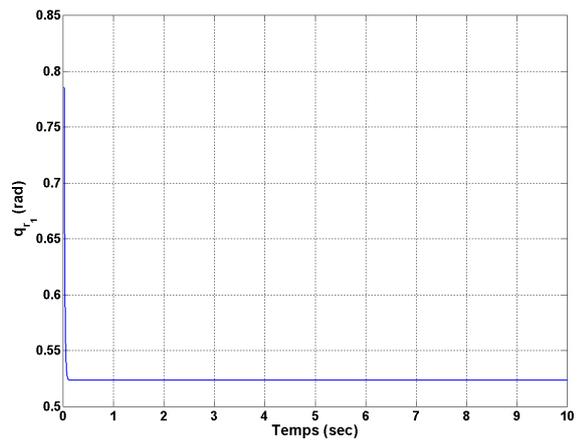


FIGURE III.34 : Courbe d'évolution de l'articulation q_{r_1}

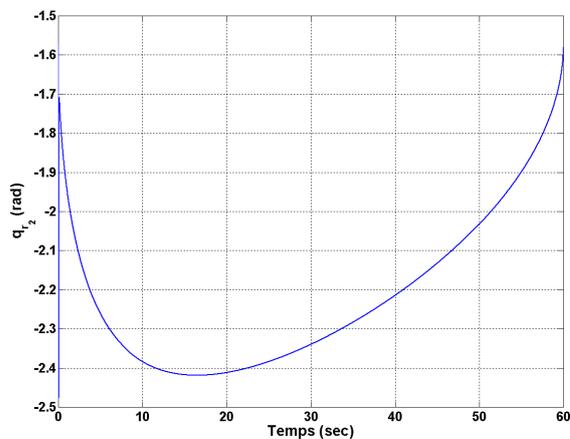


FIGURE III.35 : Courbe d'évolution de l'articulation q_{r_2}

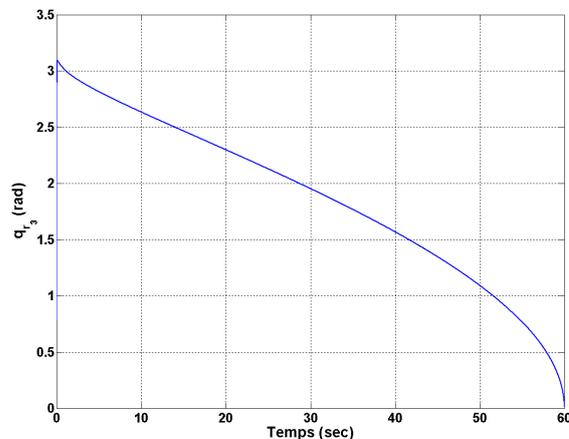


FIGURE III.36 : Courbe d'évolution de l'articulation q_{r_3}

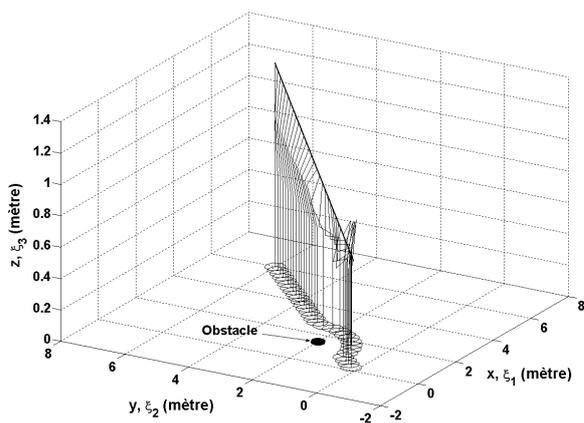


FIGURE III.37 : Vue 3D de l'évolution du bras manipulateur et de la plate-forme dans le cas d'un seul obstacle

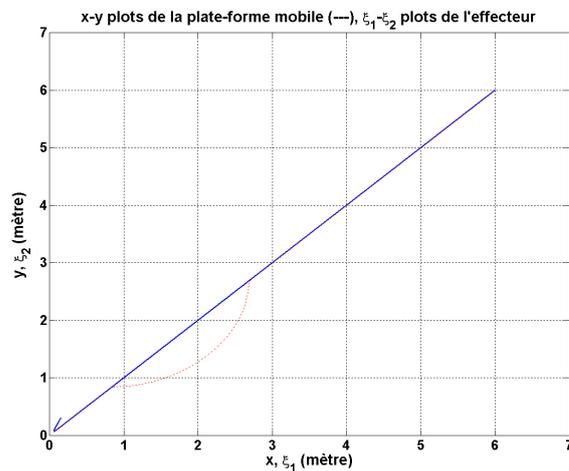


FIGURE III.38 : Trajectoires de l'effecteur de la plate-forme

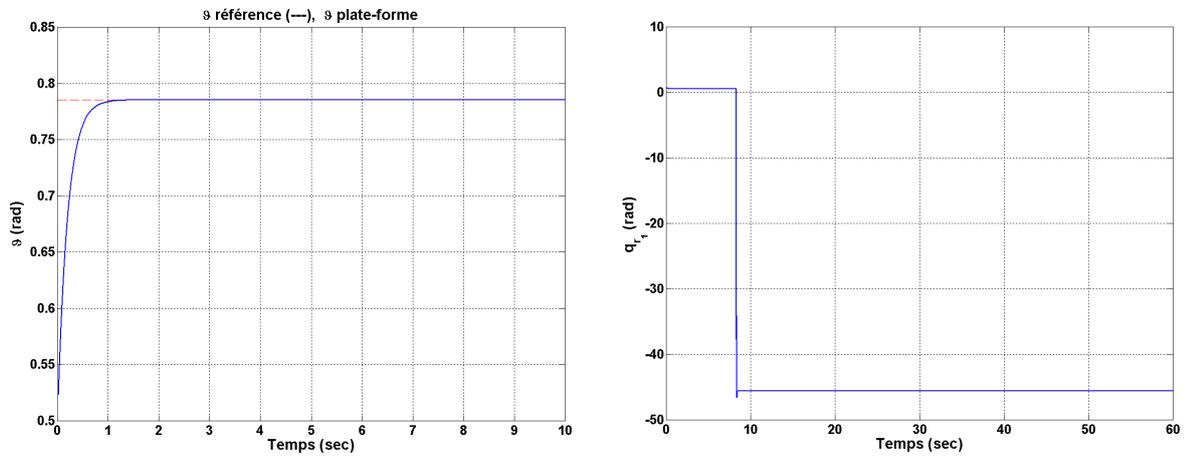


FIGURE III.39 : Courbe d'évolution de l'orientation ϑ FIGURE III.40 : Courbe d'évolution de l'articulation q_{r1}

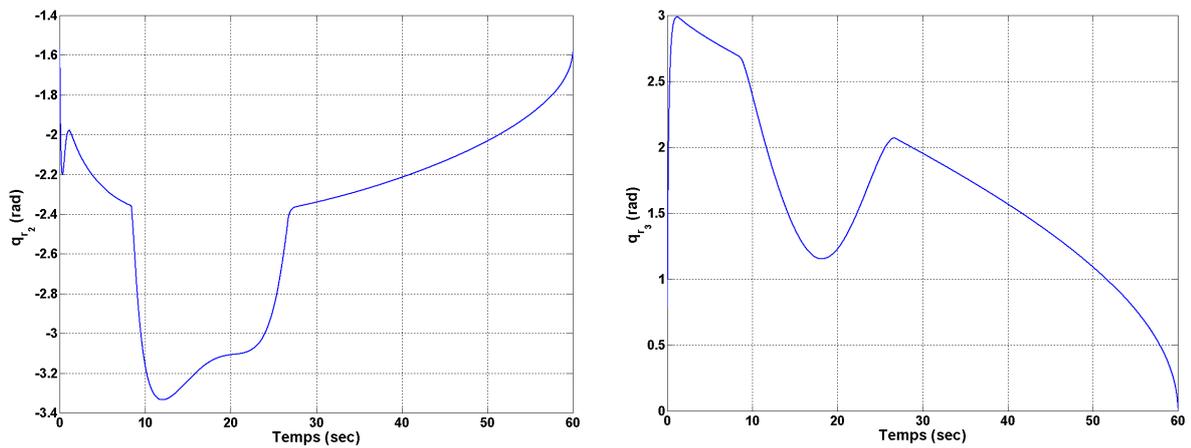


FIGURE III.41 : Courbe d'évolution de l'articulation q_{r2} FIGURE III.42 : Courbe d'évolution de l'articulation q_{r3}

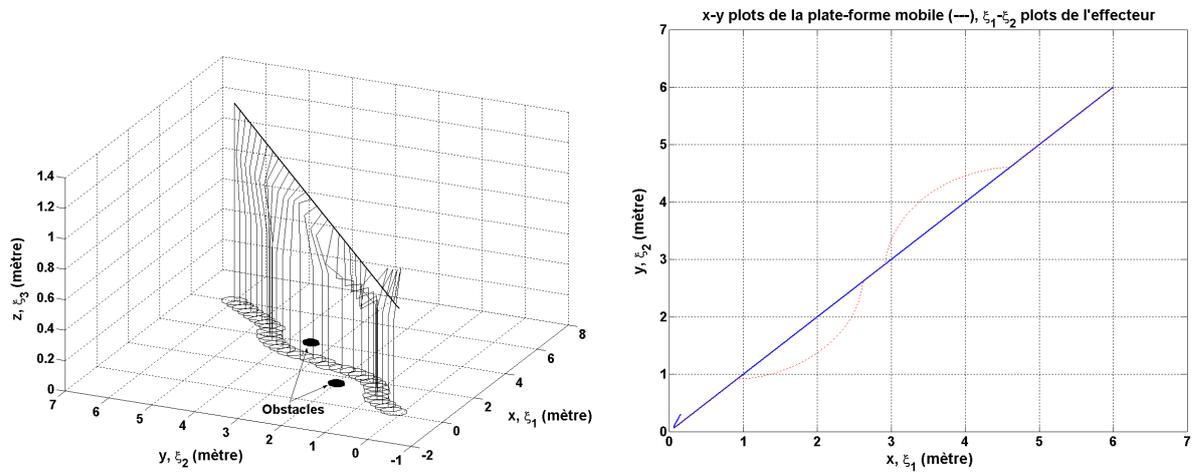


FIGURE III.43 : Vue 3D de l'évolution du bras manipulateur et de la plate-forme dans le cas de deux obstacles

FIGURE III.44 : Trajectoires de l'effecteur et de la plate-forme

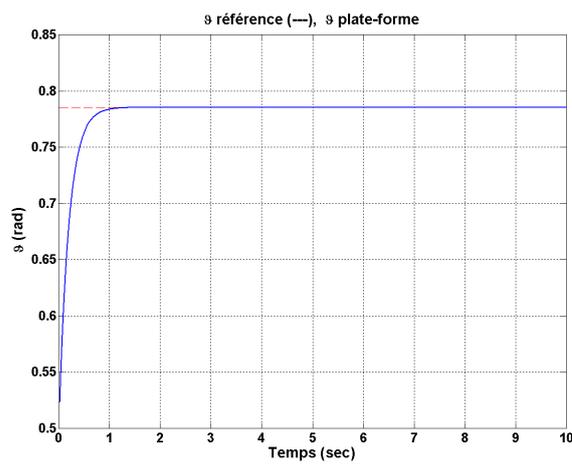


FIGURE III.45 : Courbe d'évolution de l'orientation ϑ

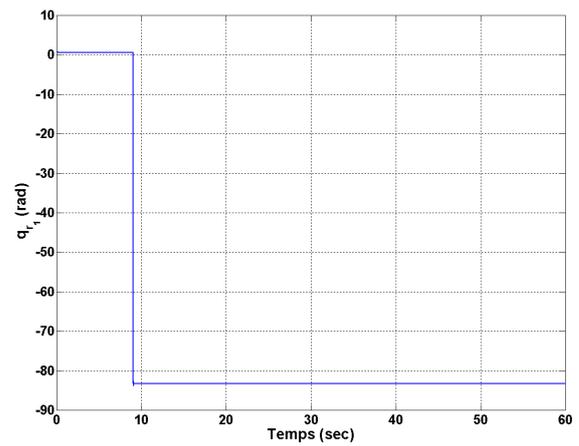


FIGURE III.46 : Courbe d'évolution de l'articulation q_{r_1}

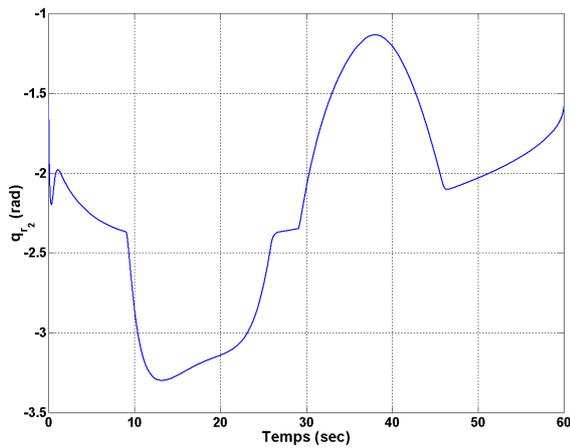


FIGURE III.47 : Courbe d'évolution de l'articulation q_{r_2}

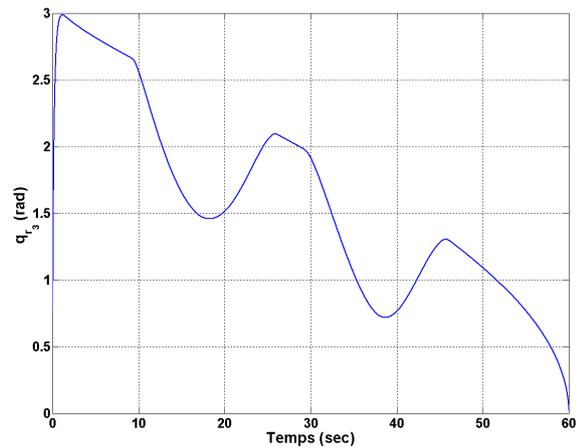


FIGURE III.48 : Courbe d'évolution de l'articulation q_{r_3}

III.7 Conclusion

Nous avons établi dans ce chapitre une stratégie de commande cinématique pour un manipulateur mobile évoluant à mouvement opérationnel imposé, à partir de la connaissance de son MCS. La méthode utilisée est celle des tâches additionnelles. Il s'agit d'une méthode d'inversion du MCS. Nous avons montré l'intérêt de cette méthode qui permette à la fois, dans la limite de la redondance du système vis-à-vis de la tâche, d'obtenir un mouvement opérationnel imposé et de résoudre des tâches additionnelles imposées. Nous avons pris l'exemple d'un manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons et l'exemple d'un manipulateur mobile omnidirectionnel avec un bras manipulateur à trois liaisons rotoïdes. Et pour réaliser l'objectif de l'évitement d'obstacles, nous avons proposé la stratégie du contrôle cinématique augmenté d'une boucle externe de contrôle en impédance.

Chapitre IV

Commande par retour d'état linéarisant

IV.1 Introduction

Notre travail, comme dans le chapitre 3, consiste à permettre au manipulateur mobile d'évoluer dans un environnement encombré. Il doit aussi coopérer avec d'autres robots. Dans le chapitre 3, nous nous sommes limités à la cinématique. Mais cette dernière n'est pas suffisante dans le cas d'interaction entre robots. Dans ce chapitre, nous allons voir comment transformer le modèle non linéaire du manipulateur mobile omnidirectionnel dans une forme linéaire plus pratique pour implémenter les commandes souhaitées. Les développements détaillés des théories de la commande non linéaire ainsi que des exemples d'application peuvent être retrouvés dans plusieurs publications Isidori [93], Slotine [94], Sastri [95], etc. Cependant, afin de faciliter la compréhension des développements présentés dans ce chapitre, nous allons rappeler certaines définitions et théorèmes et montrer les procédures à suivre pour réaliser une commande linéarisante pour le robot mobile omnidirectionnel et pour le manipulateur mobile omnidirectionnel. Nous allons ajouter la stratégie du contrôle dynamique augmenté d'une boucle externe de contrôle en impédance pour l'évitement d'obstacles.

IV.2 Rappel sur la commande par retour d'état linéarisant

IV.2.1 Théorie de la linéarisation par retour d'état

IV.2.1.1 Difféomorphisme

Définition IV.1. [94] Soit X^0 un point d'opération et $U \subset \mathbb{R}^n$ un voisinage de X^0 . L'application $\phi : \mathbb{R}^n \rightarrow \mathbb{R}^n$, définie dans la région U est appelée *difféomorphisme local* si elle est différentiable et son inverse ϕ^{-1} existe et est différentiable. De plus, si $U = \mathbb{R}^n$ alors ϕ est un *difféomorphisme global*.

IV.2.1.2 Champ de vecteurs

Un champ de vecteurs sur \mathbb{R}^n est une fonction différentiable :

$$f(X) : \mathbb{R}^n \rightarrow \mathbb{R}^n \quad f(X) = \begin{bmatrix} f_1(X) \\ f_2(X) \\ \cdot \\ \cdot \\ \cdot \\ f_n(X) \end{bmatrix} \quad (\text{IV.1})$$

IV.2.1.3 Dérivée de Lie

Définition IV.2. [94] Soit $h : \mathbb{R}^n \rightarrow \mathbb{R}$, une application différentiable et $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, un champ de vecteurs défini dans \mathbb{R}^n . L'application scalaire représentée par la dérivée de h le long du champ f et définie par :

$$L_f h = \nabla h f \quad (\text{IV.2})$$

est appelée *dérivée de Lie* dans la direction de f . Le vecteur ligne ∇h représente le gradient de h par rapport à un vecteur X et est décrit par :

$$\nabla h = \frac{\partial h}{\partial X} = \left[\frac{\partial h}{\partial x_1}, \dots, \frac{\partial h}{\partial x_n} \right] \quad (\text{IV.3})$$

Nous pouvons définir les relations suivantes :

$$L_f^0 h = h \quad L_f^1 h = L_f h \quad (\text{IV.4})$$

$$L_f^i h = \nabla(L_f^{i-1} h) f \quad \text{pour } i = 1, 2, \dots \quad (\text{IV.5})$$

IV.2.1.4 Crochet de Lie

Définition IV.3. [94] Soit f et g deux champs de vecteurs. Le crochet de Lie de f et g est un champ de vecteurs défini par :

$$[f, g] = \nabla g f - \nabla f g \quad (\text{IV.6})$$

où ∇g et ∇f représentent respectivement les gradients de g et f par rapport à X . Le crochet de Lie est représenté souvent par $ad_f g$. Nous définissons aussi :

$$ad_f^0 g = g \quad ad_f^i g = [f, ad_f^{i-1} g] \quad \text{pour } i = 1, 2, \dots \quad (\text{IV.7})$$

Les crochets de Lie ont les propriétés suivantes :

– Asymétrie

$$[f, g] = -[g, f] \quad (\text{IV.8})$$

– Bilinéarité

$$[\alpha f_1 + \beta f_2, g] = \alpha [f_1, g] + \beta [f_2, g] \quad (\text{IV.9})$$

– Identité de Jacobi

$$[f_1, [f_2, f_3]] + [f_2, [f_3, f_1]] + [f_3, [f_1, f_2]] = 0 \quad (\text{IV.10})$$

IV.2.1.5 Involutivité

Soit $\Delta = \{f_1, \dots, f_n\}$ un ensemble de champs de vecteurs linéairement indépendants défini dans \mathfrak{R}^n .

– **Définition IV.4.** [94] L'ensemble Δ est involutif si et seulement s'il existe des fonctions scalaires $\alpha_{ijk} : \mathfrak{R}^n \rightarrow \mathfrak{R}$ tel que :

$$[f_i, f_j](X) = \sum_{k=1}^n \alpha_{ijk}(X) f_k(X) \quad \forall X \text{ et } \forall i, j \quad (\text{IV.11})$$

– **Définition IV.5.** *L'ensemble Δ est involutif si et seulement si :*

$$\forall f_i, f_j \in \Delta, i \neq j : [f_i, f_j] \in \Delta \quad (\text{IV.12})$$

L'involutivité exprime que le crochet de Lie de champs de vecteurs reste dans l'ensemble de départ.

IV.2.2 Linéarisation au sens entrée-état d'un système monovari- riable

Pour faciliter la description de la méthode, nous considérons, en premier lieu, le cas d'un système non linéaire monovarié. En ce sens, soit le système non linéaire d'ordre n décrit par les équations suivantes :

$$\begin{cases} \dot{X} = f(X) + g(X)u \\ y = h(X) \end{cases} \quad (\text{IV.13})$$

où $X \in \mathfrak{R}^n$ est le vecteur d'état du système, f et $g : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ sont des champs de vecteurs différentiables, u et y sont l'entrée et la sortie du système respectivement. Posons $X^0 \in \mathfrak{R}^n$ un point d'opération du système et $U \subset \mathfrak{R}^n$ un voisinage de X^0 .

Définition IV.6. [93] *Un système monovarié de la forme (IV.13) est linéarisable au sens entrée-état s'il existe une région $U \subset \mathfrak{R}^n$, un difféomorphisme $\phi : U \subset \mathfrak{R}^n$ et une loi de commande non linéaire $u = \Gamma(X) + \Lambda(X)v$ telles que les nouvelles variables d'état $z = \phi(X)$ et la nouvelle entrée v forme un système linéaire invariant dans le temps de la forme canonique suivante :*

$$\dot{z} = Az + Bv \quad (\text{IV.14})$$

où

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & 1 \\ 0 & 0 & 0 & \dots & \dots & 0 \end{bmatrix}_{(n \times n)} \quad B = \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix}_{(n \times 1)}$$

Le nouveau vecteur de coordonnées $\phi(X)$ et la loi de commande non linéaire u s'appellent respectivement l'état linéarisant et la commande linéarisante du système.

IV.2.2.1 Conditions pour une linéarisation au sens entrée-état

Théorème IV.7. [93] Soit un système de la forme (IV.13). Ce système est linéarisable au sens entrée-état si et seulement s'il existe un voisinage U telles que les conditions suivantes sont vérifiées :

- Les champs de vecteurs $\{g, ad_f g, \dots, ad_f^{m-1} g\}$ sont linéairement indépendants dans U .
- L'ensemble de vecteurs $\{g, ad_f g, \dots, ad_f^{m-2} g\}$ est involutif dans U .

La première condition peut être interprétée comme une condition de commandabilité du système non linéaire.

IV.2.2.2 Etapes à suivre

La linéarisation au sens entrée-état est réalisée en quatre étapes :

- La première consiste à construire les champs de vecteurs $\{g, ad_f g, \dots, ad_f^{n-1} g\}$.
- La deuxième consiste à vérifier si les conditions de commandabilité et d'involutivité sont satisfaites.
- La troisième consiste à choisir le premier élément z_1 du nouveau vecteur d'état z de façon à respecter les conditions suivantes :

$$\nabla_{z_1} ad_f^i g = 0 \quad \text{pour } i = 0, \dots, (n-2) \quad (\text{IV.15})$$

$$\nabla_{z_1} ad_f^{n-1} g \neq 0 \quad (\text{IV.16})$$

- La quatrième consiste à construire le nouveau vecteur d'état $z = [z_1 \ L_f z_1 \ \dots \ L_f^{n-1} z_1]^T$ ainsi que la transformation de la commande définie par $u = \Gamma(X) + \Lambda(X)v$:

$$\Gamma(X) = -\frac{L_f^n z_1}{L_g L_f^{n-1} z_1} \quad (\text{IV.17})$$

$$\Lambda(X) = \frac{1}{L_g L_f^{n-1} z_1} \quad (\text{IV.18})$$

où v est la nouvelle entrée du système calculée à partir de la dynamique de l'erreur désirée.

IV.2.3 Linéarisation par retour d'état des systèmes multivariables

Dans cette section, nous décrivons la théorie de linéarisation par retour d'état appliquée aux systèmes multivariables :

$$\dot{X} = f(X) + \sum_{j=1}^m g_j(X)u_j \quad (\text{IV.19})$$

où $X \in \mathfrak{R}^n$ est le vecteur d'état du système, f, g_1, g_2, \dots, g_m des champs de vecteurs définis dans \mathfrak{R}^n .

Définition IV.8. [93] Soit un système de la forme (IV.19) avec un vecteur de sorties déjà établi :

$$y = [h_1(X) \ h_2(X) \ \dots \ h_m(X)]^T$$

Ce système possède les indices caractéristiques r_1, \dots, r_m au voisinage U d'un point d'opération X^0 si :

$$\sum_{j=1}^m L_{g_j} L_f^k h_i(X) u_j = 0 \quad \text{pour} \quad 0 \leq k \leq r_i - 2 \quad \text{et} \quad 0 \leq i, j \leq m \quad \forall X \in U \quad (\text{IV.20})$$

et la matrice de dimension $m \times m$

$$\alpha(X) = \begin{bmatrix} L_{g_1} L_f^{r_1-1} h_1(X) & L_{g_2} L_f^{r_1-1} h_1(X) & \dots & L_{g_m} L_f^{r_1-1} h_1(X) \\ L_{g_1} L_f^{r_2-1} h_2(X) & L_{g_2} L_f^{r_2-1} h_2(X) & \dots & L_{g_m} L_f^{r_2-1} h_2(X) \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ L_{g_1} L_f^{r_m-1} h_m(X) & L_{g_2} L_f^{r_m-1} h_m(X) & \dots & L_{g_m} L_f^{r_m-1} h_m(X) \end{bmatrix} \quad (\text{IV.21})$$

est inversible $\forall X \in U$.

L'indice caractéristique r_i est égal au nombre de dérivations nécessaires de la sortie y_i pour retrouver au moins une entrée u_j . Ceci est traduit par :

$$\begin{aligned} \sum_{j=1}^m L_{g_j} L_f^k h_i(X) u_j = 0 \quad \text{pour} \quad 0 \leq k \leq r_i - 2 \\ \exists j \quad \text{tel que} \quad L_{g_j} L_f^{r_i-1} h_i(X) \neq 0 \end{aligned} \quad (\text{IV.22})$$

Théorème IV.9. [93] Le système de la forme (IV.19) est linéarisable au sens entrée-état si et seulement s'il existe un voisinage U de X^0 et m fonctions réelles h_1, h_2, \dots, h_m définies dans U tel que le système

$$\begin{cases} \dot{X} = f(X) + \sum_{j=1}^m g_j(X)u_j \\ y = [h_1(X) \ h_2(X) \ \dots \ h_m(X)]^T \end{cases} \quad (\text{IV.23})$$

possède un indice caractéristique $r = r_1 + r_2 + \dots + r_m = n$.

S'il est possible de trouver un vecteur de sortie tel que l'indice caractéristique du système $r = n$ donc nous pouvons définir un difféomorphisme :

$$z = \phi(X) \quad \text{avec} \quad \phi(0) = 0 \quad (\text{IV.24})$$

et un vecteur de commande de la forme :

$$u_i = \Gamma_i(X) + \sum_{j=1}^m \Lambda_{ij}(X)v_j \quad \text{avec} \quad \Gamma(0) = 0 \quad i = 1, \dots, m \quad (\text{IV.25})$$

Les nouvelles coordonnées z et les nouvelles entrées v forment un système linéaire de la forme :

$$\dot{z} = Az + Bv \quad (\text{IV.26})$$

avec

$$\begin{aligned} A_{(n \times n)} &= \text{diag}(A_1, \dots, A_m) \\ B_{(n \times m)} &= \text{diag}(B_1, \dots, B_m) \\ z_{(n \times 1)} &= \text{col}(z^1, \dots, z^m) \\ v &= [v_1, \dots, v_m]^T \end{aligned} \quad (\text{IV.27})$$

où

$$A_j = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & 1 \\ 0 & 0 & 0 & \dots & \dots & 0 \end{bmatrix}_{(k_j \times k_j)} \quad B_j = \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix}_{(k_j \times 1)}$$

k_j est l'indice caractéristique associé à la sortie y_j . Si nous choisissons $y_j = [1 \ 0 \ \dots \ 0]z^j$ avec $\dim(z^j) = k_j$ pour $1 \leq j \leq m$, les indices caractéristiques k_j deviennent égales aux indices caractéristiques r_j et l'indice caractéristique total du système devient $k = k_1 + k_2 + \dots + k_m = n$.

Lorsque le vecteur de sortie est déjà choisi, nous vérifions si l'indice caractéristique total r est égale à l'ordre du système n . Si c'est vrai le système est linéarisable au sens entrée-état. Si $\alpha(X)$ est non singulière, la loi de commande linéarisante devient :

$$u = \Gamma(X) + \Lambda(X)v \quad (\text{IV.28})$$

avec

$$\Gamma(X) = -\alpha(X)^{-1} \begin{bmatrix} L_f^{r_1} h_1(X) \\ L_f^{r_2} h_2(X) \\ \cdot \\ \cdot \\ L_f^{r_m} h_m(X) \end{bmatrix} \quad (\text{IV.29})$$

$$\Lambda(X) = \alpha(X)^{-1} \quad (\text{IV.30})$$

où $v = [v_1, \dots, v_m]^T$ est la nouvelle entrée du système.

