

Pauline LAFOUX
élève à l'INSA Strasbourg
Spécialité Mécatronique
Parcours Robotique
Promotion 2018

Rapport du Projet de Fin d'Études :

Commande référencée vision d'un mécanisme à cinq barres en bois



Tuteurs :	Pierre RENAUD Sébastien BRIOT	Professeur, ICube, INSA Strasbourg Chercheur CNRS au LS2N
Organisme d'accueil :	LS2N	1 rue de la Noë 44321 NANTES
Dates :	du 05/03/2018	au 06/08/2018

Condensé

Observer l'état d'un robot à l'aide d'une caméra déportée permet d'asservir la position de l'effecteur sans la connaissance précise du modèle géométrique. Cet avantage des commandes référencées vision par rapport aux commandes référencées modèles peut permettre de bonnes performances en précision dans l'espace opérationnel, même pour des robots sujets à des variations dimensionnelles comme les robots en bois. Parmi les nombreux types de commande référencée vision, une méthode adaptée aux robots parallèles consiste à observer les bras du robot plutôt que simplement l'effecteur. Ces techniques qui exploitent l'extraction de droites dans l'image, semblent intéressantes pour les performances en précision.

Ce rapport se concentre sur le cas d'un prototype de mécanisme parallèle à cinq barres possédant des membres en bois. Il a été considéré différents types d'informations visuelles et matrices d'interactions existantes qui se basent sur l'observation de l'effecteur ou des contours des bras. Une étude du comportement des contrôleurs sur un modèle cinématique et dynamique du robot a été réalisée par simulation. Grâce à des essais expérimentaux nous avons pu caractériser les performances en précision des deux types de contrôleurs. Il a été montré qu'une précision de positionnement inférieure à $100\ \mu m$ peut être atteinte dans un espace opérationnel de $200mm \times 800mm$ grâce au contrôleur basé sur l'observation de l'effecteur. Et enfin une étude en simulation a permis de comparer la robustesse des contrôleurs aux variations de longueur et de diamètre des bras.

Abstract

The control of robots' end-effector position can be done thanks to an eye-to-hand camera. In this case the precision does not depend on an accurate calibration of geometric parameters. This advantage of visual servoing enables to get good precision performances in the workspace, even for robots with dimensional variations like wooden robots. Among the different types of visual servoing techniques, one of them consist of observing the arms of the robot instead of the classical end-effector observation. This technique is especially well suited for parallel robots.

This report focus on a five bars mecanism prototype with wooden arms. It has been considered differents types of features and interaction matrices : based on the observation of the end-effector or of the edges of the arms. The kinematic and dynamic respons of the robot has been simulated and improved for the different controllers considered. Thanks to experiments, the precision performances have been qualified for the two types of controllers. A positioning precision of $100\ \mu m$ has been reached in a large workspace of $200mm \times 800mm$ thanks to the controller based on the end-effector observation. Finally a study by simulation has been performed to compare the robustness of the controllers towards dimensional variations of the length and diameter of the arms.

Remerciements

Tout d'abord je tiens à remercier toute l'équipe de RobEcolo avec qui j'ai travaillé durant ce stage. Merci à Sébastien Briot pour son intérêt et ses retours rapides. Merci à Lila Kaci pour le soutien. Et merci à David Llevat pour sa patience et son aide précieuse.

Je remercie M. Kermorgant d'avoir pris le temps de répondre à mes questions. Plus généralement je remercie tous les membres de l'équipe ARMEN avec qui j'ai pu échanger sur mes problématiques et les pistes de solutions. Merci pour votre intérêt et vos propositions. Merci également pour le partage sur vos travaux respectifs.

Je remercie mes tuteurs académiques M. Renaud de l'INSA Strasbourg et M. Gangloff de Télécom Physique Strasbourg pour le suivi tout au long de mon stage.

Je voudrais remercier les stagiaires et doctorants qui ont participé à la bonne ambiance durant les pauses. Je tiens à remercier tous les membres du LS2N qui m'ont partagé leur expérience professionnelle et m'ont permis d'avoir un regard plus éclairé sur le travail dans le domaine de la recherche.

Et enfin je tiens à remercier ma famille et mes amis qui ont été d'un grand soutien.

Table des matières

Condensé	i
Remerciements	iii
Notations	vii
Introduction	1
1 Définition du sujet	3
1.1 Présentation du laboratoire LS2N	3
1.1.1 Description générale	3
1.1.2 Fonctionnement hiérarchique du laboratoire	3
1.2 Le projet RobEcolo	4
1.3 Intérêt de la commande référencée vision	4
1.4 Objectifs du stage	6
2 Etat de l'art	9
2.1 Principes fondamentaux de la commande référencée vision	9
2.1.1 Modélisation des informations visuelles	9
2.1.2 Commande	10
2.1.3 Stabilité	11
2.2 Asservissement visuel des robots parallèles	14
2.2.1 Intérêt de la vision pour les robots parallèles	14
2.2.2 Représentation géométrique des jambes	14
2.2.3 Commandes basées sur l'observation des jambes	16
2.2.4 Commandes basées sur l'observation de l'effecteur	17
2.3 Techniques d'asservissement visuel avancées	19
2.3.1 Amélioration des performances	19
2.3.2 Commande référencée multi-capteurs	19
3 Différents contrôleurs envisagés	23
3.1 Contrôleur basé observation de l'effecteur	23
3.2 Contrôleur basé contours des jambes	24
3.2.1 Matrice d'interaction basée direction des bras	25
3.2.2 Matrice d'interaction basée direction et position des bras	25
3.3 Contrôleur basé reconstruction	26
3.4 Contrôleur hybride	26
3.5 Bilan des contrôleurs envisagés	27
4 Simulation de la réponse du système	31
4.1 Modélisation du système complet sur Simulink	31
4.1.1 Modèles utilisés	31
4.2 Amélioration de la commande	32
4.2.1 Limite en vitesse des informations visuelles	32
4.2.2 Contrainte sur l'espace de travail	33
4.2.3 Compatibilité avec la commande bas-niveau	34
4.3 Bilan de la simulation du système	36

5	Caractérisation en précision	37
5.1	Méthode et critère de précision	37
5.1.1	Méthode d'essai	37
5.1.2	Critère de précision dans l'espace opérationnel	37
5.2	Résultats obtenus	38
5.2.1	Vérification en simulation	38
5.2.2	Résultats d'essais sur robot réel	39
5.2.3	Analyse du bruit dans l'image	41
5.3	Robustesse aux variations dimensionnelles	41
5.4	Bilan de la caractérisation en précision	43
	Conclusion	45
6	Annexes	47
	Annexes	47
6.1	Détails sur les contrôleurs basés sur l'observation des jambes	47
6.1.1	Construction des informations visuelles \mathbf{s}_n	47
6.1.2	Construction des informations visuelles \mathbf{s}_r	47
6.1.3	Matrice d'interaction basée direction des jambes	48
6.1.4	Matrice d'interaction basée direction et position des bras	49
6.2	Détails sur la saturation en vitesse avec le gain adaptatif	50
6.3	Résultats robustesse aux variations dimensionnelles	51

Notations

Règles générales

λ minuscule	:	scalaire
e minuscule gras	:	vecteur
<u>e</u>	:	vecteur unitaire
M majuscule gras	:	matrice
s^*	:	grandeur de référence pour s
\hat{s}	:	valeur estimée pour s
\dot{s}	:	dérivée de s
${}^i s$:	s exprimé dans le repère i

Algèbre

M^T	:	matrice transposée de M
M^+	:	matrice pseudo-inverse de M
$\text{Ker}(M)$:	noyau de M
$\mathbf{0}_{p,n}$:	matrice nulle de dimension $p \times n$
\mathbb{I}_n	:	matrice identité de dimension $n \times n$

Robotique et commande

q	:	position articulaire
p	:	configuration de l'effecteur
s	:	informations visuelles
$\mathbf{e} = \mathbf{s} - \mathbf{s}^*$:	erreur de la boucle vision
L	:	matrice d'interaction
$\boldsymbol{\tau}$:	torseur cinématique de la caméra par rapport à la scène observée
λ	:	gain de l'asservissement visuel

Introduction

L'industrie recherche des robots manipulateurs précis et fiables. En comparaison aux robots séries, les robots parallèles intéressent le monde industriel principalement pour leur rapidité. D'un autre côté les nouvelles problématiques environnementales poussent les industriels à réfléchir à leur impact. L'utilisation de matériaux bio-sourcés tel que le bois, pour la conception des robots manipulateurs fait partie de la réponse proposée par le projet RobEcolo.

Par rapport aux matériaux métalliques, le bois possède certains avantages mécaniques comme son rapport masse/raideur mais ces dimensions sont plus sensibles aux conditions hygrométriques. Une variation du taux d'humidité peut entraîner des variations dimensionnelles de l'ordre du mm pour un bras de quelques dizaines de cm de longueur. Or l'industrie cherche souvent des performances en précision en dessous du mm dans l'espace opérationnel. Une simple commande articulaire, basée sur la connaissance d'un modèle précis au mm près s'avère donc inadéquate.

Pour les commandes référencées vision, l'erreur est définie par l'observation d'un objet dans l'image. La méthode classique consiste à observer directement l'effecteur. Une connaissance imprécise du modèle géométrique du robot n'affecte donc plus directement le positionnement de l'effecteur. Parmi les nombreuses possibilités de commande référencée vision, certaines exploitent l'observation des bras du robot. Ces commandes sont particulièrement bien adaptées aux robots parallèles. En effet l'état d'un robot parallèle peut être entièrement défini par la position de ces bras, alors qu'une configuration de l'effecteur peut admettre plusieurs états possibles du robot.

Durant ce stage nous utiliserons le prototype de mécanisme à cinq barres en bois réalisé au cours du projet RobEcolo. Avec ce robot parallèle nous chercherons à atteindre un objectif de précision en positionnement de l'effecteur en dessous de $500\mu m$, dans un large espace de travail de $800mm \times 200mm$. Assurer l'objectif en précision à l'aide de la commande référencée vision revient à répondre à différentes problématiques :

- Quelles informations visuelles choisir pour permettre une meilleure précision et une robustesse par rapport aux variations dimensionnelles ?
- Quelle matrice d'interaction choisir pour éviter les problèmes de singularités et de minima locaux ?
- Comment prendre en compte certaines contraintes, telles que les limites de l'espace de travail ou le maintien du suivi des informations visuelles dans l'image, pour assurer une commande fiable ?

Pour répondre à ces questions, il sera réalisé une étude des différentes lois de commande existantes qui sera présentée en partie 2. Le choix des matrices d'interactions et des informations visuelles qui semblent pertinentes pour notre application sera présenté en partie 3. Le système complet sera modélisé et utilisé en simulation pour tester le comportement des matrices d'interactions retenues à partir d'un modèle dynamique du robot RobEcolo. Assurer une bonne précision du système, implique d'assurer la convergence et la stabilité du système complet. Des moyens d'améliorer la commande seront implémentés pour répondre à certaines contraintes, qui seront présentés en partie 4. Et enfin les performances en précision seront caractérisées à l'aide de simulations et d'essais expérimentaux en partie 5.

1 Définition du sujet

Le projet de fin d'études (PFE) a été réalisé dans le laboratoire LS2N sur le site de Centrale Nantes, dans le cadre du projet RobEcolo.

1.1 Présentation du laboratoire LS2N

1.1.1 Description générale

Le Laboratoire des Sciences du Numérique de Nantes (LS2N - UMR6004) est créé en janvier 2017, résultant de la fusion de l'Institut de recherche en communications et cybernétique de Nantes (IRCCyN - UMR6597) et du Laboratoire d'informatique de Nantes Atlantique (LINA - UMR6241). Il réunit 450 personnes autour des sciences du numérique sur 5 sites différents (dont Centrale Nantes) et dépend de 4 tutelles (cf. Figure 1).



FIGURE 1 – Tutelles du LS2N

Son activité de recherche est structurée en 5 pôles de compétence :

- Conception et conduite de systèmes (CCS)
- Robotique, Procédés, Calculs (RPC)
- Science des Données et de la Décision (SDD)
- Signaux, Images, Ergonomie et Langues (SIEL)
- Science du Logiciel et des Systèmes distribués (SLS)

Elle s'oriente aussi autour de 5 thèmes transverses :

- Entreprise du Futur
- Gestion de l'énergie et maîtrise des impacts environnementaux
- Sciences du vivant
- Véhicules et mobilités
- Création, culture et sociétés numériques

Autour de chacun des thèmes de compétences, on retrouve entre 4 et 6 équipes de recherche. Les thèmes transverses regroupent eux toutes les équipes et sont encadrés par des chargés de mission étant eux mêmes des chercheurs.

Le PFE se positionne dans le pôle de compétence Robotique, Procédés, Calculs, au sein de l'équipe ARMEN (Autonomie des Robots et Maîtrise des interactions avec l'ENVironnement). Cette équipe cherche à concevoir et maîtriser des systèmes capables d'adresser les deux facettes du terme "environnement" (au sens du milieu dans lequel le robot évolue ou au sens écologique du terme).

1.1.2 Fonctionnement hiérarchique du laboratoire

Comme indiqué précédemment, 23 équipes de recherche dirigées par un chef d'équipe s'articulent autour des 5 thèmes de recherche ayant eux mêmes un responsable. Celles-ci sont sous la direction de C. Jard et C. Chevalleray, directeur et directeur adjoint du LS2N. Autour de

ces équipes de recherche, il y a quelques postes plus classiques telles que les postes de coordination technique et les postes administratifs. On y retrouve notamment le soutien à la recherche, le service informatique, le secrétariat de direction, les finances, les ressources humaines et la valorisation/communication sur le travail effectué. Plus de détails sur l'organigramme et l'organisation sont disponibles sur le site officiel du laboratoire [LS2].

1.2 Le projet RobEcolo

Le stage s'intègre dans le projet RobEcolo. Ce dernier cherche à diminuer l'impact environnemental des robots industriels tout en garantissant leurs performances en termes de rigidité et précision. Un des champs de recherche consiste à remplacer les pièces métalliques du robot par des corps en matériaux bio-sourcés, tels que le bois. Si le bois est un matériau compétitif en termes de ratio masse/raideur, il possède deux inconvénients majeurs pour la conception de robots : la connaissance approximative des performances mécaniques (modules d'élasticité) et la variation des dimensions des corps avec l'hygrométrie.

Le projet est mené principalement par Sébastien Briot un chercheur CNRS, Lila Kaci une doctorante et David Llevat Pamies un ingénieur du LS2N (Laboratoire des Sciences du Numérique de Nantes) avec la collaboration de membres de l'ESB (École Supérieure du Bois).

Au cours du projet un prototype de mécanisme parallèle cinq barres a été réalisé. J'ai d'ailleurs eu l'occasion de participer à la conception du prototype durant mon stage de Master 1 en réalisant la CAO de ce dernier. Les articulations sont en matériaux métalliques alors que les bras sont en bois. Le projet vise à terme à vérifier les critères de performances en raideur et précision sur un large espace de travail, défini par un rectangle de $800mm \times 200mm$ (représenté en vert Figure 5). Les performances du robot en raideur ont été étudiées et validées sur le prototype. Il reste maintenant à assurer la précision.

Les paramètres géométriques du robot ont été déterminés par un algorithme d'optimisation réalisé durant le projet. Cette optimisation assure des dimensions optimales des membres du robot pour une commande référencée vision avec observation des contours des jambes [KBBP16] [KBBM18]. Les variations dimensionnelles maximum des membres en bois dans des conditions hygrométriques défavorables ont été évaluées lors d'études préliminaires [Kac18]. Les résultats prévoient une variation de moins de $1mm$ de la longueur et du diamètre de chaque bras.

Le dernier objectif du projet est : assurer la précision en positionnement, garantie inférieure à $0.5mm$ dans l'espace opérationnel, malgré les variations potentielles de dimension des corps, grâce à une commande référencée vision.

1.3 Intérêt de la commande référencée vision

La commande des robots manipulateurs, doit permettre le positionnement de l'effecteur du robot par rapport à son environnement, l'espace cartésien. Pour réaliser une consigne dans cet espace à l'aide de capteurs proprioceptifs il existe deux méthodes classiques : la commande articulaire avec utilisation du modèle géométrique inverse (MGI) (Figure 2) et la commande cartésienne référencée modèle (Figure 3). Ces commandes ne sont pas bien adaptées aux robots parallèles, notamment dû au fait qu'une position articulaire \mathbf{q} peut admettre plusieurs solutions de configuration de l'organe terminal. La configuration d'un objet est notée \mathbf{p} et regroupe les informations de position et d'orientation de cet objet par rapport à un repère de base. De plus, la précision dans l'espace opérationnel avec ces méthodes dépend de l'estimation des modèles géo-

métriques du robot. En pratique leur détermination et leur utilisation peuvent poser problème. Ces modèles sont susceptibles de changer, notamment dans le monde industriel, par exemple suite au port d'un outil ou à cause de variations dimensionnelles.

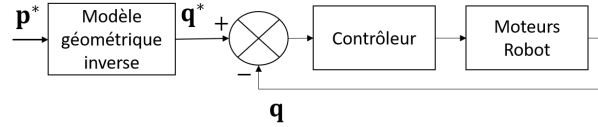


FIGURE 2 – Commande articulaire avec consigne dans l'espace cartésien

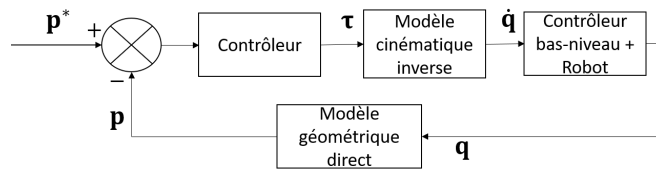


FIGURE 3 – Commande cartésienne référencée modèle

Les capteurs extéroceptifs, tels que les caméras, donnent par définition des informations sur l'environnement dans lequel ils évoluent. Les techniques d'asservissement visuel (Figure 4) consistent à utiliser les informations fournies par une ou plusieurs caméras afin de contrôler les mouvements d'un système robotique. Ces techniques, aussi regroupées sous le nom de commande référencée vision, consistent à intégrer directement les mesures fournies par un système de vision dans les lois de commande en boucle fermée sur les informations visuelles extraites [Cha02].

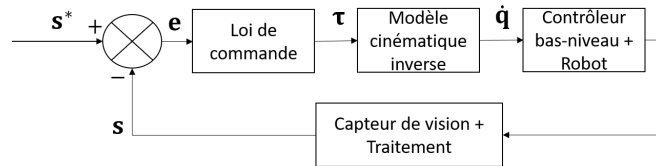


FIGURE 4 – Commande référencée vision

L'intérêt des commandes référencées vision pour la robotique est multiple. Tout d'abord elles peuvent permettre une plus grande précision de positionnement que les commandes basées modèles. En effet grâce à ces capteurs extéroceptifs, l'estimation de la position de l'organe terminal peut s'affranchir de la connaissance précise du modèle géométrique du manipulateur. L'estimation de ces paramètres est toujours nécessaire pour la commande dans le modèle cinématique inverse. Cependant les erreurs d'estimation n'affectent plus la précision comme pour les commandes basées exclusivement sur capteurs proprioceptifs (Figure 2 et Figure 3). Ces erreurs peuvent néanmoins amener le système en instabilité [CH07]. Les problématiques majeures des asservissements visuels sont justement l'étude de leur stabilité et de leur robustesse face aux

bruits, aux perturbations visuelles et aux erreurs de calibration. Or, pour que ce type de commande puisse être utilisé dans des conditions industrielles, il est nécessaire qu'elle soit fiable et précise. C'est pourquoi un des domaines de la recherche actuelle, vise à rendre ces commandes les plus performantes possible en terme de précision.

