



HAL
open science

Modélisation et calibration pour une numérisation robotisée

Matthias Bordron

► **To cite this version:**

Matthias Bordron. Modélisation et calibration pour une numérisation robotisée. Automatique / Robotique. Université Paris Saclay (COMUE), 2019. Français. NNT : 2019SACLN014 . tel-02172323

HAL Id: tel-02172323

<https://theses.hal.science/tel-02172323>

Submitted on 3 Jul 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Modélisation et calibration pour une numérisation robotisée

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'École Normale Supérieure Paris-Saclay

École doctorale n°579 Sciences mécaniques et énergétiques,
matériaux et géosciences (SMEMAG)
Pôle : solides, structures, matériaux

Thèse présentée et soutenue à Cachan, le 6 juin 2019, par

Matthias Bordron

Composition du Jury :

Gérard Poisson Professeur des Universités Université d'Orléans (PRISME)	Président du jury
Faïz Ben Amar Professeur des Universités Sorbonne Universités (ISIR)	Rapporteur
Sébastien Remy Maître de Conférences HDR Université de Technologie de Troyes (LASMIS)	Rapporteur
David Daney Chargé de recherche HDR INRIA Bordeaux	Examineur
Luc Mathieu Professeur des Universités Université Paris-Saclay (LURPA, ENS Paris-Saclay)	Examineur
Olivier Bruneau Professeur des Universités Université Paris-Saclay (LURPA, ENS Paris-Saclay)	Directeur de thèse
Charyar Mehdi-Souzani Maître de Conférences Université Paris 13 (LURPA, ENS Paris-Saclay)	Co-Encadrant de thèse

A mes parents, Christine, Nordine, fans de ma vie

A mon frère Alexandre, avec qui je grandis

A ma moitié Carole, ensemble tout nous sourit

Remerciements

Le doctorat est l'ultime étape de la vie d'étudiant, mettant à l'épreuve l'endurance des participants. Ce marathon qui m'a pris plus de 3 années nécessite d'être bien entouré pour tenir la distance. Et sur ce point, je dois dire que j'ai eu la chance d'être soutenu par de nombreux sponsors que je tiens à remercier dans ces quelques lignes.

En premier lieu, je souhaite remercier le roboticien et le métrologue qui ont cru en moi du premier jour jusqu'à la soutenance. La liberté d'action qui m'a été laissée et l'appui que j'ai reçu quand cela était nécessaire ont rendu ces années riches et passionnantes malgré les contraintes qui m'étaient imposées. Mention spéciale pour Olivier qui a été attentif à mes baisses de moral, et qui a toujours défendu mes efforts face à ceux qui ne les percevaient pas. Je salue également au passage Sylvain qui m'a guidé sur les voies de la recherche étant petit caneton, avec mon ancienne partenaire de robotique Margot.

Je remercie Faïz Ben Amar et Sébastien Remy d'avoir rapporté mon manuscrit de thèse, Gérard Poisson d'avoir présidé ma soutenance, et enfin David Daney et Luc Mathieu d'être venus assister à ma soutenance. Je remercie tous les membres du jury pour leurs questions et remarques intéressantes dans les domaines de la robotique et de la numérisation, et pour le temps qu'ils ont consacré à mes travaux.

Bien sûr le LURPA qui m'a accueilli est un lieu de travail, mais c'est aussi un lieu de vie dans lequel rien ne serait possible sans cette ambiance bienveillante d'entraide et de camaraderie. En particulier, je tiens à remercier tous les doctorants avec qui j'ai partagé rires, discussions, repas, soirées, mais aussi mes problèmes. Je pense notamment aux soirées LAN du temps des anciens, avec ces chers PAB, El commandante jovial, NdM ou KViard. Je suis également heureux d'avoir pu présider et guider le CIVIL en tant que Duck, à l'instar de mon prédécesseur la cruelle Conquérante et de mon successeur le despotique [GOD]ineau, avec qui j'ai eu quelques conflits. Je souhaite d'ailleurs tous mes vœux de réussite au prochain meneur [Banana]parte ; les activités, traditions, sorties, même en dehors du cadre du CIVIL, sont des bouffées d'oxygène essentielles aux doctorants et donnent vie au laboratoire. En ce sens je remercie Minh mon cobureau pour m'avoir supporté, Clément pour ses nombreuses initiatives plus ou moins bonnes, Yann avec qui j'ai réalisé la dernière ligne droite, Yannick pour son espièglerie parfois bien placée, El gringo pour sa motivation toujours au top... Pour des raisons évidentes de flemme mais pour tenter de ne pas frustrer trop de monde, voici une liste non exhaustive des personnes qui ont également contribué au maintien de mon moral au cours de la thèse : Charly, Sif, Mathieu, David, chef, Baptiste, Jean Loup, Sylvain. Pour toute plainte sur un oubli, vous pouvez m'adresser un courrier écrit avant le 6 juillet 2019, qui vous sera facturé un japonais à volonté.

Tout cela n'aurait pas non plus été possible sans les efforts combinés des trois piliers de ce laboratoire : l'infatigable Philippe, l'engagé Marc, et l'arrangeante Claudine.

Dans un autre registre, je vais remercier mes amis qui m'ont soutenu durant ces années et qui m'ont permis de faire des pauses au cours des vacances. Là aussi la liste est longue mais ils ont tous participé à leur façon à garder mon moral au beau fixe. Merci aux Béglais de toujours : Cyril, Pierre, Antoine, Adrien et Soso, merci aux camarades du trio Bordelais : Arnaud et Manu, et merci à ce cher Christopote.

Enfin, je tiens à remercier ma famille, papa, maman, Alex, qui sont venus de loin pour m'aider dans les préparatifs du pot de ma soutenance, mais pas seulement. Durant toutes les années qu'a duré le doctorat vous avez été là pour me soutenir, en venant à Paris, par téléphone ou au cours de mes vacances de retour dans le foyer familial. Comme tout enfant je souhaitais faire la fierté de mes parents, comme tout grand frère je voulais être un exemple, mais par-dessus tout vous me manquiez et venir vous voir dans le sud avait quelque chose d'apaisant dans le tumulte de la thèse.

À vous tous je dis merci, merci pour tout. Mais un merci ne suffit pas pour toi. Il est difficile de décrire avec des mots à quel point ta présence dans ma vie a été importante, surtout pour cette épreuve. Tu as été là pour moi, dans mes colères, mes nuits blanches, mes peines, mes doutes. Tu me donnais le sourire, me supportais en me préparant de bons petits plats quand je rentrais tard, m'aidais à me lever le matin. Bref tu vivais avec moi, avec douceur et attention. Je t'aime Carole.

« La science, ce sont d'abord des hommes et des femmes qui cherchent. »
Professeur Thibault (Alain Chabat), Les Nuls.

Sommaire

Remerciements	i
Sommaire	iii
Table des matières	iv
Liste des figures	vii
Liste des tableaux	x
Nomenclature	xi
Introduction générale	1
Chapitre 1 : Méthodologie de génération de trajectoire pour la numérisation robotisée	5
Chapitre 2 : Identification géométrique générique basée sur le mouvement	39
Chapitre 3 : Identifiabilité sélective des paramètres non géométriques	75
Chapitre 4 : Cartographies volumétriques prédictives des performances du robot	115
Chapitre 5 : Mise en place d'une numérisation robotisée	135
Chapitre 6 : Conclusion et perspectives	153
Bibliographie	157
Annexes	165
Résumé en Anglais	173
Résumé en Français	174

Table des matières

Chapitre 1	Méthodologie de génération de trajectoire pour la numérisation robotisée	5
1.1	Introduction et contexte	5
1.1.1	Origine du besoin	5
1.1.2	Matériel et cellule robotisée	7
1.2	Stratégie de génération de trajectoire pour la numérisation robotisée	10
1.3	Approches pour la modélisation et l'identification de défauts d'un robot sériel	12
1.3.1	Généralités sur la modélisation des robots	12
1.3.2	Les modèles géométriques	14
1.3.3	Défauts non géométriques	17
1.3.4	Modèle élasto-géométrique	20
1.3.5	Sélection d'un modèle	22
1.3.6	Identification classique des paramètres d'un modèle	25
1.3.7	Méthode d'identification basée sur le mouvement : CPA	27
1.4	Exploitation des performances du robot	29
1.4.1	Indices basés sur la jacobienne	30
1.4.2	Indices basés sur des mesures ou des caractéristiques du robot	31
1.5	Mise en place de la cellule de numérisation	33
1.5.1	Capteur laser de numérisation	34
1.5.2	Système de suivi externe	35
1.6	Synthèse du chapitre	36
Chapitre 2	Identification géométrique générique basée sur le mouvement	39
2.1	Sélection et distinction des paramètres utiles	39
2.2	Identification des paramètres par une méthode classique	43
2.2.1	Choix d'une méthode classique	43
2.2.2	Jacobienne et résolution par la méthode classique	44
2.2.3	Résultats sur l'identification des paramètres géométriques par la méthode classique	45
2.3	Méthode d'identification par analyse continue d'arcs réduits	46
2.3.1	Adaptation de la méthode CPA à notre contexte : proposition de la méthode ANCAR	46
2.3.2	La méthode ANCAR globale	48
2.4	Optimisation de l'identification par la méthode ANCAR	50
2.4.1	Problématiques sur la fiabilité des paramètres identifiés par ANCAR	50
2.4.2	Amélioration de l'étape d'optimisation	52
2.4.3	Correction des distorsions sur la mesure du C-Track	53
2.4.4	Paramètres identifiés par la méthode ANCAR locale optimisée	56
2.5	Méthode ANCAR résolue avec une jacobienne	57

2.5.1	Mise en place d'une modélisation adaptée à notre méthode	57
2.5.2	Détails du calcul de la matrice jacobienne pour la méthode ANCAR	58
2.5.3	Corrections sur la résolution de la méthode ANCAR avec jacobienne.....	61
2.5.4	Adaptabilité de la jacobienne	62
2.5.5	Identification des paramètres par ANCAR, avec jacobienne.....	63
2.5.6	Apports d'une jacobienne dans notre méthode ANCAR	64
2.6	Comparaison et validation des paramètres identifiés	65
2.6.1	Estimations des temps de mesure et de résolution	66
2.6.2	Fiabilité théorique sur les paramètres identifiés.....	66
2.6.3	Fiabilités observées sur l'identification des paramètres.....	68
2.6.4	Justesse relative et absolue sur le positionnement du capteur laser	70
2.6.5	Conclusions sur l'étude comparative et la validation des paramètres identifiés	73
2.7	Conclusion sur la modélisation géométrique	73
Chapitre 3	Identifiabilité sélective des paramètres non géométriques	75
3.1	Contexte et estimation du besoin d'une modélisation enrichie du robot.....	75
3.1.1	Modélisation et indices de performance.....	76
3.1.2	Modélisation et trajectoire commande	76
3.1.3	Modélisation et suivi de l'effecteur.....	77
3.2	Intégration de paramètres non géométriques.....	81
3.2.1	Intégration des paramètres non géométriques au modèle existant	82
3.2.2	Modèle des élasticités et jeux dans une articulation.....	84
3.2.3	Modélisation des articulations 1 à 6 du robot.....	88
3.2.4	Modélisation élastique d'un segment.....	94
3.2.5	Modélisation élastique des segments 1 à 6 du robot	96
3.2.6	Modèle élasto-géométrique complet du robot.....	100
3.3	Identifiabilité des paramètres et simplification	100
3.3.1	Méthode d'identification pour paramètres élastiques.....	101
3.3.2	Calcul de la jacobienne du modèle élasto-géométrique	104
3.3.3	Identifiabilité	109
3.4	Conclusion sur la modélisation élasto-géométrique.....	113
Chapitre 4	Indices de performances pour la qualification du robot.....	115
4.1	Indices de performance et contrôle de la vitesse	115
4.1.1	Insuffisance des indices de vitesse usuels	116
4.1.2	Proposition d'indices de vitesse pour la génération de trajectoire	118
4.1.3	Exemple de trajectoire d'ébauche et cas singulier	119
4.1.4	Calcul des indices de vitesse moyenne et minimale.....	120
4.1.5	Perspectives d'évolution.....	123

4.2	Contrôle de la qualité de pose	123
4.2.1	Définition d'un ellipsoïde de répétabilité	124
4.2.2	Définition d'un indice pour la répétabilité.....	126
4.2.3	Identification des répétabilités des articulations et perspectives	128
4.3	Limites en mouvement	128
4.3.1	Disponibilité articulaire sur trajectoire entière	128
4.3.2	Barrières physiques et immatérielles.....	130
4.4	Regroupement des indices de performance	131
4.5	Synthèse sur les indices exploités.....	133
Chapitre 5	Calibration du capteur laser plan dans la cellule robotisée.....	135
5.1	Matériel et communication entre composants	135
5.2	Calibration entre modèle de cibles et capteur laser	138
5.2.1	Énoncé du problème et compensation	138
5.2.2	Calcul de la matrice complémentaire	139
5.2.3	1 ^{ère} méthode : calibration rapide sur robot.....	141
5.2.4	2 ^{ème} méthode : calibration sur MMT.....	145
5.3	Evaluation qualitative des méthodes de calibration présentées	147
5.4	Synthèse du chapitre.....	150
Chapitre 6	Conclusion et perspectives	153

Liste des figures

Figure 1 : Cellule de numérisation	2
Figure 2 : Eléments de la cellule de numérisation robotisée	7
Figure 3 : Interfaces robot/Zephyr (à gauche) et robot/Pièce (à droite)	8
Figure 4 : Capteur laser Zephyr doté de cibles de suivi	8
Figure 5 : C-Track et cible réfléchissante	8
Figure 6 : Stratégie de génération de trajectoires	11
Figure 7 : Défauts d'un segment i	14
Figure 8 : Défauts d'une liaison entre deux segments	14
Figure 9 : Paramétrage DH	15
Figure 10 : DH avec paramètre de Hayati	15
Figure 11 : Schéma modélisation VJM	21
Figure 12 : Convergence de la pose déformée du robot	22
Figure 13 : Axes robot identifiés par CPA	28
Figure 14 : Principe méthode CPA	28
Figure 15 : Ellipsoïde de vitesses	30
Figure 16 : Repérage C-Track	35
Figure 17 : Descriptions des objectifs requis dans la mise en place de la numérisation robotisée	37
Figure 18 : Modélisation DH modifiée	40
Figure 19 : Modélisation géométrique complète du robot	41
Figure 20 : Repérage et segments définis par la modélisation DH modifiée	42
Figure 21 : Corrections pour paramètres du robot, 5 essais par méthode classique	46
Figure 22 : Les 3 jeux de cibles sur l'effecteur	46
Figure 23 : Méthode d'identification par ANCAR	47
Figure 24 : Protocole ANCAR	48
Figure 25 : reconstruction d'un grand arc à partir de plusieurs positions du C-Track	49
Figure 26 : Axes i et $i+1$ dans le repère des cibles pour la méthode ANCAR globale	49
Figure 27 : Arcs mesurés lors d'une calibration ANCAR locale	50
Figure 28 : Corrections pour paramètres propres au robot, ANCAR (locale) sans amélioration	51
Figure 29 : Corrections pour paramètres du robot, ANCAR (locale) avec optimisation des moindres carrés	53
Figure 30 : Distance entre 2 ensembles de cibles lors de la rotation d'un axe	53
Figure 31 : Modèle rigide basé sur la position des ensembles de cibles	54
Figure 32 : Modèle rigide basé sur la pose (position et orientation) des ensembles de cibles	54
Figure 33 : Arrangement des cibles mesurées pour identification	54
Figure 34 : Corrections pour paramètres du robot, ANCAR (locale) avec hypothèse de solide rigide	55
Figure 35 : Orientation du solide fixée par la rotation	55
Figure 36 : Corrections pour paramètres du robot, ANCAR (locale) avec hypothèse de solide rigide et d'orientation contrainte	56
Figure 37 : Corrections pour paramètres du robot, ANCAR locale optimisée, après 3 mois	57
Figure 38 : Différence entre point mesuré et point théorique, résolution ANCAR	61
Figure 39 : Corrections pour paramètres du robot, ANCAR locale optimisée avec jacobienne, 10 itérations	64
Figure 40 : Corrections pour paramètres du robot, ANCAR locale optimisée avec jacobienne, 500 itérations	64
Figure 41 : Corrections pour paramètres du robot, ANCAR locale	68
Figure 42 : Corrections pour paramètres du robot, ANCAR locale optimisée	68
Figure 43 : Corrections pour paramètres du robot, ANCAR locale optimisée, 3 mois après	69
Figure 44 : Corrections pour paramètres du robot, méthode classique	69
Figure 45 : Génération des points d'arrivée pour trajet de justesse relative	70

Figure 46 : Setup validation distance suivant verticale	71
Figure 47 : Influence de la distance de numérisation sur le bruit de numérisation [Mehdi-Souzani et al. 2016].....	77
Figure 48 : Influence de l'orientation du capteur sur le bruit de numérisation [Mehdi-Souzani et al. 2016].....	77
Figure 49 : Erreur due à la différence entre la position réelle du capteur et celle donnée par le modèle	78
Figure 50 : Etablissement du modèle élasto-géométrique	81
Figure 51 : Description segments et articulations pour détermination d'un torseur d'efforts.....	83
Figure 52 : Schéma transversal d'une articulation sous efforts.....	86
Figure 53 : Efforts en jeu sur l'articulation 2.....	89
Figure 54 : Efforts en jeu sur l'articulation 4.....	90
Figure 55 : Efforts en jeu sur l'articulation 6.....	92
Figure 56 : Schéma d'une poutre chargée.....	94
Figure 57 : Segment 4 du modèle géométrique et segments 4 et 5 réels.....	98
Figure 58 : Interface pour chargement du robot	101
Figure 59 : Interface de chargement.....	101
Figure 60 : Retournement à 180° de l'axe 1 et inversion du jeu $b\theta r1$	102
Figure 61 : Inversion du jeu $b\theta r1$ dans l'articulation 1	102
Figure 62 : Itérations pour l'identification des paramètres géométriques et élastiques.....	103
Figure 63 : Etablissement de la jacobienne totale élastique	105
Figure 64 : Influence maximale de chaque paramètre sur la pose des cibles (échelle logarithmique) 112	
Figure 65 : Cartographie de la manipulabilité, orientation vers le haut et coude haut	116
Figure 66 : Cartographie de la manipulabilité, orientation vers la gauche et coude bas	116
Figure 67 : Cartographie de manipulabilité, modèle théorique modifié.....	117
Figure 68 : Différence entre les figures 65 et 67.	117
Figure 69 : Exemple de trajectoire en 3 passes pour numérisation	119
Figure 70 : Trajectoire simple, même posture.....	120
Figure 71 : Trajectoire avec changement de posture	120
Figure 72 : Indice de vitesse minimale orientation 1, en m/s	121
Figure 73 : Indice de vitesse minimale orientation 2, en m/s	121
Figure 74 : Indice de vitesse moyenne avec discontinuité et orientation 1, en m/s.....	122
Figure 75 : Indice de vitesse moyenne avec discontinuité et orientation 2, en m/s.....	122
Figure 76 : Indice de répétabilité maximale pour 1 ^{ère} orientation, en mm	127
Figure 77 : Indice de répétabilité maximale pour 2 ^{ème} orientation, en mm	127
Figure 78 : Indice d'accessibilité pour 1 ^{ère} orientation, entre 0 et 1	129
Figure 79 : Indice d'accessibilité pour 2 ^{ème} orientation, entre 0 et 1	129
Figure 80 : Répétabilité maximale pour 1 ^{ère} orientation et domaine limité, en mm	130
Figure 81 : Indice de vitesse minimale pour 1 ^{ère} orientation et domaine limité, en m/s.....	130
Figure 82 : Répétabilité maximale pour 1 ^{ère} orientation et domaine plus limité, en mm	130
Figure 83 : Indice de vitesse minimale pour 1 ^{ère} orientation et domaine plus limité, en m/s	130
Figure 84 : Zone admissible pour la numérisation sous contrainte de la pièce, 1 ^{ère} orientation.....	131
Figure 85 : Zone admissible pour la numérisation sous contrainte de la pièce, 2 ^{ème} orientation.....	131
Figure 86 : Éléments de la cellule de numérisation.....	136
Figure 87 : Interface transmettant la position et l'orientation de l'ensemble de cibles, donnée par le C-Track ou le robot	137
Figure 88 : Numérisation du visage, vue isométrique	137
Figure 89 : Visage usiné servant de pièce test.....	137
Figure 90 : Matrice complémentaire dans notre interface et calculs internes à Polygonia	138
Figure 91 : Points acquis et trace laser dans le repère externe, identification de Q_{com}	142
Figure 92 : Points acquis et trace laser dans le repère externe, identification de Q_{com2}	142

Figure 93 : Numérisation du plan étalon sans notre matrice complémentaire, vue XY en mm	143
Figure 94 : Numérisation du plan étalon sans notre matrice complémentaire, vue ZY en mm	143
Figure 95 : Numérisation du plan étalon avec notre première rotation complémentaire Q_{com} , vue XY en mm.....	143
Figure 96 : Numérisation du plan étalon avec notre première rotation complémentaire Q_{com} , vue ZY en mm.....	143
Figure 97 : Numérisation du plan étalon incliné à 45° avec Q_{com} , vue XY en mm.....	143
Figure 98 : Numérisation du plan étalon incliné à 45° avec Q_{com} , vue isométrique en mm.....	143
Figure 99 : Numérisation du plan étalon incliné à 45° avec T_{com} complet, vue XY en mm	144
Figure 100 : Numérisation du plan étalon incliné à 45° avec T_{com} complet, vue isométrique en mm	144
Figure 101 : Visage numérisé avec notre matrice complémentaire, vue de côté en mm	144
Figure 102 : Visage numérisé avec notre matrice complémentaire, vue isométrique en mm.....	144
Figure 103 : Remplissage de la fenêtre par la ligne laser.....	146
Figure 104 : Numérisation du cylindre incliné, mm	146
Figure 105 : Visage numérisé avec 2 ^{ème} méthode de calibration, vue de côté en mm	147
Figure 106 : Visage numérisé avec 2 ^{ème} méthode de calibration, vue isométrique en mm.....	147
Figure 107 : Orientations du plan étalon (et du Zephyr) pour numérisation.....	148
Figure 108 : Bruit de numérisation, fonction de la configuration, de la vitesse et de la méthode	149
Figure 109 : Marches calibrées : artéfact pour l'estimation de la justesse	149
Figure 110 : Marches 1 à 4 et distances au plan de référence	149
Figure 111 : Justesse sur la hauteur des marches de l'artéfact, pour les différentes méthodes et vitesses	150
Figure 112 : Justesse sur la hauteur des marches de l'artéfact, pour les différentes configurations ...	150
Figure 113 : Erreur de mouvement radial	165
Figure 114 : Erreur de mouvement axial.....	165
Figure 115 : Erreur de mouvement d'inclinaison.....	165
Figure 116 : Erreur de positionnement angulaire.....	165
Figure 117 : Calcul de l'incertitude sur l'estimation d'un axe du robot	166
Figure 118 : Convergence de l'incertitude sur l'estimation du centre, par rapport au nombre de points sur l'arc	166
Figure 119 : Centres identifiés pour 1000 simulations.....	167
Figure 120 : Intersection des médiatrices de 2 arcs.....	167
Figure 121 : Centres identifiés pour 1000 simulations, à partir de l'intersection des médiatrices.....	167
Figure 122 : Communication VXelements, Polygonia	169
Figure 123 : Pièce test, visage	170
Figure 124 : Numérisation du visage, vue isométrique.....	170
Figure 125 : Numérisation du visage, vue de dessus (plan XZ).....	170
Figure 126 : Numérisation visage, avec correction grossière de la pose du capteur par rapport aux cibles	171

Liste des tableaux

Tableau 1: Qualification des moyens de mesure (bon, moins bon, mauvais/incapacité)	6
Tableau 2 : Influence des sources d'erreur d'une opération robotisée [Damak 1996].....	13
Tableau 3 : Nombre de paramètres DH de passage pour un robot à i axes	15
Tableau 4 : Nombre de paramètres et particularités des modèles géométriques	16
Tableau 5 : Description de modèles de paramètres non géométriques.....	19
Tableau 6 : Critères pour la sélection de configurations de mesure	26
Tableau 7 : Indices de performance donnant les capacités de l'effecteur pour une configuration donnée	33
Tableau 8 : Les 18 paramètres géométriques théoriques propres au robot (en mm et °).....	42
Tableau 9 : Corrections sur les paramètres de distance propres au robot par méthode classique (mm).....	46
Tableau 10 : Corrections sur les paramètres angulaires propres au robot par méthode classique (10^{-3} °)	46
Tableau 11 : Corrections sur les paramètres théoriques de distance propres au robot (mm)	56
Tableau 12 : Corrections sur les paramètres théoriques angulaires propres au robot (10^{-3} °).....	56
Tableau 13 : Temps requis pour les différentes méthodes (mesure et calcul).....	66
Tableau 14 : Incertitudes sur les paramètres de distance avec méthode CPA modifiée et méthode classique	67
Tableau 15 : Incertitudes sur les paramètres angulaires avec méthode CPA modifiée et méthode classique	67
Tableau 16 : Écarts en mm entre la distance commandée et les distances parcourues avec les différents modèles.....	71
Tableau 17 : Écarts moyens et écarts types entre les poses commandées et les poses mesurées pour chaque modèle	72
Tableau 18 : Paramètres élastiques et de jeu des articulations du robot.....	94
Tableau 19 : Mesures de la déformée de l'extrémité de chaque segment seul du robot	96
Tableau 20 : Paramètres non géométriques des segments du robot	100
Tableau 21 : Ordres de grandeur des paramètres d'une articulation i.....	110
Tableau 22 : Ordres de grandeur des paramètres d'un segment i, et masses	110
Tableau 23 : Répétabilité brutes des axes du robot, en 10^{-3} °.....	126
Tableau 24 : Répétabilité identifiées des axes du robot, en 10^{-3} °	127

Nomenclature

Notations des paramètres

- θ_i : coordonnée articulaire de l'axe i / articulation i / liaison i .
- w_i : $i^{\text{ème}}$ paramètre d'un modèle.
- $d_i, \alpha_i, r_i, \delta\theta_i$: paramètres entre un axe i et un axe $i+1$ (en convention DH modifiée) ou entre un axe $i-1$ et un axe i (en convention DH).
- k_{txt} : paramètre de raideur, par exemple k_{t3} est le paramètre de raideur en torsion associé au segment 3.
- s_{txt} : paramètre de souplesse (associé à un paramètre de raideur).
- b_{txt} : paramètre de jeu, par exemple $b_{\theta z3}$ est le paramètre de jeu angulaire de la liaison 3.
- m_i : paramètre de masse d'un segment i .
- σ_{txt} : écart type, par exemple σ_5 est la valeur d'écart type associée à la répétabilité de mise en position de l'articulation 5 du robot.
- λ_i : $i^{\text{ème}}$ valeur propre associée à une matrice.
- cst, ρ, c, \dots : constantes.

Notation des matrices

- M^+, M^t, M^{-1} : notations de matrice désignant respectivement la pseudo-inverse, la transposée et l'inverse de la matrice M .
- $T_{1,2}$: matrice homogène de passage d'un élément 1 à un élément 2, de dimension 4×4 .
- $Td_{i,i+1}$: matrice homogène de passage d'un axe i à un axe $i+1$ en convention DH ou DH modifié, dimension 4×4 .
- $Th_{i,i+1}$: notations de matrice désignant respectivement la pseudo-inverse, la transposée et l'inverse de la matrice M , 4×4 .
- $\text{Trans}(z_i, w_i)$: matrice homogène de passage, effectuant une translation de direction z_i et de longueur w_i , 4×4 .
- $\text{Rot}(z_i, w_i)$: matrice homogène de passage, effectuant une rotation de direction z_i et d'angle w_i , 4×4 .
- $T_{\text{élast}}$: matrice homogène de passage, intégrant des paramètres non géométrique d'un modèle, par exemple $T_{\text{élastart2}}$ comprend les paramètres non géométrique de l'articulation 2, 4×4 .
- K_x, K_θ : matrices de rigidité du robot respectivement dans le domaine cartésien et articulaire, 6×6 .
- $Q_{R_1, R_2}, Q(z_i, w_i)$: matrices de rotation respectivement d'un repère R_1 à un repère R_2 , ou suivant une direction z_i et un angle w_i , 3×3 .

Notations des Jacobiennes

- J_θ : matrice jacobienne articulaire liant les 6 mouvements sur les articulations du robot à la pose de l'effecteur, 6×6 .
- J_w : matrice jacobienne paramétrique liant les g mouvements sur les paramètres du robot à la pose de l'effecteur, $6 \times g$.
- J_{w_i} : colonne de J_w correspondant au paramètre w_i , 6×1 .
- J_{tot_w} : jacobienne paramétrique totale sur m configurations de mesure, concaténant plusieurs jacobienes paramétriques J_w , $(6m) \times g$.
- $J_{w,p}$: partie supérieure de la jacobienne, uniquement focalisée sur le positionnement de l'effecteur, $3 \times g$.
- $J_{w,o}$: partie inférieure de la jacobienne, uniquement focalisée sur l'orientation de l'effecteur, $3 \times g$.

- J_n, \bar{J} : jacobiennes respectivement normalisée avec le condition number ou adimensionnée.
- $J_{w,o}$: partie inférieure de la jacobienne, uniquement focalisée sur l'orientation de l'effecteur, $3 \times g$.
- J_{Ai} : jacobienne paramétrique obtenue par notre méthode ANCAR, liant les g mouvements sur les paramètres à la pose d'un axe i du robot, $6 \times g$.
- J_A : jacobienne totale obtenue par ANCAR, pour les 6 axes du robot, $(6 \times 6) \times g$.
- J_A' : jacobienne totale obtenue par ANCAR et corrigée, pour les 6 axes du robot, $(6 \times 5) \times g$.
- J_{mix}' : jacobienne totale par ANCAR corrigée, pour les 6 axes et/ou d'autres entités géométriques, $(m \times 5) \times g$.

Notations des vecteurs et n-uplets

- P : pose de l'effecteur / de l'outil (position et orientation), 6×1 .
- dP : variation sur la pose P , 6×1 .
- P_p, P_o : respectivement la partie position de la pose et la partie orientation, 3×1 .
- P_{tot} : ensemble de m poses de m configurations de mesure, $(6 \times m) \times 1$.
- θ : coordonnées articulaires des 6 articulations du robot, 6×1 .
- $d\theta$: variation sur les coordonnées θ , 6×1 .
- $\dot{\theta}$: vitesses des 6 articulations du robot, 6×1 .
- w : vecteur des g paramètres du modèle, $g \times 1$.
- dw : variation sur les paramètres w , $g \times 1$.
- \dot{w} : vitesses des g paramètres du modèle, $g \times 1$.
- q : charges sur l'effecteur (forces et moments), 6×1 .
- F : vecteur des forces, 3×1 .
- M : vecteur des forces, 3×1 .
- Γ : couples sur les 6 articulations du robot, 6×1 .
- T : vecteur de translations, 3×1 .
- R : vecteur de rotations, 3×1 .
- V : vecteur vitesses de translation, 3×1 .
- Ω : vecteur vitesses de rotation, 3×1 .
- Pt_i : coordonnées d'un point Pt_i , 3×1 .
- AB : vecteur entre les points A et B, 3×1 .
- Z : direction ou axe, 3×1 .
- \bar{Z} : direction ou axe sur les figures, 3×1 .

Autres notations

- I_{txt} : valeur associée à un indice de performance, par exemple $I_{V_{moy}}$ donne la valeur de l'indice de vitesse moyenne.

Rext, Rcibles, ... : repères utilisés dans la cellule robotisée.

Introduction générale

Les premiers robots sériels de grandes dimensions ont été développés pour réaliser des opérations répétitives sur des chaînes de production, notamment dans le domaine de l'automobile. Leur implantation a permis d'augmenter les cadences de production d'une part en proposant un travail quasi-continu avec peu de temps d'arrêt de la chaîne, et d'autre part en diminuant les durées des opérations. Ils sont principalement utilisés pour des opérations de manutention, de soudage ou de peinture, pour lesquelles leur bonne répétabilité (de l'ordre du dixième de millimètre) est suffisante et pour lesquelles une bonne précision absolue sur la pose de l'effecteur n'est pas nécessaire. Ce type d'opérations pourrait être assuré plutôt par des machines spécialisées à la place de robots, ce qui donnerait de meilleurs résultats et un meilleur rendement. Mais le dernier grand avantage des robots anthropomorphes c'est la flexibilité qu'ils apportent à la chaîne de production. Un robot est facilement adaptable ; en fonction de l'outil qui lui est associé il peut effectuer de nombreuses tâches différentes et donc convertir la chaîne de production pour de nouvelles pièces ou un nouveau procédé.

Avec les avancées technologiques, les chaînes de production comportent de plus en plus d'étapes et d'opérations pour produire des pièces plus complexes. De nouvelles opérations demandant beaucoup plus de précision font alors leur apparition. Mais dans le même temps, les chaînes doivent être toujours plus automatisées et flexibles. Les robots sont donc toujours plus sollicités et leur faible précision absolue devient problématique vis-à-vis de la qualité requise par ces nouvelles opérations. Même leur répétabilité réputée bonne doit être prise en compte avec attention. Pour répondre à ce besoin, les fabricants de robots (Stäubli, Kuka,...) proposent alors des robots aux technologies innovantes de plus en plus performants d'année en année. Cependant, les industries clientes de ces fabricants, ne peuvent pas renouveler aussi régulièrement leur parc de robots avec des robots plus récents. Beaucoup de travaux se penchent donc sur une amélioration et une optimisation de l'utilisation de robots moins récents, comme ceux que l'on retrouve en industrie [Olabi 2011, Marie et al. 2013, Guillo et Dubourg 2016, Vosniakos et Matsas 2010, Lin et al. 2017, Zargarbashi et al. 2012, Zhang et al. 2016]. Par ailleurs, même les robots les plus récents ne sont pas exempts de défauts et requièrent des améliorations pour satisfaire les besoins de ces nouvelles opérations complexes.

Une opération de plus en plus présente dans les chaînes de production, et de plus en plus recherchée par les industriels, est le contrôle des pièces au cours du cycle de production. Ce contrôle permet de limiter les rebuts et les pertes en détectant au plus tôt des problèmes de réglage de la chaîne. Cela permet également de retravailler une pièce mauvaise en connaissant ses défauts ou même de faire de l'appairage de pièces pour de petites séries. On retrouve donc de nombreuses étapes de mesure sur la chaîne, qui sont pour la plupart manuelles et échantillonnées. Les opérateurs doivent bien souvent extraire une pièce de la chaîne pour la contrôler avec du matériel adapté, comme une machine de mesure tridimensionnelle (MMT). C'est un procédé long et contraignant qui ne permet de contrôler que peu de pièces pour ne pas trop impacter la cadence de production. La volonté d'automatiser toujours plus les procédés amène les industriels à chercher des solutions pour mesurer les pièces directement sur la chaîne. Une des solutions actuellement développée, étant à la fois pratique et flexible, est d'utiliser un robot pour effectuer une numérisation des pièces directement sur la chaîne de production (Metrologic, Perception, Kréon,...) [Phan et al. 2018, Larsson et Kjellander 2006, Brosed et al. 2010, Sansoni et al. 2009, Santolaria et al. 2009, Sharifzadeh et al. 2018].

L'objet des travaux de cette thèse est de réaliser une numérisation utilisant un robot sériel comme système support de la numérisation. La mise en place de cette numérisation (installation et calibration) doit être facile, pratique, rapide et adaptative pour pouvoir être réalisée sur chaîne de production. En effet nous rappelons que le but est de gagner du temps lors de l'automatisation de la mesure de pièces sur chaîne en milieu industriel. Bien sûr la qualité de la numérisation, dépendante des performances du

robot qui sert de support, est aussi un facteur important à prendre en compte. Nous allons donc chercher à améliorer et à optimiser l'utilisation d'un robot pour obtenir une meilleure qualité de positionnement du système support de la numérisation. Dans ce but nous nous aiderons de modèles du comportement du robot et de cartographies de ses capacités.

Les travaux de cette thèse sont également motivés par le besoin d'une cellule de numérisation robotisée au sein du laboratoire d'accueil LURPA de l'ENS Paris Saclay. Notre laboratoire s'est spécialisé dans la cotation fonctionnelle, la mesure de pièces de tout type, et l'étude de pièces larges et d'assemblages d'aéronautique. Le LURPA est également spécialisé dans l'usinage de pièces complexes sur machine 5 axes, et s'est récemment investi dans l'impression 3D de plastique et de métal. Pour valider les recherches effectuées, les pièces complexes réalisées par impression 3D ou en usinage classique, ainsi que des pièces encombrantes d'aéronautique, doivent être mesurées. Le laboratoire dispose d'une MMT pour ces mesures, mais il reste difficile et parfois impossible de correctement numériser des pièces complexes ou encombrantes. En effet la MMT possède un faible volume de mesure ce qui limite les dimensions des pièces mesurées. De plus l'orientation de la tête de mesure est discrète ce qui rend impossible une réorientation continue de la tête pour une trajectoire de mesure adaptée aux formes complexes d'une pièce [Phan et al. 2018]. La mise en place d'une cellule de numérisation (figure 1) utilisant un robot comme système support permet de répondre à ces besoins. Grâce à l'envergure du robot lui conférant un vaste domaine accessible, il est possible d'effectuer la numérisation de pièces encombrantes. Les 6 degrés de liberté (DDL) offerts par le robot permettent de maîtriser l'orientation et la position de l'effecteur du robot. Associés à un capteur laser pour la numérisation il est alors possible d'avoir une réorientation continue du capteur lors de la numérisation. Cette association permet donc de travailler sur une trajectoire adaptée aux contours d'une pièce complexe, donnant le meilleur résultat de mesure possible.



Figure 1 : Cellule de numérisation

Les objectifs principaux qui vont conduire les travaux sont donc : la mise en place d'une numérisation robotisée pratique, rapide et flexible pour répondre aux besoins industriels, un positionnement de qualité du capteur laser vis-à-vis du niveau de détails requis par l'opération de numérisation, la génération d'une trajectoire adaptée au profil de la pièce numérisée (pièce complexe ou encombrante). Ces objectifs nous amènent à articuler les travaux de cette thèse autour de cinq chapitres :

- Une analyse de l'existant et l'expression des besoins dans la cellule qui nous conduiront à une génération originale de trajectoire adaptée et optimale.

- Une modélisation géométrique du robot ainsi qu'une méthode d'identification des paramètres pratique et rapide.
- Une modélisation élastique sélectionnée, ayant un niveau de détail adapté à la qualité requise sur l'opération de numérisation (qualité du positionnement).
- Une étude des capacités du robot à l'aide d'indices de performances originaux, permettant d'optimiser l'utilisation de ce dernier (vitesse, qualité, accessibilité).
- Une présentation de l'intégration des différents éléments dans la cellule de numérisation (robot, capteur laser), et de la calibration de la cellule.

Le chapitre 1 est dédié à la description des différentes méthodes et stratégies qui seront mises en place pour aboutir à cette cellule de numérisation, tout en respectant les contraintes présentées précédemment (pratique, rapide, qualité du résultat). Ce chapitre s'appuie sur un état de l'art concernant les différents aspects à traiter pour cette mise en place et permettra ainsi de situer nos travaux. Dans un premier temps le matériel de la cellule est présenté ainsi que son fonctionnement. Puis la stratégie d'ensemble utilisée pour générer les trajectoires de numérisation est décrite. Elle est basée sur la CAO de la pièce à numériser, des contraintes choisies, une modélisation fidèle du robot et des critères de performance. C'est la raison pour laquelle nous proposons dans les chapitres 2 et 3 du manuscrit, des méthodes pour parvenir à une modélisation du robot, ainsi qu'une sélection de critères de performances décrivant les capacités du robot dans le chapitre 4.

Dans ce premier chapitre, notre contribution est une stratégie originale de génération de trajectoire visant à exploiter au mieux les performances du robot et du capteur laser.

Le chapitre 2 décrit la construction, l'identification et l'utilisation du modèle de base du robot : le modèle géométrique. Cette première modélisation essentielle doit bien représenter le comportement du robot, c'est pourquoi une attention particulière est donnée à la sélection des paramètres du modèle. En partant de la méthode d'identification dite CPA (Circle Point Analysis), nous proposons une méthode d'identification originale des paramètres à la fois rapide et pratique. Nous comparons notre méthode avec une méthode d'identification plus classique. Enfin, nous présentons les résultats d'essais permettant de valider le modèle géométrique établi et les paramètres géométriques identifiés avec notre méthode.

Dans ce chapitre, la contribution principale est la généralisation et l'amélioration d'une méthode d'identification (CPA), pour une identification rapide et pratique des paramètres du modèle géométrique. Les paramètres géométriques obtenus par cette nouvelle méthode permettent d'améliorer le positionnement du robot.

Dans le chapitre 3, nous poursuivons sur la modélisation du robot en intégrant les notions d'élasticité et de jeu dans les articulations et segments du robot. Ces nouveaux paramètres doivent s'intégrer au modèle géométrique existant, ce que nous faisons en nous inspirant de la modélisation VJM (Virtual Joint Modeling).

Ce troisième chapitre met en avant une autre de nos principales contributions : une sélection des paramètres du modèle élasto-géométrique pour notre contexte, à partir d'une modélisation trop détaillée des élasticités et des jeux.

L'étude des performances du robot dans son espace de travail est décrite dans le chapitre 4. Des critères de performance usuels de robotique y sont sélectionnés et présentés. Nous introduisons également nos propres critères en vitesse et répétabilité, utiles dans notre contexte. Ces critères sont ensuite combinés pour obtenir une représentation complète des capacités du robot. Ils sont combinés avec des pondérations différentes dépendant de l'application ciblée : recherche de rapidité ou recherche de qualité de pose. Des représentations 2D (cartographies) de ces critères sont présentées au fur et à mesure afin de visualiser les capacités du robot.

Introduction générale

Pour finir, le chapitre 5 présente la mise en place de la cellule de numérisation et notamment l'intégration des différents composants de la cellule. Un travail sur la communication entre les composants y est présenté, ainsi que deux méthodes de calibration spécifiques pour notre capteur laser. En effet le capteur laser est usuellement associé à une MMT qui est une structure 3axes. L'associer à une structure à 6 DDL requiert une calibration adaptée au système support (robot).

Dans ce dernier chapitre, les principales contributions consistent en deux méthodes de calibration différentes dans leur praticités et dans leur qualité. La première peut être réalisée seulement avec le robot support et un système de suivi externe (par exemple système de stéréovision). La deuxième nécessite l'utilisation d'une MMT afin de gagner en qualité de calibration, mais elle est plus longue et contraignante.

Enfin dans un chapitre de conclusion, nous discutons des contributions apportées par nos travaux et de leurs intérêts ainsi que des résultats obtenus lors des différents essais. La discussion est ensuite élargie pour envisager les applications possibles de nos contributions au-delà du contexte de la thèse. Puis ce chapitre s'achève par une conclusion sur le travail réalisé durant la thèse et sur les perspectives des travaux qui vont suivre concernant la cellule de numérisation robotisée.

Chapitre 1

Méthodologie de génération de trajectoire pour la numérisation robotisée

Ce premier chapitre présente notre cellule de numérisation robotisée dans son ensemble : ses différents composants et une méthodologie de génération de trajectoire que nous avons développée. De cette méthodologie, nous tirons les objectifs auxquels nous devons répondre pour assurer la mise en place de cette cellule. Ces objectifs sont contraints par les besoins suivants : la mise en place possible en contexte industriel (praticité, rapidité, qualité), la numérisation de pièces encombrantes ou encore la numérisation de pièces complexes. Ce chapitre est principalement constitué d'un état de l'art autour de nos objectifs, ce qui permet de positionner nos travaux présentés dans les chapitres suivants.

Dans un premier temps, un contexte général sur l'utilisation des robots, amène à formuler des besoins autour de la mise en place d'une cellule de numérisation robotisée. Pour répondre à ces besoins, nous proposons alors une stratégie de génération de trajectoire, qui s'appuie sur plusieurs éléments clés. Le premier est centré sur la modélisation du robot. Nous présentons donc des modèles de robotique et méthodes associées pouvant répondre à notre contexte. Le deuxième élément concerne la caractérisation du robot qui nous amène à lister les différents indices de performance existant pour le qualifier. Nous terminons ensuite, par une caractérisation des moyens de mesures : le système de stéréovision et le capteur laser plan. L'ensemble de ces éléments est repris au travers d'objectifs auxquels nous répondons dans chacun des chapitres.

1.1 Introduction et contexte

1.1.1 Origine du besoin

Les robots qui étaient habituellement utilisés pour des tâches simples et répétitives (manutention, peinture) sont désormais sollicités pour de nouvelles opérations complexes comme l'usinage, la fabrication additive, et surtout la mesure. En effet le nombre important de sociétés proposant des solutions de mesures robotisées (Nikon, Perceptron, Créaform, Kréon, Metrologic,...) atteste de l'intérêt grandissant des industriels dans ce domaine. L'association d'un robot et d'un capteur laser pour effectuer des mesures en ligne confère une bonne accessibilité lors de la mesure, une rapidité d'exécution, mais surtout de la flexibilité dans la chaîne de production. Ces avantages très attractifs et le besoin d'une bonne cadence de production font que l'on préfère les robots aux autres solutions de mesure présentés dans le tableau 1 [Barbero et Ureta 2011, Savio et al. 2007, Sansoni et al. 2009]. Dans ce tableau 1, un robot porteur d'un capteur laser et suivi par un système de tracking externe possède le meilleur compromis de performances, tout en rendant la numérisation automatisable.

La MMT et le tomographe sont des machines lourdes qui ne sont pas directement sur la chaîne (atmosphère contrôlée). Le temps d'exécution est donc grandement rallongé par des opérations de montage et de démontage. Ces opérations manuelles rendent le processus non automatisable en plus de réduire la cadence de production. De même, les méthodes manuelles (laser tracker et bras polyarticulé) qui sont très attrayantes car très flexibles, ne permettent pas d'aboutir à une bonne cadence car elles sont

non automatisables. Il faut de plus rester prudent avec ces méthodes manuelles, car la qualité de la numérisation dépend alors énormément de l'opérateur [González-Madruga et al. 2014].

		Laser tracker (manuel)	Robot + Laser tracker	MMT	Stéréo-corrélation	Tomographe	Bras polyarticulé (manuel)
Temps d'exécution (numérisation + manutention)		••	••		••		••
Echelle des pièces à numériser	grande	••	••				•
	moyenne	••	••	••	•	•	••
	petite	•	•	••	••	••	•
Qualité de la numérisation		•	•	••	••	•	•
Accessibilité, orientations		••	••	•	•	••	••
Flexibilité sur la chaîne		••	••	•	•	•	••
Automatisation			••		••		

Tableau 1: Qualification des moyens de mesure (bon, moins bon, mauvais/incapacité)

La stéréo-corrélation est une méthode de mesure très bien automatisable [Dubreuil 2017] et qui donne une bonne qualité de numérisation, mais nos contraintes de départ imposent de choisir un moyen flexible (flexibilité de la chaîne de production). De plus la cellule de mesure qui résultera de nos travaux doit répondre à deux autres contraintes (pour notre laboratoire) : permettre la numérisation de pièces encombrantes (type panneau d'avion) et de pièces complexes (réalisées en impression 3D). Le manque de flexibilité de la stéréo-corrélation ne permet pas tout cela à la fois, et l'ensemble {robot + laser tracker} reste le choix le plus intéressant.

Un robot n'est cependant pas originalement conçu pour des opérations complexes telles que la mesure. Il faut donc améliorer leurs performances, comme la précision de pose définie dans la norme ISO 9283 [Sansoni et al. 2009, ISO.9283 2015]. De plus, que ce soit en laboratoire ou dans l'industrie, et il n'est que rarement possible de renouveler le parc des robots existant pour des robots neufs ayant de meilleures performances. Il faut alors chercher à améliorer l'utilisation de robots plus anciens, déjà à disposition.

Cette amélioration passe par la maîtrise du comportement du robot pour une utilisation optimale de ses capacités. Classiquement une modélisation du robot est faite pour pouvoir compenser ses défauts [Abele et al. 2008, Olabi et al. 2012]. On peut aussi positionner la tâche à effectuer, et donc les trajectoires à suivre, dans des zones où le robot est le plus performant en vitesse, précision absolue ou répétabilité suivant le but recherché [Lin et al. 2017]. Enfin, comme dans la solution retenue du tableau 1, l'opération de mesure faite par le robot est souvent suivie par un système de tracking externe pour correction [Phan et al. 2018, Posada et al. 2016]. En fait, il est rare de pouvoir se passer d'un suivi externe pour obtenir une numérisation de qualité. Des solutions sont tout de même en développement pour se passer de ce système de suivi qui ajoute une contrainte de visibilité sur l'effecteur, comme la cellule CMR de Metrologic (Coordinate Measuring Robot).

Malgré cet engouement pour la numérisation robotisée, peu de travaux se penchent sur la trajectoire de numérisation. Pour les industriels comme dans la recherche [Brosed et al. 2010, Santolaria et al. 2009], le robot n'est utilisé que pour effectuer des numérisations à orientation fixe alors que ce dernier offre plus de possibilités.

Dans ce contexte nous présentons donc une cellule de numérisation robotisée, fonctionnant avec ou sans système de suivi externe. Utiliser le système de suivi permet d'avoir une meilleure qualité de numérisation tandis que s'en priver rend la numérisation plus pratique. Une partie de notre travail consiste donc à maîtriser le comportement du robot au travers de modèles, afin d'atteindre la qualité de

pose requise par une numérisation sans suivi externe. Dans ce but également nous cherchons à générer une trajectoire de numérisation exploitant au mieux les capacités spatiales du robot utilisé. Notre trajectoire exploite de plus les 6 DDL du robot pour une réorientation continue adaptée au profil de la pièce à numériser au lieu d'utiliser des orientations fixes. Les améliorations que nous proposons dans ces travaux sont donc centrées autour d'un générateur de trajectoire permettant de maîtriser la qualité de la numérisation et d'optimiser l'utilisation des éléments de la cellule (section 1.2). Par ailleurs, lors de la mise en place de cette cellule et pour répondre à des besoins industriels, nous nous efforçons de proposer des méthodes rapides, pratiques et garantissant une qualité pour l'application de numérisation.

L'expression du générateur de trajectoire que nous proposons en section 1.2, permet en partie de lister les objectifs menant à la mise en place finale de la cellule de mesure. Un schéma global des objectifs à atteindre dans ce projet de cellule robotisée est donné en fin de chapitre 1 dans la figure 17, les objectifs atteints dans cette thèse y sont détaillés.

1.1.2 Matériel et cellule robotisée

Les éléments matériels choisis pour cette cellule, représentés sur la figure 2, sont : un capteur laser Zephyr (Kreon) pour la numérisation, un robot 6 axes Kr150L110 (Kuka) comme support et un système de stéréovision C-Track (Creaform) comme tracker. Il y a également à disposition un marbre pour placer les pièces à numériser de petite ou moyenne taille, les pièces de grande taille disposant de leur propre support. La cellule est entourée d'une barrière physique l'isolant du reste de la plateforme expérimentale. Une barrière numérique a été ajoutée, exploitant les butées logicielles programmables de l'armoire de commande du robot, afin d'éviter toute collision avec des éléments atteignables dans la cellule (sol, mur).

À cela il faut ajouter un ordinateur portable sur lequel sont installés les différents logiciels liés au matériel utilisé. Ce même ordinateur est utilisé pour les nombreux calculs réalisés dans cette thèse, principalement sous matlab. L'ordinateur sous un système d'exploitation Windows7 64 bits, dispose de 8 Go de RAM et d'un processeur i7-4710 à 4 cœurs (2,5 GHz).