IV.3 Linéarisation par retour d'état du robot mobile omnidirectionnel *ROMNI*

IV.3.1 Calcul de loi de commande

Afin d'écrire une loi de commande globale, considérons $X = [x_1, x_2, x_3, x_4, x_5, x_6]^T = [x, y, \vartheta, \omega_{v_1}, \omega_{v_2}, \omega_{v_3}]^T$ comme vecteur d'état du robot mobile. L'équation d'état est donnée par [59] :

$$\dot{X} = f(X) + g(X)\tau_v \quad (\text{IV.31})$$

où f et $g : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ sont des champs de vecteurs différentiables.

$$f(X) = \begin{bmatrix} r J_p(\xi_p)^{-1} \omega_v \\ -M_{v_1}(\xi_p)^{-1} C_{v_1}(\xi_p, \omega_v) \omega_v \end{bmatrix} \quad (\text{IV.32})$$

En remplaçant les expressions de $J_p(\xi_p)$, $M_v(\xi_p)$ et $C_v(\xi_p, \omega_v)$, nous trouvons :

$$f(X) = \begin{cases} f_1(X) = \frac{2}{3} r x_4 \sin x_3 + r \left(\frac{1}{\sqrt{3}} \cos x_3 - \frac{1}{3} \sin x_3 \right) x_5 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \cos x_3 - \frac{1}{3} \sin x_3 \right) x_6 \\ f_2(X) = -\frac{2}{3} r x_4 \cos x_3 + r \left(\frac{1}{\sqrt{3}} \sin x_3 + \frac{1}{3} \cos x_3 \right) x_5 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \sin x_3 + \frac{1}{3} \cos x_3 \right) x_6 \\ f_3(X) = -H r x_4 - H r x_5 - H r x_6 \\ f_4(X) = K x_4 + L x_5 + L x_6 + G(x_4 x_5 - x_4 x_6 + x_5^2 - x_6^2) \\ f_5(X) = L x_4 + K x_5 + L x_6 + G(-x_4 x_5 + x_5 x_6 - x_4^2 + x_6^2) \\ f_6(X) = L x_4 + L x_5 + K x_6 + G(x_4 x_6 - x_5 x_6 + x_4^2 - x_5^2) \end{cases} \quad (\text{IV.33})$$

$$g(X) = \begin{bmatrix} 0 \\ M_{v_1}(\xi_p)^{-1} E_v \end{bmatrix} \quad (\text{IV.34})$$

En remplaçant les expressions de $M_v(\xi_p)$ et E_v , nous trouvons :

$$g(X) = \begin{cases} g_1(X) = [0, 0, 0, E, F, F]^T \\ g_2(X) = [0, 0, 0, F, E, F]^T \\ g_3(X) = [0, 0, 0, F, F, E]^T \end{cases} \quad (\text{IV.35})$$

Tous les paramètres E, F, G, H, K et L sont des constantes.

Nous considérons le système non linéaire multivariables décrit dans l'espace d'état par les équations suivantes :

$$\begin{cases} \dot{X} = f(X) + \sum_{j=1}^3 g_j(X) \tau_{v_j} \\ y_j = h_j(X) \quad j = 1, 2, 3 \end{cases} \quad (\text{IV.36})$$

Selon [93], nous pouvons rappeler les conditions pour la résolution du problème de la linéarisation des systèmes MIMO dans l'espace d'état. Définissons les distributions :

$$\begin{aligned} G_0(X) &= \{g_1(X), g_2(X), \dots, g_m(X)\} \\ G_1(X) &= \{g_1(X), \dots, g_m(X), ad_f g_1(X), \dots, ad_f g_m(X)\} \\ &\cdot \\ &\cdot \\ &\cdot \\ G_i(X) &= \{ad_f^k g_j : 0 \leq k \leq i, 1 \leq j \leq m\} \end{aligned}$$

Proposition 1 : Le système non linéaire est linéarisable si et seulement si

- i) Pour chaque $0 \leq i \leq n - 1$, la distribution G_i a une dimension constante dans un voisinage U de X^0
- ii) La distribution G_{n-1} a une dimension n
- iii) Pour chaque $0 \leq i \leq n - 2$, la distribution G_i est involutive

Théoreme 1 : Le changement de variables nous permet la linéarisation du système

(IV.31) :

$$\left\{ \begin{array}{l} x_1 = z_1 \\ x_2 = z_3 \\ x_3 = z_5 \\ x_4 = \frac{1}{r}z_2 \sin z_5 - \frac{1}{r}z_4 \cos z_5 - \frac{1}{r}R_1z_6 \\ x_5 = \frac{1}{r} \left(\frac{\sqrt{3}}{2} \cos z_5 - \frac{1}{2} \sin z_5 \right) z_2 + \frac{1}{r} \left(\frac{\sqrt{3}}{2} \sin z_5 + \frac{1}{2} \cos z_5 \right) z_4 - \frac{1}{r}R_2z_6 \\ x_6 = \frac{1}{r} \left(-\frac{\sqrt{3}}{2} \cos z_5 - \frac{1}{2} \sin z_5 \right) z_2 + \frac{1}{r} \left(-\frac{\sqrt{3}}{2} \sin z_5 + \frac{1}{2} \cos z_5 \right) z_4 - \frac{1}{r}R_3z_6 \end{array} \right. \quad (\text{IV.37})$$

Preuve 1 : Dans le cas du robot mobile, nous devons trouver trois fonctions $h_1(X)$, $h_2(X)$, $h_3(X)$ tel que :

$$L_{g_j}L_f^k h_j(X) = 0 \quad (\text{IV.38})$$

pour tous les $0 \leq k \leq r_j - 2$, $1 \leq j \leq m$. Les degrés caractéristiques (relatifs) r_j doivent remplir la condition $r_1 + r_2 + r_3 = n$. Dans le cas du robot mobile omnidirectionnel, la distribution $G_0 = \{g_1, g_2, g_3\}$ a une dimension $m = 3$. En outre, depuis $[g_i, g_j] = 0$ pour tous les $i, j \in \{1, 2, 3\}$, nous remarquons que la distribution en question est involutive.

Nous considérons $G_1 = \{g_1, g_2, g_3, ad_f g_1, ad_f g_2, ad_f g_3\}$, la distribution est de dimension maximale $n = 6$ qui est la dimension du vecteur d'état. Nous remarquons que $\dim(G_i) = 6$ pour $i \in [1, 5]$ et G_j pour $j \in [1, 4]$ sont trivialement involutifs. Le système satisfait les hypothèses de la proposition 1. Afin de résoudre le problème de la linéarisation, nous devons trouver des fonctions $h_j(x)$ qui vérifient la condition (IV.38). Il est facile de conclure que nous devons avoir :

$$\frac{\partial h_j}{\partial x_4} = \frac{\partial h_j}{\partial x_5} = \frac{\partial h_j}{\partial x_6} = 0 \quad \text{pour } j = 1, 2, 3$$

Nous pouvons choisir $h_1(X) = x_1$, $h_2(X) = x_2$ et $h_3(X) = x_3$. Le système a un vecteur des degrés relatifs $\{r_1, r_2, r_3\}$. Nous pouvons vérifier que $L_{g_1}h_1(X) = L_{g_2}h_2(X) = L_{g_3}h_3(X) = 0$.

Les degrés relatifs sont $r_1 = r_2 = r_3 = 2$. Ensuite, nous pouvons vérifier que la matrice $\alpha(X)$ n'est pas singulière.

$$\alpha(X) = \begin{bmatrix} L_{g_1}L_f h_1(X) & L_{g_2}L_f h_1(X) & L_{g_3}L_f h_1(X) \\ L_{g_1}L_f h_2(X) & L_{g_2}L_f h_2(X) & L_{g_3}L_f h_2(X) \\ L_{g_1}L_f h_3(X) & L_{g_2}L_f h_3(X) & L_{g_3}L_f h_3(X) \end{bmatrix} \quad (\text{IV.39})$$

Les fonctions lisses qui définissent le changement de variables qui linéarise le système sont :

$$\begin{cases} z_1 = h_1(X) = x_1 \\ z_2 = L_f h_1(X) \\ z_3 = h_2(X) = x_2 \\ z_4 = L_f h_2(X) \\ z_5 = h_3(X) = x_3 \\ z_6 = L_f h_3(X) \end{cases} \quad (\text{IV.40})$$

où

$$\begin{cases} L_f h_1(X) = \frac{2}{3} r x_4 \sin x_3 + r \left(\frac{1}{\sqrt{3}} \cos x_3 - \frac{1}{3} \sin x_3 \right) x_5 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \cos x_3 - \frac{1}{3} \sin x_3 \right) x_6 \\ L_f h_2(X) = -\frac{2}{3} r x_4 \cos x_3 + r \left(\frac{1}{\sqrt{3}} \sin x_3 + \frac{1}{3} \cos x_3 \right) x_5 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \sin x_3 + \frac{1}{3} \cos x_3 \right) x_6 \\ L_f h_3(X) = -H r x_4 - H r x_5 - H r x_6 \\ \\ L_{g_1} L_f h_1(X) = +\frac{2}{3} r (E - F) \sin x_3 \\ L_{g_2} L_f h_1(X) = -\frac{1}{3} r (E - F) \sin x_3 + \frac{1}{\sqrt{3}} r (E - F) \cos x_3 \\ L_{g_3} L_f h_1(X) = -\frac{1}{3} r (E - F) \sin x_3 - \frac{1}{\sqrt{3}} r (E - F) \cos x_3 \\ L_{g_1} L_f h_2(X) = -\frac{2}{3} r (E - F) \cos x_3 \\ L_{g_2} L_f h_2(X) = +\frac{1}{3} r (E - F) \cos x_3 + \frac{1}{\sqrt{3}} r (E - F) \sin x_3 \\ L_{g_3} L_f h_2(X) = +\frac{1}{3} r (E - F) \cos x_3 - \frac{1}{\sqrt{3}} r (E - F) \sin x_3 \\ L_{g_1} L_f h_3(X) = -2 H F r - H E r \\ L_{g_2} L_f h_3(X) = -2 H F r - H E r \\ L_{g_3} L_f h_3(X) = -2 H F r - H E r \end{cases}$$

La loi de commande τ_v est comme suit :

$$\tau_v = \begin{bmatrix} \tau_{v_1} \\ \tau_{v_2} \\ \tau_{v_3} \end{bmatrix} = \alpha^{-1}(X) \left(-\beta(X) + \begin{bmatrix} v_{v_1} \\ v_{v_2} \\ v_{v_3} \end{bmatrix} \right) \quad (\text{IV.41})$$

où

$$\beta(X) = \begin{bmatrix} L_f^2 h_1(X) \\ L_f^2 h_2(X) \\ L_f^2 h_3(X) \end{bmatrix} \quad (\text{IV.42})$$

Les nouvelles coordonnées z et les nouvelles entrées v_v forment un système linéaire de la forme :

$$\dot{z} = Az + Bv_v \quad (\text{IV.43})$$

avec

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{et} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{IV.44})$$

où

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_3 \\ \dot{z}_5 \end{bmatrix} = \begin{bmatrix} z_2 \\ z_4 \\ z_6 \end{bmatrix} \quad (\text{IV.45})$$

et

$$\begin{bmatrix} \dot{z}_2 \\ \dot{z}_4 \\ \dot{z}_6 \end{bmatrix} = \begin{bmatrix} v_{v1} \\ v_{v2} \\ v_{v3} \end{bmatrix} \quad (\text{IV.46})$$

Nous obtenons une relation simple entre la sortie ξ_p et la nouvelle entrée v_v :

$$v_v = \ddot{\xi}_p = \ddot{\xi}_p^* + K_{der}\dot{\tilde{\xi}}_p + K_{pro}\tilde{\xi}_p \quad (\text{IV.47})$$

$\tilde{\xi}_p = \xi_p^* - \xi_p$, K_{der} and K_{pro} sont des matrices positives. Le schéma du contrôleur proposé est montré par la figure (IV.1).

IV.3.2 Implémentation de la commande en impédance

La loi de commande dynamique donnée par l'équation (IV.47) ne prend pas en compte la présence des obstacles. La relation d'impédance est de la forme :

$$M_d(\ddot{\xi}_p^* - \ddot{\xi}_p) + B_d(\dot{\xi}_p^* - \dot{\xi}_p) + K_d(\xi_p^* - \xi_p) = -F_{ext} \quad (\text{IV.48})$$

avec $F_{ext} = \sum_i F_{obs_i}$ représente toutes les forces répulsives exercées sur le robot. F_{obs_i} est la force répulsive due à la présence de l'obstacle i . M_d , B_d , K_d les matrices d'inertie, de frottement et de raideur de l'impédance.

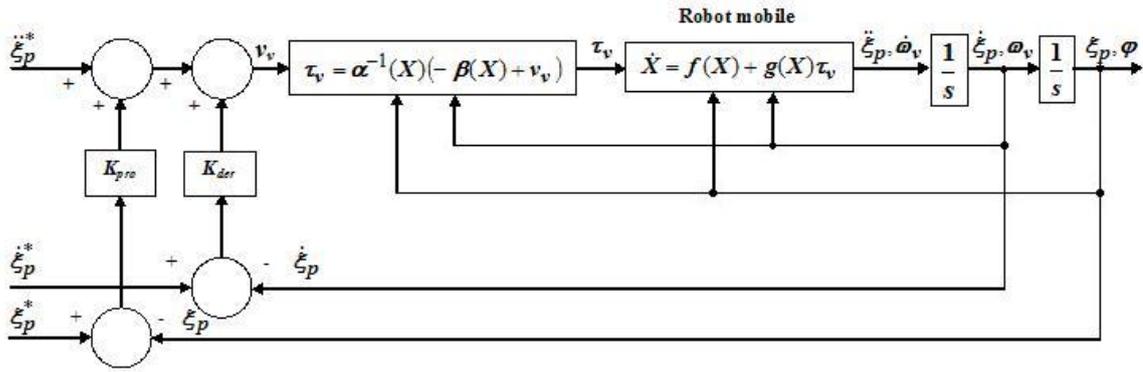


FIGURE IV.1 : Schéma du contrôleur proposé pour le robot mobile

Selon la relation (IV.48), nous pouvons constater qu'en l'absence des obstacles $F_{ext} = 0$, ce qui implique que $\xi_p^* = \xi_p$. En revanche, en présence des obstacles, la trajectoire désirée ξ_p^* doit être régulée pour vérifier la même relation (IV.48). La relation (IV.48) peut être réécrite en regroupant les termes de l'impédance désirée Z_d sous la forme :

$$(\xi_p^* - \xi_p) + \frac{F_{ext}}{Z_d} = e_v + e_1 = 0 \quad (IV.49)$$

avec $Z_d = M_d p^2 + B_d p + K_d$ est l'impédance désirée, $p \equiv \frac{d}{dt}$ la variable de Laplace, $e_v = (\xi_p^* - \xi_p)$, $e_1 = \frac{F_{ext}}{Z_d}$. Si nous remplaçons la relation (IV.49) par le nouveau vecteur d'erreur e_d , nous obtenons :

$$e_d = (\xi_p^* - \xi_p) + \frac{F_{ext}}{Z_d} \quad (IV.50)$$

Il suffit alors, pour vérifier la relation d'impédance, d'imposer à e_d de tendre vers zéro. Ainsi, nous pouvons utiliser le signal d'erreur e_d dans le cas libre ($F_{ext} = 0$) ou le cas contraint (présence des obstacles) ($F_{ext} \neq 0$). La nouvelle trajectoire désirée ξ_d^* peut être écrite comme la somme de la trajectoire désirée ξ_p^* et le terme $\frac{F_{ext}}{Z_d}$, $\xi_d^*(t) = \xi_p^*(t) + \frac{F_{ext}}{Z_d}$.

La nouvelle loi de commande v_v est donnée par :

$$v_v = \ddot{\xi}_d^* + K_{der}(\dot{\xi}_d^* - \dot{\xi}_p) + K_{pro}(\xi_d^* - \xi_p) \quad (IV.51)$$

Nous voyons ici que nous pouvons vérifier le contrôle du robot mobile libre et avec obstacles.

IV.3.3 Résultats de simulation

Toutes les simulations ont été réalisées en considérant un ensemble de paramètres physiques pour le modèle dynamique du robot mobile omnidirectionnel donnés par : la

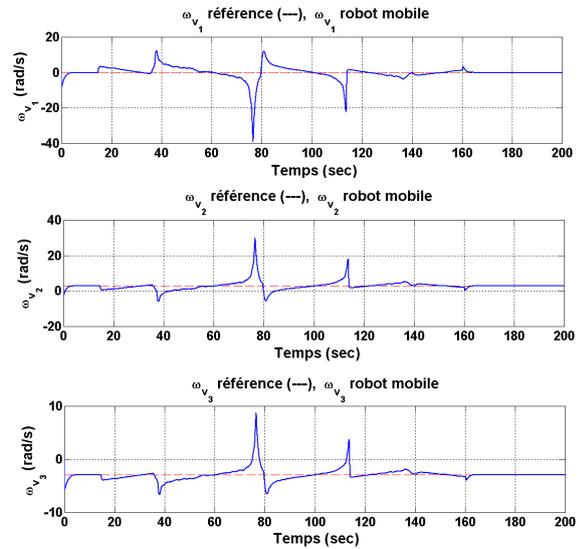
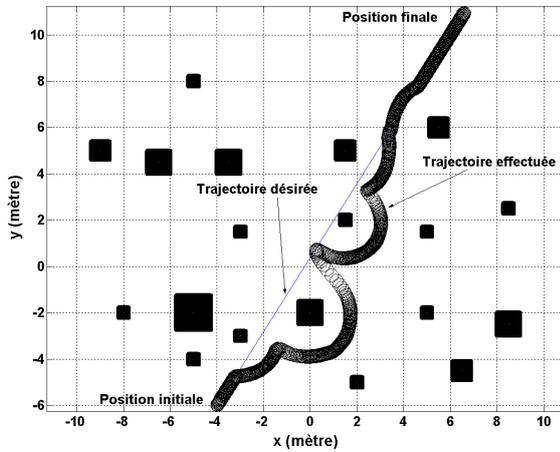


FIGURE IV.2 : Exemple 1 de navigation du robot mobile dans un milieu encombré et d'obstacles **FIGURE IV.3** : Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}

masse de la plate-forme $m_R = 20 \text{ kg}$, le rayon de la plate-forme $R = 0.3 \text{ m}$, le moment d'inertie de la plate-forme $I_R = 0.9 \text{ kg.m}^2$, le moment d'inertie du moteur $I_m = 1.380e - 5 \text{ kg.m}^2$, la résistance interne du moteur $R_a = 0.317 \text{ } \Omega$, le coefficient du frottement visqueux du moteur $b_m = 0.004 \text{ N.m}$, la constante de force électromotrice (EMF) $C_e = 3.02e - 2 \text{ V.s/rad}$, la constante de couple du moteur $C_m = 3.02e - 2 \text{ N.m/A}$, le rayon d'une sphère $r = 0.03 \text{ m}$, $R_{i_{max}} = 0.2 \text{ m}$ et $R_{i_{min}} = 0.13 \text{ m}$. Pour montrer la validité de la loi de commande proposée, nous avons effectué des tests de simulation sur le robot mobile omnidirectionnel. Différentes situations ont été examinées. Les trajectoires imposées pour le robot mobile à partir de $t_0 = 0 : [x^*(t), y^*(t), \vartheta^*(t)]^T = [0.1t, 0.1t, \pi/4]^T$. La figure (IV.2) illustre clairement la réussite du robot mobile dans sa navigation vers son but final dans un environnement assez encombré d'obstacles. Les vitesses angulaires w_{v_1} , w_{v_2} , w_{v_3} et la position de la plate-forme (x, y) sont respectivement reportées sur les figures (IV.3), (IV.4) et (IV.5). Les figures (IV.6), (IV.7), (IV.8) et (IV.9) illustrent un deuxième exemple de simulation [59].

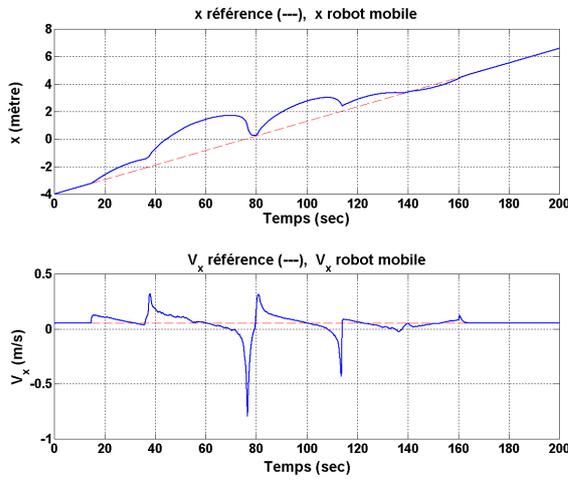


FIGURE IV.4 : Position x et vitesse \dot{x} du robot mobile

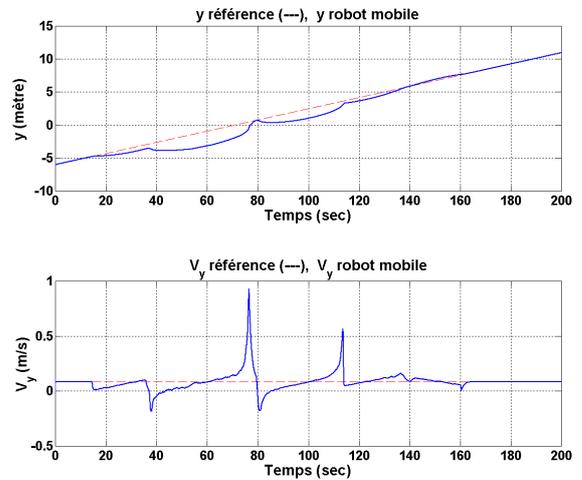


FIGURE IV.5 : Position y et vitesse \dot{y} du robot mobile

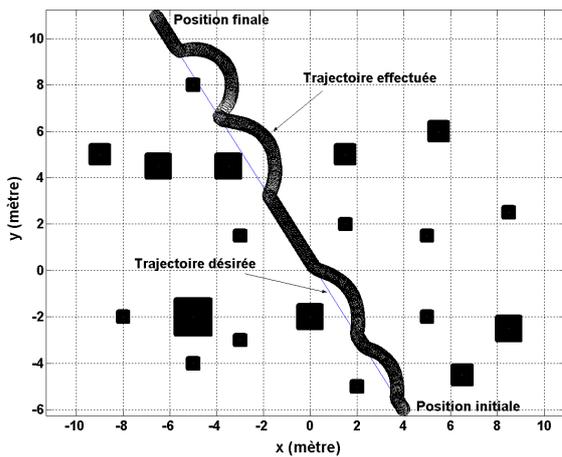


FIGURE IV.6 : Exemple 2 de navigation du robot mobile dans un milieu encombré et d'obstacles

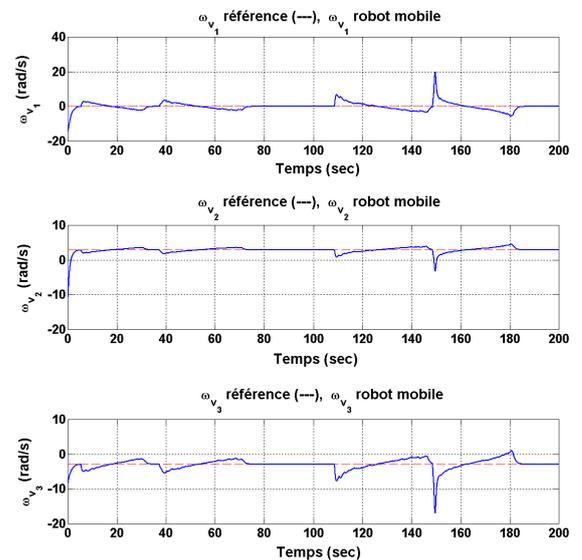


FIGURE IV.7 : Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}

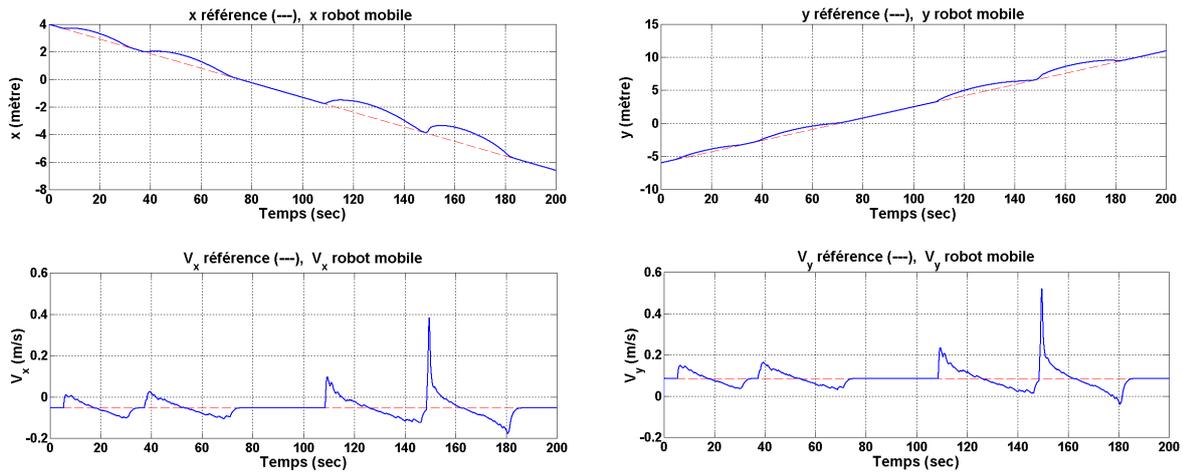


FIGURE IV.8 : Position x et vitesse \dot{x} du robot mobile

FIGURE IV.9 : Position y et vitesse \dot{y} du robot mobile

IV.4 Linéarisation par retour d'état du manipulateur mobile omnidirectionnel

IV.4.1 Calcul de loi de commande

Afin d'écrire une loi de commande globale, considérons $X = [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}]^T = [q_{r_1}, q_{r_2}, x, y, \vartheta, \omega_{r_1}, \omega_{r_2}, \omega_{v_1}, \omega_{v_2}, \omega_{v_3}]^T$ comme vecteur d'état du manipulateur mobile. L'équation d'état est donnée par [45] :

$$\dot{X} = f(X) + g(X)\tau \quad (\text{IV.52})$$

où f et $g : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ sont des champs de vecteurs différentiables.

$$f(X) = \begin{bmatrix} \omega_r \\ rJ_p(\xi_p)^{-1}\omega_v \\ -M^{-1}C \end{bmatrix} \quad (\text{IV.53})$$

En remplaçant les expressions de $J_p(\xi_p)$, M et C , nous trouvons :

$$f(X) = \begin{cases} f_1(X) = x_6 \\ f_2(X) = x_7 \\ f_3(X) = \frac{2}{3}rx_8 \sin x_5 + r \left(\frac{1}{\sqrt{3}} \cos x_5 - \frac{1}{3} \sin x_5 \right) x_9 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \cos x_5 - \frac{1}{3} \sin x_5 \right) x_{10} \\ f_4(X) = -\frac{2}{3}rx_8 \cos x_5 + r \left(\frac{1}{\sqrt{3}} \sin x_5 + \frac{1}{3} \cos x_5 \right) x_9 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \sin x_5 + \frac{1}{3} \cos x_5 \right) x_{10} \\ f_5(X) = -Hrx_8 - Hrx_9 - Hrx_{10} \\ [f_6(X), f_7(X), f_8(X), f_9(X), f_{10}(X)]^T = -M^{-1}C \end{cases} \quad (\text{IV.54})$$

$$g(X) = \begin{bmatrix} 0_{5 \times 5} \\ M^{-1} \end{bmatrix} \quad (\text{IV.55})$$

Le paramètre H est une constante.

Nous considérons le système non linéaire multivariables décrit dans l'espace d'état par les équations suivantes :

$$\begin{cases} \dot{X} = f(X) + \sum_{j=1}^5 g_j(X)\tau_j \\ y_j = h_j(X) \quad j = 1, 2, 3, 4, 5 \end{cases} \quad (\text{IV.56})$$

Théorème 2 : Le changement de variables nous permet la linéarisation du système (IV.52) :

$$\begin{cases} x_1 = z_1 \\ x_2 = z_3 \\ x_3 = z_5 \\ x_4 = z_7 \\ x_5 = z_9 \\ x_6 = z_2 \\ x_7 = z_4 \\ x_8 = \frac{1}{r}z_6 \sin z_9 - \frac{1}{r}z_8 \cos z_9 - \frac{1}{r}R_1z_{10} \\ x_9 = \frac{1}{r} \left(\frac{\sqrt{3}}{2} \cos z_9 - \frac{1}{2} \sin z_9 \right) z_6 + \frac{1}{r} \left(\frac{\sqrt{3}}{2} \sin z_9 + \frac{1}{2} \cos z_9 \right) z_8 - \frac{1}{r}R_2z_{10} \\ x_{10} = \frac{1}{r} \left(-\frac{\sqrt{3}}{2} \cos z_9 - \frac{1}{2} \sin z_9 \right) z_6 + \frac{1}{r} \left(-\frac{\sqrt{3}}{2} \sin z_9 + \frac{1}{2} \cos z_9 \right) z_8 - \frac{1}{r}R_3z_{10} \end{cases} \quad (\text{IV.57})$$

Preuve 2 : Dans le cas du manipulateur mobile, nous devons trouver cinq fonctions $h_1(X), h_2(X), h_3(X), h_4(X), h_5(X)$ tel que :

$$L_{g_j} L_f^k h_j(X) = 0 \quad (\text{IV.58})$$

pour tous les $0 \leq k \leq r_j - 2, 1 \leq j \leq m$. Les degrés relatifs r_j doivent remplir la condition $r_1 + r_2 + r_3 + r_4 + r_5 = n$. Dans le cas du manipulateur mobile omnidirectionnel, la distribution $G_0 = \{g_1, g_2, g_3, g_4, g_5\}$ a une dimension $m = 5$. En outre, depuis $[g_i, g_j] = 0$ pour tous les $i, j \in \{1, 2, 3, 4, 5\}$, nous remarquons que la distribution en question est involutive.