1.4 Objectifs du stage

L'intitulé du sujet est "*Commande référencée vision pour un mécanisme à cinq barres en bois*". Le stage se focalise sur l'influence des informations visuelles et du correcteur de la boucle de vision, aussi appelé loi de commande, dans le cadre d'un asservissement visuel indirect. La dynamique du robot est prise en compte dans une boucle bas-niveau avec retour par capteurs proprioceptifs. Les tâches de calibration, extraction, mise en correspondance et suivi de primitives dans l'image ainsi que la boucle de commande bas-niveau du robot ne seront pas directement abordées, car elles font l'objet du travail d'un autre membre de l'équipe, David Llavat Pamies. La thèse exploitera également les résultats d'une étude d'optimisation de la position de la caméra faite en simulation au cours du projet RobEcolo [Kac18]. Cette étude a permis d'établir la position et l'orientation optimale de la caméra pour garantir la précision souhaitée dans le cas de l'observation des jambes (cf Figure 5).

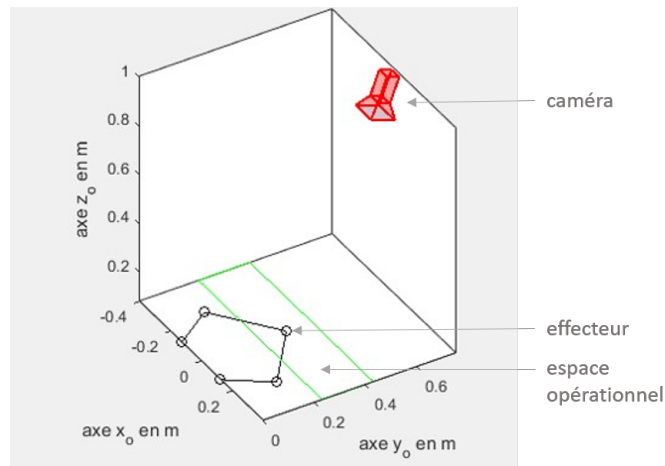


FIGURE 5 – Positionnement de la caméra par rapport au robot

Le stage consiste donc à étudier différentes lois de commande existantes qui permettent de répondre à l'objectif en précision fixé. Les correcteurs pourront différer selon :

- l'objet observé : observation de l'effecteur, observation des bras (cf. Figure 6) ;
- les informations visuelles utilisées : point, direction des droites, coordonnées de Plücker de droites ;
- la matrice d'interaction utilisée ;
- l'espace des tâches : plan image, reconstruction dans l'espace 3D ;

Une modélisation du système complet sera créée pour tester le comportement des différents contrôleurs retenus, sur le modèle dynamique du robot RobEcolo. Assurer une bonne précision du système, implique d'assurer la convergence et la stabilité du système complet. Il sera étudié et implémenté quelques moyens simples d'améliorer la commande avec la prise en compte de certaines contraintes. On peut citer le maintien du suivi des primitives visuelles et l'évitement des

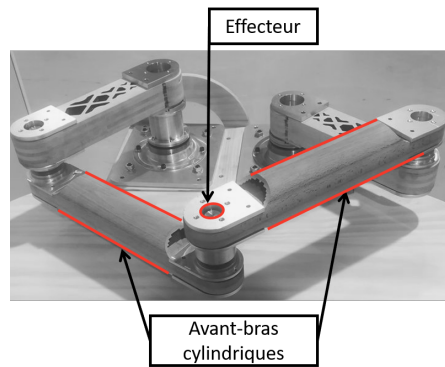


FIGURE 6 – Parties observées pour l’asservissement visuel de RobEcolo

positions singulières du mécanisme. Et enfin les performances en précision seront caractérisées à l’aide de simulations et d’essais expérimentaux.

Avant de rentrer en détail sur le travail réalisé durant le stage, il convient de rappeler les principes fondamentaux de la commande référencée vision. Un état de l’art plus précis sera présenté sur la commande des robots parallèles par vision. Et enfin nous verrons les différentes possibilités de commande référencée multi-capteurs.

2 Etat de l'art

2.1 Principes fondamentaux de la commande référencée vision

Tout asservissement visuel vise à minimiser une fonction de tâche qui traduit l'objectif à atteindre [SMB91][ECR90]. Un exemple typique de fonction de tâche pour l'asservissement visuel est l'erreur entre k informations visuelles dans la situation courante \mathbf{s} , et les k informations visuelles dans la situation souhaitée \mathbf{s}^* :

$$\mathbf{e}(t) = \mathbf{s} - \mathbf{s}^* \quad (1)$$

Il existe de nombreuses manières de définir ces informations visuelles et de réguler cette fonction de tâche.

2.1.1 Modélisation des informations visuelles

Matrice d'interaction La matrice d'interaction \mathbf{L} est un élément fondamental dans la commande référencée vision. Cette matrice permet de relier les variations des informations visuelles $\dot{\mathbf{s}}$ au mouvement relatif entre la caméra et la scène observée, tel que :

$$\dot{\mathbf{s}} = \mathbf{L}\boldsymbol{\tau} \quad (2)$$

$\boldsymbol{\tau}$ représente le torseur cinématique relatif entre la caméra et la scène, exprimé dans le repère caméra R_c en son origine O_c . On considère que seuls les mouvements du robot sont susceptibles de faire varier la valeur de l'information visuelle.

Il est ainsi possible de remonter aux vitesses articulaires du robot, à condition de connaître la matrice de transformation ${}^c\mathbb{T}_i$ entre repère caméra et un repère quelconque R_i du robot, ainsi que la matrice jacobienne du robot ${}^i\mathbf{J}$ exprimée dans ce même repère. Dans le cas d'une caméra déportée, fixe dans le repère de base R_o du robot, qui observe l'organe terminal, l'expression des variations des informations visuelles en fonction des vitesses articulaires du robot peut être donnée par :

$$\dot{\mathbf{s}} = -\mathbf{L}{}^c\mathbb{T}_o{}^o\mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \quad (3)$$

La matrice d'interaction peut être calculée analytiquement (cf. Section 2.1.1) ou bien déterminée par apprentissage sous forme numérique [LDJC04] [HA94] [JFN97]. Cependant ces méthodes nécessitent une connaissance précise du torseur cinématique de la caméra en plus des variations des informations visuelles. L'étude théorique de stabilité et robustesse n'est pas non plus réalisable.

Informations visuelles 2D ou 3D Une étape importante consiste à déterminer le type d'informations visuelles \mathbf{s} utilisé dans la fonction de tâche. Il existe principalement deux grands types d'informations :

- informations visuelles de type 2D, c.à.d. exprimées dans le plan image 2D (nommé aussi espace capteur) ;
- informations visuelles de type 3D, c.à.d. extraites de l'image mais reconstruites et définies dans l'espace opérationnel 3D (ou espace cartésien)

Ces deux grands types ont donné lieu à l'établissement de deux méthodes d'asservissement très connues : l'asservissement visuel basé image (IBVS pour *Image Based Visual Servoing*) et l'asservissement visuel basé objet (PBVS pour *Position Based Visual Servoing*). Chaque méthode a ses avantages et inconvénients (cf. Section 2.1.3 pour plus de détails). De manière plus générale nous appellerons l'espace dans lequel est défini l'erreur \mathbf{e} , l'espace des tâches.

Choix des primitives géométriques Les informations visuelles, appelées parfois primitives visuelles, ne sont pas à confondre avec les primitives géométriques. En effet les informations visuelles dépendent des primitives (ou entités) géométriques choisies et observées dans la scène 3D. Dans la suite du document nous emploierons le terme primitive pour parler de primitive géométrique.

La plus répandue est la primitive point, mais d'autres sont aussi utilisées : droite, ellipse, contour. Le paramétrage de ces primitives peut différer en fonction de l'espace considéré (capteur en 2D ou opérationnel en 3D) et des coordonnées choisies. Par exemple une droite peut être définie dans l'espace opérationnel, par ces coordonnées paramétriques ou de Plücker normées (cf. Section 2.2.2) [Plu65]. La matrice d'interaction, qui dépend du type d'informations visuelles, a été déterminée analytiquement pour un certain nombre de ces primitives :

- informations de type 2D : pour des primitives de type point, droite et ellipse [ECR90], contour [Cha04] etc.
- information de type 3D : pour des primitives de type point, droite [AEH02][VBPC14] etc.

Estimation de pose On pourra citer deux méthodes principales en mono-caméra pour estimer la pose d'un objet d'intérêt :

- Une connaissance suffisante de la géométrie de l'objet d'intérêt et de primitives géométriques identifiables peut permettre d'estimer la pose de l'objet [VBPC14] ;
- Par le placement d'un marqueur visuel *ad hoc* (exemple : mire) sur l'objet d'intérêt, la loi de commande et le traitement d'image peuvent s'avérer très efficaces [DAMM06].

2.1.2 Commande

Une fois les informations visuelles définies, la réalisation de la fonction de tâche par asservissement visuel nécessite l'élaboration d'une loi de commande en boucle fermée sur ces informations.

Loi de commande La loi de commande, également appelée correcteur de la boucle de vision, permet de réguler la fonction de tâche. Différentes lois de commande ont été imaginées (cf. Section 2.3.1). La plus connue et utilisée est la loi de commande proportionnelle qui vise une décroissance exponentielle de la fonction de tâche, à savoir une variation de l'erreur désirée :

$$\dot{\mathbf{e}}^* = -\lambda \mathbf{e} \quad (4)$$

avec $\lambda > 0$ le gain de l'asservissement. Si le système est stable, la valeur de ce gain va imposer une convergence plus ou moins rapide de l'erreur. Le contrôleur de la boucle de vision permet de calculer une consigne de vitesse dans l'espace opérationnel ou dans l'espace articulaire.

Commande dans l'espace des vitesses opérationnelles En dérivant l'équation (1), en supposant les informations désirées constantes par rapport au temps $\dot{\mathbf{s}}^* = 0$ et en utilisant (2), nous obtenons la variation de l'erreur réelle :

$$\dot{\mathbf{e}} = \mathbf{L}\boldsymbol{\tau}$$

Ainsi la consigne de vitesse idéale qui en découle est $\boldsymbol{\tau}^* = -\lambda \mathbf{L}^+ \mathbf{e}$. Le symbole $^+$ désigne la pseudo-inverse de la matrice. Or \mathbf{L}^+ ou \mathbf{L} ne sont jamais connues parfaitement en pratique. On utilise donc dans la loi de commande une approximation de ces matrices, notée grâce au symbole $\hat{\cdot}$. La consigne de vitesse devient :

$$\boldsymbol{\tau}^* = -\lambda \widehat{\mathbf{L}}^+ \mathbf{e} \quad (5)$$

Cette loi de commande proportionnelle dépend de l'approximation de la matrice d'interaction.

Commande dans l'espace des vitesses articulaires Il arrive fréquemment que la commande bas-niveau du robot se fasse dans l'espace des vitesses articulaires, plutôt que dans l'espace des vitesses opérationnelles, notamment lorsque le système possède moins de 6 degrés de liberté (ddl). La commande peut se trouver par le même raisonnement que précédemment. Posons la matrice $\mathbf{M} \in \mathbb{R}^{k \times n}$ avec k le nombre d'informations visuelles et n le nombre d'articulations motorisées du robot. Si on se place dans le cas d'une caméra déportée d'après (3) on a :

$$\dot{\mathbf{e}} = \mathbf{M}\dot{\mathbf{q}} \text{ avec } \mathbf{M} = -\mathbf{L}^c \mathbb{T}_o \circ \mathbf{J}(\mathbf{q})$$

La consigne envoyée à la boucle bas-niveau du robot est donc :

$$\dot{\mathbf{q}}^* = -\lambda \widehat{\mathbf{M}}^+ \mathbf{e} \quad (6)$$

$\widehat{\mathbf{M}}^+$ est définie à partir de l'approximation des matrices $\mathbf{L}, {}^c\mathbb{T}_o$ et $\circ \mathbf{J}$.

Il est important de remarquer que le comportement du système qui en découle sera différent en fonction de l'espace de commande de la boucle bas-niveau. Par exemple la commande (5) assure que les vitesses opérationnelles seront minimales alors que (6) assure que les vitesses articulaires seront minimales.

Approximation de la matrice d'interaction L'approximation de la matrice d'interaction (ou de sa pseudo-inverse) peut être déterminée de différentes manières, ce qui va influencer le comportement de la commande. Prenons en exemple le cas d'informations de type 2D (IBVS) avec des primitives points en considérant le cas plus simple d'une commande dans l'espace des vitesses opérationnelles. La matrice d'interaction analytique dépend alors de la position des points dans l'image et de leur profondeur, c.à.d. la distance entre le point dans le plan image et le point dans l'espace opérationnel. Cependant cette dernière donnée n'est pas connue dans le cadre de la méthode IBVS en mono-caméra. Pour l'approximation de la matrice d'interaction plusieurs calculs ont été proposés :

- À chaque itération la profondeur de chaque point est estimée et la matrice d'interaction calculée ($\widehat{\mathbf{L}}^+ \approx \mathbf{L}^+$);
- On peut aussi considérer simplement la profondeur des points dans l'état désiré [ECR90]. La matrice d'interaction devient ainsi constante ($\widehat{\mathbf{L}}^+ \approx \mathbf{L}_{e^*}^+$);
- Il est possible de prendre la moyenne des deux méthodes ($\widehat{\mathbf{L}}^+ \approx \frac{\mathbf{L}^+ + \mathbf{L}_{e^*}^+}{2}$). Cette dernière méthode introduite par [Mal04] a montré de meilleures performances en terme de trajectoire (plus lisse dans l'image et l'espace opérationnel) et de convergence (moins sensible aux minima locaux et domaine de convergence plus grand).

2.1.3 Stabilité

La stabilité constitue une des problématiques majeures des asservissements visuels. Dans ce document le terme stabilité regroupe les critères qui assurent la convergence du système à la position désirée, à savoir : l'absence de singularité de la matrice d'interaction, de minima globaux, de minima locaux, ainsi que la stabilité asymptotique. La stabilité dépend d'un grand nombre de facteurs, comme : le type d'information visuelle (2D ou 3D), le nombre et choix de primitives, l'approximation des modèles (paramètres de la caméra et géométrie du robot) et des singularités du robot.

Conditions de stabilité asymptotique L'analyse de la stabilité d'un système robotique à corps rigide en boucle fermée sur des informations visuelles a été étudiée dès le début par [SMB91] et [ECR90]. François Chaumette décrit une condition de stabilité asymptotique globale sous la forme d'une inégalité matricielle, pour une commande dans l'espace des vitesses opérationnelles

[CH06] et articulaires [CH07]. Nous présentons ici le deuxième cas, qui est le plus proche de notre problématique.

Tout d'abord pour assurer la convergence du système à l'état désiré \mathbf{s}^* , il faut s'assurer que le système bouclé ne présente qu'un unique point d'équilibre $\mathbf{e}^* = 0$ dans l'espace de travail. Cette condition est vérifiée si $\text{Ker}(\widehat{\mathbf{M}\mathbf{M}^+}) = \emptyset$, c'est à dire si le nombre de colonnes de $\widehat{\mathbf{M}\mathbf{M}^+}$ est égal à son rang. On parle alors de matrice de rang plein.

Puis grâce à la méthode de Lyapunov, comme le gain λ est défini positif, on peut définir la condition suffisante pour une stabilité asymptotique globale, à savoir :

$$\widehat{\mathbf{M}\mathbf{M}^+} > 0 \quad (7)$$

Si le nombre d'informations visuelles est égal au nombre de degrés de liberté de l'effecteur ($k = n$) et que les informations visuelles et la commande sont choisies telles que les matrices \mathbf{M} et $\widehat{\mathbf{M}^+}$ sont de rang plein, alors la condition de stabilité (7) est assurée si l'approximation de $\widehat{\mathbf{M}^+}$ n'est pas trop grossière [CH06]. Cela implique que les matrices $\mathbf{L}^c, \mathbb{T}_o$ et ${}^o\mathbf{J}$ ne soient pas "trop" éloignées des modèles réels. La stabilité dépend alors directement de l'étalement de la caméra et du modèle géométrique du robot. Des travaux montrent par expérimentation qu'une approximation, même parfois grossière, de ces matrices peut suffire [CHKA03]. Mais il est cependant difficile de définir une règle générale sur les erreurs de modélisation acceptables pour assurer une bonne convergence [MBR02].

Comparaison IBVS et PBVS Comme la stabilité dépend du nombre et type d'informations visuelles, il y a nécessairement une différence notable entre la stabilité de la méthode IBVS et PBVS. Pour comparer la stabilité et robustesse de ces deux techniques, nous nous plaçons dans le cas général d'une caméra embarquée sur l'effecteur, de mouvement à $n = 6$ degrés de liberté (ddl) et de primitives de types points.

IBVS Dans le cas d'informations de type 2D, trouver $k = 6$ informations visuelles conduisant au bon comportement du système bouclé est un des principaux problèmes [Chr08]. Par exemple si \mathbf{s} est construit à partir des coordonnées de 3 points dans l'image, la matrice d'interaction associée possède des singularités pour certaines configurations bien précises [MR93]. De plus il existe quatre positions de la caméra pour lesquelles $\mathbf{e} = 0$ [DRLR90]. Ces configurations sont appelées minimas globaux. Pour contourner ces problèmes, on utilise généralement plus de 3 points.

Cependant lorsque $k > n$ la condition de stabilité globale (7) ne peut pas être assurée car $\widehat{\mathbf{M}\mathbf{M}^+} \in \mathbf{R}^{k \times k}$ n'est pas de plein rang, mais au plus de rang n . Ceci implique la présence de minimas locaux. Le système peut alors converger vers une position qui n'est pas la solution désirée. Seule une stabilité asymptotique locale peut donc être obtenue pour la méthode IBVS. La condition de stabilité asymptotique locale pour $k > n$ autour du voisinage $e = 0$ peut être donnée par [CH06] :

$$\widehat{\mathbf{M}^+}\mathbf{M} > 0 \quad (8)$$

Seuls de petits déplacements autour de la position désirée assurent une convergence du système vers l'état désiré.

Une des caractéristiques fondamentales de la méthode IBVS est le fait que la fonction de tâche est définie dans le plan image. Ceci entraîne une trajectoire souvent optimisée dans l'image, mais

pas dans l'espace opérationnel. Elle est même parfois impossible à réaliser en pratique par le mécanisme. Cependant cette caractéristique permet une certaine robustesse comparée à celle de la méthode PBVS. Les approximations de la matrice \mathbf{M}^+ n'affectent pas la précision du système, car le calcul de l'erreur ne dépend pas de ces approximations. Mais ces dernières affectent tout de même la trajectoire.

PBVS Dans cette méthode, un des premiers problèmes réside dans le fait de reconstruire à partir d'une ou plusieurs images 2D la pose de l'objet d'intérêt. Dans le cas de l'utilisation d'une seule caméra, cette méthode implique souvent la connaissance du modèle 3D de cet objet en plus des paramètres intrinsèques et extrinsèques de la caméra.

Contrairement à la méthode IBVS, le choix des informations visuelles ne pose pas problème pour la méthode PBVS. En effet ce choix se porte souvent sur la pose de l'objet d'intérêt comprenant les coordonnées de l'origine du repère et d'un vecteur $\mathbf{u}\theta$ caractérisant son orientation. Suite à ce choix, la matrice d'interaction possède l'avantage d'être de plein rang $k = 6$, sauf dans un cas bien identifié où elle est singulière : lorsque θ est multiple de 2π .