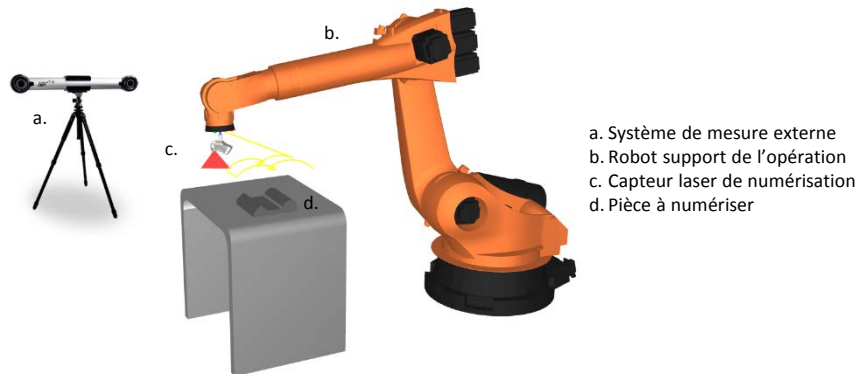


Figure 2 : Eléments de la cellule de numérisation robotisée

Le robot porteur dans cette cellule est un Kuka Kr150L110 à 6 axes d'une envergure allant jusqu'à plus de 3 mètres pour un poids de 1300kg. Ce dernier peut porter une charge maximale de 110kg et possède une répétabilité globale de pose à vide de 0,12mm d'après le constructeur. La manipulation du robot se fait au travers d'une interface appelée KCP (Kuka Control Panel), manuellement ou avec un programme préenregistré. Le programme comprend essentiellement des trajectoires et permet d'atteindre des poses enregistrées dans un repère cartésien (repère du robot ou autre), ou avec les coordonnées articulaires du robot. Il est possible d'éditer ou de créer ces programmes en dehors de l'armoire de commande. Il est donc possible de générer des trajectoires avec un code personnalisé ou avec un logiciel tiers (comme roboDK). Les pièces interfaces de la figure 3 ont été créées pour permettre au robot d'être porteur du capteur laser ou de la pièce à numériser si celle-ci est de petite ou moyenne taille.



Figure 3 : Interfaces robot/Zephyr (à gauche) et robot/Pièce (à droite)

Le capteur laser est un Zephyr de la marque Kreon. Il utilise une technologie classique basée sur un plan laser et une matrice CCD, l'intersection du plan laser et de la surface à numériser donnant une ligne laser capturée par la matrice. À ce stade on a donc une ligne laser 2D, et la connaissance de ses paramètres internes permettent au capteur de calculer cette ligne en 3D dans son propre repère. Afin de pouvoir positionner plusieurs lignes entre elles lorsque le capteur se déplace, il faut renseigner la pose du capteur au contrôleur du Zephyr. Cette pose est fournie par un système de mesure externe (tracking, MMT, coordonnées robot), donc dans un repère externe, et nécessite une calibration. Cette calibration donne la pose du repère interne du capteur dans le repère du système de mesure externe. Enfin la numérisation est obtenue sur le logiciel Polygonia sur un PC, avec lequel il est possible de retravailler le nuage de point et de l'enregistrer. Pour obtenir une bonne numérisation, il faut régler la puissance du laser en fonction de la réflexion sur la surface de la pièce à mesurer. Il existe aussi une distance et une inclinaison optimale entre le capteur et la pièce à rechercher afin d'améliorer le résultat de numérisation.



Figure 4 : Capteur laser Zephyr doté de cibles de suivi

Le système de tracking externe est un C-Track de la marque Creaform. C'est un système de stéréovision permettant de suivre la position 3D de cibles réfléchissantes à l'aide d'une projection d'infrarouges et de 2 caméras. Le volume visible par le C-Track est de $4m^3$ et il est possible d'associer plusieurs cibles pour créer un modèle de cibles, dont le suivi est plus robuste et donne accès à l'orientation d'un objet en plus de sa position. Des cibles sont donc disposées sur l'effecteur du robot pour le tracking de ce dernier. Les informations sur la pose sont transmises au contrôleur du C-Track puis au logiciel VXelements sur PC, qui permet la visualisation et l'enregistrement des mouvements des cibles sous forme d'un tableau (au format csv).



Figure 5 : C-Track et cible réfléchissante

Le principe de base de la cellule de numérisation est le suivant : le robot porteur permet au capteur de suivre les formes de la pièce à numériser, pendant que le système de mesure externe traque les mouvements de l'effecteur du robot. Les mouvements ainsi capturés sont donnés au fur et à mesure au capteur laser pour qu'il puisse reconstruire un nuage de points à partir des lignes laser mesurées. Deux configurations sont alors possibles : soit le capteur laser est monté sur le robot et la pièce est fixe dans la cellule [Santolaria et al. 2009], soit le capteur est fixe et la pièce est rattachée au robot [Brosed et al. 2010]. Pour la suite de ces travaux, nous nous mettons dans la première configuration plus intuitive où le robot porte le capteur laser, l'autre configuration sera abordée plus tard en perspective.

Le C-Track suit l'effecteur et donc le Zephyr, afin d'acquérir et de communiquer la pose de ce dernier. Mais le C-Track peut perdre la trace des cibles suivies si ces dernières sortent de son champ de vision, ou ne sont plus visibles (inclinaison de la cible, obstacle). Afin de ne pas interrompre la numérisation, la pose du Zephyr est alors donnée par une modélisation du robot et par la connaissance des coordonnées articulaires actuelles du robot. Cette méthode basée sur un modèle et des coordonnées est moins précise que le suivi par le C-Track, mais n'a pas de problème de visibilité et est utilisable pour toute configuration du robot. Lors d'une numérisation, on tente donc de conserver au maximum la visibilité des cibles suivies par le C-Track, et le modèle du robot est exploité lors de la perte du tracking. Lorsque les cibles sont de nouveau visibles, la pose du Zephyr est à nouveau acquise par le C-Track. À terme, avec une modélisation fidèle du comportement du robot, il sera possible de suivre le Zephyr uniquement avec le modèle et sans système de tracking externe, pour des raisons de praticité.

Un dernier élément important est la dalle sur laquelle reposent le robot, le C-Track et le support de la pièce à numériser. Elle est une source d'erreur importante car elle assure la position relative des éléments de la cellule. Une dalle peut être spécialement prévue pour une bonne stabilité et de faibles interactions avec l'extérieur ; elle peut être séparée des autres dalles de béton et reposer sur des piliers profondément ancrés. Ce type de dalle, utilisée dans notre cellule, permet l'hypothèse que les éléments restent fixes les uns par rapport aux autres, et qu'aucune perturbation extérieure ne viendra entacher les mesures.

Parmi les éléments de la cellule présentés, le robot, le C-Track et le capteur laser influent sur la qualité de la numérisation. La maîtrise de la qualité de la numérisation passe donc par la qualification de ces différents éléments. La calibration des capteurs lasers est bien connue [Santolaria et al. 2009], mais il subsiste un manque de standardisation pour l'étude des performances d'un capteur laser. Nous exploitons donc un protocole de qualification développé autour de notre capteur [Mehdi-Souzani et al. 2016]. Ce manque de standardisation existe également pour les systèmes de tracking. En effet ces systèmes servent avant tout à faire des prises de points avec un artéfact doté de cibles et d'un palpeur. La qualification du constructeur est donc faite sur l'ensemble C-Track et artéfact suivant la norme [ASME.B89.4.22-2004 2014] pour maîtriser la pose du palpeur. Or nous cherchons à assurer la qualité de suivi d'une cible par le C-Track. C'est pourquoi nous mettons en place notre propre qualification. Enfin, les modèles utilisés pour les robots industriels sont souvent très simplifiés et ne prennent en compte que 1 ou 2 paramètres en plus qu'un modèle DH [Denavit 1955, Olabi 2011]. Il est nécessaire d'avoir une modélisation plus précise pour pouvoir remplacer le suivi des cibles par le C-Track, lors de la perte de ces cibles. Nous choisissons donc un modèle de paramètres géométriques et élastiques, et nous proposons dans le même temps, une méthode rapide et pratique pour l'identification des paramètres. D'autre part, les performances du robot sont hétérogènes dans son espace de travail [Moreno et al. 2012], et il est crucial de sélectionner des zones dans cet espace pour assurer la vitesse et la qualité du positionnement du robot. Pour cela, il est possible de contraindre la trajectoire du robot à partir de critères [Vosniakos et Chronopoulos 2009], notamment des indices de performance du robot. À partir de cartographies de performance du robot (basées sur des indices de performance) et des performances du capteur laser, une stratégie de génération de trajectoire est introduite dans la section suivante. Une recherche sur l'exploitation des performances des éléments de la cellule via la génération de trajectoire permettra de répondre au besoin de qualité sur la numérisation, puis d'optimiser la vitesse de numérisation pour répondre au besoin sur la cadence de production.

1.2 Stratégie de génération de trajectoire pour la numérisation robotisée

Le capteur laser Zephyr et le robot ont des performances qui varient en fonction de leur positionnement. Travailler sur le positionnement de chacun permet alors de jouer sur leurs performances et de tirer le meilleur des deux systèmes. La trajectoire de numérisation dicte le positionnement du Zephyr par rapport à la pièce et la configuration du robot. Ainsi, étudier la génération de trajectoire de numérisation permet d'exploiter les performances du Zephyr et du robot. Nous présentons dans cette section, une stratégie de génération de trajectoire nous donnant une trajectoire optimale au regard des performances du robot et du Zephyr.

Pour le Zephyr, la problématique du positionnement vient de la technologie du capteur. L'utilisation d'un plan laser et d'une matrice CCD impose une fenêtre de mesure qui donne une distance minimale et une distance maximale entre le capteur et la surface numérisée. En dehors de cette fenêtre le Zephyr est incapable de faire une acquisition. De plus la ligne laser doit être bien visible par la matrice, au-delà d'une certaine inclinaison la réflexion du laser sur la surface ne suffit plus pour l'acquisition. Il y a donc également une inclinaison maximale à respecter (qui peut dépendre du matériau de la pièce). Pour numériser entièrement une pièce de forme et de surface complexes, il faut donc plusieurs orientations du capteur au cours de la numérisation afin de respecter les contraintes en distance et en inclinaison avec la surface. Il est nécessaire d'avoir la capacité de réorienter le capteur au cours de la numérisation, ce que permet le robot porteur. On peut donc changer régulièrement l'orientation pour toujours respecter les contraintes. Cependant le robot permet plus que cela, il autorise une réorientation continue au cours de la trajectoire, ce qui permet d'ajouter un autre objectif : optimiser la distance et l'inclinaison du capteur en continu. En effet, outre les contraintes de bases décrites précédemment, le bruit résultant sur la numérisation dépend de la distance et de l'inclinaison du capteur [Mehdi-Souzani et al. 2016]. La liberté sur le positionnement du capteur permet une optimisation de la distance et de l'inclinaison par rapport à la surface numérisée pour réduire le bruit. Il faut donc respecter les contraintes de base (inclinaison maximale, distance), puis optimiser le positionnement du capteur par rapport à la surface complexe, ce qui aboutit à une trajectoire optimale entre le capteur et la pièce dépendante des formes de la pièce et des performances du Zephyr.

Pour le robot, la problématique de positionnement vient de l'hétérogénéité des performances du robot dans son espace de travail, ces performances peuvent être observées au travers d'indices de performance [Moreno et al. 2012]. La génération de trajectoire est une étape inhérente aux robots et permet de tirer avantage de leurs 6 DDL (degrés de liberté) pour améliorer la pose de l'effecteur [Kim et Sarma 2002, Larsson et Kjellander 2006, Mineo et al. 2016, Zha 2002]. Mais l'utilisation d'indices de performance est plutôt rare [Lin et al. 2017], la plupart des indices existant en robotique ne sont pas pris en compte pour la génération de trajectoire [Zha 2002]. Ici nous voulons prendre en compte un ensemble d'indices de performance utiles pour notre application, relatifs en particulier à la qualité de pose de l'effecteur et à la vitesse du robot. La qualité de pose d'un robot détermine avec quelle précision il peut suivre une trajectoire commandée. Dans notre cas, la trajectoire commandée est celle obtenue précédemment entre le Zephyr et la pièce. Tout écart du robot relativement à cette trajectoire nuit à l'optimisation faite sur celle-ci, augmentant le bruit de numérisation. Nous étudions donc les problèmes de positionnement du robot, car celui-ci possède des performances hétérogènes au sein de son espace de travail et donc une qualité de pose différente en fonction de sa configuration. Il faut alors rechercher où placer la pièce dans l'espace de travail du robot pour que ce dernier réalise la trajectoire en exploitant au maximum ses performances pour une bonne qualité de pose et un bruit acceptable. Bien sûr plusieurs zones dans l'espace de travail du robot peuvent répondre à ce besoin, et il faut un critère supplémentaire pour les départager. Là aussi les indices de performance permettent de faire une sélection, cette fois selon un autre critère comme la vitesse du robot, et donc de numérisation (cadence de production).

Il y a donc deux points sur lesquels notre générateur de trajectoire va s'appuyer : la trajectoire entre le capteur laser et la pièce, faite sans considérer le robot, mais en utilisant les performances du Zephyr, puis le positionnement de cette trajectoire dans l'espace de travail du robot pour exploiter au mieux la qualité de pose et la vitesse d'exécution de ce dernier (avec des indices de performance). Pour la génération de trajectoire, il y a donc des contraintes à respecter, en termes de qualité de numérisation (bruit) et de minimum de visibilité pour le capteur, puis des libertés permettant une optimisation selon un critère laissé au choix de l'opérateur (ici plutôt la vitesse pour la cadence). Nous proposons alors, sur la figure 6, une stratégie de génération de trajectoires respectant des contraintes données en termes de qualité de numérisation (ou autre), puis permettant une optimisation selon un ou des critères supplémentaires ajoutés par l'opérateur (vitesse, répétabilité).

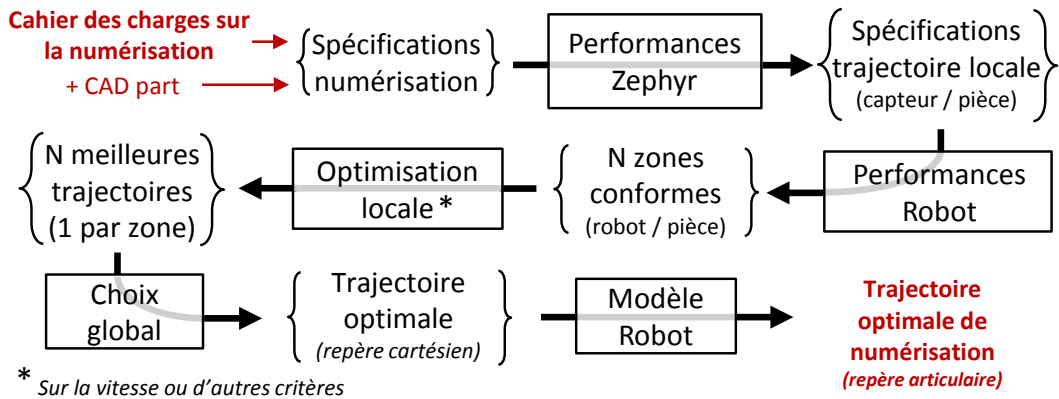


Figure 6 : Stratégie de génération de trajectoires

Notre algorithme de génération de trajectoire amène à sélectionner la meilleure trajectoire à implémenter sur le robot, pour satisfaire une qualité requise pour la numérisation et avoir une vitesse optimale (ou autre critère si besoin). La trajectoire finale est composée de points de passage décrivant la configuration du robot et donc la pose de l'effecteur. Il est commun de donner une trajectoire dans un repère cartésien à l'armoire de commande du robot. Celle-ci fait alors la conversion des points de passages de la trajectoire, avec un modèle du robot présent dans l'armoire, ce qui donne des coordonnées articulaires pour commander les axes du robot. Mais pour utiliser notre propre modèle plus fiable, nous fournissons à l'armoire une trajectoire directement en coordonnées articulaires.

Comme décrit sur la figure 6, notre stratégie de génération de trajectoire commence avec l'étude d'un cahier des charges sur la numérisation à effectuer, en termes de qualité, de vitesse et autre. Une CAO de la pièce à numériser est aussi requise pour connaître la géométrie de la pièce à numériser. Nous supposons cette CAO disponible car nous travaillons soit dans le cadre d'une chaîne de production -la CAO est alors connue- soit au laboratoire pour mesurer des pièces réalisées en fabrication additive ou des panneaux d'avion dont les CAO sont connues en interne. Avec la connaissance des performances du capteur (distance, inclinaison) et les spécifications de numérisation (CAO + cahier des charges), on obtient un ensemble de spécifications sur la trajectoire locale capteur/ pièce, respectant les contraintes du cahier des charges et les conditions minimales d'utilisation du capteur (pas d'optimisation pour l'instant). Ensuite les indices de performance du robot permettent de sélectionner des placements de la pièce à numériser dans l'espace de travail du robot, où les spécifications locales sont respectées. Généralement plusieurs zones (N zones) répondent à ce besoin. Un jeu de zones admissibles par rapport au cahier des charges et aux conditions d'utilisation du Zephyr est alors identifié.

Viennent ensuite des étapes d'optimisation. En effet il reste des libertés sur la définition de la trajectoire, et un robot peut théoriquement atteindre une même pose (position et orientation) avec 8 configurations différentes. Tout d'abord une trajectoire optimale est générée pour chaque zone en cherchant à minimiser le bruit pour le capteur (distance, inclinaison) et en ajoutant un autre critère d'optimisation au problème (en vitesse ou autre selon ce que souhaite l'opérateur). Il y a alors autant de trajectoires

optimisées que de zones, et finalement, la meilleure trajectoire est sélectionnée parmi les N trajectoires, toujours suivant les mêmes critères (bruit avec distance et inclinaison, vitesse). Ainsi la trajectoire finale respecte les contraintes du cahier des charges et les conditions minimales d'utilisation du Zephyr, et elle est optimisée selon des critères choisis (bruit, vitesse).

Ainsi pour cette stratégie il faut bien connaître les performances du capteur laser, ce qui est le cas pour le capteur laser Zephyr [Mehdi-Souzani et al. 2016]. Il est également nécessaire d'avoir une modélisation du robot pour la génération finale de trajectoire dans un repère articulaire, et de bien connaître les performances du robot au travers d'indices de performance. Les indices de performance sont pour la plupart calculés à partir d'un modèle du robot, il est donc d'autant plus nécessaire d'avoir une bonne modélisation du robot pour cette stratégie.

Dans les sections suivantes, nous nous focaliserons donc sur une modélisation du robot afin de répondre à ce besoin. Puis nous présenterons les indices de performance classiques en robotique pour le générateur. Enfin nous décrirons une étude des performances du capteur laser Zephyr et du système de mesure externe C-Track.

Cette stratégie nécessite beaucoup d'étapes préliminaires avant de pouvoir être mise en place. La description de celle-ci nous a permis de dégager des besoins auxquels il faut répondre pour pouvoir créer un générateur de trajectoires. Le schéma global de la figure 17 (fin de chapitre 1) reprend ces besoins afin de situer les travaux qui ont été effectués dans cette thèse.

1.3 Approches pour la modélisation et l'identification de défauts d'un robot sériel

1.3.1 Généralités sur la modélisation des robots

L'intérêt d'une modélisation pour un robot est d'identifier des erreurs et défauts pour corriger la CAO théorique du robot. Mais bien souvent l'emploi d'un « modèle » pour le robot prend en compte bien plus qu'une simple modélisation du comportement physique du robot. Les sources d'erreur classiques lors de l'utilisation d'un robot ont été répertoriées par Damak [Damak 1996], qui met en avant la différence entre réalité et théorie. Il cite notamment les erreurs dues aux algorithmes des logiciels CAO qui diffèrent des algorithmes des baies de commande pour la génération et l'interpolation de trajectoire. On peut ajouter que l'asservissement des axes du robot, qui génère des erreurs, n'est pas pris en compte dans les simulateurs CAO. D'autre part Damak fait référence aux erreurs de modélisation : les pièces ne correspondant pas aux valeurs théoriques de leur CAO ce qui induit des différences entre modèle et réel que ce soit pour le robot, la pièce, le capteur laser ou les différents supports et interfaces. Enfin il ne faut pas oublier le positionnement des éléments les uns par rapport aux autres. Il est en effet nécessaire de connaître la position relative entre le robot et la pièce opérée. Ceci passe par la connaissance de la position des repères robot, pièce, capteur laser, C-track, et des autres pièces d'interface et de support, par rapport à un repère global attaché à la cellule. D'après le tableau 2 ci-après proposé par Damak, on peut résumer les sources d'erreurs les plus importantes liées à la modélisation du robot, à l'asservissement et à l'interpolation.

Cette description des erreurs n'est valable que si l'on passe par un modèle du robot pour avoir la pose de l'effecteur. Lorsque l'effecteur est suivi par notre système de mesure externe (C-Track), ces erreurs sont en partie évitées. Parmi les sources d'erreur, celles d'interprétation de l'armoire de commande (interpolation et asservissement) seront moindres car nous n'utilisons pas de logiciel tiers pour générer la trajectoire, et nous commandons directement en coordonnées articulaires. De plus la trajectoire de numérisation comportera essentiellement des points d'arrêt et pas ou peu de points de passage.

Sources d'erreur		Erreurs sur les trajectoires		
		Point à point		Continue
		Points de passage	Points d'arrêt	
C.A.O.	Interpolation	10 à 500 mm	0	0 à 10 mm
	Asservissement	1 à 10 mm	0	1 à 5mm
Cellule	Position Pièce/Robot	<5 mm	<5 mm	<5 mm
	Outil (Zephyr)	0.1 à 0.5 mm	0.1 à 0.5 mm	0.1 à 0.5 mm
	Pièce	<2 mm	<2 mm	<2 mm
	Robot	0 à 15 mm	0 à 15 mm	0 à 15 mm

Tableau 2 : Influence des sources d'erreur d'une opération robotisée [Damak 1996]

Les positions de la pièce à numériser et du robot dans la cellule sont prises avec le C-Track. Nous utilisons donc le repère du C-Track comme repère global de la cellule. Le positionnement du robot dans ce repère est fait en même temps que l'identification des paramètres du modèle du robot, en ajoutant 6 paramètres de positionnement. Le positionnement de la pièce n'est pas investigué ici. Nous ne cherchons pas la position du C-Track puisque son repère propre sert de repère global. Nous ne cherchons pas non plus la position du capteur laser (Zephyr) dans le repère global, mais plutôt sa position dans le repère de l'effecteur du robot.

Les erreurs dues aux défauts de l'outil, ici le Zephyr, sont minimales et pris en compte par la qualification du capteur laser (section 1.5.1). Il est ensuite normal qu'il y ait des défauts sur la pièce puisque, dans notre application, nous cherchons à la mesurer. La principale source d'erreur restante est alors due à la structure physique du robot, dont le comportement diffère de son modèle théorique. Les défauts dans la physique d'un robot sont de deux types : géométriques et non géométriques. Ils sont la source des erreurs de mise en position du robot, et leurs effets peuvent être quantifiés par l'exactitude et la répétabilité de pose [Conrad et al. 2000], en respectant la norme [ISO.9283 2015]. Les défauts géométriques, regroupés dans un modèle dit géométrique, sont liés aux différences entre les dimensions théoriques de la CAO du robot et les dimensions réelles du robot. Les défauts non géométriques, regroupés souvent dans un modèle plus complet dit élasto-géométrique, regroupent quant à eux toutes les autres erreurs possibles qui ne sont pas d'ordre géométrique, mais dues à des variations des conditions (charge, température).

En résumé, pour compenser les différentes sources d'erreur de notre opération robotisée, nous prenons en compte : une modélisation de la structure physique du robot (modèle géométrique et élasto-géométrique), le positionnement du robot dans le repère global du C-Track, le positionnement de la pièce dans le repère global (non traité dans cette thèse), le positionnement du Zephyr sur l'effecteur du robot, la qualification du Zephyr. Étant donné le rôle central du C-Track il est également important d'avoir une qualification de ce dernier afin de connaître la qualité des positionnements et calibrations effectués. Ces qualifications du Zephyr et du C-Track sont traitées en section 1.5. Le positionnement du Zephyr est traité dans le chapitre 5. Le positionnement du robot dans le repère global (repère C-Track) est intégré dans la modélisation du robot, cette modélisation est abordée dans cette section.

La prise en compte de ces sources d'erreurs va servir différents points de notre opération robotisée : la génération de la trajectoire commande pour le robot, le calcul des indices de performance du robot, le suivi de l'effecteur au cours de l'opération quand le C-Track ne s'en charge pas (pour recalage des lignes laser). Le modèle du robot intervient donc sur différents points importants de notre numérisation robotisée. L'importance du modèle dans notre cas implique donc d'être rigoureux dans la sélection et la définition de ce modèle [Elatta et al. 2004, Hollerbach 1989]. Cette modélisation est à deux échelles de détail : géométrique ou élasto-géométrique. Le besoin en détails du modèle n'est pas le même, que ce soit pour effectuer le suivi de l'effecteur ou pour calculer des indices de performance, qui sont moins sensibles à des erreurs de modélisation. Une modélisation géométrique est donc probablement suffisante pour les indices tandis qu'il faut approfondir le modèle avec des paramètres élastiques pour le suivi ou la génération finale de la trajectoire.

Établir une modélisation du robot signifie qu'il faut identifier les paramètres du modèle choisi. Quel que soit le niveau de détails du modèle, les étapes d'une identification sont généralement les suivantes [Elatta et al. 2004, Roth et al. 1987] :

- Détermination du modèle mathématique représentant le robot.
- Campagne de mesure suivant le robot dans différents mouvements, configurations, contraintes, efforts.
- Identification des paramètres du modèle mathématique à partir des mesures.
- Intégration du modèle identifié dans la boucle de commande du robot, soit au moment de la génération de trajectoire, soit au moment de l'interprétation de la trajectoire.

Dans cette section, nous nous attarderons sur les choix de modèles de robot et sur les méthodes d'identification des paramètres.

1.3.2 Les modèles géométriques

Une modélisation géométrique permet de prendre en compte les défauts géométriques de pièces ou d'assemblages. Ces défauts sont dus au montage des pièces et aux tolérances d'usinage qui modifient la géométrie théorique du robot. Des erreurs de positionnement apparaissent alors dans les liaisons et les segments du robot. Dans les figures 7 et 8 représentant des erreurs dans un segment et une liaison, le repère R_{i2} devient le repère R_{i3} ce qui modifie la position et l'orientation de toutes les pièces en aval. Pour une structure de 3 segments (de 400mm), les tolérances, les défauts d'alignement, de perpendicularité peuvent générer plus de 3mm d'erreur sur la pose de l'effecteur [Weill et Shani 1991]. Une grande partie de cette erreur est due aux défauts d'orientation qui sont amplifiés, comme notamment la justesse autour de l'axe de la liaison qui peut être responsable à 50% de l'erreur de pose [Caenen 1993] voire même 97% [Conrad et al. 2000].

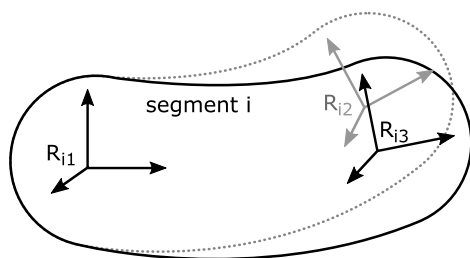


Figure 7 : Défauts d'un segment i

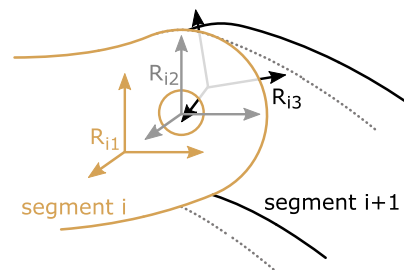


Figure 8 : Défauts d'une liaison entre deux segments

Ces défauts géométriques peuvent être pris en compte de différentes façons. Il existe de très nombreuses modélisations géométriques pour les robots sériels [Xuan et al. 2014], permettant d'intégrer et compenser ces défauts. Tout d'abord il existe des modèles d'erreur et des cartographies empiriques de l'erreur de pose de l'effecteur [Zeng et al. 2016, Damak 1996, He et al. 2010], qui permettent de connaître l'erreur de pose du robot en tout point de son espace de travail. Cette erreur est ensuite ajoutée lors de la génération ou de l'interprétation d'une trajectoire théorique. Mais l'utilisation d'un modèle d'erreur suppose bien souvent une linéarisation au premier ordre des erreurs, ce qui empêche d'ajouter la modélisation de défauts non géométriques [Ziegert et Datsaris 1988]. Il est donc préférable de ne pas faire un modèle donnant l'erreur de pose, mais plutôt un modèle donnant directement la pose du robot, comme le propose la convention de Denavit et Hartenberg [Denavit 1955].

Sur les figures 7 et 8, les déplacements entre les repères R_{i2} et R_{i3} correspondent à 3 translations et 3 rotations [Coorevits et al. 1991]. Pour un ensemble {articulation + segment} il y a donc 6 translations et 6 rotations que l'on peut sommer pour revenir à 3 translations et 3 rotations. En suivant cette logique, il suffit donc d'un torseur de 6 déplacements, et donc 6 paramètres, pour prendre en compte les défauts géométriques dans le passage entre deux axes consécutifs. La convention DH exploite la perpendiculaire

commune entre 2 axes consécutifs pour n'utiliser que 4 paramètres de passage. Un exemple de paramétrage pour passer d'un axe i à un axe $i+1$ est donné en figure 9. La modélisation DH ne cherche pas à positionner correctement les repères des liaisons suivant la réalité physique du robot, qui sont finalement inutiles pour déterminer la pose de l'effecteur. Étant donné que placer un axe par rapport à un autre requiert moins d'informations que de placer un repère par rapport à un autre, 4 paramètres suffisent. Avec ces 4 paramètres du modèle DH, on peut donc prendre en compte seulement la partie utile des 6 déplacements issus de défauts géométriques sur l'ensemble {articulation + segment}.

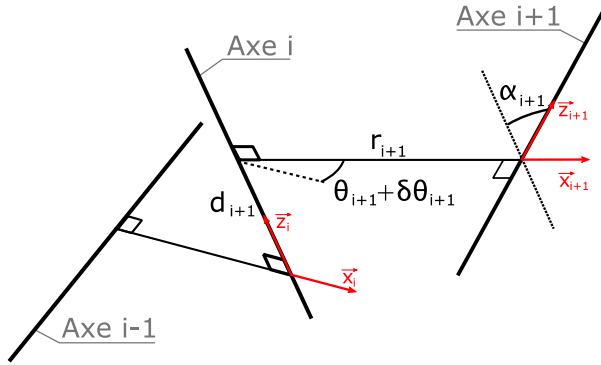


Figure 9 : Paramétrage DH

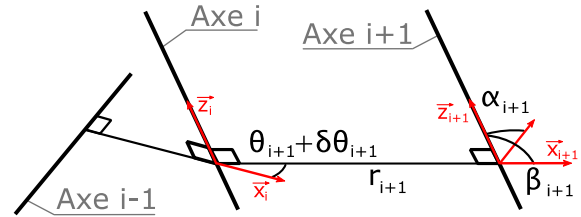


Figure 10 : DH avec paramètre de Hayati

En utilisant la paramétrisation DH à 4 paramètres, il est alors possible de mettre en place une modélisation pour le robot entier, allant de la base du robot à son outil. Comme il n'est pas utile de positionner la base physique du robot dans le repère global de la cellule, nous appelons base ce repère global. Dans ses travaux, Damak montre que les paramètres de la base à l'outil ne sont pas tous des caractéristiques propres au robot [Damak 1996]. En effet les paramètres propres aux robots sont ceux entre la première et la dernière liaison du robot, comme le montre le tableau 3. Ainsi entre la base et l'axe 1, ainsi qu'entre l'axe 1 et l'axe 2, on retrouve les paramètres de positionnement du robot dans le repère global de la cellule, puis on a les caractéristiques de l'outil après l'axe 6. Cette décomposition reprend les différentes sources d'erreur citées dans la section précédente à savoir : un modèle propre à la structure physique du robot Kuka, un modèle de positionnement du robot par rapport au C-Track (6 paramètres de pose dans repère global), et le positionnement de l'outil sur l'effecteur (Zephyr).

passages (robot n axes)	Paramètres de positionnement	Paramètres propres au robot	Paramètres outil
base à 1	4	0	0
1 à 2	2	2	0
2 à 3	0	4	0
...	0	4	0
n-1 à n	0	4	0
n à outil	0	0	6

Tableau 3 : Nombre de paramètres DH de passage pour un robot à i axes

Concernant la méthode de modélisation elle-même, le passage d'un axe i à un axe $i+1$ à l'aide des paramètres DH peut s'écrire sous forme d'une matrice homogène de passage. On retrouve cette expression dans l'équation 1, qui est composée de deux rotations et deux translations comme pour le passage sur la figure 9. Elle est souvent enrichie avec de nombreux paramètres en fonction du besoin. En effet la convention originale DH possède de nombreux défauts, notamment des singularités posant problème lors de l'identification des paramètres, une mauvaise représentation des défauts d'alignement des axes, une notation lourde et peu représentative pour les longues chaînes [Santolaria et al. 2008, Elatta et al. 2004, Everett et al. 1987].

$$Td_{i,i+1} = \text{Trans}(\mathbf{Z}_i, d_{i+1})\text{Rot}(\mathbf{Z}_i, \theta_{i+1} + \delta\theta_{i+1})\text{Trans}(\mathbf{X}_{i+1}, r_{i+1})\text{Rot}(\mathbf{X}_{i+1}, \alpha_{i+1}) \quad \text{Eq. 1}$$

C'est pourquoi de nombreux auteurs ont cherché de nouvelles modélisations répondant à leurs besoins. Beaucoup de ces nouveaux modèles ajoutent un paramètre afin de prendre en compte les défauts d'alignement, comme le Zero reference model basé sur la formule de rotation de Rodrigues en version matricielle [Mooring 1984]. Mais ce paramètre supplémentaire apporte de la redondance (problème d'indépendance des paramètres) dans ces modèles ce qui complique la phase d'identification des paramètres. Hollerbach propose alors d'utiliser le paramètre de Hayati [Hayati 1983] (correspondant à ce paramètre supplémentaire), en remplacement d'un autre paramètre et uniquement dans le cas où deux axes consécutifs sont théoriquement parallèles [Hollerbach et Wampler 1996], comme dans la figure 10. Une translation est alors remplacée par une rotation et on obtient l'équation 2. Le nombre de paramètres entre deux liaisons reste donc le même que celui de la convention DH (4 paramètres) et les défauts d'alignement majeurs sont pris en compte.

$$T_{h_{i,i+1}} = \text{Rot}(Z_i, \theta_{i+1} + \delta\theta_{i+1}) \text{Trans}(X_{i+1}, r_{i+1}) \text{Rot}(X_{i+1}, \alpha_{i+1}) \text{Rot}(Y_{i+1}, \beta_{i+1}) \quad \text{Eq. 2}$$

Des formalismes comptant plus de paramètres ont également été développés pour éviter les singularités dans le modèle ou dans l'identification ainsi que pour mieux approcher la réalité physique du robot [Goswami et al. 1993]. Ainsi le S-model [Stone et al. 1986] permet de bien positionner le centre de liaison afin que ce dernier ne soit pas en dehors de la structure. Le modèle CPC [Zhuang et Roth 1992] lui cherche à aboutir à un modèle complet et continu. Les principaux modèles, le nombre de paramètres requis entre deux liaisons et les particularités prises en compte sont présentés dans le tableau 4.

Modèle	paramètres P	Particularités
Modèle DH [Denavit 1955]	4P	Modèle de base simple
Zero Reference Model [Mooring 1984]	DH + 1P	Alignement des axes
Paramètre de Hayati [Hollerbach et Wampler 1996]	DH + 0P	Alignement des axes
S – Model [Stone et al. 1986]	DH + 2P	Centre des liaisons
Modèle CPC [Zhuang et Roth 1992]	DH + 2P	Continuité et complétude
Modèle MCPC [Zhuang et al. 1993]	4P	Singularités
Formule POE (modèle d'erreurs) [He et al. 2010]	7P	Représentation des erreurs

Tableau 4 : Nombre de paramètres et particularités des modèles géométriques

Quel que soit le nombre de paramètres choisi pour les matrices de passage, la modélisation complète allant de la base à l'outil, et donc la pose de l'outil dans le repère de la base, se fait en multipliant toutes les matrices homogènes de passage comme dans l'équation 3.

$$T_{\text{base,outil}} = T_{\text{base},1} T_{d_{1,2}} T_{h_{2,3}} \dots T_{d_{n-1,n}} T_{n,\text{outil}} \quad \text{Eq. 3}$$

À partir des n coordonnées articulaires du robot et des paramètres géométriques choisis, on peut calculer la matrice $T_{\text{base,outil}}$ qui contient alors les informations de position et d'orientation de l'outil. En se basant sur cette modélisation il est également possible de déduire un modèle inverse [Khalil et Dombre 2004] afin cette fois-ci d'obtenir les coordonnées articulaires du robot à partir de la pose de l'outil.

On peut bien sûr formuler la pose de l'effecteur (ou de l'outil ici) autrement que par une matrice homogène, en extrayant les angles α_{outil} , β_{outil} et γ_{outil} de la matrice de rotation présente dans la matrice homogène $T_{\text{base,outil}}$ ce qui donne (toujours dans le repère de la base) :

$$P = [X_{\text{outil}}, Y_{\text{outil}}, Z_{\text{outil}}, \alpha_{\text{outil}}, \beta_{\text{outil}}, \gamma_{\text{outil}}]^t \quad \text{Eq. 4}$$

Cela permet d'introduire pour la suite, la notion de jacobienne du modèle géométrique. La jacobienne est une dérivée du vecteur P représentant la pose de l'outil. Il y a plusieurs types de jacobiennes : une jacobienne J_θ obtenue par dérivation par rapport aux n coordonnées articulaires du robot θ_i , et une jacobienne J_w obtenue par dérivation par rapport aux g paramètres w_i du modèle.

$$\dot{P} = \frac{dP}{dt} = \sum_{i=1}^n \frac{\partial P}{\partial \theta_i} \frac{d\theta_i}{dt} = J_\theta [\dot{\theta}_1, \dot{\theta}_2, \dots, \dot{\theta}_n]^t \quad \text{Eq. 5}$$

$$\dot{P} = \sum_{i=1}^g \frac{\partial P}{\partial w_i} \frac{dw_i}{dt} = J_w [\dot{w}_1, \dot{w}_2, \dots, \dot{w}_g]^t \quad \text{Eq. 6}$$

La jacobienne est donc constituée de colonnes liées à chaque coordonnée ou chaque paramètre. Le calcul de P à partir de la matrice obtenue dans l'équation 3 puis les dérivations de l'équation 6 s'avèrent souvent complexes. Des méthodes existent donc pour calculer chaque colonne des jacobiennes indépendamment, et donc plus simplement [Khalil et Besnard 2002].

La jacobienne J_θ permet d'avoir un modèle cinématique du robot, car elle permet de calculer la vitesse de l'effecteur en fonction des vitesses de chaque articulation. Et si l'on supprime le facteur temps dt , on observe que cette même jacobienne va donner l'influence d'une petite variation d'une coordonnée articulaire $d\theta_i$ sur la pose de l'effecteur dP . On peut donc connaître la propagation de l'erreur de positionnement de chaque articulation sur l'erreur de pose globale, de même que la résolution du robot pour une configuration donnée. J_θ est donc très utilisée pour le calcul d'indices de performance du robot.

De même la jacobienne J_w donne l'influence d'une petite variation des paramètres du modèle dw_i sur la pose de l'effecteur dP . On a ainsi un modèle de propagation d'erreur, des paramètres sur la pose. J_w est utile pour trouver les configurations où les paramètres sont très influents, ce qui aide à la sélection ou à l'identification de paramètres. Beaucoup de critères de sélection et d'identification se basent sur cette jacobienne.

Nous utiliserons donc ces jacobiennes dans le chapitre 4 sur les indices de performance et dans les chapitres 2 et 3 lors de la définition des paramètres des modèles utilisés. Le choix d'une modélisation pour notre robot Kuka, à partir notamment du tableau 4, est présenté dans le chapitre 2. À présent nous allons voir comment enrichir ce modèle pour prendre en compte des comportements non géométriques.

1.3.3 Défauts non géométriques

Un modèle élasto-géométrique est une extension du modèle géométrique, comprenant plus de paramètres qui vont intégrer les défauts non géométriques. Ces défauts non géométriques sont liés à l'environnement (température, chargement, gravité) et se traduisent le plus souvent par des déformations des segments et des liaisons [Elatta et al. 2004]. Les jeux dans les liaisons et les erreurs de transmission (engrenages) sont également traditionnellement considérés comme des défauts non géométriques [Elatta et al. 2004, Olabi 2011], on les retrouve donc dans la modélisation élasto-géométrique.

De nombreux travaux estiment qu'une modélisation géométrique est suffisante, car les erreurs géométriques sont très grandes devant les autres erreurs [Elatta et al. 2004, Rolland 2003]. Par exemple, les défauts non géométriques sont responsables de 5% à 10% de l'erreur de pose de l'effecteur dans des cas présentés dans [Judd et Knasinski 1990, Becquet 1987]. Cependant, dès [Whitney et al. 1985] il est montré que les déformations élastiques sont à prendre en compte si l'on souhaite atteindre une bonne précision de pose, tout comme les jeux qui ont un impact fort sur la pose du robot [Schneider et al. 2015]. De plus, si l'on omet des défauts non géométriques influents, les paramètres géométriques vont beaucoup varier d'une identification à l'autre [Caenen 1993]. Le modèle géométrique permet de corriger en partie l'erreur de positionnement de l'effecteur, et le modèle élastique, s'il est employé, est affiné afin d'obtenir une erreur résiduelle acceptable en fonction de l'application ciblée (ici numérisation).

Les défauts non géométriques sont localisés dans les articulations et les segments du robot. Comme pour les défauts géométriques, l'erreur se compose de 3 translations et 3 rotations pour chaque articulation ou segment, sous forme d'un torseur de déplacement appliqué à l'extrémité de l'entité. Cependant les erreurs dépendent ici des efforts, de la température, du mouvement du robot et ne sont donc pas des constantes. Le modèle élasto-géométrique prend donc en entrée les coordonnées articulaires du robot, mais aussi les efforts sur chaque entité (articulation ou segment), la température, le mouvement actuel du robot, et toute entrée utile à la modélisation d'un comportement non géométrique.

Parmi tous les phénomènes qui interviennent sur le robot, il a été démontré à maintes reprises que la déformation autour des axes des articulations était le défaut non géométrique le plus influent [Zhang et al. 2005, Alici et Shirinzadeh 2005, Judd et Knasinski 1990]. Si ce défaut, appelé justesse angulaire, est aussi important, c'est parce qu'il regroupe les déformations de tous les éléments transmettant le mouvement de rotation, c'est-à-dire du moteur à la liaison. Ces déformations forment la partie passive du défaut. Une autre partie dite active prend en compte les défauts d'asservissement. Cette partie active de la justesse angulaire peut varier avec le temps, mais cette variation est négligeable devant l'ordre de grandeur des défauts [Alici et Shirinzadeh 2005]. Les parties actives et passives sont difficilement séparables et on ne retient généralement qu'un seul paramètre pour la justesse angulaire. Elles permettent tout de même de lister les sources d'erreur possibles.

D'autres phénomènes que la justesse angulaire sont généralement ajoutés afin de raffiner le modèle jusqu'à l'obtention d'une erreur de pose acceptable. Par exemple, autour des axes des articulations on retrouve également la notion de jeu [Whitney et al. 1985]. Certains auteurs se basent sur la théorie du contact pour modéliser la complexité des jeux dans les articulations [Flores et Ambrósio 2004, Tian et al. 2018]. En effet les jeux dans les articulations génèrent un comportement fortement non linéaire [Schneider et al. 2015]. D'autres auteurs au contraire, font le choix de simplifier le modèle en étudiant les sollicitations des articulations pour négliger jeux et élasticités [Nubiola et Bonev 2013]. Afin de poser un cadre aux différents phénomènes intervenant dans le modèle, on peut lister les directions de sollicitation possible sur les éléments du robot (segments et articulations).

La norme ISO 230-1 permet de fixer le vocabulaire à employer dans les articulations ainsi que les différents défauts de position et d'orientation par rapport à la ligne moyenne de l'axe de liaison pouvant occurrer [ISO.230-1 2012] :

- dr : Erreur de mouvement radial (perpendiculaire à la ligne moyenne)
- dx : Erreur de mouvement axial (coaxial à la ligne moyenne)
- δr : Erreur de mouvement d'inclinaison (angle par rapport à la ligne moyenne)
- δx : Erreur de positionnement angulaire du mouvement

Ces erreurs sont illustrées dans l'annexe 1. Pour cette norme, l'erreur de positionnement angulaire correspond à la justesse angulaire. Cette description des erreurs exploite la symétrie de révolution de l'articulation que beaucoup d'auteurs utilisent pour n'avoir que 4 directions dans lesquelles décrire les erreurs [Marie et al. 2013]. Sur chacune de ces 4 directions il est possible de décrire une déformation et un jeu, ce qui donne alors 4 paramètres de raideur et 4 paramètres de jeu. Il est à noter, pour adapter le niveau de détail du modèle à l'application, que les erreurs suivant dr et dx sont généralement assez faibles [Schneider et al. 2015].

Ensuite, la déformation des segments est souvent décrite en s'appuyant sur la théorie d'Euler-Bernoulli (théorie des poutres) [Khalil et Besnard 2002, Marie et al. 2013]. Elle est parfois remplacée par la théorie de Timoshenko quand la poutre n'est pas mince et que l'on ne peut pas négliger les effets de cisaillement [Timoshenko 1922]. La poutre est soumise à un chargement à son extrémité ainsi qu'à son poids propre. Le torseur des efforts sur l'extrémité de la poutre est calculé à partir d'une formulation d'équilibre local (traction, flexions, torsion, moments de flexion), puis les déplacements sont obtenus avec les formules usuelles liant effort et déplacement (ici Timoshenko) :

$$N = ES \frac{\partial a}{\partial x} \quad ; \quad T_y = GS \left(\frac{\partial b}{\partial x} - \gamma \right) \quad ; \quad T_z = GS \left(\frac{\partial c}{\partial x} + \beta \right) \quad \text{Eq. 7}$$

$$M_t = GI_0 \frac{\partial \alpha}{\partial x} \quad ; \quad M_{f_y} = EI_y \frac{\partial \beta}{\partial x} \quad ; \quad M_{f_z} = EI_z \frac{\partial \gamma}{\partial x} \quad \text{Eq. 8}$$

Ces équations nécessitent pour chaque segment, les valeurs des moments quadratiques I_y et I_z , du moment quadratique polaire I_0 , des modules de Young et de Coulomb respectivement notés E et G , et de la section de la poutre S . Ces 6 paramètres suffisent à modéliser les déplacements suivant les 6 directions à l'extrémité d'un segment (3 translations $[a, b, c]$ et 3 rotations $[\alpha, \beta, \gamma]$). L'utilisation de la théorie des poutres est pratique et permet un calcul rapide et peu coûteux des déformées des segments du robot [Deblaise 2008]. Mais l'hypothèse poutre (longueurs grandes par rapport aux dimensions transverses) ne convient pas à tous les segments du robot.

Il existe donc une autre méthode pour décrire les déformations dans les segments : la modélisation éléments finis. Cette modélisation, aussi appelée FEM (Finite Elements Modeling), utilise les éléments finis afin de simuler et calculer les déformations d'un segment [Bouzgarrou et al. 2004, Wang et al. 2017]. Ce calcul requiert une bonne connaissance de la géométrie des pièces du robot et des propriétés des matériaux employés, ce qui peut être contraignant pour un simple utilisateur du robot. De plus, à cause des difficultés à obtenir les conditions aux limites, les prédictions sur les déformations par modélisation éléments finis sont limitées [Yang et Sadler 1992, Gong et al. 2000]. Cette méthode est généralement utilisée quand le robot est complètement maîtrisé, par des fabricants de robots notamment.

Outre les déformations sous effort et les jeux, on retrouve des phénomènes de déformation liés à la température ou à la cinématique du robot. Ces phénomènes peuvent être pris en compte par des modèles dynamiques du comportement du robot [Lee et Shah 1988, Neuman et Khosla 1985, Piras et al. 2005], et des modèles de déformations thermiques [Gong et al. 2000, Li et Zhao 2016]. Ces ajouts sont complexes à mettre en place et les paramètres correspondants sont difficiles à identifier, mais sont utiles pour les robots dont la dynamique est très sollicitée, ou fonctionnant longtemps, induisant donc une chauffe des composants du robot.

Le tableau 5 de synthèse reprend les différents paramètres abordés. Le modèle élasto-géométrique du chapitre 3 se base sur une sélection de ces paramètres non géométriques, utiles dans notre contexte. La sous-section suivante traite de l'intégration de ces paramètres à un modèle géométrique existant.

Modèle	paramètres P	Description
[Zhang et al. 2005]	1P	Déformation (efforts) justesse angulaire
[Whitney et al. 1985]	1P	Jeu justesse angulaire
[Tian et al. 2018]	>2P	Théorie du contact pour jeu et déformation angulaire
[ISO.230-1 2012, Marie et al. 2013]	8P	Jeux et déformations (efforts) pour une articulation complète
[Khalil et Besnard 2002]	5P	Déformations (efforts) pour segment complet, théorie des poutres
[Bouzgarrou et al. 2004]	21P	Déformations (efforts) pour segment complet, matrice rigidité
[Piras et al. 2005]	37P	Déformations (dynamique) pour segment complet, matrice dynamique
[Li et Zhao 2016]	~5P	Déformations (température) pour segment ou articulation

Tableau 5 : Description de modèles de paramètres non géométriques

1.3.4 Modèle élasto-géométrique

Il existe différentes méthodes pour intégrer des paramètres non géométriques à un modèle géométrique. La plupart des travaux intègrent ces paramètres de deux façons différentes : sous forme d'une matrice de raideur, ou sous forme de matrices de passage élastiques.