Nous considérons $G_1 = \{g_1, g_2, g_3, g_4, g_5, ad_f g_1, ad_f g_2, ad_f g_3, ad_f g_4, ad_f g_5\}$, la distribution est de dimension maximale $n = 10$ qui est la dimension du vecteur d'état. Nous remarquons que $dim(G_i) = 10$ pour $i \in [1, 9]$ et G_j pour $j \in [1, 8]$ sont trivialement involutifs. Le système satisfait les hypothèses de la proposition 1. Afin de résoudre le problème de la linéarisation, nous devons trouver des fonctions $h_j(x)$ qui vérifient la condition (IV.58). Il est facile de conclure que nous devons avoir :

$$\frac{\partial h_j}{\partial x_6} = \frac{\partial h_j}{\partial x_7} = \frac{\partial h_j}{\partial x_8} = \frac{\partial h_j}{\partial x_9} = \frac{\partial h_j}{\partial x_{10}} = 0 \quad \text{pour } j = 1, 2, 3, 4, 5$$

Nous pouvons choisir $h_1(X) = x_1, h_2(X) = x_2, h_3(X) = x_3, h_4(X) = x_4, h_5(X) = x_5$. Le système a un vecteur des degrés relatifs $\{r_1, r_2, r_3, r_4, r_5\}$. Nous pouvons vérifier que $L_{g_1} h_1(X) = L_{g_2} h_2(X) = L_{g_3} h_3(X) = L_{g_4} h_4(X) = L_{g_5} h_5(X) = 0$.

Les degrés relatifs sont $r_1 = r_2 = r_3 = r_4 = r_5 = 2$. Ensuite, nous pouvons vérifier que la matrice $\alpha(X)$ n'est pas singulière.

$$\alpha(X) = \begin{bmatrix} L_{g_1} L_f h_1(X) & \cdot & \cdot & \cdot & L_{g_5} L_f h_1(X) \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ L_{g_1} L_f h_5(X) & \cdot & \cdot & \cdot & L_{g_5} L_f h_5(X) \end{bmatrix} \quad (\text{IV.59})$$

Les fonctions lisses qui définissent le changement de variables qui linéarise le système

sont :

$$\left\{ \begin{array}{l} z_1 = h_1(X) = x_1 \\ z_2 = L_f h_1(X) \\ z_3 = h_2(X) = x_2 \\ z_4 = L_f h_2(X) \\ z_5 = h_3(X) = x_3 \\ z_6 = L_f h_3(X) \\ z_7 = h_4(X) = x_4 \\ z_8 = L_f h_4(X) \\ z_9 = h_5(X) = x_5 \\ z_{10} = L_f h_5(X) \end{array} \right. \quad (\text{IV.60})$$

où

$$\left\{ \begin{array}{l} L_f h_1(X) = x_6 \\ L_f h_2(X) = x_7 \\ L_f h_3(X) = \frac{2}{3} r x_8 \sin x_5 + r \left(\frac{1}{\sqrt{3}} \cos x_5 - \frac{1}{3} \sin x_5 \right) x_9 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \cos x_5 - \frac{1}{3} \sin x_5 \right) x_{10} \\ L_f h_4(X) = -\frac{2}{3} r x_8 \cos x_5 + r \left(\frac{1}{\sqrt{3}} \sin x_5 + \frac{1}{3} \cos x_5 \right) x_9 + \\ \quad r \left(-\frac{1}{\sqrt{3}} \sin x_5 + \frac{1}{3} \cos x_5 \right) x_{10} \\ L_f h_5(X) = -H r x_8 - H r x_9 - H r x_{10} \end{array} \right.$$

La loi de commande τ est comme suit :

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \\ \tau_5 \end{bmatrix} = \alpha^{-1}(X) \left(-\beta(X) + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} \right) \quad (\text{IV.61})$$

où

$$\beta(X) = L_f^2 h_j(X) \quad j = 1, 2, 3, 4, 5 \quad (\text{IV.62})$$

Les nouvelles coordonnées z et les nouvelles entrées v forment un système linéaire de la forme :

$$\dot{z} = Az + Bv \quad (\text{IV.63})$$

avec

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{et} \quad B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{IV.64})$$

où

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_3 \\ \dot{z}_5 \\ \dot{z}_7 \\ \dot{z}_9 \end{bmatrix} = \begin{bmatrix} z_2 \\ z_4 \\ z_6 \\ z_8 \\ z_{10} \end{bmatrix} \quad (\text{IV.65})$$

et

$$\begin{bmatrix} \dot{z}_2 \\ \dot{z}_4 \\ \dot{z}_6 \\ \dot{z}_8 \\ \dot{z}_{10} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} \quad (\text{IV.66})$$

Nous obtenons une relation simple entre la sortie q et la nouvelle entrée v :

$$v = \ddot{q} = \ddot{q}^* + K_{der}\dot{\tilde{q}} + K_{pro}\tilde{q} \quad (\text{IV.67})$$

$\tilde{q} = q^* - q$, K_{der} and K_{pro} sont des matrices positives. Le schéma du contrôleur proposé est montré par la figure (IV.10).

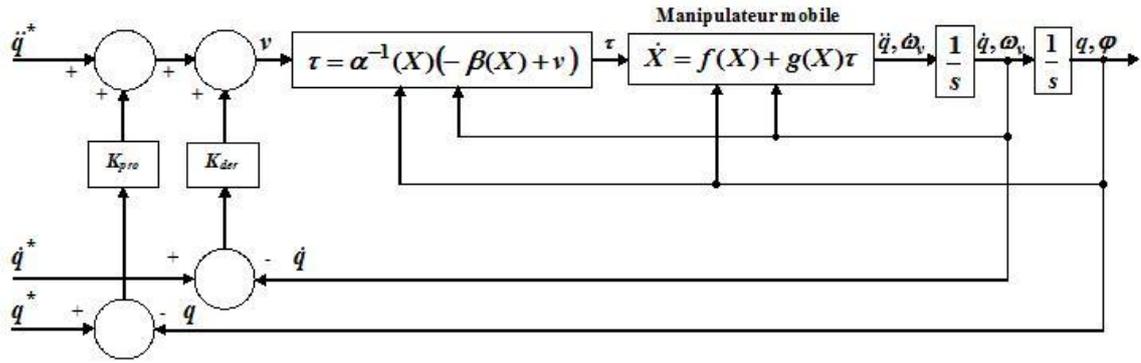


FIGURE IV.10 : Schéma du contrôleur proposé pour le manipulateur mobile

IV.4.2 Implémentation de la commande en impédance

La loi de commande dynamique donnée par l'équation (IV.67) ne prend pas en compte la présence des obstacles. La relation d'impédance est de la forme :

$$M_d(\ddot{q}^* - \ddot{q}) + B_d(\dot{q}^* - \dot{q}) + K_d(q^* - q) = -F_{ext} \quad (\text{IV.68})$$

Selon la relation (IV.68), nous pouvons constater qu'en l'absence des obstacles $F_{ext} = 0$, ce qui implique que $q^* = q$. En revanche, en présence des obstacles, la trajectoire désirée q^* doit être régulée pour vérifier la même relation (IV.68). La relation (IV.68) peut être réécrite en regroupant les termes de l'impédance désirée Z_d sous la forme :

$$(q^* - q) + \frac{F_{ext}}{Z_d} = e + e_1 = 0 \quad (\text{IV.69})$$

avec $e = (q^* - q)$, $e_1 = \frac{F_{ext}}{Z_d}$. Si nous remplaçons la relation (IV.69) par le nouveau vecteur d'erreur e_d , nous obtenons :

$$e_d = (q^* - q) + \frac{F_{ext}}{Z_d} \quad (\text{IV.70})$$

Il suffit alors, pour vérifier la relation d'impédance, d'imposer à e_d de tendre vers zéro. Ainsi, nous pouvons utiliser le signal d'erreur e_d dans le cas libre ($F_{ext} = 0$) ou le cas contraint (présence des obstacles) ($F_{ext} \neq 0$). La nouvelle trajectoire désirée q_d^* peut être écrite comme la somme de la trajectoire désirée q^* et le terme $\frac{F_{ext}}{Z_d}$, $q_d^*(t) = q^*(t) + \frac{F_{ext}}{Z_d}$.

La nouvelle loi de commande v est donnée par :

$$v = \ddot{q}_d^* + K_{der}(\dot{q}_d^* - \dot{q}) + K_{pro}(q_d^* - q) \quad (\text{IV.71})$$

IV.4.3 Résultats de simulation

Toutes les simulations ont été réalisées en considérant un ensemble de paramètres physiques pour le modèle dynamique du manipulateur mobile omnidirectionnel donnés par : la masse du bras1 $m_1 = 10 \text{ kg}$, la masse du bras2 $m_2 = 5 \text{ kg}$, le moment d'inertie du bras1 $I_1 = 0.05 \text{ kg.m}^2$, le moment d'inertie du bras2 $I_2 = 0.025 \text{ kg.m}^2$, la longueur d'un bras $l_1 = l_2 = 0.5 \text{ m}$, la masse de la plate-forme $m_R = 20 \text{ kg}$, le rayon de la plate-forme $R = 0.3 \text{ m}$, le moment d'inertie de la plate-forme $I_R = 0.9 \text{ kg.m}^2$, le moment d'inertie du moteur $I_m = 1.380e - 5 \text{ kg.m}^2$, la résistance interne du moteur $R_a = 0.317 \Omega$, le coefficient du frottement visqueux du moteur $b_m = 0.004 \text{ N.m}$, la constante de force électromotrice (EMF) $C_e = 3.02e - 2 \text{ V.s/rad}$, la constante de couple du moteur $C_m = 3.02e - 2 \text{ N.m/A}$, le rayon d'une sphère $r = 0.03 \text{ m}$, $R_{i_{max}} = 0.2 \text{ m}$ et $R_{i_{min}} = 0.13 \text{ m}$, $a_1 = a_2 = 0$. Pour montrer la validité de la loi de commande proposée, nous avons effectué des tests de simulation sur le manipulateur mobile omnidirectionnel. Les résultats de simulation sont présentés par les figures (IV.11) à (IV.14), et indiquent l'efficacité de la loi de contrôle pour les trajectoires imposées pour la plate-forme à partir de $t_0 = 0$: $x^*(t) = 0.05 \sin(1.6t + 0.2) + 0.09 \sin(2t + 0.15)$ et $y^*(t) = 0.08 \sin(2.6t + 0.02) + 0.02 \sin(1.2t + 0.35)$. Le mouvement articulaire imposé pour le bras manipulateur est $q_r^* = [q_{r_1}^*, q_{r_2}^*]^T = [0.1 \sin(2t + 0.1), 0.2 \cos(t + 0.1)]^T$. La position de la plate-forme (x, y) ainsi que les variations des différents angles d'articulation q_{r_1} et q_{r_2} du bras manipulateur sont respectivement reportées sur les figures (IV.11), (IV.12), (IV.13) et (IV.14). Les vitesses angulaires w_{v_1} , w_{v_2} , w_{v_3} sont reportées sur la figure (IV.15). Dans une deuxième simulation, la figure (IV.16) illustre clairement la réussite du manipulateur mobile dans sa navigation vers son but final dans un environnement assez encombré d'obstacles. Les trajectoires imposées pour la plate-forme à partir de $t_0 = 0$: $[x^*(t), y^*(t), \vartheta^*(t)]^T = [0.1t, 0.1t, \pi/4]^T$. Le mouvement articulaire imposé pour le bras manipulateur est $q_r^* = [q_{r_1}^*, q_{r_2}^*]^T = [\pi/2, -\pi/2]^T$. La figure (IV.16) montre la navigation du manipulateur mobile dans un environnement assez encombré d'obstacles. Les vitesses angulaires w_{v_1} , w_{v_2} , w_{v_3} et la position de la plate-forme (x, y) sont respectivement reportées sur les figures (IV.17), (IV.18) et (IV.19). Les variations des différents angles d'articulation q_{r_1} et q_{r_2} du bras manipulateur sont respectivement reportées sur les figures (IV.20) et (IV.21) [45].

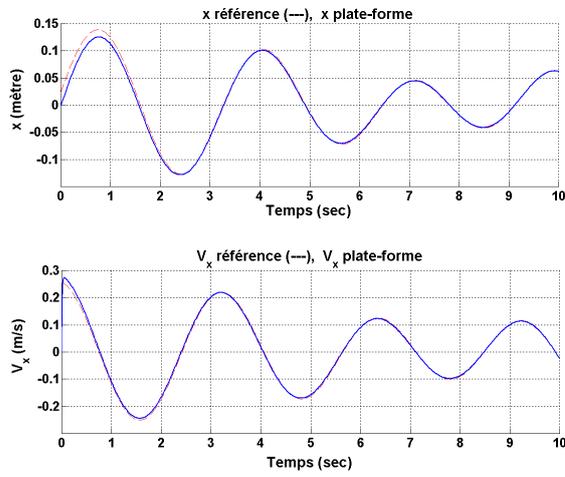


FIGURE IV.11 : Position x et vitesse \dot{x} de la plate-forme

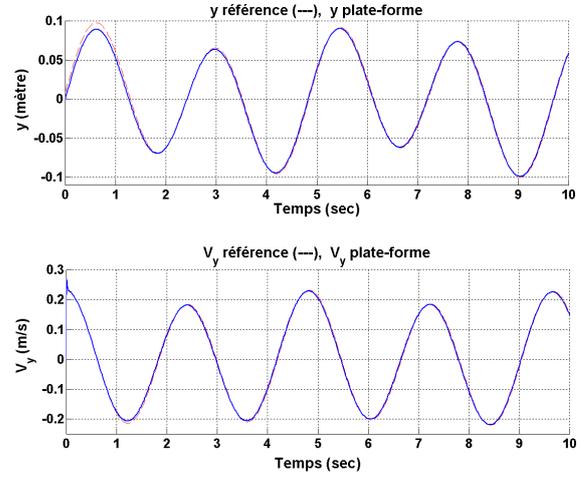


FIGURE IV.12 : Position y et vitesse \dot{y} de la plate-forme

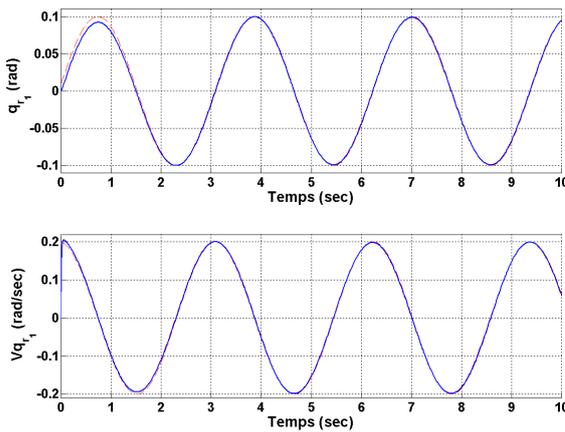


FIGURE IV.13 : Orientation q_{r_1} et vitesse angulaire \dot{q}_{r_1} du bras1

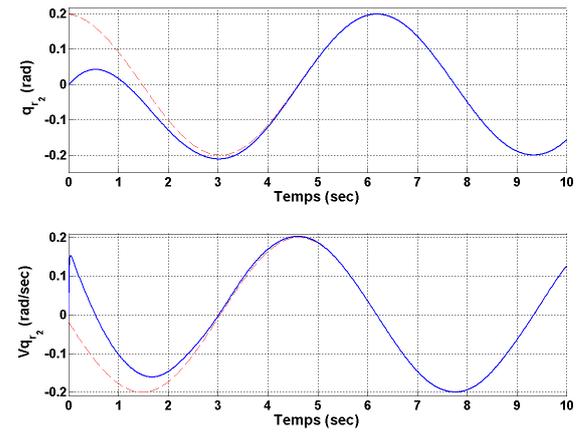


FIGURE IV.14 : Orientation q_{r_2} et vitesse angulaire \dot{q}_{r_2} du bras2

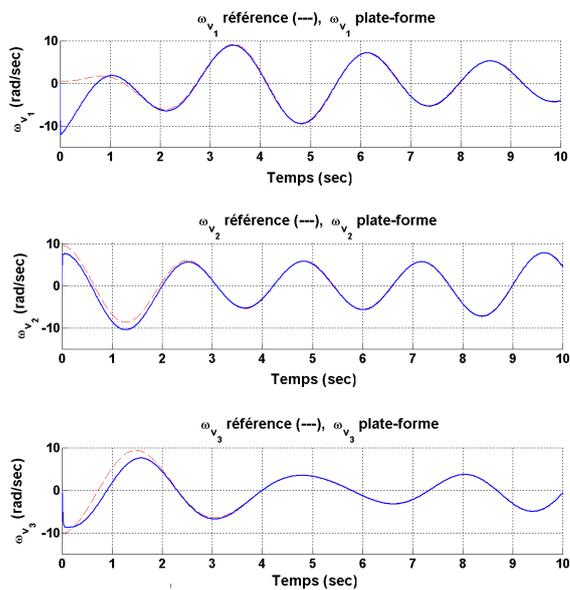


FIGURE IV.15 : Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}

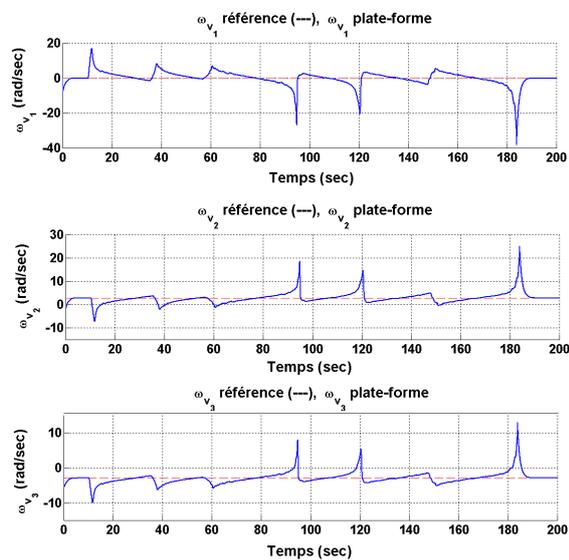
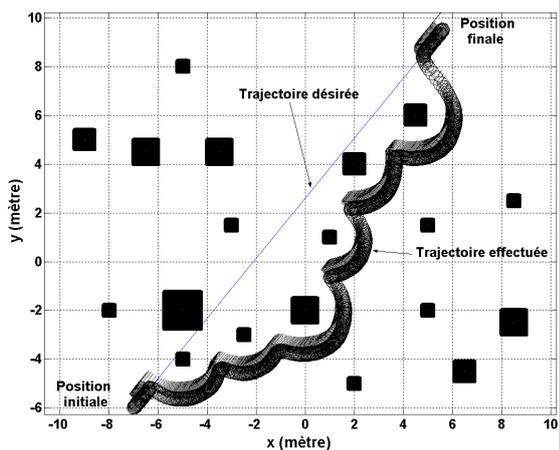


FIGURE IV.16 : Exemple de navigation du manipulateur mobile dans un milieu encombré d'obstacles

FIGURE IV.17 : Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}

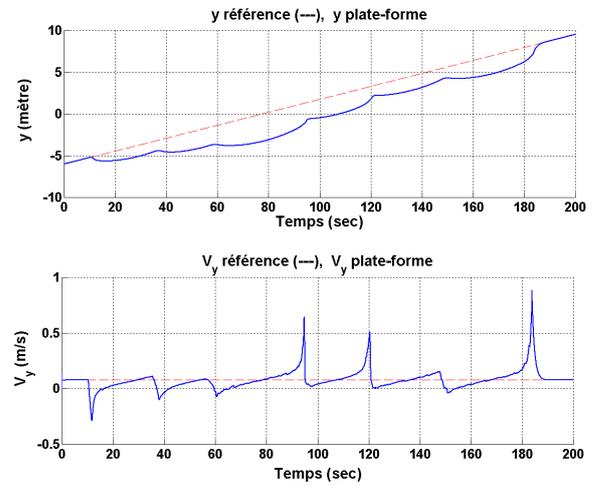
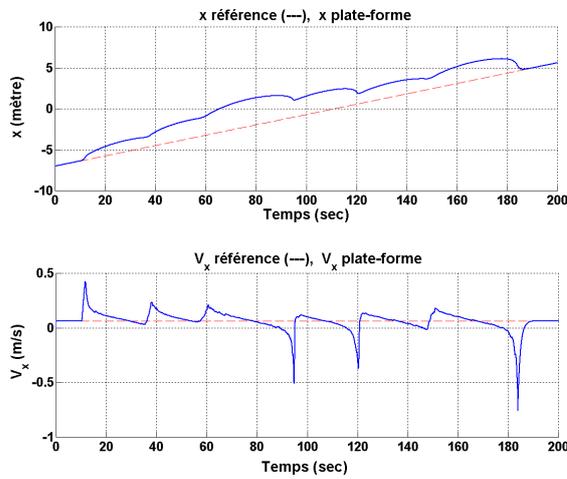


FIGURE IV.18 : Position x et vitesse \dot{x} de la plate-forme

FIGURE IV.19 : Position y et vitesse \dot{y} de la plate-forme

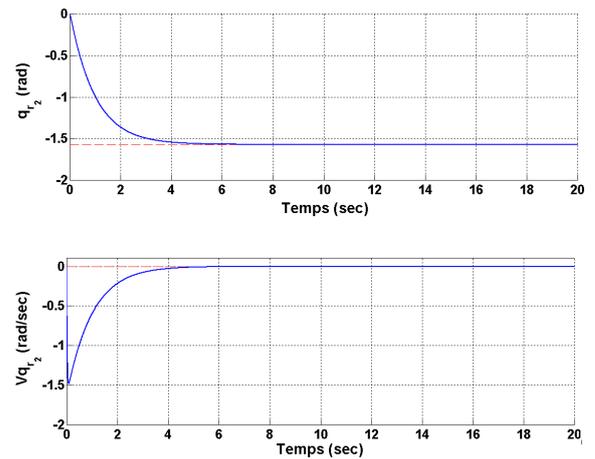
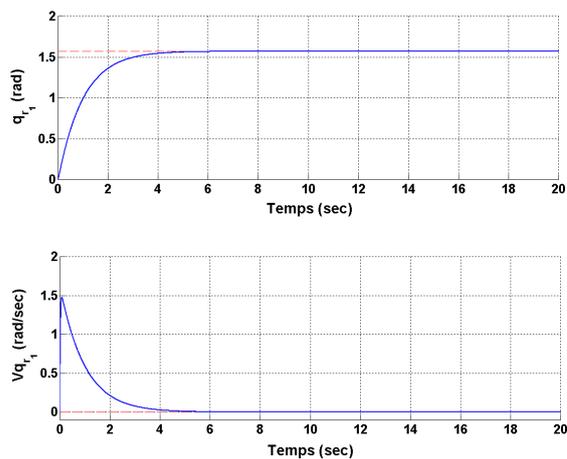


FIGURE IV.20 : Orientation q_{r1} et vitesse angulaire \dot{q}_{r1} du bras1

FIGURE IV.21 : Orientation q_{r2} et vitesse angulaire \dot{q}_{r2} du bras2

IV.5 Conclusion

Dans ce chapitre, nous avons détaillé les différents aspects théoriques liés à la commande linéarisante. Nous avons présenté l'application de cette technique de commande non linéaire sur le robot mobile omnidirectionnel et sur le manipulateur mobile omnidirectionnel. Nous avons ajouté la stratégie du contrôle dynamique augmenté d'une boucle externe de contrôle en impédance pour l'évitement d'obstacles. Après plusieurs tests de simulation, le problème rencontré est le choix des paramètres de l'impédance Z_d . Il s'agit de M_d , B_d , K_d , les matrices d'inertie, de frottement et de raideur de l'impédance.

Chapitre V

Adaptation des paramètres de l'impédance & identification des paramètres dynamiques

V.1 Introduction

A partir de concepts mathématiques relativement simples, la logique floue permet de résoudre des problèmes complexes. Tous les problèmes de contrôle, de diagnostic ou d'aide à la décision sont des clients potentiels de la logique floue. Ainsi, la logique floue s'est imposée dans des domaines aussi variés, l'industrie, l'automatisme, la robotique, la gestion de la circulation routière, le contrôle aérien, l'environnement, la médecine, afin de résoudre les problèmes d'identification, de régulation de processus, d'optimisation, de classification, de détection de défauts ou de prise de décision.

La commande floue est le domaine d'application de la logique floue. Cette approche de commande floue a pour objectif le traitement des problèmes de commande de processus qui sont, soit difficilement automatisables par une approche classique, soit les sources d'information sont jugées non précises ou incertaines, et cela à partir des connaissances des experts ou d'opérateurs qualifiés travaillant sur le processus. La particularité de cette commande est de reproduire le comportement humain plutôt que de réaliser un modèle mathématique du système et le contrôleur flou peut ainsi être vu comme un algorithme qui peut convertir une stratégie formelle de commande basée sur les connaissances d'un

expert en une stratégie automatique de commande. Cet algorithme de commande se base sur une collection de règles floues appelée base de règles. L'ensemble des règles de la commande sont reliées par les concepts d'implication, de composition floue et des règles d'inférence floue.

Dans le chapitre 4, nous avons ajouté la stratégie du contrôle dynamique augmenté d'une boucle externe de contrôle en impédance pour l'évitement d'obstacles. Le problème rencontré est le choix des paramètres de l'impédance Z_d . Il s'agit de M_d , B_d , K_d , les matrices d'inertie, de frottement et de raideur de l'impédance. Dans ce chapitre, nous allons introduire un algorithme adaptatif flou comme une solution de contrôle intelligent pour choisir le comportement désiré Z_d . Dans ce chapitre aussi, nous traitons de l'identification des paramètres dynamiques du manipulateur mobile omnidirectionnel. La connaissance précise du modèle dynamique du robot est primordiale. Après avoir introduit les paramètres dynamiques à identifier, nous présentons la méthode d'identification par moindres carrés récursifs.

V.2 Les concepts de la logique floue

V.2.1 Histoire de la logique floue

La théorie de la logique floue a été développée dans les années soixante à l'université de Californie à Berkeley par le professeur Lotfi Zadeh. Le texte "*Fuzzy Sets*" est apparu la première fois en 1965 dans la revue "*Information and Control*" [96]. Depuis 1965, le professeur Zadeh n'a pas arrêté d'enrichir la théorie dans ce domaine avec ses innombrables publications dans différents journaux et conférences [97, 98, 99, 100]. Mais ce n'est que depuis les années quatre vingt, que les idées formulées dans les années soixante ont été acceptées. En effet, en 1974, Mamdani et ses étudiants motivés par les articles de Zadeh, ont réussi à établir le premier rapprochement entre la théorie du flou et le contrôle des systèmes. Il s'agissait de la régulation d'une machine à vapeur. Ce succès surprenant a motivé par la suite King et Mamdani au contrôle de la température d'un réacteur chimique [101]. Cependant, la première application industrielle était en 1980 destinée à la conduite d'un four à chaux. Une autre application pratique de cette technique a été réalisée en 1983 au Japon pour la conduite d'installation d'assainissement d'eau.

V.2.2 Commande et réglage par la logique floue

Le principe fondamental de la logique floue est de permettre à des informations ambiguës d'être classées dans des ensembles à limites non tranchants, c'est à dire que la forme des fonctions d'appartenance n'est pas rigide et peut dépendre du contexte envisagé à savoir professionnel, social ou autre. Cette construction permet une description des fonctions de contrôle avec des termes linguistiques vagues, tels que : "Large", "Petit", "Rapide", etc. Un des avantages principaux de la logique floue est de permettre l'incorporation des connaissances d'experts sur le comment commander le système sans pour autant connaître la description mathématique du problème. C'est ainsi que la logique floue se prête particulièrement bien pour des processus mal modélisés. Nous nous sommes intéressés dans cette thèse à résoudre le problème de l'adaptation des paramètres de l'impédance utilisée pour contrôler le déplacement du robot à l'aide de la logique floue. Nous examinerons dans cette partie les différentes étapes de la construction systématique d'un contrôleur flou.

V.2.2.1 Construction d'un contrôleur flou

La structure conventionnelle d'une commande floue est présentée par la figure (V.1). Elle est composée de quatre blocs distincts dont les définitions sont données ci-dessous. L'espace des entrées d'un contrôleur flou constitue par rapport à un certain univers de discours la prémisse de la règle dont la forme générique est "Si-Alors". L'espace des sorties constitue la conséquence de la règle.

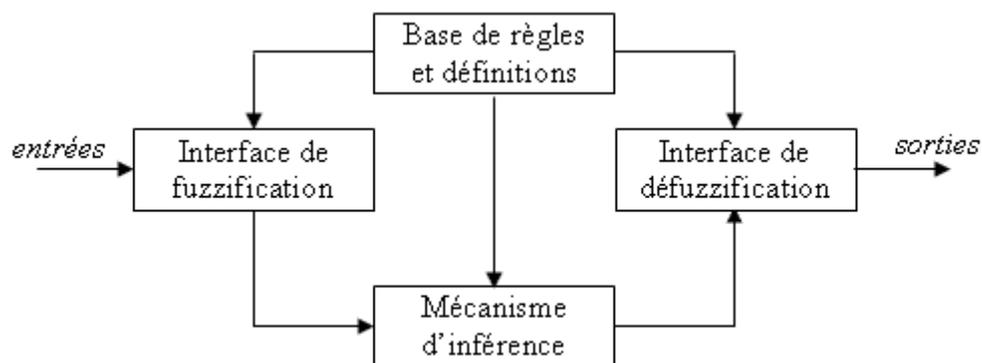


FIGURE V.1 : Structure d'une commande floue

Base de règles et définitions

Nous regroupons dans ce bloc l'ensemble des définitions utilisées dans la commande floue (univers de discours, partition floue, choix des opérateurs,...), ainsi que la base de règles, transcription sous forme de règles "Si-Alors".