Cependant l'analyse de la stabilité soulève des difficultés. Tout d'abord la matrice d'interaction dépend de paramètres de pose calculés. Et il est difficile d'obtenir une expression analytique de ces paramètres de pose estimés en fonction des erreurs d'étalonnage. Il est donc difficile de connaître les limites acceptables aux erreurs d'étalonnage caméra.

Un autre inconvénient de la méthode PBVS pour la stabilité vient du fait que l'espace des tâches est défini dans l'espace opérationnel et que le calcul de l'erreur lui même dépend des paramètres de pose estimés. De petites déviations de mesure dans l'image peuvent conduire à une grande déviation de la pose de l'objet, ce qui peut impacter significativement la précision et la stabilité du système. La méthode est donc plus sensible au bruit dans l'image.

Un des avantages de la commande PBVS comparé à l'IBVS réside dans sa trajectoire dont l'orientation et la translation peuvent facilement être découplées. La trajectoire est souvent optimisée dans l'espace opérationnel mais pas dans l'image. L'objet d'intérêt peut donc quitter le champs de vision de la caméra.

2.2 Asservissement visuel des robots parallèles

2.2.1 Intérêt de la vision pour les robots parallèles

Un mécanisme parallèle peut être défini comme un mécanisme *en chaîne cinématique fermée, constituée d'un organe terminal à n degrés de liberté et d'une base fixe, reliés entre eux par au moins deux chaînes cinématiques indépendantes, la motorisation s'effectuant par n actionneurs simples* [Mer97]. Les avantages des robots parallèles comparés aux sériels sont notamment leur capacité de charge et leur performance dynamique. Ils ont donc rapidement intéressé le monde industriel. Cependant leur commande peut devenir difficile en raison de la complexité de l'architecture, du fort couplage des liaisons et de la présence de singularité. L'un des problèmes majeurs réside dans la résolution du modèle géométrique direct. Cette résolution possède souvent plusieurs solutions \mathbf{p} pour une seule configuration articulaire \mathbf{q} .

Pour contourner ce problème une solution consiste à intégrer des capteurs proprioceptifs sur les liaisons passives. Les mesures supplémentaires fournies par ces capteurs permettent d'obtenir une solution unique du modèle géométrique direct [BA00]. Cependant, l'implantation des capteurs dans les liaisons passives n'est pas toujours réalisable et doit être prise en compte dès la conception du mécanisme.

D'autres travaux proposent d'utiliser des capteurs extéroceptifs tels que les caméras qui donnent des informations externes à la structure mécanique du robot. L'utilisation de ces capteurs permet une identification géométrique précise du robot [Ren03]. Ils permettent également une commande basée sur un modèle géométrique du robot simplifié [And05], voire une commande sans modèle géométrique, dans le cas classique de l'observation directe de l'organe terminal. Voir [DAMM06] pour une plus grande argumentation sur les avantages de l'utilisation de l'asservissement visuel pour les robots parallèles.

Choix de l'observation Pour la suite du document nous nous placerons dans le cas d'une ou plusieurs caméras déportées. Pour réaliser l'asservissement visuel de l'effecteur de robot parallèle, la solution la plus classique est d'observer directement l'effecteur. Mais dans certains cas il peut s'avérer être difficile à identifier. Cela peut être dû simplement à la difficulté d'extraire des primitives géométriques sur l'effecteur ou suite à l'obstruction de ces dernières, par exemple à cause de la prise d'un outil ou d'un objet.

Les robots parallèles possèdent l'avantage d'être souvent équipés de jambes aux formes allongées et rectilignes. Ces jambes sont donc faciles à extraire dans une image et à paramétrer par des primitives géométriques. De plus la pose de l'effecteur d'un robot parallèle peut être définie grâce à la position et l'orientation de ses jambes. Il a donc été envisagé d'observer plutôt les jambes pour contrôler la pose de l'effecteur. De plus l'observation des jambes donne un plus grand nombre de points observés dans l'image et donc peut permettre une meilleure précision de l'estimation de pose à partir de l'image.

2.2.2 Représentation géométrique des jambes

Si le choix de l'observation se penche vers les jambes du robot, il faut définir un moyen de représenter géométriquement ces jambes dans l'espace cartésien et dans le plan image. Si ces jambes sont cylindriques, les manières les plus intuitives sont l'utilisation de l'axe central du cylindre [And05] ou les contours du cylindre [ADM07]. Les primitives géométriques extraites seront donc de type droite. Il existe un certain nombre de travaux sur l'asservissement visuel utilisant

des primitives de type droite [ECR90] [And05] [AEH02] [VBPC14] [MH01] [MBR02]

Coordonnées de Plücker Un paramétrage des droites utilisées [And05] pour une commande référencée vision, est les coordonnées de Plücker bi-normalisées introduites par [AEH02]. Suivant ce paramétrage, une droite \mathcal{L} est définie dans l'espace par :

$$\mathcal{L} \equiv (\underline{\mathbf{u}}, \underline{\mathbf{n}}, n) \quad (9)$$

où $\underline{\mathbf{u}}$ est le vecteur unitaire donnant l'orientation spatiale de la droite, $\underline{\mathbf{n}}$ est un vecteur unitaire perpendiculaire au plan d'interprétation passant par le centre de projection de la caméra et la droite \mathcal{L} et enfin n est la profondeur de la droite, c.à.d. la plus courte distance entre la droite et le centre de projection de la caméra. $\underline{\mathbf{n}}$ est également appelé le vecteur moment de Plücker.

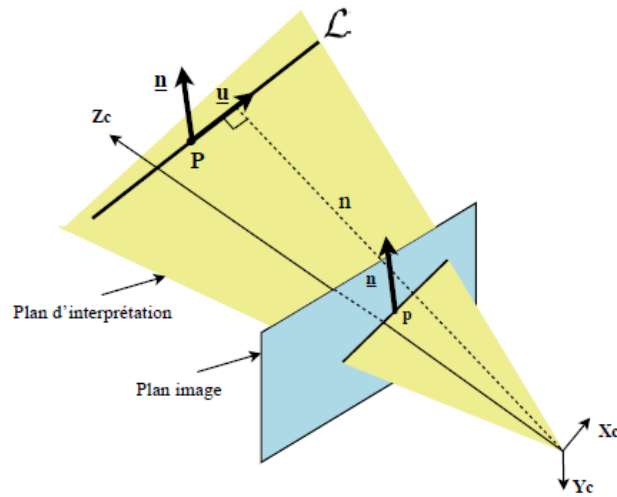


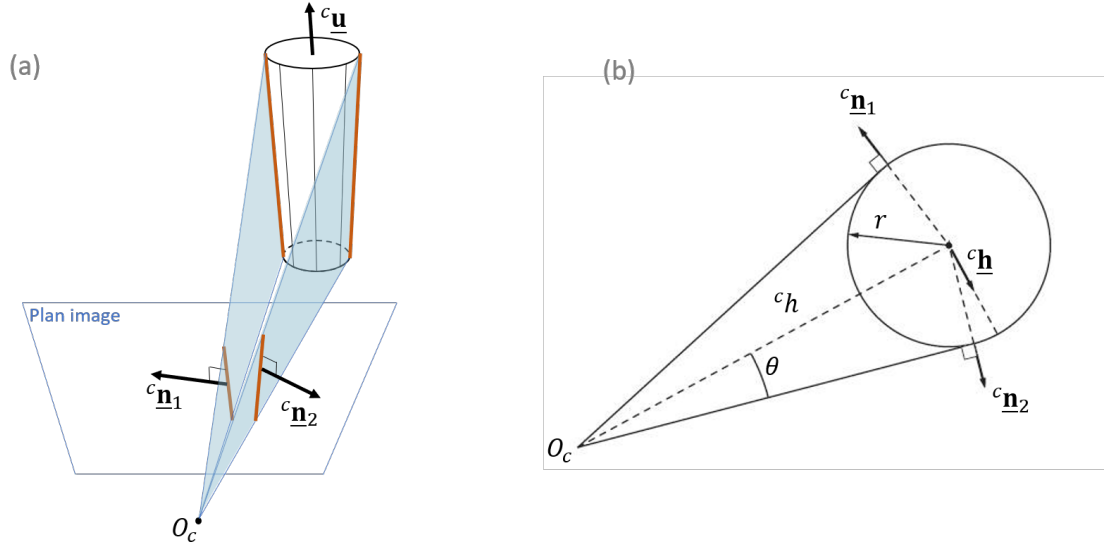
FIGURE 7 – Représentation géométrique d'une droite 3D par les coordonnées de Plücker

Cette paramétrisation provient des coordonnées de Plücker normalisées [Plu65] définies par :

$$\mathcal{L} \equiv (\underline{\mathbf{u}}, \underline{\mathbf{n}}) \quad (10)$$

avec $\underline{\mathbf{n}} = n\underline{\mathbf{n}}$. D'après [AEH02] l'utilisation des coordonnées de Plücker bi-normalisées, présente des avantages d'un point de vue commande référencée vision. Un d'entre eux est le fait que la profondeur de la droite devient une variable indépendante. Or cette variable ne peut être connue dans le cas de la projection d'une droite 3D dans le plan image. En effet la projection d'une droite 3D dans l'espace 2D permet de déterminer le moment de Plücker normalisé $\underline{\mathbf{n}}$. Cependant dans le cas de géométrie particulière tel que le cylindre, la projection des contours d'un cylindre dans l'image permet de reconstruire les coordonnées de Plücker complètes de l'axe central.

Reconstruction de l'axe central d'un cylindre à partir de ses contours Sous l'hypothèse que l'axe de cylindre ne coïncide pas avec l'axe de la caméra, la projection d'un cylindre dans le plan image peut être définie par deux droites, les limbes ou contours du cylindre. Chaque



9

FIGURE 8 – **Projection du cylindre** (a) : sur le plan image; (b) : sur un plan perpendiculaire à l'axe central du cylindre; O_c correspond au centre de projection de la caméra.

droite dans l'image peut être entièrement définie par son moment de Plücker (${}^c\mathbf{n}$) [AEH02]. À partir de la position de ces deux droites dans l'image (${}^c\mathbf{n}_1, {}^c\mathbf{n}_2$), et de la connaissance du rayon r du cylindre, il est possible de déterminer (${}^c\mathbf{u}, {}^c h {}^c\mathbf{h}$) les coordonnées de Plücker de l'axe du cylindre, par des relations de géométrie projective (cf. Figure 8). La méthode détaillée est décrite dans [ADM07] et [VBPC14]. Les relations données sont :

$$\begin{cases} {}^c\mathbf{u} = \frac{{}^c\mathbf{n}_1 \times {}^c\mathbf{n}_2}{\|{}^c\mathbf{n}_1 \times {}^c\mathbf{n}_2\|} \\ {}^c h {}^c\mathbf{h} = \frac{r}{\sin(\theta)} \left(\frac{{}^c\mathbf{n}_1 + {}^c\mathbf{n}_2}{\|{}^c\mathbf{n}_1 + {}^c\mathbf{n}_2\|} \times {}^c\mathbf{u} \right) \end{cases} \quad (11)$$

À l'inverse la connaissance des coordonnées de Plücker de l'axe central du cylindre permet de reconstruire les moments de Plücker des contours, grâce aux équations suivantes :

$$\begin{cases} {}^c\mathbf{n}_1 = -\cos(\theta){}^c\mathbf{h} - \sin(\theta){}^c\mathbf{u} \times {}^c\mathbf{h} \\ {}^c\mathbf{n}_2 = +\cos(\theta){}^c\mathbf{h} - \sin(\theta){}^c\mathbf{u} \times {}^c\mathbf{h} \end{cases} \quad (12)$$

avec $\cos(\theta) = \frac{\sqrt{{}^c h^2 - r^2}}{{}^c h}$ et $\sin(\theta) = \frac{r}{{}^c h}$.

2.2.3 Commandes basées sur l'observation des jambes

Commande basée sur l'orientation des jambes Dans [And05], une approche utilisant la vision est proposée et appliquée à un mécanisme parallèle. La vision est ici utilisée pour mesurer l'orientation des jambes d'une plateforme de Gough-Stewart. En utilisant le concept de redondance métrologique, cette méthode permet l'asservissement de la pose de la plateforme. Dans cette approche Nicolas Andreff utilise les coordonnées de Plücker bi-normalisées (9). Le contrôleur utilise l'orientation des jambes (\mathbf{u}) comme information visuelle. La position de chaque

jambe (droite) est connue et fixe en un point : à la base du robot.

Cependant le contrôleur nécessite la connaissance d'une matrice jacobienne liant la pose de l'effecteur dans l'espace cartésien à la direction des jambes. Cette jacobienne peut avoir des singularités qui s'additionnent à celles de l'architecture mécanique, comme pour le robot 5 barres [VBPC14]. Pour trouver ces singularités [Ros15] propose le concept de robot caché. Ce dernier est un robot virtuel, dont la cinématique représente le passage de l'espace des directions des jambes à l'espace de la pose de l'effecteur. Les singularités de ce robot caché correspondent aux singularités du passage entre ces deux espaces.

Commande basée sur l'orientation et la position des jambes [VBPC14] reprend le travail réalisé dans [And05] et compare sa méthode avec une nouvelle approche qui vise également l'asservissement visuel de robots parallèles en utilisant l'observation des jambes. La nouvelle méthode se différencie par la nature des informations visuelles qui ne sont plus simplement la direction des jambes comme [And05], mais les coordonnées de Plücker normalisées (10). Ce paramétrage des jambes prend en compte la position de la droite par rapport au repère caméra en plus de son orientation. [VBPC14] présente les avantages de sa nouvelle méthode comparée à la précédente, à savoir :

- Elle peut être utilisée dans des cas particuliers d'architecture mécanique, où l'information de la direction des jambes ne suffit pas [AM06] ;
- L'estimation de la pose de l'effecteur ne dépend pas de paramètres géométriques estimés, excepté le diamètre des jambes ;
- Les singularités du contrôleur coïncident avec celles du mécanisme. Aucune nouvelle position singulière n'est ajoutée.

Pour reconstruire l'information de profondeur des droites [VBPC14] n'utilise qu'une caméra et se base sur la connaissance du diamètre réel des bras cylindriques et leur mesure dans l'image. Le calcul de l'erreur dépend donc de l'approximation de ce diamètre.

Commande basée sur le contour des jambes Une autre approche [ADM07] a été envisagée reprenant le travail de [And05]. Au lieu d'utiliser des informations visuelles 3D reconstruites (orientation des jambes), les informations visuelles 2D sont directement utilisées : le contour des jambes. La nouvelle commande bénéficie des avantages de la méthode basée sur l'orientation : moins de paramètres à calibrer et peu de dépendance à la valeur des articulations, ainsi que la possibilité d'asservir l'effecteur même si celui-ci n'est pas visible. De plus il est prouvé en pratique une meilleure robustesse, dû au fait que certains paramètres calibrés (comme les paramètres de la caméra) et les erreurs numériques sont maintenant simplement présents dans la matrice d'interaction et n'apparaissent plus dans le calcul de la fonction de tâche \mathbf{e} . Cette méthode utilise un modèle cinématique du robot basé sur l'orientation des jambes.

2.2.4 Commandes basées sur l'observation de l'effecteur

Comme nous l'avons vu, des contrôleurs ont été étudiés pour des robots parallèles en utilisant l'orientation et la position des jambes mais la fonction de tâche de ces méthodes dépend toujours en partie de certains paramètres géométriques calibrés. Néanmoins, dans le cas d'un système possédant des flexibilités ou des variations de dimensions, ces commandes basées sur le modèle ne peuvent pas assurer une précision de l'effecteur dans l'espace opérationnel. Si une commande référencée capteurs fait à un moment donné appel au modèle cinématique inverse du robot pour

passer de l'espace opérationnel à l'espace articulaire, il est néanmoins possible de se débarrasser de la dépendance du modèle du robot dans l'estimation de la pose de l'effecteur. Il suffit d'utiliser la méthode plus classique de l'asservissement visuel (Figure 4) où \mathbf{s} est obtenue par observation de l'effecteur. Grâce à l'utilisation de capteurs extéroceptifs, la pose de l'effecteur peut être estimée dans l'espace opérationnel sans connaissance du modèle du robot. Ainsi la précision ne dépend plus que de l'identification des paramètres de la caméra et de l'extraction des primitives dans l'image.

Types d'informations visuelles Si l'observation directe de l'effecteur du robot semble être la solution la plus simple et intuitive, le choix des informations visuelles l'est moins. Est-il préférable d'utiliser des informations de type 2D ou 3D pour un robot parallèle ?

D'après [DAMM06] parmi les techniques d'asservissement visuel, l'estimation de la pose 3D (méthode PBVS) est la commande la plus simple et robuste pour une commande cinématique des robots parallèles. Si la trajectoire dans l'espace opérationnel n'est pas contrôlée (ce qui est le cas pour la méthode IBVS), rien n'empêche la commande d'amener le robot dans des positions singulières. Or un robot parallèle comme le robot cinq barres, possède un certain nombre de positions singulières notamment au centre de son espace de travail opérationnel. De plus contrairement au robot série, l'état d'un système parallèle ne peut être entièrement défini par la valeur de ses variables articulaires, mais nécessite la valeur de pose de l'effecteur. Or le modèle cinématique inverse utilisé pour passer de l'espace des vitesses opérationnelles aux articulaires, nécessite la connaissance de l'état du système. La pose de l'effecteur doit donc dans tous les cas être estimée pour le calcul de ce modèle inverse. Un autre argument en faveur de l'asservissement visuel avec l'information de pose 3D est le fait que des méthodes simples [DD92] et précises [LVD98] d'estimation de pose ont été créées. L'estimation de cette pose peut être considérée suffisamment précise pour être utilisée comme l'information visuelle à réguler.

Il faut tout de même rappeler que la précision et la stabilité de la méthode PBVS sont impactées par les erreurs d'estimation de pose et sont plus sensibles au bruit dans l'image que celles de la méthode IBVS. Si les critères de performances recherchés sont la précision et la stabilité, la méthode IBVS pourra s'avérer plus robuste en mono-caméra.

2.3 Techniques d'asservissement visuel avancées

2.3.1 Amélioration des performances

Depuis les débuts et les principes fondamentaux des asservissements visuels, des travaux ont cherché de nouvelles techniques pour résoudre certaines problématiques des méthodes classiques et améliorer la performance des commandes référencées vision. Les principales méthodes sont répertoriées dans [CH07]. Nous citerons en exemple celles qui nous semblent importantes de mentionner.

Découplage et linéarité Une commande des mouvements découplée et linéaire sont souvent des caractéristiques recherchées par les asservissements surtout lorsque le robot possède 6 ddl. Le premier exemple connu est l'asservissement visuel hybride ou 2D 1/2 [MCB99]. Cette commande utilise un contrôleur basé image (IBVS) pour les mouvements de translation et un contrôleur basé objet (PBSV) pour les mouvements de rotation. Un autre exemple réside dans le choix particulier d'informations visuelles exprimées dans l'image [CH01]. On peut citer les méthodes basées sur des moments dans l'image [Cha04] qui peuvent permettre les mêmes propriétés de découplage et linéarité. Ces moments peuvent être issus d'objet plan de n'importe quelle forme ou d'une collection de points [TC05].

Contraintes de mouvements du robot D'autres méthodes permettent d'améliorer la performance de l'asservissement visuel, par une conception de contrôleur particulière qui intègre les contraintes de mouvement du robot. On peut citer l'utilisation de la commande linéaire quadratique (LQ) gaussienne dans [HK93] [PKK93] et la prise en compte des limites articulaires et des singularités dans [NK93]. L'utilisation de coefficients de pondération particuliers permet de prendre en compte certaines contraintes comme le maintien des informations visuelles dans un certain espace [KC14].