■ Les défauts non géométriques peuvent être considérés sous la forme d'une matrice de raideur globale du robot K_x présentée dans l'équation 9. Cette matrice permet de calculer les déflexions du robot au niveau de l'effecteur dP en fonction de la configuration, de la charge q (sous forme d'un torseur des efforts appliqué à l'effecteur), et d'autres phénomènes tels que la température. Ce calcul des déformations résultantes sur l'effecteur permet ensuite de corriger une trajectoire déjà générée avec un modèle uniquement géométrique [Klimchik et al. 2015]. Cette matrice de rigidité donne donc un modèle des erreurs non géométriques sur la trajectoire. Le calcul des déformations sur l'effecteur permet également de rechercher des poses où le robot dispose de la meilleure rigidité pour effectuer une opération sensible comme de l'usinage [Dumas et al. 2011, Abele et al. 2008].

$$q = K_x dP \quad \text{Eq. 9}$$

Classiquement la matrice K_x est obtenue à partir d'une matrice K_θ reliant des déplacements $d\theta$ sur les coordonnées articulaires aux couples Γ appliqués sur ces articulations. On ne considère alors comme paramètre non géométrique, que les déformations élastiques angulaires autour de chaque articulation (justesse angulaire), jugées prépondérantes. On peut alors lier les matrices K_θ et K_x à l'aide de la jacobienne J_θ [Alici et Shirinzadeh 2005, Abele et al. 2007, Wu 2014] :

$$\Gamma = K_\theta d\theta \quad \text{Eq. 10}$$

$$K_x = J_\theta^{-t} K_\theta J_\theta^{-1} \quad \text{Eq. 11}$$

L'équation 11 est obtenue à partir des équations 9 et 10, ainsi qu'avec l'équation 5 et le principe des travaux virtuels. Cependant des travaux montrent que l'équation 11 ne respecte pas le principe de conservation de l'énergie [Alici et Shirinzadeh 2005] et ajoutent dans leur calcul une matrice dite complémentaire K_c [Dumas et al. 2011], incorporant les interdépendances entre les paramètres :

$$K_x = J_\theta^{-t} (K_\theta - K_c) J_\theta^{-1} \quad \text{Eq. 12}$$

Afin d'être plus rigoureux, des auteurs ont cherché à prendre en compte également les masses propres des segments du robot qui agissent elles aussi sur la pose au même titre que la charge sur l'effecteur q [Klimchik et al. 2012]. Il faut également ajouter les autres déformations dans les articulations (inclinaison, radiale, axiale) ainsi que les déformations des segments pour aboutir à une modélisation élastique et une matrice de rigidité K_x complètes [Marie et al. 2013]. Les matrices de rigidité alors nécessaires pour les segments peuvent être calculées par éléments finis (FEM) [Wang et al. 2017], ou avec la théorie des poutres en suivant la méthode MSA (Matrix Structural Analysis) [Khalil et Besnard 2002].

Bien que la matrice de rigidité soit utile pour étudier la raideur du robot sur l'effecteur, on va plus souvent retrouver des méthodes intégrant le comportement non géométrique des segments et articulations directement au sein du modèle géométrique. En effet une méthode de correction de la trajectoire est peu flexible et permet difficilement de calculer des matrices utiles comme la matrice jacobienne.

■ Les défauts non géométriques peuvent également être considérés sous la forme de matrices de passage élastiques T_{elasti} . Elles prennent en compte l'élasticité des segments i et des articulations i , et s'intercalent avec les matrices géométriques pour former un modèle élasto-géométrique (équation 13). Avec un tel modèle, on peut directement calculer la pose de l'effecteur soumise aux défauts non

géométriques et donc générer directement des trajectoires correctes à l'inverse de la correction qui était effectuée avec la matrice de raideur.

$$T_{\text{base,outil}} = T_{\text{base},1} T_{\text{elast}1} T_{d_{1,2}} T_{\text{elast}2} T_{h_{2,3}} \dots T_{d_{n-1,n}} T_{\text{elast}n} T_{n,\text{outil}} \quad \text{Eq. 13}$$

Là aussi il existe différents niveaux de modélisation. Lorsqu'il y a peu de paramètres non géométriques à prendre en compte, il est possible de simplement les ajouter au modèle géométrique, directement dans les matrices existantes [Whitney et al. 1985]. Sinon on peut décrire les paramètres non géométriques des segments et des articulations dans des matrices à part, comme dans l'équation 13. C'est ce que fait la méthode VJM (Virtual Joint Modeling) de manière plus flexible : elle intercale des torseurs virtuels comme sur la figure 11 dans le modèle géométrique tels des « coquilles vides à remplir » [Klimchik et al. 2014, Pashkevich et al. 2011, Wu 2014]. L'utilisateur est alors libre de prendre en compte les paramètres non géométriques de son choix dans ces torseurs. De plus ces torseurs sont au plus proche de l'élément concerné, car la méthode VJM sépare segment et articulation du modèle géométrique :

$$T_{\text{base,outil}} = T_{\text{seg}1} T_{\text{elastseg}1} T_{\text{art}1} T_{\text{elastart}1} T_{\text{seg}2} \dots T_{\text{art}n} T_{\text{elastart}n} T_{n,\text{outil}} \quad \text{Eq. 14}$$

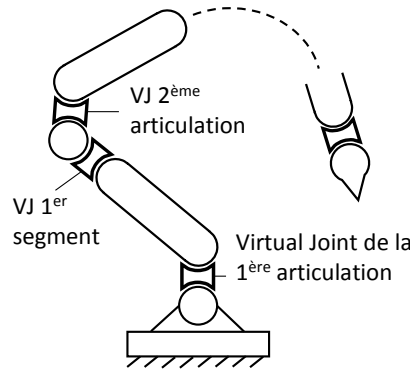


Figure 11 : Schéma modélisation VJM

Les matrices homogènes de passage élastiques sont construites à partir des déplacements (3 translations et 3 rotations) des segments et articulations dus aux phénomènes non géométriques [Khalil et Besnard 2002]. Cette construction est donnée dans les équations 15 et 16, où l'on passe d'un torseur à 6 composantes à une matrice homogène de passage T_{elast} comprenant les rotations R et les translations T (Q matrice de rotation 3×3).

$$\begin{bmatrix} dx & d\alpha \\ dy & d\beta \\ dz & d\gamma \end{bmatrix} = \begin{bmatrix} T & R \end{bmatrix} \quad ; \quad Z_R = \frac{R}{\|R\|} \quad ; \quad \theta = \|R\| \quad \text{Eq. 15}$$

$$T_{\text{elast}} = \begin{bmatrix} Q(Z_R, \theta) & T \\ 0_3 & 1 \end{bmatrix} \quad \text{Eq. 16}$$

Ainsi les défauts non géométriques peuvent être pris en compte de deux façons différentes : soit sous la forme d'une matrice de raideur globale, soit sous la forme de matrices de passage. Si l'on souhaite directement générer une trajectoire à partir du modèle élasto-géométrique on utilisera la représentation sous forme de matrices de passages et si l'on souhaite corriger une trajectoire déjà existante, on préférera la matrice de raideur. Ce choix dépend bien souvent des possibilités de commande du robot au travers de l'armoire de commande (commande ouverte ou fermée).

Tout comme avec le modèle géométrique, il est possible de calculer une jacobienne élastique J_{welast} . Cette matrice va lier le vecteur dw représentant les variations sur les paramètres non géométriques du modèle élasto-géométrique, aux variations dP sur la pose de l'effecteur.

$$dP = J_{\text{welast}} dw \quad \text{Eq. 17}$$

Enfin, l'utilisation d'un modèle élastique, et notamment de paramètres de déformation sous effort, suppose la connaissance des torseurs d'efforts dans chaque liaison et à l'extrémité de chaque segment. Afin de connaître les déformations des éléments du robot, il faut donc une méthode pour calculer ces efforts. La tâche peut sembler simple en connaissant la charge sur l'effecteur et le poids propre de chaque segment, mais c'est un calcul d'équilibre itératif qu'il faut mettre en place. En effet une fois les déformées calculées, la pose (position et orientation) de chaque segment et articulation est mise à jour, et un changement dans la pose signifie que les torseurs d'efforts sur ces éléments doivent être à nouveau calculés. Cette notion d'équilibre à atteindre est une source d'erreur si elle n'est pas prise en compte. Un calcul itératif des déformations et des torseurs d'efforts est alors mis en place suivant la figure 12. Ce calcul permet de déterminer la configuration d'équilibre du modèle total (élasto-géométrique) avec un critère d'arrêt sur la convergence des déformées [Khalil et Besnard 2002].

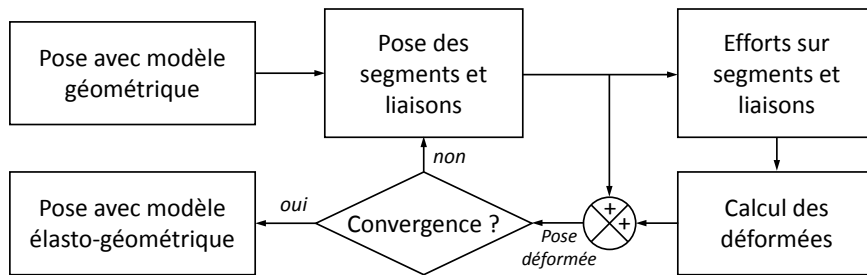


Figure 12 : Convergence de la pose déformée du robot

Ce calcul récursif de la pose déformée peut également être directement inclus dans la méthode de modélisation afin de rendre le calcul plus rapide, comme avec la méthode VJM [Klimchik et al. 2012]. Néanmoins, le gain en rapidité se fait au détriment de la flexibilité de la méthode. En effet avec des itérations dans la modélisation, il devient impossible de prendre en compte des phénomènes non linéaires comme les jeux.

La modélisation élastique choisie dans le chapitre 3, sera inspiré de la méthode VJM qui est très flexible. Toutefois, nous n'intégrerons pas les itérations du calcul des efforts afin de pouvoir modéliser tous les phénomènes influents pour notre application de numérisation.

1.3.5 Sélection d'un modèle

La sélection d'un bon modèle, représentant bien le robot et répondant aux besoins de l'application (numérisation ici), passe par l'étude de caractéristiques. Les premières caractéristiques qui ont été étudiées dans les modèles sont la complétion, la proportionnalité et l'équivalence [Everett et al. 1987]. De nos jours, les critères et termes utilisés sont plutôt la complétion, la continuité, la redondance et l'identifiabilité des paramètres [Xuan et al. 2014]. Bien souvent on retrouve également l'irréductibilité ou la minimisation du nombre de paramètres [Zhuang 1989] dont le but ressemble à la complétion proposée par Everett (équation 18).

Disposer d'un modèle complet signifie avoir un nombre de paramètres suffisant pour représenter la cinématique du robot. De plus, ces paramètres doivent pouvoir décrire toute déviation entre la cinématique nominale et la cinématique réelle du robot. La notion de modèle complet s'applique bien à la sélection de modèles géométriques, mais est inappropriée pour les modèles élasto-géométriques ayant plus de paramètres et décrivant plus que la simple cinématique du robot. La formulation de ce nombre minimum de paramètres à atteindre pour espérer un modèle complet, s'exprime en fonction du nombre de liaisons rotoïdes r (pivots) et du nombre de liaisons prismatiques p (glissières) du robot [Everett et al. 1987] :

$$c = 4r + 2p + 6 \quad \text{Eq. 18}$$

Ici, les 6 paramètres en plus servent à positionner le robot par rapport à un repère extérieur. Cette formule est souvent associée au modèle DH afin de prouver que cette modélisation à 4 paramètres est suffisante. Un modèle géométrique peut être complet ou non, mais cette notion ne s'applique pas aux modèles élasto-géométriques car la formule de l'équation 18 ne peut décrire que la simple cinématique du robot.

Un autre critère, qui met à mal la modélisation DH, doit être pris en compte : c'est la continuité ou la proportionnalité. Un modèle est dit continu si une petite variation de la pose de l'effecteur lors de l'identification entraîne une petite variation des paramètres identifiés. Ce qui n'est malheureusement pas le cas de la modélisation DH lorsque deux axes consécutifs sont parallèles, ce qui entraîne l'utilisation du paramètre de Hayati. D'après sa définition, on remarque que la continuité est liée à la méthode d'identification du modèle. D'ailleurs beaucoup de critères permettent non pas la sélection d'un modèle, mais la sélection d'un ensemble {modèle ; méthode d'identification}.

Afin d'obtenir un modèle continu, il est possible d'ajouter des paramètres en plus dans le modèle. Mais il faut être prudent, car l'ajout de paramètres peut amener des redondances. Un modèle est redondant lorsque plusieurs paramètres influencent de la même façon la pose de l'effecteur. Les paramètres redondants doivent donc, soit être fixés comme constante, soit être supprimés du modèle afin de pouvoir réaliser l'identification des paramètres.

L'identifiabilité est également un critère qui permet de détecter des paramètres redondants. Plus globalement, l'identifiabilité décrit la possibilité d'identifier expérimentalement la valeur d'un paramètre. Il y aura donc des paramètres dits identifiables, que l'on pourra identifier lors d'une étape d'identification des paramètres, et des paramètres dits non identifiables. Cependant, l'étude de l'identifiabilité est une étude continue et les paramètres ne sont pas directement classés en deux catégories, mais plutôt hiérarchisés. C'est un critère fixé par l'utilisateur qui permettra un tri entre paramètres identifiables et non identifiables. Cette notion est liée à l'observabilité des paramètres, c'est-à-dire l'impact d'un paramètre sur le positionnement de l'effecteur. Les définitions d'identifiabilité et d'observabilité peuvent changer d'un auteur à l'autre car ce sont des termes non normés, nous allons donc prendre le soin de leur donner une définition commentée ici :

- L'identifiabilité est une notion propre à chaque paramètre. Le critère de sélection déterminant si un paramètre est identifiable ou non est généralement donné par la précision de calcul de l'ordinateur utilisé. Elle permet donc de connaître si un paramètre peut être identifié ou si son influence sera perçue comme du bruit lors du calcul d'identification.
- L'observabilité est l'étude de l'influence des paramètres sur la pose de la partie mesurable du robot (généralement l'effecteur), pour une configuration donnée. C'est un critère plutôt utilisé pour observer les interactions entre les paramètres et pour sélectionner les meilleures configurations possible pour la phase de mesure de l'identification. C'est donc une notion plus globale, sur le modèle entier. Elle va permettre d'agir sur la qualité de l'identification. L'observabilité sera utilisée dans la section 1.3.6 sur l'identification des paramètres.

L'étude de l'identifiabilité est basée sur l'étude des colonnes de la matrice jacobienne J_w pour le modèle géométrique, ou $J_{w_{elast}}$ pour le modèle élasto-géométrique. L'estimation de l'identifiabilité des paramètres se fait donc dans une configuration et donnera des résultats pouvant être différents dans une autre configuration. Tout d'abord, si une de ces colonnes est nulle ou quasi nulle au vu de la précision de calcul, c'est que le paramètre associé à cette colonne n'est pas identifiable car son influence ne sera pas perçue. Ensuite, il est possible que la combinaison d'un groupe de colonnes puisse être proche d'une autre colonne, c'est donc que ces paramètres ne sont pas indépendants et qu'ils ne pourront pas être tous identifiés en même temps. Pour avoir des paramètres indépendants, c'est-à-dire ne pas avoir un paramètre ou un groupe de paramètres ayant la même influence qu'un autre paramètre sur la pose, les vecteurs colonnes de la matrice jacobienne doivent être indépendants. La jacobienne doit donc être de rang plein [Khalil et Dombre 2004], et si ce n'est pas le cas, certains des paramètres redondants doivent être fixés ou supprimés. Le rang de la jacobienne donne le nombre de paramètres identifiables. Si ce

nombre est plus petit que le nombre de paramètres du modèle c'est que certains paramètres sont non identifiables ou redondants. On peut utiliser une décomposition QR pour classer les paramètres par influence et savoir quels sont les paramètres peu influents à délaisser. Les valeurs sur la diagonale de R correspondent à chaque paramètre et sont liées à la visibilité de ces paramètres (et aux valeurs singulières de la jacobienne). En choisissant une borne minimale pour ces valeurs, on peut éliminer les paramètres peu influents. La borne inférieure la plus souvent utilisée inclut la précision de l'ordinateur :

$$\text{borne} : \tau = r\epsilon \max |R_{ii}| \quad \text{Eq. 19}$$

Avec r le nombre de lignes de la jacobienne, ϵ la précision de calcul du code utilisé, et R_{ii} les termes de la diagonale de R. Il est également possible de ne pas décomposer la matrice jacobienne et de simplement faire une étude des sous-matrices de la jacobienne [Meggiolaro et al. 2005]. Pour aller plus loin, on peut ensuite utiliser la notion d'observabilité pour vérifier que l'influence de l'ensemble des paramètres n'est pas trop hétérogène, c'est-à-dire que les paramètres à identifier n'ont pas une influence trop différente sur la pose de l'effecteur. On étudie alors le conditionnement de la matrice jacobienne J_w . Une trop grande valeur du conditionnement signifie qu'il est préférable de ne pas identifier ces paramètres ensemble, les paramètres trop influents masquant l'effet des paramètres peu influents. Les paramètres peu influents sont alors soit supprimés, soit identifiés à part après l'identification des paramètres très influents. Pour classer les paramètres par ordre d'influence on utilise également une décomposition QR. L'étude de l'identifiabilité ou de l'observabilité est donc capitale pour choisir les nombreux paramètres d'un modèle et en réussir l'identification, comme pour les modèles élasto-géométriques.

Il est facile de déceler les paramètres non identifiables, mais trouver les paramètres redondants est peu évident. Le rang de la matrice jacobienne donne le nombre de paramètres identifiables, mais ne permet pas de connaître quels sont les paramètres liés entre eux. Seule une étude des combinaisons linéaires possible (à la précision de calcul près) donne cette information, l'utilisateur devant ensuite faire son choix sur quels paramètres supprimer afin d'aboutir à un nombre de paramètres égal au rang. Une méthode originale de classification est donnée dans [Pashkevich 2001], qui fait référence à des paramètres identifiables, des paramètres semi-identifiables, et des paramètres non identifiables. Les paramètres semi-identifiables sont les paramètres redondants. Les mêmes méthodes sont utilisées pour éliminer les paramètres non identifiables avec les colonnes de la jacobienne, mais Pashkevich utilise une méthode basée sur une décomposition SVD (au lieu de QR) pour repérer les paramètres semi-identifiables.

La jacobienne J_w est calculée pour une configuration donnée (elle dépend des θ_i), or les méthodes d'identification se basent sur plusieurs configurations pour identifier les paramètres. L'identifiabilité des paramètres doit donc être estimée à partir de ces configurations, et il faut alors calculer une jacobienne pour chaque configuration. Pour une méthode classique d'identification avec mesure de la pose (position et orientation) de l'effecteur, le calcul de la jacobienne totale J_{tot_w} concaténant les jacobiennes des m configurations de mesure est le suivant :

$$dPtot = [J1_w, J2_w, \dots, Jm_w]^t dw = J_{tot_w} dw \quad \text{Eq. 20}$$

Avec $dPtot$ vecteur concaténant les m variations sur la pose de l'effecteur. Le plus souvent ces variations sont la différence entre les mesures et les poses calculées avec le modèle théorique. L'identifiabilité est alors estimée à partir de la matrice J_{tot_w} , aussi appelée matrice d'observation. Il faut un vecteur $dPtot$ au moins aussi grand que le vecteur dw pour espérer obtenir une matrice d'observation de rang plein, et le nombre de paramètres identifiables dépend alors de la taille de ce vecteur c'est-à-dire du nombre de mesures m utilisées. Dans les cas où seule la position de l'effecteur $Ptot_p$ peut être mesurée, l'équation 20 est réduite et plus de mesures sont nécessaires pour avoir une jacobienne de rang plein :

$$dPtot_p = J_{tot_w,p} dw \quad \text{Eq. 21}$$

Lors de l'utilisation de la jacobienne J_{tot_w} avec position et orientation pour certains calculs, comme le conditionnement de la jacobienne lors de l'étude d'observabilité, il faut prendre soin d'adimensionner la partie position de celle-ci, en utilisant par exemple la longueur caractéristique obtenue par minimisation du condition number [Khan et Angeles 2006]. Sans quoi il ne serait pas possible d'estimer et de comparer des influences avec des dimensions différentes. Le calcul analytique et numérique du nombre de paramètres identifiables, pour un modèle, un robot et une méthode de mesure donnés, a été explicité premièrement dans [Khalil et al. 1991]. Il a été repris dans [Guyot 1995] ou l'on peut voir le lien entre la quantité d'informations disponibles pour chaque pose (chaque configuration de mesure) et le nombre de paramètres identifiables. Le choix de la méthode d'identification est alors aussi important que le choix du modèle, car l'identification va limiter le nombre de paramètres et donc le niveau de détail que l'on peut atteindre avec le modèle.

L'identifiabilité est plutôt utilisée pour des modèles disposant d'un bon nombre de paramètres, comme des modèles géométriques très poussés, ou des modèles élasto-géométriques, où les paramètres sont plus diversifiés en termes d'échelle, d'unité, et donc moins bien maîtrisés.

Au final ces critères nous amènent à chercher dans le chapitre 2, un modèle géométrique complet, continu et non redondant pour représenter notre robot Kuka. Pour la modélisation élasto-géométrique plus poussée du chapitre 3, une étude sur l'identifiabilité des paramètres va permettre la sélection des paramètres non géométriques à mettre dans notre modélisation. Cette étude permettra dans le même temps, la sélection de configurations pour la phase d'identification et pour une méthode d'identification choisie (compatible avec le modèle qui aura été sélectionné). Ces critères nous permettent donc de sélectionner des modèles pour notre application, et il faut à présent des méthodes pour identifier les paramètres de ces modèles choisis.

1.3.6 Identification classique des paramètres d'un modèle

Afin d'identifier les paramètres du modèle, ou bien d'estimer les erreurs sur les paramètres théoriques connus, de nombreuses méthodes ont été mises au point. Ces méthodes peuvent être classées suivant différentes catégories en fonction du besoin recherché [Hollerbach et Wampler 1996]. Il y a tout d'abord les méthodes closed-loop, où l'on cherche à fermer la chaîne cinématique du robot sériel en ajoutant une contrainte sur l'effecteur. Cela peut être une contrainte sphérique, une contrainte plane, ou même une contrainte à 1 DDL avec un ball-bar [Goswami et al. 1993]. De manière générale les méthodes closed-loop utilisent des moyens de mesure avec contact, inclus dans la chaîne cinématique.

Il y a ensuite les méthodes open-loop, où l'effecteur n'est pas contraint, mais suivi en position et parfois aussi en orientation. Étant plus pratiques à mettre en place ces méthodes sont plus nombreuses [Khalil et Besnard 2002, Yin et Gao 2012, Nguyen et al. 2013]. Pour faire l'acquisition d'informations sur la pose de l'effecteur, beaucoup d'artéfacts de calibration et de techniques de mesure ont été imaginés [Wu 2014, Pashkevich 2001, Alici et Shirinzadeh 2005]. Mais à chaque fois, l'identification requiert un grand nombre de mises en position de l'effecteur, c'est pourquoi il y a également beaucoup de travaux recherchant les meilleures poses du robot pour la phase d'identification [Wu 2014, Sun et Hollerbach 2008, Borm et Meng 1991, Daney 2002, Khalil et al. 1991, Nahvi et Hollerbach 1996]. En effet, pour chaque configuration, les paramètres auront plus ou moins d'influence sur la pose du robot, et seront donc plus ou moins visibles lors de l'identification. La recherche des configurations donnant la meilleure visibilité pour les paramètres et minimisant les sources de bruits ou d'erreurs, vise à assurer une bonne identification de ces paramètres avec une incertitude minimisée. Cette recherche passe par une optimisation selon des critères, basés essentiellement sur la notion d'observabilité, c'est-à-dire sur l'étude du conditionnement de la matrice d'observation J_{tot_w} (matrice jacobienne totale) calculée dans l'équation 20. Les critères les plus employés pour la sélection de configurations sont présentés dans le tableau 6 de la page suivante.

Critère	Formule	Utilisation
[Khalil et al. 1991, Guyot 1995]	$\min_{(\theta_1, \dots, \theta_n)} \text{cond}(J_{\text{tot}_w}) = \min_{(\theta_1, \dots, \theta_n)} \frac{\lambda_{\max}}{\lambda_{\min}}$	Réduire écart de visibilité entre w_{\max} et w_{\min}
[Driels et Pathre 1990, Mooring et al. 1991]	$\text{cond}(J_{\text{tot}_w}^t J_{\text{tot}_w})$	Réduire la sensibilité de l'identification au bruit
[Borm et Meng 1991]	$\max_{\{\theta_1, \dots, \theta_n\}} \sqrt[p]{\lambda_1 \lambda_2 \dots \lambda_g}$	Maximiser la visibilité globale, peut créer de grandes disparités
[Nahvi et al. 1994, Wang et al. 2017]	$\max_{\{\theta_1, \dots, \theta_n\}} \lambda_{\min}$	Assurer un minimum de visibilité
[Nahvi et Hollerbach 1996]	$\max_{\{\theta_1, \dots, \theta_n\}} \frac{\lambda_{\min}^2}{\lambda_{\max}}$	S'éloigner des configurations générant le plus de bruit

Tableau 6 : Critères pour la sélection de configurations de mesure

Dans ce tableau, $J_{\text{tot}_w}^+$ est la pseudo-inverse de la matrice d'observation, λ_{\max} et λ_{\min} sont les valeurs singulières maximale et minimale de la matrice d'observation associées aux paramètres w_{\max} et w_{\min} ayant respectivement le plus et le moins d'influence sur la pose de l'effecteur. Pour obtenir les meilleures poses pour une identification des g paramètres, il faut donc trouver les vecteurs $\theta = (\theta_1, \dots, \theta_n)$ des coordonnées articulaires qui satisfont au mieux un ou plusieurs des critères proposés.

L'utilisation du conditionnement d'une matrice jacobienne suppose que tous les termes de cette matrice sont comparables. Or en utilisant l'orientation de l'effecteur comme information d'entrée au problème, les lignes de J_{tot_w} ont des unités différentes et le calcul de conditionnement n'a pas de sens. La jacobienne doit donc être normalisée, ce qui peut être fait en divisant la partie supérieure (lié à la position) par une valeur appelée la longueur caractéristique L_c . La matrice étant adimensionnée il est possible de travailler dessus. La longueur caractéristique s'obtient par minimisation du conditionnement (ou condition number) de la matrice normalisée J_n [Khan et Angeles 2006] :

$$\min_{\theta_1, \theta_2, \dots, \theta_n, L_c} \|J_n\| \|J_n^{-1}\| \quad \text{Eq. 22}$$

Avec la matrice jacobienne normalisée obtenue à partir de la jacobienne totale J_{tot_w} telle que :

$$J_n = \left[\frac{J_{\text{tot}_w, p}}{L_c}, J_{\text{tot}_w, o} \right]^t \quad ; \quad [dPtot_p, dPtot_o]^t = [J_{\text{tot}_w, p}, J_{\text{tot}_w, o}]^t dw \quad \text{Eq. 23}$$

Ici $dPtot_p$ et $dPtot_o$ représentent les variations sur les positions et orientations. Une fois les configurations de mesure optimales trouvées et les mesures sur la pose de l'effecteur effectuées, il faut résoudre l'identification et calculer les valeurs des paramètres à partir de ces mesures. Il y a deux types de résolution possibles : une résolution directe par inversion ou une résolution par minimisation.

■ La résolution par inversion est simple et permet d'obtenir directement les erreurs sur les paramètres du modèle [Nubiola et Bonev 2013, Nguyen et al. 2013]. En partant de l'équation 20 on calcule la pseudo inverse de la jacobienne totale pour obtenir ensuite ces erreurs :

$$dw = (J_{\text{tot}_w}^t J_{\text{tot}_w})^{-1} J_{\text{tot}_w}^t dPtot \quad \text{Eq. 24}$$

Dans l'équation 24, $dPtot$ contient les erreurs sur la pose de l'effecteur pour les m configurations de mesure, c'est-à-dire les différences entre les poses obtenues avec le modèle théorique et les poses mesurées (position et orientation). J_{tot_w} contient les matrices jacobiennes pour chacune de ces configurations de mesure. On obtient alors dw représentant les corrections à appliquer à l'ensemble des paramètres w du modèle.

■ La résolution par minimisation permet de calculer des corrections à appliquer aux paramètres, comme pour la résolution précédente [Khalil et Besnard 2002]. Afin de trouver ces corrections à appliquer aux paramètres on cherche à minimiser (dw est la variable de minimisation) :

$$dPtot - J_{tot_w} dw \quad \text{Eq. 25}$$

Cette équation peut être minimisée par une méthode classique ou bien avec un algorithme de minimisation comme celui de Levenberg-Marquardt.

Afin de prendre en compte le bruit sur la mesure et ainsi éliminer une part d'incertitude sur l'identification des paramètres, certains auteurs ajoutent parfois l'expression d'une erreur [Wu et al. 2015] :

$$dPtot - J_{tot_w} dw - \varepsilon \quad \text{Eq. 26}$$

Par ailleurs, dans un modèle élasto-géométrique ne comportant que des raideurs autour des articulations (justesse angulaire), on peut identifier à part ces paramètres k_θ en les extrayant de la jacobienne totale, à l'aide de la jacobienne articulaire et des équations 9 et 11 [Wu et al. 2015, Dumas et al. 2011] :

$$dP = (J_\theta K_\theta^{-1} J_\theta^t)^{-1} q = Adk_\theta \quad \text{Eq. 27}$$

Enfin on retrouve d'autres types de résolution moins commune et imaginée pour des cas particuliers, permettant par exemple d'identifier les paramètres un à un [Abele et al. 2007], ou utilisant un réseau de neurones [Lu et al. 1997].

Dans ces méthodes de résolution, la problématique de recherche des configurations optimales est toujours présente. Cette recherche est fastidieuse à mettre en place, c'est pourquoi notre méthode d'identification (dans le chapitre 2) s'inspire plutôt d'une méthode moins classique présentée dans la section suivante : la méthode CPA (Circle Point Analysis). Une des méthodes classiques introduite ici sera tout de même employée dans le chapitre 2, pour pouvoir effectuer un comparatif avec la méthode que nous proposons.

1.3.7 Méthode d'identification basée sur le mouvement : CPA

La méthode du Circle Point Analysis (CPA) est une méthode d'identification moins classique, sur laquelle on ne peut pas appliquer les techniques basées sur la jacobienne vues précédemment.

Cette méthode CPA est la plus facile à appréhender car elle repose sur une étude de mouvements géométriques simples, ici des rotations pour les articulations du robot, et la résolution n'est qu'un recalage sur les formes obtenues c'est-à-dire des arcs de cercle [Sklar 1989]. Outre sa simplicité, cette méthode est également appréciée pour sa représentativité de la vraie cinématique du robot, même si sa mise en place sur le robot est plus complexe car il faut instrumenter tous les axes du robot [Santolaria et Ginés 2013, Stone et al. 1986, Nubiola 2011]. De plus cette méthode ne requiert pas la connaissance de paramètres théoriques préalables, elle permet d'identifier directement les valeurs des paramètres. Mais parce qu'elle est basée sur une étude de la cinématique du robot, cette méthode n'est réservée qu'à l'identification de paramètres géométriques, même si une étude de cercles déformés par des défauts non géométriques peut être envisageable.

Dans cette méthode, les articulations du robot sont sollicitées les unes après les autres pendant qu'un système de mesure externe va suivre les mouvements résultants de l'effecteur. À chaque articulation correspond un mouvement en arc de cercle et la mesure de cet arc permet ensuite de retrouver l'axe et le centre de rotation. Les points mesurés d'un arc sont d'abord projetés dans un plan des moindres carrés, comme sur la figure 14, puis on recherche le cercle 2D passant au mieux par ces points projetés [Mooring et al. 1991]. Avec ce cercle 2D on a alors un centre et une direction de l'axe associés à une articulation,

qu'il suffit de repasser dans le repère global, ici le repère du moyen de mesure. Avec les 3 coordonnées du centre et les 3 coordonnées du vecteur de direction, la position de l'articulation du robot et son axe sont parfaitement déterminés. La connaissance de la position de chaque axe sur le robot (figure 13) permet de calculer les paramètres géométriques entre les articulations. Pour une meilleure représentation de la géométrie réelle du robot, et pour une meilleure résolution des problèmes utilisant les moindres carrés, les arcs sont pris les plus longs possible. On essaye de couvrir au maximum la disponibilité articulaire de chaque axe lors de la mesure d'un arc, afin d'être représentatif de l'articulation pour toute coordonnée articulaire. Ainsi, grâce à cette méthode, les paramètres du modèle représentent bien la géométrie réelle du robot dans tout son espace de travail et pas seulement en quelques configurations de mesure comme pour la plupart des méthodes classiques vues précédemment. Cette méthode ne tente pas de minimiser une erreur de position sur quelques configurations, mais d'estimer directement les paramètres géométriques.

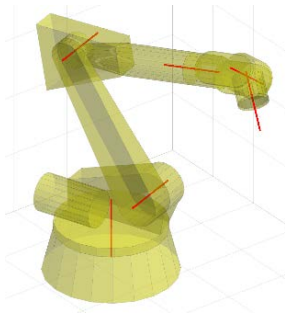


Figure 13 : Axes robot identifiés par CPA

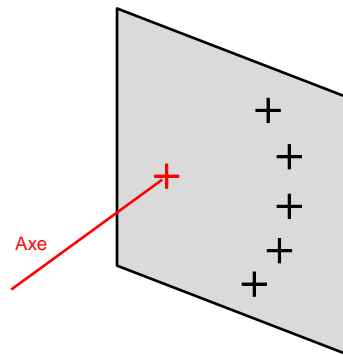


Figure 14 : Principe méthode CPA

Le moyen de mesure peut être le même que celui des méthodes d'identification classiques précédentes, à savoir un système de suivi externe (Laser tracker), qui est le choix fait en début de ce chapitre. Le point négatif de cette méthode est la lourde instrumentation à mettre en place sur chaque axe du robot. Les cibles à suivre par le système de tracking doivent alors être disposées sur chaque axe [Santolaria et Ginés 2013]. Puis les mesures sur chaque axe doivent être faites les unes après les autres individuellement, par rotation d'un seul axe à la fois. Le repère commun entre chaque mesure peut être le repère du système de mesure, qui ne doit pas être déplacé d'une mesure à l'autre. Si le système de tracking doit être déplacé, un autre repère fixe doit être utilisé à la place.

Comme mentionné précédemment, la méthode CPA ne permet en théorie d'identifier que des paramètres géométriques. On retrouve des adaptations de cette méthode utilisées pour retrouver des paramètres non linéaires. L'étude se fait alors sur un cercle déformé [Schneider et al. 2015]. Mais cette adaptation requiert une instrumentation encore plus lourde du robot ainsi qu'un temps considérable pour l'étape l'identification.

D'autres travaux reprennent en partie l'idée de la méthode CPA pour rendre leur mesure plus robuste. C'est le cas dans [Nguyen et al. 2013] pour laquelle une identification classique est mise en place, mais où une rotation de l'effecteur est faite pour chaque configuration de mesure. L'identification du cercle résultant permet d'obtenir la pose de l'effecteur de manière plus sûre qu'avec une simple prise de position.

Pour conclure cette section, la modélisation des robots permet de prendre en compte les défauts géométriques et non géométriques de ces derniers. C'est tout d'abord à l'utilisateur de choisir quel niveau de détail il désire pour la modélisation, en fonction de l'application ciblée pour le robot et de l'environnement. Il existe un grand nombre de modélisations différentes parmi lesquelles l'utilisateur peut choisir celle adaptée à son besoin. Après une première sélection d'une modélisation ad hoc et « idéalisée » (i.e. sans paramètres représentant les défauts) pour le robot, plusieurs vérifications doivent être effectuées suivant différents critères de sélection, afin d'avoir un modèle non plus « idéalisé », mais

réaliste. Dans cette étape on recherche si les paramètres représentent bien le robot et s'ils pourront être identifiés, des modifications sont alors apportées au modèle pour ajouter des paramètres manquants ou retirer des paramètres inutiles ou redondants. Enfin une méthode d'identification est associée au modèle sélectionné. Les configurations pour la phase de mesure sont alors optimisées et la mesure est ensuite réalisée. Le calcul d'identification des paramètres est finalement effectué.

La modélisation choisie pour notre application est présentée dans les chapitres 2 et 3. Le chapitre 2 se focalise sur une modélisation géométrique pratique et rapide à mettre en place afin de satisfaire les besoins de l'application (type chaîne de production). Dans ce but nous proposons une méthode d'identification des paramètres basée sur une amélioration de la méthode CPA pour gagner en rapidité et en praticité, mais pouvant aussi exploiter tous les avantages des méthodes classiques. La suite de la modélisation prenant en compte les défauts non géométriques est traitée dans le chapitre 3. Afin de sélectionner les paramètres non géométriques pour notre modèle, nous proposons dans ce même chapitre une étude de l'identifiabilité des paramètres, proche des besoins de l'application et de la physique du robot.

Cette section d'introduction bibliographique et les deux chapitres qui vont suivre sur la modélisation sont essentiels dans notre travail de mise en place d'une cellule de numérisation. En effet si l'on prend autant de soin à aboutir à une modélisation pour le robot c'est qu'elle est utilisée dans plusieurs étapes clés :

- Suivi de l'effecteur au cours de la trajectoire si l'on n'utilise pas de système de tracking.
- Générations finales des trajectoires de numérisation dans un repère articulaire.
- Base pour les indices de performance du robot, utiles pour une exploitation optimale du robot.

Le modèle du robot est une base importante pour les indices de performance car bon nombre d'entre eux se calculent à partir de l'expression de la jacobienne. Dans la section suivante, la présentation de différents indices en robotique, de leur formulation et de leur utilisation, permettra, dans le chapitre 4, de sélectionner les indices pertinents pour notre application.

1.4 Exploitation des performances du robot

Des indices de performance du robot sont nécessaires pour connaître les capacités du robot dans son espace de travail, et ce afin de pouvoir optimiser l'utilisation du robot. Dans notre cas ils sont utiles dans la stratégie de génération de trajectoire pour trouver des zones de travail et une ébauche de trajectoire qui permet au robot de respecter les exigences de l'opération de numérisation. Ces mêmes indices sont ensuite utilisés pour optimiser la trajectoire de numérisation suivant un critère, généralement la vitesse d'exécution ou la qualité de pose. Cette section est donc dédiée à la présentation d'indices de performance les plus employés en robotique [Moreno et al. 2012], et dépendant de la configuration du robot.

Les indices les plus utilisés pour générer des trajectoires concernent principalement la qualité de pose de l'effecteur et sa vitesse. Les formulations de ces indices dans lesquelles on retrouve l'expression de la vitesse de l'effecteur ou de sa pose s'appuient alors sur le modèle du robot, et plus particulièrement sa jacobienne articulaire J_0 , c'est pourquoi une bonne modélisation du robot est si importante. Un modèle représentant fidèlement le comportement du robot permettra d'obtenir des indices de performance représentant fidèlement les capacités du robot.

Mais d'autres indices requièrent des mesures supplémentaires et ne peuvent être décrits uniquement avec la modélisation du robot. On retrouve notamment des indices permettant d'aboutir à une trajectoire continue et bien lissée ou de maîtriser les incertitudes du robot. D'autres même s'attaquent à la consommation d'énergie de l'opération.

1.4.1 Indices basés sur la jacobienne

Un des premiers indices créés pour obtenir des informations sur le comportement de l'effecteur est l'ellipsoïde de vitesses. Cet indice permet tout simplement de calculer un ellipsoïde représentant les vitesses possibles sur l'effecteur suivant n'importe quelle direction, pour une configuration donnée. L'utilité ne réside pas dans le calcul d'une vitesse suivant une direction, mais dans la possibilité de comparer les axes de l'ellipse et voir s'il y a une bonne répartition des capacités en vitesse. Le calcul se base sur la jacobienne articulaire de l'équation 5. On fixe une valeur maximale pour les vitesses des articulations afin de générer les bornes du futur ellipsoïde, puis on transforme cette limite sur les articulations en une limite sur la position de l'effecteur en utilisant la jacobienne articulaire J_θ et la variation sur la pose de l'effecteur dP tel que :

$$\|\dot{\theta}\|^2 = \dot{\theta}^t \dot{\theta} = \dot{P}^t J_\theta^{-t} J_\theta^{-1} \dot{P} \quad ; \quad \|\dot{\theta}\| \leq 1 \quad \text{Eq. 28}$$

Ce qui donne la formulation de l'ellipsoïde suivante :

$$\dot{P}^t J_\theta^{-t} J_\theta^{-1} \dot{P} \leq 1 \quad \text{Eq. 29}$$

L'ellipsoïde des vitesses sur l'effecteur pour des vitesses articulaires fixées est ainsi obtenu. Il est également possible de calculer les axes de l'ellipsoïde à partir des valeurs singulières de la jacobienne comme sur la figure 15 [Moreno et al. 2012]. Il n'y a que trois axes sur un ellipsoïde et pourtant on peut obtenir 6 valeurs singulières si P représente la position et l'orientation de l'effecteur. En réalité il y a un ellipsoïde pour la génération de vitesses de translation et un autre ellipsoïde pour les vitesses de rotation. Quand une valeur singulière tend vers 0, l'ellipsoïde s'aplatit, ce qui signifie que l'on atteint une configuration singulière.

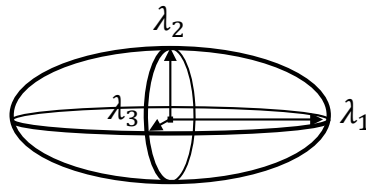


Figure 15 : Ellipsoïde de vitesses

L'ellipsoïde de vitesse permet de bien visualiser la distribution des vitesses sur l'effecteur. Mais il est souvent plus pratique d'utiliser une valeur d'indice plutôt qu'une représentation, afin de pouvoir plus facilement comparer des configurations. Dans la continuité de l'ellipsoïde de vitesse, on retrouve alors l'indice de manipulabilité.

L'indice de manipulabilité créé par Yoshikawa [Yoshikawa 1985] est un critère de performance relié à la vitesse ou la qualité de pose de l'effecteur du robot. Il définit le potentiel de mouvement de l'effecteur pour une configuration donnée c'est-à-dire la capacité du robot à générer de la vitesse au niveau de l'effecteur. Pour un robot non redondant à 6 axes, le calcul de la manipulabilité dans l'équation 30 est basé sur le produit des valeurs singulières de la matrice jacobienne J_θ du modèle du robot.

$$I_{\text{manip}} = |\det(J_\theta)| = \lambda_1 \lambda_2 \lambda_3 \dots \lambda_6 \quad \text{Eq. 30}$$

Cet indice représente donc la capacité en vitesse de l'effecteur, et quand une valeur singulière tend vers 0, l'indice tend vers 0 ce qui permet de détecter puis d'éviter les configurations singulières. Un indice petit signifie que le robot ne peut pas générer d'importantes vitesses. Dans ce cas un grand mouvement sur les articulations crée un petit mouvement de l'effecteur. La résolution du robot est donc meilleure pour un petit indice. L'indice de manipulabilité représente la capacité en vitesse du robot et l'inverse de la manipulabilité représente la qualité de pose du robot. Cet indice peut être utilisé pour rechercher de la vitesse d'exécution ou une bonne résolution et une bonne qualité de mise en position. Cependant si

l'on prend en compte la vitesse de translation et de rotation de l'effecteur, il faut adimensionner la partie de la jacobienne liée à la translation (généralement la moitié supérieure de la matrice), avec la longueur caractéristique L_c (obtenue avec une minimisation du condition number).

Un autre indice très utilisé est le condition number c'est-à-dire le conditionnement de la matrice jacobienne. L'étude ressemble à celle faite dans la présentation des méthodes d'identification, mais avec la jacobienne articulaire J_θ . Cet indice est notamment utilisé par Khan [Khan et Angeles 2006], afin de limiter la propagation d'erreur dans les calculs basés sur le modèle du robot. La formulation de cet indice s'obtient simplement avec celle du conditionnement :

$$I_{cn} = \frac{1}{6} \sqrt{\text{tr}(J_\theta J_\theta^t) \text{tr}((J_\theta J_\theta^t)^{-1})} \quad \text{ou bien} \quad I_{cn} = \frac{\lambda_{\max}}{\lambda_{\min}} \quad \text{Eq. 31}$$

On voit alors que le condition number est un indice utile pour connaître la répartition des vitesses et pour se tenir éloigné des configurations singulières. Un indice proche de 1 indique une bonne répartition et un indice très haut montre que l'on perd un degré de liberté pour générer de la vitesse. Mais il y a d'autres applications pour cet indice, comme le contrôle de la dextérité du robot pour générer des trajectoires [Zargarbashi et al. 2012]. Comme le condition number est l'image de la distribution du mouvement de l'effecteur sur toutes les articulations, un indice bas permet d'aboutir à une trajectoire plus lisse car le mouvement est mieux réparti entre les articulations. On a donc moins de variations sur chaque axe et pas de pic de sollicitation. Pour Dumas [Dumas et al. 2011], le condition number aide à la sélection de zones dans lesquelles on va réaliser l'identification de paramètres sensibles comme les raideurs articulaires du robot.

On peut citer pour finir l'indice de rigidité [Guo et al. 2015] servant à faire des cartographies de la rigidité au niveau de l'effecteur du robot. Cet indice est très apprécié pour des opérations impliquant des efforts importants sur l'effecteur [Lin et al. 2017]. Une bonne rigidité permet au robot de bien suivre une trajectoire malgré des efforts. Cet indice I_{rig} se calcule à partir de l'équation 11, pour avoir tout d'abord l'inverse de la matrice de rigidité de l'effecteur :

$$C = K_x^{-1} = J_\theta K_\theta^{-1} J_\theta^t = \begin{bmatrix} C_{tt} & C_{tr} \\ C_{rt} & C_{rr} \end{bmatrix} \quad \text{Eq. 32}$$

Puis on en déduit :

$$I_{rig} = \frac{1}{\sqrt[3]{\det(C_{tt})}} \quad \text{Eq. 33}$$

Ici le calcul est simple et repose uniquement sur la rigidité des articulations du robot, mais des indices sur la rigidité des robots comprenant plus de détails, comme les jeux, existent aussi [Schneider et al. 2015], afin d'avoir une meilleure précision de pose pour des applications à grande sollicitation en effort sur l'effecteur (usinage ou perçage).

1.4.2 Indices basés sur des mesures ou des caractéristiques du robot

La disponibilité articulaire est un indice pratique pour lisser la trajectoire, il permet de contrôler la position des articulations par rapport à leurs butées. Cet indice a été initialement utilisé par Liegeois pour rester loin des butées articulaires et pour conserver un potentiel de mouvement [Liegeois 1977]. La trajectoire alors obtenue est plus lissée et les sollicitations de mouvement sont bien réparties entre les articulations, comme pour l'utilisation du condition number par Zargarbashi. Cet indice est généralement combiné avec d'autres indices capables de situer les configurations singulières comme la manipulabilité ou le condition number.

Cet indice est simplement calculé à partir des limites sur les n articulations en prenant la distance entre la coordonnée actuelle et la coordonnée la plus éloignée des butées pour chaque articulation i , comme sur l'équation 35. Une disponibilité articulaire qui tend vers 1 signifie que les axes sont proches de leurs limites, tandis qu'un indice proche de 0 signifie que l'on possède une disponibilité maximale.

$$\Delta\theta_i = \left| \theta_i - \frac{\theta_{i\max} + \theta_{i\min}}{2} \right| \quad \text{Eq. 34}$$

$$I_{da} = \frac{1}{n} \sum_1^n \left(\frac{\Delta\theta_i}{\Delta\theta_{i\max}} \right)^2 \quad \text{Eq. 35}$$

Cet indice aide à rester éloigner des limites articulaires pour obtenir une trajectoire plus lissée, sans à-coup de changement de configuration quand un axe est trop proche de sa limite. De plus, les cartographies de cet indice indiquent le domaine accessible du robot, ce qui aide à rapidement éliminer les trajectoires impossibles dans l'espace de travail du robot. Ainsi on combine également de nombreux indices de performance avec la disponibilité articulaire pour aboutir sur les performances que l'on peut réellement atteindre et espérer avec le robot.

Il existe des indices plus récents traitant également des limites articulaires des axes comme l'indice global présenté dans [Moreno et al. 2010]. Cet indice s'inspire grandement du précédent pour déterminer si une configuration est accessible ou non. L'indice de disponibilité globale dans la formule suivante doit être compris entre 0 et 1 pour que la configuration soit accessible.

$$I_{dg} = \max_i \frac{\Delta\theta_i}{\Delta\theta_{i\max}} \quad \text{Eq. 36}$$

Un autre indice bien connu est la répétabilité. La répétabilité est l'incertitude sur la pose du robot. Ainsi, sur des mouvements répétés de mise en position de l'effecteur, la répétabilité qualifie la répartition gaussienne sur cette mise en position. La définition exacte pour un système robotisé est la suivante :

- La répétabilité de pose exprime l'étroitesse de l'accord entre les poses atteintes pour la même pose commandée répétée n fois suivant la même direction [ISO.9283 2015].

Bien sûr les mesures faites pour estimer la répétabilité du robot doivent respecter les conditions de répétabilité décrites par le Vocabulaire International de Métrologie [BiPM et al. 2008]. Cet indice est habituellement donné sous la forme d'une valeur maximale obtenue sur l'espace de travail du robot, qui sera la valeur de répétabilité que l'on peut au moins espérer pour une configuration donnée. Pourtant nous avons vu jusqu'ici combien il est important de pouvoir prédire le comportement du robot et non pas d'avoir une simple borne que l'on peut espérer. De plus la répétabilité dépend énormément de la configuration du robot, il est donc intéressant de dresser une cartographie de la répétabilité de pose de l'effecteur, dans l'espace de travail du robot, afin de maîtriser l'incertitude sur cette pose.

Il est possible de mesurer de manière empirique la répétabilité du robot pour chaque configuration dans son espace de travail, mais c'est un travail long dont la quantité d'informations à stocker est considérable. Il est plus intéressant de chercher un moyen de modéliser cette incertitude. C'est ce que propose Brethé dans ses travaux [Brethé et al. 2006], où il reconstruit la répétabilité sur l'effecteur pour une configuration à partir du modèle géométrique du robot et de la répétabilité en positionnement angulaire de chaque articulation. Le postulat de base est le suivant : chaque articulation possède une répétabilité sur son positionnement angulaire qui peut être représentée par un écart type σ_i . À partir de l'expression d'une loi normale et de l'équation 5, on pose :

$$dP = J_\theta d\theta \quad ; \quad D = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_n \end{bmatrix} \quad ; \quad C = J_\theta D J_\theta^t \quad \text{Eq. 37}$$

Et on parvient à la densité de probabilité sur la pose de l'effecteur à partir des écarts types sur chaque axe :

$$g(dP) = \rho \exp \left[-\frac{1}{2} dP^t C^{-1} dP \right] \quad \text{Eq. 38}$$

Où la constante ρ est obtenue également à partir de la concaténation des lois normales sur chaque articulation. Cette densité de probabilité permet de construire un ellipsoïde de répétabilité sur l'effecteur pour une configuration donnée. En effet la répétabilité dépend de la direction observée, elle n'est pas la même suivant la direction pour une même configuration. Cet indice aide donc à maîtriser les incertitudes du robot, à sélectionner des zones de confiance et optimiser la direction d'avance pour la génération de trajectoire.

De nombreux autres critères existent, mais sont moins utilisés. Certains sont basés sur ceux déjà présentés, d'autres visent à qualifier d'autres grandeurs physiques comme par exemple la consommation d'énergie [Vergnano et al. 2012]. Parmi tous ces indices rappelés dans le tableau 7, il faut faire une sélection de celui ou de ceux qui seront utiles à l'application. Dans notre stratégie de génération de trajectoire, les indices servent à sélectionner des zones dans l'espace de travail du robot où ce dernier sera apte à respecter les contraintes imposées sur l'ébauche de trajectoire, ébauche obtenue à partir de la CAO, du cahier des charges sur la numérisation et des performances du capteur laser. D'autre part ils servent aussi à optimiser la génération de la trajectoire finale à partir de l'ébauche, en recherchant une meilleure vitesse d'exécution, une meilleure qualité de pose ou encore la valeur optimale d'un autre critère. Plusieurs indices de performance peuvent donc entrer en compte. Lorsque plusieurs indices sont utilisés, il est possible comme Lin [Lin et al. 2017] de les prendre en compte les uns après les autres lors de la génération de trajectoire, du plus important au moins important. Mais il peut être intéressant de les appliquer tous en même temps, avec des pondérations. Le chapitre 4 présente une sélection et une étude des indices de performance choisis pour notre application, ainsi qu'une méthode simple de concaténation des indices avec pondération.