- **Partition floue** : La création et l'utilisation d'une base de règles nécessitent l'existence, pour chaque univers de discours considéré, de sous-ensembles flous particuliers. La définition de ces sous-ensembles flous fait l'objet de la partition floue.
- **L'univers de discours et sa normalisation** : Dans un contrôleur flou, un univers de discours peut être continu ou discrétisé. Si l'univers de discours est discrétisé, alors nous définissons un ensemble flou par des degrés d'appartenance attribués à chaque élément quantifié. Si la sortie du contrôleur flou est également discrétisé alors la base de données contient les tables de décision définissant la stratégie numérique. Dans le cas d'un univers continu, le nombre de niveaux de quantification doit être assez grand pour fournir une approximation adéquate mais tout de même faible pour limiter les besoins en mémoire. Dans ce cas, nous imposons un choix des fonctions d'appartenance et des facteurs d'échelle.
- **Base de règles** : Une fois la partition des univers de discours réalisée, il est possible de définir la base de règles. Celle-ci caractérise les relations entre les classes d'évènements possibles en entrée et les commandes correspondantes. La base de règles constitue un ensemble d'expressions linguistiques structurées autour d'une connaissance d'expert, représentée sous forme de règles "Si-Alors", comme :

Si "état du processus" Alors "sortie du contrôleur"

La partie "Si" de cette règle est appelée la prémisse de la règle et la partie "Alors" est appelée conséquence de la règle. Dans cette partie de la règle, nous devons prendre en considération :

- Le choix des variables d'état d'entrée ainsi que les variables de sortie du contrôleur.
- Choix des termes linguistiques.
- La dérivation des règles.
- Choix du continu des règles.

Fuzzification

Les opérateurs utilisés dans la commande floue agissent sur des sous-ensembles flous. Par conséquent, il est nécessaire de transformer les variables non floues en des sous-ensembles flous. Pour ce faire, nous utilisons un opérateur dit de fuzzification. Les variables numériques réelles sont traitées par des variables linguistiques définies par des fonctions d'appartenance étalées sur un axe appelé l'univers de discours.

Mécanisme d'inférence

En utilisant la base de connaissances et la logique floue, le moteur d'inférence détermine les commandes floues à appliquer au processus. La fonction principale d'un moteur d'inférence est de déterminer la valeur globale de la variable de sortie de contrôle calculée à partir de la contribution de chaque règle. Les deux fonctions principales qui constituent un moteur d'inférences sont l'inférence floue et l'agrégateur de règles.

- **Inférence floue** : La stratégie de réglage dépend essentiellement des inférences adoptées. Elles lient les grandeurs mesurées, qui sont les variables d'entrée x (transformées en variables linguistiques à l'aide de la fuzzification), à la variable de sortie x_R . Cette dernière est également exprimée comme variable linguistique. Dans le cas du réglage par logique floue, il s'agit d'inférences avec plusieurs règles. Dans les inférences du régulateur par logique floue interviennent les opérateurs ET et OU. L'opérateur ET s'applique aux variables à l'intérieur d'une règle, tandis que l'opérateur OU lie les différentes règles.

l^{eme} règles : $R^l : Si x_1 est F_1^l ET \dots ET x_k est F_k^l Alors x_R est G^l$

Dans cette écriture, x_j , $j = 1, \dots, k$ les variables de la prémisse, F_j^l , $l = 1, \dots, m$ et $j = 1, \dots, k$ est le sous-ensemble flou de la règle R^l sur l'univers de la j^{eme} composante du vecteur de mesure x et G^l est le sous-ensemble flou relatif à la commande à appliquer pour la règle R^l .

- **Agrégateur de règles** : Il existe plusieurs possibilités pour réaliser les opérateurs qui s'appliquent aux fonctions d'appartenance. C'est la notion de méthode d'inférence. Elle détermine la réalisation des différents opérateurs dans une inférence, permettant ainsi un traitement numérique de cette dernière. Pour le réglage par logique floue, nous utilisons en général une des méthodes suivantes [102, 103, 104] :
 - méthode d'inférence max-min,

- méthode d'inférence max-prod,
- méthode d'inférence somme-prod.

Défuzzification

Les méthodes d'inférence fournissent une fonction d'appartenance résultante pour la variable de sortie x_R . Il s'agit donc d'une information floue. Etant donné que l'organe de commande nécessite un signal de commande précis à son entrée, il faut prévoir une transformation de cette information floue en une information déterminée. Cette transformation est appelée défuzzification. Ainsi, la défuzzification est un module qui permet de convertir une commande floue en une valeur réelle. Plusieurs méthodes de défuzzification ont été proposées. Parmi les plus utilisées, nous citons [103, 104] :

- méthode du maximum,
- méthode de la moyenne des maximas,
- méthode du centre de gravité,
- méthode des hauteurs.

V.2.2.2 Le contrôleur de Takagi-Sugeno

Une autre forme de systèmes logiques flous qui a suscité beaucoup d'attention est celle de Takagi-Sugeno [105]. La propriété fondamentale du modèle flou de Takagi-Sugeno est d'exprimer la dynamique locale de chaque implication de la règle floue par un modèle linéaire. Nous trouvons dans cette méthode une forme particulière des règles de Mamdani [106], où les conséquences ne sont pas des ensembles flous mais des fonctions analytiques des entrées.

Implication floue

Le format d'une implication floue R suggérée par Takagi-Sugeno est le suivant :

$$R : \text{Si } x_1 \text{ est } A_1 \text{ ET } \dots \text{ ET } x_k \text{ est } A_k \text{ Alors } y = f(x_1, x_2, \dots, x_k) \quad (\text{V.1})$$

où, R est une fonction analytique des entrées x_j , $j = 1, \dots, k$, A_j sont les ensembles flous, y est une variable de la conséquence inférée à partir de la règle, $x = [x_1, x_2, \dots, x_k]^T$ sont les variables de la prémisse.

Raisonnement flou

Si nous supposons que nous avons m implications floues, alors nous avons une valeur de la conséquence inférée à partir de chaque implication, nous écrivons :

$$\left\{ \begin{array}{l} R_1 : Si \ x_1 \text{ est } A_{11} \ ET...ET \ x_k \text{ est } A_{1k} \ Alors \ y_1 = f_1(x_1, x_2, \dots, x_k) \\ R_2 : Si \ x_1 \text{ est } A_{21} \ ET...ET \ x_k \text{ est } A_{2k} \ Alors \ y_2 = f_2(x_1, x_2, \dots, x_k) \\ \cdot \\ \cdot \\ \cdot \\ R_m : Si \ x_1 \text{ est } A_{m1} \ ET...ET \ x_k \text{ est } A_{mk} \ Alors \ y_m = f_m(x_1, x_2, \dots, x_k) \end{array} \right. \quad (V.2)$$

La valeur finale y inférée à partir des m -implications est égale à la moyenne de toutes les conséquences pondérées résultantes des implications individuelles :

$$y = \frac{\sum_{l=1}^m \mu_l y_l}{\sum_{l=1}^m \mu_l} \quad (V.3)$$

$\mu_l = T(\mu_{A_{l1}}(x_1), \mu_{A_{l2}}(x_2), \dots, \mu_{A_{lk}}(x_k))$, $l = 1, 2, \dots, m$ et m désigne le nombre des règles utilisées, μ_l est le poids affecté à la l^{eme} règle, T est une t-norme choisit égale à l'opérateur produit ou bien l'opérateur *min*, x_j sont les entrées avec $j = 1, 2, \dots, k$ et k est le nombre des entrées, y_l est une variable de la conséquence inférée à partir de la l^{eme} règle.

V.2.3 Adaptation des paramètres de l'impédance

Comme mentionné précédemment, nous introduisons un algorithme adaptatif flou comme une solution de contrôle intelligent pour choisir le comportement désiré Z_d [107, 108]. Le schéma fonctionnel du système contrôlé est illustré par la figure (V.2). Nous pouvons constater que les entrées du contrôleur flou sont les forces virtuels F_{obs} et les distances mesurées nommées d_4 et d_5 et la sortie est l'impédance désirée Z_d . Les distances d_4 et d_5 sont les distances mesurées avec les capteurs latéraux. Les trois entrées du contrôleur flou, F_{obs} , d_4 et d_5 , sont évaluées par rapport aux sous-ensembles flous à partir de six, cinq fonctions d'appartenance respectivement. La figure (V.3) montre les fonctions d'appartenance pour l'entrée F_{obs} et la figure (V.4) montre les fonctions d'appartenance pour les entrées d_4 et d_5 . La forme des fonctions d'appartenance est choisit triangulaire pour les variables d'entrée et singleton pour la variable de sortie. Les valeurs linguistiques peuvent être décrites par rapport aux sous-ensembles flous $L_{F_{obs}}$ et

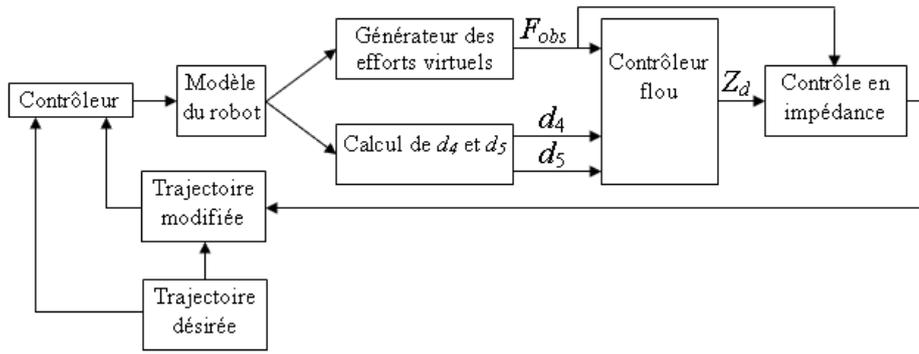


FIGURE V.2 : Schéma fonctionnel du système à contrôler

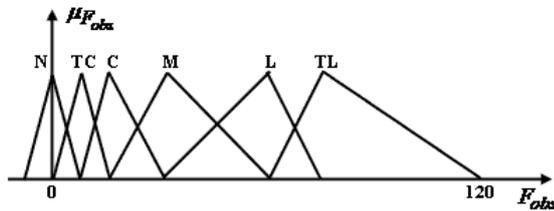


FIGURE V.3 : Fonctions d'appartenance pour la variable F_{obs}

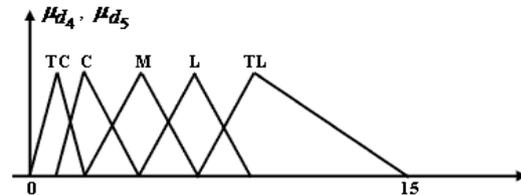


FIGURE V.4 : Fonctions d'appartenance pour les variables d_4 et d_5

$L_{d_{4,5}}$ comme : $L_{F_{obs}} = \{N, TC, C, M, L, TL\}$ et $L_{d_{4,5}} = \{TC, C, M, L, TL\}$. Le sens de chaque valeur linguistique est évident à partir de son mnémotechnique ; par exemple, N pour Nul, TC pour Très Cours, C pour Cours, M pour Moyen, L pour Long, TL pour Très Long. Nous construisons aussi cinq valeurs numériques pour le comportement désiré Z_d , $L_{Z_d} = \{TP, P, M, G, TG\}$. Le sens de chaque valeur linguistique est : TP pour Très Petit, P pour Petit, M pour Moyen, G pour Grand, TG pour Très Grand.

La deuxième étape de la conception du contrôleur flou est le mécanisme d'inférence. Nous pouvons décrire l'inférence à l'aide d'un ensemble de règles :

$$\begin{aligned}
 R_1 : & \text{ Si } F_{obs} \text{ est } TC \text{ ET } d_4 \text{ est } TL \text{ ET } d_5 \text{ est } TL \text{ Alors } Z_{d1} \text{ est } TP \\
 R_2 : & \text{ Si } F_{obs} \text{ est } C \text{ ET } d_4 \text{ est } TL \text{ ET } d_5 \text{ est } TL \text{ Alors } Z_{d2} \text{ est } P \\
 R_3 : & \text{ Si } F_{obs} \text{ est } M \text{ ET } d_4 \text{ est } TL \text{ ET } d_5 \text{ est } TL \text{ Alors } Z_{d3} \text{ est } M \\
 R_4 : & \text{ Si } F_{obs} \text{ est } L \text{ ET } d_4 \text{ est } TC \text{ ET } d_5 \text{ est } TC \text{ Alors } Z_{d4} \text{ est } G \\
 R_5 : & \text{ etc.}
 \end{aligned} \tag{V.4}$$

La dernière étape d'un système d'inférence flou est la défuzzification. Cette étape

consiste à agréger les règles et d'en tirer une valeur réelle présentant la sortie du contrôleur flou. Nous nous basons sur les règles ci-dessus, la stratégie de Sugeno est choisie tel que décrit dans [105] afin d'en tirer la sortie Z_d , nous avons :

$$Z_d = \frac{\sum_{l=1}^{n_{regles}} \mu_l Z_{d_l}}{\sum_{l=1}^{n_{regles}} \mu_l} \quad (V.5)$$

$l = 1, 2, \dots, n_{regles}$, n_{regles} désigne le nombre des règles utilisées, Z_{d_l} est une variable de la conséquence inférée à partir de la l^{eme} règle, μ_l est le poids affecté à la l^{eme} règle.

V.2.4 Résultats de simulation

Pour montrer l'efficacité de la méthode dans des situations plus adéquates, nous avons testé l'évolution du manipulateur mobile omnidirectionnel dans un couloir. Les trajectoires imposées pour la plate-forme à partir de $t_0 = 0$: $[x^*(t), y^*(t), \vartheta^*(t)]^T = [0.1t, 0.1t, \pi/4]^T$. Le mouvement articulaire imposé pour le bras manipulateur est $q_r^* = [q_{r1}^*, q_{r2}^*]^T = [\pi/2, -\pi/2]^T$. La figure (V.5) montre la navigation du manipulateur mobile dans un couloir. Les vitesses angulaires w_{v_1} , w_{v_2} , w_{v_3} et la position de la plate-forme (x, y) sont respectivement reportées sur les figures (V.6), (V.7) et (V.8). Les variations de la force F_{obs} et l'impédance Z_d sont respectivement reportées sur les figures (V.9) et (V.10).

V.3 Identification des paramètres dynamiques

Les performances des lois de commande basées sur l'utilisation du modèle dynamique dépendent en partie de la qualité des valeurs estimées des paramètres du modèle qui décrit la dynamique du robot. La robustesse quant à elle dépend en partie de la qualité des incertitudes fournies par les estimations. Nous comprenons alors l'importance de la bonne connaissance de ces paramètres et des incertitudes associées. Pour déterminer les paramètres du modèle dynamique, trois méthodes sont possibles :

- La mise en œuvre d'essais expérimentaux spécifiques sur chaque corps du robot pris isolément afin d'identifier les paramètres dynamiques. Cependant, cette méthode n'est envisageable que dans l'étape d'avant-projet précédant le montage du robot.
- A partir de considérations géométriques sur les différents corps, l'utilisation de la CAO pour calculer les paramètres inertiels des robots. Cette méthode n'offre guère

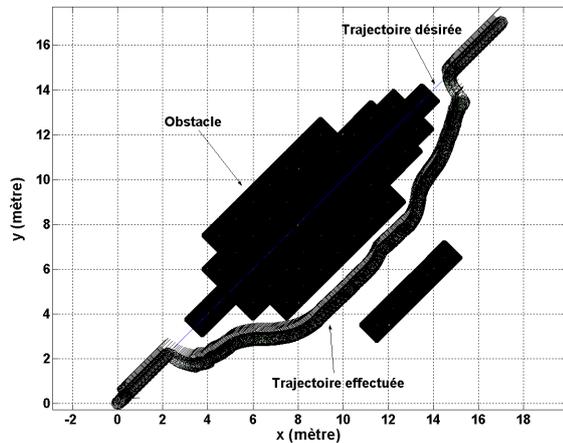


FIGURE V.5 : Exemple de navigation du manipulateur mobile dans un couloir

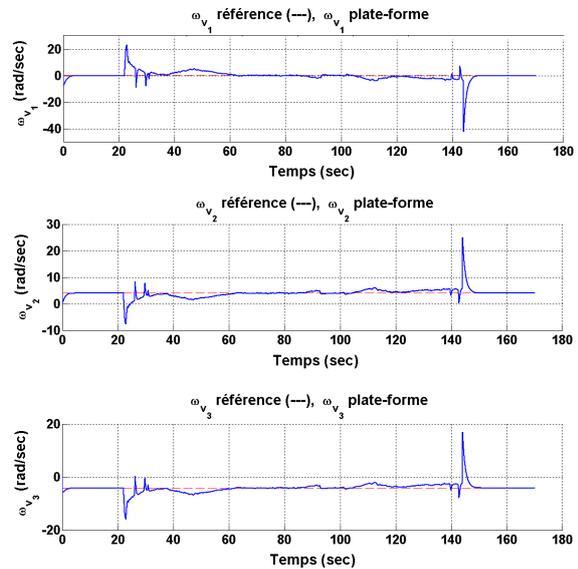


FIGURE V.6 : Vitesses angulaires ω_{v_1} , ω_{v_2} et ω_{v_3}

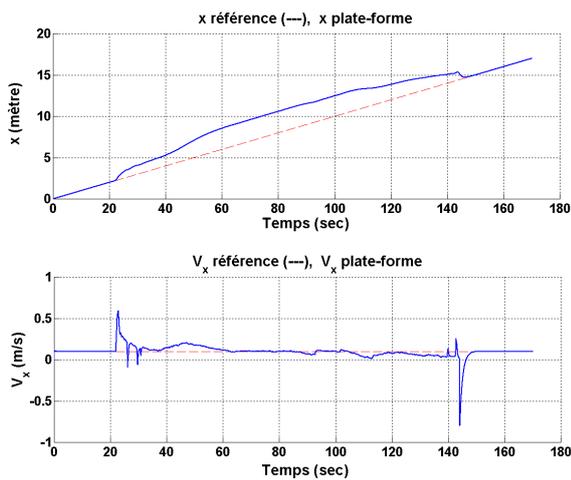


FIGURE V.7 : Position x et vitesse \dot{x} de la plate-forme

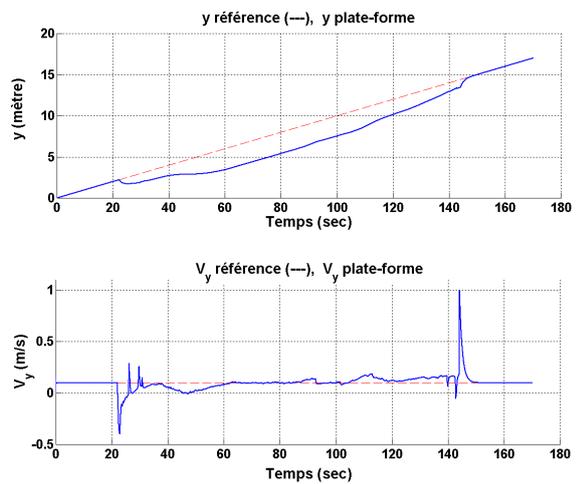


FIGURE V.8 : Position y et vitesse \dot{y} de la plate-forme

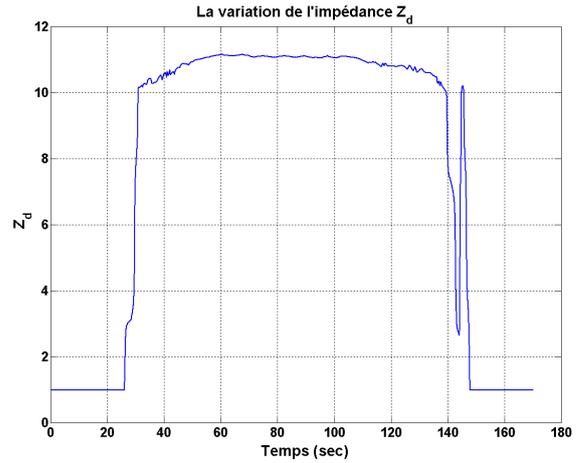
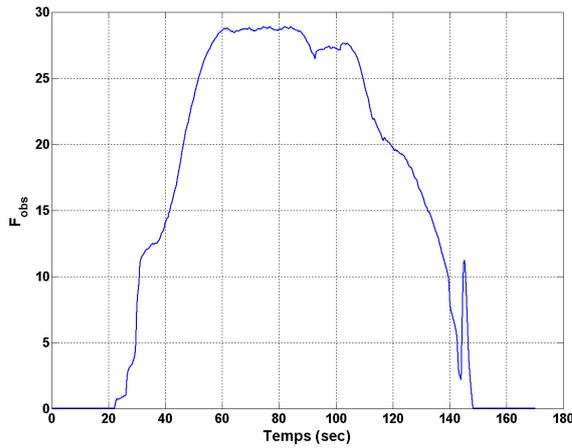


FIGURE V.9 : La variation de la force F_{obs} **FIGURE V.10** : La variation de l'impédance Z_d

de précision sur l'estimation des paramètres recherchés. Les deux méthodes précédentes conduisent à des résultats généralement sans estimation de la précision obtenue et sont souvent difficiles à mettre en œuvre.

- Vu les difficultés de mise en œuvre ou les imprécisions des deux méthodes précédentes, il est préférable d'utiliser des techniques d'identification.

V.3.1 Identification des paramètres dynamiques du manipulateur mobile omnidirectionnel

De l'équation dynamique du manipulateur mobile (II.69), nous obtenons :

$$M(q)\dot{\omega} + C(q, \omega) = Y(q, \omega, \dot{\omega})Pe = \tau \quad (\text{V.6})$$

où $Y(q, \omega, \dot{\omega})$ est une matrice de dimension $n_Y \times m_Y$ des fonctions connues. Pe est un vecteur de dimension $m_Y \times 1$ des paramètres du manipulateur mobile.

Nous avons appliqué un algorithme d'identification de type moindres carrés récursifs [109]. Le système à identifier est mis sous la forme de produit d'un vecteur des mesures ψ et d'un vecteur de paramètres à identifier Pe .

$$\tau = \psi(k)^T Pe \quad (\text{V.7})$$

avec τ est un vecteur de dimension $n_Y \times 1$, $\psi(k)$ est une matrice de dimension $m_Y \times n_Y$

des fonctions connues, et Pe est un vecteur de dimension $m_Y \times 1$ des paramètres inconnus.

Le modèle est indexé par la variable k , qui représente le temps d'échantillonnage.

Les paramètres inconnus Pe peuvent être estimés en minimisant la fonction de coût suivante [110] :

$$J_k(Pe) = \frac{1}{2} \sum_{i=1}^k [\tau(i) - \psi(i)^T Pe]^T [\tau(i) - \psi(i)^T Pe] + \frac{1}{2} [Pe - \hat{P}e(0)]^T P(0)^{-1} [Pe - \hat{P}e(0)] \quad (V.8)$$

avec $P(0) = P(0)^T > 0$ et $\hat{P}e(0)$ représente l'estimation initiale.

la valeur estimée du vecteur Pe sur la base des observations du vecteur $\tau(k)$ et de la variable $\psi(k)$ est obtenue en minimisant la fonction de coût (V.8), que nous désignons comme $\hat{P}e(k)$ satisfaisant à l'équation suivante :

$$\begin{aligned} \hat{P}e(k) &= P(k) \left[P(0)^{-1} \hat{P}e(0) + \sum_{i=1}^k \psi(i) \tau(i) \right] \\ P(k) &= \left[P(0)^{-1} + \sum_{i=1}^k \psi(i) \psi(i)^T \right]^{-1} \end{aligned} \quad (V.9)$$

où $P(k)$ est une matrice de dimension $n_Y \times n_Y$ nommée matrice de covariance.

La version récursive de la méthode des moindres carrés est donnée par les équations suivantes [110] :

$$\hat{P}e(k) = \hat{P}e(k-1) + P(k-1) \psi(k) \times [I + \psi(k)^T P(k-1) \psi(k)]^{-1} e(k) \quad (V.10)$$

$$\begin{aligned} P(k) &= P(k-1) - P(k-1) \psi(k) \\ &\times [I + \psi(k)^T P(k-1) \psi(k)]^{-1} \psi(k)^T P(k-1) \end{aligned} \quad (V.11)$$

où l'erreur de prédiction $e(k)$ est définie comme :

$$e(k) = \tau(k) - \psi(k)^T \hat{P}e(k-1) \quad (V.12)$$

Dans le cas particulier de notre manipulateur mobile, les paramètres dynamiques sont donnés par :

$$\left\{ \begin{array}{l} \hat{P}e_1 = m_1 \\ \hat{P}e_2 = m_2 \\ \hat{P}e_3 = I_1 \\ \hat{P}e_4 = I_2 \\ \hat{P}e_5 = J_R \\ \hat{P}e_6 = m_R \\ \hat{P}e_7 = J_m \\ \hat{P}e_8 = \frac{C_m C_e}{R_a} + b_m \end{array} \right.$$

Les paramètres $\hat{P}e_i$, $i = 1..8$ peuvent être interprétés comme une combinaison de certaines propriétés physiques du manipulateur mobile qui sont invariants dans le temps. Les paramètres individuels du robot sont définis dans la section suivante. Les valeurs des paramètres du robot ont été obtenues par mesure directe. Dans la section suivante, nous présentons quelques résultats de simulation.

V.3.2 Résultats de simulation

Les simulations ont été réalisées en considérant un ensemble de paramètres physiques pour le modèle dynamique du manipulateur mobile omnidirectionnel donnés par : la masse du bras1 $m_1 = 10 \text{ kg}$, la masse du bras2 $m_2 = 5 \text{ kg}$, le moment d'inertie du bras1 $I_1 = 0.05 \text{ kg.m}^2$, le moment d'inertie du bras2 $I_2 = 0.025 \text{ kg.m}^2$, la longueur d'un bras $l_1 = l_2 = 0.5 \text{ m}$, la masse de la plate-forme $m_R = 20 \text{ kg}$, le rayon de la plate-forme $R = 0.3 \text{ m}$, le moment d'inertie de la plate-forme $I_R = 0.9 \text{ kg.m}^2$, le moment d'inertie du moteur $I_m = 1.380e-5 \text{ kg.m}^2$, la résistance interne du moteur $R_a = 0.317 \Omega$, le coefficient du frottement visqueux du moteur $b_m = 0.004 \text{ N.m}$, la constante de force électromotrice (EMF) $C_e = 3.02e-2 \text{ V.s/rad}$, la constante de couple du moteur $C_m = 3.02e-2 \text{ N.m/A}$, le rayon d'une sphère $r = 0.03 \text{ m}$, $R_{i_{max}} = 0.2 \text{ m}$ et $R_{i_{min}} = 0.13 \text{ m}$, $a_1 = a_2 = 0$. Les résultats de simulation sont présentés par les figures (V.11) à (V.14). Les trajectoires imposées pour la plate-forme à partir de $t_0 = 0$: $x^*(t) = 0.05 \sin(1.6t + 0.2) + 0.09 \sin(2t + 0.15)$ et $y^*(t) = 0.08 \sin(2.6t + 0.02) + 0.02 \sin(1.2t + 0.35)$. Le mouvement articulaire imposé pour le bras manipulateur est $q_r^* = [q_{r_1}^*, q_{r_2}^*]^T = [0.1 \sin(2t + 0.1), 0.2 \cos(t + 0.1)]^T$. La matrice de covariance initiale $P(0) = \text{diag}(1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000)$ et le vecteur initial des paramètres estimés $\hat{P}e(0) = 0$.