Les techniques de commande multi-capteurs offrent également la possibilité de traiter de nombreuses contraintes.

2.3.2 Commande référencée multi-capteurs

La commande référencée multi-capteurs repose sur la régulation à zéro de m erreurs (ou fonctions de tâche). Chaque erreur \mathbf{e}_i est définie comme la différence entre un signal \mathbf{s}_i délivré par un capteur et un signal de référence \mathbf{s}_i^* :

$$\mathbf{e}_i = \mathbf{s}_i - \mathbf{s}_i^* \quad \forall i \in [1, m]$$

Les signaux peuvent être de nature multiple : informations visuelles 2D, 3D, nombre pixel, angle...ect Le terme $\dot{\mathbf{e}}_i^*$ qui correspond à la variation de l'erreur \mathbf{e}_i souhaitée est aussi appelé tâche. Dans les nombreuses possibilités qui s'offrent dans l'élaboration d'une commande regroupant plusieurs tâches, [Ker11] distingue 4 grandes classes de méthodes pour la commande référencée multi-capteurs.

1. La commande partitionnée qui consiste à assigner à chaque capteur la commande de certains degrés de liberté du système.
2. L'approche hiérarchique qui impose une hiérarchie entre les tâches. Dans ce cas, certains capteurs sont prioritaires sur d'autres. La difficulté est alors de permettre aux tâches secondaires de pouvoir se réaliser au mieux sous la contrainte de la tâche prioritaire.

3. La commande commutative, où une seule tâche est appliquée à la fois et des critères sont définis pour passer d'une tâche à une autre.

4. La commande hybride qui consiste à réaliser un compromis entre les différentes tâches.

Les deux premiers types de commande sont intéressants lorsque le nombre de ddl est élevé. Pour notre cas, on s'intéressera plus particulièrement aux deux derniers types commandes :

La commande commutative a été utilisée dans quelques applications d'asservissement visuel pour passer d'une commande PBVS à une commande IBVS en fonction de certaines conditions. Par exemple [CHPV04] réalise une commande PBVS tant que l'objet ne risque pas de sortir du champ de vue. [GH07], lui, passe d'informations 2D à 3D en fonction de l'espace où l'erreur est maximum. Si les méthodes commutatives peuvent sembler simples, les conditions permettant le passage d'une tâche à une autre ne sont pas toujours évidentes à estimer.

Les commandes hybrides régulent l'ensemble des informations capteur à la fois, sans hiérarchie stricte dans la structure de la loi de commande. Les méthodes hybrides diffèrent par la façon de combiner les mesures ou de réaliser le compromis entre elles. On peut distinguer dans la littérature, différentes manières de réaliser une commande hybride.

Une première classe de commande hybride consiste à additionner les tâches dans l'espace des vitesses. Cela revient à fusionner les contributions des différentes lois de commande par l'addition comme dans [MMC01].

$$\tau = h_1 \mathbf{L}_1^+ \dot{\mathbf{e}}_1^* + \dots + h_m \mathbf{L}_m^+ \dot{\mathbf{e}}_m^* \quad (13)$$

$$\tau = (\mathbf{L}_1^+ \quad \dots \quad \mathbf{L}_m^+) \mathbf{H} \begin{pmatrix} \dot{\mathbf{e}}_1^* \\ \vdots \\ \dot{\mathbf{e}}_m^* \end{pmatrix} \quad (14)$$

avec $\sum_{i=1}^m h_i = 1$ les coefficients de pondération qui équilibrent les contributions des différents capteurs. Ces coefficients peuvent être regroupés dans une matrice de pondération diagonale \mathbf{H} tel que :

$$\mathbf{H} = \begin{pmatrix} h_1 \mathbb{I}_{k_1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & h_m \mathbb{I}_{k_m} \end{pmatrix}$$

Une deuxième approche est fondée sur le jacobien étendu \mathbf{L} [Nak90] obtenue par concaténation des matrices d'interactions de chaque tâche $\mathbf{L} = \begin{pmatrix} \mathbf{L}_1 \\ \vdots \\ \mathbf{L}_m \end{pmatrix}$. Le jacobien étendu regroupe l'ensemble des apports des données dans une même matrice pseudo-inverse \mathbf{L}^+ tel que :

$$\tau = \mathbf{L}^+ \dot{\mathbf{e}}^* \quad (15)$$

Comme précédemment il est également possible de pondérer la contribution de chaque tâche à l'aide d'une matrice de pondération \mathbf{H} . Pour obtenir les meilleures performances, il est intéressant de considérer la matrice de pondération dans le jacobien inverse. La loi de commande devient alors :

$$\tau = (\mathbf{HL})^+ \mathbf{H} \dot{\mathbf{e}}^* \quad (16)$$

avec $(\mathbf{HL})^+$ appelé l'inverse généralisée.

Parmi les commandes hybrides avec utilisation de l'inverse généralisée, une approche, proposée par [Ker11] repose sur des tâches à dimension variable. Avec cette méthode les poids variables traduisent l'intérêt d'utiliser une information visuelle. Un indice de confiance du traitement d'image est calculé pour chaque information visuelle. Les informations sûres ont un poids de 1 et les informations aberrantes ont un poids nul.

La commande référencée multi-capteurs offre donc une grande liberté de commande. Elle permet de s'adapter à de nombreuses problématiques et applications différentes. Elle semble bien adaptée pour améliorer les performances d'asservissement visuel classique.

3 Différents contrôleurs envisagés

L'étude des techniques d'asservissement visuel a permis une meilleure connaissance du domaine et a abouti à des choix de correcteur pour le projet.

Dans la suite nous présenterons et justifierons les différents types d'informations visuelles \mathbf{s} et matrices d'interactions \mathbf{L} retenues pour notre application. Nous nous intéressons à la position de l'effecteur et non à son orientation. Seule la vitesse relative en translation de l'effecteur par rapport à la caméra, est donc considérée pour $\boldsymbol{\tau}$.

Les correcteurs envisagés peuvent se classer selon les informations visuelles utilisées, en quatre groupes distincts :

- contrôleur basé observation effecteur
- contrôleur basé contours des jambes
- contrôleur basé reconstruction
- contrôleur hybride

Nous les présentons en détail ci-dessous.

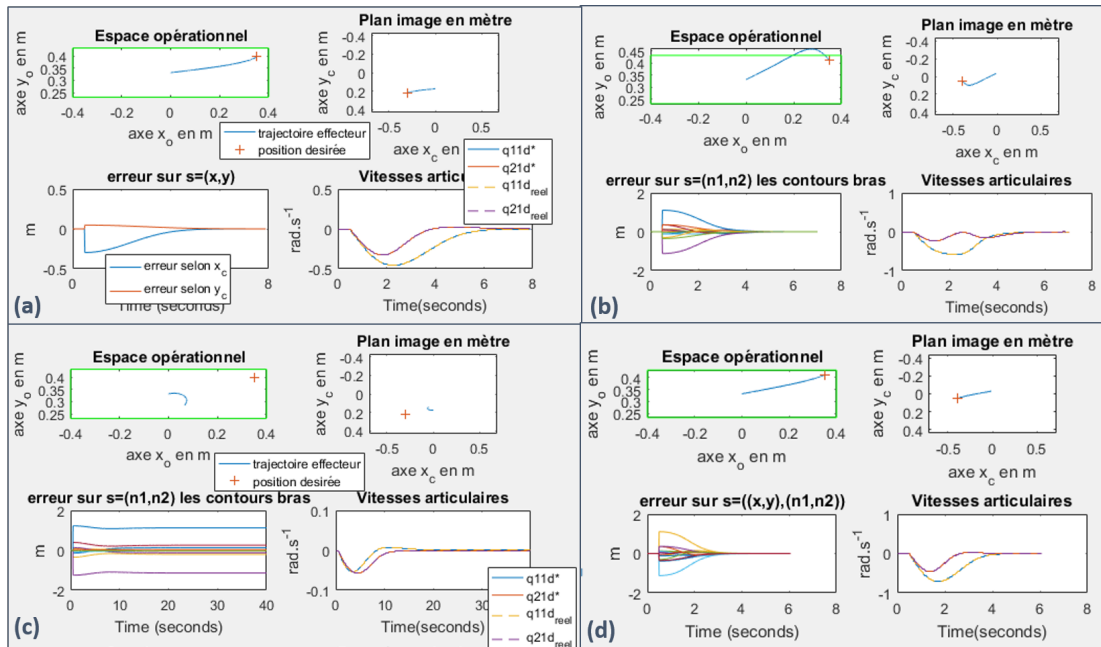


FIGURE 9 – Exemple de la réponse du système pour différents contrôleurs (a) : contrôleur basé observation effecteur ; (b) : contrôleur basé contours \mathbf{N}_u ; (c) : contrôleur basé contours \mathbf{N}_{uh} ; (d) : contrôleur hybride basé fusion ;

3.1 Contrôleur basé observation de l'effecteur

Justification du choix de \mathbf{s} : La commande référencée vision par observation de l'effecteur en mono-caméra est la méthode la plus classique et simple des asservissements visuels. Le mécanisme parallèle à cinq barres ne possède que deux degrés de liberté (ddl) en translation. Deux informations visuelles $\mathbf{s} \in \mathbb{R}^k$, $k = 2$ sont donc suffisantes.

Dans notre cas, la position de l'effecteur peut être déterminée directement par extraction du centre du cercle de la liaison pivot (Figure 6). Ainsi il n'y a pas la nécessité de rajouter un marqueur visuel tel qu'une mire.

Concernant le type d'informations visuelles \mathbf{s} , la méthode PBVS possède l'avantage de permettre un contrôle des trajectoires dans l'espace cartésien et de découpler les mouvements de rotation et translation du robot. Or dans notre cas, n'ayant que 2ddl en translation, un contrôle de la trajectoire de l'effecteur dans l'image est suffisant. Une trajectoire rectiligne dans le plan image, reste rectiligne dans l'espace cartésien. De plus, la méthode IBVS est plus robuste aux bruits dans l'image [CH06]. Un autre argument, en faveur de la méthode IBVS dans notre cas est le fait que l'estimation de la pose d'un objet nécessite la connaissance du modèle de cet objet en mono-caméra. Si aucun marqueur visuel n'est ajouté à la structure du robot, cela implique la connaissance précise d'une partie du modèle du robot. Nous avons donc choisi de privilégier les informations visuelles 2D. Avec $\mathbf{s} = \mathbf{s}_p = (x, y)$ la position du centre du cercle dans l'image, la matrice d'interaction associée à ce point que nous appellerons \mathbf{P} prend la forme :

$$\mathbf{P} = \begin{bmatrix} \frac{-1}{Z} & 0 \\ 0 & \frac{-1}{Z} \end{bmatrix} \quad (17)$$

Z correspond à la profondeur (axe z) du point 3D observé par rapport à la caméra. Cette variable peut être estimée à chaque itération $Z = Z^i$ ou fixée à sa valeur finale $Z = Z^*$.

Avantages et inconvénients du contrôleur : Le gros avantage de cette matrice d'interaction est sa forme diagonale. Ceci entraîne une commande en vitesse linéaire et découplée (cf. Figure 9.(a)). De plus il est très simple d'assurer la stabilité asymptotique globale dans ce cas. Nous avons un nombre k d'informations visuelles égal au nombre de degrés de liberté ($k = n = 2$), donc pas de problèmes de minima locaux. Il y a seulement une configuration du robot possible pour une position du point de l'effecteur dans l'image, donc pas de minima globaux. Et enfin la matrice et sa pseudo inverse (qui est simplement son inverse) sont de rang plein tant que $Z \neq 0$. Or ceci correspond à une position absurde de l'effecteur dans la caméra.

3.2 Contrôleur basé contours des jambes

Justification du choix de \mathbf{s} : L'observation des bras revient à extraire des lignes dans l'image. Ces lignes sont construites à partir de plus de points dans l'image que le centre du cercle de l'effecteur, du fait de leur dimension. Leur position dans l'image est donc moins sensible au bruit, et permet d'atteindre une résolution sub-pixellique [KBBP16]. Cependant pour ne pas perdre de vue les bras, la caméra doit se situer à une plus grande distance de l'espace de travail que pour l'observation de l'effecteur.

Pour assurer la précision, il faut choisir des informations visuelles qui dépendent le moins possible des paramètres géométriques variables. Or les asservissements visuels basés sur l'observation des bras tel que [VBPC14] [And05] reconstruisent en 3D l'orientation et la position des bras pour définir \mathbf{s} , et donc exploitent la connaissance de paramètres géométriques (tel que le diamètre des bras ou la position d'un point fixe sur l'axe des cylindres). Or dans notre cas, le diamètre est potentiellement variable, voir incertain car la section des avant-bras de notre robot n'est pas tout à fait circulaire mais elliptique. Nous privilégierons donc les droites issues des contours des bras

comme informations visuelles $\mathbf{s} = \mathbf{s}_n = \begin{pmatrix} \mathbf{n}_{11} \\ \mathbf{n}_{12} \\ \mathbf{n}_{21} \\ \mathbf{n}_{22} \end{pmatrix}$ avec \mathbf{n}_{ij} le vecteur moment de plücker unitaire

(9) définissant le contour j de la jambe i dans l'image. La construction de ces vecteurs est décrit pour notre cas en Annexe 6.1.1. Les matrices d'interactions \mathbf{N} associées aux contours des bras du robot prennent la forme générale :

$$\dot{\mathbf{s}}_{\mathbf{n}} = \mathbf{N}\boldsymbol{\tau}$$

Nous avons retenu deux matrices d'interactions différentes utilisant les contours $\mathbf{s}_{\mathbf{n}}$ qui diffèrent par leur construction et leur comportement que nous présentons ci dessous.

3.2.1 Matrice d'interaction basée direction des bras

Cette méthode basée IBVS a été introduite dans [ADM07]. Cependant cette dernière n'est pas directement utilisable dans notre cas. En effet la matrice d'interaction se base sur les coordonnées d'un point de l'axe de chaque cylindre observé, point qui est fixe par rapport à la base du robot. Or la structure du mécanisme à cinq barres ne possède pas de point fixe sur l'axe des avant-bras. [Dal07] présente dans sa thèse un modèle de matrice d'interaction, associée aux contours des jambes, générique pour différentes architectures de robot parallèle (cf. Annexe 6.1.3). Nous avons construit cette matrice dans le cas du robot parallèle cinq barres et nous l'appellerons $\mathbf{N}_{\mathbf{u}}$. Sa construction nécessite la connaissance d'un point 3D A_i sur l'axe de chaque cylindre i . De plus, $\mathbf{N}_{\mathbf{u}}$ dépend d'une autre matrice d'interaction $\mathbf{M}_{\mathbf{u}}$ associée aux directions des jambes \mathbf{u} tel que $\dot{\mathbf{u}}_i = \mathbf{M}_{\mathbf{u}}\boldsymbol{\tau}$. Or pour le robot parallèle cinq barres cette matrice $\mathbf{M}_{\mathbf{u}}$ possède des singularités supplémentaires aux singularités du mécanisme, comme le montre [VBPC14]. Une autre matrice d'interaction associée aux contours des jambes a donc été imaginée.

3.2.2 Matrice d'interaction basée direction et position des bras

Une matrice d'interaction basée sur la direction et position des bras a été réalisée par Lila Kaci [Kac18]. Elle sera nommée $\mathbf{N}_{\mathbf{uh}}$ dans le reste du document et est définie plus en détail en Annexe 6.1.4. Cette matrice dépend seulement de la connaissance du diamètre r_i de chaque cylindre i en plus des contours dans l'image. De plus, elle n'est pas construite avec $\mathbf{M}_{\mathbf{u}}$ mais avec $\mathbf{M}_{\mathbf{uh}}$ la matrice d'interaction associée aux directions \mathbf{u} et moments \mathbf{h} des axes des jambes tel que $\begin{bmatrix} \dot{\mathbf{u}} \\ \dot{\mathbf{h}} \end{bmatrix} = \mathbf{M}_{\mathbf{uh}}\boldsymbol{\tau}$. [VBPC14] a montré que cette matrice $\mathbf{M}_{\mathbf{uh}}$ possède les mêmes singularités que les singularités du mécanisme cinq barres. Au cours de sa thèse Lila Kaci a vérifié que dans l'espace de travail défini pour RobEcolo, $\mathbf{N}_{\mathbf{uh}}$ ne possédait aucune singularité.

Avantages et inconvénients des contrôleurs : Les avantages de cette matrice $\mathbf{N}_{\mathbf{uh}}$ comparé à $\mathbf{N}_{\mathbf{u}}$ est le fait qu'elle ne nécessite que la connaissance du diamètre r des cylindres en plus de leurs contours. Si les singularités de $\mathbf{N}_{\mathbf{uh}}$ ont été vérifiées en amont, ni les singularités de $\mathbf{N}_{\mathbf{u}}$, ni les minima locaux des contrôleurs basés $\mathbf{N}_{\mathbf{u}}$ et $\mathbf{N}_{\mathbf{uh}}$ n'ont été vérifiés pour notre application. Comme la détermination analytique des singularités et minima locaux de ces contrôleurs est complexe, nous nous contenterons dans ce stage de vérifier la stabilité des contrôleurs par la simulation.

Grâce à cette dernière, nous avons remarqué que la trajectoire des deux contrôleurs basés contours n'est pas rectiligne et peut sortir de l'espace de travail (cf. Figure 9.(b)). De plus, des minima locaux dans l'espace de travail de RobEcolo (cf. Figure 9.(c)) ont été découverts avec le contrôleur $\mathbf{N}_{\mathbf{uh}}$. Ces minima locaux peuvent être évités si la position de départ n'est pas trop éloignée de la position d'arrivée. Nous exploiterons les ressources des commandes hybrides (Section 3.4) pour éviter ces minima locaux.

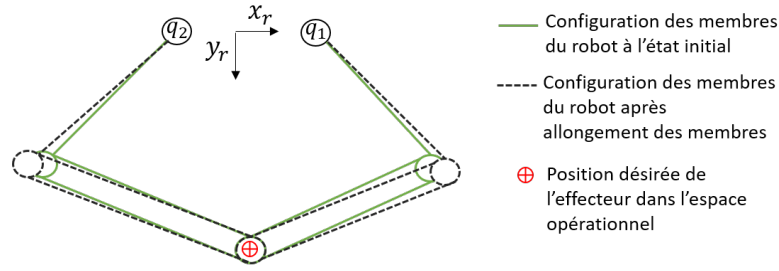


FIGURE 10 – Schéma de l'influence des variations dimensionnelles sur la configuration des membres du robot pour une même position de l'effecteur

3.3 Contrôleur basé reconstruction

Justification du choix de \mathbf{s} : Même si les contours des bras sont obtenus directement depuis l'image, les informations visuelles dépendent toujours en partie des variations dimensionnelles. Le schéma du robot cinq barres Figure 10 illustre cette dépendance. Si il se produit des variations de longueur ou de diamètre des membres après l'enregistrement de la position désirée dans l'image (état initial), le \mathbf{s}^* n'est plus possible à atteindre parfaitement. Nous analyserons par la simulation si ce phénomène impacte significativement la précision, dans la Section 5.3. Cependant une autre manière de faire peut être envisagé pour toujours profiter de la précision sub-pixellique de l'observation des jambes. Il est possible de reconstruire la position de l'effecteur dans l'image à partir des contours et d'utiliser cette position reconstruite comme \mathbf{s} . La position reconstruite sera nommée \mathbf{s}_r . Cette reconstruction est décrite en annexe.