Indice	Référence	Information sur les capacités du robot
Ellipsoïde de vitesses	[Moreno et al. 2012]	Distribution des vitesses
Manipulabilité	[Yoshikawa 1985]	Potentiel de mouvement ou résolution
Condition Number	[Zargarbashi et al. 2012, Dumas et al. 2011]	Propagation des erreurs de calcul (et lissage de la trajectoire)
Rigidité	[Guo et al. 2015]	Rigidité sur l'effecteur
Disponibilité articulaire	[Liegeois 1977]	Distance aux butées (lissage de la trajectoire)
Indice global	[Moreno et al. 2010]	Domaine accessible
Répétabilité	[Brethé et al. 2006]	Incertitude sur la pose de l'effecteur

Tableau 7 : Indices de performance donnant les capacités de l'effecteur pour une configuration donnée

1.5 Mise en place de la cellule de numérisation

La cellule de numérisation est constituée d'un robot anthropomorphe 6 axes Kuka KR150-L130, d'un capteur laser de numérisation Zephyr, et d'un système de suivi externe C-Track. Lors de la numérisation d'une pièce, le robot effectue la trajectoire de numérisation en portant le capteur laser, et durant l'opération, le système de mesure externe fait l'acquisition de la pose du capteur laser en temps réel pour que ce dernier puisse reconstruire un nuage de point à partir de lignes laser. Nous avons vu l'importance d'une modélisation du robot dans cette opération, ainsi que le besoin d'une qualification du robot au travers d'indices de performance afin de maîtriser le robot et de générer une trajectoire optimale. À présent il faut étudier la qualification des autres éléments de la cellule, à savoir le capteur laser et le système de suivi, afin de maîtriser le comportement de la cellule entière et les différents aspects de la numérisation en particulier le couple {qualité ; vitesse}.

1.5.1 Capteur laser de numérisation

Avant de chercher à qualifier le capteur de numérisation, il faut que ce dernier soit calibré, c'est-à-dire qu'il ait en sa possession tous les paramètres et toutes les informations lui permettant de fonctionner correctement soit : la connaissance de sa propre géométrie et sa position dans son environnement. Toutes ces informations sont importantes pour d'une part réaliser une bonne acquisition d'une ligne laser représentant la géométrie de la pièce à numériser, d'autre part pour pouvoir assembler les lignes lasers acquises dans différentes configurations. Ces deux types d'information constituent les paramètres intrinsèques et les paramètres extrinsèques. La calibration intrinsèque du capteur laser est le plus souvent effectuée par le concepteur du capteur et il est rarement nécessaire de la refaire. Elle permet de connaître toutes les propriétés nécessaires à l'acquisition d'une ligne laser : le positionnement de la ligne laser, de la matrice CCD et les paramètres optiques. Avec ces informations le capteur peut capturer la ligne laser avec la matrice CCD et la positionner dans son repère interne.

La calibration extrinsèque elle, doit être effectuée lors de l'association du capteur laser avec le système qui lui donnera sa position, qui est généralement le système porteur (MMT, robot). La calibration extrinsèque permet au capteur de connaître sa configuration dans un repère de référence, c'est-à-dire le repère dans lequel on lui donnera sa position au cours de la numérisation. Dans notre situation le robot sera porteur, mais c'est le système de suivi externe qui donnera la position du capteur, la calibration doit donc être effectuée entre le capteur laser et le repère du système de suivi. Ainsi les cibles de suivi utilisées avec le système C-Track de stéréovision, doivent être disposées sur le capteur laser et non pas sur l'effecteur afin de n'introduire aucune déviation en cas de montage ou de démontage du capteur sur le robot.

En plus de la calibration, il faut également prendre en compte la qualification du capteur qui va suivre. La qualification ne sera pas effectuée dans la cellule (robot + système de suivi externe), car à ce stade la qualité de numérisation n'est pas encore assurée et une mesure avec la cellule ne peut pas servir de référence. Pour une qualification de référence, le capteur est positionné sur une MMT, les résultats obtenus sont ainsi plus fiables. La calibration extrinsèque utilise un artéfact dont les dimensions sont connues. À titre d'exemple, la plupart des entreprises proposant des capteurs laser utilisent des sphères de calibration, mais on peut retrouver des calibrations faites avec des plans, des cylindres ou même des pièces complexes [Santolaria et al. 2009]. La numérisation de cet artéfact permet au capteur laser de connaître ses paramètres extrinsèques pour réaliser une bonne numérisation.

Comme pour le robot, la qualification du capteur va reposer sur des critères de performance de numérisation souvent utilisés. Les principaux critères utilisés pour qualifier un nuage de points sont le bruit et la justesse [Van Gestel et al. 2009, Mahmud et al. 2011].

Le bruit est estimé en mesurant une surface de référence, un plan généralement, dont la géométrie est bien connue. À partir d'un nuage de m points, la formulation du bruit δ sur le nuage est la suivante :

$$\delta = \text{cst} \sqrt{\frac{1}{m} \sum_{i=1}^m (d_i - \bar{d})^2} \quad \text{Eq. 39}$$

Où d_i est la distance de chaque point avec la surface de référence, \bar{d} la moyenne de ces distances et cst le facteur de recouvrement (généralement fixé à $\text{cst}=1$). La distribution du bruit est plutôt gaussienne, la valeur de δ représente alors un écart type qui reflète la fiabilité sur les points numérisés. Des cartographies peuvent là aussi être tirées, car le bruit dépend de la distance et de l'inclinaison entre le capteur laser et la surface de référence, ces cartographies ont été réalisées pour le capteur Zephyr [Mehdi-Souzani et al. 2016]. Dans ces cartographies, la valeur du bruit est comprise entre $10,5\mu\text{m}$ et $14\mu\text{m}$ pour une variation sur la distance, et entre $11\mu\text{m}$ et $37\mu\text{m}$ pour une variation sur l'inclinaison.

La notion de justesse pour un nuage de points utilise une définition de métrologie [JCGM.200 2012] :

- La justesse de mesure est l'écart entre la moyenne d'un nombre infini de valeurs mesurées répétées et une valeur de référence.

On peut donc qualifier la justesse associée à un moyen de mesure, à l'aide d'un artéfact disposant d'une distance calibrée. Il est possible par exemple de numériser un artéfact disposant de deux plans à des hauteurs différentes et dont la distance h_0 est maîtrisée. Une formulation pour la justesse est proposée dans [Mehdi-Souzani et al. 2016], utilisant deux tels plans. En posant (A_1, n_1) le barycentre et la normale du premier nuage de points, et (A_2, n_2) ceux du deuxième, la formulation est la suivante :

$$\text{justesse} = \frac{1}{2} (A_1 A_2 n_1 + A_2 A_1 n_2) - h_0 \quad \text{Eq. 40}$$

Une cartographie fonction de la distance a également été réalisée pour la justesse par Mehdi-Souzani. La justesse y est comprise entre $-1\mu\text{m}$ et $25\mu\text{m}$. La qualification du capteur laser Zephyr a donc été réalisée sur les critères présentés, et les performances du capteur sont maîtrisées. Il reste cependant un problème sur la calibration externe du capteur. Le capteur Zephyr et le logiciel associé Polygonia sont conçus pour une mise en place facile sur MMT à 3 axes, à l'aide d'un artéfact (sphère de calibration). Mais la calibration sur un robot 6 axes n'est pas prévue dans les fonctionnalités, une stratégie de calibration doit être développée pour ce cas de figure. Le chapitre 5 de ce manuscrit traite de l'intégration du capteur laser Zephyr et du C-Track dans la cellule de numérisation. On pourra y retrouver une proposition de méthode adaptée pour la calibration du capteur.

1.5.2 Système de suivi externe

Le système de mesure externe est un système de stéréovision, le C-Track, qui permet le suivi de cibles disposées sur le robot ou sur le capteur laser. L'objectif de ce système de suivi est d'effectuer l'acquisition de la pose du capteur laser au cours de la trajectoire de numérisation. Il est également employé pour la calibration du robot. Il est donc nécessaire de disposer d'une qualification de ce système de mesure afin de maîtriser l'incertitude lors de l'identification des paramètres ou du suivi du capteur laser. Le constructeur du C-Track ne fournit pas de qualification sur les performances de suivi d'une cible. Une qualification du système a donc été effectuée afin de l'utiliser dans les meilleures conditions.

La première information connue est son champ de vision de $3,8\text{m}^3$, il faut ensuite qualifier le système C-Track en suivant les mêmes habitudes que pour un système de mesure. Comme pour le capteur laser on étudie le bruit sur la mesure d'une cible, ainsi que la justesse de mesure entre deux positions pour une cible, ou plus simplement la distance entre deux cibles. Le bruit obtenu sur la mesure d'une cible fixe par le C-Track, sur une durée de 60 secondes (fréquence d'acquisition de 29 Hz) est de $3,8\mu\text{m}$ suivant l'axe X , de $6,3\mu\text{m}$ suivant Y et de $21,4\mu\text{m}$ suivant Z . Ces valeurs sont données dans le repère du C-Track (figure 16). Ce bruit peut également être considéré comme un défaut de répétabilité que nous noterons avec des écarts types $[\sigma_x, \sigma_y, \sigma_z]$. La justesse mesurée pour deux cibles distantes de 958mm est de $26\mu\text{m}$, également pour une durée de 60 secondes.

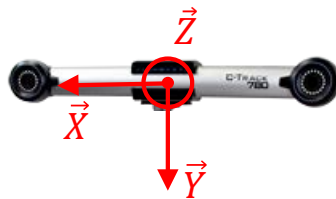


Figure 16 : Repérage C-Track

D'autres informations et précautions sont à prendre en compte lors de l'utilisation d'un tel système de suivi. Tout d'abord il est possible d'utiliser plusieurs cibles pour former un modèle de cibles, qui rend le suivi plus robuste aux changements d'orientation. Ensuite, il est possible de réduire le bruit pour les mesures de poses fixes du robot. Il suffit de calculer la moyenne des positions des points mesurés pendant un certain temps. La fréquence d'acquisition du C-Track étant de 29Hz, pour une mesure de 3 secondes on obtient 87 points. Le bruit de mesure est alors réduit à $1,6\mu\text{m}$ suivant l'axe X , à $3,2\mu\text{m}$ suivant Y et à $10,8\mu\text{m}$ suivant Z . Enfin il a été observé sur le robot que la mesure déviait au cours du temps, pour une mesure d'une durée d'une heure la déviation peut atteindre $50\mu\text{m}$. Ce phénomène est présent dans tout le champ de vision du C-Track, et la déviation est similaire pour tous les points. On peut donc facilement atténuer la déviation en passant d'une mesure absolue à une mesure relative. Jusqu'à maintenant le point était mesuré dans le repère du C-Track fixe. Pour prendre en compte la déviation qui est similaire pour tous les points, il faut ajouter un ensemble de cibles fixes dans la scène qui serviront de repère. Comme ce repère sera soumis à la même déviation que les points mesurés, exprimer la position des points dans ce repère atténuera fortement l'effet des déviations sur la mesure. On passe ainsi à une déviation de $5\mu\text{m}$ sur une heure de mesure d'un point fixe.

Une autre qualification devra être effectuée lors de l'association du système de suivi externe et du capteur laser, c'est le temps de réponse du C-Track. Lors de la numérisation le capteur laser a besoin de connaître sa pose dans un repère de référence pour chaque ligne laser acquise. Le calcul de positionnement de la ligne laser se fait en ligne durant la numérisation et non pas en post-process. L'information sur la pose doit donc être rapidement transmise au contrôleur du capteur laser car un temps long créerait un décalage entre la pose où le capteur a pris la ligne et la pose utilisée pour le calcul. Pour des vitesses rapides de numérisation, ce décalage crée un défaut important dans le nuage de point obtenu à la fin. Une qualification de ce temps de réponse permettra de savoir s'il faut chercher à l'améliorer, ou même s'il faut faire le calcul du nuage de points à partir des lignes laser en post-process.

Grâce à la qualification du C-Track, il est possible d'utiliser ce dernier dans les meilleures conditions. Avec la maîtrise des erreurs de ce moyen de mesure, il est également possible de faire un calcul de propagation de l'erreur pour connaître l'incertitude due au C-Track sur la position d'une cible suivie ou sur des paramètres identifiés du modèle du robot. Dans ce cas cela peut aider à sélectionner une méthode d'identification minimisant la propagation de l'erreur de mesure. On se servira donc de ces informations dans ce but.

Il reste alors à qualifier le résultat d'une numérisation effectuée avec la cellule complète. Il faut donc mettre en place le capteur laser sur le robot, faire la calibration de ce capteur en association avec le système de suivi. Le chapitre 5 traite de la mise en place de la cellule, et détaillera nos travaux sur la communication établie entre le Zephyr et le C-Track, et la calibration du Zephyr sur le robot. Une fois les éléments de la cellule connectés, une première qualification de la numérisation sera effectuée.

1.6 Synthèse du chapitre

En analysant les besoins actuels en robotique, notamment dans le domaine de la mesure, nous avons pu faire ressortir une problématique. Les robots sériels très adaptables et habiles (6 DDL, domaine de travail) pourraient être utilisés pour des opérations de mesure sur chaîne de production. Mais ils ne disposent pas d'une qualité de pose suffisante et la mise en place d'une numérisation avec robot est difficile. Dans ce chapitre nous avons proposé une stratégie de génération de trajectoire répondant à cette problématique de qualité et de praticité. La définition de cette stratégie a permis de faire ressortir des besoins qui sont alors traités comme des objectifs dans la mise en place de notre cellule de mesure robotisée. Les différents objectifs à atteindre sont décrits sur la figure 17 où l'on retrouve : la modalisation des éléments de la cellule de numérisation, l'étude des performances des éléments, l'intégration des éléments et la mise en place de la numérisation.

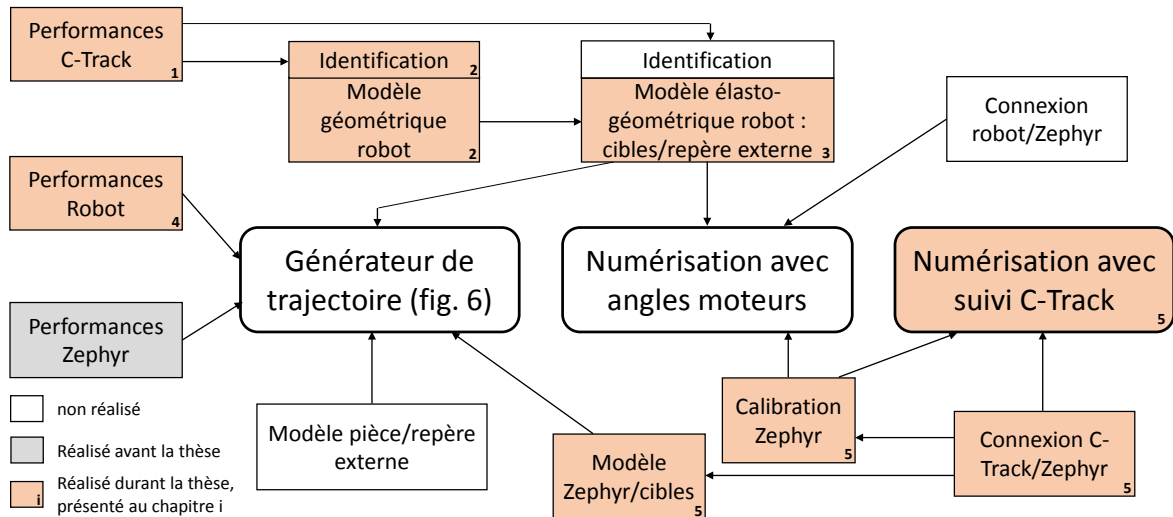


Figure 17 : Descriptions des objectifs requis dans la mise en place de la numérisation robotisée

Nous avons vu dans ce chapitre, comment sont modélisés les robots sériels dans la littérature, afin de nous guider dans la mise en place d'un modèle de robot approprié pour notre application. En particulier nous avons vu les différents défauts à prendre en compte en fonction du contexte, le nombre de paramètres à utiliser en fonction du besoin (géométriques, élastiques, jeux), les méthodes de sélection d'un modèle et les méthodes d'identification qui vont avec. Les chapitres 2 et 3 sur la modélisation géométrique et élasto-géométrique utilisent donc les modèles et méthodes présentés, ou s'en inspirent pour générer des méthodes et modèles originaux adaptés à nos besoins.

De même nous avons passé en revue les performances d'un robot et la façon de les représenter au travers d'indices de performance. Nous avons présenté les indices les plus pertinents pour notre application, nous renseignant notamment sur la qualité ou la vitesse de numérisation. Nous utilisons ces indices dans le chapitre 4, afin de qualifier le comportement de notre robot.

Puis nous avons qualifié et étudié les performances du capteur laser et du système de suivi externe qui se traduisent respectivement par : du bruit sur le nuage de points, de l'incertitude sur le suivi d'une cible. Les performances du C-Track sont utilisées tout au long de la thèse afin de maîtriser les mesures faites avec ce dernier lors des multiples calibrations et identifications.

Cette revue de littérature nous permet de répondre en partie aux objectifs présentés dans la figure 17, et en partie suscités par les besoins du générateur de trajectoire. Les chapitres suivants reprennent alors ce que nous avons vu dans ce chapitre afin d'appliquer des méthodes existantes, mais présentent également de nouvelles approches que nous avons dus créer. En particulier le chapitre qui suit détaille le choix d'un modèle géométrique pour notre application de numérisation, sa mise en place et son identification. Le tout en respectant nos besoins de qualité et de vitesse, mais aussi les contraintes liées à un contexte industriel qui sont la praticité, l'adaptabilité, et une mise en place facile et rapide.

Chapitre 2

Identification géométrique générique basée sur le mouvement

Ce deuxième chapitre porte sur la modélisation et l'identification géométrique de l'ensemble de la chaîne de mise en position du capteur de numérisation, à savoir : le positionnement du robot dans un repère externe (repère C-Track), la structure géométrique du robot sériel, et enfin le positionnement du capteur laser sur le robot. Afin de satisfaire nos contraintes énoncées dans le chapitre 1, liées à la rapidité, la praticité et la qualité, nous proposons alors une nouvelle méthode d'identification et son application pour identifier les différents paramètres du modèle géométrique.

Afin de concrétiser ces objectifs, une sélection de paramètres adéquats pour obtenir un modèle complet et non redondant nous amène tout d'abord à définir une modélisation géométrique pour le robot. Une nouvelle méthode d'identification des paramètres du modèle basée sur les mouvements du robot, que nous appelons ANalyse Continue d'Arcs Réduits (ANCAR) est alors proposée. Pour résoudre des problématiques de singularités, de robustesse et de compensation des distorsions du moyen de mesure, cette méthode est alors optimisée à plusieurs niveaux. Par ailleurs, afin de rendre cette méthode plus générique et « ouverte » à des résolutions plus complexes (intégrant redondance et hétérogénéité des entrées), elle est formalisée à l'aide d'une jacobienne permettant une résolution par minimisation. Le chapitre conclut sur une étude comparative montrant les avantages de notre méthode en fiabilité et en qualité relativement à une méthode classique largement utilisée par d'autres auteurs.

2.1 Sélection et distinction des paramètres utiles

Au sein de la cellule de numérisation, la modélisation du robot s'avère essentielle à trois niveaux : le calcul d'indices de performance, le suivi du robot en cas de perte du tracking, et la génération de trajectoires. La qualité de modélisation requise par ces trois facteurs n'est évidemment pas du même ordre. Le modèle géométrique sans prise en compte de défauts élastiques pourra certainement suffire au calcul d'indices de performance tandis qu'un modèle plus poussé sera nécessaire pour le suivi de l'effecteur. À ce titre, nous voyons justement au début du troisième chapitre, que le niveau de détails de cette modélisation géométrique ne permet de répondre qu'aux besoins de deux facteurs sur les trois. Cependant même si le modèle géométrique n'est pas suffisant, il est la base d'un modèle élasto-géométrique plus détaillé que nous traiterons dans le chapitre 3. Dans cette section nous allons donc sélectionner des paramètres pour établir un modèle géométrique de la chaîne d'action de notre cellule de numérisation. Nous modéliserons le positionnement du robot dans la cellule, le robot lui-même ainsi que le positionnement du capteur laser sur le robot.

Comme nous venons de le préciser, la modélisation va s'articuler autour des trois éléments suivants : le positionnement du robot par rapport au C-Track (système de mesure externe), le modèle du robot lui-même, le positionnement de l'outil (capteur laser) sur le robot. Dans notre cas, c'est-à-dire pour une opération de numérisation, la partie de l'outil que l'on cherche à positionner et à commander est la fenêtre de mesure du capteur. Ainsi lorsque nous parlons de positionnement de l'outil sur le robot, cela correspond plus particulièrement au positionnement de cette fenêtre laser sur le robot. Si l'on prend une

notation homogène pour décrire ces trois éléments, il est possible de décomposer le modèle complet de la manière suivante :

$$T_{\text{base,outil}} = T_{\text{base,robot}} T_{\text{robot}} T_{\text{robot,outil}} \quad \text{Eq. 41}$$

Lors de l'identification, nous comptons utiliser une méthode sans contact avec le C-Track, qui donne la pose de l'effecteur ou de l'outil grâce à un jeu de cibles réfléchissantes. Le C-Track donne donc la pose du jeu de cibles, mais le positionnement de la fenêtre laser est inconnu. Il faut donc une étape de calibration supplémentaire pour obtenir le positionnement de la fenêtre laser par rapport au jeu de cibles. Cette calibration est faite au chapitre 5 lors de la calibration du capteur laser Zephyr, où le positionnement de la fenêtre laser par rapport aux cibles est noté $T_{\text{cibles,outil}}$. La notation alors utilisée pour ce chapitre de modélisation et d'identification est réduite à :

$$T_{\text{base,cibles}} = T_{\text{base,robot}} T_{\text{robot}} T_{\text{robot,cibles}} \quad \text{Eq. 42}$$

Les cibles peuvent être disposées sur l'effecteur du robot, dans ce cas il faudra refaire la calibration de la matrice $T_{\text{cibles,outil}}$ pour tout changement dans le montage du capteur (outil). Si les cibles sont disposées sur le capteur laser, ce sera la matrice $T_{\text{robot,cibles}}$ qui devra être réévaluée en cas de changement. Dans le chapitre 5, nous traitons de la calibration du capteur laser, et celle-ci doit être faite avec un repère de cibles attachées au capteur. Cela permet entre autre, de calibrer le capteur à part, sur un banc dédié exploitant une MMT. C'est pourquoi dans notre application finale nous fixons les cibles sur le Zephyr, ce qui n'empêche pas des étapes intermédiaires avec des cibles sur le robot. La phase de calibration du Zephyr faite à part donne la matrice $T_{\text{cibles,outil}}$ qui est alors propre à l'ensemble {Zephyr, cibles}. Ensuite le capteur est monté sur le robot pour soit identifier seulement $T_{\text{robot,cibles}}$ par une calibration simple (si le robot a déjà été calibré), soit effectuer la calibration complète de $T_{\text{base,cibles}}$.

Dans ce chapitre nous calculons donc $T_{\text{base,cibles}}$ à partir d'une modélisation géométrique du robot et d'une identification. Le robot Kuka KR150-L130 à modéliser est un robot sériel 6 axes. Comme souligné dans la section 1.3.2 la modélisation DH présente de nombreux défauts : les singularités posent problème lors de l'identification, les défauts d'alignement des axes sont mal représentés, la notation est lourde et peu représentative pour les longues chaînes. Nous lui préférons donc la version modifiée par [Khalil et Kleinfinger 1986] qui utilise le même nombre de paramètres, mais une convention différente (figure 18 et équation 43). Cette approche permet de lever des ambiguïtés de notation présentes dans la modélisation DH, et mène à des calculs plus simples pour des algorithmes numériques.

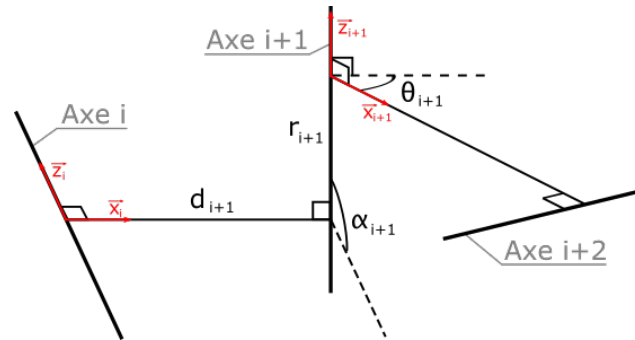


Figure 18 : Modélisation DH modifiée

$$T_{d_{i,i+1}} = \text{Trans}(\mathbf{X}_i, d_{i+1}) \text{Rot}(\mathbf{X}_i, \alpha_{i+1}) \text{Trans}(\mathbf{Z}_{i+1}, r_{i+1}) \text{Rot}(\mathbf{Z}_{i+1}, \theta_{i+1} + \delta\theta_{i+1}) \quad \text{Eq. 43}$$

Cette formulation peut ensuite être modifiée avec le paramètre de Hayati (comme dans l'équation 2) afin de mieux prendre en compte les défauts d'alignement entre deux axes consécutifs parallèles :

$$T_{h_{i,i+1}} = \text{Trans}(\mathbf{X}_i, d_{i+1}) \text{Rot}(\mathbf{X}_i, \alpha_{i+1}) \text{Rot}(\mathbf{Y}_i, \beta_{i+1}) \text{Rot}(\mathbf{Z}_{i+1}, \theta_{i+1} + \delta\theta_{i+1}) \quad \text{Eq. 44}$$

On utilise ensuite ce formalisme pour modéliser les trois éléments de l'équation 42 : le positionnement du robot dans le repère de base, le robot avec ses 6 axes, et le positionnement des cibles. La figure 19 et les équations 45 à 47 présentent cette modélisation, dans laquelle $T_{\text{base,robot}}$ contient 6 paramètres, T_{robot} 18 paramètres et 6 variables, et $T_{\text{robot,cibles}}$ 6 paramètres pour un total de 30 paramètres et 6 variables. Les variables désignent les coordonnées articulaires. On détermine ainsi totalement la position et l'orientation de l'ensemble des cibles par rapport au repère de base du C-Track.

$$T_{\text{base,robot}} = \text{Trans}(\mathbf{Z}_{\text{CT}}, r_{\text{base}}) \text{Rot}(\mathbf{Z}_{\text{CT}}, \delta\theta_{\text{base}}) T_{d_{\text{base},1}}(d_1, \alpha_1, r_1, \delta\theta_1) \quad \text{Eq. 45}$$

$$T_{\text{robot}} = T_{d_{\text{base},1}}(\theta_1) T_{d_{1,2}} T_{\text{Th}_{2,3}} T_{d_{3,4}} T_{d_{4,5}} T_{d_{5,6}}(d_6, \alpha_6, \theta_6) \quad \text{Eq. 46}$$

$$T_{\text{robot,cibles}} = \begin{cases} T_{d_{5,6}}(r_6, \delta\theta_6) T_{d_{6,c}} & \text{si } \mathbf{Z}_c \parallel \mathbf{Z}_6 \\ T_{d_{5,6}}(r_6, \delta\theta_6) T_{\text{Th}_{6,c}} & \text{si } \mathbf{Z}_c \perp \mathbf{Z}_6 \end{cases} \quad \text{Eq. 47}$$

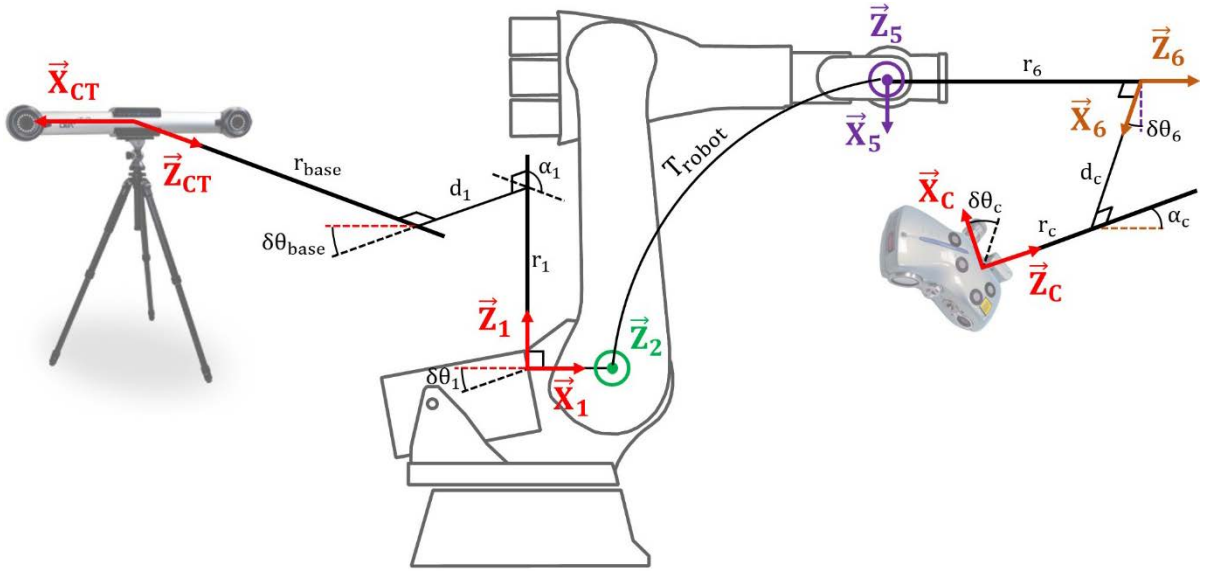


Figure 19 : Modélisation géométrique complète du robot

Les axes \mathbf{Z}_c et \mathbf{X}_c correspondent au repère associé à l'ensemble des cibles utilisées pour la calibration, c'est-à-dire ici aux cibles disposées sur le capteur. On peut remarquer que les 6 paramètres de $T_{\text{robot,cibles}}$, et plus particulièrement r_6 et $\delta\theta_6$, dépendent du positionnement de ces axes. C'est pourquoi ces deux paramètres ne font pas parti de T_{robot} , ce ne sont pas des constantes propres au robot. Pour obtenir tous les paramètres géométriques jusqu'au capteur, l'identification doit donc être faite avec le capteur sur le robot. Si pour une raison, l'identification ne peut pas être effectuée avec le capteur sur le robot, d'autres cibles doivent être disposées sur l'effecteur pour l'identification des paramètres. Les 6 derniers paramètres ne seront alors que temporaires en attendant de pouvoir monter le capteur. Puis lorsque le capteur pourra enfin être monté, la matrice $T_{\text{robot,cibles}}$ devra être identifiée pour obtenir les derniers paramètres.

La matrice T_{robot} est détaillée dans l'équation 48, pour faire apparaître tous les paramètres. On y retrouve le paramètre de Hayati dans la matrice $\text{Th}_{2,3}$, car dans la structure du robot, les axes 2 et 3 sont parallèles.

$$T_{\text{robot}} = T_{d_{\text{base},1}}(\theta_1) T_{d_{1,2}}(d_2, \alpha_2, r_2, \theta_2, \delta\theta_2) \text{Th}_{2,3}(d_3, \alpha_3, \beta_3, \theta_3, \delta\theta_3) \\ \times T_{d_{3,4}}(d_4, \alpha_4, r_4, \theta_4, \delta\theta_4) T_{d_{4,5}}(d_5, \alpha_5, r_5, \theta_5, \delta\theta_5) T_{d_{5,6}}(d_6, \alpha_6, \theta_6) \quad \text{Eq. 48}$$

Ainsi avec cette formulation, les paramètres de la matrice T_{robot} sont propres au robot et ne changeront pas quelle que soit l'utilisation de ce dernier et la configuration de la cellule de mesure. Les paramètres $T_{\text{base,robot}}$ eux seront amenés à changer en fonction de la position du C-Track ou du repère de base utilisé.

Les paramètres de $T_{robot,cibles}$ changent en fonction de l'outil utilisé, et du placement des cibles sur l'outil. Cette modélisation DH modifiée définit le positionnement des articulations 1 à 6 du robot, des segments 1 à 6, ainsi qu'un repérage présenté sur la figure 20.

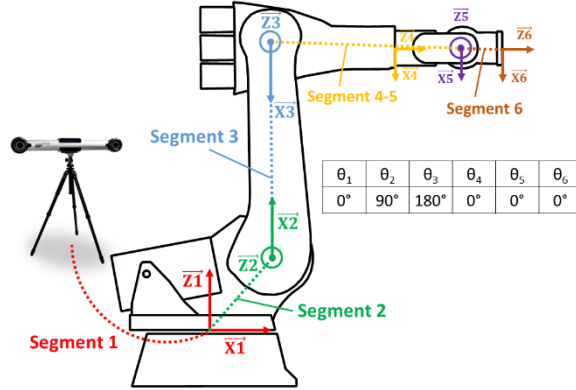


Figure 20 : Repérage et segments définis par la modélisation DH modifiée

Il reste à présent à montrer que notre modèle est bien complet, c'est-à-dire que le comportement géométrique du robot est bien représenté sans qu'aucun paramètre ne soit redondant. Sans un modèle complet l'étape d'identification ne peut pas être menée correctement. Avec cette modélisation il faudra donc identifier 30 paramètres dont 18 paramètres propres à la géométrie du robot. Ce résultat coïncide avec la formule d'Everett (équation 18), qui prescrit un minimum de 30 paramètres pour espérer avoir un modèle complet sur un robot à 6 axes de rotation.

Malheureusement la formule d'Everett ne permet pas de prouver qu'un modèle est complet, car ces 30 paramètres pourraient ne pas représenter la cinématique du robot. Pour vérifier la validité de ce type de modèle, Wu s'est basée sur les travaux de Zhuang pour retrouver un modèle complet et irréductible en partant d'une modélisation volontairement trop paramétrée avec des redondances, soit 6 paramètres pour passer d'une axe à l'autre [Wu 2015, Zhuang 1989]. Après une réduction du nombre de paramètres par élimination logique des redondances, elle retombe sur 18 paramètres nécessaires propres au robot, qui correspondent exactement aux paramètres propres au robot que nous avons établis dans notre modélisation. La description par Wu d'un modèle irréductible et complet d'un robot 6 axes est donnée dans l'équation 49. Les 18 paramètres y sont présentés sous la forme simple de translations T_x et rotations R_x suivant des axes appartenant aux liaisons du robot L_i .

$$T_{robot} = R_z(q_1) [T_x R_x T_z]_{L1} R_z(q_2 + \delta q_2) [T_x R_x T_y]_{L2} R_z(q_3 + \delta q_3) [T_x R_x T_z]_{L3} \times R_z(q_4 + \delta q_4) [T_x R_x T_z]_{L4} R_z(q_5 + \delta q_5) [T_x R_x]_{L5} R_z(q_6) \quad \text{Eq. 49}$$

Les 6 paramètres de positionnement du robot sont nécessaires dans un repère quelconque, de même que les 6 paramètres de positionnement du capteur laser. Au final notre modèle est complet et irréductible. Les 18 paramètres théoriques du robot sont présentés dans le tableau 8. Chez beaucoup d'auteurs, le nombre total de 30 paramètres semble faire consensus, car même si certains ajoutent des paramètres pour pouvoir par exemple bien situer les centres des liaisons, ils fixent ensuite ces paramètres redondants lors de la phase d'identification [Goswami93]. Comme nous l'avons déjà précisé, deux paramètres de $Td_{5,6}$ ne sont pas présents dans ce tableau car ils ne sont pas propres au robot et dépendent de Z_c .

	$Td_{1,2}$	$Th_{2,3}$	$Td_{3,4}$	$Td_{4,5}$	$Td_{5,6}$
α_i	90	0	-90	90	-90
d_i	350	1250	55	0	0
$\delta\theta_i$	0	0	0	0	x
r_i	0	$\beta_i = 0$	1500	0	x

Tableau 8 : Les 18 paramètres géométriques théoriques propres au robot (en mm et °)

Il est à noter qu'habituellement 2 paramètres non identifiables sont retirés lorsqu'un axe est aligné avec un segment, ce qui est le cas ici pour l'axe 4 et le segment entre les axes 3 et 4 [Pashkevich 2001]. Mais le formalisme DH modifié simplifiant déjà le modèle, ce problème n'apparaît pas.

Nous avons la formulation d'un modèle géométrique complet pour exprimer le positionnement du capteur laser dans un référentiel externe. Il faut à présent identifier les paramètres de ce modèle, en utilisant une méthode d'identification répondant aux besoins et contraintes de notre cellule. L'opération de numérisation et la possibilité d'une application en contexte industriel requièrent : un encombrement réduit et une grande praticité de mise en place de la méthode, une méthode rapide, des résultats de qualité suffisante pour notre application. Nous proposons donc une méthode appelée ANalyse Continue d'Arcs Réduits (ANCAR) à la fois pratique et rapide, que nous optimisons pour atteindre la qualité souhaitée sur les paramètres identifiés. Afin de pouvoir confronter les résultats de notre méthode avec d'autres résultats, nous appliquons également une méthode classique pour identifier les paramètres géométriques. Cette méthode classique est développée dans la section suivante. Les performances des différentes méthodes d'identification sont présentées en fin de chapitre pour conclure sur la pertinence de la méthode que nous proposons.

2.2 Identification des paramètres par une méthode classique

2.2.1 Choix d'une méthode classique

Afin de connaître l'efficacité de la méthode que nous allons proposer, il faut appliquer également une méthode classique d'identification des paramètres et dresser une comparaison. La méthode classique choisie est une méthode en boucle ouverte très employée [Khalil et Besnard 2002, Olabi 2011, Yin et Gao 2012]. Son concept est très simple et, dans notre cas, elle est d'autant plus facile à mettre en œuvre qu'elle requiert un système de mesure externe dont nous disposons déjà (le C-Track). L'étape d'identification des paramètres utilise simplement l'équation 24 avec inversion pour obtenir directement les paramètres, et l'étape de mesure associée consiste donc en différentes mises en position de l'effecteur pour calculer la différence entre pose mesurée et théorique. Le modèle à identifier est celui présenté en première section du chapitre (équations 45 à 47), à savoir le modèle propre du robot dont les paramètres identifiés seront présentés, ainsi que les positionnements de l'outil sur le robot et du robot dans un repère externe.

Pour cette méthode classique, il faut mettre le robot dans différentes poses, qui doivent être choisies. Nous avons vu qu'il est possible d'optimiser ce choix en travaillant notamment sur l'observabilité des paramètres et sur le conditionnement de la matrice jacobienne. Cependant cette optimisation est complexe et longue à réaliser et nous souhaitons seulement effectuer une identification permettant une comparaison avec notre méthode ANCAR (ANalyse Continue d'Arcs Réduits). Seule une étude d'identifiabilité sera faite afin de s'assurer que les configurations choisies pour la mesure permettent l'identification des paramètres. Il est à noter que cette étude est absente de la méthode d'identification que nous proposons, car la notion d'identifiabilité est liée à la jacobienne du modèle, qui n'est pas utilisée dans la méthode ANCAR.

Dans cette méthode classique comme dans beaucoup de méthodes, la jacobienne du modèle est nécessaire pour connaître l'identifiabilité des paramètres ainsi que pour l'étape d'identification des paramètres. La jacobienne du modèle géométrique complet doit donc être calculée. Comme il a été dit lors de la présentation de la jacobienne, dériver le modèle géométrique pour obtenir la jacobienne est très fastidieux et non automatisable ou programmable, mais il est possible de la calculer simplement colonne par colonne [Khalil et Besnard 2002], cette méthode est présentée ici.

2.2.2 Jacobienne et résolution par la méthode classique

La matrice jacobienne servait initialement à définir une relation entre des vitesses comme dans l'équation 6. Chaque valeur de la matrice jacobienne permet d'isoler la contribution de la vitesse d'un paramètre w_i sur la vitesse de P (pose de l'outil). Une colonne i de la jacobienne correspond donc à la translation et à la rotation de l'outil dues à une variation du paramètre w_i . Il faut alors calculer l'effet de la vitesse de ce paramètre sur les vitesses de l'outil à partir de lois de vitesse de la cinématique classique pour aboutir à la colonne de la jacobienne J_{w_i} . Soit $L_{i,outil}$ le vecteur allant de l'axe portant le paramètre w_i à la pose P de l'outil. On obtient la vitesse de translation sur l'outil étant donné une vitesse de translation ou de rotation correspondant à un paramètre i , à partir de la formule de Varignon :

$$V_{P,i} = V_{w_i} + \Omega_{w_i} \wedge L_{i,outil} \quad \text{Eq. 50}$$

Pour la vitesse de rotation de l'outil, seuls les paramètres angulaires y contribuent, sans modification à apporter :

$$\Omega_{P,i} = \Omega_{w_i} \quad \text{Eq. 51}$$

Pour plus de praticité, la recherche des effets des paramètres se sépare en deux parties : une recherche sur les paramètres de distance (translation) et une recherche sur les paramètres angulaire (rotation).

■ Les paramètres de distance sont associés à une translation dans la relation cinématique de l'équation 6, ils ont une direction et une vitesse. Soit w_i un paramètre de translation, associé à un vecteur Z_{w_i} qui est la direction du paramètre. On peut exprimer sa vitesse par la relation suivante :

$$V_{w_i} = Z_{w_i} \frac{dw_i}{dt} \quad \text{Eq. 52}$$

Le paramètre w_i étant un paramètre de distance (translation), la composante Ω_{w_i} de l'équation 50 est nulle. La vitesse sur l'outil induite par la vitesse de w_i est alors (translation et rotation) :

$$\frac{dP_i}{dt} = \left[Z_{w_i} \frac{dw_i}{dt} \quad 0_{\times 3} \right]^t \quad \text{Eq. 53}$$

Ainsi pour un paramètre de distance, la colonne de J_w correspondante vaut :

$$J_{w_i} = [Z_{w_i} \quad 0_{\times 3}]^t \quad \text{Eq. 54}$$

■ Les paramètres angulaires sont associés à une rotation dans la relation cinématique de l'équation 6, ils ont un axe et une vitesse. Soit w_i un paramètre de rotation, associé à un axe de rotation Z_{w_i} . On peut exprimer sa vitesse par la relation suivante :

$$\Omega_{w_i} = Z_{w_i} \frac{dw_i}{dt} \quad \text{Eq. 55}$$

Le paramètre w_i étant un paramètre angulaire (rotation), la composante V_{w_i} de l'équation 50 est nulle. La vitesse sur l'outil induite par la vitesse de w_i est alors (translation et rotation) :

$$\frac{dP_i}{dt} = \left[(Z_{w_i} \wedge L_{i,outil}) \frac{dw_i}{dt} \quad Z_{w_i} \frac{dw_i}{dt} \right]^t \quad \text{Eq. 56}$$

Pour un paramètre angulaire, la colonne de J_w correspondante s'écrit alors :

$$J_{w_i} = [Z_{w_i} \wedge L_{i,outil} \quad Z_{w_i}]^t \quad \text{Eq. 57}$$

Les colonnes ainsi calculées pour chaque paramètre du modèle peuvent être concaténées pour donner la matrice jacobienne. Pour un modèle de g paramètres on obtient :

$$J_w = [J_{w1} \quad \dots \quad J_{wg}] \quad \text{Eq. 58}$$

La jacobienne ainsi obtenue peut être utilisée pour déterminer l'identifiabilité et identifier les paramètres. Cependant les parties supérieure et inférieure de la jacobienne n'ont pas les mêmes unités. Pour être exploitable, la jacobienne doit donc être adimensionnée. On divise alors la partie supérieure de la jacobienne par la longueur caractéristique (obtenue par minimisation du Condition Number) déjà présentée dans l'équation 22.

Avant de déterminer l'identifiabilité des paramètres, il est nécessaire d'avoir la matrice jacobienne complète. Le protocole d'identification utilise plusieurs mises en position de l'effecteur pour déterminer les paramètres, il y a donc plusieurs configurations du robot lors de la mesure et autant de matrices jacobiennes différentes J_w . De même, chaque mesure dans une configuration nous donne un vecteur d'entrée différent dP (équation 24) contenant les erreurs sur la pose de l'effecteur. Il y a donc autant de vecteurs d'entrée que de configurations, qui sont tous regroupés en un vecteur d'entrée total dP_{tot} . Pour regrouper les différentes matrices jacobiennes, on utilise la matrice jacobienne totale J_{tot_w} . Pour m mesures (dans m configurations différentes) on a :

$$J_{tot_w} = \begin{bmatrix} J_{1_w} \\ \vdots \\ J_{m_w} \end{bmatrix} \quad \text{Eq. 59}$$

Durant les mesures, la position et l'orientation de l'outil seront relevées, la taille du vecteur dP_{tot} est donc égale à 6 fois le nombre de configurations mesurées. Afin de pouvoir identifier les paramètres il faut que ce vecteur d'entrée dP_{tot} soit au moins aussi grand que d_w . Pour obtenir les 30 paramètres géométriques de notre modèle il faut donc au moins 5 configurations. Avec 5 configurations les paramètres sont généralement tous identifiables à moins de choisir certaines configurations singulières. Pour avoir une meilleure robustesse sur l'identification, il est cependant préférable d'utiliser bien plus de configurations que ce minimum, c'est pourquoi nous utilisons 30 configurations, ce qui donne une jacobienne de dimensions 180x30.

2.2.3 Résultats sur l'identification des paramètres géométriques par la méthode classique

Après avoir sélectionné les 30 configurations de manière répartie dans le champ de vision du C-Track, la matrice J_{tot_w} est construite et la longueur caractéristique est calculée (948 mm dans notre cas). Le calcul du rang de la matrice jacobienne totale (rang = 30) montre que tous les paramètres sont identifiables. On peut donc passer à l'étape de mesure puis à l'identification des paramètres. Le modèle complet du robot, avec 30 paramètres, donne la pose de l'outil dans le repère C-Track. Le vecteur dP_{tot} est donc la différence entre cette pose mesurée et cette pose calculée pour chaque configuration. En suivant l'équation 24, le vecteur dP_{tot} et la matrice J_{tot_w} permettent d'obtenir les corrections à apporter sur les 30 paramètres du modèle géométrique que nous avons établi (propres au robot, positionnements du robot et du capteur).

Nous vérifions la fiabilité des résultats (paramètres identifiés) en répétant l'identification et avec 3 jeux de cibles disposés différemment sur l'effecteur (nous ferons de même pour la méthode ANCAR). Cinq identifications ont été réalisées avec ces trois jeux de cibles, afin d'identifier à chaque fois les paramètres du modèle géométrique. La figure 21 présente les écarts identifiés par rapport aux paramètres théoriques (mais « non exacts » et correspondant au 0 sur le graphe). On peut y voir les valeurs des paramètres pour les 5 identifications effectués (chaque identification étant représenté par un cercle) et pour les 3 jeux de cibles illustrés par les mêmes couleurs sur la figure 22.

Sur cette figure, les écarts sont visualisés, pour chaque paramètre r_i et d_i , par les distances entre les cercles suivant la direction horizontale. On trouve lors de ces identifications, des écarts entre les paramètres identifiés allant jusqu'à 2,5mm. Ces grands écarts sont présents entre des paramètres identifiés par des jeux de cibles différents. Pour une même couleur, les valeurs des 5 identifications sont proches. La méthode d'identification semble donc répétable, mais les résultats sont trop dépendants du positionnement des cibles sur l'effecteur. Cette erreur est principalement due à des distorsions dans la mesure du C-Track. Ces distorsions sont compensées dans notre méthode d'identification ANCAR, mais cette compensation ne peut pas être appliquée dans cette méthode classique. Afin de tirer des corrections à appliquer aux paramètres géométriques théoriques à partir de cette méthode, malgré l'étendue des valeurs, une moyenne a été faite sur les résultats de la figure 21. Ces paramètres sont présentés dans le tableau 9 et le tableau 10.

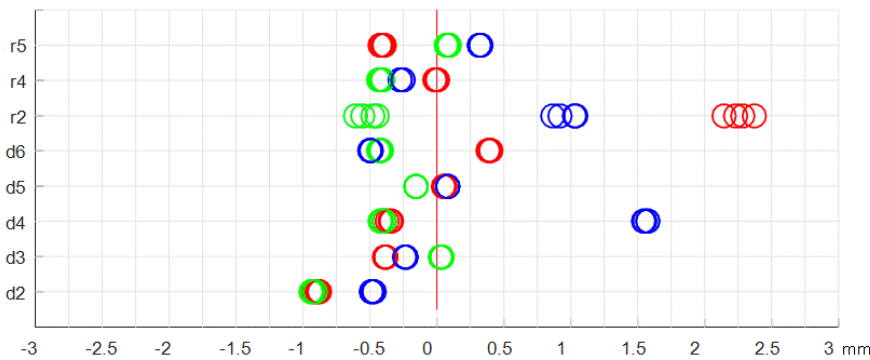


Figure 21 : Corrections pour paramètres du robot, 5 essais par méthode classique



Figure 22 : Les 3 jeux de cibles sur l'effecteur

d2	d3	d4	d5	d6	r2	r4	r5
-1,81	-0,40	0,03	-0,33	-0,50	-0,40	1,46	0.06

Tableau 9 : Corrections sur les paramètres de distance propres au robot par méthode classique (mm)

α_2	α_3	α_4	α_5	α_6	$\delta\theta_2$	$\delta\theta_3$	$\delta\theta_4$	$\delta\theta_5$	β_3
-0,67	-0,36	1,3	-0,33	1,1	-0,41	0,17	0,17	-1,7	-0,97

Tableau 10 : Corrections sur les paramètres angulaires propres au robot par méthode classique (10^{-3}°)

Ces résultats sont exploités dans l'étape de validation de la section 2.6, qui permet de vérifier si les valeurs identifiées sont correctes. Il est tout de même possible de conclure dès à présent sur la fiabilité de cette méthode qui donne des corrections pouvant varier de plusieurs millimètres en fonction de la position des cibles. En comparaison, nous verrons que les corrections ne variaient au maximum que de 0,5 mm avec notre méthode ANCAR optimisée, pour des positions des cibles et du C-Track différentes et cela, même après plusieurs mois d'écart entre les différentes campagnes de mesure.

2.3 Méthode d'identification par analyse continue d'arcs réduits

2.3.1 Adaptation de la méthode CPA à notre contexte : proposition de la méthode ANCAR

Parmi les méthodes d'identification des paramètres, il faut en sélectionner une rapide et pratique pour répondre aux contraintes d'une chaîne de production. Il faut également une bonne qualité de l'identification des paramètres. Notre choix se porte sur la méthode CPA (Circle Point Analysis) qui donne des paramètres de bonne qualité, et à laquelle nous apportons des améliorations pour gagner en rapidité et mieux répondre à nos contraintes. Pour rappel la méthode CPA présentée dans la section

1.3.7, consiste à déterminer le positionnement des 6 axes du robot pour en tirer les paramètres DH modifiés. La résolution est géométrique ce qui en fait une méthode proche de la réalité physique du robot, qui permet une identification de bonne qualité. Mais il faut instrumenter chaque axe du robot pour les mesurer un à un, et le système de mesure doit pouvoir suivre entièrement les arcs générés par chaque axe. Il faut donc que le robot ne soit pas de grande taille ou que le système de mesure ait une grande fenêtre de mesure. Cette méthode est lourde à mettre en place, peu pratique, et peut s'avérer peu rapide. Cependant à partir de cette méthode CPA, il est possible d'élaborer une variante rapide à exécuter et facile à mettre en place. Cette nouvelle méthode que nous proposons s'appelle ANalyse Continue d'Arcs Réduits (ANCAR).