V.4 Conclusion

Dans ce chapitre, nous avons présenté notre approche qui consiste en la conception d'un contrôleur en se basant sur la commande linéarisante ainsi que sur l'approche du contrôle flou. Dans la première couche de ce contrôleur figure la commande par retour d'état linéarisant. Nous avons ajouté la stratégie du contrôle dynamique augmenté d'une boucle externe de contrôle en impédance pour l'évitement d'obstacles. La deuxième couche

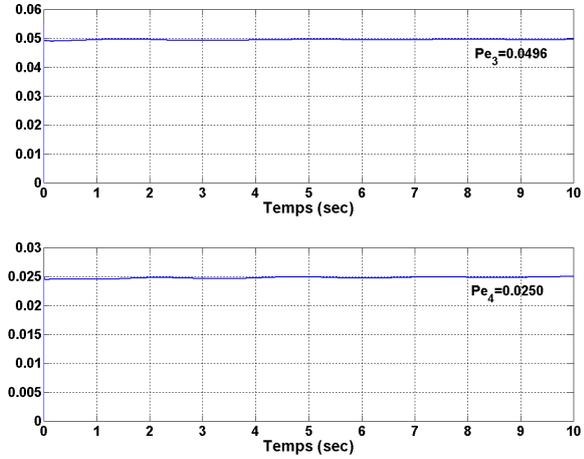
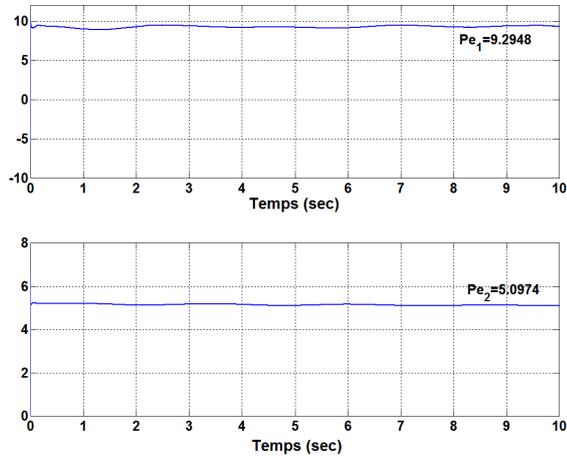


FIGURE V.11 : Les valeurs estimées \hat{P}_{e1} et \hat{P}_{e2} FIGURE V.12 : Les valeurs estimées \hat{P}_{e3} et \hat{P}_{e4}

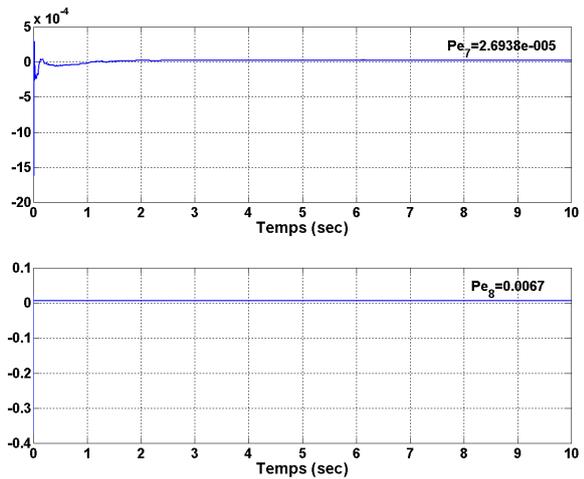
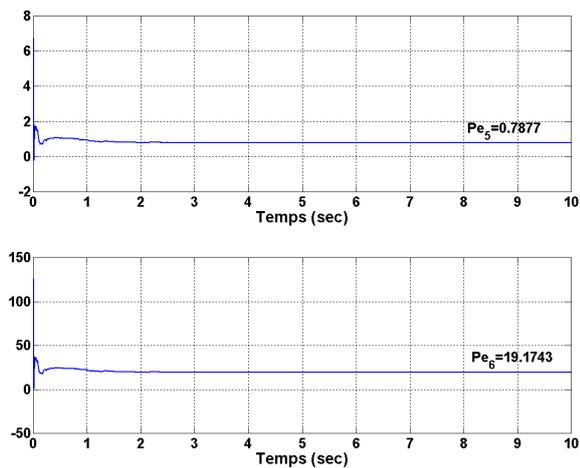


FIGURE V.13 : Les valeurs estimées \hat{P}_{e5} et \hat{P}_{e6} FIGURE V.14 : Les valeurs estimées \hat{P}_{e7} et \hat{P}_{e8}

présente un contrôleur flou dont la base des règles est construite dans le but d'adapter les paramètres de l'impédance. Dans ce chapitre aussi, nous avons présenté l'identification des paramètres dynamiques du manipulateur mobile omnidirectionnel en utilisant l'approche des moindres carrés récurrents.

Chapitre VI

Les systèmes multi-agents

VI.1 Introduction

Un système multi-agents est caractérisé par des formes de comportements et d'interactions simples ou complexes suivant le type de l'agent : réactif ou cognitif. De tels systèmes arrivent à une résolution collective des problèmes à l'aide de comportements individuels simples. Tirant profit des propriétés de coordination, de collaboration et de leur prise de décision, les SMA peuvent apporter beaucoup dans les situations d'évolution rapides.

Dans ce chapitre, nous nous intéressons à l'accomplissement d'une tâche coopérative par un système multi-agents. Le système est constitué d'un ensemble de manipulateurs mobiles, chacun peut communiquer avec les autres afin de réaliser la tâche demandée. Cette dernière réside dans la coopération entre deux robots ou plus dans une manipulation continue de portage. Ce chapitre se décompose en deux sections traitant d'une part des systèmes multi-agents et de la robotique collective, d'autre part de la méthode proposée.

VI.2 Systèmes multi-agents et applications de robotique collective

Les Systèmes Multi-Agents (SMA) et la robotique collective sont des domaines d'étude complémentaires. En effet, la nature intrinsèquement distribuée des systèmes robotiques collectifs en fait un champ d'application privilégié des SMA dont l'objectif est d'étudier l'organisation de systèmes autonomes situés dans un environnement.

Ainsi, certaines études de l'Intelligence Artificielle Distribuée (IAD) se sont orientées vers des applications multi-robots. Réciproquement les recherches en robotique, initialement orientées vers la conception de systèmes permettant à un robot unique de réaliser l'ensemble des tâches d'une mission, se sont progressivement intéressés aux apports éventuels de l'IAD et ont donc envisagé la réalisation d'applications de robotique collective.

Cette section a pour objectif de présenter les relations entre l'étude des SMA et ses applications dans le domaine de la robotique. Nous allons présenter les différentes notions spécifiques à l'IAD telles : la définition d'un système multi-agents, les notions d'agents cognitifs et réactifs. Puis nous présenterons quelques applications à la robotique modulaire et à la robotique collective mobile.

VI.2.1 Etude des systèmes multi-agents

Contrairement à l'Intelligence Artificielle (IA), qui modélise le comportement par une centralisation de l'expertise, l'Intelligence Artificielle Distribuée (IAD) s'intéresse à des comportements intelligents qui sont le résultat de la coopération de plusieurs entités appelées agents. L'IAD s'appuie donc sur la distribution de l'intelligence entre agents formant une société (le système multi-agents) dans lequel chacun possède une autonomie et travaille d'une façon concurrente.

VI.2.1.1 Définition d'un agent

Un agent est un système informatique, *situé* dans un environnement, qui agit d'une façon *autonome* et *flexible* pour atteindre les objectifs pour lesquels il a été conçu [111].

Il y a trois concepts clés présents dans cette définition :

- Situé signifie que l'agent peut recevoir des entrées sensorielles provenant de son environnement et qu'il peut effectuer des actions qui sont susceptibles de changer cet environnement.
- Autonome signifie que l'agent est capable d'agir sans l'intervention directe d'un humain (ou d'un autre agent) et qu'il a le contrôle de ses actions et de son état interne.
- Flexible signifie que l'agent :

- peut percevoir son environnement et répondre rapidement aux changements qui s’y produisent,
- n’agit pas simplement en réponse à son environnement,
- est capable d’interagir avec les autres agents afin de compléter ses tâches ou aider les autres à compléter les leurs.

VI.2.1.2 Définition d’un système multi-agents (SMA)

Dans la plupart des situations réelles, l’agent n’est pas seul dans son environnement, il y a d’autres agents présents autour de lui. Il nous faut donc aborder des systèmes où plusieurs agents doivent interagir entre eux pour effectuer leurs tâches. De tels systèmes sont appelés *systèmes multi-agents*. Une définition plus formelle est donnée par Ferber [112] :

Définition VI.1. *Nous appelons système multi-agents, un système composé des éléments suivants :*

- *Un environnement E .*
- *Un ensemble d’objets O , ces objets sont situés, c’est à dire, pour tout objet, il est possible, à un moment donné, de leur associer une position dans l’environnement. Ces objets sont passifs, c’est à dire qu’ils peuvent être perçus, manipulés, modifiés, détruits par les agents.*
- *Un ensemble d’agents A , qui sont des objets actifs du système. Ces objets sont capables de percevoir, produire, consommer, transformer et manipuler les objets passifs de l’environnement.*
- *Un ensemble de relations R qui unit les agents entre eux.*
- *Un ensemble d’opérations O_p permettant aux agents de A de percevoir, manipuler et transformer les objets de O .*

Une illustration d’un système multi-agents est donnée par la figure (VI.1) [112] .

VI.2.1.3 Architectures d’agents

Il existe plusieurs manières de concevoir des agents, un agent peut toujours être vu comme une fonction liant ses perceptions à ses actions. Plus précisément, un agent perçoit l’environnement à l’aide de ses capteurs et il agit sur son environnement à l’aide de ses

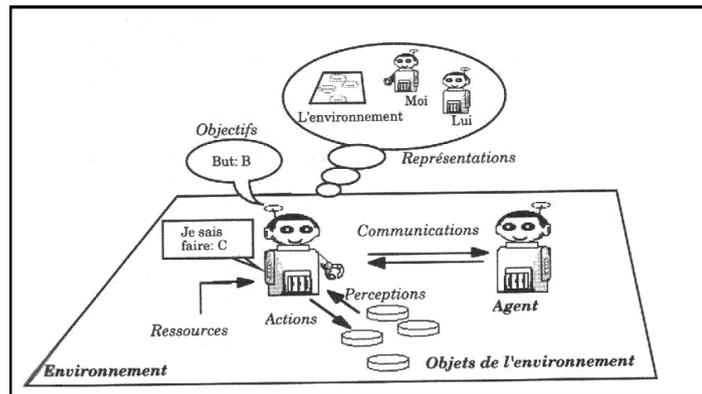


FIGURE VI.1 : Représentation d'un système multi-agents

effecteurs. Ce qui différencie les différentes architectures d'agents, c'est la manière dont les perceptions sont liées aux actions.

Les auteurs Russel et Norvig [1] regroupent les architectures d'agents en quatre types, à savoir :

- a) les agents à réflexes simples,
- b) les agents conservant une trace du monde,
- c) les agents ayant des buts,
- d) les agents utilisant une fonction d'utilité.

Les sections suivantes détaillent ces quatre architectures, mais en les regroupant deux à deux. Les deux premiers types d'agents sont considérés comme des agents réactifs et les deux derniers types sont considérés comme des agents cognitifs.

Agents réactifs

Un agent réactif ne fait que réagir aux changements qui surviennent dans l'environnement. Autrement dit, un tel agent ne fait ni délibération ni planification, il se contente simplement d'acquérir des perceptions et de réagir à celles-ci en appliquant certaines règles prédéfinies. Etant donné qu'il n'y a pratiquement pas de raisonnement, ces agents peuvent agir et réagir très rapidement.

a) Agents à réflexes simples

Ce type d'agent agit uniquement en se basant sur ses perceptions courantes. Il utilise un ensemble de règles prédéfinies, du type *Si "condition" Alors "action"*, pour choisir ses actions. A chaque fois, l'agent ne fait qu'exécuter l'action correspondant à la règle activée par ses perceptions.

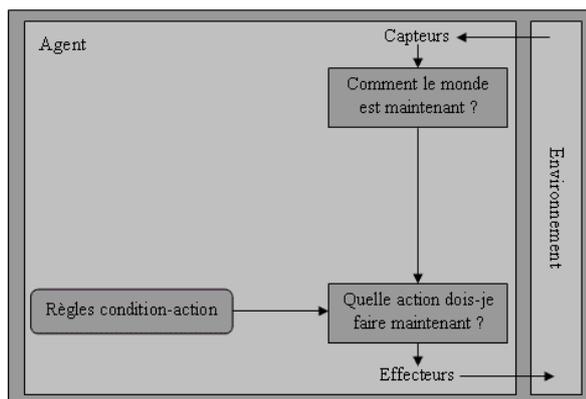


FIGURE VI.2 : Schéma d'un agent à réflexes simples [1]

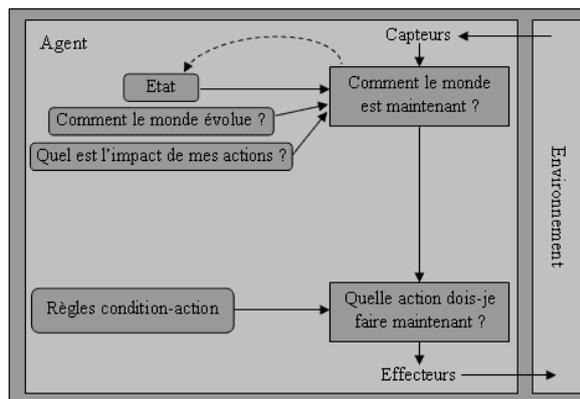


FIGURE VI.3 : Schéma d'un agent conservant une trace du monde [1]

La figure (VI.2) montre l'architecture d'un agent à réflexe simple. Les rectangles représentent l'état interne de l'agent dans son processus de décision et les ovales représentent les informations qui sont utilisées dans le processus. L'agent se bâtit une représentation du monde à l'aide de ses perceptions lui venant de ses capteurs. Par la suite, il utilise ses règles pour choisir l'action qu'il doit effectuer selon ce qu'il perçoit de l'environnement.

b) *Agents conservant une trace du monde*

Le type d'agent qui a été décrit précédemment ne peut fonctionner que s'il peut choisir ses actions en se basant uniquement sur sa perception actuelle. Les capteurs de l'agent ne fournissent pas une vue complète du monde, alors il doit maintenir un état interne qui dépend de l'historique des perceptions. Pour que l'agent puisse faire évoluer ses informations internes sur l'état du monde, il a besoin de deux types d'informations :

- * informations sur la manière dont le monde évolue indépendamment de l'agent.
- * informations sur la manière dont ses propres actions affectent l'environnement.

La figure (VI.3) montre l'architecture d'un agent conservant une trace du monde. Il utilise ses informations internes (état précédent du monde, l'évolution du monde et l'impact de ses actions) pour mettre à jour ses perceptions actuelles. Par la suite, il choisit ses actions en se basant sur cette perception améliorée du monde qui l'entoure.

Agents cognitifs

Les agents cognitifs sont des agents qui effectuent une certaine délibération pour choisir leurs actions. Une telle délibération peut se faire en se basant sur les buts de l'agent ou sur une certaine fonction d'utilité. Elle peut prendre la forme d'un plan qui reflète la suite d'actions que l'agent doit effectuer en vue de réaliser son but.

c) *Agents ayant des buts*

Dans la section précédente, les agents utilisaient leurs connaissances sur l'état actuel de l'environnement pour choisir leurs actions. Connaître l'état courant de l'environnement n'est pas toujours suffisant pour décider de l'action à accomplir. L'agent a besoin d'une information sur le but qui décrit les situations désirables. Il peut combiner les informations sur les buts avec les informations sur les résultats de ses actions pour choisir les actions qui vont lui permettre d'atteindre ses buts. Contrairement aux agents réactifs, les agents cognitifs, qui raisonnent sur les buts, tiennent compte d'une certaine projection dans le futur.

La figure (VI.4) montre l'architecture d'un agent basé sur les buts. La structure d'un agent basé sur les buts est identique à celle d'un agent réactif gardant une trace de l'environnement sauf qu'il se projette dans le futur pour voir l'impact de ses actions et qu'il choisit ses actions en se basant sur ses buts, contrairement à l'agent réactif qui ne faisait qu'appliquer des règles prédéfinies pour relier ses perceptions à ses actions.

d) *Agents utilisant une fonction d'utilité*

Les buts seuls ne sont pas suffisants pour générer un comportement intéressant dans beaucoup d'environnements. Une mesure de performance plus générale doit permettre une comparaison entre les différents états du monde en fonction de la satisfaction exacte de l'agent s'ils sont atteints. L'utilité est une fonction qui attribue une valeur numérique à chacun des états. Cette valeur décrit le degré de satisfaction associé à l'état. Plus l'état a une grande valeur, plus il est désirable pour l'agent.

La figure (VI.5) montre le schéma d'un agent basé sur l'utilité. L'agent utilise la fonction d'utilité pour évaluer la pertinence d'une action. Il choisit donc les actions qui l'amèneront dans les états ayant la plus grande valeur d'utilité pour lui.

Maintenant que nous avons donné les différents agents, nous allons décrire leur utili-

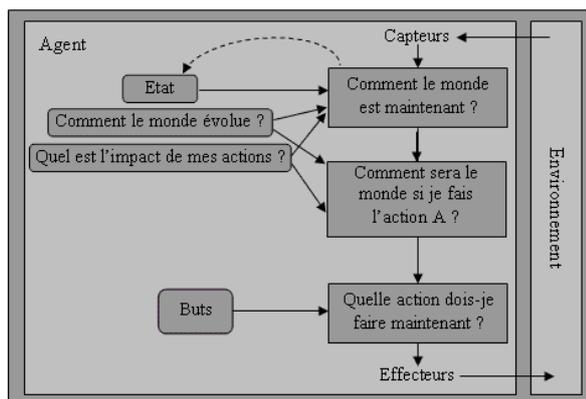


FIGURE VI.4 : Schéma d'un agent ayant des buts [1]

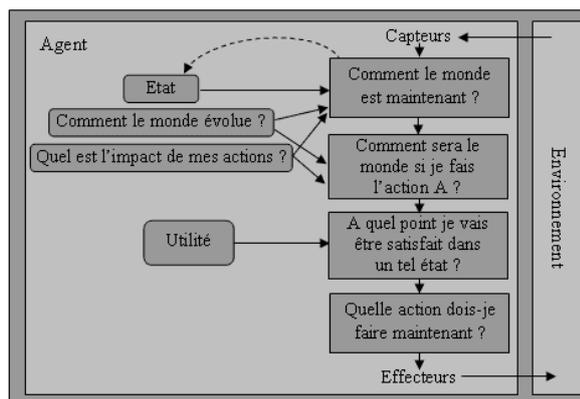


FIGURE VI.5 : Schéma d'un agent basé sur l'utilité [1]

sation en robotique.

VI.2.2 Applications à la robotique

Un agent physique est un agent qui possède une entité physique qui lui permet d'agir dans le monde réel [112]. Le mariage des techniques multi-agents et de la robotique a fait l'objet de nombreuses études. Les systèmes robotiques distribués se prêtent naturellement bien à l'utilisation des techniques multi-agents. Nous distinguons deux types de systèmes robotiques distribués : la robotique modulaire et les systèmes de robotique collective.

VI.2.2.1 Application à la robotique modulaire

L'application des SMA à la robotique modulaire reste un domaine d'étude important. Plusieurs travaux ont néanmoins été proposés sur ce thème [113]. Ces travaux s'appuient sur une décomposition du robot (généralement un bras manipulateur) en modules, chaque module représentant un agent. La résolution du problème général (le parcours d'une trajectoire ou le déplacement à une position donnée) est alors le produit coopératif des résolutions des sous-problèmes effectués par chacun des agents (Fig. VI.6).

VI.2.2.2 Application à la robotique collective

La plupart des recherches en SMA appliquées à la robotique portent sur les applications multi-robots. Cette approche, souvent inspirée ou motivée par l'étude des colonies

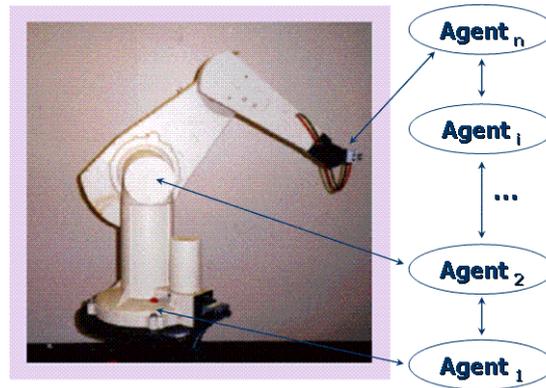


FIGURE VI.6 : Modélisation d'un robot manipulateur par un système multi-agents

d'animaux sociaux (comme nombre d'insectes), propose une approche collective et coopérative de la robotique pour la réalisation d'une mission.

L'exploration : Le problème de l'exploration est certainement celui qui présente le plus d'applications en robotique collective [114, 115, 116]. Son principe est, pour un ensemble de robots, de chercher et de rapporter un ensemble d'éléments (objets) dispersés dans un environnement non connu. Les objets peuvent être saisis et manipulés par un seul robot, ce qui fait que la mission peut être réalisée de façon individuelle.

L'origine de cette problématique est directement inspirée de l'éthologie et particulièrement des colonies de fourmis. Le problème de l'exploration est typiquement celui de la recherche de nourriture pour les fourmis. Les applications pratiques envisagées à partir de l'exploration multi-robots sont les tâches de nettoyage, de récolte ou de dépollution.

Le transport collectif : La tâche de transport collectif représente également un problème classique des applications multi-robots [117, 118, 119]. L'objectif est ici de transporter un ou plusieurs objets lourds, qu'un robot ne peut déplacer à lui seul. Cette problématique trouve sa transposition éthologique dans le transport de proie ou de nourriture. Les applications pratiques de ce type de problématique sont évidemment le transport de matériel.

Contrairement au problème d'exploration, un robot ne peut réaliser la mission de façon individuelle ce qui nécessite une collaboration implicite ou explicite entre les différents robots. Ce type d'application pose donc le problème de la coordination collective. En effet, les robots possèdent un but commun (déplacer l'objet), les ressources sont insuffisantes (plusieurs robots vont devoir se partager la même ressource) et les compétences des agents

sont insuffisantes pour réaliser la tâche tout seul. Les agents doivent donc coordonner leurs actions afin de disposer de la synergie de l'ensemble de leurs compétences.

Les mouvements en formation : Le mouvement en formation représente un nouvel exemple de collaboration coordonnée. Ce problème s'inspire des bancs de poissons ou du vol en formation des oiseaux. L'objectif est alors de permettre à un groupe de robots de se déplacer en respectant les contraintes liées à l'environnement (obstacles) ainsi qu'aux spécificités de la mission, tout en restant dans une configuration donnée (ligne, cercle, etc). Le problème peut être complété en demandant aux robots d'adopter différentes formations de déplacement en fonction de l'état de la mission et/ou de l'environnement.

La gestion de trafic : Le dernier problème de la robotique collective est celui de la gestion du trafic dans un environnement encombré. Ce problème regroupe un ensemble de robots devant réaliser des mouvements a priori indépendants dans un environnement généralement défini sous la forme d'un réseau routier. Les robots doivent donc éviter les collisions et gérer de façon générale l'accès à l'espace routier.

Après avoir introduit les systèmes multi-agents, nous allons voir maintenant comment les utiliser dans notre problématique de coopération entre deux robots ou plus dans une manipulation continue de portage.

VI.3 Description de la méthode proposée

VI.3.1 Objectif de la mission

L'objectif de cette partie est de mettre en œuvre une simulation de deux robots ou plus permettant de prendre d'une manière autonome un objet d'une position initiale S et l'amener à une position finale G . Pour effectuer cette mission, les robots doivent mettre en œuvre trois comportements de base correspondant aux trois tâches : déplacement vers le but, transport de l'objet et l'évitement d'obstacles (Fig. VI.7). Ces trois comportements sont réalisés par le module d'action (Fig. I.16). Le contrôle de robots dans un environnement inconnu a besoin de gérer ces trois comportements de base de manière réactive. Fondamentalement, le contrôleur favorise le comportement d'évitement d'obstacles à proximité d'un obstacle ou le comportement de négociation près d'un autre robot puis il favorise le comportement de déplacement.

*Le transport de
l'objet de la position
S vers la position G
avec l'évitement
d'obstacles*

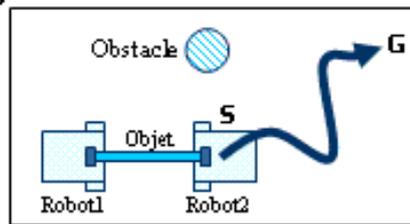


FIGURE VI.7 : Navigation dans un environnement encombré d'obstacles

Dans ce qui suit, nous détaillons les différents comportements de notre robot.

VI.3.2 Comportements du robot

Trois fonctions différentes pour transporter l'objet par les robots. La première fonction est une exploration qui consiste à chercher l'objet d'une façon aléatoire, la deuxième est le déplacement vers l'objet une fois trouvé puis le premier robot qui l'a trouvé diffuse un message d'aide. Nous remarquons que le comportement de déplacement n'est pas une tâche facile à cause des obstacles multiples. Les comportements du robot sont repris dans cette partie.

- Robot inactif
 - Si pas d'objet transporté, déplacement aléatoire et recherche de l'objet.
 - Si détection d'un objet dans son voisinage non transporté ;
 - * Positionnement sur l'une des extrémités pour le transporter,
 - * Appel d'un collègue (le plus près inactif vient).
- Robot actif
 - Transport de l'objet.
 - * Déplacement en évitant les obstacles,
 - * Négociation si robot empêché de passer.

Les comportements d'un objet sont repris dans la partie suivante.

- Objet inactif
 - Objet libre
- Objet actif

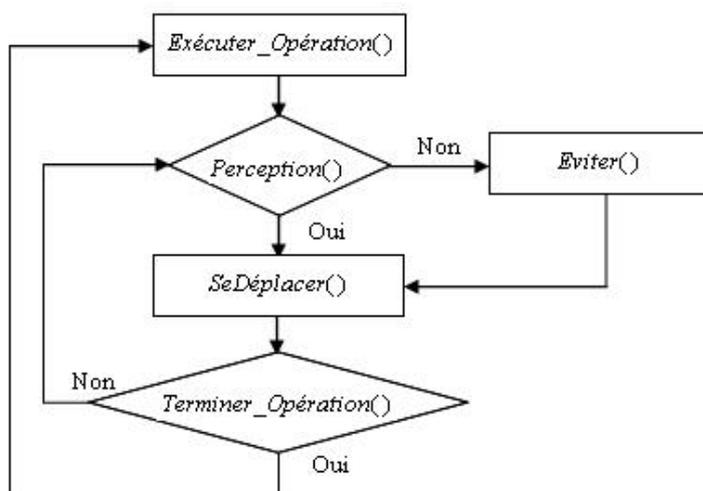


FIGURE VI.8 : Organigramme de déplacement pour le robot

- Une seule extrémité est active,
 - * L'objet attend un autre robot.
- Les deux extrémités sont actives,
 - * L'objet attend deux robots.

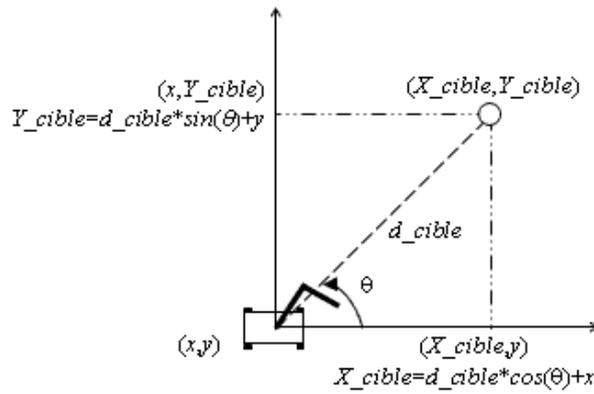
Le comportement de déplacement des robots est donné par la figure (VI.8). La procédure "*Exécuter_Opération*" définit le comportement désiré du robot. Elle peut être :

- une recherche aléatoire de l'objet,
- déplacement vers l'objet,
- transport de l'objet vers l'endroit désiré.

La procédure "*Perception*" consiste à obtenir quelques informations sur le voisinage du robot. Le robot doit vérifier qu'aucun obstacle n'est présent sur son passage. Si un obstacle est détecté, la procédure appelée "*Eviter*" est activée. Sinon, le robot se déplace (la procédure "*SeDéplacer*") afin d'être plus proche que l'emplacement objectif.

VI.3.3 Fixer une cible

Une cible principale appelée "cible primaire" est définie au début de notre programme. Cette cible est déterminée en fonction de l'action à réaliser par l'agent. Dans notre cas la fonction du robot est de trouver la position de l'objet. Dans le cas de plusieurs objets, nous parlons de plusieurs cibles principales. Ensuite, des cibles intermédiaires sont empilées de

FIGURE VI.9 : Détermination de l'angle θ

manière à atteindre cette cible primaire. Les coordonnées de ces cibles sont définies soit par des négociations entre les robots, soit par des algorithmes d'évitement d'obstacles. Dans notre implémentation, ces cibles intermédiaires sont appelées "cibles secondaires".

VI.3.4 Recherche aléatoire d'un objet

Le robot se déplace aléatoirement dans l'environnement. Quand il détecte l'objet, il se déplace vers l'extrémité la plus proche. Le comportement de déplacement est constitué de deux opérations exécutées séquentiellement, "*faire tourner*" et "*avancer*". Quand le robot détecte les coordonnées de sa cible (l'objet), il tourne et se dirige vers cet endroit. Une méthode pour calculer l'angle θ entre le repère du robot et une cible est montrée par la figure (VI.9). L'angle θ correspond à l'angle vers lequel le robot doit regarder pour atteindre le centre de sa cible. Ainsi, cet angle est utilisé dans la détection locale pour repérer la présence ou non d'un obstacle.

$$\theta = \tan^{-1} \left(\frac{Y_{cible} - y}{X_{cible} - x} \right) \quad (\text{VI.1})$$

avec x et y sont les coordonnées de la position du robot, X_{cible} et Y_{cible} sont les coordonnées de la position de la cible, d_{cible} est la distance entre le robot et la cible.

VI.3.5 Evitement d'obstacles

L'évitement d'obstacles est un comportement de base présent dans tous les objets mobiles car cette fonctionnalité est indispensable pour permettre aux robots de fonctionner

dans un environnement dynamique de manière autonome. Nous avons traduit l'évitement d'un obstacle par le contournement de ce dernier. Ce contournement doit pouvoir s'effectuer quelque soit le type et le positionnement des obstacles dans l'environnement. Tous les objets, tels que les robots, les objets à transporter ou les autres obstacles sont modélisés par des cercles ayant un rayon spécifique. Cet attribut indique alors le rayon d'occupation de l'objet à partir de ses coordonnées dans l'environnement. Cette manière de procéder permet de simuler le système au plus près de la réalité. En effet, un objet occupe une certaine surface et ne peut être représenté par un point unique.

A partir de cette modélisation, nous avons alors réalisé un algorithme dont le but est de définir une trajectoire de contournement d'obstacles. Cette procédure de contournement est composée d'une suite de points clés (ou cibles), vers lesquels le robot se dirige de manière successive.