Avantages et inconvénients du contrôleur : Cette méthode présente un gros avantage par rapport aux contrôleurs basés contours :

- Comme $\mathbf{s} = (x, y)$ la matrice d'interaction associée est la matrice (17) qui présente de nombreux avantages pour sa stabilité et la trajectoire rectiligne de l'effecteur.

Mais elle présente aussi quelques inconvénients :

- La reconstruction de \mathbf{s} n'est pas complètement indépendante des paramètres géométriques supposés variables, car elle nécessite la connaissance du diamètre r .
- La reconstruction de \mathbf{s} possède le désavantage des méthodes PBVS, à savoir être plus sensible aux approximations des calculs numériques et au bruit dans l'image.

3.4 Contrôleur hybride

La fusion de plusieurs types d'informations visuelles peut permettre d'améliorer la commande. Dans notre cas l'utilisation du point de l'effecteur dans l'image est très intéressante pour la stabilité et la trajectoire sur de longues distances. D'un autre côté, l'utilisation des contours des bras peut être très intéressante lorsque l'effecteur est proche de la position désirée, pour une meilleure précision et robustesse au bruit.

Les données utilisées seront donc $\mathbf{s} = \begin{pmatrix} \mathbf{s}_p \\ \mathbf{s}_n \end{pmatrix}$. Pour pouvoir pondérer l'exploitation des informations visuelles issues de l'observation de l'effecteur $\mathbf{s}_p \in \mathbb{R}^{k_p \times 1}$ ou des jambes $\mathbf{s}_n \in \mathbb{R}^{k_n \times 1}$ nous utiliserons une matrice de pondération qui prend la forme diagonale suivante :

$$\mathbf{H} = \begin{pmatrix} h_p \mathbb{I}_{k_p} & \mathbf{0}_{k_p, k_n} \\ \mathbf{0}_{k_n, k_p} & h_n \mathbb{I}_{k_n} \end{pmatrix}$$

h_p et h_n sont les coefficients de pondération allant de 0 à 1. Ils sont définis tels que :

$$\begin{aligned} h_p &= 1 - \exp(-xv) \\ h_n &= \exp(-xv) \end{aligned}$$

avec $x = \frac{1}{N} \sum_{i=1}^N \|\mathbf{e}\|$. Ainsi lorsque l'effecteur est proche de la position désirée, x la moyenne de la norme de l'erreur s'approche de 0, le coefficient h_p tend alors vers 0 et h_n vers 1. Nous prenons la moyenne de la norme de l'erreur et non pas simplement la norme de l'erreur, pour réduire l'effet du bruit sur x . N et v sont des paramètres qui permettent de régler le passage des gains de 0 à 1. Ils seront réglés à la main grâce à la simulation.

Pour faire la fusion de ces données nous avons choisis de nous baser sur deux méthodes de commande hybride (cf. Section 2.3.2) que nous appellerons l'addition et la fusion de données capteur et que nous définissons ci dessous :

Commande hybride basée addition :

$$\mathbf{L}^+ = (\mathbf{P}^+ \quad \mathbf{N}_{\mathbf{uh}}^+) \quad (18)$$

$$\tau = \lambda \mathbf{L}^+ \mathbf{H} \mathbf{e} \quad (19)$$

$$\tau = \lambda (h_p \mathbb{I}_{k_p} \mathbf{P}^+ \mathbf{e}_p + h_n \mathbb{I}_{k_n} \mathbf{N}_{\mathbf{uh}}^+ \mathbf{e}_n) \quad (20)$$

Commande hybride basée fusion :

$$\mathbf{L} = \begin{pmatrix} \mathbf{P} \\ \mathbf{N}_{\mathbf{uh}} \end{pmatrix} \quad (21)$$

$$\tau = \lambda (\mathbf{H}\mathbf{L})^+ \mathbf{H} \mathbf{e} \quad (22)$$

Avantages et inconvénients des contrôleurs : L'analyse en stabilité des commandes hybrides est complexe. Dans le cas de la commande hybride basée fusion, le rang de l'inverse généralisée $(\mathbf{H}\mathbf{L})^+$ varie en fonction de la pondération \mathbf{H} . Concernant la commande hybride basée addition, il est également difficile d'assurer la convergence du contrôleur sachant que la contribution de chaque tâche \mathbf{e}_p et \mathbf{e}_n peut demander un mouvement contradictoire qui en s'additionnant s'annulerait. De plus pour éviter les minima locaux du contrôleur basé contours $\mathbf{N}_{\mathbf{uh}}$ il faut que \mathbf{H} soit construit de sorte à exploiter ce contrôleur seulement proche de la position désirée. Il faut donc avoir un ordre d'idée du "suffisamment proche" pour assurer la stabilité. Au final pour notre étude, nous ne ferons pas d'étude analytique de stabilité. Nous utiliserons seulement les tests en simulation pour fixer les gains de \mathbf{H} et vérifier la stabilité des contrôleurs dans l'espace de travail.

Suite à cette phase de réglage des gains, les résultats en simulation des deux contrôleurs hybrides envisagés se sont avérés très similaires et très satisfaisants en terme de trajectoire et stabilité. Un exemple de résultat en simulation pour un des contrôleur hybride est présenté Figure 9.(d).

3.5 Bilan des contrôleurs envisagés

Pour résumer cette partie qui se base principalement sur une analyse qualitative, nous présentons dans les Figures 11 et 12, les avantages et inconvénients des contrôleurs sur plusieurs critères :

La résolution correspond à la plus petite unité à laquelle l'information visuelle peut être mesurée.

La robustesse au bruit est évaluée sous l'hypothèse que l'extraction des droites dans l'image implique une meilleure robustesse au bruit.

La robustesse aux variations dimensionnelles est dans la majorité des cas dure à prédire pour notre application et sera évaluée plus précisément Section 5.3.

La stabilité du contrôleur qui considère la présence de singularité, minimum local et minimum global. Ce critère a été vérifié dans l'espace de travail par des essais en simulation.

La trajectoire a été évaluée par la simulation en cinématique.

La résolution et la robustesse au bruit impactent directement la convergence du système à la position désirée dans l'espace opérationnel et donc impactent directement la performance en précision. La stabilité et la trajectoire ne sont pas des fortes problématiques dans notre cas. En effet pour tous les contrôleurs choisis, si la position de départ est suffisamment proche de la position désirée, la trajectoire restera dans l'espace de travail et les minima locaux seront évités. La performance en précision peut donc toujours être évaluée. Cependant il est préférable d'avoir des contrôleurs capables de converger pour n'importe quel point de départ dans l'espace opérationnel.

Objet observé	Informations visuelles \mathbf{s}	Résolution	Robustesse au bruit	Robustesse aux variations dimensionnelles
Effecteur	$\mathbf{s}_p = (x, y)^T$	pixel	Robuste	Robuste
Avant-bras	$\mathbf{s}_n = \begin{pmatrix} \mathbf{n}_{11} \\ \mathbf{n}_{12} \\ \mathbf{n}_{21} \\ \mathbf{n}_{22} \end{pmatrix}$	Sub-pixellique	Très robuste	\mathbf{s} Dépend de paramètre géométrique variable
Avant-bras	$\mathbf{s}_r = (x, y)^T$	Sub-pixellique	Difficile à prévoir car \mathbf{s} dépend d'un calcul numérique	\mathbf{s} Dépend de paramètre géométrique variable
Effecteur et avant-bras	$\begin{pmatrix} \mathbf{s}_p \\ \mathbf{s}_n \end{pmatrix}$	Sub-pixellique	Très robuste	\mathbf{s} Dépend de paramètre géométrique variable

FIGURE 11 – Tableau comparatif des performances attendues en fonction des informations visuelles utilisées

Informations visuelles s	Matrice d'interaction et loi de commande	Nom du contrôleur	Stabilité	Trajectoire
$\mathbf{s}_p = (x, y)^T$	$\mathbf{L} = \mathbf{P}$ $\tau = k\mathbf{P}^+ \mathbf{e}$	Contrôleur basé observation effecteur	Stable	Rectiligne
$\mathbf{s}_n = \begin{pmatrix} \mathbf{n}_{11} \\ \mathbf{n}_{12} \\ \mathbf{n}_{21} \\ \mathbf{n}_{22} \end{pmatrix}$	$\mathbf{L} = \mathbf{N}_u$ $\tau = \lambda \mathbf{N}_u^+ \mathbf{e}$	Contrôleur basé contours \mathbf{N}_u	Aucune singularité ou minimum local détecté	Non-optimisée, dans certains cas impossible à réaliser en pratique
	$\mathbf{L} = \mathbf{N}_{uh}$ $\tau = \lambda \mathbf{N}_{uh}^+ \mathbf{e}$	Contrôleur basé contours \mathbf{N}_{uh}	Présence de minima locaux	
$\mathbf{s}_r = (x, y)^T$	$\mathbf{L} = \mathbf{P}$ $\tau = \lambda \mathbf{P}^+ \mathbf{e}$	Contrôleur basé reconstruction	Stable	Rectiligne
$\begin{pmatrix} \mathbf{s}_p \\ \mathbf{s}_n \end{pmatrix}$	$\mathbf{L}^+ = (\mathbf{P}^+ \quad \mathbf{N}_u^+)$ $\tau = \lambda \mathbf{L}^+ \mathbf{H} \mathbf{e}$	Contrôleur hybride basé addition	Dépend de la construction de H	Dépend de la construction de H, plutôt optimisée et rectiligne
	$\mathbf{L} = \begin{pmatrix} \mathbf{P} \\ \mathbf{N}_u \end{pmatrix}$ $\tau = \lambda (\mathbf{H}\mathbf{L})^+ \mathbf{H} \mathbf{e}$	Contrôleur hybride basé fusion		

FIGURE 12 – Tableau comparatif des contrôleurs

4 Simulation de la réponse du système

4.1 Modélisation du système complet sur Simulink

Au cours du stage un modèle du système complet a été réalisé sur Simulink (cf. Figure 13). Cette modélisation a dans un premier temps permis de tester le comportement de chaque loi de commande sur un modèle cinématique du robot. Puis dans le souci d'appréhender au mieux les problématiques de la commande sur le robot réel, il a été rajouté au modèle Simulink la prise en compte de :

- la réponse dynamique du système avec la boucle bas-niveau (déjà codée sous Simulink)
- la fréquence, le bruit et la quantification en pixel de la caméra

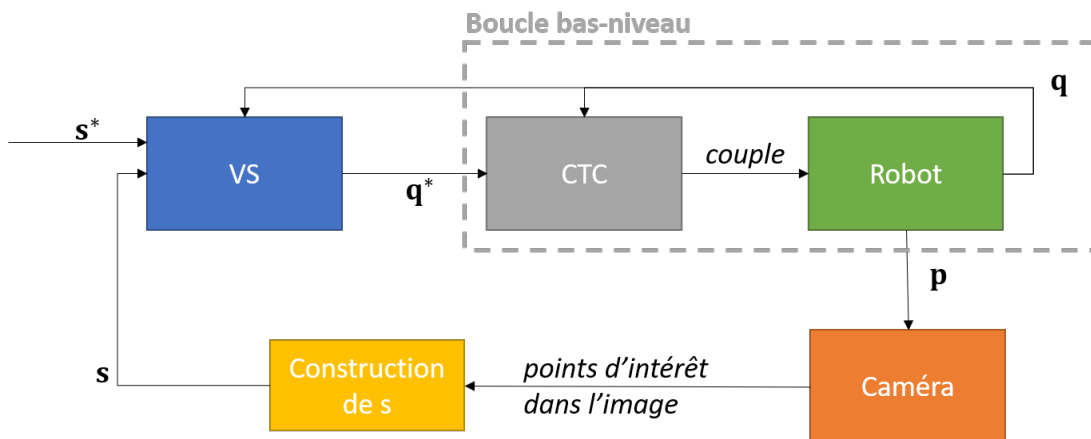


FIGURE 13 – Structure du schéma simulink réalisé

Le code de la commande référencée vision de la simulation (Sous-système "VS" en bleu sur la Figure 13) a été conçu de sorte à ce qu'il puisse être simplement copié et intégré sur la carte de commande du robot réel. Le fichier simulink a donc été conçu pour être compatible et optimisé pour la génération de code C et l'exécution temps réel.

4.1.1 Modèles utilisés

Modélisation du robot : Les modèles cinématiques, dynamiques, inverses et directs du mécanisme à cinq barres ont déjà été étudiés et définis, comme dans [Bri17]. Ces modèles analytiques utilisés dans notre cas ont été vérifiés à l'aide d'un modèle ADAMS du robot. Une fois ces modèles validés, nous avons testé les contrôleurs référencés vision sur un modèle cinématique dans un premier temps. Les vitesses articulaires \dot{q}^* issues de l'asservissement visuel sont donc directement envoyées sur le modèle cinématique du robot. Le robot peut être considéré comme un intégrateur pur. Puis les contrôleurs sont testés sur un modèle plus réaliste, le modèle dynamique du mécanisme. Ce modèle dynamique prend notamment en compte les efforts de friction.

Modélisation de la caméra : Le modèle utilisé pour la caméra est le modèle classique de projection perspective dans lequel un point $M = (X, Y, Z)^T$ dans l'espace 3D, se projette dans l'image en un point $m = (x, y, 1)^T$ tel que :

$$\begin{cases} x = X/Z \\ y = Y/Z \end{cases} \quad (23)$$

La transformation qui permet de passer des coordonnées $(x, y)^T$ en mètre, aux coordonnées $(u, v)^T$ en pixel est simplement :

$$\begin{cases} u = u_0 + x \frac{f}{l_x} \\ v = v_0 + y \frac{f}{l_y} \end{cases} \quad (24)$$

avec (u_0, v_0) le point central dans l'image, f la distance focale et (l_x, l_y) les dimensions d'un pixel en mètre. Ces paramètres correspondent aux paramètres intrinsèques de la caméra et sont définis en amont par calibration.

Il a été négligé dans le modèle caméra les effets non-linéaires de distorsion qui sont pris en compte dans le système réel par le traitement d'image avec la librairie ViSP et OpenCV.

Il a cependant été pris en compte dans le modèle Simulink de la caméra les effets de quantification de l'image en pixel ainsi qu'un bruit de mesure gaussien.

La fréquence à laquelle le bloc "Caméra" (en orange sur la Figure 13) fournit les points d'intérêt dans l'image a également été limitée à 50Hz.

4.2 Amélioration de la commande

Pour améliorer la réponse du système réel avec la boucle de vision, certaines contraintes ont été prises en compte :

- le "tracking" des informations visuelles ne fonctionne que sous la condition d'un déplacement limité des objets observés entre deux images ;
- En dehors de l'espace opérationnel de $800mm \times 200mm$ défini, le mécanisme peut entrer en position singulière ou en collision ;
- la commande bas-niveau est une commande en position articulaire qui s'exécute à une fréquence 10 fois supérieure à celle de la caméra ;

Dans cette partie nous détaillerons les solutions retenues pour améliorer les performances de la commande en prenant en compte ces contraintes.

4.2.1 Limite en vitesse des informations visuelles

Notre système de traitement d'image qui permet l'extraction et le suivi d'informations visuelles est limité en vitesse. Cela implique qu'une information visuelle peut être perdue si cette dernière se déplace trop rapidement entre deux images. Pour éviter ce problème nous avons choisi de limiter le déplacement désirée des informations visuelles dans l'image. Cela revient à borner la norme de $\dot{\mathbf{s}} = \lambda \mathbf{e}$ avec une limite l tel que :

$$\dot{\mathbf{s}} = \begin{cases} \|\lambda \mathbf{e}\| \underline{\mathbf{e}}, & \text{si } \|\lambda \mathbf{e}\| < l \\ l \underline{\mathbf{e}}, & \text{si } \|\lambda \mathbf{e}\| \geq l \end{cases} \quad (25)$$

Cette méthode très simple possède l'inconvénient de créer des discontinuités dans la commande des vitesses articulaires résultantes (cf. Figure 14.(b)). Il existe une autre méthode pour limiter la vitesse des informations visuelles désirées $\dot{\mathbf{s}}$ sans introduire de discontinuité, comme l'emploi d'un gain adaptatif.

Le gain adaptatif se définit comme suit [Vis] :

$$\lambda(x) = (\lambda_0 - \lambda_\infty) \exp\left(\frac{-\mu}{\lambda_0 - \lambda_\infty} x\right) + \lambda_\infty \quad \text{avec } x = \|\mathbf{e}\|_\infty$$

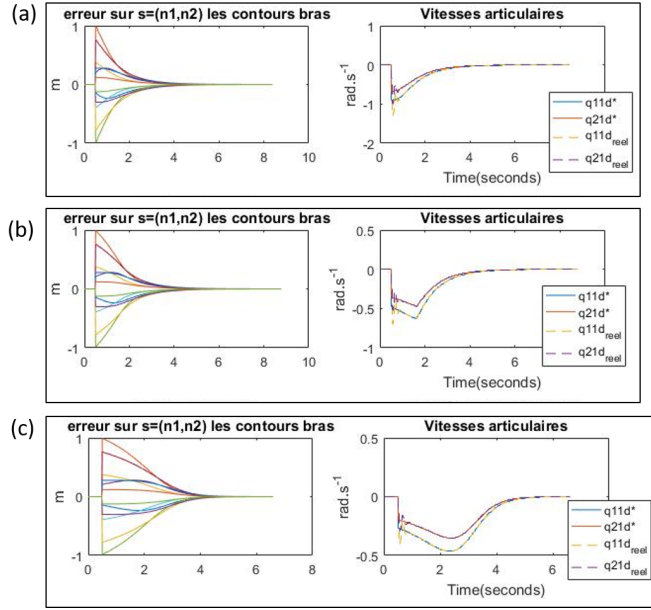


FIGURE 14 – Comparaison des résultats pour différentes limites en vitesse (a) : sans limite de vitesse; (b) : avec une saturation de la norme λe ; (c) : avec un gain adaptatif

;

Ce gain $\lambda(x)$ tend vers $\lambda_\infty \geq 0$ lorsque l'erreur e est grande et tend vers $\lambda_0 \geq 0$ lorsque l'erreur est petite. En choisissant $\lambda_0 > \lambda_\infty$, on amplifie la commande lorsque l'erreur est faible et on réduit la commande lorsque l'erreur est grande. La vitesse de passage d'un terme à l'autre est régulée par le facteur $\mu \geq 0$. Le résultat (cf. Figure 14.(c)) permet une convergence plus rapide et des vitesses désirées plus acceptables pour les moteurs. Pour un choix $\lambda_\infty = 0$, l'utilisation de ce gain revient à borner \dot{s} à une limite $l = \frac{\lambda_0^2}{\mu}$ (cf. Annexe 6.2 pour plus de détails).

Cependant ce gain à lui seul n'empêche pas complètement les discontinuités des vitesses, comme le montre la Figure 14 où les vitesses désirées \dot{q}^* au début de la commande changent soudainement. Or, éviter ce saut permet d'éviter un couple au démarrage très élevé. De plus, assurer la continuité de la commande en vitesse permet un meilleur suivi de la consigne et donc un meilleur contrôle du robot dans l'espace de travail.

4.2.2 Contrainte sur l'espace de travail

Continuité de la commande en vitesse : Des méthodes très simples existent pour garantir la continuité des consignes en vitesse au démarrage comme celle présentée dans [MC07]. Une commande en vitesse articulaire continue facilite le suivi de la commande par les moteurs. Ainsi la réponse du robot dans l'espace opérationnel est plus facile à prévoir et à contrôler.