Tout d'abord il faut adapter la méthode au faible volume de mesure du C-Track (3,8 m³) et aux grandes dimensions du robot. Il n'est pas possible de mesurer la rotation complète de chaque axe avec ce système de mesure, nous travaillons donc sur de petits arcs de cercle. De plus afin de rendre la méthode plus rapide et moins lourde à mettre en place, nous décidons de suivre des cibles uniquement disposées sur l'effecteur (ou sur l'outil) et non pas sur chaque axe. On peut alors identifier le positionnement de chaque axe en une seule étape sans déplacer le C-Track comme sur la figure 23. Dans ce nouveau protocole de mesure, les axes sont sollicités un à un pour mouvoir l'effecteur dans la zone visible par le C-Track. On obtient alors 6 arcs de cercle permettant de remonter à la position des 6 axes. Cette nouvelle disposition, comparée à la méthode CPA originale, rend notre méthode ANCAR peu intrusive, rapide et facile à mettre en place. De plus, tous les arcs étant dans le même repère (base du C-Track), il est plus facile d'en tirer les paramètres. En effet dans la méthode CPA classique, il faut définir un repère commun dans chacune des mesures indépendantes pour réussir ensuite à remettre tous les axes dans un même repère [Santolaria et Ginés 2013]. Ce qui rend la méthode classique d'autant plus complexe à mettre en place. Par ailleurs, si l'on ne désire pas avoir à recalculer la matrice $T_{base,robot}$ il faut que la zone visible par le C-Track pour la calibration soit également la zone de travail pour l'application de numérisation.

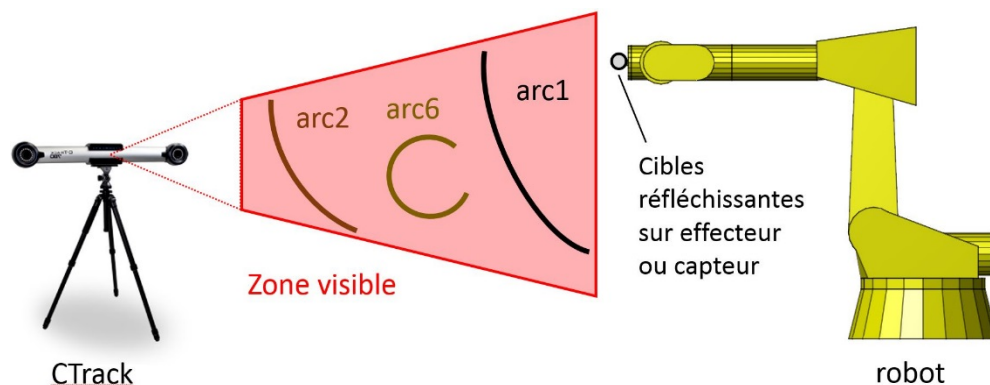


Figure 23 : Méthode d'identification par ANCAR

Dans le détail le protocole que nous proposons suit les étapes suivantes :

- Le robot doté de cibles sur l'effecteur, ou sur le capteur, est placé dans une configuration où les cibles se trouvent au centre du champ de vision du C-Track. Les coordonnées articulaires sont relevées et cette configuration est notée comme configuration de référence dans laquelle sont identifiés les paramètres. L'objectif est de connaître le positionnement des axes dans cette configuration.
- L'axe 6 est mis en mouvement seul. Si les cibles sont sur l'effecteur, un cercle complet de l'axe 6 peut être suivi par le C-Track, mais si les cibles sont sur le capteur laser, alors seul un arc de 90° peut être suivi (les cibles sont perdues pour un arc plus grand). La limitation est ici due à l'inclinaison des cibles dans le champ de vision. À la fin de cette étape, le robot revient dans la configuration de référence.

- Sans stopper la mesure, l'axe 5 est ensuite mis en mouvement seul. À l'inverse du cas précédent, si les cibles sont sur l'effecteur, un arc de 90° peut être effectué et si les cibles sont sur le capteur, l'axe 5 peut être sollicité jusqu'aux butées articulaires (arc maximum). Ici aussi la limitation est l'inclinaison des cibles. Le robot revient ensuite dans la configuration de référence.
- L'axe 4 est mis en mouvement, avec un arc de 90° . Puis le robot est remis en configuration de référence.
- L'axe 3 est mis en mouvement, tel que les cibles atteignent les limites visibles du C-Track de part et d'autre du mouvement. La limitation est à présent la taille du champ de vision. Puis le robot est remis en configuration de référence. Le même processus que pour l'axe 3 est adopté pour les axes 2 et 1.
- La mesure est alors stoppée et la pose de chaque axe peut être déterminée. Il faut d'abord séparer les arcs par filtrage de la mesure totale obtenue. Ainsi sont extraits les 6 arcs de cercle à partir du nuage total de points. Pour chacun de ces arcs, on calcule un plan des moindres carrés dont la normale donne la direction de l'axe. On projette les arcs de cercle dans ces plans pour simplifier le problème 3D en un problème 2D puis on identifie les centres de ces arcs de cercle. Pour chaque axe on obtient alors un point et une direction donnant le positionnement des axes pour la configuration de référence.
- On calcule ensuite les perpendiculaires communes aux axes ainsi identifiés, pour pouvoir estimer les différents paramètres géométriques selon la convention de DH modifiée.

L'acquisition de données est effectuée sans interruption de la mesure, mais la trajectoire est mise en pause 3 secondes entre chaque arc. Ces points d'arrêt sont facilement repérables lors du traitement des données et permettent de dissocier les arcs. Les vitesses des trajectoires pour chaque arc sont choisies pour avoir suffisamment de points de mesure. Il a ainsi été estimé que pour une bonne minimisation au sens des moindres carrés, tant pour le plan que pour le cercle, il faut au moins 2000 points (cf. annexe 2) soit une trajectoire d'une durée de 69 secondes pour un arc (fréquence du C-Track de 29 Hertz). La mesure obtenue avec le C-Track est un tableau de points, qui est ensuite traité sous Matlab pour séparer les arcs. Une fonction basée sur la minimisation aux moindres carrés du plan puis du cercle, prend en entrée un arc pour en déduire un point et une direction. Tous les points et directions sont ensuite utilisés dans un script de calcul des paramètres géométriques, en suivant la modélisation DH modifiée. Ce calcul n'est pas générique, car il est basé sur un raisonnement géométrique pour déduire les paramètres, le calcul de chaque paramètre est différent. De plus, lors de ce calcul il faut retrouver le repérage utilisé lors de la modélisation DH et notamment le sens des axes pour ne pas avoir d'erreur de signe. Le calcul des paramètres à partir du positionnement des axes doit donc être réalisé avec précaution, car les sources d'erreurs sont très nombreuses. La figure 24 résume simplement ce cheminement.

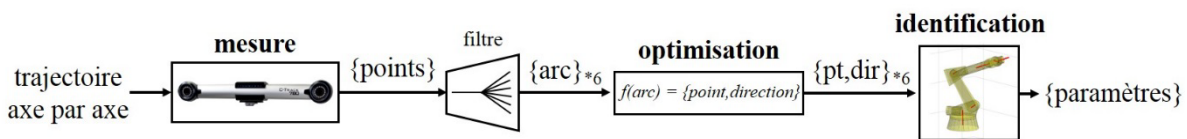


Figure 24 : Protocole ANCAR

2.3.2 La méthode ANCAR globale

Cette nouvelle méthode ANCAR, bien que pratique et rapide, est locale. Elle ne s'applique qu'à des volumes de travail limités par la taille du champ de vision du C-Track. Pour une calibration globale dans tout l'espace accessible du robot il faut déplacer le système de mesure. Nous proposons une deuxième variante de notre méthode ANCAR, qui permet justement d'identifier les axes du robot en leur faisant parcourir toute leur disponibilité articulaire, malgré un volume de mesure limité. Pour cette méthode dite globale, on suit toujours l'ensemble des cibles sur l'effecteur ou le capteur formant un repère. L'acquisition des données est cette fois plus longue, car le champ de vision du CTrack ne permet pas

d'observer le mouvement d'un axe sur tout son débattement articulaire. Il faut donc découper la course de l'axe pour pouvoir la suivre zone par zone comme sur la figure 25. Il faut stopper le mouvement du robot puis déplacer le C-Track pour pouvoir suivre la trajectoire dans la zone suivante. Une fois que toutes les mesures sont effectuées, il est possible de reconstruire l'arc de cercle complet en faisant correspondre les mesures des points extrêmes de chaque arc. Ces points sont communs d'un arc à l'autre, car le robot ne bouge pas lorsque l'on déplace le C-Track. C'est pourquoi ils sont utilisés comme référence pour regrouper les arcs. Lors de l'arrêt du robot, les cibles doivent donc être bien visibles dans les deux zones afin de pouvoir fusionner ensuite les arcs.

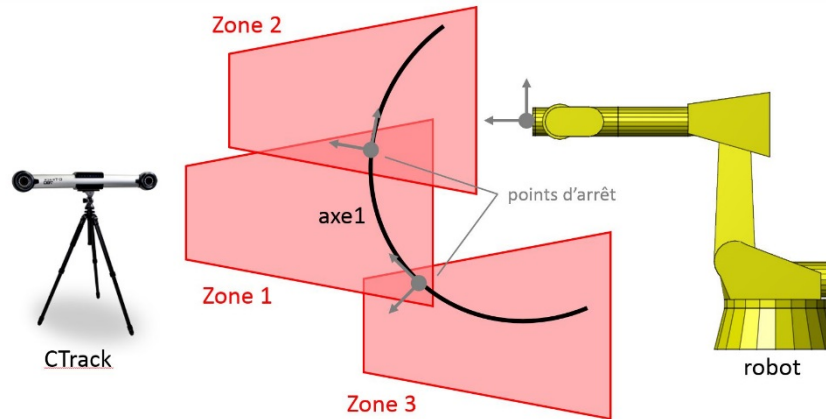


Figure 25 : reconstruction d'un grand arc à partir de plusieurs positions du C-Track

Plusieurs mesures sont nécessaires pour chaque axe. Une fois les mesures effectuées, les arcs complets sont reconstruits en utilisant les points extrêmes communs. Chaque arc complet ainsi obtenu est exprimé dans un repère différent, ce qui rend difficile l'identification à partir de ces arcs complets. On opte donc pour une identification des paramètres de proche en proche exploitant le seul repère commun à toutes ces mesures : le repère de l'ensemble des cibles. L'identification des paramètres se fait de l'effecteur vers la base en identifiant les paramètres 4 par 4 (avec seulement deux axes à la fois).

Si l'on recherche les paramètres entre les axes i et $i+1$, on suppose alors les paramètres entre $i+1$ et le repère des cibles déjà identifiés. On étudie alors l'arc complet de l'axe i . Grâce à ce dernier il est possible d'exprimer la pose de l'axe i dans le repère des cibles de l'effecteur comme sur la figure 26. En effet la rotation de ce repère étant autour de l'axe i lui-même, la pose de l'axe i est fixe dans le repère des cibles. Pour déterminer à présent la pose de l'axe $i+1$ dans ce même repère, il faut utiliser les paramètres et les coordonnées articulaires entre $i+1$ et le repère de cibles. Les paramètres sont supposés connus, et les coordonnées sont celles des axes du robot (autres que i) lors de sa rotation autour de l'axe i . Il est alors possible de reconstruire le positionnement de l'axe $i+1$ dans le repère des cibles comme sur la figure 26. Les axes $i+1$ et i sont donc exprimés dans le même repère, et les 4 paramètres de passage peuvent être identifiés. On applique cette méthodologie en partant de l'axe 6 jusqu'à la base du robot pour obtenir tous les paramètres géométriques.

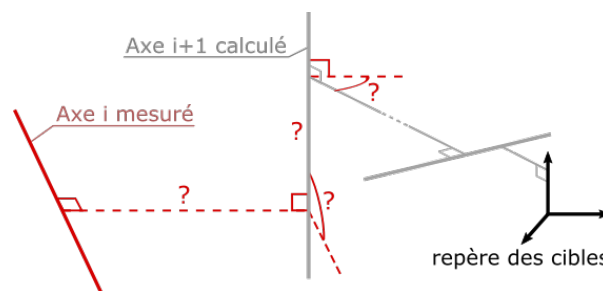


Figure 26 : Axes i et $i+1$ dans le repère des cibles pour la méthode ANCAR globale

Contrairement aux méthodes CPA classiques qui nécessitent d'instrumenter chaque segment du robot, ici seuls l'effecteur ou l'outil sont instrumentés. Cette technique est moins intrusive et peut être facilement mise en place sur n'importe quelle application, et il n'est pas utile de changer la tête du robot pour un outil spécifique à la calibration. L'avantage de cette variante de la méthode ANCAR, que nous appelons méthode ANCAR de type globale, est de pouvoir faire une calibration du robot dans tout son espace, avec une instrumentation minimale. Par opposition nous appelons notre méthode précédente, où le C-Track n'est pas déplacé, méthode ANCAR de type locale. L'avantage de la méthode locale est de pouvoir identifier les paramètres géométriques en une seule étape de mesure, simple et rapide. L'utilisation de l'une ou l'autre méthode impliquera des différences sur les paramètres identifiés. En effet la méthode locale donnera de très bons résultats dans la zone ciblée, mais les paramètres seront moins représentatifs du robot en dehors de cette zone. Quant à la méthode globale, elle mènera à des résultats moyens sur la qualité de positionnement du robot, dans tout l'espace atteignable par celui-ci. Ces résultats seront certes moins bons que ceux de la variante locale dans une zone ciblée, mais bien meilleurs partout ailleurs. Dans la suite des travaux, nous optons pour la méthode locale qui permet une bonne calibration dans un volume défini, qui sera le volume de l'application de numérisation. La méthode globale pourra être utilisée dans de futurs travaux requérant des volumes de travail plus grands (numérisation de pièces encombrantes). Le volume mesurable lors de la calibration locale donne les arcs présentés sur la Figure 27.

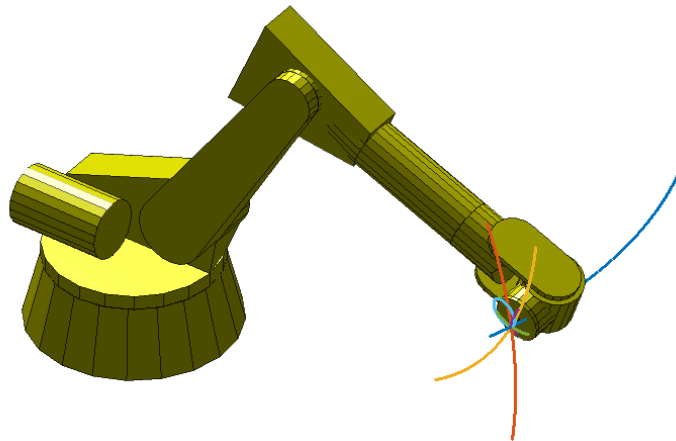


Figure 27 : Arcs mesurés lors d'une calibration ANCAR locale

2.4 Optimisation de l'identification par la méthode ANCAR

2.4.1 Problématiques sur la fiabilité des paramètres identifiés par ANCAR

Les arcs mesurés, lorsque l'on applique notre méthode CPA locale, sont courts comparés à un cercle complet. Ils rendent donc la méthode très sensible aux différentes erreurs présentes dans la procédure d'identification. Tout d'abord, lors de l'utilisation des moindres carrés, ces petits arcs rendent difficile l'estimation d'un plan puis d'un cercle, et génèrent une grande incertitude sur cette estimation. De plus, étant donné la taille des arcs, la moindre erreur lors de la mesure a des conséquences importantes lors de l'identification. Afin de mettre en évidence l'effet des différentes sources d'erreur, des essais ont été réalisés avec 3 jeux de cibles différents disposés sur l'effecteur. Les paramètres sur lesquels l'étude est menée sont les paramètres géométriques propres au robot, qui ne doivent pas varier d'un essai à l'autre si le moyen de mesure est déplacé. Tout comme pour les résultats avec la méthode d'identification classique, la figure 28 présente l'écart entre les paramètres théoriques (0 sur le graphe) et les paramètres identifiés (uniquement paramètres de longueur). On peut voir les valeurs des paramètres pour les 5 essais effectués (chaque essai étant représenté par un cercle) et pour les 3 jeux de cibles sur l'effecteur.

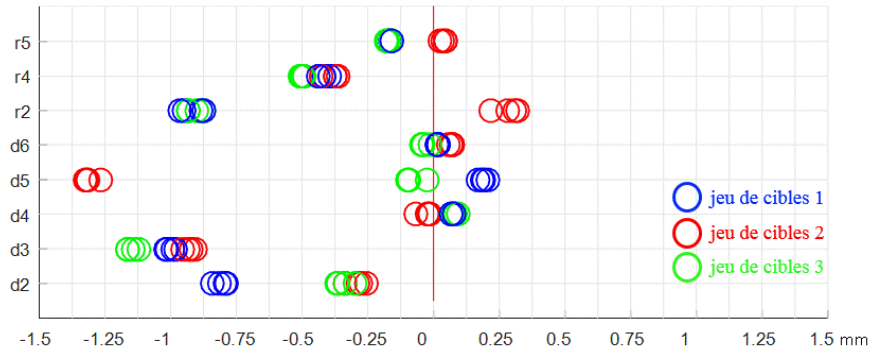


Figure 28 : Corrections pour paramètres propres au robot, ANCAR (locale) sans amélioration

Sur cette figure, les écarts sont visualisés, pour chaque paramètre r_i et d_i , par les distances entre les cercles suivant la direction horizontale. On trouve lors de cette identification, des écarts allant jusqu'à 1,5 mm sur les valeurs des paramètres identifiés. On peut difficilement se fier à des valeurs si variables, trop dépendantes du positionnement des cibles sur l'effecteur. Il faut donc minimiser l'écart entre ces valeurs, c'est-à-dire minimiser l'effet des sources d'erreur. Nous cherchons donc à rendre la méthode plus robuste et moins sensible aux bruits ou erreurs de calcul. Dans notre méthode CPA il y a 3 étapes clés pouvant être source d'erreur que l'on retrouve sur la figure 24 : l'étape de mesure, l'étape d'optimisation puis l'étape d'identification.

Nous décrivons ci-dessous les problèmes liés à chacune de ces étapes.

- Comme évoqué précédemment, l'étape d'identification n'est pas générique, car basée sur une reconstruction géométrique paramètre après paramètre. La construction des différents repères puis des paramètres à partir du positionnement des axes se fait au cas par cas. Ainsi, de nombreuses erreurs humaines, comme par exemple des erreurs de signe, peuvent se glisser dans la résolution. Il n'existe pas de méthode particulière permettant d'éliminer ces erreurs mis à part des simulations d'identification à partir de paramètres connus et de mesures simulées.

- L'étape d'optimisation est théoriquement l'une des plus grandes sources d'erreur. En effet, l'utilisation de la méthode des moindres carrés pour la minimisation génère de nombreux problèmes bien connus dans la littérature. Par exemple, la présence de points aberrants influe énormément cette méthode, tout comme un mauvais choix dans l'écriture du problème de minimisation. De plus, les arcs étant très courts, une différence minime dans la répartition des points sur l'arc peut donner des positionnements d'axe très différents. Par exemple, le bruit de mesure du C-Track a une influence de plusieurs μm sur le centre estimé de l'arc.

- Les erreurs sur les paramètres identifiés peuvent aussi être dues à l'étape de mesure, c'est-à-dire dues au robot ou au moyen de mesure. Les essais répétés desquels sont tirés les résultats de la figure 29, sont affectés par la répétabilité du robot, ainsi que par le bruit et les distorsions dans la mesure du C-Track. Sur cette figure, on peut observer que les 5 cercles de même couleur, pour chaque couleur et chaque ligne, sont proches. Les 5 essais donnent donc des résultats répétables, et les écarts entre les cercles de différentes couleurs sont donc dus à d'autres phénomènes que la répétabilité du robot et le bruit de mesure du C-Track. Les différentes couleurs correspondent à différents ensembles de cibles placées sur l'effecteur. Toutes ces cibles sont fixées sur l'effecteur, ce dernier pouvant être considéré comme un solide rigide, le mouvement des cibles est censé être identique. Malgré cela, les paramètres identifiés sont différents en fonction de l'ensemble de cibles considéré. On peut donc supposer que la mesure du C-Track dépend de la position et de l'orientation des cibles dans son champ de vision, ce que l'on peut alors considérer comme des distorsions.

2.4.2 Amélioration de l'étape d'optimisation

Une première amélioration simple consiste à supprimer les points jugés aberrants. Nous avons donc mis en place un filtre éliminant 1% des points les plus éloignés du plan des moindres carrés ou du cercle des moindres carrés. Au-delà de cette valeur, l'information de la mesure est dégradée et les valeurs identifiées sur la figure 28 s'écartent.

On sait également que la méthode des moindres carrés génère des erreurs si l'on ne prête pas attention à la formulation du problème. Classiquement, l'équation d'un plan de vecteur normal $\mathbf{n} = [a \ b \ -1]$ peut s'écrire comme dans l'équation 60, puis cette équation est utilisée pour chaque point devant appartenir au plan ce qui donne un système avec autant d'équations que de points. Ce système s'écrit sous la forme matricielle de l'équation 61 et peut alors être résolu pour trouver les paramètres a , b et d du meilleur plan passant par les points.

$$ax + by + d = z \quad \text{Eq. 60}$$

$$[a \ b \ d]^t = M^+ \mathbf{Pt}_z \quad \text{avec} \quad M = [\mathbf{Pt}_x \ \mathbf{Pt}_y \ U] \quad \text{Eq. 61}$$

Ici \mathbf{Pt}_x , \mathbf{Pt}_y et \mathbf{Pt}_z sont des vecteurs colonnes regroupant les coordonnées des points, U est un vecteur colonne unitaire et M^+ est la pseudo inverse de M . Une telle formulation pose problème si la normale du plan se trouve plutôt suivant X ou Y , donnant une grande erreur sur l'estimation de la normale. Pour éviter cette erreur, nous identifions la normale du plan des moindres carrés en plusieurs itérations. Une première normale \mathbf{n}_1 est identifiée dans le repère R_0 après identification du plan des moindres carrés, avec l'équation 61. Le processus itératif est ensuite le suivant. Lorsqu'une normale \mathbf{n}_i est obtenue, nous choisissons deux axes orthogonaux à \mathbf{n}_i pour former un repère R_i . Une matrice de passage $T_{i-1,i}$ est déduite, et tous les points sont déplacés dans ce nouveau repère R_i donnant \mathbf{Pt}_{xi} , \mathbf{Pt}_{yi} et \mathbf{Pt}_{zi} . Ces nouveaux points sont utilisés pour identifier la normale suivante \mathbf{n}_{i+1} d'un nouveau plan des moindres carrés. Au bout de k itérations jugées suffisantes, nous obtenons les paramètres a_k , b_k et d_k d'un plan des moindres carrés identifié à partir des coordonnées \mathbf{Pt}_{xk-1} , \mathbf{Pt}_{yk-1} et \mathbf{Pt}_{zk-1} dans un repère R_{k-1} . Dans notre cas, habituellement 3 itérations suffisent pour atteindre une convergence sur la normale estimée. Pour vérifier s'il y a convergence il suffit de comparer la normale \mathbf{n}_k avec l'axe $[0, 0, 1]$ du repère R_{k-1} . Les points sont ensuite projetés dans le plan identifié en suivant la normale identifiée \mathbf{n}_k , le problème devient alors 2D dans le repère R_k . La normale donne la direction de l'axe du robot, mais pour pouvoir le positionner il faut un point, c'est pourquoi on cherche ensuite le centre du cercle correspondant à l'arc. De même que précédemment il faut résoudre un problème en utilisant la méthode des moindres carrés, pour un cercle 2D. L'équation d'un cercle peut s'écrire comme dans l'équation 62. En l'appliquant à chaque point de l'arc, on obtient un système d'équations que l'on peut écrire sous la forme matricielle de l'équation 63.

$$\frac{1}{-2y_0}(x^2 + y^2) + \frac{2x_0}{-2y_0}x + x_0^2 + y_0^2 - r^2 = y \quad \text{Eq. 62}$$

$$\begin{bmatrix} 1 & 2x_0 \\ -2y_0 & -2y_0 \end{bmatrix}^t \begin{bmatrix} x_0^2 + y_0^2 - r^2 \end{bmatrix} = M^+ \mathbf{Pt}_y \quad \text{avec} \quad M = [\mathbf{Pt}_x^2 + \mathbf{Pt}_y^2 \ \mathbf{Pt}_x \ U] \quad \text{Eq. 63}$$

En réalisant des itérations comme pour le problème précédent, on obtient les paramètres $[x_0 \ y_0]$ qui sont les coordonnées du centre de l'arc de cercle positionnant l'axe. Ces coordonnées sont ramenées dans le repère 3D R_k , pour donner les coordonnées d'un point 3D. Puis la normale \mathbf{n}_k et ce point sont ramenés dans le repère R_0 en utilisant les matrices de passage $T_{i-1,i}$ calculées auparavant. Au final nous avons donc un point et une direction permettant de décrire la posture d'un axe du robot.

Après une étude théorique, il a été constaté que l'erreur sur l'estimation des cercles (par les moindres carrés) crée une erreur sur les paramètres géométriques beaucoup plus importante que celle créée par l'estimation des plans. Il peut être alors intéressant de chercher à améliorer la méthode d'estimation du centre des cercles à partir d'arcs réduits. Des pistes ont été investiguées dans ce sens, comme celle

présentée en annexe 3, mais sans succès. Le problème majeur rencontré est l'effet du bruit de mesure sur des arcs si courts, ce qui rend difficile toute amélioration de l'estimation d'un centre. Au final, en tenant compte des différentes remarques faites sur l'étape d'optimisation, les nouvelles corrections des paramètres du robot sont alors obtenues (figure 29).

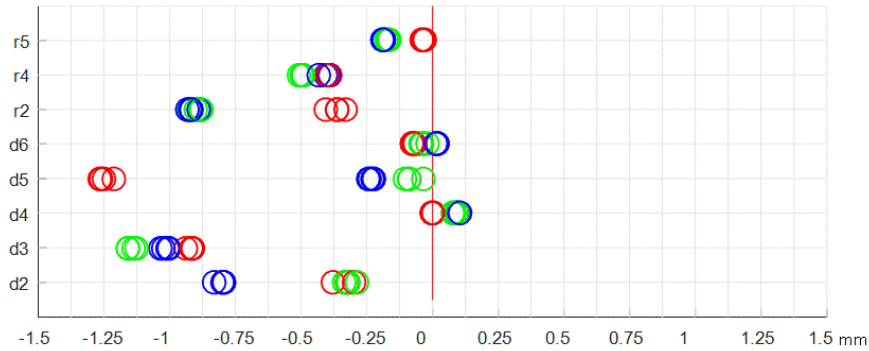


Figure 29 : Corrections pour paramètres du robot, ANCAR (locale) avec optimisation des moindres carrés

Bien que l'amélioration soit faible, elle est néanmoins visible notamment pour les paramètres $r5$, $r2$ et $d5$. De plus, elle prévient des risques encourus lors de l'utilisation des moindres carrés, notamment lorsque l'on atteint certaines configurations singulières pour l'estimation des cercles.

2.4.3 Correction des distorsions sur la mesure du C-Track

Les écarts entre les paramètres identifiés par les différents jeux de cibles sur la figure 28, nous laisse penser que la mesure du C-Track dépend de la position et de l'orientation des cibles dans son champ de vision. Nous supposons donc que la mesure du C-Track est sujette à des phénomènes de distorsion. Afin de vérifier cette hypothèse, la distance entre deux ensembles de cibles disposées sur l'effecteur a été mesurée au cours d'une trajectoire en arc de cercle sollicitant l'axe 4 du robot entre -45° et $+45^\circ$. Comme on le voit sur la figure 30, cette distance qui devrait être constante varie au cours de la trajectoire. Il y a donc une distorsion des distances lors de la mesure du C-Track, et il faut pallier ce problème.

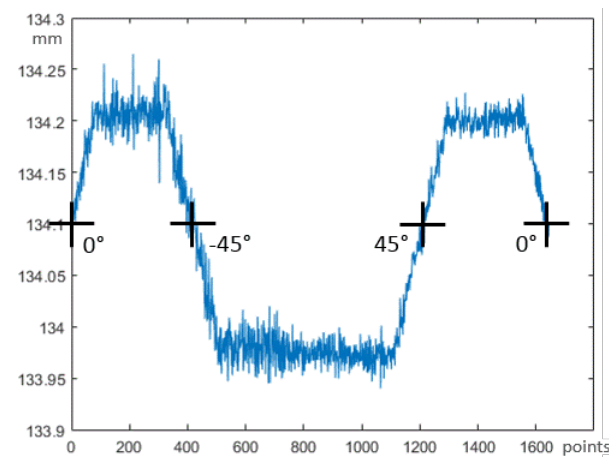


Figure 30 : Distance entre 2 ensembles de cibles lors de la rotation d'un axe

De nombreux essais ont été effectués et il apparaît que cette distorsion est complexe. Il serait long et difficile de caractériser ou cartographier la distorsion de la mesure du C-Track dans son champ de vision, car trop de facteurs entrent en compte (distance entre cibles, position de la cible dans le champ de vision, orientation des cibles, mouvement des cibles). Une solution non empirique et générique est donc plutôt recherchée.

Après cette étude et vu les ordres de grandeur des distorsions, il est évident que ce problème est majoritairement responsable de nos erreurs sur l'identification des paramètres. On ne peut maîtriser les valeurs des distorsions dans le champ de vision du C-Track, mais il faut tout de même réussir à résoudre ce problème. L'hypothèse de solide rigide sur l'effecteur peut être conservée, car elle est légitime, il faut donc travailler à annuler le mouvement relatif perçu entre les ensembles de cibles. Pour cela une configuration de référence est fixée, lorsque les cibles sont bien face au C-Track et au centre de son champ de vision. Cette configuration est supposée être sans distorsion (en réalité avec des distorsions minimales). Un modèle rigide est alors élaboré, basé sur le positionnement relatif des 3 ensembles de cibles. Ce modèle définissant un solide rigide, peut être soit basé uniquement sur la position des ensembles, soit utiliser également l'orientation respectivement comme sur la figure 31 et la figure 32.

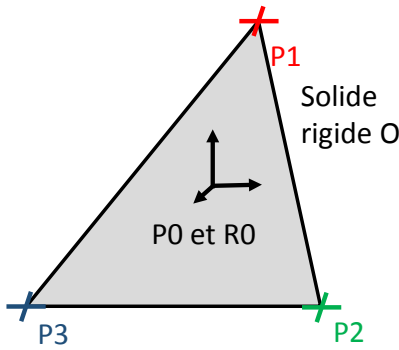


Figure 31 : Modèle rigide basé sur la position des ensembles de cibles

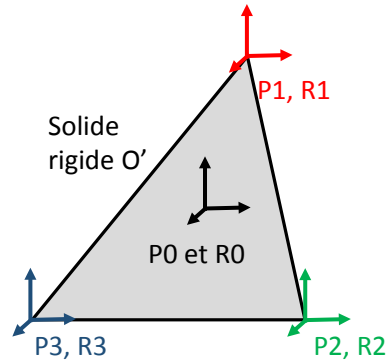


Figure 32 : Modèle rigide basé sur la pose (position et orientation) des ensembles de cibles

Lors d'un mouvement de l'effecteur suivi par le C-Track et pour chaque pas de mesure, il faut recalculer les sommets du modèle du solide O sur les positions des ensembles de cibles mesurés (par minimisation des distances sommets – mesures). Une fois le solide recalculé sur toute la trajectoire, on suppose que la véritable position des ensembles est donnée par les sommets du solide. Les données alors utilisées comme entrée pour l'identification sont les 3 sommets du solide O, et non les mesures du C-Track. Le protocole ANCAR de la figure 24 est donc modifié avec l'ajout de la figure 33.

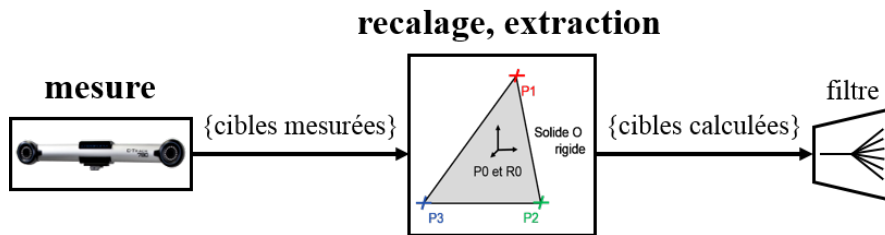


Figure 33 : Arrangement des cibles mesurées pour identification

Le deuxième modèle de solide utilise en plus l'orientation des ensembles de cibles pour définir un solide O' rigide. Le recalage de ce solide O' au cours du mouvement se fait donc avec la position et l'orientation des ensembles de cibles. Mais le C-Track est peu performant pour donner l'orientation des ensembles, et des tests montrent que cette deuxième méthode dégrade fortement les paramètres identifiés. Nous utilisons donc la première définition uniquement basée sur la position des cibles ou des ensembles (solide O). Les résultats sur l'identification des paramètres sont donnés sur la figure 34, où l'on observe une légère amélioration sur certains paramètres. Mais ce que l'on observe surtout, c'est toujours une grande différence entre les 3 ensembles de cibles recalés (les 3 couleurs) malgré l'utilisation d'un modèle rigide. Cette différence restante n'est plus due à une distorsion dans le champ de vision du C-Track puisqu'à présent les points utilisés pour ces ensembles sont les sommets de O qui est rigide.

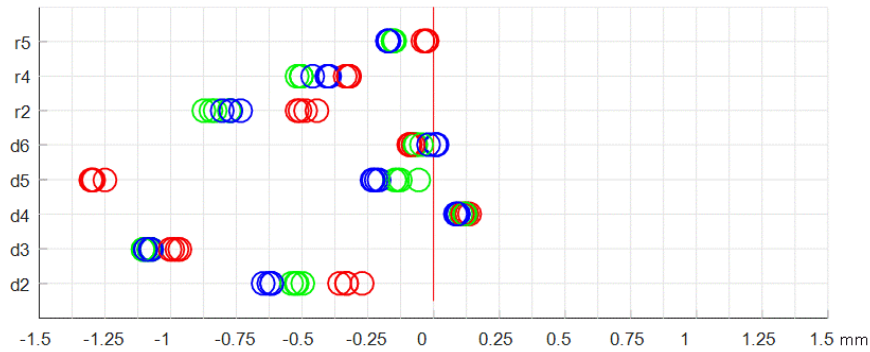


Figure 34 : Corrections pour paramètres du robot, ANCAR (locale) avec hypothèse de solide rigide

Même si les distances entre les sommets de O sont fixes, le mouvement de chaque sommet peut être différent. En effet, bien que le mouvement de l'effecteur soit une rotation dans notre méthode ANCAR, les erreurs de mesures vont créer des variations dans l'orientation de O , autres que celles dues à la rotation. Ces variations d'orientation lors de la rotation, font que les sommets de O suivent des tracés différents les uns des autres. Le mouvement doit être une rotation autour d'un axe, et tout autre mouvement est une perturbation. Ainsi, afin que les sommets de O suivent le même mouvement, nous ajoutons des contraintes sur l'orientation de O .

Tout d'abord, nous limitons les variations d'orientation autorisées autour des directions X et Y pour un mouvement de rotation autour de Z (un axe du robot). Nous fixons comme limite quelques centièmes de degrés entre deux orientations successives de O , c'est-à-dire entre deux points de mesure.

Ensuite il faut également contraindre l'orientation autour de l'axe de rotation Z . Cette orientation va beaucoup varier car elle suit le mouvement de rotation, mais il ne faut pas la laisser libre pour autant car les erreurs de mesure jouent également sur cette orientation. Ici il faut limiter les variations d'orientation autour de Z non pas pour se rapprocher d'une variation nulle comme autour de X et Y , mais pour se rapprocher de l'orientation que doit avoir O au cours de la rotation. Cette orientation que doit avoir le solide O est donnée par l'angle α de la figure 35.

Nous utilisons l'arc de cercle qui est généré par la rotation pour en déduire l'axe de la rotation. Puis pour chaque position de l'effecteur sur l'arc (chaque point de mesure), on calcule la pose du solide rigide O avec les contraintes déjà mentionnées. On peut ensuite calculer la droite d entre l'axe du mouvement de rotation et chaque position du centre du solide O . Cette droite d forme l'angle α avec la normale du solide comme sur la figure 35. Le mouvement effectué par le robot, et donc par le solide O , est une rotation autour de l'axe, l'angle α doit donc rester constant tout au long du mouvement. La dernière contrainte imposée sur le solide O est donc sur cet angle α , qui doit rester constant tout au long du mouvement, avec une tolérance de 0.01° .

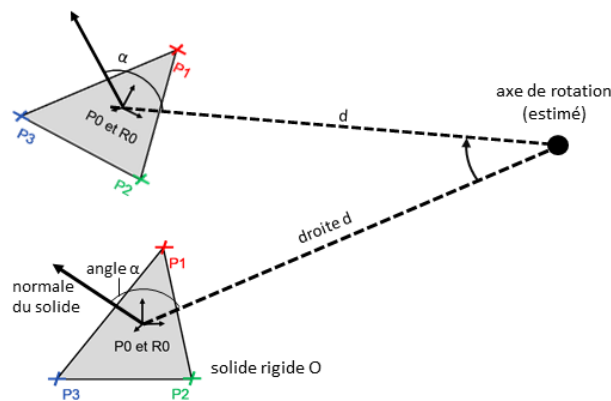


Figure 35 : Orientation du solide fixée par la rotation

Les résultats obtenus sur l'identification des paramètres propres au robot, présentés sur la figure 36, sont cette fois jugés satisfaisants en termes de fiabilité. L'incertitude sur les paramètres est de l'ordre de grandeur de la répétabilité du robot donnée par le constructeur (0,12mm). L'incertitude maximale est de l'ordre de 0,25mm sur le paramètre r2 ce qui est normal, car c'est le paramètre subissant le plus d'incertitude. En effet, dans le modèle, ce paramètre dépend de l'estimation des axes 1, 2, 3 et 4.

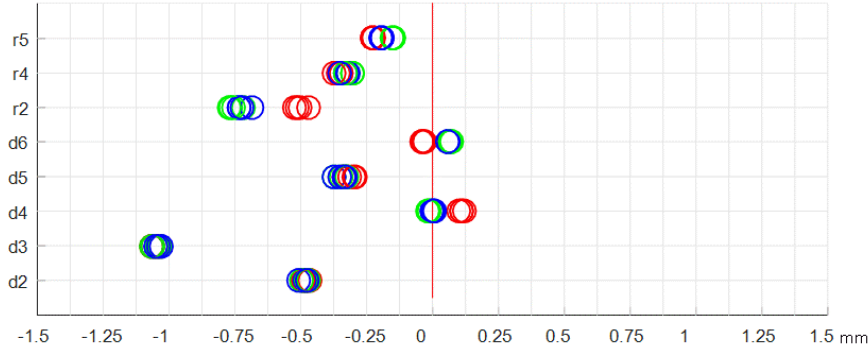


Figure 36 : Corrections pour paramètres du robot, ANCAR (locale) avec hypothèse de solide rigide et d'orientation contrainte

Dans ces travaux, nous appelons méthode ANCAR optimisée, notre méthode ANCAR utilisant ces améliorations sur la méthode des moindres carrés et les hypothèses de solide rigide et d'orientation contrainte.

2.4.4 Paramètres identifiés par la méthode ANCAR locale optimisée

Les valeurs de correction obtenues avec notre méthode ANCAR locale optimisée, prenant en compte les améliorations sur les étapes de mesure, d'optimisation et d'identification, sont présentées dans les tableaux suivants. Le tableau 11 donne les corrections sur les paramètres de distance propres au robot, en mm, et le tableau 12 donne les corrections sur les paramètres angulaires du robot, en millièmes de degrés (10^{-3}°).

d2	d3	d4	d5	d6	r2	r4	r5
-0,48	-1,05	0,03	-0,33	0,03	-0,66	-0,33	-0,19

Tableau 11 : Corrections sur les paramètres théoriques de distance propres au robot (mm)

α_2	α_3	α_4	α_5	α_6	$\delta\theta_2$	$\delta\theta_3$	$\delta\theta_4$	$\delta\theta_5$	β_3
-2,2	-40,3	38,1	-7,2	-29,7	531	-383	374	13,7	10,3

Tableau 12 : Corrections sur les paramètres théoriques angulaires propres au robot (10^{-3}°)

Nous avons donc réussi à réduire les différences entre les paramètres identifiés par des essais différents, et identifiés avec des cibles placées différemment. Afin de conforter nos résultats, une autre série d'essais a été mise en place plusieurs mois après, les résultats sont présentés sur la figure 37. Les tendances sont les mêmes, mais on note des corrections qui diffèrent d'environ 0,25mm avec les valeurs identifiées précédemment, notamment sur les paramètres r4, r2, d2 et d3. Cette légère différence peut s'expliquer par les modifications des conditions expérimentales, notamment le positionnement du C-Track dans la cellule, et les variations en luminosité et en température. Il est donc important de noter qu'une calibration régulière du robot est nécessaire pour compenser les variations des conditions de travail au cours du temps. On peut d'autre part souligner le besoin d'une modélisation plus fine du robot si l'on souhaite mieux modéliser son comportement et maîtriser en partie ces variations.

Il n'a été présenté ici que les résultats sur les paramètres propres au robot, mais les paramètres de positionnement du robot dans un repère externe et de l'outil sur l'effecteur sont également obtenus avec cette méthode. Étant donné que ces autres paramètres dépendent des positionnements très variables de

l'outil utilisé et du C-Track, ils seront de toute façon réévalués lors de la mise en place finale de l'application et auront des valeurs tout autres.

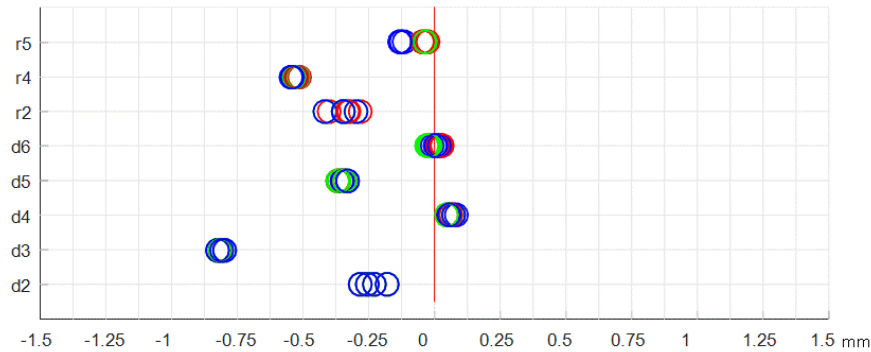


Figure 37 : Corrections pour paramètres du robot, ANCAR locale optimisée, après 3 mois

2.5 Méthode ANCAR résolue avec une jacobienne

La méthode ANCAR qui a été présentée, tout comme la méthode CPA originale, permet d'avoir une mesure et une identification proche de la géométrie concrète du robot. Cette méthode se base sur des constructions géométriques pour retrouver les paramètres du robot, ce qui la rend plus simple à appréhender. Cependant cet avantage est aussi une limitation qui la rend plus lourde à mettre en place, car ces constructions ne sont pas génériques : elles sont effectuées au cas par cas comme évoqué dans la section 2.3. D'autre part, à cause de son principe même, elle ne permet l'identification que des paramètres géométriques. C'est pourquoi de nos jours la méthode CPA est si peu utilisée pour l'identification des paramètres d'un robot [Hollerbach et Lokhorst 1995, Santolaria et al. 2013]. La complexité de sa mise en place (code au cas par cas non générique), le nombre d'erreurs pouvant s'y glisser, ainsi que le fait qu'elle ne puisse pas évoluer pour obtenir d'autres paramètres (jeux, élasticités) en font une méthode peu attractive malgré ses avantages. La principale cause de ces désavantages est le manque de formalisme et de généralité de cette méthode. Ainsi la formulation des paramètres du modèle géométrique à identifier change d'une géométrie de robot à une autre, et d'un auteur à un autre. Nous proposons donc ici une nouvelle formulation innovante pour notre méthode ANCAR, lui donnant accès aux avantages des méthodes classiques tout en conservant ses atouts.

2.5.1 Mise en place d'une modélisation adaptée à notre méthode

Les méthodes classiques d'identification, comme celle utilisée dans la section 2.2, utilisent la jacobienne J_w du modèle du robot, qui lie les paramètres du modèle aux mesures de la pose de l'effecteur. Notre méthode ANCAR, elle, se base sur une reconstruction purement géométrique des paramètres. Nous allons formaliser cette reconstruction en adaptant la modélisation géométrique du chapitre 2, afin d'en tirer une jacobienne J_A liant les mesures des axes de notre méthode ANCAR aux paramètres géométriques. La méthode ANCAR est constituée de deux étapes. Premièrement une étape de mesure qui, grâce à des mouvements d'arcs de cercle, permet d'obtenir la pose (position et orientation) des axes du robot dans le repère du moyen de mesure. La deuxième étape est l'identification des paramètres à partir de la connaissance de ces axes, par construction géométrique. Le formalisme que nous cherchons s'inscrit donc dans l'étape d'identification des paramètres du modèle géométrique. Il permettra d'obtenir une jacobienne liant une différence entre la mesure et le calcul de la pose des axes, et la correction à apporter aux paramètres géométriques.

Le raisonnement qui va suivre part donc du principe que l'étape d'identification dispose, comme donnée d'entrée, de la pose des axes du robot dans le repère du système de mesure (C-Track). Cela se traduit pour la pose d'un axe i par la connaissance de sa direction Z_{imes} et des coordonnées Pt_{imes} d'un point sur cet axe i , dans le repère du C-Track :

$$[Z_{imes} \quad Pt_{imes}]^t \quad \text{Eq. 64}$$

On connaît donc la position réelle mesurée d'un axe i , que l'on voudra comparer avec la position théorique de cet axe. Il faut calculer la direction théorique Z_{ith} et la position Pt_{ith} pour cet axe i :

$$[Z_{ith} \quad Pt_{ith}]^t \quad \text{Eq. 65}$$

Pour calculer les valeurs du vecteur de l'équation 65, nous allons tout simplement utiliser les matrices homogènes de passage de notre modèle géométrique, permettant d'exprimer le passage du repère du moyen de mesure jusqu'à l'axe considéré.

$$T_{base,i} = T_{base,1} T_{1,2} \dots T_{i-2,i-1} \text{Trans}(X_{i+1}, d_i) \text{Rot}(X_{i+1}, \alpha_i) \quad \text{Eq. 66}$$

$$T_{base,i} = \begin{bmatrix} A_i & t_i \\ 0_{\times 3} & 1 \end{bmatrix} \quad \text{avec} \quad A_i = [s_i \quad n_i \quad a_i] \quad \text{Eq. 67}$$

Comme on peut le constater dans l'équation 66, pour arriver au positionnement d'un axe il faut utiliser les matrices homogènes de notre modèle, mais aussi des transformations élémentaires. Afin d'éviter cette notation lourde, il est possible dans ce calcul d'exprimer le modèle avec la notation DH (ici non modifiée). La direction de l'axe est donnée par la direction a_i de la matrice de rotation A_i , un point de l'axe i est tout simplement t_i . Les paramètres théoriques du robot nous amènent donc, via les matrices homogènes, à un modèle théorique simple de chaque axe i :

$$[Z_{ith} \quad Pt_{ith}]^t = [a_i \quad t_i]^t \quad \text{Eq. 68}$$

Il est à présent possible d'observer l'écart entre les axes mesurés et les axes théoriques :

$$[Z_{imes} - Z_{ith} \quad Pt_{imes} - Pt_{ith}]^t = [dZ_i \quad dPt_i]^t = [da_i \quad dt_i]^t \quad \text{Eq. 69}$$

À l'instar des méthodes classiques d'identification, il faut maintenant lier cette variation sur la pose des axes à une variation sur les paramètres géométriques au travers d'une jacobienne. Nous allons donc devoir formuler pour chaque axe i , une jacobienne J_{Ai} basée sur notre modèle de g paramètres géométriques $[w_1 \quad \dots \quad w_g]$ telle que :

$$[dZ_i \quad dPt_i]^t = J_{Ai} [dw_1 \quad \dots \quad dw_g]^t \quad \text{Eq. 70}$$

La jacobienne J_{Ai} faisant le lien entre la différence sur la pose des axes mesurés et calculés, et les corrections à apporter aux paramètres. Ces corrections peuvent donc être calculées par résolution de l'équation 70 et avec des méthodes classiques de minimisation ou utilisant la pseudo-inverse.

2.5.2 Détails du calcul de la matrice jacobienne pour la méthode ANCAR

Avant toute résolution il faut connaître les valeurs de la matrice jacobienne. Par définition, la jacobienne est déduite de la dérivée partielle du modèle de position par rapport aux g paramètres, c'est-à-dire dans notre cas, de la dérivée de la direction a_i et du point t_i par rapport aux paramètres w_j :

$$J_{Ai} = \begin{bmatrix} \frac{\partial a_i}{\partial w_1} & \dots & \frac{\partial a_i}{\partial w_g} \\ \frac{\partial t_i}{\partial w_1} & \dots & \frac{\partial t_i}{\partial w_g} \end{bmatrix} \quad \text{Eq. 71}$$

Cependant la formulation de l'équation 71 est peu pratique, car elle requiert un calcul analytique pour obtenir les dérivées. Comme dans la section 2.2.2, il est possible de calculer cette matrice jacobienne colonne par colonne, ce qui rend le calcul plus pratique et surtout générique. Ce calcul de J_{Ai} est également valable si l'on exprime les variations de l'équation 70 par rapport au temps :

$$\left[\frac{dZ_i}{dt} \quad \frac{dPt_i}{dt} \right]^t = J_{Ai} \left[\frac{dw_1}{dt} \quad \dots \quad \frac{dw_g}{dt} \right]^t \quad \text{Eq. 72}$$

Comme dans la section 2.2.2, le calcul par colonne repose sur la formule de Varignon, qui peut exprimer l'effet de la vitesse d'un paramètre w_j sur la vitesse d'un point Pt_i . Soit L_{w_j, Pt_i} le vecteur allant de l'axe portant le paramètre w_j au point Pt_i :

$$[V_{Pt_i}]_j = V_{w_j} + \Omega_{w_j} \wedge L_{w_j, Pt_i} \quad \text{Eq. 73}$$

On obtient ainsi le taux de variation par rapport au temps des coordonnées Pt_i du au paramètre w_j . Pour pouvoir calculer de la même façon, le taux de variation de la direction Z_i , nous exprimons le vecteur Z_i comme étant la différence entre deux points $\{C_i, D_i\}$ de l'axe i , distants l'un de l'autre de la valeur de la norme de Z_i . On peut alors calculer le taux de variation de Z_i par rapport au temps, comme étant la différence des variations des coordonnées C_i et D_i des points, puis appliquer simplement l'équation 73.

$$\frac{dZ_i}{dt} = \frac{dC_i}{dt} - \frac{dD_i}{dt} = V_{C_i} - V_{D_i} \quad \text{Eq. 74}$$

Il faut à présent déterminer les colonnes de J_{Ai} . Chaque colonne est associée à un paramètre qui peut être de translation ou de rotation. Deux cas de figure se présentent donc.