Pour éviter un obstacle, le robot ou l'objet mobile procède en trois temps :

- repositionnement du robot,
- contournement de l'obstacle,
- direction vers un point de sûreté.

VI.3.5.1 Repositionnement du robot

Si le robot est en contact ou très proche du champ d'occupation de l'obstacle, nous le faisons reculer dans la direction opposée à la droite formée par le centre du robot et le centre de l'obstacle. Cette manipulation a pour but de positionner le robot dans un lieu où un mouvement est possible pour lui (Fig. VI.10).

VI.3.5.2 Contournement de l'obstacle

Une fois qu'un mouvement est possible, le robot cherche à contourner l'obstacle. Pour cela, il se dirige en arc de cercle jusqu'au point B , puis il suit la direction de la droite (BC) jusqu'à atteindre le point C . De cette façon, le robot sera placé de façon optimale pour se rediriger vers sa cible. Nous définissons le centre de l'obstacle $(X_obstacle, Y_obstacle)$, le centre de la cible (X_cible, Y_cible) et la distance entre le robot et l'obstacle $d_obstacle$. La position du point que l'on recherche B peut être calculé en remarquant que ce point est l'intersection du cercle de centre $(X_obstacle, Y_obstacle)$ et de rayon $d_obstacle$

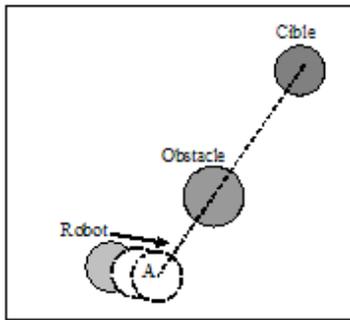


FIGURE VI.10 : Déplacement vers le point A

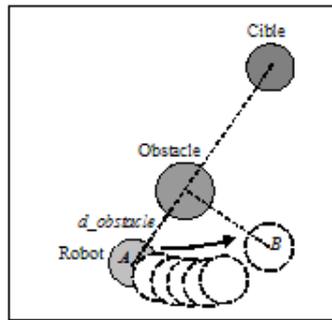


FIGURE VI.11 : Contournement de l'obstacle

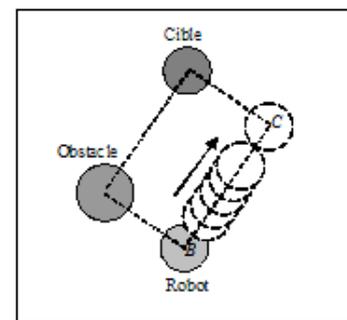


FIGURE VI.12 : Déplacement vers le point C

avec la perpendiculaire à la droite formée par les points $(X_obstacle, Y_obstacle)$ et (X_cible, Y_cible) (Fig. VI.11).

VI.3.5.3 Direction vers le point de sûreté

Malgré le repositionnement précédent du robot, il peut encore rencontrer l'obstacle en se dirigeant vers sa cible. En effet, il pourrait toucher le coin inférieur gauche de l'obstacle. Afin d'éviter ce problème, un point de sûreté C est fixé sur la perpendiculaire à la droite formée par le centre de l'obstacle $(X_obstacle, Y_obstacle)$ et celui de la cible (X_cible, Y_cible) passant par le centre de la cible (Fig. VI.12).

Maintenant que nous avons expliqué le comportement du robot, voyons maintenant comment s'effectue la communication entre les robots. Ce protocole est réalisé par le module de communication (Fig. I.16).

VI.3.6 Protocole de communication entre les robots

La communication est une notion sensible dans les systèmes multi-agents. Celle-ci peut avoir lieu selon plusieurs modes. La mise en place d'un système de messages a pour objectif de réaliser une communication efficace entre les différents robots du système. Nous décrivons deux différents modes de communication :

- *La communication asynchrone* consiste à envoyer un message à un destinataire avec l'intention de l'informer, de lui demander un service, etc. L'instance qui envoie un message n'attend pas une réponse à cet envoi pour poursuivre l'exécution de son

comportement. En effet, elle n'a aucune certitude sur la manière dont va réagir le destinataire.

- *La communication par diffusion* consiste à émettre des événements sans se soucier des destinataires. Ainsi, les instances qui sont sensibles à certains types d'événements réagissent quand ceux-ci sont émis. Ce type de communication est la métaphore de la radio-diffusion ; un émetteur radio diffuse des programmes et seules les personnes à l'écoute de leur récepteur sont sensibles aux informations émises.

Dans notre application, nous avons choisi le protocole de communication suivant :

Dans le cas de deux robots,

- Un robot émetteur (qui détecte l'objet) envoie un message d'aide à l'autre robot.
- Un robot récepteur (qui est sensible au message) se déplace vers l'objet.

En cas de plusieurs robots,

- Un robot émetteur envoie un message d'aide aux autres robots,
- Les robots récepteurs (qui sont sensibles au message) se déplacent vers l'objet. Le premier robot qui arrive à l'autre extrémité de l'objet signale aux autres robots d'abandonner leur mission.

VI.3.7 Protocole de négociation entre les robots

Lorsqu'un robot rencontre un obstacle, naturellement, il va tenter de l'éviter. Un système de négociation entre deux robots à été mis en place. Ce système permet de limiter des actions d'évitement d'obstacles par les robots. Afin de faciliter la compréhension du problème, nous définissons deux termes :

- *robot négociateur* : robot qui lance la négociation,
- *robot négocié* : robot avec qui le premier robot va négocier.

Après avoir détecté un obstacle via une détection locale, le robot "négociateur" regarde si cet obstacle est oui ou non un objet mobile. Si c'est le cas, il rentrera alors en négociation avec celui-ci et attendra un message d'acceptation ou de refus de la part du robot "négocié". L'issue de la négociation est définie par la comparaison d'une valeur numérique représentant la priorité de chaque robot. Cette priorité est définie pour chaque robot suivant l'action qu'il est en train d'effectuer. Un robot inactif a une priorité nulle, tandis qu'un robot actif possède une priorité maximale. En cas de conflit de priorité, autrement

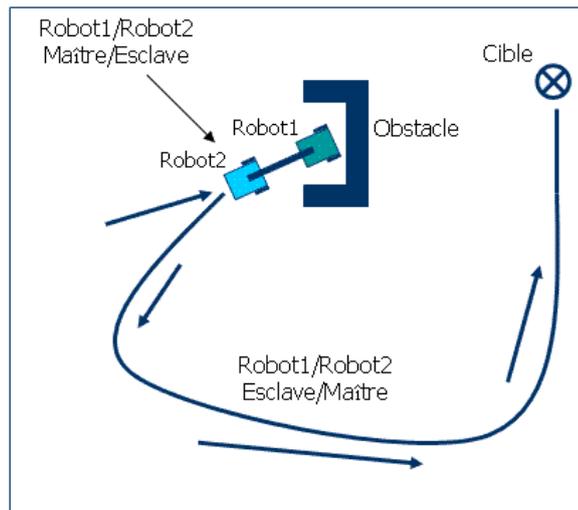


FIGURE VI.13 : Basculement de rôles dans le cas de blocage

dit, dans le cas où les priorités de chaque robot sont égales, c'est le robot initiateur de la négociation qui remporte le droit de passage.

VI.3.8 Définition des rôles

La tâche de transportation par plusieurs robots suppose une mise en place de rôles particuliers. Un rôle définit l'objectif de chaque robot et donc la nature des informations correspondantes. Les informations de la tâche peuvent être décomposées sur deux niveaux : l'information globale correspondant seulement à l'objectif à atteindre et des informations locales correspondant à la perception de l'environnement par les capteurs. La figure (VI.13) montre le système proposé. Il se compose de deux robots, dont l'un agit comme le maître tandis que l'autre est le suiveur. Il est supposé que les deux robots portent un objet entre eux en utilisant leurs bras.

Le robot maître dirige le groupe (objet et robot suiveur) vers le but. Il peut trouver sa propre voie. Il doit tenir compte des contraintes dues à l'environnement. Le robot suiveur suit le robot maître, il peut aussi éviter les obstacles. Le robot suiveur doit satisfaire le robot maître et la tâche de transportation.

La tâche de motivation assume le contrôle d'exécution pour des modifications externes. La réaction peut être faite pour des situations de blocage. Un obstacle, par exemple, peut perturber l'exécution du robot. Si c'est nécessaire, les rôles peuvent être inversés entre

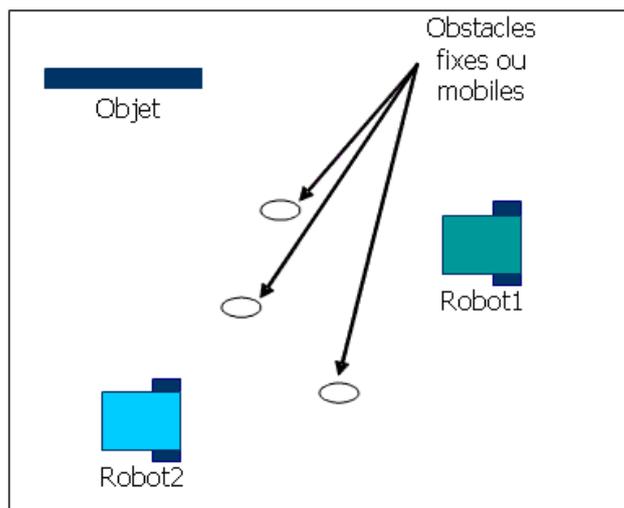


FIGURE VI.14 : Schéma de l'environnement de simulation

les robots. La figure (VI.13) montre une situation de conflit de la navigation. Une telle situation doit être réglée en changeant les rôles. Lorsque robot1, comme maître, arrive dans une situation de blocage. Le robot1 est bloqué, une tâche de basculement des rôles permet d'en sortir de cette situation et à reprendre la mission. Robot1 devient suiveur et robot2 maître.

Maintenant que nous avons défini les outils utilisés par chaque agent pour réaliser sa tâche, voyons comment se fait la mise en œuvre.

VI.4 Résultats de simulation

Les fondements de cette simulation reposent en premier lieu sur la création de l'environnement de simulation en 3D. Il convient, ensuite, de définir les différents comportements de chaque robot du système. La première étape concerne alors la détection des obstacles présents dans l'environnement qui peuvent être de deux types (Fig. VI.14) :

- des obstacles fixes (meubles, etc),
- des obstacles mobiles (des êtres humains, d'autres robots, etc).

Pour réaliser ce travail, le choix des outils de développement s'est porté sur oRis [120]. C'est un simulateur multi-agents gratuit fonctionnant sous le système Linux et permettant l'exécution de simulations diverses. En effet, cet outil utilise un langage orienté objet qui ressemble fortement au langage C++. oRis est à la fois un langage permettant de

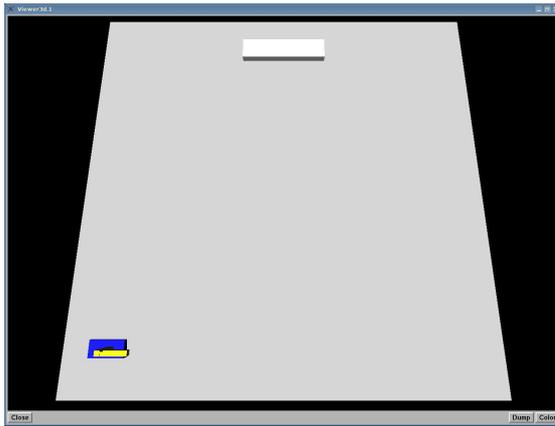


FIGURE VI.15 : Robot dans la position initiale

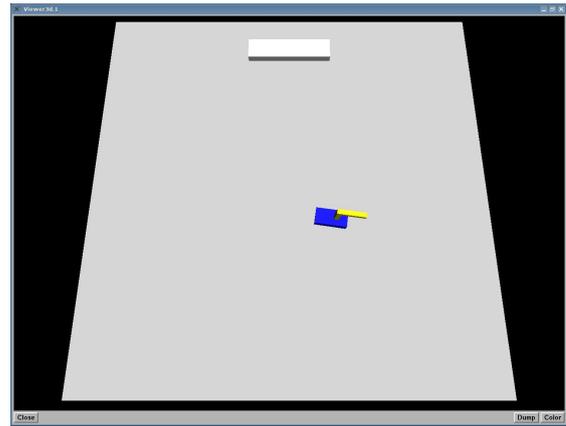


FIGURE VI.16 : Recherche aléatoire de l'objet

décrire le comportement d'objets actifs et un environnement de développement pour ce langage. L'objectif de cet atelier est de faciliter le prototypage dynamique d'applications de simulation, en permettant la modification des comportements des objets pendant leur exécution afin d'en visualiser immédiatement les conséquences. Le fonctionnement général d'oRis est présenté plus en détail dans la partie annexe B.

Nous présentons par la suite quelques résultats de simulation du système multi-agents.

Les figures (VI.15) à (VI.19) illustrent la stratégie de déplacement. Le robot se déplace aléatoirement dans l'environnement ; quand il détecte l'objet, il va vers l'extrémité la plus proche.

Dans le cas de deux robots, les résultats de simulation sont présentés par les figures (VI.20) à (VI.24).

- Fig. VI.20 : Deux robots dans la position initiale.
- Fig. VI.21 : Recherche aléatoire de l'objet.
- Fig. VI.22 : Un robot détecte l'objet. L'autre robot reste dans la recherche aléatoire. Le premier robot se déplace vers l'extrémité la plus proche, puis envoie un message "Je suis arrivé à l'extrémité la plus proche. Aidez-moi".
- Fig. VI.23 : Le deuxième robot reçoit le message d'aide envoyé par le premier robot et envoie un message "j'arrive", et il va à l'autre extrémité de l'objet. Quand il arrive, il envoie un message "Je suis arrivé à l'autre extrémité".
- Fig. VI.24 : Elle illustre les messages envoyés par les deux robots.

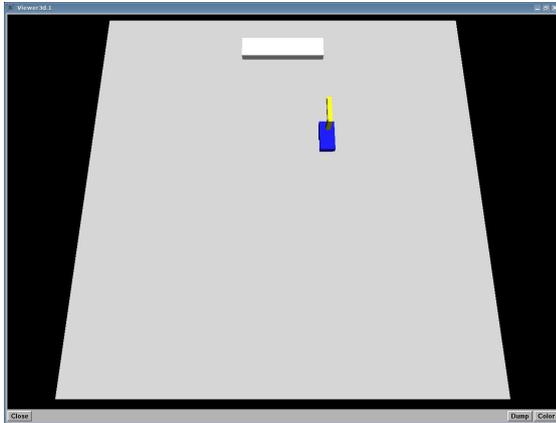


FIGURE VI.17 : Détection de l'objet

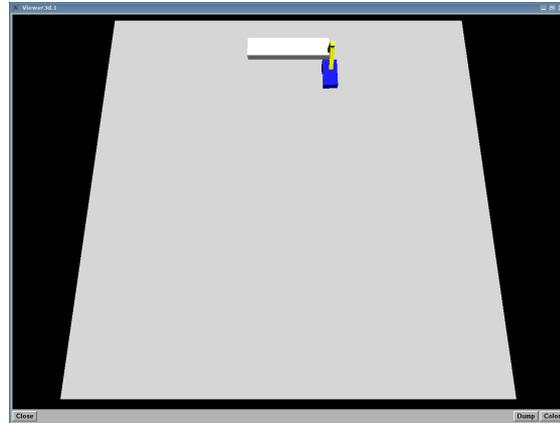


FIGURE VI.18 : Déplacement vers l'extrémité la plus proche

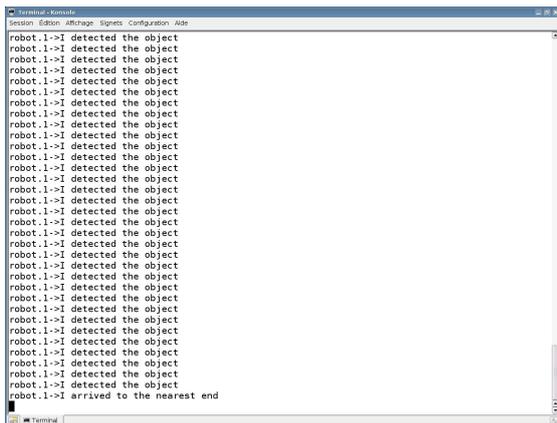


FIGURE VI.19 : Le message envoyé par le robot

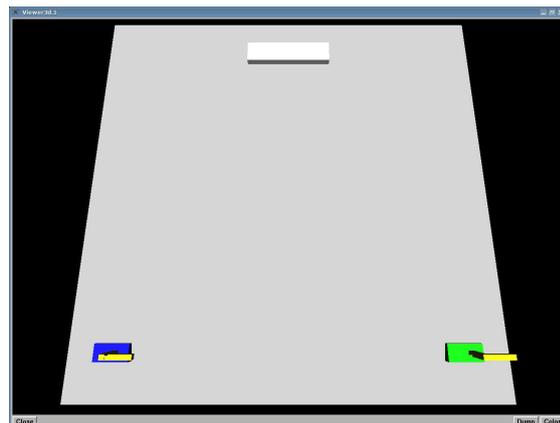


FIGURE VI.20 : Deux robots dans la position initiale

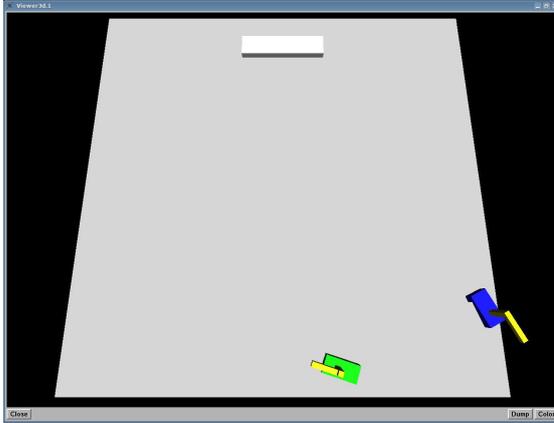


FIGURE VI.21 : Recherche aléatoire de l'objet

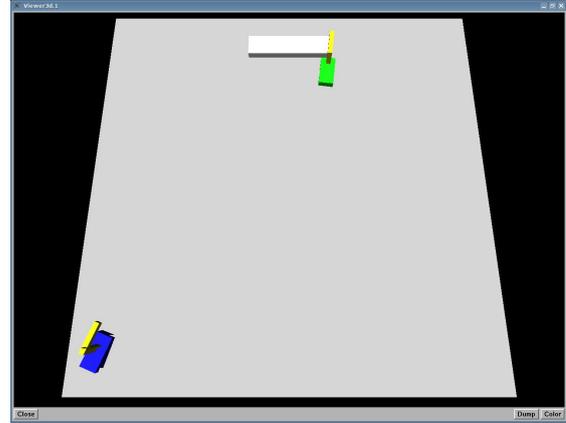


FIGURE VI.22 : Détection de l'objet et déplacement vers l'extrémité la plus proche

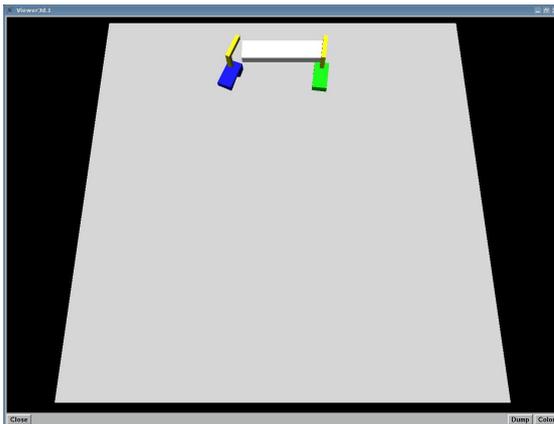


FIGURE VI.23 : Déplacement vers l'autre extrémité

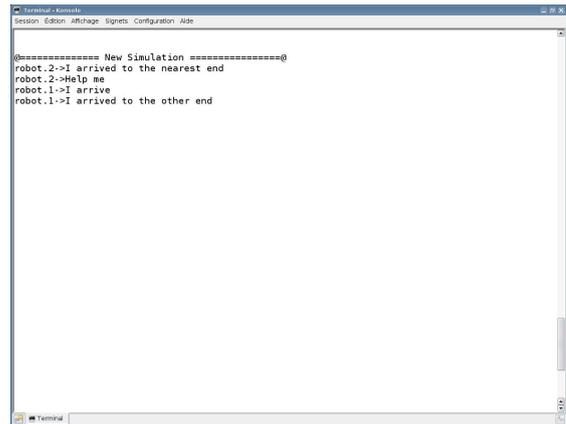


FIGURE VI.24 : Les messages envoyés par les deux robots

Les figures (VI.25) à (VI.29) montrent les résultats de la simulation dans le cas de plusieurs robots.

- Fig. VI.25 : Robots dans la position initiale.
- Fig. VI.26 : Recherche aléatoire de l'objet.
- Fig. VI.27 : Un robot détecte l'objet. Les autres robots restent dans la recherche aléatoire. Le premier robot se déplace vers l'extrémité la plus proche, puis envoie un message "Je suis arrivé à l'extrémité la plus proche. Aidez-moi". Les autres robots commencent à se déplacer vers l'autre extrémité de l'objet après avoir reçu le message d'aide envoyé par le premier robot.
- Fig. VI.28 : Un robot arrive à l'extrémité vide, il envoie un message "Je suis arrivé à l'autre extrémité. Je suis le premier arrivé" pour que les autres robots abandonnent cette tâche.
- Fig. VI.29 : Elle illustre les messages envoyés par les robots.

Les figures (VI.30) à (VI.36) montrent les résultats de simulation dans le cas de plusieurs robots et deux objets.

- Fig. VI.30 : Robots dans la position initiale.
- Fig. VI.31 : Recherche aléatoire de l'objet.
- Fig. VI.32 : Un robot détecte le premier objet. Les autres robots restent dans la recherche aléatoire. Le premier robot se déplace vers l'extrémité la plus proche du premier objet, puis envoie un message "Je suis arrivé à l'extrémité la plus proche. Aidez-moi". Les autres robots commencent à se déplacer vers l'autre extrémité du premier objet après avoir reçu le message d'aide envoyé par le premier robot.
- Fig. VI.33 : Un robot arrive à l'extrémité vide du premier objet, il envoie un message "Je suis arrivé à l'autre extrémité. Je suis le premier arrivé" pour que les autres robots abandonnent cette tâche et restent dans la recherche aléatoire du deuxième objet.
- Fig. VI.34 et Fig. VI.35 : montrent la même procédure pour le deuxième objet.
- Fig. VI.36 : Elle illustre les messages envoyés par les robots.

Les figures (VI.37) à (VI.40) montrent l'interaction entre le robot et son environnement en cas de présence d'un obstacle. Fig. VI.38 et Fig. VI.39 montrent comment le robot réussit à éviter l'obstacle, puis son déplacement vers l'objectif lorsque le chemin est libre

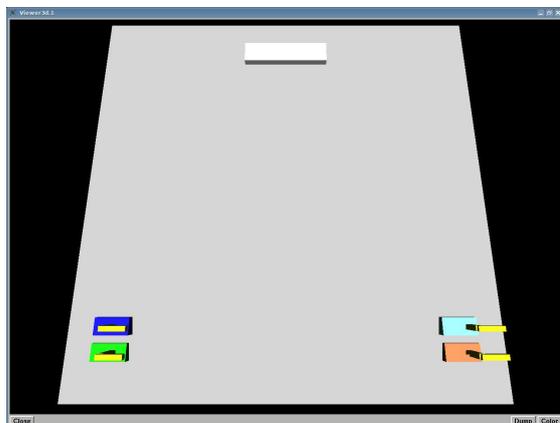


FIGURE VI.25 : Robots dans la position initiale

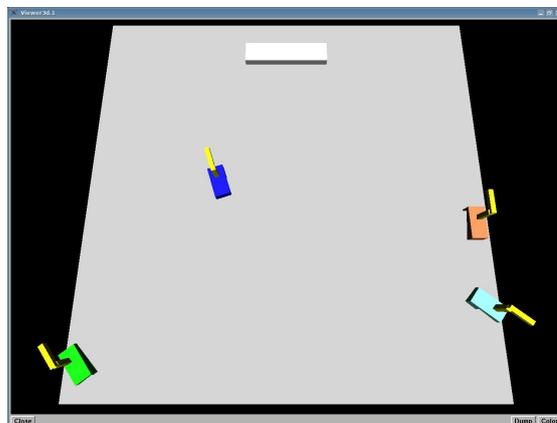


FIGURE VI.26 : Recherche aléatoire de l'objet

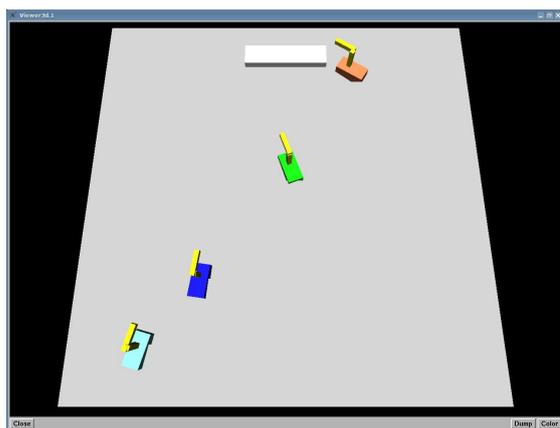


FIGURE VI.27 : Détection de l'objet et déplacement vers l'extrémité la plus proche

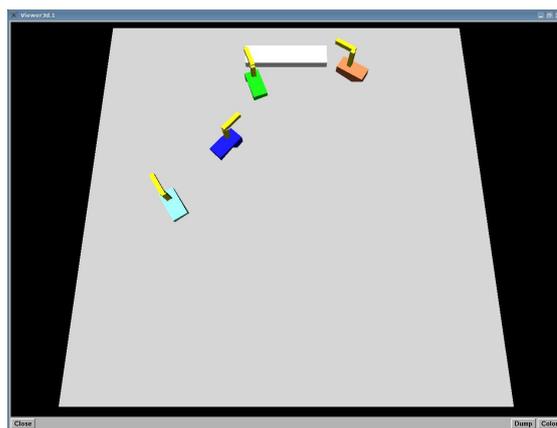


FIGURE VI.28 : Déplacement vers l'autre extrémité

```

Terminal - Konsole
Session Edition Affichage Signets Configuration Aide

@===== New Simulation =====@
robot.4->I arrived to the nearest end
robot.4->Help me
robot.1->I arrive
robot.2->I arrive
robot.3->I arrive
robot.2->I arrived to the other end
robot.2->I am the first arrived
robot.1->ok
robot.3->ok

```

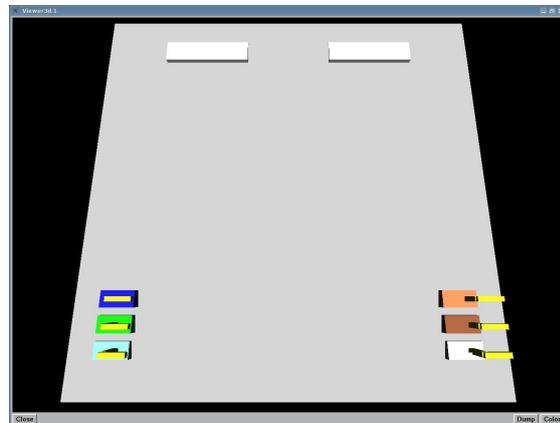


FIGURE VI.29 : Les messages envoyés par les robots

FIGURE VI.30 : Robots dans la position initiale

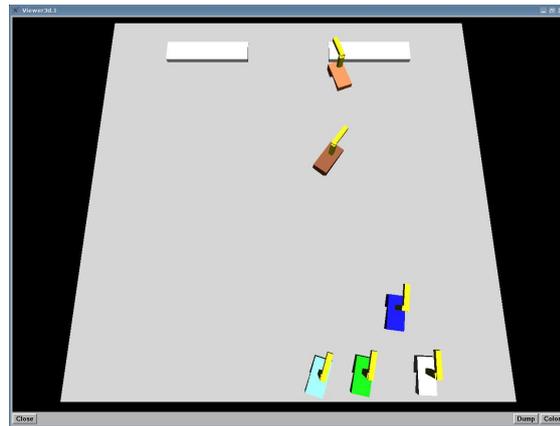
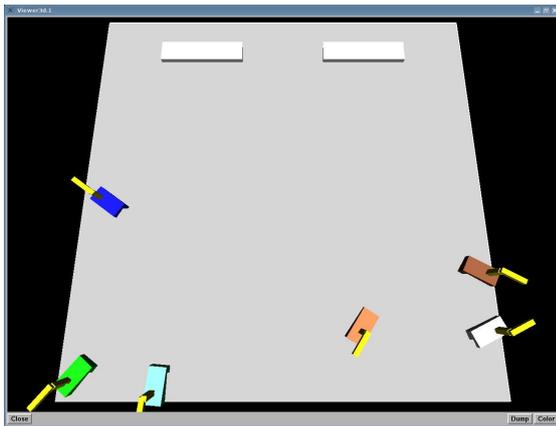


FIGURE VI.31 : Recherche aléatoire de l'objet

FIGURE VI.32 : Détection du premier objet et déplacement vers l'extrémité la plus proche