Cette méthode revient simplement à enregistrer la valeur de λe au début de la commande (considéré à $t = 0$). Cette constante $\lambda e(t = 0)$ est multipliée par une exponentielle décroissante e^{-vt} qui permet d'annuler l'effet du mode continu au cours du temps. Avec le mode continu activé, la commande en vitesse articulaire désirée devient :

$$\dot{q}(t) = M(\lambda e - \underbrace{\lambda e(t=0)e^{-vt}}_{\text{terme du mode continu}}) \quad (26)$$

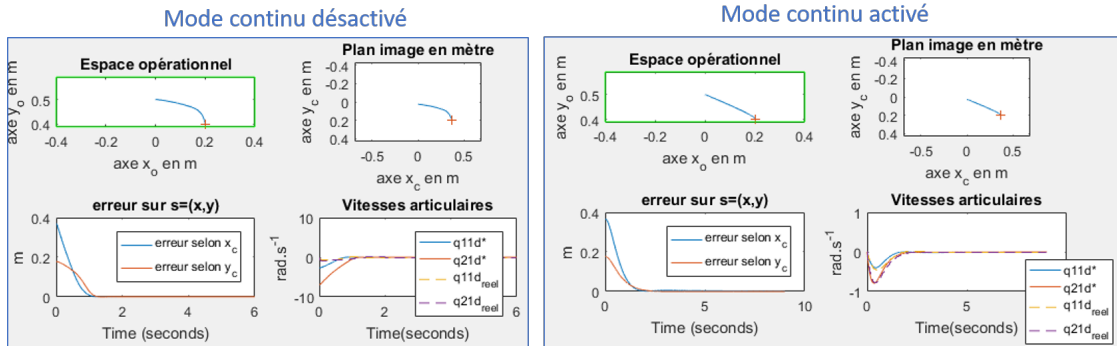


FIGURE 15 – Résultats de simulation montrant l'influence du mode continu pour le contrôleur basé observation effecteur

avec v une constante positive qui détermine la vitesse à laquelle le terme du mode continu s'annulera.

Sur la Figure 15 on peut voir que le mode continu permet le suivi de la commande des vitesses articulaires. La trajectoire du robot dans l'espace opérationnel correspond bien à une droite comme attendu pour le contrôleur basé effecteur.

Vérification de la trajectoire : Si la trajectoire est rectiligne pour le contrôleur basé observation de l'effecteur, la trajectoire des autres contrôleurs est beaucoup moins prévisible et peut sortir de l'espace opérationnel (cf. Figure 9). Un moyen très simple pour éviter les positions singulières du mécanisme et assurer la visibilité des informations visuelles, est de faire en sorte que la trajectoire de l'effecteur reste dans l'espace opérationnel défini. L'espace opérationnel de $800mm \times 200mm$ représenté en vert a été défini pour éviter que le mécanisme ne tombe dans une position singulière. La simulation sur Simulink réalisée constitue un bon moyen de calculer en avance une prévision de la trajectoire du système. Si cette trajectoire sort de l'espace de travail, la commande n'est pas envoyée au robot réel. Cette méthode n'est pas robuste au changement de modèle. Mais nous faisons l'hypothèse que les variations dimensionnelles sont négligeables pour nos contraintes sur la trajectoire. De plus une sécurité a été prise dans la définition de l'espace opérationnel de RobEcolo, qui permet d'en sortir de quelques centimètres sans rencontrer de singularité.

4.2.3 Compatibilité avec la commande bas-niveau

La commande bas-niveau ainsi que le modèle dynamique du robot ont été réalisés et implémentés sur Simulink par David Llevat Pamies au cours du projet RobEcolo.

Commande bas-niveau en position : La commande bas-niveau du système est une commande PID sur la position articulaire \mathbf{q} du robot. Cependant la boucle de vision ou boucle haut-niveau fournit une référence en vitesse articulaire $\dot{\mathbf{q}}^*$. La manière dont la commande en position articulaire désirée \mathbf{q}^* est calculée à partir de la commande en vitesse, a une influence sur la réponse du système. Considérons Δt la période d'exécution de la boucle bas-niveau. Si la commande articulaire est calculée à partir de la position courante mesurée \mathbf{q}^{curr} tel que :

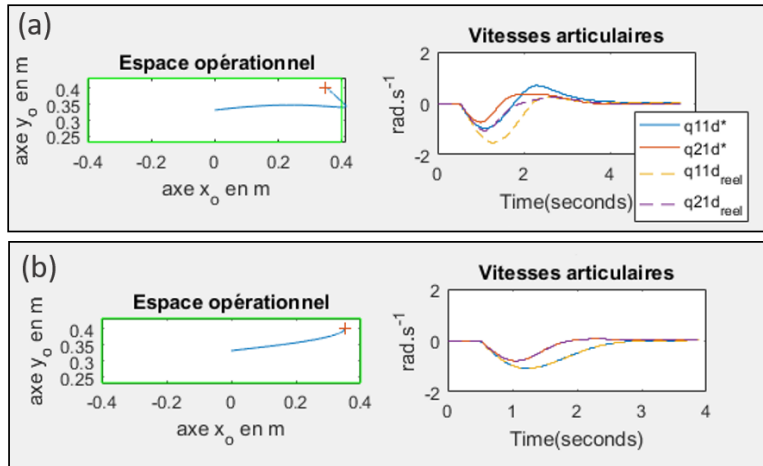


FIGURE 16 – Commande en position articulaire bas-niveau calculée à partir de (a) : la position articulaire courante; (b) : la commande en position précédente

$$\mathbf{q}^* = \mathbf{q}^{curr} + \dot{\mathbf{q}}^* \Delta t \quad (27)$$

nous obtenons les résultats Figure 16.(a). Si la commande articulaire est calculée à partir de la position articulaire désirée précédente \mathbf{q}_{i-1}^* tel que :

$$\mathbf{q}^* = \mathbf{q}_{i-1}^* + \dot{\mathbf{q}}^* \Delta t \quad (28)$$

nous obtenons les résultats Figure 16.(b).

Le fait de calculer la position désirée de la première manière (27) rend le système plus lent et se traduit par un dépassement non négligeable. Un dépassement n'est pas acceptable dans notre cas pour deux raisons : la trajectoire est plus imprévisible et sort de l'espace de travail pour des positions désirées aux bords de celui ci. La deuxième raison est le fait que la caractérisation en précision devient difficile avec l'utilisation de comparateur comme moyen de mesure. Nous utiliserons donc la méthode (28) dans la commande.

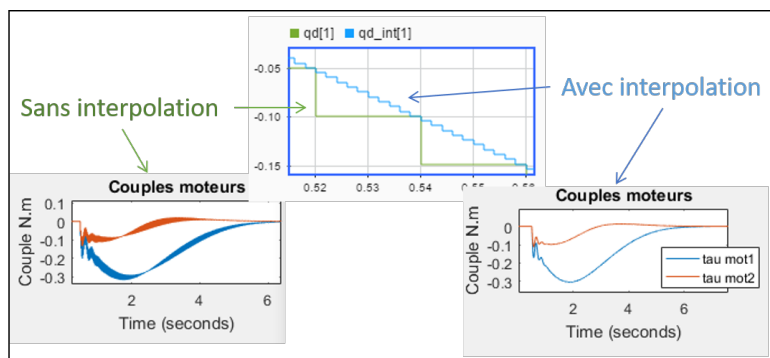


FIGURE 17 – Effet de l'interpolation de la commande en vitesse sur le couple moteur

Interpolation de la commande en vitesse : La boucle haut-niveau est limitée à $50Hz$, ce qui comprend le temps d'acquisition d'une image, de son traitement, et de la communication des points d'intérêt dans l'image à la carte de commande. La boucle bas-niveau tourne, elle, à une fréquence plus rapide de $500Hz$.

Dans ces conditions, les consignes en vitesses articulaires issues de la boucle haut niveau $\dot{\mathbf{q}}^*$ ne sont actualisées que toutes les 0.02 secondes. Si aucune étape d'interpolation n'est réalisée, la boucle bas-niveau perçoit une commande en vitesse en forme de paliers. Cela se traduit par une commande en accélération (donc en couple) en dent de scie comme on peut le voir Figure 17. Pour éviter ce problème, une simple interpolation linéaire de la consigne en vitesse permet de lisser la commande en couple, avec pour inconvénient de retarder d'une période de 0.02 secondes la consigne en vitesse.

4.3 Bilan de la simulation du système

Dans cette partie nous avons dans un premier temps modélisé le système complet sur Simulink pour tester la commande référencée vision.

Nous avons analysé quelques moyens simples d'améliorer la commande pour prendre en compte certaines contraintes du système réel. Pour la suite nous retenons dans la commande, l'utilisation d'un gain adaptatif qui limite la vitesse des informations visuelles. Pour assurer une trajectoire dans l'espace de travail et ainsi éviter collisions et singularités du mécanisme, nous utilisons le mode continu et la vérification de trajectoire. Et enfin pour assurer une bonne réponse dynamique du système avec la commande bas-niveau, la position articulaire désirée est calculée par intégration de la vitesse (28), qui est elle même obtenue par interpolation linéaire.

Une fois cette commande référencée vision bien définie et améliorée, elle peut être utilisée avec les différents contrôleurs (cf. Partie 3), pour caractériser leur performance en précision en simulation et sur le robot réel.

5 Caractérisation en précision

5.1 Méthode et critère de précision

5.1.1 Méthode d'essai

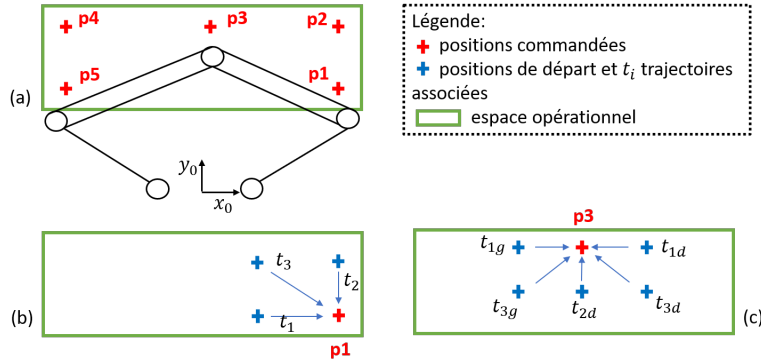


FIGURE 18 – **Schéma des essais** - (a) : positions commandées dans l'espace de travail ; (b) : Exemple des trajectoires de la position 1 ; (c) : Exemple des trajectoires de la position 3 ;

En s'inspirant de la norme ISO9283 [ISO98], nous avons choisi cinq positions à commander dans l'espace de travail (cf. Figure 18.(a)). Chaque position est atteinte depuis plusieurs positions de départ. Les positions de départ ont été définies de sorte à atteindre la position commandée suivant différentes directions : t_1 pour une direction suivant x_0 , t_2 pour une direction suivant y_0 et t_3 pour une direction diagonale. Pour simplifier l'appellation, nous emploierons le terme de trajectoire pour parler d'une position de départ même si il n'y a pas de suivi de trajectoire à proprement parlé. Chaque position commandée est donc testée sur trois trajectoires (cf. Figure 18.(b)) sauf la position commandée 3 qui est testée sur cinq trajectoires (cf. Figure 18.(c)). Chaque trajectoire est répétée 20 fois. Chaque position commandée est donc testée sur 60 cycles.

En pratique nous utilisons deux comparateurs d'une résolution de $10\mu m$ pour mesurer la position relative entre la position obtenue par asservissement visuel et la position commandée (cf. Figure 19.(a)). Avant les essais d'une trajectoire, la position commandée est atteinte par une commande en position articulaire. Les informations visuelles désirées \mathbf{s}^* ainsi que la valeur des comparateurs à la position commandée sont alors enregistrée (cf. Figure 20). Pour s'assurer que la structure qui maintient les comparateurs est bien fixe et que la mesure des comparateurs enregistrée à la position commandée est répétable, nous effectuons quelques répétitions de la commande en position articulaire à la position désirée et vérifions que les comparateurs redonnent la même mesure au $20\mu m$ près.

5.1.2 Critère de précision dans l'espace opérationnel

Nous définissons notre objectif de précision de $500\mu m$, par une exactitude et répétabilité de pose inférieures à $500\mu m$ dans tout l'espace de travail. Nous considérons seulement la position de l'effecteur dans le plan horizontal (selon les axes x_0 et y_0 Figure 18.(a)) et négligeons la position en hauteur et l'orientation de l'effecteur. En s'inspirant des définitions dans la norme ISO9283, nous définissons la précision (AP) et l'exactitude (RP) de pose pour notre cas ci-dessous.

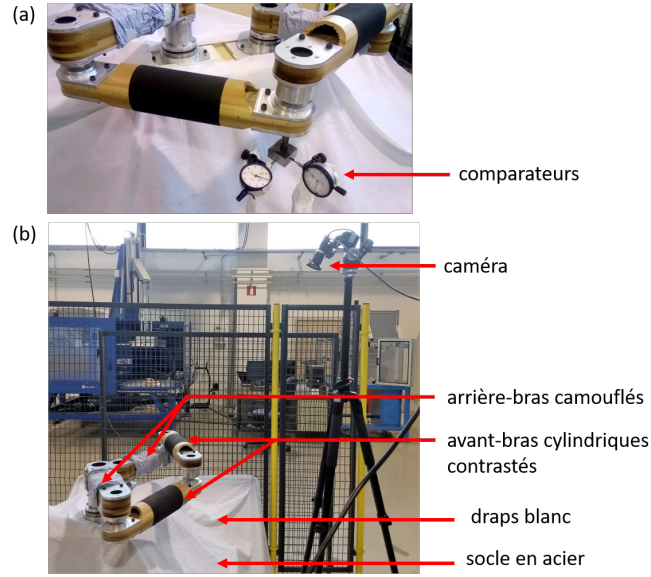


FIGURE 19 – **Photos de l'installation expérimentale** avec (a) : le positionnement des comparateurs sur le socle en acier pour le robot commandé à la position 3 ; (b) : installation globale de la caméra et de l'environnement du robot

Avec (x_{ai}, y_{ai}) la position atteinte au cycle i et (x_c, y_c) la position commandée, la moyenne des positions atteintes est : $(\bar{x} = \frac{1}{N} \sum x_{ai}, \bar{y} = \frac{1}{N} \sum y_{ai})$, la précision de pose est définie tel que :

$$AP = \sqrt{(\bar{x} - x_c)^2 + (\bar{y} - y_c)^2}$$

Avec la distance l_i de la position atteinte i par rapport à la moyenne :

$$l_i = \sqrt{(x_{ai} - \bar{x})^2 + (y_{ai} - \bar{y})^2}$$

et la moyenne \bar{l} et l'écart type S_l des distances l_i :

$$\bar{l} = \frac{1}{N} \sum l_i, \quad S_l = \sqrt{\frac{\sum (l_i - \bar{l})^2}{N-1}}$$

la répétabilité est définie tel que :

$$RP = \bar{l} + 3S_l$$

Avant de caractériser la précision du robot réel, nous avons vérifié les performances en précision en simulation.

5.2 Résultats obtenus

5.2.1 Vérification en simulation

Dans la simulation, l'effecteur est considéré comme un point donc quantifié au pixel près et bruité par un bruit gaussien d'écart type 0.1 pixel. Pour les segments des contours construits à l'aide de plus de 100 points dans l'image, il est appliqué aux extrémités des segments un bruit

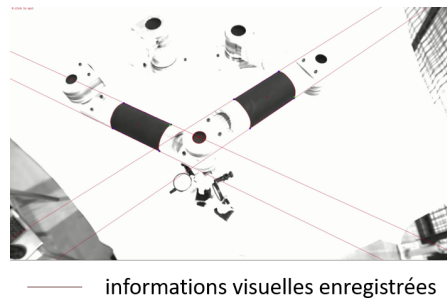


FIGURE 20 – Image obtenue par la caméra après convergence du robot à la position commandée 3

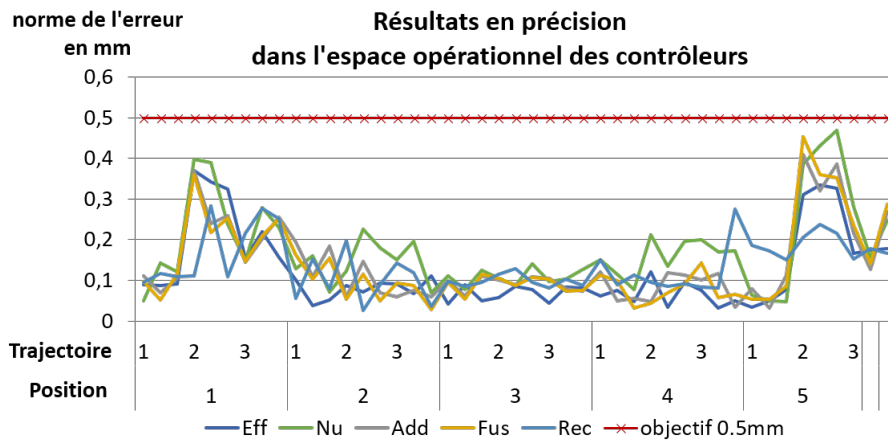


FIGURE 21 – Résultats en simulation de l'erreur de pose dans l'espace opérationnel en fonction des positions commandées

gaussien d'écart type 0.01 pixel et une quantification au 0.1 pixel près. Les résultats sont présentés Figure 21.

On remarque qu'il y a peu de différence au niveau de la performance en précision des différents contrôleurs en simulation. On peut souligner qu'aux positions 1 et 5 les résultats sont légèrement moins bons. Cela s'explique par le fait que ce sont les points les plus éloignés de la caméra (cf. Figure 5). La norme de l'erreur dans l'espace opérationnel reste inférieure à $500\mu m$. Toutes les positions atteintes par le robot se trouvent dans un cercle de rayon $500\mu m$ autour de la position commandée. L'exactitude de pose est donc nécessairement comprise dans ce cercle.

Une fois les performances en précision vérifiées en simulation, elles ont été étudiées sur le robot réel.

5.2.2 Résultats d'essais sur robot réel

Tous les contrôleurs retenus en partie 3 ont été testés sur le robot réel. Cependant par manque de temps, seulement trois d'entre eux ont pu voir leurs performances en précision testées selon la

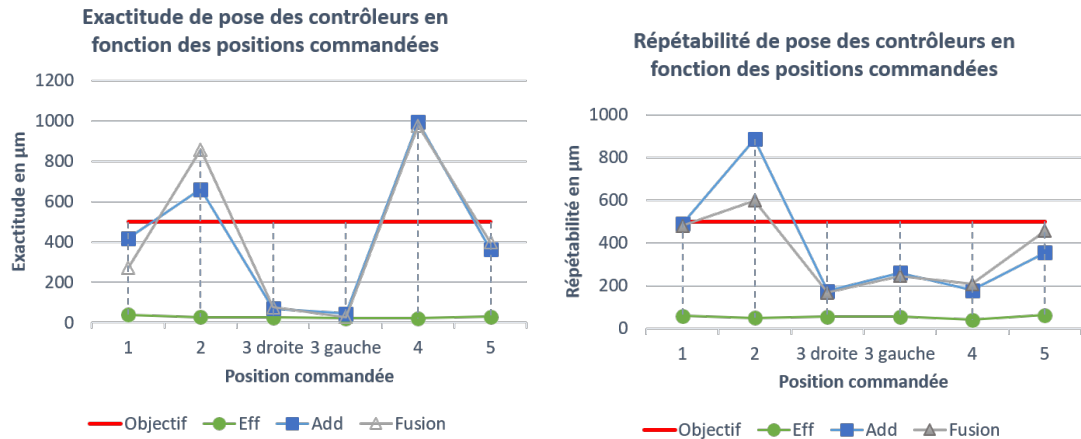


FIGURE 22 – Résultats sur robot réel de l’exactitude et répétabilité de pose des contrôleurs en fonction des positions commandées. Les résultats au point 3 droite correspondent aux trajectoires $t1d$ $t2d$ et $t3d$ et 3 gauche aux trajectoires $t1g$ et $t3g$.

procédure énoncée Section 5.1.1. Ces derniers sont le contrôleur basé observation effecteur et les deux contrôleurs hybrides. Cela permet de comparer les performances en précision des informations visuelles s_p et s_n . On remarque Figure 22 que les résultats d’exactitude et répétabilité du contrôleur basé observation de l’effecteur sont très bons. Ils arrivent en dessous des espérances avec une exactitude et une répétabilité à moins de $100\mu m$ dans tout l’espace de travail.