■ Paramètre de translation

Soit w_j un paramètre de translation dont la valeur est guidée par un vecteur u_w . On peut exprimer sa vitesse (de translation) avec son vecteur directeur et sa dérivée par la relation suivante :

$$V_{w_j} = u_w \frac{dw_j}{dt} \quad \text{Eq. 75}$$

Le paramètre w_j étant un paramètre de translation, la composante Ω de l'équation 73 est nulle. La vitesse des coordonnées Pt_i induite par la vitesse de w_j est alors :

$$\left[\frac{dPt_i}{dt} \right]_j = [V_{Pt_i}]_j = V_{w_j} = u_w \frac{dw_j}{dt} \quad \text{Eq. 76}$$

De la même manière, on peut déduire les vitesses de C_i et de D_i induites par w_j :

$$[V_{C_i}]_j = [V_{D_i}]_j = V_{w_j} \quad \text{Eq. 77}$$

Et on en déduit facilement le taux de variation de la direction Z_i induite par la vitesse d'un paramètre de translation w_j :

$$\left[\frac{dZ_i}{dt} \right]_j = [V_{C_i}]_j - [V_{D_i}]_j = 0_{\times 3} \quad \text{Eq. 78}$$

À partir des équations 72, 76 et 78, on obtient les valeurs d'une colonne de J_{Ai} correspondant à un paramètre de translation :

$$J_{Aij} = [0_{\times 3} \quad u_w]^t \quad \text{Eq. 79}$$

■ Paramètre de rotation

Soit w_j un paramètre de rotation. Ce paramètre est une valeur associée à un vecteur v_w qui guide la rotation. On peut exprimer la vitesse de ce paramètre avec son vecteur directeur et sa dérivée par la relation suivante :

$$\Omega_{w_j} = v_w \frac{dw_j}{dt} \quad \text{Eq. 80}$$

Le paramètre w_j étant un paramètre de rotation, tout point appartenant à l'axe de rotation possède alors une vitesse de translation nulle (V_{w_j} nulle dans l'équation 73). La vitesse de Pt_i induite par la vitesse de w_j est alors :

$$\left[\frac{dPt_i}{dt} \right]_j = [V_{Pt_i}]_j = \Omega_{w_j} \wedge L_{w_j, Pt_i} = (v_w \wedge L_{w_j, Pt_i}) \frac{dw_j}{dt} \quad \text{Eq. 81}$$

De la même manière, on trouve les vitesses de C_i et de D_i induites par w_j :

$$[V_{C_i}]_j = \Omega_{w_j} \wedge L_{w_j, C_i} \quad \text{Eq. 82}$$

$$[V_{D_i}]_j = \Omega_{w_j} \wedge L_{w_j, D_i} \quad \text{Eq. 83}$$

On en déduit le taux de variation de Z_i :

$$\left[\frac{dZ_i}{dt} \right]_j = [V_{C_i}]_j - [V_{D_i}]_j = \Omega_{w_j} \wedge L_{w_j, C_i} - \Omega_{w_j} \wedge L_{w_j, D_i} = \Omega_{w_j} \wedge (L_{w_j, C_i} - L_{w_j, D_i}) \quad \text{Eq. 84}$$

$$\left[\frac{dZ_i}{dt} \right]_j = \Omega_{w_j} \wedge Z_i = (v_w \wedge Z_i) \frac{dw_j}{dt} \quad \text{Eq. 85}$$

Ainsi pour un paramètre w_j de rotation, la colonne de J_{A_i} correspondante vaut :

$$J_{A_{ij}} = [v_w \wedge Z_i \quad v_w \wedge L_{w_j, Pt_i}]^t \quad \text{Eq. 86}$$

■ Paramètre aval

Soit w_j un paramètre aval à un axe i . Ce paramètre n'intervient donc pas sur le positionnement de l'axe i et la colonne de J_{A_i} correspondante vaut :

$$J_{A_{ij}} = [0_{\times 3} \quad 0_{\times 3}]^t \quad \text{Eq. 87}$$

■ Concaténation de la jacobienne

Maintenant que la jacobienne peut être obtenue numériquement sans difficulté, on peut chercher à la construire. Pour un axe i , la relation entre les variations sur les g paramètres du modèle géométrique et les variations dZ_i et dPt_i s'écrit :

$$[dZ_i \quad dPt_i]^t = [J_{A_{i1}} \quad \dots \quad J_{A_{ig}}] [dw_1 \quad \dots \quad dw_g]^t \quad \text{Eq. 88}$$

Pour notre modèle géométrique de robot complet à 6 axes et à g paramètres, nous obtenons donc :

$$[dZ_1 \quad dPt_1 \quad \dots \quad dZ_6 \quad dPt_6]^t = \begin{bmatrix} J_{A_{11}} & \dots & J_{A_{1g}} \\ \vdots & \ddots & \\ J_{A_{61}} & & J_{A_{6g}} \end{bmatrix} \begin{bmatrix} dw_1 \\ \vdots \\ dw_g \end{bmatrix} \quad \text{Eq. 89}$$

2.5.3 Corrections sur la résolution de la méthode ANCAR avec jacobienne

La première étape de la méthode ANCAR, basée sur la mesure d'arcs de cercle, nous donne la pose (position et orientation) des axes que nous traduisons par une direction Z_i et les coordonnées Pt_i d'un point sur l'axe. Dans notre cas, nous appelons Pt_{imes} le centre identifié à partir de l'arc de cercle mesuré, et Z_{imes} l'axe identifié à partir de ce même arc. Le principe de la méthode ANCAR, implique que les coordonnées Pt_{imes} sur l'axe dépendent du positionnement des cibles sur l'effecteur et de la configuration du robot lors de la rotation. Le point Pt_{imes} ne correspond donc pas au point théorique Pt_{ith} du modèle DH modifié, qui est l'intersection entre la perpendiculaire commune et l'axe i représentés sur la figure 38. Il serait de toute façon difficile expérimentalement de faire correspondre ces points, car le point théorique Pt_{ith} n'a aucune réalité physique sur le robot.

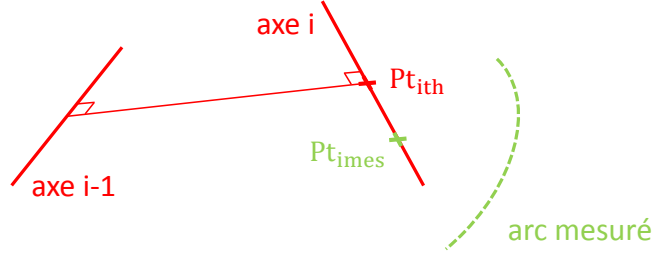


Figure 38 : Différence entre point mesuré et point théorique, résolution ANCAR

Ainsi, il n'est pas possible de mesurer la position de Pt_{ith} avec la méthode ANCAR, et il est difficilement possible de calculer la position théorique du point Pt_{imes} (paramètres manquants). Or pour l'identification, il est nécessaire d'avoir la position calculée et mesurée d'un point de l'axe de rotation.

Un point à l'origine de l'axe peut satisfaire ce rôle. Ce point est situé à l'intersection de l'axe et d'un plan et correspond donc aux ordonnées à l'origine de l'axe. Par exemple, pour un axe plutôt orienté selon Z , qui coupe le plan XY du repère externe (du C-Track) en un point Pt_{i0} on a :

$$Pt_{i0} = (x_{i0} \quad y_{i0} \quad 0) \quad \text{Eq. 90}$$

Les coordonnées Pt_{i0} de ce point peuvent être obtenues par le modèle théorique et par mesure, en partant des points Pt_{ith} et Pt_{imes} en suivant la direction de l'axe Z_{ith} ou Z_{imes} jusqu'à couper le plan XY. On peut alors calculer et mesurer la position du point Pt_{i0} , ce qui donne les points Pt_{i0th} et Pt_{i0mes} .

Il y a bien sûr deux autres cas de figure, car en fonction de la direction prépondérante de Z_i , nous choisirons de couper des plans différents. Ainsi nous définissons le plan d'intersection et le point Pt_{i0} par l'équation 91, qui est fonction du vecteur directeur Z_i .

$$Z_i = [x_{Zi} \quad y_{Zi} \quad z_{Zi}] , \quad \begin{cases} \text{si } x_{Zi} > y_{Zi} \text{ et } z_{Zi} \rightarrow Pt_{i0} = (0 \quad y_{i0} \quad z_{i0}) \\ \text{si } y_{Zi} > x_{Zi} \text{ et } z_{Zi} \rightarrow Pt_{i0} = (x_{i0} \quad 0 \quad z_{i0}) \\ \text{si } z_{Zi} > y_{Zi} \text{ et } x_{Zi} \rightarrow Pt_{i0} = (x_{i0} \quad y_{i0} \quad 0) \end{cases} \quad \text{Eq. 91}$$

Nous prendrons pour la suite l'exemple d'un axe i dont la direction prépondérante est suivant l'axe Z du repère externe. La reformulation de l'équation 88 utilisant ce nouveau point et une nouvelle jacobienne devrait s'écrire :

$$[dZ_i \quad dx_{i0} \quad dy_{i0}]^t = J_{Ai}' [dw_1 \quad \dots \quad dw_g]^t \quad \text{Eq. 92}$$

La partie supérieure de J_{Ai}' correspondant à dZ_i reste inchangée. Il faut reformuler la partie inférieure de la jacobienne pour correspondre à dx_{i0} et dy_{i0} au lieu de dPt_i . En se basant sur les formules déjà calculées de Z_i , dZ_i , Pt_i et dPt_i dans la section 2.5.2, nous allons retrouver dPt_{i0} . Tout d'abord il faut

déterminer Pt_{i0} à partir du point Pt_i déjà présent sur l'axe et de la direction Z_i . Pt_{i0} est le point où $z_{i0} = 0$, ce qui nous permet d'obtenir :

$$\begin{cases} x_{i0} = Pt_i(1) - Z_i(1) \frac{Pt_i(3)}{Z_i(3)} \\ y_{i0} = Pt_i(2) - Z_i(2) \frac{Pt_i(3)}{Z_i(3)} \end{cases} \quad \text{Eq. 93}$$

On calcul ensuite dPt_{i0} à partir de Pt_{i0} en passant par une dérivée partielle au premier ordre :

$$\begin{cases} dx_{i0} = dPt_i(1) - dPt_i(3) \frac{Z_i(1)}{Z_i(3)} - dZ_i(1) \frac{Pt_i(3)}{Z_i(3)} + Z_i(1) \frac{Pt_i(3)}{Z_i(3)^2} dZ_i(3) \\ dy_{i0} = dPt_i(2) - dPt_i(3) \frac{Z_i(2)}{Z_i(3)} - dZ_i(2) \frac{Pt_i(3)}{Z_i(3)} + Z_i(2) \frac{Pt_i(3)}{Z_i(3)^2} dZ_i(3) \end{cases} \quad \text{Eq. 94}$$

En exploitant les valeurs déjà calculées dans l'ancienne jacobienne J_{Aij} , on peut exprimer les valeurs de dPt_i et de dZ_i , on obtient :

$$\begin{cases} dx_{i0} = \sum_{j=1:g} \left(J_{Aij}(4) - J_{Aij}(6) \frac{Z_i(1)}{Z_i(3)} - J_{Aij}(1) \frac{Pt_i(3)}{Z_i(3)} + Z_i(1) \frac{Pt_i(3)}{Z_i(3)^2} J_{Aij}(3) \right) dw_j \\ dy_{i0} = \sum_{j=1:g} \left(J_{Aij}(5) - J_{Aij}(6) \frac{Z_i(2)}{Z_i(3)} - J_{Aij}(2) \frac{Pt_i(3)}{Z_i(3)} + Z_i(2) \frac{Pt_i(3)}{Z_i(3)^2} J_{Aij}(3) \right) dw_j \end{cases} \quad \text{Eq. 95}$$

Donc pour un axe i et pour un paramètre w_j , la colonne correspondante de la nouvelle jacobienne vaut :

$$J_{Aij}' = \begin{bmatrix} J_{Aij}(1:3) \\ J_{Aij}(4) - J_{Aij}(6) \frac{Z_i(1)}{Z_i(3)} - J_{Aij}(1) \frac{Pt_i(3)}{Z_i(3)} + Z_i(1) \frac{Pt_i(3)}{Z_i(3)^2} J_{Aij}(3) \\ J_{Aij}(5) - J_{Aij}(6) \frac{Z_i(2)}{Z_i(3)} - J_{Aij}(2) \frac{Pt_i(3)}{Z_i(3)} + Z_i(2) \frac{Pt_i(3)}{Z_i(3)^2} J_{Aij}(3) \end{bmatrix} \quad \text{Eq. 96}$$

Il est alors possible de construire la nouvelle jacobienne totale. La configuration des axes du robot pour ce calcul de jacobienne permet de choisir les plans de coupe de ces axes (pour dx_{i0} , dy_{i0} ou dz_{i0}) :

$$[dZ_1 \quad dx_{10} \quad dz_{10} \quad \dots \quad dZ_6 \quad dx_{60} \quad dy_{60}]^t = \begin{bmatrix} J_{A11}' & \dots & J_{A1g}' \\ \vdots & \ddots & \vdots \\ J_{A61}' & & J_{A6g}' \end{bmatrix} \begin{bmatrix} dw_1 \\ \vdots \\ dw_g \end{bmatrix} \quad \text{Eq. 97}$$

$$[Z_{1mes} - Z_{1th} \quad x_{10mes} - x_{10th} \quad z_{10mes} - z_{10th} \quad \dots \quad y_{60mes} - y_{60th}]^t = J_A' \begin{bmatrix} dw_1 \\ \vdots \\ dw_g \end{bmatrix} \quad \text{Eq. 98}$$

2.5.4 Adaptabilité de la jacobienne

La jacobienne totale établie dans l'équation 97 et la pose des axes du robot permettent d'identifier les paramètres propres du robot et les paramètres de positionnement du robot dans le repère externe (repère C-Track) que l'on retrouve dans $T_{base,robot}$ et T_{robot} de l'équation 42. Mais la pose des axes ne permet pas d'obtenir les paramètres de $T_{robot,cibles}$ qui positionnent des cibles sur l'effecteur, et qui se situent après l'axe 6. Afin d'identifier ces paramètres, il faut ajouter des informations dans le vecteur d'entrée, comme la variation sur la position des cibles $dP_{cibles,p}$. Il suffit ensuite de compléter la jacobienne totale

J_A' de l'équation 97, avec la jacobienne J_w de l'équation 6, pour lier les variations sur les paramètres à cette nouvelle entrée. Les paramètres outil peuvent alors être identifiés avec l'équation 99 obtenue.

$$[dZ_1 \quad dx_{10} \quad dz_{10} \quad \dots \quad dZ_6 \quad dx_{60} \quad dy_{60} \quad dP_{cibles,p}]^t = \begin{bmatrix} J_{A11}' & \dots & J_{A1g}' \\ \vdots & \ddots & \\ J_{A61}' & & J_{A6g}' \\ J_{w1} & & J_{wg} \end{bmatrix} \begin{bmatrix} dw_1 \\ \vdots \\ dw_g \end{bmatrix} \quad \text{Eq. 99}$$

Il est possible d'avoir également les paramètres d'orientation de l'ensemble de cibles si on ajoute en entrée une mesure d'orientation de l'effecteur $dP_{cibles,o}$. Ce que permet de souligner cet ajout dans le vecteur d'entrée, c'est les évolutions possibles de la matrice jacobienne. Notre méthode ANCAR étant à présent dotée d'une matrice jacobienne, il est possible de manipuler cette dernière pour identifier les paramètres avec un vecteur d'entrée différent, bien plus riche que simplement la position et l'orientation de 6 axes en entrée. Il est notamment possible de mesurer plusieurs fois chaque axe à l'instar de la méthode d'identification classique et de ses mises en position répétées de l'effecteur. On a alors la liberté d'apporter de la redondance et de la robustesse dans l'identification par ANCAR qui n'était basée que sur une série simple d'essais. De plus grâce à l'expression d'une jacobienne dans la méthode ANCAR, il est à présent possible de faire une méthode mixte. Le vecteur d'entrée peut être plus riche et varié, en utilisant à la fois la pose des axes, mais aussi la position (et/ou l'orientation) de l'effecteur ou d'autres points d'intérêt. La jacobienne est alors formulée différemment en fonction de la composition du vecteur d'entrée.

Un autre avantage important apporté par l'utilisation d'une matrice jacobienne lors de l'identification par ANCAR, est que l'on peut à présent appliquer toutes les techniques déjà existantes dans les méthodes classiques qui exploitent la jacobienne. On peut notamment calculer l'identifiabilité des paramètres pour la méthode ANCAR et étudier le conditionnement du problème.

2.5.5 Identification des paramètres par ANCAR, avec jacobienne

Afin de vérifier la validité de cette approche, il faut à présent utiliser la jacobienne et surtout l'équation 99 pour identifier les paramètres du modèle, c'est-à-dire les paramètres du robot, les paramètres de base et les paramètres de position de l'outil. Nous n'utilisons pas l'orientation de l'outil ici, et nous n'avons donc que $30-3=27$ paramètres. Soient dw le vecteur des variations sur les paramètres et $d\varepsilon$ le vecteur d'entrée (différences entre mesure et théorie). On peut simplifier l'écriture de l'équation 99 en :

$$d\varepsilon = J_{mix}' dw \quad \text{Eq. 100}$$

Tout d'abord on vérifie l'identifiabilité des 27 paramètres recherchés, en utilisant la méthode déjà mise en place dans la section 2.2. Le rang de la jacobienne J_{mix}' ainsi calculé est de 27, tous les paramètres sont donc bien identifiables.

L'identification des paramètres est ensuite réalisée à partir de l'équation 100, en calculant la pseudo inverse de J_{mix}' , puis en faisant un calcul direct du vecteur dw :

$$(J_{mix}' J_{mix}'^t)^{-1} J_{mix}'^t d\varepsilon = dw \quad \text{Eq. 101}$$

Pour cette identification, les données d'entrée utilisées sont celles qui ont servi à l'identification des paramètres par notre méthode ANCAR locale optimisée présentée en section 2.4, ceci afin de pouvoir comparer les résultats obtenus. Les mesures utilisées ont donc reçu le même traitement sur les moindres carrés et les distorsions de mesure. Pour cette nouvelle identification exploitant une jacobienne, les résultats sur les corrections à apporter aux paramètres géométriques propres au robot (longueurs) sont donnés sur la figure 39. La résolution de l'identification est effectuée avec 10 itérations.

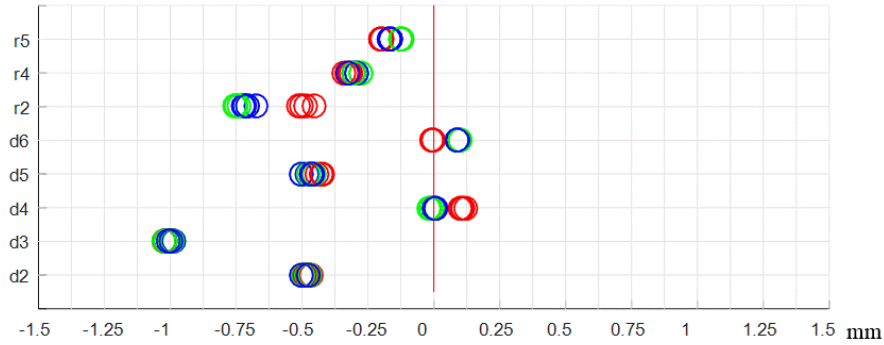


Figure 39 : Corrections pour paramètres du robot, ANCAR locale optimisée avec jacobienne, 10 itérations

Si le nombre d'itérations est suffisamment grand, les paramètres obtenus sont exactement ceux trouvés avec la méthode ANCAR locale optimisée sans jacobienne. Pour 500 itérations, on retrouve alors les valeurs présentées sur la figure 40, qui sont les mêmes que sur la figure 36. Pour passer de 10 à 500 itérations, le temps de calcul n'est augmenté que de 5 secondes.

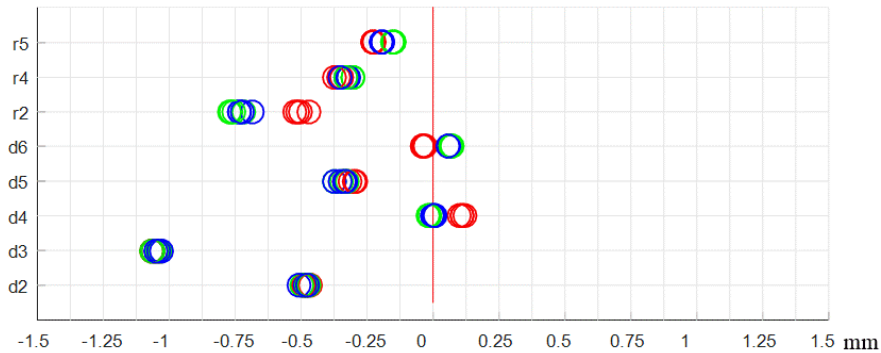


Figure 40 : Corrections pour paramètres du robot, ANCAR locale optimisée avec jacobienne, 500 itérations

Pour chacune des deux méthodes employées (avec jacobienne ou construction géométrique), la phase de mesure donnant la position et l'orientation des axes du robot est la même. Mais ensuite la phase d'identification à partir de ces axes diffère. Cependant même si elles diffèrent, la méthode basée sur une reconstruction géométrique classique et cette nouvelle méthode exploitant une jacobienne donnent les mêmes résultats pour les mêmes données d'entrée. L'utilisation de notre jacobienne ne dégrade donc pas la qualité de l'identification et il est possible de l'utiliser pour exploiter tous les avantages de notre nouvelle méthode d'identification ANCAR. Le seul inconvénient découlant de l'utilisation d'une jacobienne pour la résolution du problème, comme avec toutes les méthodes classiques d'identification, est que nous avons besoin de valeurs initiales assez proches des valeurs réelles pour que le calcul converge vers la bonne solution. La méthode ANCAR n'utilisant pas la jacobienne ne requiert aucun a priori de départ sur les valeurs des paramètres.

2.5.6 Apports d'une jacobienne dans notre méthode ANCAR

Comme vu précédemment l'utilisation d'une jacobienne ne change en rien la qualité des résultats obtenus si l'on s'en tient à une identification géométrique. Mais elle permet de simplifier l'implémentation de la méthode ANCAR dans un code et réduire les erreurs de calcul et de code. En effet notre construction générique d'une jacobienne à partir des matrices homogènes est bien plus pratique que la reconstruction géométrique habituellement utilisée, déduisant les paramètres du modèle au cas par cas. Elle réduit donc les risques d'implémenter des erreurs. C'est pourquoi nous choisissons d'utiliser cette jacobienne. De plus, ce n'est pas le seul avantage à l'utilisation d'une jacobienne dans

une identification par ANCAR, nous présentons dans ce qui suit les différents avantages et perspectives de notre méthode d'identification.

■ Notre formulation de la méthode d'identification par ANCAR utilisant une jacobienne permet une standardisation de la méthode CPA originale. Jusqu'à présent cette méthode différait d'un auteur à l'autre, d'un robot à l'autre et seul le principe de base (étude d'arcs de cercle) était commun. La jacobienne offre une généralité dans la méthode qui peut alors être reproduite et facilement adaptée à différentes structures de robots. D'autre part la méthode CPA était jusqu'à lors peu utilisée, car elle offre peu de perspectives et d'adaptabilité. Une standardisation de la méthode avec une jacobienne la remonte au rang des méthodes classiques. Cette jacobienne lui permet d'exploiter tout ce qui a déjà été fait sur les méthodes d'identification classiques, comme les travaux sur l'identifiabilité des paramètres, tout en conservant les avantages d'une identification par méthode CPA (ici par méthode ANCAR).

■ La structure même de la jacobienne apporte énormément de flexibilité à l'identification. Il est possible de la construire pour un vecteur d'entrée aussi grand que l'on souhaite, de même que pour le vecteur de paramètres. Il est donc possible de fournir plus de données d'entrée, par exemple en mesurant plusieurs fois les axes, afin d'obtenir une identification plus robuste et des valeurs de paramètres plus fiables. Mais surtout, cela permet d'envisager l'identification de paramètres non géométriques avec notre méthode ANCAR, en ajoutant des paramètres de jeu ou d'élasticité et en approvisionnant le vecteur d'entrée avec plus de mesures. De plus, la jacobienne permet de vérifier si les paramètres ajoutés sont bien identifiables avec les données d'entrée, en appliquant des études classiques d'identifiabilité des paramètres. Par ailleurs, nous avons intégré dans ce chapitre, la position et l'orientation d'axes dans le vecteur d'entrée de la jacobienne. Cela implique qu'il est possible de fournir au vecteur d'entrée, des données de type géométrique. Il est donc envisageable d'intégrer des formes autres que des axes, afin d'identifier des paramètres géométriques ou non géométriques à partir d'études de trajectoires complexes suivies par le robot. Et cette étude de trajectoire complexe passera simplement par l'utilisation d'une jacobienne lors de la résolution.

■ Enfin, la jacobienne permet par exemple d'estimer l'identifiabilité des paramètres. Nous l'utilisons également pour calculer la propagation de l'incertitude du moyen de mesure à l'incertitude sur les paramètres identifiés dans la section 2.6.

2.6 Comparaison et validation des paramètres identifiés

Notre méthode ANCAR a été choisie pour répondre à certains critères sur l'identification des paramètres, comme par exemple la praticité de la méthode, la vitesse d'exécution ou encore la précision des résultats obtenus. Maintenant que notre méthode a été mise en place, il est nécessaire d'une part de vérifier ses performances et de les comparer à celles d'une méthode classique, et d'autre part de valider les paramètres géométriques qui ont été identifiés. Dans ce but, pour la méthode classique et pour notre méthode ANCAR, nous avons étudié : le temps de mesure puis le temps de calcul pour l'identification, la propagation de l'incertitude du moyen de mesure sur les paramètres identifiés. Quant à la notion de praticité, elle est quasiment identique dans les deux méthodes puisque nous utilisons le même jeu de cibles sur l'outil ainsi que le même volume de mesure du C-Track. De plus la génération de la trajectoire n'est pas plus complexe dans un cas que dans l'autre. Enfin, un protocole de validation des paramètres est présenté afin de déterminer la qualité de positionnement du robot obtenue avec les 2 modèles aux paramètres identifiés. On peut alors comparer la qualité des paramètres donnés par les deux méthodes (classique et ANCAR). Bien sûr les paramètres utilisés pour notre méthode sont ceux obtenus avec la méthode ANCAR optimisée.

2.6.1 Estimations des temps de mesure et de résolution

Le temps de l'étape de mesure de chaque méthode, appelé temps de mesure, représente le temps d'exécution de la trajectoire d'identification servant à générer les données pour l'étape d'identification. Ce critère de comparaison est donc mesuré pour chaque méthode. De même, on mesure le temps de résolution de l'identification, appelé temps de calcul, qui est le temps de traitement pour obtenir les paramètres du robot une fois les données enregistrées. Ce traitement est effectué, pour chaque méthode, avec le même ordinateur présenté en section 1.1.2. Le tableau 13 présente les différents résultats obtenus.

	ANCAR optimisée	ANCAR optim (1000 points)	Identification classique
Temps de mesure	6 min	3 min	4 min
Temps de calcul	1,6 min	1,6 min	0,2 min

Tableau 13 : Temps requis pour les différentes méthodes (mesure et calcul)

La méthode ANCAR optimisée est légèrement plus longue à appliquer qu'une méthode classique. Le temps de mesure est dû à la vitesse imposée sur les arcs de cercle pour obtenir 2000 points au moins sur ces arcs. Il est possible de diviser ce temps par deux en ne prenant que 1000 points. Une étude sur la validité de paramètres estimés avec seulement 1000 points est investiguée dans cette section lors de la validation des paramètres. Le temps de calcul est dû aux rectifications d'une partie des distorsions du moyen de mesure que nous avons vu en section 2.4.3. La connaissance du mouvement de rotation de la méthode ANCAR permet la mise en place de telles rectifications sur la mesure du C-Track. Mais la simple mise en position de l'effecteur dans la méthode classique ne le permet pas. C'est pourquoi la résolution classique est plus rapide, mais sujette à ces distorsions observée dans la mesure avec le C-Track. La seule amélioration qui allonge le temps de mesure de la méthode classique est un temps d'attente de 3 secondes sur chaque pose de l'effecteur pour effectuer une moyenne sur la mesure de la pose et ainsi réduire le bruit de mesure. Le temps d'identification est donc du même ordre de grandeur pour les deux méthodes, même si la méthode ANCAR optimisée présente une résolution bien plus longue à cause des différentes optimisations sur le traitement de la mesure.

2.6.2 Fiabilité théorique sur les paramètres identifiés

Nous comparons ensuite la fiabilité des paramètres identifiés pour un moyen de mesure donné, c'est-à-dire la propagation de l'incertitude du moyen de mesure sur les paramètres. Nous étudions cette propagation dans le cas de la méthode classique et dans le cas de notre méthode ANCAR optimisée. Pour la méthode classique, il est possible d'obtenir ce résultat analytiquement en exploitant la jacobienne du modèle J_{tot_w} utilisée lors de la résolution de l'identification.

L'évaluation de l'incertitude sur les paramètres avec la méthode classique est assez simple. Connaissant l'incertitude du moyen de mesure suivant X , Y et Z dans le repère du C-Track, on peut créer un vecteur d'incertitude sur les données d'entrée \widetilde{dPtot} . L'incertitude du C-Track correspond aux valeurs de répétabilité $[\sigma_x, \sigma_y, \sigma_z]$ obtenues à dans la section 1.5.2. On pourrait penser que l'utilisation de la pseudo inverse de J_{tot_w} permettrait ensuite d'obtenir directement le vecteur d'incertitude sur les paramètres \widetilde{dw} en suivant l'équation de base 24. Cependant le vecteur d'incertitudes \widetilde{dPtot} est composé d'écarts types associés à des lois normales, et des règles différentes s'appliquent pour sommer des écarts types. En effet, pour deux lois normales L_1 et L_2 de même variable aléatoire, pondérées de coefficients c_1 et c_2 , de moyennes μ_1, μ_2 et d'écarts types σ_1, σ_2 , la somme est décrite dans l'équation suivante 102.

$$c_1 L_1(\mu_1, \sigma_1) + c_2 L_2(\mu_2, \sigma_2) = L(c_1 \mu_1 + c_2 \mu_2, \sqrt{|c_1| \sigma_1^2 + |c_2| \sigma_2^2}) \quad \text{Eq. 102}$$

L'utilisation de la pseudo inverse de J_{tot_w} et du vecteur \widetilde{dPtot} dans l'équation de base 24 revient à

sommer des lois normales, il faut donc utiliser l'équation 102 pour en déduire une formule donnant les incertitudes sur les paramètres. Dans notre cas, l'expression de l'incertitude sur un paramètre \widetilde{dw}_i se fait alors en fonction du carré des incertitudes sur les $6 \times m$ mesures (position et orientation) et des valeurs absolues des composantes de $J_{tot_w}^+ = A$, comme présenté dans l'équation 103.

$$|A_{i,1}| \widetilde{dP}_1^2 + |A_{i,2}| \widetilde{dP}_2^2 + \dots + |A_{i,6m}| \widetilde{dP}_{6m}^2 = \widetilde{dw}_i^2 \quad \text{Eq. 103}$$

Comme lors de l'identification, on utilise $m=30$ poses de l'effecteur pour évaluer de manière plus vraisemblable la propagation d'incertitude. Lors des essais nous avons de plus fait des pauses de 3 secondes durant les 30 mises en position ce qui a permis de réduire de moitié l'incertitude du C-Track selon X , Y et Z , pour l'étude de propagation d'erreur sur la méthode classique. A partir des valeurs d'incertitude du C-Track rentrées dans les \widetilde{dP}_k ($k=1:6m$), du calcul de la pseudo inverse de J_{tot_w} , et de l'équation 103, on obtient les valeurs théoriques des incertitudes sur les paramètres géométriques. Les résultats d'incertitude sur ces paramètres identifiés par méthode classique, sont présentés dans le tableau 14 et le tableau 15.

Pour notre méthode ANCAR optimisée, l'étude statistique présentée en annexe 2 servant à la base à déterminer le nombre de points de mesure sur un arc, permet de connaître la propagation de l'incertitude de mesure du C-Track à l'incertitude sur le positionnement et l'orientation des 6 axes du robot. Pour remonter ensuite jusqu'aux paramètres à partir des axes, on utilise la jacobienne J_A (équation 89) de la méthode ANCAR introduite dans la section 2.5.2, et rappelée dans l'équation 104 :

$$[dZ_1 \quad dPt_1 \quad \dots \quad dZ_6 \quad dPt_6]^t = dD = J_A dw \quad \text{Eq. 104}$$

Le positionnement et l'orientation des 6 axes s'exprime au travers des 6 directions et 6 points que l'on retrouve dans le terme de gauche dD de l'équation 104. L'étude de l'annexe 2 nous permet donc de construire le vecteur d'incertitude sur les données d'entrée \widetilde{dD} . La pseudo inverse J_A^+ est ensuite calculée. Dans l'équation 103, $J_{tot_w}^+$ est remplacée par J_A^+ , et \widetilde{dPtot} par \widetilde{dD} , pour pouvoir aboutir au vecteur d'incertitude sur les paramètres \widetilde{dw} . Les incertitudes théoriques sur les paramètres propres au robot obtenues avec chaque méthode (classique et ANCAR) sont présentées dans le tableau 14 pour les paramètres de distance et le tableau 15 pour les paramètres angulaires.

Longueurs en μm	d2	r2	d3	d4	r4	d5	r5	d6
ANCAR optimisée	9,7	7,4	6,4	5,2	8,0	8,8	5,9	8,6
Classique	22,1	52,7	13,1	37,3	19,2	9,2	88,3	105,7

Tableau 14 : Incertitudes sur les paramètres de distance avec méthode CPA modifiée et méthode classique

Angles, rad 10^{-3}	α_2	$\delta\theta_2$	α_3	β_3	$\delta\theta_3$	α_4	$\delta\theta_4$	α_5	$\delta\theta_5$	α_6
ANCAR optimisée	0,0040	0,21	0,013	0,011	0,21	0,0084	0,014	0,015	0,11	0,091
Classique	0,81	0,91	0,75	0,70	0,99	1,3	1,0	1,1	7,0	5,8

Tableau 15 : Incertitudes sur les paramètres angulaires avec méthode CPA modifiée et méthode classique

On retrouve dans ces tableaux un facteur 10 entre les incertitudes sur les paramètres identifiés avec les deux méthodes, pour une même incertitude du moyen de mesure. Notre méthode ANCAR a été optimisée pour éviter des problèmes de distorsion entre des cibles proches sur l'effecteur et pour avoir une orientation cohérente avec le mouvement de rotation. Ces améliorations traitent de problèmes qui n'affectent pas cette étude théorique sur l'incertitude, et n'avantagent donc en aucun cas la méthode ANCAR optimisée. Comme le laissait supposer l'identification des paramètres géométriques du robot, notre méthode peut être jugée meilleure en ce qui concerne la propagation de l'erreur du moyen de mesure. La méthode classique peut tout de même être légèrement améliorée en sélectionnant les configurations de mesure minimisant cette propagation, en utilisant notamment le conditionnement de la matrice jacobienne.

Bien sûr les résultats de propagation théorique de ces tableaux ne sont pas du même ordre de grandeur que la répétabilité constatée sur les identifications réelles des paramètres faites à la section précédente. Seules les tendances sont similaires, et des phénomènes plus complexes autres que l'incertitude du C-Track entrent en compte et ajoutent de l'incertitude sur les paramètres. Il y a notamment des distorsions de grande échelle dans le champ de vision du C-Track, avec lesquelles la distance entre deux points éloignés peut varier légèrement (de l'ordre de 0,5 mm). Ces déviations ou distorsions affectent tout autant la méthode ANCAR optimisée et la méthode classique et augmentent l'incertitude sur la mesure, mais elles sont difficilement quantifiables d'où la différence entre incertitudes théoriques et incertitudes mesurées.

2.6.3 Fiabilités observées sur l'identification des paramètres

La fiabilité théorique, que nous avons calculée par propagation de l'incertitude du moyen de mesure, permet de montrer que le principe même de notre méthode ANCAR la rend plus fiable que la méthode classique. En effet, c'est une étude théorique, elle permet donc de comparer les résultats obtenus avec les définitions théoriques de chaque méthode d'identification. Une telle étude théorique permet également de vérifier l'incertitude sur les paramètres à identifier avec chaque méthode. Si la valeur d'un paramètre est trop peu fiable, il est alors possible de modifier la configuration de mesure. Pour la méthode classique cela se traduit par un changement des 30 configurations utilisées. Pour notre méthode ANCAR il faut modifier la disposition des arcs de cercle dans le champ de vision du C-Track, en déplaçant ce dernier et en modifiant les trajectoires du robot. Une nouvelle étude théorique est ensuite menée pour tester à nouveau la fiabilité des paramètres identifiés.

Cependant, même si l'étude théorique joue un rôle important en amont, de nombreux phénomènes sur la mesure ne sont pas pris en compte. Il faut donc effectuer des essais réels pour vérifier les tendances données par l'étude théorique. Les figures de correction des paramètres de longueur, générées dans les sections précédentes, sont reprises pour étudier et comparer les différentes fiabilités obtenues lors des identifications.

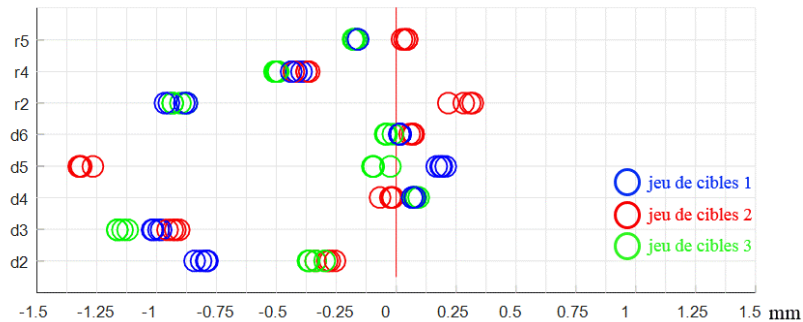


Figure 41 : Corrections pour paramètres du robot, ANCAR locale

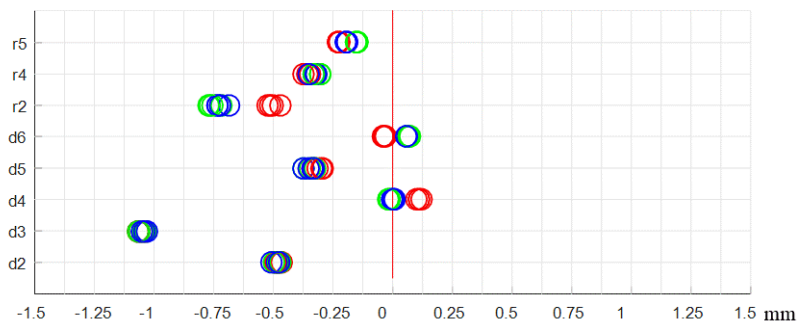


Figure 42 : Corrections pour paramètres du robot, ANCAR locale optimisée

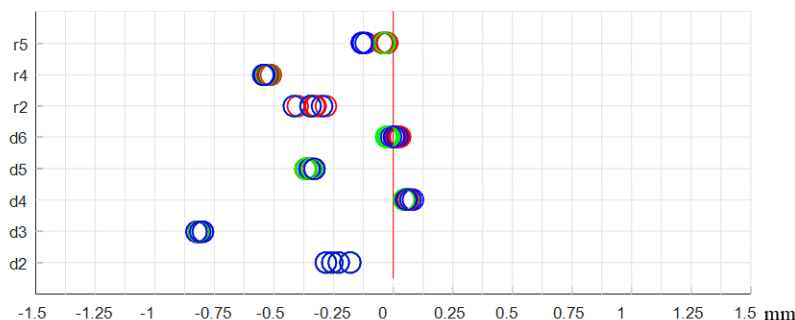


Figure 43 : Corrections pour paramètres du robot, ANCAR locale optimisée, 3 mois après

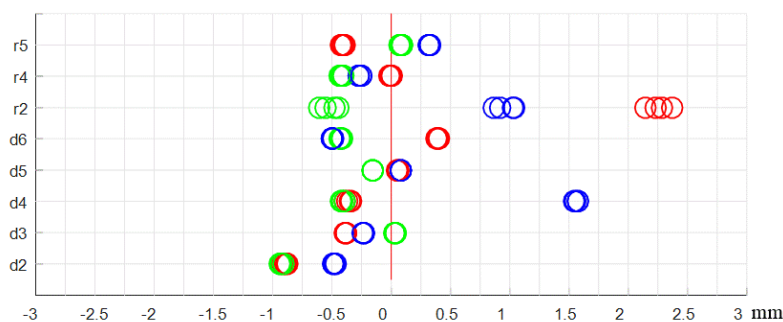


Figure 44 : Corrections pour paramètres du robot, méthode classique

La figure présentant les résultats obtenus avec la méthode ANCAR locale optimisée utilisant une jacobienne, n'est pas représentée ici car pour un même jeu de donnée (mesures) et pour 500 itérations, elle est exactement identique à la figure 42. On peut voir ici une nette amélioration de la fiabilité sur les paramètres entre la figure 41 et la figure 42. L'optimisation de la méthode ANCAR a donc une grande importance pour une mesure avec le C-Track, mais comme nous l'avons déjà dit, il n'est pas possible de l'appliquer pour améliorer les résultats de la figure 44. On peut également constater que malgré un délai de 3 mois, la figure 43 présente quasiment les mêmes valeurs et les mêmes fiabilités que la figure 42. Cela permet notamment de se rassurer sur les valeurs des paramètres identifiés que l'on va utiliser dans la modélisation du robot.

Sur ces figures on retrouve la même tendance donnant la méthode ANCAR plus fiable que la méthode classique d'un facteur 2 à 4. Mais l'ordre de grandeur sur les fiabilités est différent. Sur la théorie, la fiabilité sur les paramètres identifiés par la méthode ANCAR optimisée est de 5 à 10 μ m, tandis que sur les figures elle est de 60 à 250 μ m. Cette différence dans l'ordre de grandeur est encore plus flagrante pour la méthode classique. On a donc la fiabilité observée sur les figures qui est plus de 10 fois supérieure à celle obtenue théoriquement. Cette large différence s'explique par les nombreux phénomènes parasites sur la mesure du C-Track, qui ne sont pas pris en compte dans la théorie car trop complexes (distorsions de grande échelle, changement de température, éclairage).

Si l'on regarde les paramètres individuellement, on peut également retrouver une même tendance. On voit ressortir le paramètre r2 que ce soit dans les figures ou dans la théorie, avec la méthode classique ou notre méthode ANCAR. En effet ce paramètre est bien celui qui est le plus sujet à des incertitudes car, avec la modélisation choisie dans le chapitre 2, sa valeur dépend du positionnement des axes 1, 2, 3 et 4. Cette étude sur les résultats obtenus lors de l'identification des paramètres avec les différentes méthodes confirme l'avantage de notre méthode ANCAR optimisée sur la fiabilité de ces paramètres. De même elle confirme la présence de paramètres plus sensibles que d'autres, sur lesquels il pourrait être intéressant de travailler (changement des configurations de mesure).

2.6.4 Justesse relative et absolue sur le positionnement du capteur laser

Après avoir identifié les valeurs des paramètres géométriques, il faut évidemment vérifier la validité de ces valeurs en commandant le positionnement du robot avec notre nouveau modèle. Afin d'effectuer une comparaison nous étudierons ici le positionnement de l'effecteur en utilisant les modèles suivants : le modèle déjà présent dans l'armoire de commande, notre modèle identifié par ANCAR optimisée, le modèle identifié par une méthode classique, et enfin le modèle identifié par ANCAR optimisée, mais avec des arcs de seulement 1000 points. Il faut bien différencier cette étude de celle sur la propagation d'incertitude : une méthode qui propage peu l'incertitude du moyen de mesure ne donne pas nécessairement les bonnes valeurs des paramètres. Afin de valider les paramètres et de comparer les modèles, on recherche alors l'écart entre une pose commandée et la moyenne des poses atteintes.

Deux positionnements sont étudiés, un positionnement relatif et un positionnement absolu que nous qualifions de justesse relative du robot et justesse absolue du robot.

Justesse relative :

Tout d'abord une étude sur le positionnement relatif du robot permet de tester la validité des paramètres identifiés propres au robot. La démarche est la suivante : pour une direction donnée, le robot doté de cibles va parcourir une distance égale à un étalon lui aussi doté de cibles et disposé dans le champ de vision du C-Track comme sur la figure 46. Les mesures sont faites dans le repère du référentiel et non le repère du moyen de mesure afin d'éviter de possibles dérives de la mesure au cours du temps. Des dérives ont en effet été observées lors de longues campagnes de mesure comme c'est le cas ici. L'étalon utilisé a une longueur de 959,390 mm. Le point de départ de ce trajet est fixé et commun à toutes les trajectoires effectuées. Le point d'arrivée lui, sollicitera les différents modèles et donnera donc différentes trajectoires.

La génération des différents points d'arrivée est détaillée sur la figure 45. Le point d'arrivée du trajet est calculé avec les différents modèles obtenus avec chaque méthode c'est-à-dire : méthode classique, méthode ANCAR optimisée, méthode ANCAR optimisée avec seulement 1000 points par arc. Un point d'arrivé est également généré pour solliciter le modèle de l'armoire de commande, afin d'estimer la justesse relative obtenue avec la modélisation déjà présente dans l'armoire. Tous les points d'arrivée sont calculés pour que la distance théorique parcourue entre les deux points soit de 959,390 mm, soit la longueur de l'étalon.

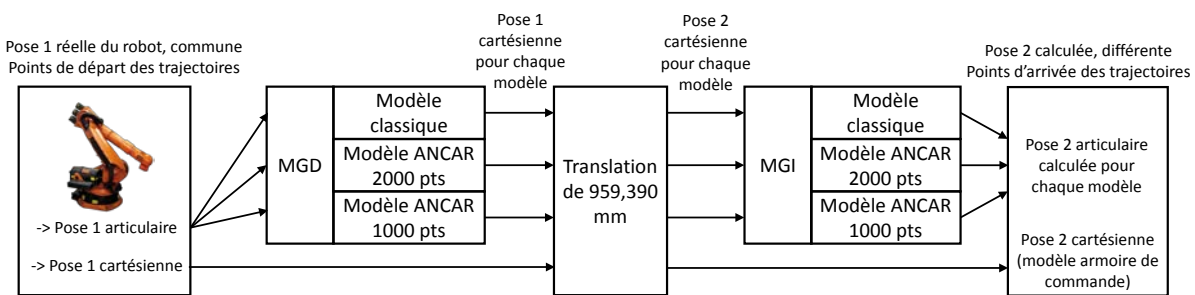


Figure 45 : Génération des points d'arrivée pour trajet de justesse relative

Pour générer ces points d'arrivée, le robot est placé dans une configuration de départ, dite pose 1, le point courant est alors enregistré dans chaque trajectoire comme point de départ. Les coordonnées articulaires du robot sont également relevées. A partir de ces coordonnées articulaires, le modèle direct (MGD) de chaque modèle est utilisé pour déduire la pose cartésienne de l'effecteur (pose1), d'après ces modèles. Puis un déplacement de 959,390 mm est effectué dans le domaine cartésien pour atteindre la pose 2. Cette étape n'utilise pas les modèles, c'est une simple translation suivant une même direction depuis chaque pose 1 cartésienne. Le point obtenu est ensuite ramené dans le domaine articulaire en utilisant le modèle inverse de chaque modèle. On obtient alors les poses 2 articulaires correspondant à

chaque méthode d'identification des paramètres. On enregistre alors ces poses comme points d'arrivée de chaque trajet, en coordonnées articulaires. Ainsi, avec des coordonnées articulaires, lorsque le robot ira de la pose 1 à la pose 2, il n'utilisera pas le modèle déjà présent dans l'armoire. Nous pouvons ainsi vraiment tester les résultats de chaque méthode sans interférence. Seule la dernière trajectoire, dont la pose 2 est exprimée en cartésien, usera du modèle de l'armoire afin de tester les capacités de ce modèle déjà présent.

Ces différentes trajectoires sont implémentées dans l'armoire de commande du robot et les distances réellement parcourues par l'effecteur sont mesurées et enregistrées par le C-Track. La présence de l'étalon dans la scène comme sur la figure 46, permet de compenser les possibles distorsions de mesures dues au C-Track. Les directions retenues pour les trajectoires sont suivant les axes X , Y et Z du moyen de mesure, et ces trajectoires sont répétées 6 fois. Les différences entre les distances commandées et les distances mesurées sont ensuite calculées. Les moyennes de ces différences sur 6 essais sont présentées dans le tableau 16, l'écart type maximal sur les mesures étant de 0,04 mm.

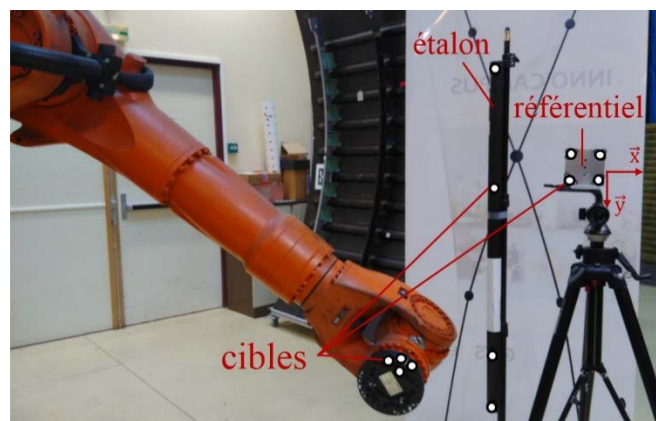


Figure 46 : Setup validation distance suivant verticale

en mm	armoire	classique	ANCAR optimisée	ANCAR 1000 pts
suivant X	0,1541	-0,03569	-0,2930	-0,3442
suivant Y	-0,6914	-0,8434	-0,6877	-0,7562
suivant Z	-0,1074	-0,2397	0,0657	-0,0510

Tableau 16 : Écarts en mm entre la distance commandée et les distances parcourues avec les différents modèles

Dans le repère utilisé, Y désigne la verticale, et Z la profondeur. Les résultats de cette première étude ne permettent pas de faire ressortir une méthode en particulier. Les valeurs sont relativement proches et chaque méthode donne des résultats plus ou moins bons suivant la direction considérée. Il ressort cependant que la méthode ANCAR n'utilisant que 1000 points par arc donne des valeurs proches de la méthode ANCAR qui utilise 2000 points. Il est donc possible de réduire le temps de mesure sans perdre en qualité sur les paramètres identifiés. Ce que l'on remarque également c'est un grand écart suivant Y quel que soit le modèle. Cette direction est celle subissant le plus l'effet de déformations sous l'effet de la gravité. C'est pourquoi les modèles purement géométriques ont du mal à positionner correctement l'effecteur. Au final, les perturbations dues aux erreurs non géométriques empêchent de différencier les différents modèles en termes de qualité de positionnement, sur la justesse relative. Pour obtenir des résultats plus concluants sur cette justesse relative, il faut donc utiliser des paramètres non géométriques. La modélisation des paramètres non géométriques est traitée dans le chapitre 3. Les différents modèles obtenus (avec différentes méthodes) ont donc pour la même validité au regard de cette justesse, tant que ces erreurs non géométriques ne seront pas prises en compte dans la modélisation.

Cette étude ne sollicite que les paramètres propres du robot, car l'objet de l'étude est la distance entre deux points obtenus par les modèles, peu importe donc le positionnement du robot dans un repère externe (base – axe 1) ainsi que la configuration de l'outil (modélisation entre axe 6 et les cibles).