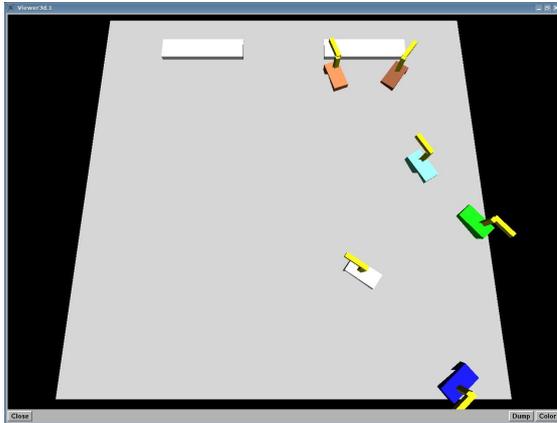


FIGURE VI.33 : Déplacement vers l'autre extrémité du premier objet

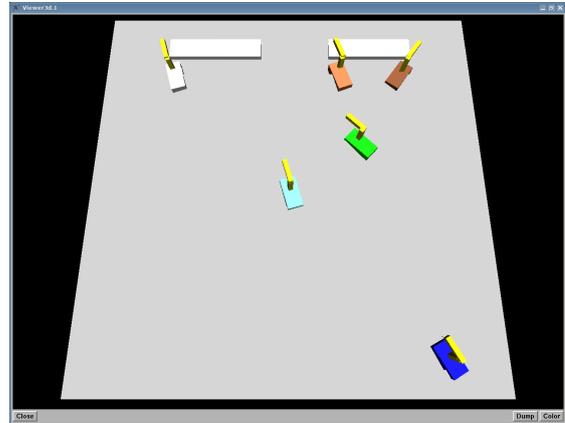


FIGURE VI.34 : Détection du deuxième objet et déplacement vers l'extrémité la plus proche

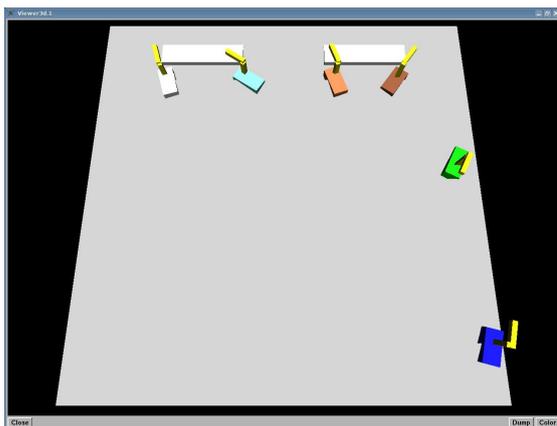


FIGURE VI.35 : Déplacement vers l'autre extrémité du deuxième objet

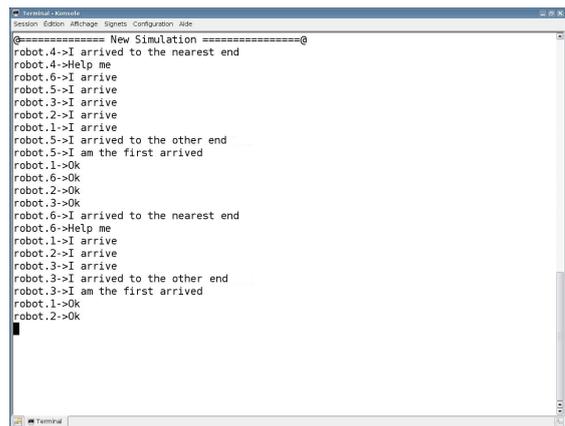


FIGURE VI.36 : Les messages envoyés par les robots

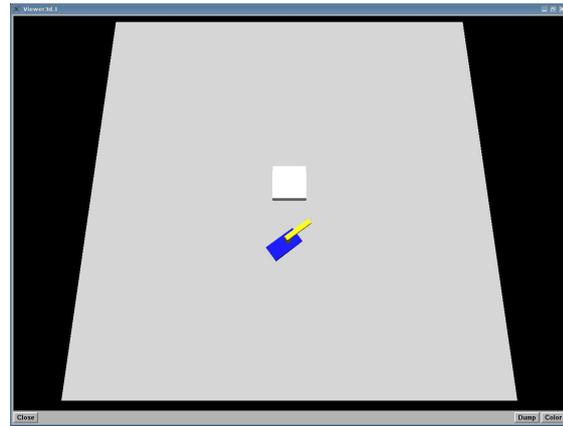
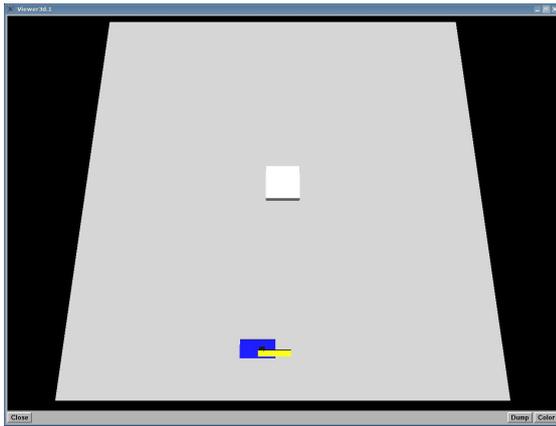


FIGURE VI.37 : Robot dans la position ini- **FIGURE VI.38** : Evitement de l'obstacle
tiale (a)

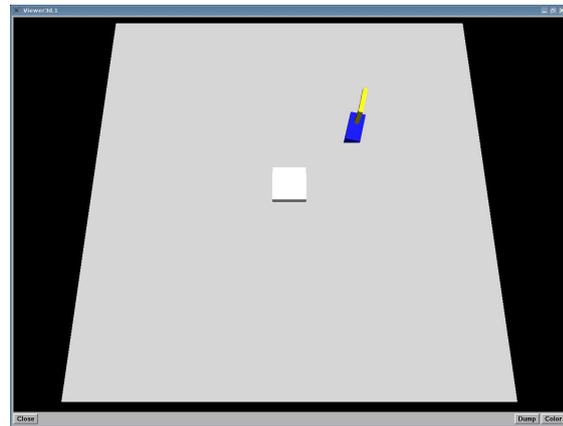
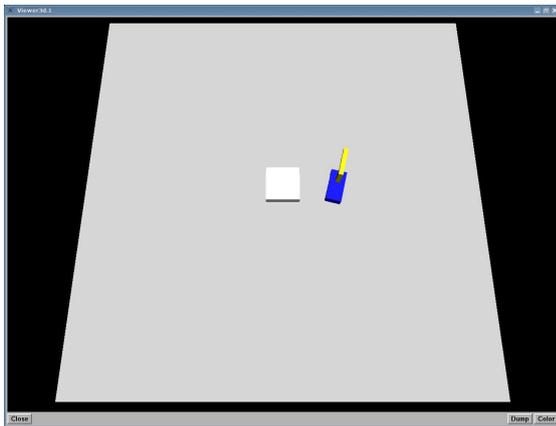


FIGURE VI.39 : Evitement de l'obstacle **FIGURE VI.40** : Déplacement vers le but
(b)

(Fig. VI.40).

Dans la situation de plusieurs obstacles, le robot trouve un moyen de trouver un chemin libre vers son objectif final (Figs VI.41 à VI.45).

Lorsque l'objet est actif, les deux robots vont coopèrent pour le transporter, (Fig. VI.46 et Fig. VI.47). En présence d'autres robots, un système de négociation est utilisé, (Figs. VI.48 à VI.53).

- Fig. VI.48 : Objet dans la position initiale.
- Fig. VI.49 : Les messages envoyés par les robots.
- Fig. VI.50 et Fig. VI.51 : Négociation entre les robots. Le robot inactif envoie un

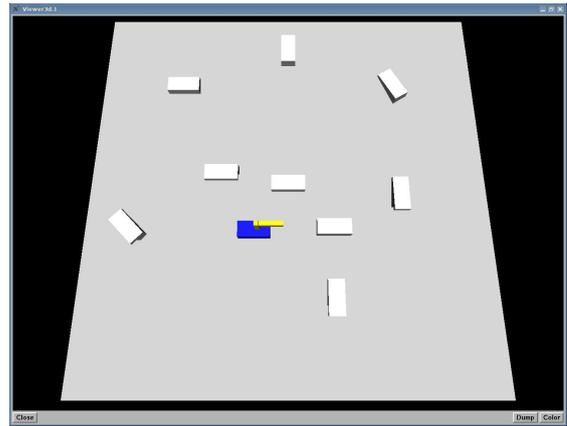
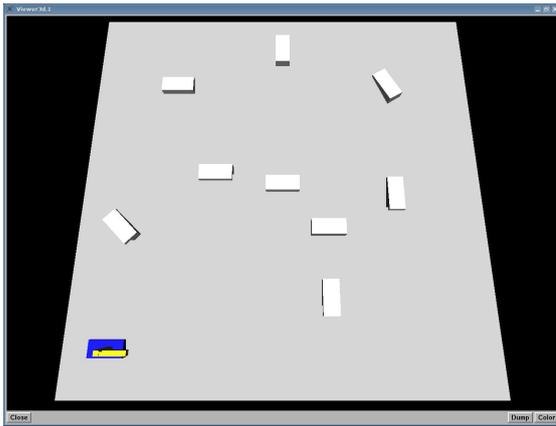


FIGURE VI.41 : Robot dans la position initiale **FIGURE VI.42** : Evitement des obstacles (a)

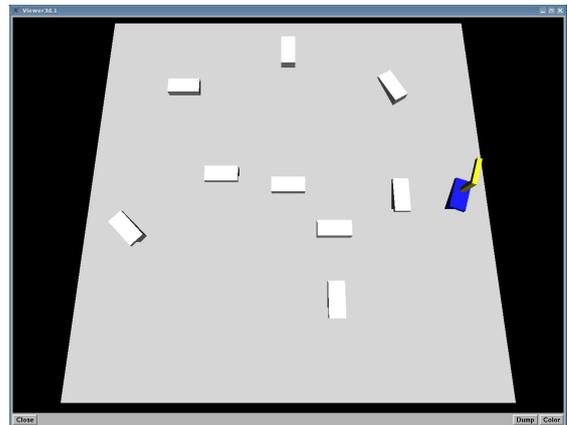
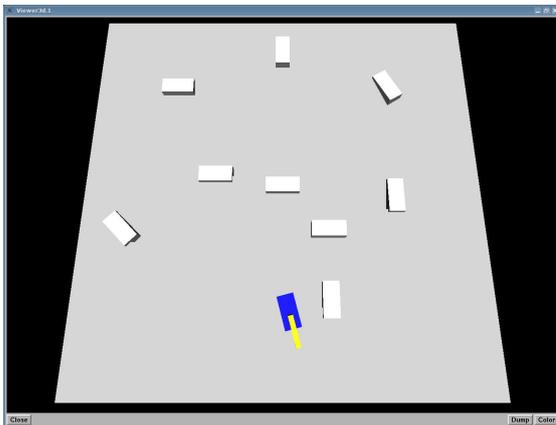


FIGURE VI.43 : Evitement des obstacles **FIGURE VI.44** : Evitement des obstacles (b) (c)

message "Je suis un robot inactif" et laisse les deux robots actifs transporter l'objet vers la cible.

- Fig. VI.52 : Le message envoyé par le robot inactif.
- Fig. VI.53 : Transport de l'objet.

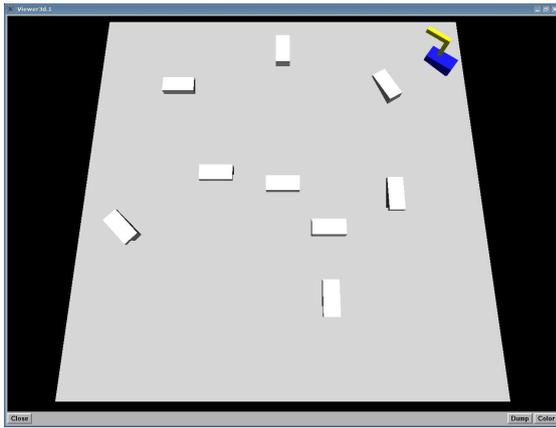


FIGURE VI.45 : Déplacement vers le but

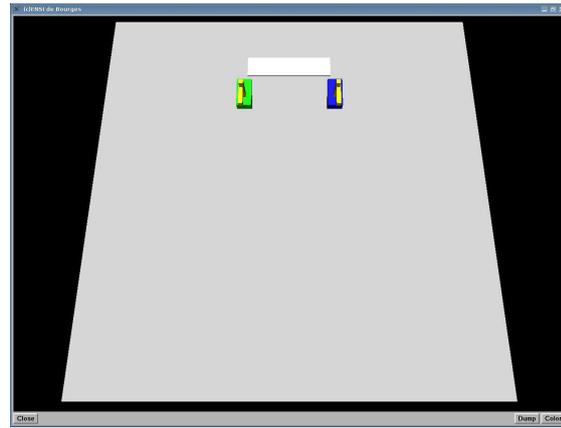


FIGURE VI.46 : Objet dans la position initiale

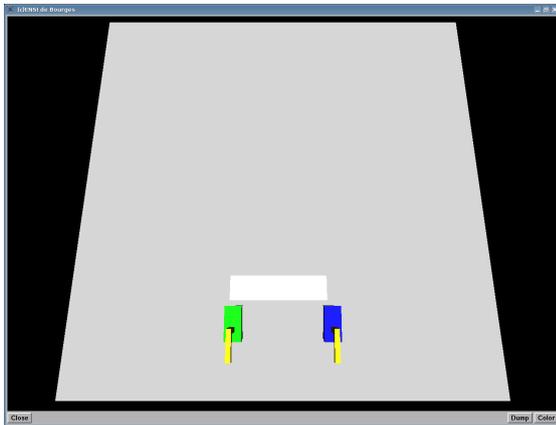


FIGURE VI.47 : Transport de l'objet

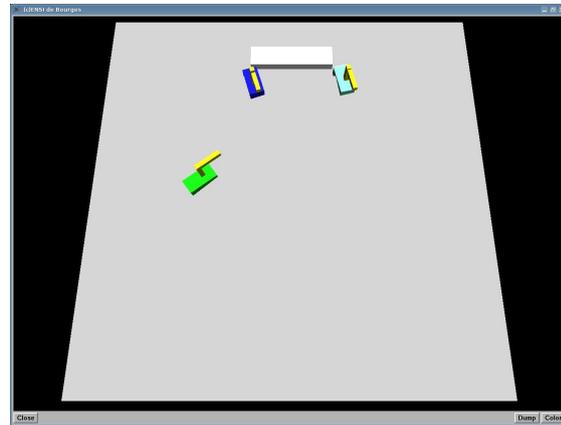


FIGURE VI.48 : Objet dans la position initiale

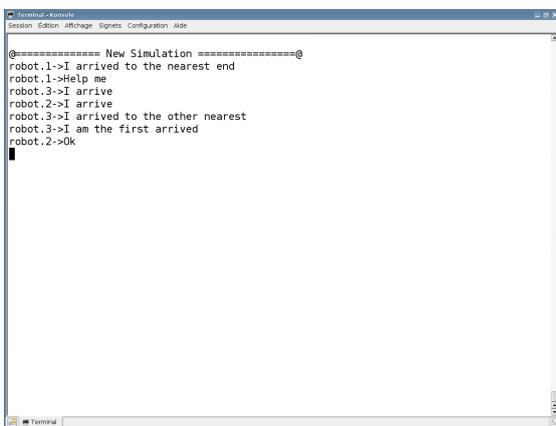


FIGURE VI.49 : Les messages envoyés par les robots

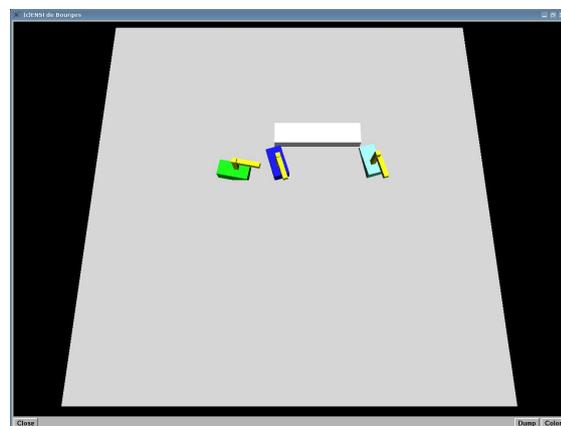


FIGURE VI.50 : Négociation entre les robots (a)

Chapitre VII

Conclusion générale

Le sujet de la thèse a pour objectif d'arriver à contrôler un nombre de robots dans l'exécution de tâches complexes nécessitant coopération et coordination. En effet, le sujet propose une partie de l'approche multi-agents où l'exécution de la tâche dans un environnement complexe repose sur l'intelligence distribuée et l'apport de chaque agent dans la réussite finale. Ainsi, le problème se réduit à de simples comportements qui émergent des nombreuses interactions entre les agents et leur environnement. Nous avons proposé une structure d'un manipulateur mobile qui représente un agent. Etant donné que le manipulateur mobile est composé d'une plate-forme mobile sur laquelle repose un bras manipulateur. Le système constitué d'un ensemble de manipulateurs mobiles, chacun peut communiquer avec les autres afin de réaliser la tâche demandée, cette dernière réside dans la coopération entre deux manipulateurs mobiles ou plus dans une manipulation continue d'un objet. Le choix de la stratégie doit prendre en compte toutes les dynamiques celle de l'environnement imprévisible ainsi que celle des autres robots. Pour mener à bien ce travail, nous avons proposé une architecture de commande pour la coopération des manipulateurs mobiles et nous avons montré la faisabilité de cette architecture ainsi que les lois de commandes associées.

Pour ce faire, nous avons choisi de hiérarchiser les niveaux de la commande en considérant deux niveaux :

- Un niveau haut qui concerne l'intelligence artificielle : la communication entre les robots, la négociation entre les robots, etc.
- Un niveau bas, où nous nous sommes intéressés à la modélisation du manipulateur

mobile, la commande cinématique et dynamique, l'évitement d'obstacles, etc.

La mobilité intrinsèque d'un robot est apportée par sa structure mécanique, qui donnera au système son degré d'autonomie de déplacement. Ceci nous a conduit à choisir la structure omnidirectionnelle *ROMNI* développée au laboratoire *PRISME* de Bourges. Ce robot présente une autonomie maximale de déplacement. En effet, ce robot mobile qui présente l'avantage d'être holonôme permet un déplacement plus souple dans un environnement contraignant.

La commande des robots mobiles reste à ce jour un thème de recherche important au sein de la communauté roboticienne. Les travaux théoriques dans le domaine sont prépondérants et nous proposent différentes solutions pour résoudre les problèmes de déplacement. La commande des manipulateurs mobiles reste aussi jusqu'à présent un thème de recherche important. Bien que d'innombrables travaux dans ce domaine sont apparus, des solutions globales pour de tels systèmes demeurent toujours posées en raison de leurs redondance et de la difficulté de modéliser les différentes tâches en tenant compte des aspects de la stabilité du système, de la vitesse d'exécution, de la coopération et de la coordination.

Dans cette thèse nous nous sommes intéressés à développer des algorithmes de contrôle cinématique et dynamique d'un manipulateur mobile omnidirectionnel formé de la plate-forme omnidirectionnelle *ROMNI* et d'un bras manipulateur plan à deux liaisons. Nous avons résolu autrement le problème de contrôle, qui nécessite la modélisation du système global, en lui donnant plus de flexibilité dans la gestion des tâches dans des milieux contraignants.

Dans la première partie, nous avons donné le modèle dynamique de la plate-forme omnidirectionnelle *ROMNI*. Le modèle dynamique du manipulateur mobile omnidirectionnel formé de la plate-forme omnidirectionnelle *ROMNI* et d'un bras manipulateur plan à deux liaisons a été développé en employant le formalisme explicite de Lagrange. La forme du modèle obtenu permet une application directe à la commande du manipulateur mobile.

L'approche introduite passe par la définition de lois de commande opérationnelle basées sur les modèles spécifiques aux manipulateurs mobiles à roues. Ces lois de commande ont fait l'objet de la seconde partie de cette thèse. La proposition d'une implémentation d'une commande cinématique basée sur les tâches additionnelles avec l'usage de l'impédance est

donnée dans ce chapitre. Deux exemples ont servi à illustrer ces développements dans le cas du manipulateur mobile omnidirectionnel avec un bras manipulateur plan à deux liaisons et le cas du manipulateur mobile omnidirectionnel avec un bras manipulateur à trois liaisons rotoïdes.

Dans la troisième partie, nous avons proposé une structure de commande dynamique permettant d'intégrer la théorie de la commande par retour d'état linéarisant. Cette commande nous a permis de valider le modèle dynamique du robot mobile *ROMNI*. Cette commande est en effet très développée dans la littérature, ce qui facilite son utilisation. Nous avons appliqué aussi cette commande avec une boucle externe due à l'impédance sur le manipulateur mobile omnidirectionnel. L'intérêt d'une telle approche est de permettre d'unifier les interactions robot-robot et robot-environnement. Diverses simulations ont été proposées afin de montrer le bien fondé de cette commande.

Dans la quatrième partie, nous avons traité le problème de l'adaptation des paramètres de l'impédance en utilisant l'approche floue. Nous avons présenté notre approche qui consiste en la conception d'un contrôleur en se basant sur la commande linéarisante ainsi que sur l'approche du contrôle flou. Dans la première couche de ce contrôleur figure la commande par retour d'état linéarisant. Nous avons ajouté la stratégie du contrôle dynamique augmenté d'une boucle externe de contrôle en impédance pour l'évitement d'obstacles. La deuxième couche présente un contrôleur flou dont la base des règles est construite dans le but d'adapter les paramètres de l'impédance. Cette adaptation permet de modifier le comportement du robot lorsqu'il évolue en libre ou sous contraintes lors du transport d'objets avec un autre robot. Dans ce chapitre aussi, nous avons présenté l'identification des paramètres dynamiques du manipulateur mobile omnidirectionnel en utilisant l'approche des moindres carrés récursifs.

La dernière partie concerne les systèmes multi-agents. Nous avons présenté les différentes architectures des systèmes multi-agents. Nous avons détaillé le travail d'une simulation de robots à l'aide d'un langage permettant d'effectuer des simulations multi-agents : *oRis*. Cette simulation gère une interaction entre les robots via un système avancé d'automates qui sont équipés d'un système de communication. Nous avons implémenté une gestion des obstacles nous permettant successivement de détecter puis d'éviter un objet étranger. Enfin, nous avons mis en œuvre une étape de négociation entre les robots afin

de limiter les actions d'évitement d'obstacles par les robots. Ce chapitre se termine par plusieurs résultats de simulations pour montrer la faisabilité des choix proposés.

Les résultats de simulations obtenus nous ont mis en lumière certains points de cette thèse qui restent à améliorer. Nous pourrions envisager en premier l'implémentation pratique des concepts de commande développés. En particulier, l'organisation des agents, la distribution des impédances et la mise en place du contrôle d'une flotte hétérogène de robots mobiles. En second nous pouvons envisager le développement théorique sur les techniques de commande en particulier l'amélioration de linéarisation par l'utilisation d'un contrôleur adaptatif ou à réseau de neurones pour l'identification paramétrique cela pour permettre de palier la méconnaissance de certains paramètres dynamiques des robots mobiles ou manipulateurs dans le cas de l'utilisation des manipulateurs mobiles. En effet, l'utilisation d'une commande à gains variables permettra de s'assurer de la compensation de la dynamique non linéaire avant d'envisager la boucle externe de la commande en impédance.

Annexes

Annexe A

Outils mathématiques

A.1 Inversion matricielle

Soit A une matrice de dimension $m \times n$.

A.1.1 Matrices inverses à gauche et à droite

La matrice G , de dimension $n \times m$, est une inverse à gauche de A si et seulement si $GA = I_n$. De même la matrice D , de dimension $n \times m$, est une inverse à droite de A si et seulement si $DA = I_m$.

A.1.2 Matrice inverse généralisée et matrice pseudo-inverse

A^\sharp , de dimension $n \times m$, est une inverse généralisée de la matrice A si et seulement si :

$$AA^\sharp A = A \tag{A.1}$$

A^+ , de dimension $n \times m$, est la pseudo inverse (ou inverse de Moore Penrose) de la matrice A si et seulement si :

$$AA^+A = A \tag{A.2}$$

$$A^+AA^+ = A^+ \tag{A.3}$$

$$(AA^+)^T = AA^+ \tag{A.4}$$

$$(A^+A)^T = A^+A \tag{A.5}$$

D'après (A.2), la pseudo-inverse est une inverse généralisée particulière de A . Une inverse généralisée de la matrice A est utilisée quand l'inverse A^{-1} n'existe pas, lorsque A est singulière ou rectangulaire.

A.1.2.1 Calcul de la matrice pseudo-inverse

- Soit la matrice A de dimension $(m \times n)$, où $(m \geq n)$, et de rang $r = n$. La matrice $A^T A$ est définie positive, donc inversible, et $A^+ = (A^T A)^{-1} A^T$ est une inverse à gauche de A .
- Soit la matrice A de dimension $(m \times n)$, où $(m \leq n)$, et de rang $r = m$. La matrice AA^T est définie positive, donc inversible, et $A^+ = A^T (AA^T)^{-1}$ est une inverse à droite de A .
- Soit la matrice A de dimension $(m \times n)$ de rang déficient, $r < \min(m, n)$. Supposons que, par des permutations préalables de lignes et de colonnes, A s'écrive :

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

où A_{11} est d'ordre r et de rang r . $A_1 = [A_{11}^T \ A_{21}^T]^T$ de dimension $(m \times r)$ et $A_2 = [A_{12} \ A_{22}]$ de dimension $(r \times n)$, la pseudo-inverse s'écrit :

$$A^+ = A_2^T (A_1^T A A_2^T)^{-1} A_1^T$$

A.2 Résolution d'un système linéaire

Soit le système linéaire à résoudre :

$$Ax = y \tag{A.6}$$

avec A une matrice donnée de dimension $(m \times n)$ et de rang r , y un vecteur donné de dimension $(m \times 1)$. Le problème est de trouver le vecteur x de dimension $(n \times 1)$.

- Si la matrice A est carrée ($m = n$) et non singulière, alors la solution du système est unique. Elle est donnée par :

$$x = A^{-1}y \tag{A.7}$$

- Si la matrice A est rectangulaire ($m \neq n$), on peut résoudre le problème par minimisation de critère suivant :

$$\min_{x \in \mathcal{R}^n} \frac{1}{2} \|y - Ax\|^2 \tag{A.8}$$

- Si ce minimum est nul, alors x vérifie $Ax = y$ et constitue une solution exacte de cette équation.
- Si ce minimum n'est pas nul, alors $Ax \neq y$ et y est une solution approchée de l'équation $Ax = y$.

A.2.1 Cas où A est de rang maximal

- Si $m \geq n$ et $r = n$, alors la solution de l'équation (A.6) est :

$$x = (A^T A)^{-1} A^T y = A^+ y \quad (\text{A.9})$$

- Si $m \leq n$ et $r = m$, alors l'équation (A.6) admet une infinité de solutions exactes qui peuvent s'écrire :

$$\begin{aligned} x &= A^T (A A^T)^{-1} y + (I_n - A^T (A A^T)^{-1} A) z \\ &= A^+ y + (I_n - A^+ A) z \end{aligned} \quad (\text{A.10})$$

où z est un vecteur quelconque de \mathcal{R}^n . La solution de norme euclidienne minimale est :

$$x = A^T (A A^T)^{-1} y = A^+ y \quad (\text{A.11})$$

A.2.2 Cas où A est de rang déficient

Si la matrice A est de rang déficient, $r < \min(n, m)$. Le problème peut se ramener à la résolution de :

$$\min_{x \in \mathcal{R}^n} \frac{1}{2} \|y_1 - A_2 x\|^2 \quad (\text{A.12})$$

A.3 Décomposition en valeurs singulières

Soit A une matrice de dimension $(m \times n)$. On appelle décomposition en valeurs singulières (SVD, *Singular Value Decomposition*) de la matrice A , l'expression :

$$A = U_{m \times m} \Sigma_{m \times n} V_{n \times n}^T \quad (\text{A.13})$$

où U et V sont des matrices orthogonales de dimensions m et n respectivement et Σ est une matrice diagonale de dimension $m \times n$. Σ s'écrit :

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & \dots & 0 & 0 \\ 0 & \sigma_2 & \dots & 0 & 0 \\ \cdot & \cdot & \dots & \cdot & \cdot & 0 \\ \cdot & \cdot & \dots & \cdot & \cdot \\ \cdot & \cdot & \dots & \sigma_m & 0 \end{bmatrix} \quad (\text{A.14})$$

où les scalaires $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_m$ sont les valeurs singulières de A , classées par ordre décroissant.

$$\Sigma^+ = \begin{bmatrix} 1/\sigma_1 & 0 & \dots & \dots & 0 \\ 0 & 1/\sigma_2 & \dots & \dots & 0 \\ \cdot & \cdot & \dots & \dots & \cdot \\ \cdot & \cdot & \dots & \dots & \cdot \\ 0 & 0 & \dots & \dots & 1/\sigma_m \\ 0 & 0 & \dots & \dots & 0 \\ \cdot & \cdot & \dots & \dots & \cdot \\ 0 & 0 & \dots & \dots & 0 \end{bmatrix} \quad (\text{A.15})$$

Annexe B

Le fonctionnement interne d'oRis

L'environnement oRis propose à la fois un langage orienté agent permettant de décrire la structure et le fonctionnement d'un système quelconque, et un simulateur pour simuler et modifier dynamiquement un système modélisé [121, 122].

B.1 Langage interprété et orienté objet

B.1.1 Programmation orientée objet

Les concepts de la programmation orientée objet sont repris dans oRis. De la même manière qu'en C++, écrire un programme revient à définir des classes. Une classe est une structure permettant de regrouper les données et les traitements qui lui sont associés.