Les contrôleurs hybrides possèdent de moins bons résultats que l’effecteur. Les résultats aux positions 2 et 4, aux bords extérieurs de l’espace de travail, sortent même des objectifs fixés à $500\mu m$. À ces positions commandées le contrôleur basé contours N_u ainsi que le contrôleur basé reconstruction ont été testés sur simplement deux répétitions et obtenaient le même ordre d’imprécision. Nous nous sommes donc penchés sur l’analyse du bruit des informations visuelles dans l’image.

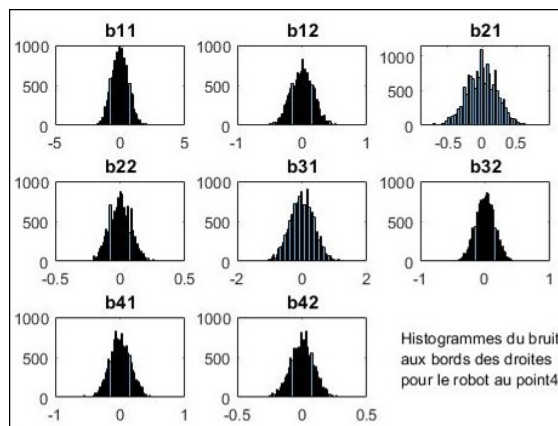


FIGURE 23 – La distribution du bruit : exemple à la position 4 pour les extrémités des droites

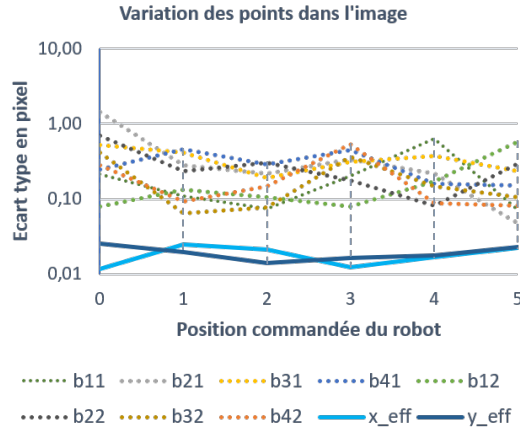


FIGURE 24 – **Ecart type de la variation des points dans l'image** en fonction de la position commandée du robot

5.2.3 Analyse du bruit dans l'image

Pour analyser le bruit dans l'image, nous avons enregistré la position des 9 points issus du traitement d'image, qui définissent les informations visuelles, à savoir :

- le point (x, y) de l'effecteur. Cela revient à analyser deux variables bruitées.
- les points aux extrémités $bj1$ et $bj2$ de chaque droite $j \in [1, 4]$ dans l'image. Comme ces points sont aux bords de l'image, dans la majorité des cas une seule variable évolue, celle suivant l'axe horizontal ou vertical de l'image. Dans la suite nous considérerons que analyser le bruit du point $bj1$ ou $bj2$ revient à analyser le bruit d'une seule variable, celle qui évolue.

Nous avons enregistré ces valeurs sur une période de 15 secondes, lorsque le robot est figé aux différentes positions commandées (cf. Figure 18.(a)).

Avant tout, il paraît intéressant de vérifier si la distribution du bruit suit bien une gaussienne. Comme le montre la Figure 23, qui est un exemple à la position 4, nous avons pu valider l'hypothèse sur la forme de la distribution du bruit. Suite à cela, nous nous sommes intéressés à la variance du bruit en fonction des informations visuelles et de la position du robot dans l'image. On remarque Figure 24 que contrairement aux attentes, la position des extrémités des droites est environ 10 fois plus bruitée que la position effecteur. Ceci peut expliquer pourquoi le contrôleur basé observation effecteur est plus performant en précision. On remarque également qu'il n'y a pas de différence notable du bruit par rapport à la position du robot. Aux positions commandées critiques 2 et 4, le bruit des extrémités des droites n'est pas plus élevé. La moins bonne performance des contrôleurs basés contours, aux positions commandées 2 et 4 ne semble pas exclusivement dues aux bruits. Il faut tout de même préciser que les mesures pour l'analyse du bruit et les mesures pour les performances en précision ont été prises à un moment différent de la journée. La luminosité de la scène était peut être légèrement différente.

5.3 Robustesse aux variations dimensionnelles

Les membres du robot étant en bois, on peut s'attendre à une évolution de leurs dimensions, dans un environnement avec un taux d'humidité variable. Nous considérons dans cette étude seule-

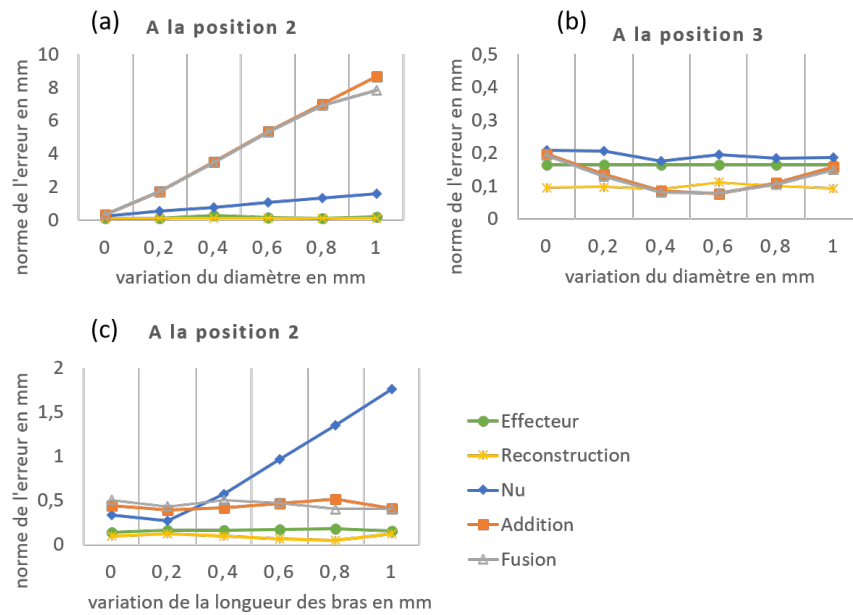


FIGURE 25 – Graphiques sur la robustesse aux variations dimensionnelles des contrôleurs (a) et (b) : pour une variation de diamètre; (c) : pour une variation de longueur des bras ;

ment le cas d’une évolution de longueur et de diamètre des bras. D’autres déformations tel que le pliage des membres ne sont pas prises en compte.

Au cours du projet une étude a permis d’estimer la variation en longueur dans le pire des cas qui est de l’ordre de 1mm . La variation du diamètre possible des bras n’a pas pu être estimée du fait de sa complexité. Nous avons donc fixé cette variation à également 1mm dans le pire des cas. Les variations dimensionnelles possibles réelles seront évaluées lors d’essais expérimentaux dans la suite du projet.

Comme nous l’avons évoqué partie 3, les informations visuelles basées sur le contours des bras ne sont pas complètement indépendantes aux paramètres géométriques. Nous nous intéressons ici à l’effet de la variation dimensionnelle des membres en bois pour les performances en précision, des différents contrôleurs.

Grâce à la simulation, nous effectuerons un enregistrement des informations visuelles désirées pour un état initial du modèle géométrique du robot. Cet état initial considère les longueurs (l_1, l_2, l_3, l_4) et diamètre des avant-bras (r) mesurés lors de l’identification. Puis l’asservissement visuel avec les différents contrôleurs sera lancé sur un modèle géométrique modifié : les longueurs des membres ($l_1 + dl, l_2 + dl, l_3 + dl, l_4 + dl$) allongés de dl et diamètre des avant-bras ($r + dr$) agrandi de dr . Les tests d’asservissement visuel ont été réalisés sur les 5 positions commandées (cf. Figure 18.(a)) suivant la trajectoire horizontale t_1 . Les résultats complets sont présentés en annexe Section 6.3. Nous présenterons ici seulement un condensé des résultats Figure 25.

Robustesse aux variations du diamètre : On peut remarquer Figure 25.(a) que les trois contrôleurs (Nu, Addition et Fusion) utilisant les contours comme informations visuelles \mathbf{s}_n

peuvent être très sensibles aux variations du diamètre des bras. La performance en précision dans l'espace opérationnel en est très impactée. Il est intéressant de noter que cet impact est plus ou moins important en fonction de la position du robot. Par exemple à la position 3 (cf. Figure 25.(b)) la variation du diamètre a beaucoup moins d'impact sur les performances qu'à la position 2 (cf. Figure 25.(a)).

Ces résultats montrent aussi que les performances du contrôleur basé sur la reconstruction de la position de l'effecteur à partir des contours ne sont pas influencées par les variations du diamètre appliquées.

Robustesse aux variations de longueur : Seule la performance du contrôleur N_u basée sur la direction des bras est significativement impactée par la variation de longueur. Les performances des contrôleurs hybrides (Addition et Fusion) qui utilisent une matrice d'interaction basée sur la direction et la position des jambes N_{uh} (cf. partie.3) sont moins impactées par la variation en longueur.

Les résultats confirment également que les performances des contrôleurs basés effecteur et reconstruction ne sont pas influencées par les variations de longueur.

5.4 Bilan de la caractérisation en précision

Dans cette dernière partie, nous avons vu que les contrôleurs hybrides, qui sont basés sur l'observation des contours ont été moins performants en pratique qu'en théorie. Nous apporterons dans ce rapport quelques éléments de réponse pour expliquer cette différence de performance. Tout d'abord, le bruit considéré en simulation est plus faible que le bruit mesuré en pratique pour les extrémités de droites. Or le bruit a une grosse influence sur la précision du système. De plus, un défaut de forme des membres du robot n'a pas été pris en compte dans la simulation. En effet la section des avant-bras n'est pas tout à fait circulaire en pratique. Elle est plutôt de forme elliptique avec une différence entre le demi-grand axe et le demi-petit axe de l'ordre de $1mm$. Le diamètre des bras perçu dans l'image peut donc varier en fonction de l'orientation des bras. De plus il a été montré que les contrôleurs hybrides sont très sensibles aux variations du diamètre entre l'enregistrement et la commande à la position désirée. Il est donc possible que le défaut de forme des bras ait une influence significative sur la précision des contrôleurs basés sur l'observation des contours.

Nous avons également vu, dans cette dernière partie que le contrôleur basé observation effecteur permet d'obtenir une exactitude et répétabilité inférieure à $100\mu m$ avec le système réel dans tout l'espace de travail. De plus les variations dimensionnelles considérées en simulation n'ont aucune influence sur les performances en précision du système. L'objectif en précision du projet RobEcolo peut être considéré comme atteint.

Conclusion

Au cours de ce stage de fin d'études, différents types de commandes basées vision ont été étudiés. Nous avons pu répondre à certaines questions et problématiques concernant le choix de cette commande pour assurer les meilleures performances en précision d'un robot en bois.

En prenant en compte la problématique de variation dimensionnelle des bras du robot, nous avons choisi des informations visuelles qui nous semblaient pertinentes pour assurer les meilleures performances en précision. Nous avons ensuite étudié les matrices d'interactions existantes associées. L'étude cinématique en simulation des contrôleurs retenus a permis de découvrir les problèmes de stabilité et de trajectoire pour les contrôleurs basés contours. Nous avons donc proposé des nouveaux contrôleurs (hybrides et reconstruction) qui permettent d'éviter ces problèmes dans tout l'espace de travail, tout en exploitant les avantages de l'observation des contours pour le positionnement en précision.

Assurer une commande robuste en pratique implique la prise en compte de certaines contraintes. Par exemple pour assurer le maintien du suivi des informations visuelles, une limite en vitesse de la commande a été mise en place. Pour éviter les singularités du mécanisme et assurer une meilleure réponse dynamique du système, quelques solutions simples dans la commande ont été retenues comme l'interpolation linéaire de la commande articulaire. Une fois la boucle de commande référencée vision complètement définie et codée sur Simulink, nous avons pu vérifier les performances en précision en simulation puis sur le robot réel.

En simulation tous les contrôleurs permettent d'atteindre l'objectif de précision à $500\mu m$. En pratique seuls deux types de contrôleurs ont été testés : les contrôleurs hybrides et le contrôleur basé observation effecteur. Les premiers n'ont pas permis d'obtenir les performances souhaitées. Le contrôleur basé observation de l'effecteur a lui permis de prouver qu'une précision inférieure à $100\mu m$ pouvait être atteinte dans tous l'espace de travail. Une dernière étude en simulation a permis de comparer la robustesse de tous les contrôleurs étudiés aux variations de longueurs et de diamètres des bras. Seuls les contrôleurs basés observation effecteur et reconstruction permettent toujours d'atteindre les objectifs en précision dans ces conditions.

Dans la suite du projet il paraît intéressant de pousser l'investigation sur les causes de la différence de performance obtenue entre la simulation et la réalité pour les contrôleurs hybrides. On pourra également s'intéresser plus en détail aux performances du contrôleur basé reconstruction, qui semble intéressant pour sa robustesse aux variations dimensionnelles. Des essais expérimentaux pour vérifier les résultats obtenus en simulation semblent pertinents à réaliser.

6 Annexes

6.1 Détails sur les contrôleurs basés sur l'observation des jambes

6.1.1 Construction des informations visuelles s_n

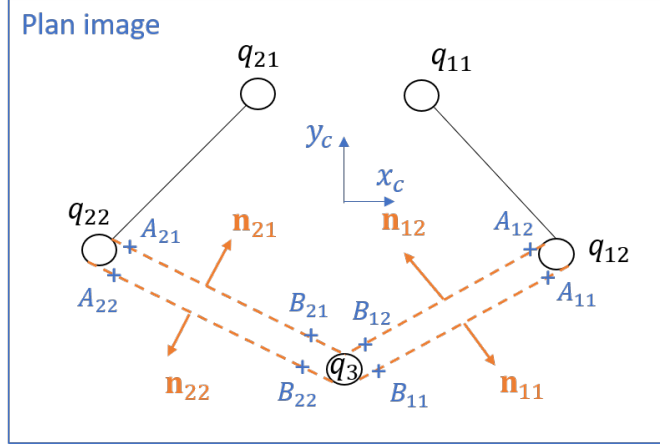


FIGURE 26 – Schéma pour la construction des informations visuelles s_n

Les droites issues des contours des cylindres sont paramétrées dans l'image par leur vecteur moment de Plücker associé $\underline{\mathbf{n}}_{ij}$ avec $i \in [1, 2]$ correspondant au numéro de la jambe (ou du cylindre) et $j \in [1, 2]$ au numéro du contour. Ce vecteur unitaire de dimension 3, peut être construit à l'aide de la connaissance de deux points A et B dans l'image. Si on considère les coordonnées normalisées, ces points dans le repère caméra R_c ont la forme : $A = ({}^c a_x \quad {}^c a_y \quad 1)^T$ et $B = ({}^c b_x \quad {}^c b_y \quad 1)^T$. Si on considère $O_c = (0 \quad 0 \quad 0)^T$ le centre de projection et du repère R_c , les vecteurs moments de Plücker sont construits tels que :

$$\begin{aligned} \mathbf{n}_{i1} &= -\overrightarrow{O_c B_{i1}} \times \overrightarrow{A_{i1} B_{i1}} \\ \mathbf{n}_{i2} &= \overrightarrow{O_c B_{i2}} \times \overrightarrow{A_{i2} B_{i2}} \\ \underline{\mathbf{n}}_{ij} &= \frac{\mathbf{n}_{ij}}{\|\mathbf{n}_{ij}\|} \end{aligned}$$

6.1.2 Construction des informations visuelles s_r

Pour construire $\mathbf{s}_r = (x, y)^T$ la position de l'effecteur dans l'image à partir des contours des bras $\underline{\mathbf{n}}_{ij}$, nous avons procédé en deux étapes. La première consiste à reconstruire la position et direction de l'axe central de chaque cylindre i sous leur forme de coordonnées de Plücker $(\underline{\mathbf{u}}_i, h_i \underline{\mathbf{h}}_i)$. On considère ici que toutes les coordonnées sont exprimées dans le repère caméra. Ainsi en reprenant les équations (11) introduites dans la Section 2.2.2, on a :

$$\underline{\mathbf{u}}_i = f_u(\underline{\mathbf{n}}_{i1}, \underline{\mathbf{n}}_{i2}) \quad (29)$$

$$\underline{\mathbf{h}}_i = f_h(\underline{\mathbf{n}}_{i1}, \underline{\mathbf{n}}_{i2}) \quad (30)$$

$$h_i = \frac{r_i}{\sin(\theta_i)} \quad (31)$$

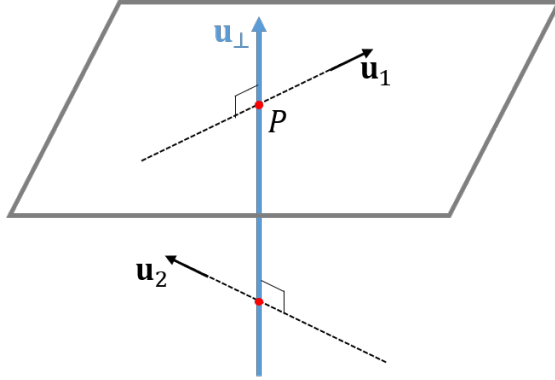


FIGURE 27 – Points d'intersections de la perpendiculaire commune à deux axes non-coplanaires

avec f_u et f_h les fonctions issues des relations (11) qui déterminent respectivement \mathbf{u}_i et \mathbf{h}_i à partir des moments des contours. Pour que les informations visuelles ne dépendent que de la connaissance des contours et du rayon r_i , nous définissons θ_i en fonction des contours de la façon suivante :

$$\theta = \frac{1}{2}(\pi - \arctan 2(\|\mathbf{c}\mathbf{n}_1 \times \mathbf{c}\mathbf{n}_2\|, \mathbf{c}\mathbf{n}_1 \cdot \mathbf{c}\mathbf{n}_2)) \quad (32)$$

avec $\arctan 2(y, x)$ la fonction arc tangente qui renvoie l'argument du nombre complexe $(x + \mathbf{i}y)$.