Justesse absolue :

À présent une étude sur le positionnement absolu de l'effecteur (justesse absolue) va permettre également de vérifier la validité des paramètres de positionnement du robot dans un référentiel externe et des paramètres liés à l'outil. 10 poses sont choisies dans le champ de vision du C-Track, décrites en coordonnées cartésiennes (position et orientation). Les modèles géométriques inverses de chaque modèle à tester sont alors utilisés pour calculer les coordonnées articulaires de ces poses, comme pour l'étude de la justesse relative. Là aussi les 10 poses sont décrites avec des coordonnées articulaires pour ne pas faire intervenir le modèle de l'armoire de commande, excepté lors du test du modèle de l'armoire lui-même. Le robot doit alors se placer sur ces poses en suivant ces commandes articulaires données par les différents modèles. Après avoir mesuré le placement du robot en ces points pour chaque modèle, il est possible de vérifier la fiabilité d'un modèle en termes de position et d'orientation. Les mesures sont décrites dans un référentiel fixe placé dans la scène pour éviter de possibles déviations au cours du temps, et l'étalon est placé entre le référentiel et les poses choisies pour compenser d'éventuelles distorsions. Cette mise en position sur 10 poses est répétée 5 fois pour chaque modèle. Le tableau 17 présente la moyenne et l'écart type sur les justesses obtenues dans les 50 positionnements, pour chaque modèle. Les moyennes et écarts types sont décomposés pour visualiser la justesse suivant chaque direction (X, Y, Z) en position et orientation, là aussi dans le repère du moyen de mesure. L'écart type dans ce tableau représente la variabilité de la justesse sur l'espace de travail du robot, et n'est pas lié à la répétabilité du robot.

en mm et °	armoire	classique	ANCAR optim	ANCAR 1000 pts
suivant X (mm)	$0,85 \pm 0,68$	$1,36 \pm 0,17$	$0,07 \pm 0,42$	$0,29 \pm 0,44$
suivant Y (mm)	$-0,98 \pm 0,30$	$-3,34 \pm 0,28$	$-0,50 \pm 0,79$	$-1,24 \pm 0,81$
suivant Z (mm)	$0,41 \pm 0,37$	$0,56 \pm 0,14$	$-0,06 \pm 0,11$	$-0,86 \pm 0,30$
autour de X (°)	$0,064 \pm 0,11$	$-0,031 \pm 0,21$	$0,002 \pm 0,12$	$0,117 \pm 0,12$
autour de Y (°)	$0,024 \pm 0,10$	$-0,022 \pm 0,14$	$-0,056 \pm 0,11$	$0,083 \pm 0,08$
autour de Z (°)	$0,022 \pm 0,02$	$-0,062 \pm 0,14$	$-0,030 \pm 0,03$	$0,040 \pm 0,04$

Tableau 17 : Écarts moyens et écarts types entre les poses commandées et les poses mesurées pour chaque modèle

L'intérêt d'un bon positionnement absolu est de placer le robot dans la scène, et d'avoir de bonnes interactions avec d'autres objets dont le positionnement a lui aussi été calibré dans la scène. C'est le cas notamment lors de la numérisation d'une pièce. Le positionnement relatif du robot vu précédemment est important pour éviter des déviations au cours de la trajectoire, mais le positionnement absolu permet à cette trajectoire d'être réalisée à l'emplacement souhaité. Là où le positionnement relatif ne test que le modèle du robot, le positionnement absolu lui vérifie le bon positionnement de l'effecteur dans un repère externe. Dans ce repère externe se trouvent d'autres objets positionnés, avec lesquels le robot va interagir.

Les résultats sur le positionnement absolu de l'effecteur avantagent notre méthode ANCAR optimisée. Comparé à la méthode classique, on obtient en moyenne de bien meilleurs résultats. Même si nous n'utilisons que 1000 points pour estimer les arcs de cercle, notre méthode reste globalement meilleure que la méthode classique.

Avant de comparer le modèle de l'armoire au nôtre, il faut noter que le modèle théorique de l'armoire de commande ne permet de décrire que la géométrie propre au robot. Il n'y a donc pas de modélisation du positionnement du robot dans son espace. Il est cependant possible de définir un outil, mais seulement si cet outil dispose d'une pointe (extrémité d'un forêt ou d'un fer à souder). On ne peut alors pas caractériser un des ensembles de cibles ou même le capteur laser. Pour effectuer les calculs des poses avec le modèle de l'armoire, nous nous basons donc sur le modèle théorique présent dans l'armoire pour la modélisation du robot, et nous utilisons les paramètres de base et d'outil identifiés par notre méthode ANCAR. Avec cette méthodologie, les résultats obtenus avec la méthode ANCAR optimisée sont

meilleurs que ceux obtenus avec le modèle de l'armoire de commande. Cela s'explique en partie par le fait que tous les paramètres géométriques (positionnement robot, outil, et paramètres propres au robot) sont identifiés ensemble avec notre méthode ANCAR optimisée. Leurs valeurs sont calculées ensemble pour donner une représentation du robot. Il est normal dans ce cas que les paramètres de base et d'outil ainsi identifiés fonctionnent mieux avec les paramètres du robot identifiés par la même méthode. Cette homogénéité ne peut donc pas être reproduite avec le modèle de l'armoire, qui doit nécessairement utiliser des paramètres de base et d'outil identifiés à part. On peut conclure que si la méthode d'identification est suffisamment fiable, et c'est le cas ici, les résultats obtenus avec le modèle de l'armoire de commande seront nécessairement moins bons lorsque l'on cherche un positionnement absolu.

2.6.5 Conclusions sur l'étude comparative et la validation des paramètres identifiés

Vu les résultats obtenus lors des différentes études, le modèle géométrique identifié ici peut être jugé satisfaisant afin de calculer des indices de performance du robot dans son espace de travail. Mais il doit également répondre aux besoins de deux autres utilisations : le suivi du robot par le modèle (si pas de tracking externe), et la génération de trajectoires dans le générateur de trajectoires. Dans le cas du suivi du robot, notre modèle doit pouvoir indiquer la pose de l'effecteur, avec une qualité permettant une bonne fusion des lignes laser par le capteur laser plan. Dans le cas de la génération de trajectoire, le modèle géométrique défini dans le chapitre 2 permet de créer les points de cette trajectoire et de la placer dans la scène. Cette trajectoire étant optimisée avec la CAO de la pièce à numériser, un écart engendrerait des défauts sur le nuage de points. L'estimation de l'impact d'un écart sur le nuage de points numérisés est traitée au début du troisième chapitre. Une modélisation plus poussée est envisagée dans ce troisième chapitre, afin de prendre en compte les comportements élastiques du robot et les jeux.

Pour conclure sur le comparatif effectué lors de ces études, la durée d'identification avec la méthode ANCAR optimisée peut être plus courte qu'une méthode classique si on utilise seulement 1000 points par arc. Elle est également plus pratique à mettre en place, car elle ne nécessite que de parcourir des arcs pour chacun des axes et non pas de trouver un certain nombre de configurations avec une étude d'identifiabilité et du conditionnement de la matrice jacobienne. De plus notre méthode ANCAR optimisée donne des résultats près de 10 fois plus fiables. Enfin la non prise en compte des comportements élastiques rend difficile une conclusion sur le positionnement relatif du robot (respect des distances parcourues), et sur la qualité des paramètres propres au robot. Cependant on note un net avantage de notre méthode sur l'estimation des paramètres d'outil et de base lors du positionnement absolu, qui sont essentiels aux interactions entre le robot et la pièce à numériser.

2.7 Conclusion sur la modélisation géométrique

Nous avons donc effectué ici la calibration géométrique d'un robot industriel Kuka servant à la numérisation et au contrôle de pièces sur chaîne de production. Les contraintes étaient de pouvoir faire l'identification des paramètres du modèle rapidement et directement sur la chaîne de production, tout en atteignant une bonne qualité sur les paramètres. Il fallait une méthode pratique, sans contact, et rapide. Nous avons donc exploité la méthode CPA qui correspond à nos attentes, et nous l'avons adaptée avec le moyen de mesure C-Track pour obtenir une méthode plus pratique nécessitant peu d'instrumentation sur le robot et sur sa cellule : la méthode ANCAR optimisée. Les paramètres obtenus avec cette méthode améliorent effectivement le positionnement absolu du robot, notamment grâce aux paramètres de positionnement de l'outil sur le robot et de positionnement du robot dans le repère externe. Cependant on voit que l'on a toujours une erreur de l'ordre du dixième de millimètre pouvant grandement nuire au suivi du robot et à la génération de trajectoires. Nous envisageons donc à présent une modélisation et une calibration des paramètres non géométriques du robot.

Lors d'une étude comparative avec une méthode classique, nous avons pu montrer que notre méthode ANCAR optimisée, légèrement plus longue à mettre en place, est plus adaptable et fournit des valeurs de paramètres menant à une meilleure qualité de positionnement du robot. De plus l'introduction d'une nouvelle formulation pour la méthode ANCAR, utilisant une jacobienne, ouvre de nombreuses possibilités en termes d'identification des paramètres, et pas seulement géométriques. De multiples combinaisons d'informations en entrée de notre résolution sont envisageables, exploitant des mesures d'axes, de points, et potentiellement d'autres formes géométriques. L'identification des paramètres ne se reposera plus seulement sur un ensemble des points bien choisis, mais sur des informations plus variées, amenant à des résultats plus robustes.

Chapitre 3

Identifiabilité sélective des paramètres non géométriques

L'objectif de ce chapitre est de représenter le comportement du robot de manière plus réaliste, en tenant compte de paramètres non géométriques. Nous commençons tout d'abord par exprimer le besoin d'une modélisation plus détaillée vis-à-vis des objectifs que nous nous sommes fixés et notamment : pouvoir numériser sans système de suivi externe (C-Track). Après avoir établi ce besoin, nous choisissons quels paramètres ajouter pour enrichir notre modèle géométrique. Notre choix se porte sur la représentation des déformations élastiques dues à des efforts ainsi que des jeux dans les liaisons. Nous mettons alors en place une modélisation des segments du robot avec des paramètres de raideur basés sur la théorie des poutres, pour les segments respectant cette théorie. Nous définissons également le comportement des articulations du robot avec un modèle original qui nous a été inspiré par la définition normée des erreurs dans une articulation. Nous incluons la modélisation élastique des segments et des articulations dans notre modèle géométrique pour donner un modèle élasto-géométrique complet.

Cependant ce premier modèle très complet, ne tient pas compte des spécificités de notre robot. Une réflexion basée sur la géométrie de notre robot et sur les caractéristiques de l'application de numérisation ciblée, permet de retirer des paramètres inutiles dans notre contexte. Afin d'approfondir cette réduction du modèle complet de base, l'avant dernière section de ce chapitre étudie l'influence des paramètres du modèle sur le positionnement de l'effecteur, au travers notamment de l'identifiabilité. Grâce à cette dernière étude, les paramètres peu influents ou redondants pourront être également retirés du modèle élasto-géométrique afin d'aboutir à un modèle identifiable et pertinent de notre robot.

3.1 Contexte et estimation du besoin d'une modélisation enrichie du robot

Comme expliqué dans le premier chapitre, notre stratégie de génération de trajectoire repose sur la connaissance des performances du robot au travers d'indices. Le calcul de ces indices est basé majoritairement sur la modélisation du robot. D'autre part, dans cette stratégie nous avons également besoin du modèle du robot pour passer de la trajectoire théorique générée à une trajectoire « commande » à fournir à l'armoire de commande du robot. Enfin, lors de la numérisation de la pièce, le suivi de l'effecteur dans un repère externe est effectué par le C-Track ou avec la modélisation du robot ainsi que les coordonnées articulaires. Le modèle du robot permet de situer l'effecteur et sa représentativité va déterminer la qualité du nuage de points résultant. La qualité et le niveau de détail du modèle du robot va donc impacter trois points importants : les indices de performance du robot, la génération de la trajectoire commande et le suivi de l'effecteur pendant la numérisation.

Dans le chapitre 2, la validation des paramètres géométriques identifiés a montré que des erreurs atteignant jusqu'à 0,5mm sur le positionnement absolu de l'effecteur (suivant Y) étaient toujours présentes. Mais peut-on dire pour autant que nous avons besoin d'un modèle plus exact pour le robot ? Afin de répondre à cette question, il faut estimer le niveau de détail de modélisation requis par les trois besoins cités. Cette problématique fait l'objet des trois sous-sections suivantes.

3.1.1 Modélisation et indices de performance

Tout d'abord, nous allons estimer le niveau de détail du modèle nécessaire pour bien exprimer les indices de performance du robot. Pour cela, il faut calculer la variation sur les valeurs des indices, due à une modification de la modélisation. Cette étude doit être menée sur chaque indice de performance utilisé. C'est pourquoi elle a été menée le long du chapitre 4, dans lequel nous faisons le tour des paramètres sélectionnés pour notre générateur de trajectoire. Nous résumons ici cette étude et donnons nos conclusions quant au besoin d'une modélisation détaillée ou non pour le calcul de nos indices de performance.

Il existe deux catégories d'indices. Des indices adimensionnés servant à comparer les performances du robot dans son espace de travail, et des indices donnant des valeurs chiffrées des performances du robot sur des notions comme la vitesse ou la répétabilité. Dans tous les cas, nous déterminons deux jeux de valeurs pour chacun de ces indices, calculés à partir de deux modèles : le modèle géométrique théorique établi dans le chapitre 2, puis ce même modèle avec des paramètres légèrement modifiés. Une comparaison entre les deux jeux de valeurs pour chaque indice permet de quantifier l'impact d'une modification des paramètres du modèle sur l'estimation des valeurs des indices. Afin de bien visualiser cet impact, nous modifions les paramètres de 5mm en moyenne pour les longueurs et de 3° pour les angles. Pour la première catégorie d'indices, utilisée pour des comparaisons, nous avons observé que les valeurs calculées avec le modèle modifié, ont la même répartition dans l'espace de travail du robot que les valeurs calculées avec le modèle théorique. Ceci sera d'ailleurs détaillé et illustré dans le chapitre 4. Ainsi, même si les paramètres sont légèrement modifiés, les études comparatives qui utilisent ces indices ne sont absolument pas affectées. Une amélioration du détail de modélisation n'affectera donc pas les conclusions tirées à partir de ces études et de ces indices.

Pour la seconde catégorie d'indices, les valeurs chiffrées sont importantes, car elles vont servir à valider la trajectoire vis-à-vis de contraintes (en qualité ou vitesse). Les deux indices sélectionnés pour cette catégorie dans le chapitre 4 sont l'indice de vitesse minimale et l'indice de répétabilité maximale. Les modifications sur le modèle théorique donnent en moyenne une différence de 0,09m/s sur les valeurs de l'indice de vitesse et de 0,002mm sur les valeurs de l'indice de répétabilité. Sur les cartographies de ces indices, les plages de valeurs vont de 0 à 4,5m/s pour la vitesse et de 0,12 à 0,26mm pour la répétabilité. Les différences sur les valeurs d'indice sont donc négligeables. Ainsi pour tous les indices étudiés ici, une modification de 5mm en moyenne sur les paramètres de longueur du modèle géométrique théorique, et de 3° sur les angles, a un impact négligeable sur les études réalisées à partir de ces indices.

D'après les résultats de l'identification ANCAR réalisée dans le chapitre 2, les modifications à apporter au modèle géométrique théorique ne sont au maximum que de l'ordre de 1mm sur les paramètres de longueur et de 0,5° sur les paramètres angulaires. Ces valeurs sont bien en dessous des modifications moyennes de 5mm et 3° utilisées précédemment. Ainsi l'utilisation de notre modèle géométrique identifié au chapitre 2 au lieu du modèle théorique, a une influence négligeable sur le calcul d'indices de performance. Le modèle théorique est suffisant pour les différentes études sur les indices du robot. Il est donc inutile de détailler plus encore la modélisation du robot dans le but d'obtenir des indices de performance plus représentatifs.

3.1.2 Modélisation et trajectoire commande

Nous allons à présent estimer le besoin du niveau de détail pour la modélisation du robot vis-à-vis de la trajectoire commande générée. En effet, le modèle du robot est utilisé pour traduire la trajectoire idéale calculée par notre générateur de trajectoire (dans le domaine cartésien), en une trajectoire commande pour le robot (dans le domaine articulaire). L'objectif est que le robot interprète la trajectoire commande, et réalise la trajectoire idéale grâce au modèle qui aura pris en compte le comportement réel du robot avec ses défauts. Cependant, comme nous l'avons vu dans le chapitre 2, notre modèle du robot n'est pas parfait. Des écarts subsistent entre la position souhaitée idéale et la position réellement atteinte par

l'effecteur. Ainsi il existe des écarts entre la trajectoire réellement suivie par le robot et la trajectoire idéale de notre générateur.

La trajectoire idéale obtenue après optimisation est calculée pour donner le meilleur résultat de numérisation. Le générateur prend notamment en compte la distance du capteur laser à la pièce à numériser ainsi que l'orientation du capteur. En effet, les performances du capteur laser Zephyr dépendent de la position de la ligne laser dans la fenêtre de mesure du capteur, et le bruit de numérisation dépend de cette distance et de cette orientation comme le montre les figures 47 et 48 [Mehdi-Souzani et al. 2016]. Ainsi si la trajectoire réellement suivie par le robot diffère de la trajectoire idéale générée, alors le capteur laser ne se trouvera plus dans une configuration idéale par rapport à la pièce, en termes de distance et d'orientation. Le bruit de numérisation sera alors plus grand.

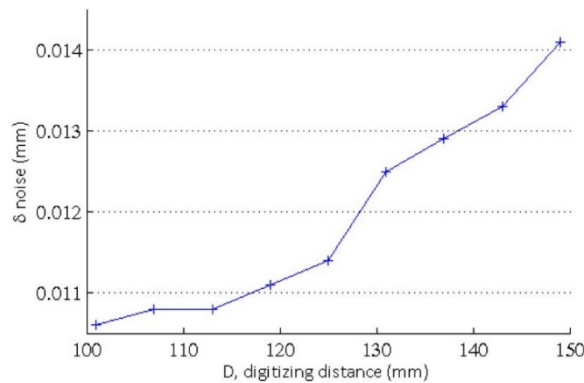


Figure 47 : Influence de la distance de numérisation sur le bruit de numérisation [Mehdi-Souzani et al. 2016]

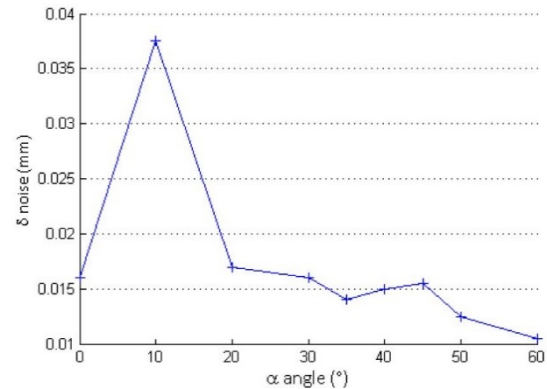


Figure 48 : Influence de l'orientation du capteur sur le bruit de numérisation [Mehdi-Souzani et al. 2016]

La qualification du chapitre 2 donne l'erreur de positionnement absolu du robot utilisant notre modèle (identifié par ANCAR optimisée). D'après cette erreur, en se plaçant dans le cas le moins favorable, la distance entre le capteur et la surface à numériser peut changer de 1,3mm. Toujours en prenant le cas défavorable, l'orientation du capteur peut varier d'un angle de 0,17°. D'après les figures 47 et 48, qualifiant la numérisation d'une surface plane, une modification sur la distance de 1,3mm augmente le bruit de moins de 0,001mm tandis qu'une modification sur l'angle de 0,17° n'a aucun effet notable.

La différence entre la trajectoire réellement suivie par le robot et la trajectoire idéale du générateur n'a donc qu'un impact négligeable sur la qualité de numérisation (en termes de bruit de numérisation). Ainsi, en ce qui concerne la génération de la trajectoire commande, l'erreur due à notre modèle est acceptable et celui-ci n'a pas besoin d'être plus détaillé. Le fait d'utiliser un capteur sans contact permet aux écarts entre le trajet suivi et le trajet souhaité, d'être beaucoup moins pénalisant pour l'opération. Ici, pour notre application de numérisation, nous concluons que notre modèle est suffisant, car cet écart n'a que peu d'influence. Cependant pour d'autres applications, le besoin peut différer. Par exemple, pour une application d'usinage avec robot, la marge d'erreur possible est de l'ordre de la qualité désirée sur la pièce usinée. Pour ce genre d'application, notre niveau de détail sur la modélisation ne conviendrait pas.

3.1.3 Modélisation et suivi de l'effecteur

Lors de l'opération de numérisation, le capteur laser requiert la connaissance de sa propre pose (position et orientation) dans un repère externe, afin de calculer et d'exprimer dans un repère commun les lignes lasers acquises. Au cours de la numérisation, chaque ligne laser est acquise dans un repère interne au capteur, puis réexprimée dans le repère externe commun dans lequel sera formé le nuage de points. La pose peut être fournie par le système de suivi externe (C-Track), mais également par le modèle du robot et ses coordonnées articulaires. Dans le deuxième cas, il faut alors que la modélisation du robot permette de situer précisément la pose réelle de l'effecteur et donc, du capteur. Nous cherchons donc à savoir si

l'exactitude de notre modélisation géométrique actuelle est suffisante pour localiser le capteur et obtenir un nuage de points avec la qualité demandée. En effet le capteur utilise cette pose pour exprimer les lignes laser dans un repère commun. Le bon positionnement des lignes les unes par rapport aux autres, et donc la qualité du nuage, sont alors directement impactés par la qualité d'estimation de cette pose.

Afin d'évaluer la qualité du nuage de points obtenu, nous partons de la justesse et de la répétabilité sur le positionnement de l'effecteur, observées avec notre modèle identifié par la méthode ANCAR optimisée. Nous en déduisons la qualité (simulée) du nuage de points résultant d'une numérisation utilisant notre modèle. Cette qualité est déterminée par le bruit de numérisation sur le nuage et par la justesse du nuage (positionnement du nuage dans le repère externe). Nous prenons l'exemple de la numérisation d'une surface plane étalon.

Pour obtenir l'impact des erreurs de notre modèle sur le nuage de points, nous commençons par fixer la justesse et la répétabilité de mise en position du robot. D'après les résultats du chapitre 2 (tableau 17), nous savons que la justesse sur le positionnement absolu de l'effecteur, dans le cas le plus défavorable, est la suivante : [0,49 -1,29 -0,17] millimètres en position suivant X , Y et Z , [0,122 -0,166 -0,060] degrés en orientation autour de X , Y et Z . Sur cette même série d'essais du chapitre 2, nous avons observé lors des multiples mises en position, la répétabilité du robot. La répétabilité sur la position est de [0,01 0,01 0,02] en millimètres, et la répétabilité sur l'orientation est de [0,068 0,039 0,014] en degrés. Cette justesse et cette répétabilité vont permettre de simuler la numérisation d'une surface plane avec une trajectoire suivie par le robot qui n'est pas la trajectoire théorique. Nous calculons dans un premier temps ce trajet du robot entaché d'erreurs, à partir du trajet théorique et des valeurs de justesse et de répétabilité. Au cours de la simulation de numérisation, nous utilisons ce trajet erroné pour représenter le comportement « réel » du robot, ce qui donne les configurations « réelles » du capteur laser dans le repère cellule (repère externe). Pour chacune de ces configurations, notées $T_{R_{ext}, R_{Créel}}$ (notation homogène), des lignes laser sont acquises dans le repère du capteur $R_{Créel}$. Puis les lignes sont au fur et à mesure réexprimées dans le repère commun externe R_{ext} , en utilisant la configuration du capteur dans ce repère externe au cours du trajet. Cependant, au cours d'une numérisation, la configuration utilisée pour réexprimer ces lignes n'est pas $T_{R_{ext}, R_{Créel}}$, mais plutôt la configuration théorique fournie par le modèle : $T_{R_{ext}, R_{Cth}}$. Or les repères $R_{Créel}$ et R_{Cth} sont bien différents, comme le montre la figure 49 a. La figure 49 b présente l'erreur obtenue quand une ligne laser acquise est exprimée dans le repère externe à l'aide de la configuration théorique. Toutes les lignes alors entachées d'erreurs et regroupées dans le repère externe, donnent un nuage de points qui est moins représentatif de la surface plane.

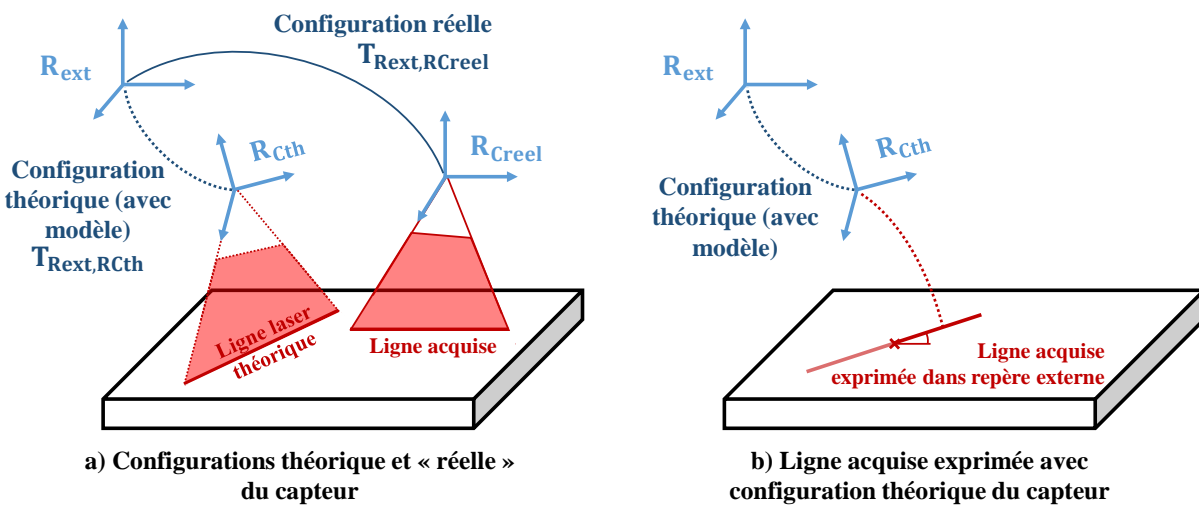


Figure 49 : Erreur due à la différence entre la position réelle du capteur et celle donnée par le modèle

La simulation de la numérisation d'une surface plane est réalisée, en utilisant le protocole décrit précédemment : dans le repère externe la trajectoire idéale est générée, puis la trajectoire « réelle » est

obtenue en ajoutant sur chaque point de passage, l'erreur de justesse de positionnement du robot ainsi qu'une valeur aléatoire tirée d'une loi normale centrée d'écart type la répétabilité sur ce positionnement. L'incertitude sur notre simulation est testée pour 1000 tirages.

Dans notre simulation, chaque point de passage « réel » donne la configuration « réelle » du repère interne au capteur laser $R_{Créel}$, dans le repère externe. La configuration de ce repère en notation homogène est notée : $T_{R_{ext}, R_{Créel}}$. A l'inverse, les points de passage théorique de la trajectoire idéale donnent la configuration théorique $T_{R_{ext}, R_{Cth}}$ du repère du capteur R_{Cth} . L'équation 105 donne le passage entre la configuration « réelle » et théorique du capteur, en utilisant les valeurs de justesse d_ε et de répétabilité σ_ε , dans des matrices homogènes de translation et de rotation suivant les axes X , Y et Z du repère externe. Les valeurs de répétabilité sont utilisées au travers d'une fonction f tirant une valeur aléatoire à l'aide d'une loi normale comme mentionné précédemment.

$$T_{R_{ext}, R_{Créel}} = \text{Trans}(f(\sigma_{\varepsilon,t}) + d_{\varepsilon,t}) T_{R_{ext}, R_{Cth}} \text{Rot}(f(\sigma_{\varepsilon,r}) + d_{\varepsilon,r}) \quad \text{Eq. 105}$$

Dans cette équation $d_{\varepsilon,t}$ correspond à la justesse en position (vecteur de 3 déplacements x, y, z) tandis que $d_{\varepsilon,r}$ correspond à la justesse en orientation (3 angles). De même $f(\sigma_{\varepsilon,t})$ correspond à la répétabilité en positionnement (3 déplacements) et $f(\sigma_{\varepsilon,r})$ à la partie orientation. Le capteur dans la configuration « réelle » $T_{R_{ext}, R_{Créel}}$, génère un plan laser que nous notons $Préel_{R_{ext}}$. Nous ajoutons des limites à ce plan qui correspondent à la visibilité du plan par le capteur, $Préel_{R_{ext}}$ se retrouve alors limité par une forme trapézoïdale. Pour simuler l'acquisition d'une ligne laser lors de la numérisation de la surface plane, il faut faire l'intersection du plan laser et de la surface à numériser. Cette surface $S_{R_{ext}}$ étant elle aussi définie dans le repère externe, nous procédons à l'intersection pour obtenir une ligne laser « réelle » $L_{réel_{R_{ext}}}$ dans le repère externe :

$$Préel_{R_{ext}} \cap S_{R_{ext}} = L_{réel_{R_{ext}}} \quad \text{Eq. 106}$$

Il est ensuite possible d'effectuer cette intersection tout au long du trajet réel pour obtenir toutes les lignes laser. Nous simulons donc ces intersections entre les différentes configurations du plan $Préel_{R_{ext}}$ et la surface plane à numériser. Le nombre d'intersections dépend de la fréquence d'acquisition du capteur et de la vitesse sur le trajet, que nous prenons en compte. Pour chacune des lignes laser calculées, on associe un ensemble de points $\{PtsL_{réel}\}_{R_{ext}}$ qui correspond à la discrétisation de cette ligne. Dans la suite, nous utilisons la notation homogène suivante : $[PtsL_{réel}]_{R_{ext}}$ de 4 lignes et autant de colonnes que de points.

Tous les ensembles de points correspondants aux lignes laser sont alors exprimés dans le repère externe et pourraient être directement regroupés pour donner le nuage de points. Mais il n'y aurait alors pas d'erreur et on retrouverait un nuage de points correspondant à un plan parfait comme la surface. Afin de simuler l'erreur décrite sur la figure 49, que l'on rencontre lors d'une opération de numérisation en contexte réel (la configuration capteur utilisée étant la configuration théorique et non réelle), il faut réexprimer ces ensembles de points en sollicitant la configuration théorique du capteur laser. Pour cela il faut tout d'abord exprimer les ensembles correspondant aux lignes réelles dans le repère interne $R_{Créel}$ du capteur :

$$T_{R_{ext}, R_{Créel}}^{-1} [PtsL_{réel}]_{R_{ext}} = [PtsL_{réel}]_{R_{Créel}} \quad \text{Eq. 107}$$

La formulation de l'équation 107 nous donne les points que le capteur acquiert en situation concrète de numérisation : un ensemble des points acquis dans son repère interne. Ensuite, pour placer ces points dans le repère externe, le logiciel d'acquisition associé au capteur requiert la connaissance de la configuration du capteur dans ce repère externe. Cette configuration est fournie par notre modèle théorique du robot, sans prise en compte de la justesse et de la répétabilité. Les ensembles de points sont donc exprimés dans le repère externe, avec la configuration théorique du repère capteur $T_{R_{ext}, R_{Cth}}$:

$$T_{R_{\text{ext}}, R_{\text{Cth}}} [P_{\text{ts}} L_{\text{réel}}]_{R_{\text{Créal}}} = [P_{\text{ts}} L]_{R_{\text{ext}}} \quad \text{Eq. 108}$$

Comme le montre la figure 49 a, la ligne laser $L_{R_{\text{ext}}}$ correspondante ne représente plus la ligne laser réelle $L_{\text{réel}_{R_{\text{ext}}}}$ dans le repère externe. Cette ligne erronée est donc un mélange de deux repérages différents, et ne représente plus la surface à numériser. Ainsi nous reproduisons l'erreur présente lors de la numérisation : la ligne laser est acquise dans une configuration « réelle » du capteur, puis elle est exprimée dans le repère externe à l'aide de la configuration théorique fournie par le modèle $T_{R_{\text{ext}}, R_{\text{Cth}}}$.

Après avoir simulé cette numérisation d'une surface plane, en prenant soin de solliciter le défaut de recalage des lignes laser, il faut estimer les erreurs sur le nuage de points résultant. Pour qualifier ce nuage de points, nous calculons tout d'abord le bruit de numérisation sur le nuage. La forme théorique que doit avoir ce nuage est un plan parfait. Nous cherchons donc le plan des moindres carrés associé au nuage de points. Nous calculons ensuite la distance entre chaque point du nuage et ce plan associé pour en déduire le bruit de numérisation (équation 39). Dans notre simulation, le bruit obtenu est de 0,024mm. Pour rappel cette valeur est l'écart type sur l'ensemble des distances calculées entre les points et le plan associé. Pour qualifier ce nuage, il faut également vérifier s'il est bien placé dans le repère externe, c'est-à-dire bien placé par rapport à la position réelle de la surface plane qui a été numérisée (justesse). Afin de qualifier ce positionnement du nuage de points, nous calculons simplement la distance entre le plan associé au nuage et la surface (définie dans la simulation). Dans notre simulation, cette distance vaut : 1,29mm. Cette valeur varie très légèrement de 0,01mm d'une extrémité à l'autre de la partie numérisée, car le plan et la surface ne sont pas parallèles.

Nous avons donc l'effet de nos erreurs de modélisation sur la qualité du nuage de points, pour le défaut de recalage des lignes laser. On peut donc à présent conclure sur le besoin de détail dans notre modèle. Le bruit de numérisation peut être jugé acceptable en fonction de la qualité souhaitée sur la surface à numériser. Mais la justesse sur le positionnement du plan numérisé n'est pas satisfaisante. Avec une telle erreur, la numérisation d'une pièce complexe en plusieurs passes avec réorientation du capteur donnerait de très mauvais résultats. Les différentes passes seraient écartées de plus d'un millimètre de leur position réelle donnant des erreurs de cet ordre de grandeur sur les distances vérifiées sur la pièce. L'opération de numérisation est censée qualifier les erreurs de fabrication sur une pièce. Une erreur de l'ordre du millimètre sur le moyen de mesure servant à mesurer la pièce, ne permettrait de mesurer que des pièces aux défauts très grossiers. Ce type de pièces ne serait de toute façon pas mesuré par une opération de numérisation. Il est donc nécessaire d'améliorer la qualité de positionnement de l'effecteur du robot.

Nous avons vu dans le chapitre 2, lors de la validation des paramètres, que le modèle de l'armoire et notre modèle présentent des défauts de justesse du même ordre de grandeur. Et malgré une identification précautionneuse avec notre méthode ANCAR optimisée, il n'est pas possible de descendre en dessous de 0,5mm de défaut de justesse. Un modèle uniquement géométrique ne permettra pas d'atteindre une meilleure qualité de positionnement, car d'autres phénomènes non géométriques entrent en compte et altèrent la qualité de prédiction de ce modèle, tels que les déformations et les jeux. Or nous venons de voir que pour le suivi du robot il est nécessaire d'avoir une meilleure modélisation. C'est pourquoi nous allons chercher à modéliser les paramètres non géométriques.

La mise en évidence d'un défaut inévitable sur le recalage des lignes laser, dû à la différence entre notre modèle du robot et son comportement réel, nous conduit à devoir détailler la modélisation du robot avec des paramètres non géométriques. La prochaine section est donc consacrée à la sélection de paramètres non géométriques pertinents pour notre application.

3.2 Intégration de paramètres non géométriques

Nous avons constaté qu'il est nécessaire d'approfondir la modélisation du robot afin d'atteindre une qualité de positionnement de l'effecteur répondant aux besoins de notre application de numérisation. Nous allons enrichir le modèle géométrique présenté dans le chapitre 2, avec des paramètres non géométriques. Le plus souvent ce sont des paramètres élastiques qui sont ajoutés, c'est pourquoi on appelle modèle élasto-géométrique un modèle géométrique enrichi avec des paramètres non géométriques. Dans la section 1.3.3, nous avons listé les sources d'erreur non géométriques les plus représentées dans la littérature, car les plus influentes. On recense notamment les déformations des différents éléments du robot sous l'effet d'un chargement et du poids propre du robot ainsi que des jeux dans les liaisons. Nous avons également vu différentes méthodes pour prendre en compte des comportements non géométriques directement dans la modélisation géométrique existante ou sous forme de corrections.

Dans cette section nous allons commencer par détailler la méthode que nous souhaitons employer pour intégrer les paramètres non géométriques. Le but est d'avoir une méthode d'intégration donnant un modèle du robot pratique, qui peut facilement évoluer avec nos besoins sur la cellule de numérisation. Cette méthode permet donc d'obtenir une modélisation élasto-géométrique à partir du modèle géométrique du robot et de paramètres non géométriques (figure 50). Les paramètres que nous choisissons d'utiliser sont liés aux sources d'erreur les plus influentes telles que les déformations dues aux efforts et les jeux (section 1.3.3). En effet, même si le capteur laser monté sur le robot est léger (360g), c'est parfois la pièce à numériser beaucoup plus lourde qui est montée sur l'effecteur et qui va déformer élastiquement le robot. De plus, le robot se déforme également sous son propre poids, les segments du robot pouvant avoir une masse d'une centaine de kilogrammes. Ainsi malgré un outil léger monté sur le robot, les déformations doivent être prises en compte dans le modèle.

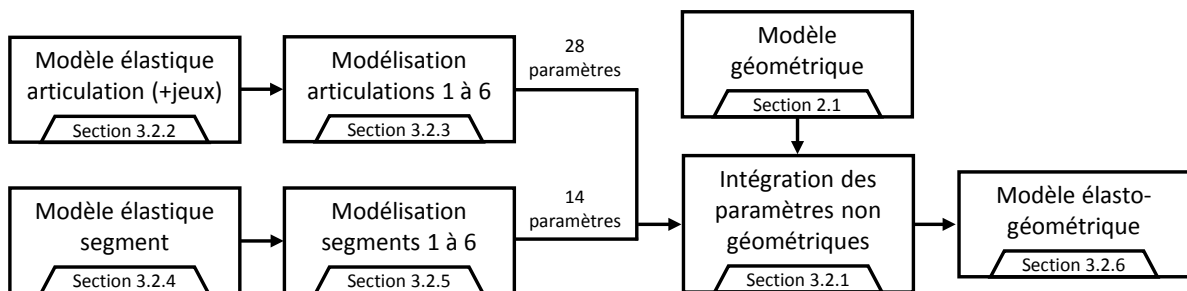


Figure 50 : Etablissement du modèle élasto-géométrique

Il est plus rare de modéliser les déformations dues aux variations de température. En effet, il y a rarement de grandes variations de la température ambiante dans une cellule robotisée. De plus, les modèles en température sont plutôt utilisés pour des structures plus sensibles à de telles variations, des structures non pas en acier ou en fonte mais en aluminium ou en plastique comme certains robots cartésiens effectuant de l'impression 3D. Il n'est pas non plus utile d'utiliser une modélisation dynamique pour le robot. Dans notre application de numérisation, la vitesse de l'effecteur sera limitée par la densité requise sur le nuage de points. Nous faisons donc l'hypothèse que l'utilisation des paramètres de jeux et de déformations sous effort suffira à répondre au besoin de notre cellule en améliorant la justesse de positionnement de l'effecteur. L'influence d'autres sources d'erreur est supposée négligeable.

Nous séparons les paramètres non géométriques choisis en deux catégories : les paramètres propres aux articulations et les paramètres des segments. En effet les déformations sur les articulations et sur les segments ne se manifestent pas de la même façon et doivent être traitées différemment. Nous établissons donc un modèle élastique général d'un segment et celui d'une articulation (avec jeux). Comme le montre la figure 50, nous appliquons ensuite ces modèles généraux à chacun des 6 segments et à chacune des 6

articulations du robot. Les paramètres finalement obtenus sont utilisés pour construire le modèle élasto-géométrique.

Partant de cette modélisation élasto-géométrique détaillée, la section 3.3 de ce chapitre sera consacrée à la sélection d'une méthode d'identification pour ce modèle, et à l'élimination de certains paramètres. Ce tri sur les paramètres est réalisé à partir d'une étude d'identifiabilité des paramètres qui va révéler si certains paramètres ne peuvent pas être identifiés ou s'ils sont trop peu influents dans le modèle. Dans cette section, nous réduirons donc le nombre de paramètres dans le modèle élasto-géométrique pour aboutir à un modèle pertinent et identifiable.

3.2.1 Intégration des paramètres non géométriques au modèle existant

Dans la section 2.1, nous avons déjà établi le modèle géométrique donnant la position et l'orientation d'un outil fixé sur l'effecteur du robot, dans un repère externe. Le repère externe utilisé est le repère du moyen de mesure, c'est-à-dire du C-Track. Nous rappelons ici l'équation 42 régissant ce modèle géométrique :

$$T_{\text{base,cibles}} = T_{\text{base,robot}} T_{\text{robot}} T_{\text{robot,cibles}} \quad \text{Eq. 42}$$

Cette équation peut être détaillée pour faire apparaître les différentes matrices homogènes de passage de translation, de rotation, avec la convention DH modifiée (pour $Z_c \neq Z_6$) :

$$T_{\text{base,cibles}} = \text{Trans}(Z_{CT}, r_{\text{base}}) \text{Rot}(Z_{CT}, \delta\theta_{\text{base}}) T_{d_{\text{base},1}}(d_1, \alpha_1, r_1, \delta\theta_1) T_{d_{\text{base},1}}(\theta_1) \\ \times T_{d_{1,2}} T_{h_{2,3}} T_{d_{3,4}} T_{d_{4,5}} T_{d_{5,6}}(d_6, \alpha_6, \theta_6) T_{d_{5,6}}(r_6, \delta\theta_6) T_{d_{6,c}} \quad \text{Eq. 109}$$

Nous souhaitons établir une modélisation élasto-géométrique qui nous permettra de déterminer directement la pose de l'outil fixé au robot, dans un référentiel externe. Nous ne souhaitons pas avoir un modèle élastique de correction qui corrige la pose calculée par un modèle uniquement géométrique. Il est alors nécessaire d'enrichir la modélisation purement géométrique donnée par l'équation 109 par des paramètres exprimant les déformations et les jeux dans les articulations, ainsi que les déformations des segments, et ce, en utilisant le formalisme VJM présenté dans le premier chapitre (section 1.3.3 et figure 11). La modélisation de notre robot 6 axes dans la cellule, avec paramètres non géométriques s'écrit alors :

$$T_{\text{base,cibles}} = \text{Trans}(Z_{CT}, r_{\text{base}}) \text{Rot}(Z_{CT}, \delta\theta_{\text{base}}) T_{d_{\text{base},1}}(d_1, \alpha_1, r_1) T_{\text{elastseg1}} \\ \times T_{d_{\text{base},1}}(\delta\theta_1 + \theta_1) T_{\text{elastart1}} T_{d_{1,2}}(d_2, \alpha_2, r_2) T_{\text{elastseg2}} T_{d_{1,2}}(\delta\theta_2, \theta_2) \\ \times T_{d_{5,6}}(d_6, \alpha_6, r_6) T_{\text{elastseg6}} T_{d_{5,6}}(\theta_6 + \delta\theta_6) T_{\text{elastart6}} T_{d_{6,c}} T_{\text{elastsegC}} \quad \text{Eq. 110}$$

Dans cette équation les matrices $T_{\text{elastseg}i}$ comprennent les déformations des segments i , et les matrices $T_{\text{elastart}i}$ correspondent aux déformations et jeux dans les articulations i . Les paramètres non géométriques sont compris dans ces matrices de passage homogènes, ce qui est pratique pour calculer rapidement la pose de l'outil (des cibles disposées sur l'outil), ou la pose d'un élément de la chaîne. Il est cependant plus facile de définir les déformations et jeux dans un repère cartésien sous la forme d'un torseur à six composantes : trois translations et trois rotations. Lorsque nous définirons concrètement ces déformations et jeux (sections 3.2.2 et 3.2.4), nous le ferons donc sous forme de torseurs. Pour convertir ces torseurs en matrices homogènes à placer dans la modélisation, nous utilisons les équations 15 et 16 que nous rappelons ici. Pour un torseur τ_{elast} contenant trois translations $[dx \ dy \ dz]$ et trois rotations $[d\alpha \ d\beta \ d\gamma]$, la matrice de passage homogène correspondante T_{elast} peut être trouvée de la manière suivante :

$$\tau_{\text{elast}} = \begin{bmatrix} dx & d\alpha \\ dy & d\beta \\ dz & d\gamma \end{bmatrix} = [T \ R] \quad ; \quad Z_R = \frac{R}{\|R\|} \quad ; \quad \theta = \|R\| \quad \text{Eq. 111}$$

$$T_{\text{elast}} = \begin{bmatrix} Q(Z_R, \theta) & T \\ 0_{3 \times 3} & 1 \end{bmatrix} \quad \text{Eq. 112}$$

Le torseur et la matrice de passage sont donc exprimés dans le même repère. Dans la suite il faudra donc calculer les torseurs dans les repères impliqués par l'ordonnement de l'équation 110.

Une composante principale dans la définition des déformations, et donc dans les matrices de passage élastiques, est la description des efforts sur les différents éléments du robot. Il est donc nécessaire de connaître les torseurs d'efforts s'appliquant sur les articulations et sur les segments. Les efforts se calculent pour une charge appliquée sur l'effecteur, une configuration du robot, et avec les poids propres de chacun des six segments du robot. Ces masses ne peuvent pas être mesurées, elles sont donc six paramètres supplémentaires à identifier dans la modélisation élasto-géométrique (masses supposées centrées sur les segments). La charge appliquée sur l'effecteur correspond, lors d'une numérisation, au poids du capteur laser et de la pièce interface. Durant l'identification des paramètres une autre pièce interface peut être fixée sur l'effecteur, avec différentes masses connues. La charge appliquée sur l'effecteur est donc maîtrisée et ne rajoute pas d'inconnue. La configuration du robot quant à elle, ne peut être fournie que par le modèle géométrique. En effet, les efforts n'étant pas encore calculés, nous n'avons pas les déformations des éléments du robot et donc pas la configuration donnée par le modèle élasto-géométrique. Une fois un premier calcul des efforts et des déformées fait, il sera possible de déterminer une configuration du robot plus proche de la configuration réelle. Il faut ainsi itérer, comme le montre la figure 12 du chapitre 1, jusqu'à converger vers une valeur stable pour la configuration. Cette configuration finale est alors la configuration d'équilibre pour laquelle nous avons les bons efforts et les bonnes déformations. Ce processus itératif fait partie de la modélisation élasto-géométrique, et doit être reproduit pour chaque nouvelle configuration du robot afin de trouver la pose de l'outil.

Pour chaque élément, il faut calculer un torseur d'efforts. Ce torseur se calcule à l'extrémité de chaque articulation ou segment. La position du point B_i désignant l'extrémité d'un segment i , dans le repère du C-Track, se calcule de la manière suivante :

$$T_{\text{base},B_i} = \text{Trans}(Z_{CT}, r_{\text{base}}) \text{Rot}(Z_{CT}, \delta\theta_{\text{base}}) \dots T_{d_{i-1,i}}(d_i, \alpha_i, r_i) T_{\text{elastsegi}} \quad \text{Eq. 113}$$

La dernière colonne de T_{base,B_i} donne la position du point B_i . Si nous ne sommes qu'à la première itération, nous ne pouvons utiliser que le modèle géométrique et les matrices de passage élastiques sont alors des matrices identités. La position du point A_i désignant l'extrémité d'une articulation i , c'est-à-dire le début du segment $i+1$, toujours dans le repère du C-Track, se calcule de la manière suivante :

$$T_{\text{base},A_i} = \text{Trans}(z_{CT}, r_{\text{base}}) \text{Rot}(z_{CT}, \delta\theta_{\text{base}}) \dots T_{d_{i-1,i}}(\delta\theta_i + \theta_i) T_{\text{elastarti}} \quad \text{Eq. 114}$$

Le calcul des torseurs d'efforts se fait de proche en proche en commençant par l'effecteur du robot. La charge appliquée sur l'effecteur étant maîtrisée, le premier torseur (au niveau de l'effecteur) est connu. Ce torseur s'applique au point B_n , extrémité du robot. La figure 51 présente un segment S_i du robot, de longueur d_i , et lié aux articulations L_{i-1} et L_i respectivement en A_i et B_i .

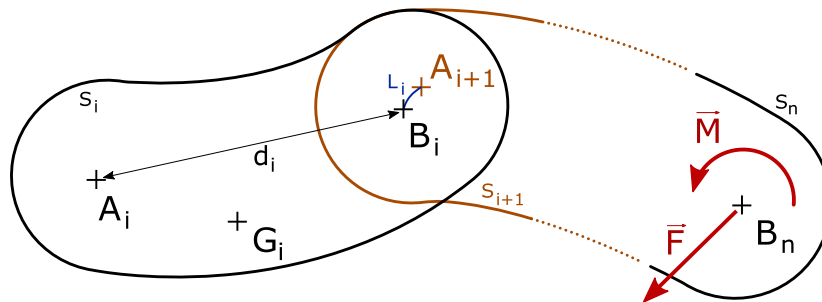


Figure 51 : Description segments et articulations pour détermination d'un torseur d'efforts

La charge appliquée sur l'effecteur crée une force globale F et un moment M en B_n . Chaque segment a également une masse m_j , et un centre de gravité G_j . Soit U , le vecteur unitaire vertical ascendant (pour la gravité), le torseur d'efforts en B_i s'écrit :

$$\left\{ F - \left(\sum_{k=i+1}^n m_k g \right) U \quad M + B_i B_n \wedge F - \sum_{k=i+1}^n m_k g B_i G_k \wedge U \right\}_{B_i} \quad \text{Eq. 115}$$

Dans cette équation, les positions des points B_i et B_n sont calculées avec l'équation 113. Le torseur des efforts sur une articulation en un point A_{i+1} se calcule de manière similaire. On calcule A_{i+1} à l'aide de l'équation 114, puis le point A_{i+1} remplace B_i dans l'équation 115. Afin de simplifier les calculs qui vont suivre, nous utilisons la notation suivante pour les différentes valeurs de ce torseur des efforts :

$$\left\{ \begin{array}{cc} F_x & M_x \\ F_y & M_y \\ F_z & M_z \end{array} \right\}_{B_i} \quad \text{Eq. 116}$$

Nous avons établi ici la structure de notre modélisation élasto-géométrique ainsi que la méthode permettant de calculer une configuration à l'équilibre du robot. Nous pouvons donc déterminer la pose d'un outil fixé sur le robot, dans notre cas la pose du capteur laser. Dans le même temps, nous avons mis en évidence la nécessité d'identifier les masses des différents segments du robot, qui sont alors six paramètres du modèle élasto-géométrique. Il faut à présent mettre en évidence les paramètres présents dans les matrices de passage élastiques, liés aux déformations et jeux dans les articulations et les segments.