Les attributs représentent les données qui sont propres à chaque instance et les méthodes sont les services communs à toutes les instances.

Les principes les plus importants sont les suivants :

- une classe est déclarée par un bloc contenant les déclarations d'attributs et méthodes précédé de "class" et du nom que l'on souhaite donner à la classe (class Exemple ...);
- une classe peut hériter d'une ou plusieurs autres classes (class AB :A,B ...);
- la déclaration d'un attribut consiste à préciser son type et son nom (int compteur);
- la déclaration d'une méthode consiste à préciser son prototype (void f (int i));
- tous les attributs et méthodes sont publics, il est toujours possible d'accéder aux attributs et méthodes de n'importe quelle instance,

- les attributs et méthodes sont directement accessibles par leur nom (comme de simples variables et fonctions) dans le code des méthodes de la classe concernée,
- les attributs et méthodes sont accessibles en faisant précéder leur nom par une référence à l'instance et `->` dans tout code extérieur aux méthodes de la classe concernée (`UnAgent -> f(5);`),
- une méthode `"new()"` sans argument ni valeur de retour peut être définie pour jouer le rôle de constructeur. Si aucun constructeur n'est défini, l'instance est tout de même créée et ses attributs sont initialisés à leur valeur nulle,
- une méthode `"delete()"` sans argument ni valeur de retour peut être définie pour jouer le rôle de destructeur. Si aucun destructeur n'est défini, l'instance est tout de même détruite mais se pose alors le problème de fuite mémoire,
- le principe de méthodes virtuelles est appliqué sous oRis de la même manière qu'en C++. Contrairement au C++, aucun mot clé n'est nécessaire.

B.1.2 Programmation structurée

Le langage oRis permet la programmation structurée d'une manière très semblable au langage C. L'écriture d'un programme, en programmation structurée, consiste donc à définir des fonctions. Les points à observer lors de l'écriture d'un programme oRis sont relativement proches de ceux du langage C :

- la déclaration d'une fonction consiste à déclarer son prototype (nom, arguments, type de retour),
- la définition d'une fonction consiste à décrire complètement son code en plus de son prototype,
- les fonctions doivent être déclarées ou définies avant leur utilisation, la déclaration d'une variable consiste à préciser son type et son nom,
- les variables doivent être déclarées avant leur utilisation et ont une portée limitée à la fonction dans laquelle elles sont déclarées. Il n'y a donc pas de variables globales en oRis.

Contrairement au langage C++, la surcharge d'une fonction et l'utilisation du `"switch case"` ne sont pas disponibles.

B.1.3 Instructions de base

Le langage oRis reprend à la fois les concepts de programmation structurée, de programmation objet (la syntaxe est d'ailleurs très proche de celle du C++). De plus, des fonctionnalités orientées agents et des propriétés dynamiques sont présentes dans le package de base.

Le langage oRis est conçu comme un outil de prototypage interactif, c'est-à-dire que l'utilisateur peut modifier ou compléter une simulation en même temps que celle-ci s'exécute :

- surdéfinir des méthodes de classes,
- définir des nouvelles classes,
- ajouter des méthodes à des instances,
- surdéfinir des méthodes d'instances existantes,
- ajouter des attributs à des instances,
- exécuter des instructions quelconques.

B.1.4 Qu'est ce qu'un bloc "execute" ?

Le code principal du programme structuré est représenté par un bloc d'instructions intitulé "execute". Ce bloc est assez semblable à la fonction "main()" du langage C dans le sens où il représente le point d'entrée du programme. Il permet également d'initialiser l'application en créant les instances des agents. Ensuite, ce sont les instances qui font évoluer l'application et non un programme principal. Ainsi, la fin du bloc "execute" ne représente pas la fin du programme. Il s'agit là d'une grande différence avec la fonction principale "main()" des langages C/C++.

B.2 Langage permettant la simulation

B.2.1 Fonctionnement des threads : agent

oRis est un langage orienté objet et il permet de manipuler des instances d'objets autonomes que l'on qualifie d'agents. Un agent correspond au sein du langage oRis à la notion d'objet actif. Dans oRis, la distinction entre un objet ou un agent se fait grâce à

la présence de la méthode "main()" dans l'objet. oRis est un simulateur chargé d'animer tous les agents en interprétant le code source sans le compiler. C'est pour cette raison que l'on parle de simulation pour désigner une application oRis.

oRis se charge d'appeler automatiquement et perpétuellement la méthode "main()" de chaque agent. Celle-ci représente le point d'entrée du comportement de l'agent concerné.

B.2.2 Le monde multi-agents

Les systèmes multi-agents représentent des environnements dans lesquels les différents constituants peuvent être dotés d'un comportement autonome leur permettant d'interagir et d'évoluer parmi les autres constituants du système (qui représentent alors leur environnement).

Dans le langage oRis, un agent est un objet qui dispose d'une méthode particulière représentant le point d'entrée de son comportement. Ainsi, dès lors que l'agent est créé, il exécute son comportement de manière cyclique sans que l'on ait besoin d'invoquer explicitement ses méthodes. Il est alors possible de simuler le comportement d'agents évoluant simultanément dans un même environnement.

L'intérêt d'une telle architecture "multi-agents" réside dans la collaboration possible (directe ou non) entre les agents. Cependant, les différentes notions sur la communication entre eux sont détaillées dans les paragraphes suivants.

Les agents peuvent avoir besoin d'une certaine autonomie, c'est pourquoi, il est alors intéressant de spécifier des particularités de comportement au niveau d'une instance particulière et non pas de la classe toute entière.

B.3 Principe de vision : "view" et "viewFirst"

L'ensemble du fonctionnement de des fonctions "view" et "viewFirst" est résumé par le schéma (B.1).

Le triangle dans la figure (B.1) représente un objet quelconque de la scène. Ce dernier possède une orientation, dans notre cas, nous avons choisi de mettre l'angle "theta" à la valeur 0, ce qui signifie que les objets sont orientés de manière parallèle à l'axe des abscisses dans le repère global.

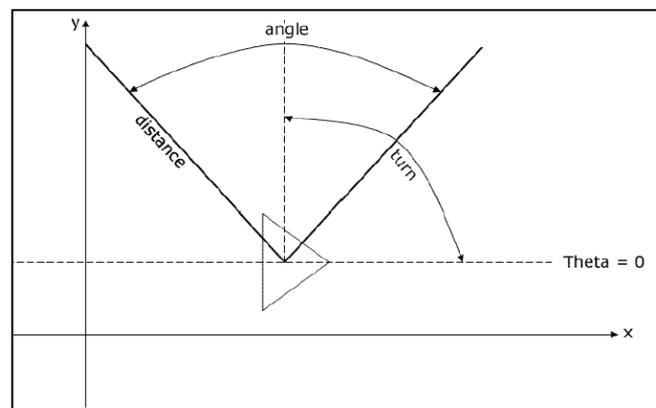


FIGURE B.1 : Fonctionnement du mécanisme de vision sous oRis

Ensuite, à l'aide des fonctions "view" et "viewFirst", il est possible de récupérer l'ensemble des objets présents dans un cône de vision donné. Pour cela, il suffit de passer aux fonctions précédentes les paramètres suivants :

- la distance,
- l'angle d'ouverture du cône,
- l'angle "turn".

Bibliographie

- [1] S. J. Russell and P. Norvig. Artificial intelligence. a modern approach. *Prentice-Hall International, Englewood Cliffs, New Jersey, USA*, 1995.
- [2] G. Campion, G. Bastin, and B. D'Andrea-Novel. Structural proprieties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 12(1) :47–62, February 1996.
- [3] F. G. Pin and S. M. Killough. A new family of omnidirectional and holonomic wheeled platforms for mobile robots. *IEEE Transactions on Robotics and Automation*, 10(4) :480–489, 1994.
- [4] U. M. Nassal, M. Damm, and T. C. Lueth. Mobile manipulation - a mobile platform supporting a manipulator system for an autonomous robot. *Fifth World Conference on Robotics Research, Cambridge, Massachusetts, USA*, September 1994.
- [5] T. C. Lueth, U. M. Nassal, and U. Rembold. Reliability and integrated capabilities of locomotion and manipulation for autonomous robot assembly. *Journal on Robotics and Autonomous Systems, Special issue : Research on Autonomous Mobile Systems in Germany*, 14(2-3) :185–198, May 1995.
- [6] G. Foulon, J.-Y. Fourquet, and M. Renaud. Planning point to point paths for nonholonomic mobile manipulators. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'98, Victoria, Canada*, 1 :374–379, October 1998.
- [7] D. P. Tsakiris, K. Kapellos, C. Samson, P. Rives, and J. J. Borrelly. Experiments in real-time vision-based point stabilization of a nonholonomic mobile manipulator. *Proceedings of the Fifth International Symposium on Experimental Robotics, ISER'97, Barcelona, Spain*, pages 463–474, June 1997.

-
- [8] A. Albers, S. Brudniok, J. Ottnad, C. Sauter, and K. Sedchaicharn. Upper body of a new humanoid robot - the design of armar 3. *6th IEEE-RAS International Conference on Humanoid Robots, Genova, Italy*, pages 308–313, Decemeber 2006.
- [9] R. O. Ambrose, R. T. Savely, S. M. Goza, P. Strawser, M. A. Diftler, I. Spain, N. Radford, and L. Martin. Mobile manipulation using nasa’s robonaut. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA’04, New Orleans, LA, USA*, 2 :2104–2109, April-May 2004.
- [10] R. Bischoff and V. Graefe. Hermes : a versatile personal assistant robot. *Proceedings of the IEEE, Special Issue on Human Interactive Robots for Psychological Enrichment, KUKA Controls GmbH, Neubiberg, Germany*, 92(11) :1759–1779, November 2004.
- [11] S. Hashimoto and *et al.* Humanoid robots in waseda university - hadalay-2 and wabian. *Journal of Autonomous Robots*, 12(1) :25–38, January 2002.
- [12] K. Kosuge, M. Sato, and N. Kazamura. Mobile robot helper. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA’00, San Francisco, CA, USA*, pages 583–588, April 2000.
- [13] A. Konno, K. Nagashima, R. Furukawa, K. Nishiwaki, T. Noda, M. Inaba, and H. Inoue. Development of a humanoid robot saika. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS’97, Grenoble, France*, pages 805–810, September 1997.
- [14] E. H. Durfee. The distributed artificial intelligence melting pot. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(6) :1301–1306, November-December 1991.
- [15] R. Arkin. Cooperation without communication : Multiagent schema-based robot navigation. *Journal of Robotic Systems*, 9(3) :351–364, 1992.
- [16] R. Arkin. Reactive robotic systems. *Handbook of Brain Theory and Neural Networks, MIT Press*, pages 793–796, 1995.
- [17] M. J. Mataric, M. Nilsson, and K. T. Simsarian. Cooperative multi-robot box-pushing. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS’95, Pittsburgh, PA*, 3 :556–561, August 1995.
- [18] M. R. Ahmad, S. H. M. Amin, and R. Mamat. Development of reactive control

- strategy for multi-agent mobile robotics system. *Journal Technology*, 35(D) :99–114, 2001.
- [19] B. S. Pimental, G. A. S. Pereira, and M. F. M. Campos. On the development of cooperative behavior based mobile manipulators. *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, New York, USA*, pages 234–239, 2002.
- [20] V. Arguenon, A. Bergues-Lagarde, P. Bro, C. Rosenberger, and W. Smari. Multi-agent prototyping of agriculture robots. *Proceedings of the International Symposium on Collaborative Technologies and Systems, CTS'06, IEEE Computer Society Washington, DC, USA*, pages 282–288, May 2006.
- [21] V. Arguenon and A. Bergues-Lagarde. Simulation de robots vendangeurs sous oris. *Projet d'application, ENSIB, France*, 2006.
- [22] A. Hentout, B. Bouzouia, and Z. Toukal. Multi-agent architecture model for driving mobile manipulator robots. *International Journal of Advanced Robotic Systems*, 5(3) :257–268, 2008.
- [23] S. I. Martinez. A formalization for multiagent decision support in cooperative environments. a framework for situated agents. *Doctoral thesis, University of Girona, Catalonia, Spain*, 2008.
- [24] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'85, St Louis, Missouri, USA*, 2 :500–505, March 1985.
- [25] R. C. Arkin and T. Balch. Cooperative multiagent robotic systems. *Chapter in Artificial Intelligence and Mobile Robots, MIT/AAAI Press, Cambridge, MA*, 1998.
- [26] T. Balch and R. C. Arkin. Motor schema-based formation control for multiagent robot teams. *Proceedings of the First International Conference on Multi-Agent Systems, San Francisco, CA*, pages 10–16, 1995.
- [27] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal. Vehicle/arm coordination and multiple mobile manipulator decentralized cooperation. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'96, Osaka, Japan*, 2 :546–553, November 1996.

- [28] O. Khatib, K. Yokoi, O. Brock, K. Chang, and A. Casal. Robots in human environments. *Proceedings of the First Workshop on Robot Motion and Control, RoMoCo'99, Kiekrz, Poland*, pages 213–221, June 1999.
- [29] H. Osumi. Cooperative strategy for multiple mobile manipulators. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'96, Osaka, Japan, 2* :554–559, November 1996.
- [30] K. Kosuge and T. Oosumi. Decentralized control of multiple robots handling an object. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'96, Osaka, Japan, 1* :318–323, November 1996.
- [31] Y. Kume, Y. Hirata, and K. Kosuge. Coordinated motion control of multiple mobile manipulators handling a single object without using force/torque sensors. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'07, San Diego, USA, 2* :4077–4082, October 2007.
- [32] Y. Hirata, Y. Kume, Z.-D. Wang, and K. Kosuge. Handling of a single object by multiple mobile manipulators in cooperation with human based on virtual 3-d caster dynamics. *JSME International Journal, Series C*, 48(4) :613–619, 2005.
- [33] J. P. Desai, C.-C. Wang, M. Zefran, and V. Kumar. Motion planning for multiple mobile manipulators. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA '96, Minneapolis, USA, 3* :2073–2078, April 1996.
- [34] M. Abou-Samah. A kinematically compatible framework for collaboration of multiple nonholonomic wheeled mobile robots. *Thesis of Masters of Engineering, McGill University, Montreal, Canada*, 2001.
- [35] M. Fang, W. Chen, and Z. Li. Adaptive tracking control of coordinated nonholonomic mobile manipulators. *Proceedings of the 17th World Congress of The International Federation of Automatic Control, Seoul, Korea, 17* :4343–4348, July 2008.
- [36] H. Su and V. Krovi. Decentralized dynamic control of a nonholonomic mobile manipulator collective : A simulation study. *Proceedings of the ASME Dynamic Systems and Control Conference, DSCC2008, Ann Arbor, Michigan*, pages 1–8, October 2008.
- [37] O. Simonin and O. Grunder. A cooperative mobile robots architecture enabling

- paralyzed robots to move. *12th IFAC Symposium on Information Control Problems in Manufacturing, INCOM'06, Saint-Etienne, France*, 1 :197–202, May 2006.
- [38] O. Simonin. Reactive multi-agent approaches for the control of autonomous mobile robots. *1st National Workshop on "Control Architecture of Robots : software approaches and issues", CAR'06, Montpellier, France*, pages 183–191, April 2006.
- [39] P. Lucidarme, P. Rongier, and A. Liégeois. Implementation and evaluation of a reactive multi-robot system. *Proceedings of of IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM'01, Italia*, 1 :165–170, July 2001.
- [40] P. Lucidarme, O. Simonin, and A. Liégeois. Implementation and evaluation of a satisfaction/altruism based architecture for multi-robot systems. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'02, Washington, DC, USA*, pages 1007–1012, May 2002.
- [41] H. Yamaguchi. A cooperative hunting behaviour by mobile-robot troops. *International Journal of Robotics Research*, 18(9) :931–940, May 1999.
- [42] C. Y. Uny, A. S. Fukunaga, and A. B. Kahng. Cooperative mobile robotics : Antecedents and directions. *Autonomous Robots*, 4(1) :7–27, 1997.
- [43] R. Fierro, A. Das, J. Spletzer, J. Esposito, V. Kumar, J. P. Ostrowski, G. Pappas, C. J. Taylor, Y. Hur, R. Alur, I. Lee, G. Grudic, and B. Southall. A framework and architecture for multi-robot coordination. *The International Journal of Robotics Research*, 21(10-11) :977–995, October-November 2002.
- [44] M. I. Ribeiro J. Sequeira. Experiments on cooperative behaviour between two mobile manipulators. *Proceedings of the 6th International Symposium on Intelligent Robotic Systems, SIRS'98, Edinburgh, Scotland, UK*, pages 27–36, July 1998.
- [45] S. Djebbrani, A. Benali, and F. Abdessemed. Modelling and feedback control of an omni-directional mobile manipulator. *IEEE Conference on Automation Science and Engineering, CASE'2011, Trieste, Italy*, pages 785–791, August 2011.
- [46] S. Djebbrani, F. Abdessemed, and A. Benali. A multi-agent strategy for a simple cooperative behavior. *International Journal of Information Acquisition (IJIA), World Scientific Publishing Company*, 7(4) :331–345, December 2010.
- [47] W. Khalil and J. Kleinfinger. A new geometric notation for open and closed loop

- robots. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'86, San Francisco, CA, USA*, 3 :1174–1180, April 1986.
- [48] J. Neimark and N. Fufaev. Dynamics of nonholonomic systems. *Translations of Mathematical Monographs*, 33, 1972.
- [49] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1) :90–98, 1986.
- [50] W. Khalil et E. Dombre. Modélisation, identification et commande des robots. *Hermès*, 1999.
- [51] M. Renaud. Comment définir l'orientation d'un corps ? *Rapport technique 96078, LAAS-CNRS*, 1996.
- [52] J. J. Craig. Introduction to robotics : Mechanics and control. *Addison-Wesley*, 1989.
- [53] W. Khalil and E. Dombre. Modeling, identification and control of robots. *Hermes Penton Science*, 2002.
- [54] L. Sciavicco and B. Siciliano. Modelling and control of robots manipulators. *2nd Edition, Springer-Verlag Advanced Textbooks in Control and Signal Processing Series, London, UK*, 2000.
- [55] V. Padois. Enchaînements dynamiques de tâches pour des manipulateurs mobiles à roues. *Ph.D. thesis, Institut National Polytechnique de Toulouse, France*, 2005.
- [56] B. Bayle. Modélisation et commande cinématique des manipulateurs mobiles à roues. *Ph.D. thesis, LAAS-CNRS, Toulouse University, France*, 2001.
- [57] G. Mourioux, C. Novales, G. Poisson, and P. Vieyres. Omni-directional robot with spherical orthogonal wheels : concepts and analyses. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'06, Orlando, Florida, USA*, pages 3374–3379, May 2006.
- [58] G. Poisson, Y. Parmantier, and C. Novales. Modélisation cinématique d'un robot mobile omnidirectionnel à roues sphériques. *XVIème Congrès Français de Mécanique, Nancy, France*, pages 3–7, September 2001.
- [59] S. Djebriani, A. Benali, and G. Poisson. Input-state linearisation of an omni-directional mobile robot. *IEEE International Symposium on Industrial Electronics, ISIE'2010, Bary, Italy*, pages 1889–1894, July 2010.

- [60] S. Djebrani, A. Benali, and F. Abdessemed. Force-position control of a holonomic mobile manipulator. *12th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines, CLAWAR'2009, Istanbul, Turkey*, pages 1023–1030, September 2009.
- [61] S. Djebrani, A. Benali, and F. Abdessemed. Impedance control of an omnidirectional mobile manipulator. *14th International Conference on Methods and Models in Automation and Robotics, MMAR'2009, Miedzydroje, Poland*, August 2009.
- [62] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar. Robot dynamics and control. *John Wiley, New-York*, 1989.
- [63] Y. Zhang, Q. Cao, , and S. Miao. Adaptive inverse control of an omni-directional mobile robot. *First International Conference on Advances in Natural Computation, ICNC 2005, Changsha, China*, 3 :723–726, August 2005.
- [64] K. Watanabe, Y. Shiraishi, S. G. Tzafestas, J. Tang, and T Fukuda. Feedback control of an omnidirectional autonomous platform for mobile service robots. *Journal of Intelligent and Robotic Systems*, 22(3-4) :315–330, 1998.
- [65] K. Watanabe. Control of an omnidirectional mobile robot. *Second International Conference on Knowledge-Based Intelligent Electronic Systems, Australia*, pages 51–60, April 1998.
- [66] specAmotor. Re-40-148867 : Electric motor datasheet. *online* : http://www.specamotor.com/pdf/datasheet_RE-40-148867_en.pdf, 2011.
- [67] K. N. Chabane. Exploitation de la redondance pour la commande coordonnée d'un manipulateur mobile d'assistance aux personnes handicapées. *Ph.D. thesis, Université d'Evry-Val d'Essonne, France*, 2006.
- [68] M. Fruchard. Méthodologie pour la commande de manipulateurs mobiles non-holonômes. *Ph.D. thesis, L'école Nationale Supérieure des Mines de PARIS - Sophia Antipolisu, France*, 2005.
- [69] H. Seraji. An on-line approach to coordinated mobility and manipulation. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'93, Atlanta, GA, USA*, 1 :28–35, May 1993.

- [70] Y. Yamamoto. Control and coordination of locomotion and manipulation of a wheeled mobile manipulator. *Ph.D. Dissertation, University of Pennsylvania, Philadelphia, USA*, 1994.
- [71] K. Liu and F. L. Lewis. Decentralized continuous robust controller for mobile robots. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA '90, Cincinnati, USA*, 3 :1822–1827, May 1990.
- [72] P. Ruaux. Conception, modélisation et commandes dynamiques d'un robot mobile autonome modulaire. application à un environnement non contraint. *Ph.D. thesis, L'université Pierre et Marie Curie, France*, 1998.
- [73] T. Yoshikawa. Foundations of robotics : Analysis and control. *The MIT Press, Cambridge, Massachusetts*, 1990.
- [74] T. Yoshikawa. Manipulability of robotic mechanisms. *International Journal of Robotics Research*, 4(2) :3–9, 1985.
- [75] T. Yoshikawa. Analysis and control of robot manipulators with redundancy. *The First International Symposium of Robotics Research, MIT Press*, pages 735–747, 1984.
- [76] F. G. Pin, C. J. Hacker, K. B. Gower, and K. A. Morgansen. Including a non-holonomic constraint in the full space parameterization method for mobile manipulators motion planing. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA '97, Albuquerque, USA*, pages 2914–2919, April 1997.
- [77] U. M. Nassal. An approach to motion planning for mobile manipulation. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. "Advanced Robotic Systems and the Real World", IROS'94, Munich, Germany*, 2 :831–838, September 1994.
- [78] Y. Nakamura. Advanced robotics, redundancy and optimisation. *Addison-Wesley Publishing*, 1991.
- [79] Y. Yamamoto and X. Yun. Unified analysis on mobility and manipulability of mobile manipulators. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA '99, Detroit, Michigan, USA*, 2 :1200–1206, May 1999.

- [80] J. F. Gardner and S. Velinsky. A kinematics of mobile manipulators and implications for design. *Journal of Robotic Systems*, 17(6) :309–320, 2000.
- [81] B. Bayle, J.-Y. Fourquet, F. Lamiraux, and M. Renaud. Kinematic control of wheeled mobile manipulators. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'02, Lausanne, Switzerland*, pages 1572–1577, October 2002.
- [82] J. Borenstein and Y. Koren. The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Journal of Robotics and Automation*, 7(3) :278–288, June 1991.
- [83] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'85, St. Louis, Missouri, USA*, 2 :116–121, March 1985.
- [84] N. Hogan. Impedance control - an approach to manipulation. i - theory. ii - implementation. iii - applications. *ASME Journal of Dynamic Systems, Measurement and Control*, 107(1) :1–24, March 1985.
- [85] N. Hogan. Stable execution of contact tasks using impedance control. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'87, Raleigh, NC, USA*, 4 :1047–1054, March 1987.
- [86] R. J. Anderson. Dynamic damping control : Implementation issues and simulation results. *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'90, Cincinnati, USA*, 1 :68–77, May 1990.
- [87] T. Arai and J. Ota. Motion planning of multiple mobile robots using virtual impedance. *Journal of Robotics and Mechatronics*, 8(1) :67–74, 1996.
- [88] R. Carelli and R. Kelly. On adaptive impedance/force controller for robot manipulators. *IEEE Transactions on Automatic Control*, 36(8) :967–971, August 1991.
- [89] A. A. Goldenberg. Implementation of force and impedance control in robot manipulators. *Proceeding of the 27th IEEE International Conference on Decision and Control, Philadelphia, PA*, 3 :1626–1632, April 1988.
- [90] A. Benali. Outils d'adaptation d'impédance pour une structure de micro-macro manipulation. *Ph.D. thesis, Laboratoire de robotique de Paris, France*, 1997.

- [91] R. Carelli, H. Secchi, and V. Mut. Algorithms for stable control of mobile robots with obstacle avoidance. *Latin American Applied Research*, 29(3/4) :191–196, 1999.
- [92] F. Abdessemed and S. Djebrani. Trajectory generation for mobile manipulators. *Robot Manipulators Trends and Development, INTECH*, pages 335–360, March 2010.
- [93] A. Isidori. Nonlinear control systems. *Third edition, Springer-Verlag, New York*, 1995.
- [94] J. J. Slotine and W. Li. Applied nonlinear control. *Prentice-Hall, New Jersey*, 1991.
- [95] Shankar Sastry. Nonlinear systems : Analysis, stability and control. *Interdisciplinary Applied Mathematics, Systems and Control, Springer-Verlag*, 1999.
- [96] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3) :338–353, June 1965.
- [97] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *III, Information Sciences*, 9(1) :43–80, December 1975.
- [98] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *I, Information Sciences*, 8(3) :199–249, July 1975.
- [99] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *II, Information Sciences*, 8(4) :301–357, October 1975.
- [100] L.A. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(1) :28–44, January 1973.
- [101] P. J. King and E. H. Mamdani. The application of fuzzy control systems to industrial processes. *Automatica*, 13(3) :235–242, 1977.
- [102] F. Abdessemed. Application de la logique floue à la commande des processus complexes de la manipulation mobile. *Ph.D. thesis, Université de Batna, Algérie*, 2003.
- [103] B. Bouchon-Meunier. La logique floue et ses applications. *Editions Addison-Wesley*, 1995.
- [104] B. Hansruedi. Réglage par logique floue. *Presses polytechniques et universitaires romandes*, 1994.
- [105] M. Sugeno and M. nishida. Fuzzy control of a model car, in : Fuzzy sets and systems. *Elsevier Science Publisher*, 16(2) :103–113, July 1985.

- [106] E. H. Mamdani and S. Assilian. Application of fuzzy algorithms for control of simple dynamic plant. *Proceedings of the Institute of Electrical Engineers*, 121(12) :1585–1588, December 1974.
- [107] F. Abdessemed and K. Benmahammed. A two-layer robot controller design using evolutionary algorithms. *Journal of Intelligent and Robotic System, Kluwer Academic Publishers Hingham, MA, USA*, 30(1) :73–94, January 2001.
- [108] F. Abdessemed, K. Benmahammed, and E. Monacelli. A fuzzy-based reactive controller for a non-holonomic mobile robot. *Journal of Robotics and Autonomous Systems*, 47(1) :31–46, 2004.
- [109] C. Canudas de Wit and A. Aubin. Parameters identification of robotic manipulators via sequential hybrid estimation algorithms. *Proc. IFAC, Congress, Tallin*, 9 :178–183, 1990.
- [110] F. Reyes and R. Kelly. Experimental evaluation of identification schemes on a direct drive robot. *Robotica*, 15 :563–571, 1989.
- [111] N. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1(1) :7–38, July 1998.
- [112] J. Ferber. Les systèmes multi-agents. vers une intelligence collective. *Informatique Intelligence Artificielle. InterEditions, Paris*, 1995.
- [113] L. Overgaard, G. H. Petersen, and J. W. Perram. Motion planning for an articulated robot : a multi-agent approach. *Distributed Software Agents and Applications : 6th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'94*, pages 206–219, August 1994.
- [114] M. J. Mataric. Reinforcement learning in the multi-robot domain. *Autonomous Robots*, 4 :73–83, 1997.
- [115] D. C. MacKenzie, R. C. Arkin, and J. M. Cameron. Multiagent mission specification and execution. *Autonomous Robots*, 4(1) :29–52, 1997.
- [116] L. Parker. The effect of action recognition and robot awareness in cooperative robotic teams. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'95, Pittsburgh, USA*, 1 :212–219, August 1995.

-
- [117] J. S. Bay. Design of the army-ant cooperative lifting robot. *IEEE Robotics and Automation Magazine*, 2(1) :36–43, March 1995.
- [118] R. G. Brown. A pusher/steerer model for strongly cooperative mobile robot manipulation. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'95, Pittsburgh, USA*, 3 :562–568, August 1995.
- [119] R. C. Kube and H. Zhang. Collective robotic intelligence. *Proceedings of the Second International Conference on Simulation of Adaptive Behavior : From Animals to Animats 2, MIT Press, Cambridge, MA, USA*, pages 460–468, September 1992.
- [120] F. Harrouet. oris : S'immerger par le langage pour le prototypage d'univers virtuels à base d'entités autonomes. *Ph.D. thesis, Université de Bretagne Occidentale, France*, 2000.
- [121] F. Harrouet. Le langage oris. guide d'utilisation et manuel de référence. *E.N.I.B/LI², France*, November 1997.
- [122] S. Virtualys. Manuel de référence oris 2.1. *E.N.I.B, France*, 1999.