La deuxième étape pour construire \mathbf{s}_r consiste à calculer l'intersection des deux axes des cylindres obtenus. Cependant en réalité sur notre mécanisme à cinq barres, les deux axes ne sont pas coplanaires. On s'intéresse alors à l'intersection de la perpendiculaire commune \mathbf{u}_\perp aux deux axes, avec l'axe du cylindre 1. Cette intersection représentée Figure 27 est donnée par [YB17] et définie telle que :

$$P = \frac{(\mathbf{h}_2 \cdot (\mathbf{u}_1 \times \mathbf{u}_2))\mathbf{u}_1 - \mathbf{h}_1 \times (\mathbf{u}_2 \times (\mathbf{u}_1 \times \mathbf{u}_2))}{\|\mathbf{u}_1 \times \mathbf{u}_2\|^2} \quad (33)$$

Et enfin \mathbf{s}_r est finalement obtenu par projection du point 3D, $P = (X, Y, Z)^T$ dans l'image :

$$\mathbf{s}_r = \begin{pmatrix} X/Z \\ Y/Z \end{pmatrix} \quad (34)$$

6.1.3 Matrice d'interaction basée direction des jambes

[Dal07] présente dans sa thèse une matrice d'interaction associée aux contours des jambes générique pour différentes architectures de robot parallèle qu'il définit comme suit :

$$\left\{ \begin{array}{l} \mathbf{N}_{ij} = \mathbf{R}_{ij}\mathbf{L}_i + \mathbf{S}_{ij}\mathbf{M}_{\mathbf{u}_i} \\ \mathbf{R}_{ij} = -\frac{(\mathbf{u}_i \times \mathbf{n}_{ij})\mathbf{n}_{ij}^T}{A_i^T(\mathbf{u}_i \times \mathbf{n}_{ij})} \\ \mathbf{S}_{ij} = -(\mathbf{I}_3 - \frac{(\mathbf{u}_i \times \mathbf{n}_{ij})A_i^T}{A_i^T(\mathbf{u}_i \times \mathbf{n}_{ij})})\mathbf{u}_i\mathbf{n}_{ij}^T \end{array} \right. \quad (35)$$

Avec $\dot{\mathbf{u}}_{ij}$ le vecteur moment unitaire de Plücker [Plu65] [And05] qui qualifie le contour j du cylindre i . La matrice d'interaction totale $\mathbf{N}_{\mathbf{u}}$ étant obtenu par concaténation des \mathbf{N}_{ij} pour $i \in [1, 2]$ et $j \in [1, 2]$. Le calcul de cette matrice dépend de

- la position 3D d'un point A_i sur l'axe du cylindre i , qui n'est pas nécessairement fixe ;
- la matrice d'interaction \mathbf{L}_i associée au point A_i ;
- la matrice d'interaction $\mathbf{M}_{\mathbf{u}}$ associée aux directions des jambes \mathbf{u} tel que $\dot{\mathbf{u}}_i = \mathbf{M}_{\mathbf{u}i}\boldsymbol{\tau}$.

Pour notre application, nous avons choisi le point de l'effecteur pour A_i dont la position dans l'espace 3D est reconstruite à partir de l'observation du centre du cercle dans l'image.

6.1.4 Matrice d'interaction basée direction et position des bras

Une matrice d'interaction basée sur la direction et position des bras a été réalisée par Lila Kaci lors de sa thèse. Elle prends la forme :

$$\mathbf{N}^+ = \mathbf{M}_{\mathbf{uh}}^+ \begin{bmatrix} {}^{\mathbf{u}}\mathbf{J}_{\mathbf{n}} \\ {}^{\mathbf{h}}\mathbf{J}_{\mathbf{n}} \end{bmatrix} \quad (36)$$

Avec

- $\mathbf{M}_{\mathbf{uh}}^+$ la matrice d'interaction associée aux directions \mathbf{u} et moments \mathbf{h} des axes des jambes tel que $\boldsymbol{\tau} = \mathbf{M}_{\mathbf{uh}}^+ \begin{bmatrix} \dot{\mathbf{u}} \\ \dot{\mathbf{h}} \end{bmatrix}$. Elle est définie dans [VBPC14] pour le robot cinq barres, et exploite l'intersection des axes des deux bras.
- Les matrices de transformation ${}^{\mathbf{u}}\mathbf{J}_{\mathbf{n}}$ et ${}^{\mathbf{h}}\mathbf{J}_{\mathbf{n}}$ définies tel que $\begin{bmatrix} \dot{\mathbf{u}} \\ \dot{\mathbf{h}} \end{bmatrix} = \begin{bmatrix} {}^{\mathbf{u}}\mathbf{J}_{\mathbf{n}} \\ {}^{\mathbf{h}}\mathbf{J}_{\mathbf{n}} \end{bmatrix} \dot{\mathbf{u}}$. Ces matrices sont issues de la dérivée des expressions de géométrie (11) reliant les moments des contours $(\mathbf{n}_{i1}, \mathbf{n}_{i2})$ aux coordonnées de Plücker normalisées $(\mathbf{u}_i, \mathbf{h}_i)$ de l'axe du cylindre i .

6.2 Détails sur la saturation en vitesse avec le gain adaptatif

Nous cherchons à assurer que la vitesse des informations visuelles $\dot{\mathbf{s}}$ ne dépasse pas une certaine limite $l > 0$ grâce à l'utilisation d'un gain λ adaptatif. Cela revient à assurer la condition suivante :

$$\|\dot{\mathbf{s}}\| < l \quad (37)$$

avec $\dot{\mathbf{s}}$ obtenue à partir de l'erreur $\mathbf{e} = (\mathbf{s} - \mathbf{s}^*)$ et du gain λ tel que :

$$\dot{\mathbf{s}} = \lambda \mathbf{e} \quad (38)$$

Posons la variable x correspondant à la norme de \mathbf{e} tel que : $x = \|\mathbf{e}\| \geq 0$, et la fonction gain adaptatif λ dépendant de x tel que :

$$\lambda(x) = (\lambda_0 - \lambda_\infty) \exp\left(\frac{-\mu}{\lambda_0 - \lambda_\infty} x\right) + \lambda_\infty \quad (39)$$

avec λ_0 , λ_∞ et μ des nombres scalaires positifs tels que $\lambda_0 > \lambda_\infty$. Ces trois scalaires sont les paramètres du gain adaptatif.

Or d'après (38) et (39) nous avons :

$$\|\dot{\mathbf{s}}\| = |\lambda(x)x| \quad (40)$$

De plus nous savons que pour tout scalaire $a > 0$ et $x \geq 0$ nous avons :

$$\exp(-ax) < \frac{1}{1+ax} < \frac{1}{ax} \quad (41)$$

Nous pouvons donc en déduire que :

$$|\lambda(x)x| = \lambda(x)x < ((\lambda_0 - \lambda_\infty) \frac{\lambda_0 - \lambda_\infty}{\mu x} + \lambda_\infty)x \quad (42)$$

En posant $k_\infty = 0$ la valeur du gain adaptatif pour une erreur infinie, nous obtenons :

$$\lambda(x)x < \left(\frac{\lambda_0^2}{\mu}\right) \quad (43)$$

L'inéquation (37) peut donc être assurée avec un gain adaptatif dont les paramètres vérifient :

$$\begin{cases} \lambda_\infty = 0 \\ l = \frac{\lambda_0^2}{\mu} \end{cases} \quad (44)$$

6.3 Résultats robustesse aux variations dimensionnelles

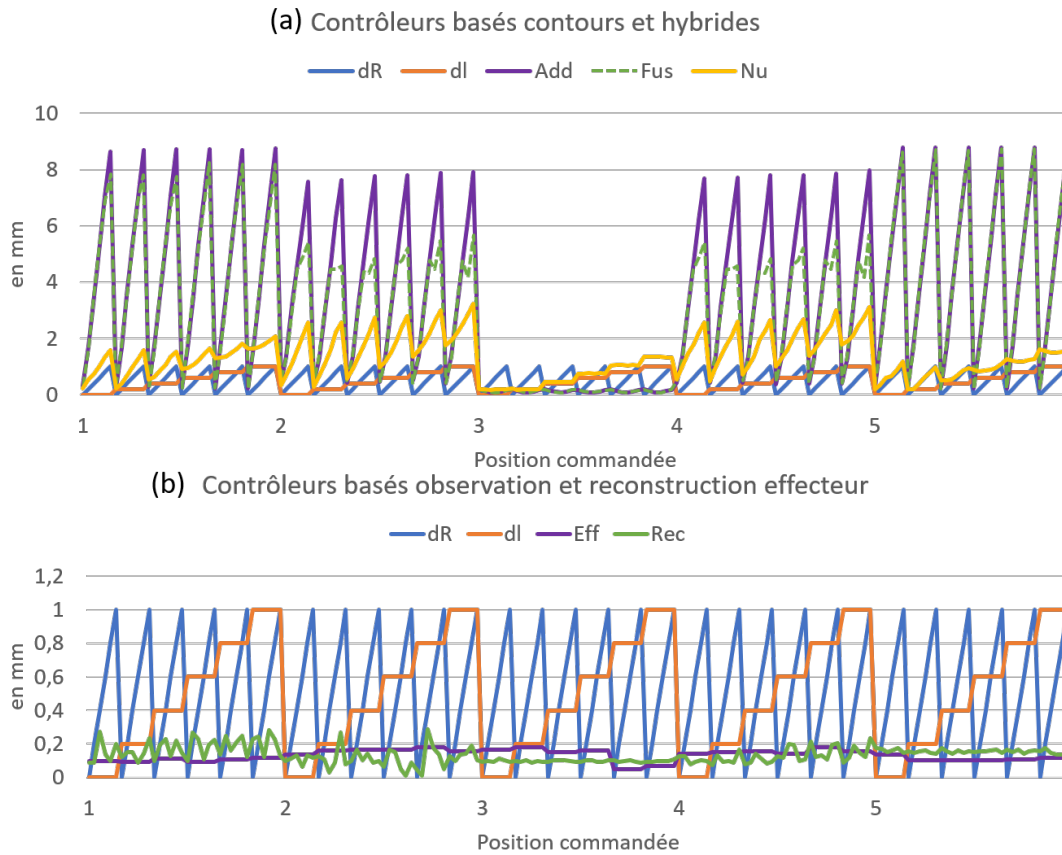


FIGURE 28 – Graphiques sur la robustesse aux variations dimensionnelles des contrôleurs (a) : basés contours et hybrides ; (b) : basés observation et reconstruction effecteur ; dR et dl sont respectivement les variations du diamètre et de longueur appliquées sur les bras ; Les courbes Add, Fus, Nu, Eff et Rec correspondent à la norme de l’erreur dans l’espace opérationnel, des contrôleurs après stabilisation. Pour ces résultats la simulation prend en compte la quantification en pixel mais pas de bruit gaussien.

Références

- [ADM07] Nicolas Andreff, Tej Dallej, and Philippe Martinet. Image-based visual servoing of a gough—stewart parallel manipulator using leg observations. *The International Journal of Robotics Research*, 26(7) :677–687, 2007.
- [AEH02] Nicolas Andreff, Bernard Espiau, and Radu Horaud. Visual servoing from lines. *The International Journal of Robotics Research*, 21(8) :679–699, 2002.
- [AM06] Nicolas Andreff and Philippe Martinet. Vision-based Kinematic Modelling of Some Parallel Manipulators for Control Purposes. In *1st European Conference on Mechanism Science (EuCoMeS)*, Obergurgl, Austria, 2006.
- [And05] Nicolas Andreff. Vision-based control of a gough-stewart parallel mechanism using legs observation. *International Conference on Robotics and Automation*, pages 677–687, 2005.
- [BA00] L. Baron and J. Angeles. The direct kinematics of parallel manipulators under joint-sensor redundancy. *IEEE Transactions on Robotics and Automation*, 16(1) :12–19, Feb 2000.
- [Bri17] S. Briot. Lectures notes on advanced dynamic modelling of robots. 2017.
- [CH01] P. I. Corke and S. A. Hutchinson. A new partitioned approach to image-based visual servo control. *IEEE Transactions on Robotics and Automation*, 17(4) :507–515, Aug 2001.
- [CH06] F. Chaumette and S. Hutchinson. Visual servo control, part i : Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4) :82–90, December 2006.
- [CH07] François Chaumette and S. Hutchinson. Visual servo control, part ii : Advanced approaches. pages 109–118, 2007.
- [Cha02] François Chaumette. Asservissement visuel. In W. Khalil, editor, *La commande des robots manipulateurs*, Traité IC2, chapter 3, pages 105–150. Hermès, 2002.
- [Cha04] François Chaumette. Image moments : A general and useful set of features for visual servoing. *IEEE Transactions on Robotics*, pages 713–723, 2004.
- [CHKA03] C. C. Cheah, M. Hirano, S. Kawamura, and S. Arimoto. Approximate jacobian control for robots with uncertain kinematics and dynamics. *IEEE Transactions on Robotics and Automation*, 19(4) :692–702, Aug 2003.
- [CHPV04] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino. Keeping features in the field of view in eye-in-hand visual servoing : a switching approach. *IEEE Transactions on Robotics*, 20(5) :908–914, Oct 2004.
- [Chr08] Collewet Christophe. *Vers des asservissements visuels sur des scènes inconnues*. Habilitation à diriger des recherches, Université Rennes 1, September 2008.
- [Dal07] Tej Dallej. *Contribution to a generic model for visual servoing of parallel robots using legs observation*. Theses, Université Blaise Pascal - Clermont-Ferrand II, December 2007.
- [DAMM06] T. Dallej, N. Andreff, Y. Mezouar, and P. Martinet. 3d pose visual servoing relieves parallel robot control from joint sensing. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4291–4296, Oct 2006.
- [DD92] Daniel F. Dementhon and Larry S. Davis. Model-based object pose in 25 lines of code. *Lecture Notes in Computer Science*, pages 335–343, 1992.

- [DRLR90] Michel Dhome, Marc Richetin, Jean-Thierry Lapresté, and Gerard Rives. Determination of the attitude of 3d objects from a single perspective view. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11 :1265–1278, 01 1990.
- [ECR90] Bernard Espiau, François Chaumette, and Patrick Rives. Une nouvelle approche de la relation vision-commande en robotique. Research Report RR-1172, INRIA, 1990.
- [GH07] N. R. Gans and S. A. Hutchinson. Stable visual servoing through hybrid switched-system control. *IEEE Transactions on Robotics*, 23(3) :530–540, June 2007.
- [HA94] K. Hosoda and M. Asada. Versatile visual servoing without knowledge of true jacobian. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 186–193 vol.1, Sep 1994.
- [HK93] Koichi Hashimoto and Hidenori Kimura. LQ Optimal and non linear approaches to visual servoing. In *Visual Servoing*, page 165–198. World Scientific, 10 1993.
- [ISO98] Iso 9283 :1998 – robots manipulateurs industriels — critères de performance et méthodes d’essai correspondantes. Technical report, International Organization for Standardization, 1998.
- [JFN97] M. Jagersand, O. Fuentes, and R. Nelson. Experimental evaluation of uncalibrated visual servoing for precision manipulation. In *International Conference on Robotics and Automation*, volume 4, pages 2874–2880, Apr 1997.
- [Kac18] L. Kaci. *Conception et commande d’un robot industriel en bois*. PhD thesis, Ecole Centrale de Nantes, 2018.
- [KBBM18] L. Kaci, S. Briot, C. Boudaud, and P. Martinet. Robecolo : Optimal design of a wooden five-bar mechanism. In *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE 2018)*, Quebec, Quebec, Canada, August 2018.
- [KBBP16] L. Kaci, S. Briot, C. Boudaud, and P. Martinet. Control-based design of a five-bar mechanism. In *Proceedings of EuCoMeS*, Nantes, France, 2016.
- [KC14] O. Kermorgant and F. Chaumette. Dealing with constraints in sensor-based robot control. *IEEE Transactions on Robotics*, 30(1), 2014.
- [Ker11] Olivier Kermorgant. *Fusion d’informations multi-capteurs en asservissement visuel*. Thèse de doctorat, Université de Rennes 1, 2011.
- [LDJC04] Jean-Thierry Lapresté, Michel Dhome, Frédéric Jurie, and François Chaumette. An efficient method to compute the inverse jacobian matrix in visual servoing. In *International Conference on Robotics and Automation*, 2004.
- [LS2] *LS2N site web*. <https://www.ls2n.fr>.
- [LVD98] J.-M Lavest, Marc Viala, and Michel Dhome. Do we really need an accurate calibration pattern to achieve a reliable camera calibration ? In *the 5th European Conference on Computer Vision*, pages 158–174, 06 1998.
- [Mal04] E. Malis. Improving vision-based control using efficient second-order minimization techniques. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1843–1848 Vol.2, April 2004.
- [MBR02] E. Malis, J. J. Borrelly, and P. Rives. Intrinsic-free visual servoing with respect to straight lines. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 384–389 vol.1, 2002.
- [MC07] Nicolas Mansard and François Chaumette. Task sequencing for high-level sensor-based control. *IEEE Transactions on Robotics*, 23 :60–72, 2007.

- [MCB99] E. Malis, F. Chaumette, and S. Boudet. 2d 1/2 visual servoing. *IEEE Transactions on Robotics and Automation*, 15(2) :238–250, Apr 1999.
- [Mer97] Jean-Pierre Merlet. *Les robots parallèles*. Hermès, 1997.
- [MH01] R. Mahony and T. Hamel. Visual servoing using linear features for under-actuated rigid body dynamics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium*, volume 2, pages 1153–1158 vol.2, 2001.
- [MMC01] Ezio Malis, Guillaume Morel, and François Chaumette. Robot control using disparate multiple sensors. *The International Journal of Robotics Research*, 20(5) :364–377, 2001.
- [MR93] Henri Michel and Patrick Rives. Singularities in the determination of the situation of a robot effector from the perspective view of 3 points. Research Report RR-1850, INRIA, 1993.
- [Nak90] Yoshihiko Nakamura. *Advanced Robotics : Redundancy and Optimization*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1990.
- [NK93] B. Nelson and P. K. Khosla. Increasing the tracking region of an eye-in-hand system by singularity and joint limit avoidance. In *IEEE International Conference on Robotics and Automation*, pages 418–423 vol.3, May 1993.
- [PKK93] N. P. Papanikolopoulos, P. K. Khosla, and T. Kanade. Visual tracking of a moving target by a camera mounted on a robot : a combination of control and vision. *IEEE Transactions on Robotics and Automation*, 9(1) :14–35, Feb 1993.
- [Plu65] Julius Plucker. On a new geometry of space. *Proceedings of the Royal Society of London*, page 53, 1865.
- [Ren03] Pierre Renaud. *Apport de la vision pour l'identification géométrique de mécanismes parallèles*. Thèse de doctorat, Université Blaise Pascal, Clermont Ferrand, 2003.
- [Ros15] Victor Rosenzweig. *Sensor-Based Design and Control of High-Speed Manipulators*. Thèse de doctorat, École Centrale, 2015.
- [SMB91] C. Samson, Le Borgne M., and Espiau B. Robot control : The task function approach. *Clarendon Press, Oxford*, 1991.
- [TC05] O. Tahri and F. Chaumette. Point-based and region-based image moments for visual servoing of planar objects. *IEEE Transactions on Robotics*, 21(6) :1116–1127, Dec 2005.
- [VBPC14] Alessia Vignolo, Sébastien Briot, Martinet Philippe, and Chao Chen. Comparative Analysis of two Types of Leg-observation-based Visual Servoing Approaches for the Control of the Five-bar Mechanism. In *Australasian Conference on Robotics and Automation (ACRA 2014)*, Melbourne, Australia, December 2014.
- [Vis] Visual Servoing Platform (ViSP). *Tutorial : How to boost your visual servo control law*. version 3.1.0. <http://visp-doc.inria.fr/doxygen/visp-daily/tutorial-boost-vs.html>.
- [YB17] Jia Yan-Bin. Plücker coordinates for lines in the space. <http://web.cs.iastate.edu/cs577/handouts/plucker-coordinates.pdf>, August 2017.