3.2.2 Modèle des élasticités et jeux dans une articulation

Nous proposons ici une modélisation des déformations et des jeux des articulations du robot, à l'aide de 8 paramètres. Dans le chapitre 1, nous avons vu que la justesse angulaire est le défaut prépondérant d'une articulation. Mais suivant la configuration de chaque axe, d'autres défauts peuvent devenir prépondérants. Notre modèle de l'articulation prend donc en compte tous les défauts possibles de jeu ou de déformation, et des simplifications pourront être réalisées au cas par cas. Afin de définir les défauts dans l'articulation et aboutir à un torseur de déplacements (en translation et en rotation), nous nous basons sur la norme ISO 230-1 [ISO.230-1 2012]. Nous reprenons le vocabulaire de cette norme déjà présenté dans le premier chapitre et illustré dans l'annexe 1 (définition par rapport à la ligne moyenne de l'axe de l'articulation) :

- dr : Erreur de mouvement radial (perpendiculaire à la ligne moyenne)
- dz : Erreur de mouvement axial (coaxial à la ligne moyenne)
- δr : Erreur de mouvement d'inclinaison (angle par rapport à la ligne moyenne)
- δz : Erreur de positionnement angulaire du mouvement

L'erreur de positionnement angulaire correspond à la justesse angulaire. Cette description des erreurs exploite la symétrie de révolution de l'articulation. Nous définissons donc nos déformations et jeux dans un repère polaire. L'expression de chaque erreur comprend donc une valeur de jeu et une valeur de raideur, ainsi qu'une force F (ou un moment M) :

$$erreur = b_{\text{jeu}} \frac{F}{\|F\|} + \frac{F}{k_{\text{raideur}}} \quad \text{Eq. 117}$$

Dans cette équation, $F/\|F\|$ donne à la fois la direction du jeu et le sens. La force F (ou le moment) est la force résultante sur l'articulation, suivant la direction de l'erreur. Le jeu et la raideur décrivent le comportement (déplacement) de l'articulation suivant cette même direction. La description des erreurs

étant dans un repère polaire, il faut décrire le torseur d'efforts appliqué à l'articulation dans ce repère. L'axe Z_p de ce repère polaire est dirigé par la direction de l'articulation (la ligne moyenne de l'axe). L'axe Z_p correspond donc à l'axe Z_i de l'articulation i donné par le modèle géométrique (avec DH modifié). L'axe r_p lui, est plus difficile à exprimer, car dirigé par la direction des efforts prépondérants sur la liaison. Il sera donc fixé en fonction des différents contextes que nous allons présenter.

Tout d'abord nous projetons le torseur d'efforts appliqué à l'articulation suivant l'axe Z_p , nous obtenons la force F_{z_p} et le moment M_{z_p} . Mais ce torseur d'efforts en cartésien est le résultat du calcul des efforts sur la liaison, qui s'exprime dans le repère de la liaison (repère DH modifié). L'axe Z du repère dans lequel est exprimé le torseur d'efforts correspond donc à Z_p , et il n'est pas nécessaire de projeter les moments et les forces. En utilisant la notation de l'équation 116, on a donc :

$$\mathbf{F}_{z_p} = F_z \quad ; \quad M_{z_p} = M_z \quad \text{Eq. 118}$$

Suivant la direction de l'axe Z_p nous pouvons formuler l'expression de l'erreur de mouvement axial et de l'erreur de positionnement angulaire, respectivement dans les équations 119 et 120.

$$dz = b_z \frac{F_{z_p}}{\|\mathbf{F}_{z_p}\|} + \frac{F_{z_p}}{k_z} \quad \text{Eq. 119}$$

$$\delta z = b_{\theta z} \frac{M_{z_p}}{\|\mathbf{M}_{z_p}\|} + \frac{M_{z_p}}{k_{\theta z}} \quad \text{Eq. 120}$$

Dans ces équations k_z représente la raideur axiale de l'articulation, $k_{\theta z}$ représente la raideur angulaire, b_z représente le jeu axial et $b_{\theta z}$ représente le jeu angulaire. Nous avons donc la description de l'erreur angulaire et de l'erreur axiale pour toute articulation en fonction des efforts sur la liaison et des paramètres de jeu et de raideur.

Nous allons à présent décrire les erreurs de mouvement radial et d'inclinaison. Dans ce but nous schématisons l'articulation comme un cylindre dans un alésage, comme le montre la figure 52 page suivante. Les jeux dans la liaison font que les contacts entre le cylindre et l'alésage dépendent du torseur d'efforts sur l'articulation. Un premier constat qui peut être fait, est que les erreurs de mouvement radial et d'inclinaison ne sont pas indépendantes. Par exemple sur la figure 52, si le cylindre est plaqué contre l'alésage en A et B, par le mouvement radial, alors il n'y a pas d'inclinaison du cylindre. Plusieurs cas de figure vont alors se présenter en fonction du torseur d'efforts. Il peut y avoir seulement une erreur de mouvement radial, seulement une erreur d'inclinaison, ou un mélange des deux minimisant l'effet de chacun. Le jeu radial et le jeu d'inclinaison se partagent l'espace libre entre le cylindre et l'alésage. Nous déclinons les différents cas que l'on peut rencontrer en 4 possibilités.

Nous appelons force radiale et moment radial, la force F_r et le moment M_r obtenus à partir du torseur des efforts sur l'articulation auquel on soustrait la force axiale F_{z_p} et le moment axial M_{z_p} . À partir du torseur d'efforts exprimé dans le repère cartésien, nous avons donc :

$$\mathbf{F}_r = F_x + F_y \quad ; \quad M_r = M_x + M_y \quad \text{Eq. 121}$$

Les différents cas de figure pouvant se présenter sur une articulation sont les suivants :

Cas 1 : dans un premier cas, nous supposons la force radiale exercée sur l'articulation nulle. Le moment radial impose alors la configuration de l'articulation. Sous l'effet de ce moment radial, l'axe de la liaison est donc incliné, ce qui génère le jeu d'inclinaison $b_{\theta r}$ ainsi que la déformation liée à la raideur en inclinaison $k_{\theta r}$. L'erreur d'inclinaison obtenue est décrite par l'équation 122. Sans force radiale, il n'y a pas de déformation liée à la raideur radiale. De plus, l'axe de liaison étant incliné, le jeu d'inclinaison occupe tout l'espace libre entre le cylindre et l'alésage, et le jeu radial n'est pas utilisé. Finalement

l'erreur radiale est nulle et seule compte l'erreur d'inclinaison. L'axe r_p est défini suivant la direction du moment radial M_r , qui dicte la direction de l'erreur.

$$\delta r = b_{\theta_r} \frac{M_r}{\|M_r\|} + \frac{M_r}{k_{\theta_r}} \quad \text{Eq. 122}$$

Cas 2 : on peut retrouver dans un deuxième cas, la configuration inverse. Cette fois-ci nous supposons que le moment radial exercé sur l'articulation est nul. La force radiale impose alors la configuration de l'articulation. Sous l'effet de cette force radiale, la liaison est donc plaquée contre l'alésage, ce qui génère le jeu radial b_r ainsi que la déformation liée à la raideur radiale k_r . Sans moment radial, il n'y a pas de déformation liée à la raideur d'inclinaison. De plus, l'axe de liaison étant plaqué, le jeu d'inclinaison n'est pas utilisé. L'erreur d'inclinaison est nulle et seule compte l'erreur de mouvement radiale. L'axe r_p est défini suivant la direction de la force radiale, qui dicte la direction de l'erreur. L'erreur radiale est décrite par l'équation 123.

$$dr = b_r \frac{F_r}{\|F_r\|} + \frac{F_r}{k_r} \quad \text{Eq. 123}$$

Cas 3 : dans un troisième cas, on retrouve à la fois le moment radial et la force radiale. Mais la particularité de ce cas par rapport à un cas général est que la direction du moment est orthogonale à la direction de la force radiale (en plus d'être orthogonale à la direction de l'articulation). Ce cas particulier est représenté sur la figure 52, où la force radiale F_r et le moment radial M_r sur la liaison s'appliquent au centre de celle-ci (point O). La force et le moment vont donc s'opposer pour placer la liaison dans la configuration du premier cas cité ou du deuxième cas. Sur la figure 52, en fonction du torseur d'efforts, les contacts entre le cylindre et l'alésage seront en B et en A ou A'. La longueur entre A et B est notée L. Pour déterminer la configuration à utiliser, il faut estimer les efforts au niveau de A et A'.

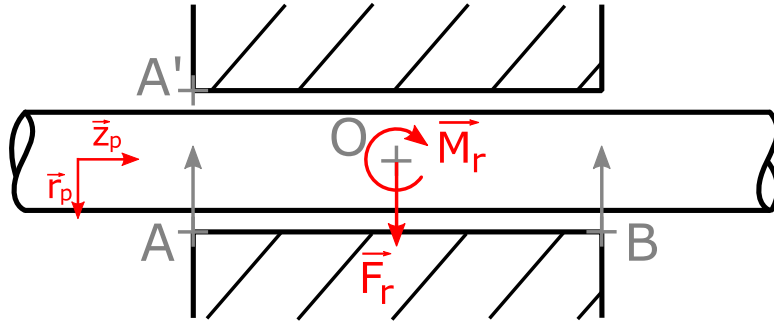


Figure 52 : Schéma transversal d'une articulation sous efforts

Pour déterminer si le contact est en A ou en A', il faut calculer l'équilibre statique de l'articulation. Nous supposons l'axe de la liaison plaquée en A et B, et nous cherchons le torseur des efforts en A. Si la force suivant r_p en A est négative alors cette configuration est la bonne. Si la force suivant r_p en A est positive, alors il faut utiliser la configuration avec axe incliné. Le principe fondamental de la statique en A s'écrit :

$$\begin{cases} F_A + F_B + F_r = 0_{\times 3} \\ \mathbf{AB} \wedge F_B + M_r + \mathbf{AO} \wedge F_r = 0_{\times 3} \end{cases} \quad \text{Eq. 124}$$

La longueur de l'articulation, c'est-à-dire la distance entre A et B est L. Le diamètre de l'articulation, c'est-à-dire la distance de A à A' est D.

$$\mathbf{AB} = LZ_p \quad ; \quad \mathbf{AO} = \frac{L}{2} Z_p - \frac{D}{2} r_p \quad \text{Eq. 125}$$

Nous cherchons à résoudre un problème plan, les moments sont donc tous exprimés suivant la même

direction et les forces suivant Z_p et r_p . Les forces F_A et F_B se décomposent donc en F_{Az} , F_{Ar} , F_{Bz} et F_{Br} qui sont les inconnues que nous cherchons. Avec les équations 124 et 125, nous obtenons donc un système de trois équations exprimé au point A, dans le plan ZR défini par les vecteurs Z_p et r_p :

$$\begin{cases} F_{Az} + F_{Bz} = 0 \\ F_{Ar} + F_{Br} + F_r = 0 \\ L F_{Br} + M_r + \frac{L}{2} F_r = 0 \end{cases} \quad \text{Eq. 126}$$

Afin de déterminer si le contact se fait bien en A ou plutôt en A', c'est-à-dire pour connaître la configuration de l'articulation, nous étudions les forces radiales aux points B et A, F_{Br} et F_{Ar} :

$$\begin{cases} F_{Br} = -\frac{M_r}{L} - \frac{F_r}{2} \\ F_{Ar} = \frac{M_r}{L} - \frac{F_r}{2} \end{cases} \quad \text{Eq. 127}$$

F_{Ar} est la force de l'alésage sur le cylindre. D'après la figure 52 et notre repérage, si F_{Ar} est négatif, alors il y a bien contact en A. S'il y a également contact en B, l'erreur est alors une erreur de mouvement radial. Si F_{Ar} est positif, cela signifie que l'alésage exerce une force d'attraction sur le cylindre en A, ce qui n'est pas possible, et le contact est alors en A'. S'il y a également contact en B, l'erreur est une erreur d'inclinaison. On peut donc en déduire la règle suivante sur l'erreur à adopter en fonction des normes de F_r et de M_r :

$$\begin{cases} \text{Si } \frac{|M_r|}{L} < \frac{|F_r|}{2} \text{ alors : } \textit{erreur} = \mathbf{dr} = b_r \frac{F_r}{\|F_r\|} + \frac{F_r}{k_r} \\ \text{Si } \frac{|M_r|}{L} > \frac{|F_r|}{2} \text{ alors : } \textit{erreur} = \mathbf{\delta r} = b_{\theta r} \frac{M_r}{\|M_r\|} + \frac{M_r}{k_{\theta r}} \end{cases} \quad \text{Eq. 128}$$

Dans cette équation nous prenons des valeurs absolues pour prendre en compte le cas où le moment est dans l'autre sens et où nous cherchons s'il y a contact en B ou en B'. Dans le cas d'une égalité, les effets du moment et de la force sont équivalents. Dans ce cas c'est la dernière configuration observée (avant l'égalité) qui fixe la configuration à adopter. En fonction du cas sélectionné par l'équation 128, on peut là aussi fixer r_p .

Cas 4 : on peut enfin avoir un quatrième cas, où on ne retrouve aucune des situations décrites précédemment. Les directions de F_r et de M_r sont bien orthogonales à l'axe Z_p , mais elles sont indépendantes l'une de l'autre (non orthogonales entre elles). Dans ce cas, le positionnement du cylindre dans l'alésage n'est pas binaire comme dans les raisonnements précédents, le moment et la force participent à ce positionnement. Il y a donc à la fois une erreur d'inclinaison et une erreur de mouvement radial. Le calcul permettant de déterminer la contribution de chacun, le positionnement de l'axe, et enfin les erreurs, est complexe. Il demande de bien connaître les dimensions géométriques de la liaison. Or pour notre modélisation, nous avons effectué une simplification de l'articulation en tant que cylindre dans un alésage, ce qui n'est pas cohérent avec ce besoin de maîtrise des dimensions géométriques de la liaison. Ce quatrième cas demande donc un niveau de modélisation plus complexe que celui que nous mettons en place. Il existe cependant toujours un terme prépondérant entre le moment et la force radiale. Afin de simplifier ce cas, nous allons appliquer ici aussi la stratégie employée pour le troisième cas. Nous utilisons donc l'équation 128 pour déterminer quelle sera l'erreur prise en compte.

Pour terminer, il faut quitter le repère polaire et exprimer les différentes erreurs de l'articulation dans le repère cartésien du torseur des efforts, c'est-à-dire le repère DH de l'articulation. L'axe Z_p étant commun, les erreurs de mouvement axial dz et de justesse angulaire δz ne sont pas à redéfinir. Pour les

erreurs de mouvement radial et d'inclinaison, nous utilisons simplement les égalités des équations 121, 122 et 123, ce qui nous permet d'obtenir :

$$\left\{ \begin{array}{l} dx = b_r \frac{F_x}{F_r} + \frac{F_x}{k_r} \\ dy = b_r \frac{F_y}{F_r} + \frac{F_y}{k_r} \\ dz = \text{signe}(F_z) b_z + \frac{F_z}{k_z} \end{array} \right. \quad \text{Eq. 129}$$

$$\left\{ \begin{array}{l} d\alpha = b_{\theta r} \frac{M_x}{M_r} + \frac{M_x}{k_{\theta r}} \\ d\beta = b_{\theta r} \frac{M_y}{M_r} + \frac{M_y}{k_{\theta r}} \\ d\gamma = \text{signe}(M_z) b_{\theta z} + \frac{M_z}{k_{\theta z}} \end{array} \right. \quad \text{Eq. 130}$$

avec

$$F_r = \sqrt{F_x^2 + F_y^2} \quad ; \quad M_r = \sqrt{M_x^2 + M_y^2} \quad \text{Eq. 131}$$

Les six termes des équations 129 et 130 correspondent aux composantes du torseur élastique des déplacements dus aux jeux et aux déformations dans la liaison (équation 111).

3.2.3 Modélisation des articulations 1 à 6 du robot

Nous appliquons à présent notre modélisation générique des articulations, sur les 6 articulations du robot, en tenant compte des spécificités de chaque liaison. Les différents modèles de chaque articulation nous permettront de lister les paramètres de jeu et de raideur qui interviennent dans notre modèle élasto-géométrique (paramètres des liaisons).

Liaison 1 : la première articulation du robot, entre la base (segment 1) et le segment 2, est verticale. Les efforts appliqués sur l'articulation 1 ne provenant que de l'effet de masses soumises à la gravité (poids du robot + chargement effecteur), le torseur des efforts sur cette articulation peut s'écrire :

$$\left\{ \begin{array}{cc} 0 & M_x \\ 0 & M_y \\ F_z & 0 \end{array} \right\}_{A_1} \quad \text{Eq. 132}$$

La force F_r s'appliquant sur l'articulation 1 est donc nul. Nous nous trouvons alors dans le premier cas d'application dans lequel nous avons l'erreur d'inclinaison, mais pas l'erreur de mouvement radial. De plus le moment autour de l'axe est lui aussi nul. Dans ce cas aucune déformation n'est possible selon l'axe correspondant à la justesse angulaire. Par contre le jeu autour de cet axe subsiste. Sans moment autour de l'axe, ce jeu est déterminé par le sens de rotation de l'articulation 1 c'est-à-dire par le signe de θ_1 . Enfin, étant donné que l'axe de l'articulation 1 ne changera jamais par rapport au sens de la gravité, la force F_z découlant de la gravité aura toujours le même signe. Le jeu axial ne sera jamais sollicité différemment, et il n'est donc ni possible ni utile de l'identifier. Ce jeu axial est alors une distance constante dans le modèle qui sera prise en compte dans la modélisation géométrique. Au final les déplacements dus aux flexibilités et aux jeux dans la première articulation peuvent s'écrire :

$$\left\{ \begin{array}{l} dx = 0 \\ dy = 0 \\ dz = \frac{F_z}{k_{z1}} \end{array} \right. \quad \text{Eq. 133}$$

$$\begin{cases} d\alpha = b_{\theta r1} \frac{M_x}{M_r} + \frac{M_x}{k_{\theta r1}} \\ d\beta = b_{\theta r1} \frac{M_y}{M_r} + \frac{M_y}{k_{\theta r1}} \\ d\gamma = \text{signe}(\dot{\theta}_1) b_{\theta z1} \end{cases} \quad \text{Eq. 134}$$

Liaison 2 : l'axe de la deuxième articulation du robot est horizontale. Cette fois-ci il peut y avoir une force radiale en plus du moment radial (de direction orthogonale donc troisième cas de figure). Il faut donc déterminer lequel est le plus influent sur l'articulation. La figure 53 présente le robot dans une configuration où le moment radial (M_1) sur la liaison 2 est maximal (M_2 est le moment axial).

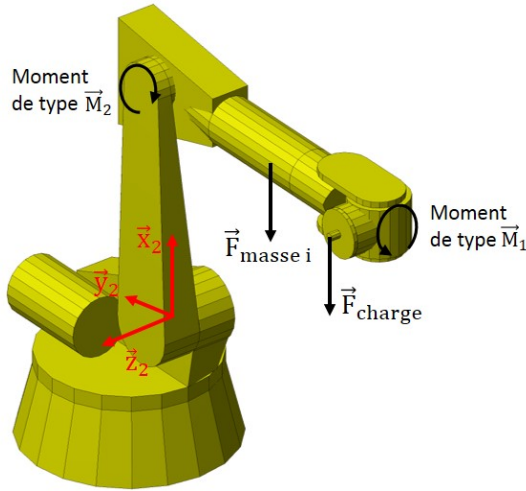


Figure 53 : Efforts en jeu sur l'articulation 2

Sur cette figure on peut voir que le moment le plus important, de type M_2 , est généré par les forces de gravité dues à la charge sur l'effecteur et aux masses des segments i du robot. Les bras de levier sont de plus très importants. Ce moment est autour de l'axe Z_2 de l'articulation 2, et il intervient donc dans l'expression de la justesse angulaire. Par contre le moment de type M_1 , n'est généré que par la charge sur l'effecteur, et le bras de levier est très petit. Ce bras de levier correspond à la longueur du segment 6 soit 230 mm en nominal. Ce faible moment est autour de l'axe Y_2 de la liaison 2, il est donc lié à l'erreur d'inclinaison. Les forces en jeu sont la charge sur l'effecteur ainsi que les poids propres à chaque segment. La force radiale et le moment radial valent alors :

$$F_r = -F_{\text{charge}} - \sum_{i=3}^6 F_{\text{masse } i} \quad ; \quad M_r = -0,230 F_{\text{charge}} \quad \text{Eq. 135}$$

Si l'on pense que le moment est prépondérant et que l'erreur d'inclinaison s'impose, il faut alors vérifier :

$$\frac{|M_r|}{L} > \frac{|F_r|}{2} \quad \rightarrow \quad L < \frac{0,46 F_{\text{charge}}}{F_{\text{charge}} + \sum_{i=3}^6 F_{\text{masse } i}} \quad \text{Eq. 136}$$

La charge sur l'effecteur, s'il porte la pièce à numériser, ne dépassera pas 20 kg. Les masses des segments 6 à 3 cumulés dépassent les 200 kg (masse totale du robot de 1260 kg). La longueur L de la liaison doit donc être au maximum de 42 mm pour espérer observer une erreur d'inclinaison. Or sur la structure du robot, la longueur de l'articulation 2 dépasse les 100 mm. Ainsi sur la liaison 2, c'est l'erreur de mouvement radial qui doit être utilisé et pas l'erreur d'inclinaison. Par ailleurs, la force radiale sur l'articulation générée par la gravité et exprimée dans son propre repère, est toujours suivant la même direction, car l'articulation 2 est toujours horizontale. Le jeu du mouvement radial b_{r2} ne sera donc

jamais sollicité différemment, il est donc égale à une distance constante prise en compte dans le modèle géométrique (comme le jeu axial de la liaison 1). D'autre part, il n'y a pas de force possible selon l'axe de l'articulation Z_2 , il n'y a donc aucune déformation axiale. Sans effort ni mouvement, le jeu axial n'est jamais sollicité et est par conséquent, lui aussi une constante masquée dans le modèle géométrique. Au final, les déplacements dus aux flexibilités et aux jeux dans l'articulation 2 peuvent s'écrire :

$$\begin{cases} dx = F_x/k_{r2} \\ dy = F_y/k_{r2} \\ dz = 0 \\ d\alpha = 0 \\ d\beta = 0 \end{cases} \quad \text{Eq. 137}$$

$$\begin{cases} d\gamma = \text{signe}(M_z) \cdot b_{\theta z2} + M_z/k_{\theta z2} \end{cases} \quad \text{Eq. 138}$$

Liaison 3 : la troisième liaison a les mêmes propriétés que la deuxième liaison (axe horizontal). Les conclusions de l'étude en efforts sont les mêmes ainsi que toutes les remarques faites sur les différentes erreurs. Les déplacements dus aux flexibilités et aux jeux dans l'articulation 3 peuvent donc s'écrire :

$$\begin{cases} dx = F_x/k_{r3} \\ dy = F_y/k_{r3} \\ dz = 0 \\ d\alpha = 0 \\ d\beta = 0 \end{cases} \quad \text{Eq. 139}$$

$$\begin{cases} d\gamma = \text{signe}(M_z) b_{\theta z3} + M_z/k_{\theta z3} \end{cases} \quad \text{Eq. 140}$$

Liaison 4 : sur l'articulation 4, on retrouve également un moment radial et une force radiale, mais de directions non orthogonales (cas 4). Il faut donc déterminer si nous avons une inclinaison ou un mouvement radial. Cette fois-ci la direction de l'axe de l'articulation peut changer par rapport à la verticale qui donne la direction des forces. Nous notons φ l'angle entre l'axe de liaison et la verticale, comme sur la figure 54. Les forces radiales sur la liaison, données par la charge sur l'effecteur et les masses des segments 5 et 6, dépendront donc de cet angle. De même les moments radiaux vont dépendre de l'angle φ . Afin de simplifier l'écriture, les forces dues aux masses des segments ($m_5 + m_6$) sont rassemblées et appliquées en un seul point à une distance d_m de la liaison 4.

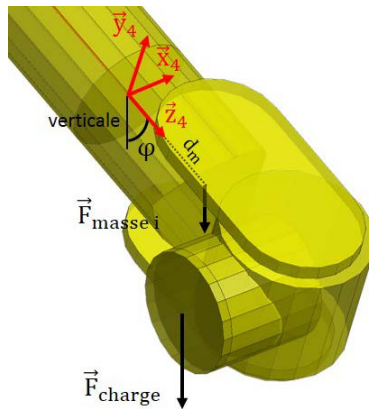


Figure 54 : Efforts en jeu sur l'articulation 4

Les efforts en jeu sont les suivants :

$$|F_r| = \left(F_{\text{charge}} + \sum_{i=5}^6 F_{\text{masse } i} \right) |\sin(\varphi)| \quad \text{Eq. 141}$$

$$|M_r| = 0,230 F_{\text{charge}} |\cos(\varphi)| + d_m |\sin(\varphi)| \sum_{i=5}^6 F_{\text{masse } i} \quad \text{Eq. 142}$$

Il est important de noter une chose : notre modèle géométrique ne permettant pas d'identifier séparément les longueurs des segments 4 et 5, le centre de l'articulation 4 a été placé, dans le modèle géométrique, au niveau du centre de l'articulation 5. Cependant ici, le torseur d'efforts à considérer pour calculer les déformations et les jeux dans la liaison 4 doit être exprimé au niveau de la véritable position de la liaison 4, c'est-à-dire dans le repère présenté sur la figure 54. Car dans les faits, le segment 5 du robot a une longueur et une masse qui ajoutent donc un effort sur l'articulation 4. Ainsi dans le calcul de ce torseur d'efforts, nous utilisons la longueur théorique du segment 5 ainsi que sa masse (qui fait partie des paramètres à identifier). Dans l'hypothèse où c'est l'erreur d'inclinaison qui est plus importante, il faut vérifier (L longueur de la liaison) :

$$\frac{|M_r|}{L} > \frac{|F_r|}{2} \quad \rightarrow \quad d_m > \frac{L}{2} + \frac{F_{charge}}{\sum_{i=5}^6 F_{masse\ i}} \left(\frac{L}{2} - \frac{0,230}{|\tan \varphi|} \right) \quad \text{Eq. 143}$$

Dans le cas particulier où φ est nul, la force radiale est nulle et c'est l'erreur d'inclinaison qui est utilisée. Pour les autres situations, il faut étudier les valeurs que peut prendre le second membre de l'équation 143 en fonction de φ . Il suffit de mener cette étude dans l'intervalle $]0 ; \pi/2[$, les autres cas étant similaires. La fonction tangente étant continue et monotone sur cet intervalle, une étude aux limites permettra de connaître la valeur minimale de d_m .

$$\lim_{\varphi \rightarrow 0} \left(\frac{L}{2} + \frac{F_{charge}}{\sum_{i=5}^6 F_{masse\ i}} \left(\frac{L}{2} - \frac{0,230}{|\tan \varphi|} \right) \right) = -\infty \quad \text{Eq. 144}$$

$$\lim_{\varphi \rightarrow \frac{\pi}{2}} \left(\frac{L}{2} + \frac{F_{charge}}{\sum_{i=5}^6 F_{masse\ i}} \left(\frac{L}{2} - \frac{0,230}{|\tan \varphi|} \right) \right) = \frac{L}{2} + \frac{F_{charge}}{\sum_{i=5}^6 F_{masse\ i}} \frac{L}{2} \quad \text{Eq. 145}$$

La valeur limitante est donc celle donnée par l'équation 145. Dans cette équation on peut légitimement supposer que les masses des segments 5 et 6 sont au moins égales sinon supérieures à la masse de la charge (maximum 20 kg de charge). Ainsi, dans un cas extrême, la limite de l'équation 145 tend vers L, et il faut alors que la distance d_m (point d'application des masses des segments et de la charge) soit supérieure à L (longueur de la liaison). Pour estimer d_m , nous supposons que les centres de gravité des segments 5 et 6 sont au milieu de ces segments. La longueur nominale (documentation technique) du segment 5, différente de la longueur nulle donnée par la modélisation DH, est de 522 mm. La longueur nominale du segment 6 elle, est de 230 mm, mais l'orientation du segment sous l'effet de l'articulation 5 va modifier la contribution de cette longueur sur la distance d_m en fonction de θ_5 . Ainsi, dans le cas le plus contraignant minimisant d_m , où le segment 6 va à l'encontre du segment 5, la distance équivalente d_m vaut 318 mm. La longueur L de la liaison ne dépassant pas les 200 mm, on peut donc conclure que l'inégalité de l'équation 143 est assurée. Ainsi, même dans le cas le moins favorable, le moment radial est plus influent que la force radiale. Nous utilisons donc l'erreur d'inclinaison pour l'articulation 4.

Étant donné que l'orientation de cette articulation est plus libre que les précédentes, aucune autre hypothèse simplificatrice ne peut être émise. À partir du torseur des efforts exprimé dans le repère cartésien de la liaison 4, on peut donc écrire les déplacements dus aux flexibilités et aux jeux dans cette liaison :

$$\begin{cases} dx = 0 \\ dy = 0 \\ dz = \text{signe}(F_z)b_{z4} + F_z k_{z4} \end{cases} \quad \text{Eq. 146}$$

$$\begin{cases} d\alpha = b_{\theta r4} \frac{M_x}{M_r} + M_x k_{\theta r4} \\ d\beta = b_{\theta r4} \frac{M_y}{M_r} + M_y k_{\theta r4} \\ d\gamma = \text{signe}(M_z)b_{\theta z4} + M_z k_{\theta z4} \end{cases} \quad \text{Eq. 147}$$

La matrice homogène $T_{\text{elastart4}}$ associée à ces déplacements est placée après $Td_{3,4}(d_4, \alpha_4, r_4)$ dans la modélisation élasto-géométrique de l'équation 110. Cependant, nous avons vu ici l'importance de la longueur réelle du segment 5, qui a une longueur nulle dans notre modèle géométrique. Cette longueur est importante pour calculer le bon torseur des efforts s'appliquant sur l'articulation, mais aussi pour avoir les bons effets des erreurs d'inclinaison et de justesse angulaire sur l'effecteur (bras de levier). Il faut donc bien placer l'articulation 4 au sein du modèle élasto-géométrique. C'est pourquoi il faut donner aux segments 4 et 5 leurs longueurs réalistes en modifiant le modèle géométrique qui sert de structure pour placer nos matrices élastiques. La modélisation de ces segments devient alors la suivante :

$$T_{3,5} = Td_{3,4}(d_4, \alpha_4, r_4)T_{\text{elastseg4}}Td_{3,4}(\delta\theta_4 + \theta_4)T_{\text{elastart4}} \quad \text{Eq. 148}$$

$$\times \text{Trans}(Z_4, l_5)Td_{4,5}(d_5, \alpha_5, r_5)T_{\text{elastseg5}}Td_{4,5}(\delta\theta_5 + \theta_5)T_{\text{elastart5}}$$

Avec r_4 qui vaut théoriquement 978 mm (au lieu de 1500mm), et l_5 qui vaut 522 mm. Seul l'un de ces deux paramètres reste un paramètre du modèle géométrique, l'autre est considéré comme une constante. En effet il n'est pas possible géométriquement de les différencier, car ils sont alignés, et il sera très difficile de les identifier dans la modélisation élasto-géométrique (car toujours très liés). Nous posons comme paramètre à identifier l_5 , et comme constante r_4 .

Liaison 5 : la direction de l'axe de l'articulation 5 peut se retrouver dans une orientation absolument quelconque, sans aucune restriction contrairement aux articulations précédentes. Il n'est donc pas possible de simplifier les déplacements dus aux flexibilités et aux jeux. À chaque fois que l'on estime la pose de l'effecteur à partir du modèle élasto-géométrique, il sera nécessaire de déterminer le facteur prépondérant sur l'articulation 5, entre le moment radial ou la force radiale. Les déplacements pourront alors être exprimés à partir des équations 129 et 130.

Liaison 6 : le cas de l'articulation 6 ressemble à celui de l'articulation 4. La différence réside dans le fait que la seule force est la charge sur l'effecteur dont le point d'application se trouve quasiment le long de l'axe de la liaison 6. Avec notre montage, il est en effet possible de supposer que la charge fixée sur l'effecteur grâce à nos pièces d'interface, est centrée sur l'effecteur. Il peut y avoir un léger décalage générant un faible bras de levier. Cependant, considérant ce faible bras de levier et la charge maximale dans notre contexte (20 kg), on peut émettre l'hypothèse qu'aucun moment axial ou radial significatif n'est produit (par ce bras de levier). On peut donc conclure que la charge appliquée ne produit aucun moment axial capable de déformer l'articulation de manière visible, suivant la justesse angulaire. Par contre le moment radial est toujours présent, car il est également généré par la distance entre le point d'application de la charge et le centre de la liaison. Soit φ , l'angle entre la verticale et la direction de l'axe de la liaison, comme indiqué sur la figure 55.

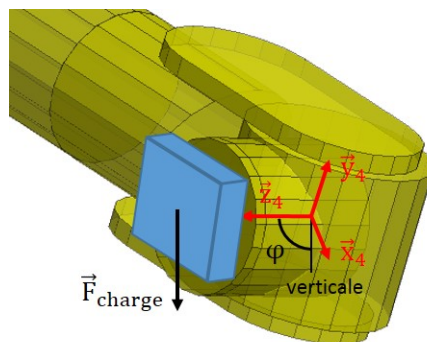


Figure 55 : Efforts en jeu sur l'articulation 6

Nous supposons que la position réaliste du centre de l'articulation 6, et non pas celle donnée par le modèle géométrique (DH), se trouve au centre du segment 6. De plus le centre de gravité de l'ensemble {pièce interface, charge} se situe en moyenne à 50 mm de l'extrémité de l'effecteur du robot. La charge

est donc appliquée à 165 mm (50+115) du centre de liaison, et les efforts de type radial sur l'articulation sont alors les suivants :

$$|F_r| = F_{\text{charge}} |\sin(\varphi)| \quad \text{Eq. 149}$$

$$|M_r| = 0,165 F_{\text{charge}} |\sin(\varphi)| \quad \text{Eq. 150}$$

Dans l'hypothèse où c'est l'erreur d'inclinaison qui est la plus importante, il faut vérifier :

$$\frac{|M_r|}{L} > \frac{|F_r|}{2} \quad \rightarrow \quad L < 0,330 \quad \text{Eq. 151}$$

La longueur totale du segment 6 étant de 230 mm, la longueur L de l'articulation ne peut pas excéder 230 mm, et l'inégalité de l'équation 151 est toujours vérifiée. Ainsi le moment radial est toujours le plus influent, et il faut utiliser l'erreur d'inclinaison dans la modélisation des jeux et déformations de l'articulation 4. Nous avons également dit que le moment axial M_z ne conduit à aucune déformation sur la justesse angulaire. Mais ce moment trop faible pour générer une déformation visible, régit tout de même le comportement du jeu $b_{\theta z6}$ sur la justesse angulaire. Comme ce moment M_z est difficile à déterminer, le sens du jeu l'est également. Cependant l'articulation 6 est la dernière du robot, et aucun bras de levier ne vient exacerber l'effet de l'erreur de justesse angulaire de la liaison 6 sur la pose de l'outil, car celui-ci est centré sur l'effecteur. Le jeu $b_{\theta z6}$ ne modifie donc pas la position de l'outil et n'influence que l'orientation de ce dernier. Nous avons déterminé la valeur de ce jeu à part, avec des essais de positionnement de l'axe 6 du robot. Le jeu $b_{\theta z6}$ vaut $0,061^\circ$, et l'orientation de l'outil est donc modifiée de cette valeur à cause de ce jeu. La faible influence sur l'orientation de l'outil, nous amène à ne pas prendre en compte ce paramètre. Par ailleurs la difficulté à estimer le sens du jeu, qui nécessiterait de nombreuses mesures (plan d'expérience par exemple), allongerait considérablement le temps d'identification du modèle complet.

À partir du torseur des efforts exprimé dans le repère cartésien de la liaison 6, on peut donc écrire les déplacements dus aux flexibilités et aux jeux dans cette liaison :

$$\begin{cases} dx = 0 \\ dy = 0 \\ dz = \text{signe}(F_z) b_{z6} + F_z k_{z6} \end{cases} \quad \text{Eq. 152}$$

$$\begin{cases} d\alpha = b_{\theta r6} \frac{M_x}{M_r} + M_x k_{\theta r6} \\ d\beta = b_{\theta r6} \frac{M_y}{M_r} + M_y k_{\theta r6} \\ d\gamma = 0 \end{cases} \quad \text{Eq. 153}$$

Il est difficile de situer le centre de la liaison 6. Ainsi, comme nous l'avons dit, nous supposons que ce centre se trouve au milieu du segment 6. Comme pour la modélisation de l'articulation 4, il faut modifier la formulation de l'équation 110 afin de placer correctement le centre de la liaison 6. En effet le modèle tel qu'il est placé ce centre en fonction de la longueur de l'outil placé sur l'effecteur. La matrice homogène $T_{\text{elastart6}}$ contenant les déplacements dus aux déformations et jeux dans la liaison 6 est donc comprise dans la modélisation avec les modifications suivantes :

$$\begin{aligned} T_{5,\text{cibles}} &= Td_{5,6}(d_6, \alpha_6, r_6) T_{\text{elastseg6}} Td_{5,6}(\delta\theta_6 + \theta_6) T_{\text{elastart6}} \\ &\times \text{Trans}(\mathbf{Z}_6, l_6) Td_{6,c} T_{\text{elastsegC}} \end{aligned} \quad \text{Eq. 154}$$

Avec d_6 qui vaut théoriquement 115 mm, et l_5 qui vaut 115 mm auxquels on ajoute une longueur donnée par l'outil. Le paramètre d_6 est considéré comme une constante, et le paramètre l_5 reste un paramètre du modèle géométrique.

Synthèse sur toutes les liaisons : nous avons donc défini les différents paramètres élastiques et de jeu apparaissant dans les articulations du robot. Ces paramètres au nombre de 28, sont résumés dans le tableau 18 ci-dessous.

	Liaison 1	Liaison 2	Liaison 3	Liaison 4	Liaison 5	Liaison 6
Raideurs	$k_{z1}, k_{\theta r1}$	$k_{r2}, k_{\theta z2}$	$k_{r3}, k_{\theta z3}$	$k_{z4}, k_{\theta r4}, k_{\theta z4}$	$k_{r5}, k_{z5}, k_{\theta r5}, k_{\theta z5}$	$k_{z6}, k_{\theta r6}$
Jeux	$b_{\theta r1}, b_{\theta z1}$	$b_{\theta z2}$	$b_{\theta z3}$	$b_{z4}, b_{\theta r4}, b_{\theta z4}$	$b_{r5}, b_{z5}, b_{\theta r5}, b_{\theta z5}$	$b_{z6}, b_{\theta r6}$

Tableau 18 : Paramètres élastiques et de jeu des articulations du robot

3.2.4 Modélisation élastique d'un segment

Afin de modéliser les segments du robot, nous nous appuyons sur une théorie des poutres comme celle d'Euler-Bernoulli souvent mise en place en robotique [Khalil et Besnard 2002, Marie et al. 2013]. Les segments du robot sont assimilés à des poutres flexibles, comme sur la figure 56 soumises à un chargement à leur extrémité (point B) ainsi qu'à leur propre poids (p).

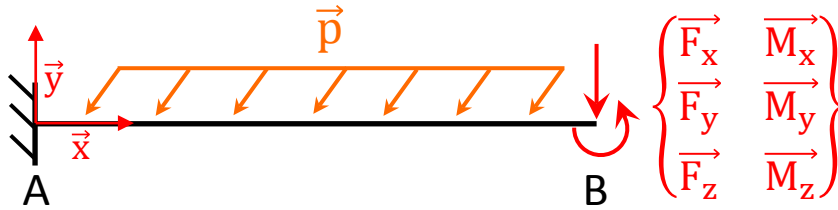


Figure 56 : Schéma d'une poutre chargée

Cette théorie ainsi que cette schématisation permet de calculer analytiquement la déformée de l'extrémité du segment, si ce dernier répond aux hypothèses de la théorie des poutres. Sur notre robot, les longueurs des poutres ne sont pas grandes devant les sections. Nous préférons donc la théorie de Timoshenko, légèrement différente de celle d'Euler-Bernoulli, permettant de prendre en compte les effets de cisaillements dans les sections. Soit un segment de longueur L et de section S , soumis aux efforts $F_x, F_y, F_z, M_x, M_y, M_z$ en son extrémité et à une charge répartie $p = p_x + p_y + p_z$ de direction variable représentant son poids propre. Le torseur des efforts internes à cette poutre (forces et moments) est le suivant :

$$\{F \quad M\} = \begin{Bmatrix} N & Mt \\ T_y & Mf_y \\ T_z & Mf_z \end{Bmatrix} \quad \text{Eq. 155}$$

L'équilibre local d'une poutre droite en fonction de l'abscisse x_a le long de la poutre s'écrit comme suit :

$$\begin{cases} \frac{\partial F}{\partial x_a} + p = 0 \\ \frac{\partial M}{\partial x_a} + x_a \wedge F = 0 \end{cases} \quad \text{Eq. 156}$$

En utilisant les conditions aux limites en effort (en B), la résolution des égalités de l'équation 156 donne les valeurs des efforts internes à la poutre en fonction des efforts appliqués sur la poutre :

$$\begin{cases} N = F_x + p_x(L - x_a) & Mt = M_x \\ T_y = F_y + p_y(L - x_a) & Mf_y = M_y + F_z(x_a - L) - \frac{p_z(x_a - L)^2}{2} \\ T_z = F_z + p_z(L - x_a) & Mf_z = M_z - F_y(x_a - L) + \frac{p_y(x_a - L)^2}{2} \end{cases} \quad \text{Eq. 157}$$

La déformée en un point de la poutre d'abscisse x_a s'écrit :

$$\begin{cases} dx(x_a) & d\alpha(x_a) \\ dy(x_a) & d\beta(x_a) \\ dz(x_a) & d\gamma(x_a) \end{cases} \quad \text{Eq. 158}$$

La théorie de Timoshenko sur les poutres permet d'écrire des égalités liant efforts internes et déformées de la poutre pour une abscisse x_a :

$$\begin{aligned} N &= ES \frac{\partial(dx)}{\partial x_a} & Mt &= GI_0 \frac{\partial(d\alpha)}{\partial x_a} \\ T_y &= GS \left(\frac{\partial(dy)}{\partial x_a} - \gamma \right) & Mf_y &= EI_y \frac{\partial(d\beta)}{\partial x_a} \\ T_z &= GS \left(\frac{\partial(dz)}{\partial x_a} + \beta \right) & Mf_z &= EI_z \frac{\partial(d\gamma)}{\partial x_a} \end{aligned} \quad \text{Eq. 159}$$

Avec I_y, I_z, I_0 les moments quadratiques par rapport à l'axe y et z et le moment quadratique polaire, E et G les modules de Young et de Coulomb, et S l'aire de la section droite de la poutre. Les déformées obtenue à l'extrémité du segment (abscisse $x=L$), en fonction des efforts appliqués sur le segment peuvent finalement être calculées :

$$\begin{cases} dx = \frac{1}{k_h} \left[F_x + p_x \frac{L}{2} \right] \\ dy = \frac{1}{k_c} \left[F_y + p_y \frac{L}{2} \right] + \frac{1}{kf_z} \left[p_y \frac{L^3}{8} + F_y \frac{L^2}{3} + M_z \frac{L}{2} \right] \\ dz = \frac{1}{k_c} \left[F_z + p_z \frac{L}{2} \right] + \frac{1}{kf_y} \left[p_z \frac{L^3}{8} + F_z \frac{L^2}{3} - M_y \frac{L}{2} \right] \end{cases} \quad \text{Eq. 160}$$

$$\begin{cases} d\alpha = \frac{1}{k_t} M_x \\ d\beta = \frac{1}{kf_y} \left[-p_z \frac{L^2}{6} - F_z \frac{L}{2} + M_y \right] \\ d\gamma = \frac{1}{kf_z} \left[p_y \frac{L^2}{6} + F_y \frac{L}{2} + M_z \right] \end{cases} \quad \text{Eq. 161}$$

Dans la formulation des déformées des équations 160 et 161, les 5 paramètres élastiques à identifier, tirés des modules et moments, sont les suivants :

$$k_h = \frac{ES}{L} \quad ; \quad k_t = \frac{GI_0}{L} \quad ; \quad kf_y = \frac{EI_y}{L} \quad ; \quad kf_z = \frac{EI_z}{L} \quad ; \quad k_c = \frac{GS}{L} \quad \text{Eq. 162}$$

Ces paramètres sont les raideurs associées à un segment du robot, en N/m pour k_h et k_c (raideurs en traction/compression et cisaillement), et en N.m/rad pour k_t , kf_y et kf_z (raideurs angulaires en torsion et flexion). Dans les équations 160 et 161 interviennent également la masse du segment m ainsi que sa longueur L . La masse m est un paramètre de plus à identifier, mais la longueur L peut être déduite du modèle géométrique identifié du robot.

Les déformées et efforts de notre modélisation d'un segment, sont données dans un repère propre au segment dont l'axe X correspond à sa direction prépondérante. Or le torseur d'efforts appliqué à l'extrémité du segment est calculé dans le repère défini par notre modèle géométrique (DH modifié). Il faut donc déterminer une matrice de passage pour convertir le torseur d'efforts du repère du modèle géométrique vers le repère du segment. Pour cela, il faut savoir comment sont placés les segments schématisés par des poutres, dans notre modèle géométrique, afin de déduire la longueur et la direction

principale de ces poutres dans les repères du modèle. Mais notre modèle géométrique ne traduit pas le positionnement réel des extrémités des segments, il n'est donc pas possible d'en déduire le placement fidèle des segments sous forme de poutres. Dans la sous-section suivante, nous faisons des hypothèses sur le placement des segments pour notre modélisation élasto-géométrique.

3.2.5 Modélisation élastique des segments 1 à 6 du robot

À partir de notre modélisation des segments, et à l'aide d'hypothèses que nous allons formuler sur chacun des segments, nous définissons ici les paramètres élastiques propres à chaque segment du robot. Nous faisons donc une étude sur chacun de ces segments. Afin d'appuyer les hypothèses à venir, sur la rigidité des segments du robot, nous avons mesuré les déformations des différents segments de ce dernier sous l'effet de différentes charges. Les déformations totales en position sont présentées dans le tableau 19, pour des charges de 12 kg et de 23 kg sur l'effecteur, et pour une configuration bras semi-tendu. Ces déformations sont celles des segments seuls et non un cumul de toute la déformation du robot jusqu'au point mesuré.

Variation de position en mm	Segment 1	Segment 2	Segment 3	Segments 4 et 5	Segment 6
Pour 12 kg	0,018	0,018	0,055	0,077	0,015
Pour 23 kg	0,013	0,029	0,089	0,114	0,026

Tableau 19 : Mesures de la déformée de l'extrémité de chaque segment seul du robot

Pour effectuer ces essais, des cibles ont été apposées sur chaque segment puis suivies durant plusieurs cycles de charge et décharge du robot. Comme il a déjà été mentionné, le segment 5 est de longueur nulle dans la modélisation géométrique, mais est bien présent entre les liaisons 4 et 5, pour une longueur de 522 mm. Cependant l'alignement des segments 4 et 5, et l'agencement de l'articulation 4 entre ces deux segments, fait que nous pourrions les considérer comme une seule poutre déformée. Cette hypothèse est détaillée plus loin, lors de la présentation des paramètres du segment 4-5. À titre d'indication, l'incertitude du moyen de mesure est de l'ordre de 0,013 mm (concaténation des incertitudes du C-Track suivant X , Y et Z).

Segment 1 : le premier segment, tel que défini dans notre modélisation géométrique, permet de lier le repère externe (repère du C-Track) à la première articulation du robot. Outre la modélisation géométrique faite, ce lien passe par le support du C-Track, puis par une dalle de béton support dans notre cellule, et enfin par la base du robot. Le support du C-Track n'est soumis à aucune variation de charge amenant à un comportement de déformation élastique durant l'opération de mesure. La dalle de béton est spécialement conçue pour isoler et supporter des machines lourdes. Elle peut être supposée infiniment rigide au vu des variations d'efforts en jeu lors de changements dans la configuration du robot ou dans la charge sur l'effecteur. Enfin la base du robot est la partie la plus massive du robot, faite d'un bloc.

Tout d'abord, l'hétérogénéité des différents éléments du segment 1, ainsi que leurs formes, fait que le segment 1 ne répond pas aux hypothèses de la théorie des poutres. On ne peut donc pas appliquer notre modélisation des segments sur ce segment 1. Cependant nous pouvons faire l'hypothèse que ce dernier ne participe que très peu aux déformations élastiques du robot. Comme nous l'avons dit, chaque élément dans ce segment est soit indépendant du torseur d'efforts appliqué à l'extrémité du segment 1, soit très rigide. Nous supposons donc que le segment 1 ne participe pas à la déformation élastique du robot. Afin d'appuyer cette hypothèse, nous pouvons utiliser les résultats du tableau 19. On peut voir que les déformées sur le segment 1 sont du même ordre de grandeur que l'incertitude du moyen de mesure. Nous choisissons donc de négliger la faible déformation du segment 1 et ne considérons pas de paramètre élastique pour ce premier segment.

Par ailleurs, étant donné que le segment 1 est le plus proche de la base du robot, le poids de ce segment ne peut influencer que lui-même. Et comme de plus nous considérons que ce segment est rigide, sa masse n'a pas d'utilité. Le paramètre m_1 est donc retiré de la liste des paramètres du modèle élasto-géométrique.

Segment 2 : le second segment est plus susceptible d'être déformé sous l'effet de la charge appliquée sur l'effecteur. Mais les résultats du tableau 19 sur les déformées du segment 2 restent faibles au regard de l'incertitude du moyen de mesure. Étant donné ce constat et la grande différence d'ordre de grandeur entre les déformées de ce segment 2 et des segments suivants, nous avons choisi de considérer ce segment comme rigide. Nous n'employons donc aucun paramètre élastique pour ce deuxième segment. Par ailleurs il est à noter que l'influence de la masse m_2 du segment 2 sur l'articulation 1 reste inchangée pour tout angle θ_1 . En effet la disposition de cette masse vis-à-vis de l'articulation 1 suit le mouvement de l'articulation, et la déformation de l'articulation (dans son propre repère) due à cette masse sera toujours la même. L'effet du paramètre m_2 sur la première articulation ne peut donc pas être dissocié des erreurs géométriques car il est constant, et il sera donc pris en compte au travers des paramètres du modèle géométrique. Étant donné que le paramètre m_2 n'a une influence que sur les segments 1 et 2 que nous considérons comme rigides, ainsi que sur l'articulation 1, ce paramètre est retiré de la liste des paramètres à identifier.

Segment 3 : les déformées du segment 3, données par le tableau 19, sont de l'ordre du dixième de millimètre et participent donc grandement à l'erreur de justesse du robot, qui elle est de l'ordre du millimètre. On ne peut donc pas considérer ce segment comme rigide. Le segment 3 relie les articulations 2 et 3, et la direction principale de ce segment suit la perpendiculaire commune à ces deux articulations. Dans notre modélisation DH modifiée, cette perpendiculaire est dirigée par l'axe X_2 du repère de la liaison 2. La direction principale X du repère propre au segment 3 suit donc l'axe X_2 , et on peut de même faire concorder les axes Y et Z . Le torseur d'efforts sera donc calculé à l'extrémité du segment, dans ce repère. Les déplacements obtenus, dus à la déformation du segment, seront alors ajoutés sous la forme d'une matrice homogène $T_{\text{elastseg3}}$, après la matrice $Th_{2,3}(d_3)$ comme indiqué dans l'équation 110. Étant donné que l'orientation du segment s'inscrit parfaitement dans notre modélisation géométrique, aucune matrice de passage n'est nécessaire. La longueur du segment 3 L_3 , est donc d_3 (longueur de la perpendiculaire commune), soit une longueur théorique égale à 1250 mm.

Une simplification peut être effectuée, car dans le repère de l'articulation 2, et donc dans le repère propre au segment 3, aucune force selon Z_2 (direction de la liaison) n'est possible. Nous avons déjà fait cette remarque pour la simplification de la modélisation de l'articulation 2. Les forces de gravité sont les seules à générer des efforts sur notre robot et ne peuvent donc pas générer de force selon la direction de Z_2 qui est horizontale. Dans la modélisation du segment 3, on peut donc supprimer la force F_z et la charge répartie p_z . Ainsi, les déformations à l'extrémité du segment 3, dans un repère permettant directement de les inclure dans la modélisation élasto-géométrique, sont exprimées comme suit :

$$\begin{cases} dx = \frac{1}{k_{h3}} \left[F_x + p_x \frac{L_3}{2} \right] \\ dy = \frac{1}{k_{c3}} \left[F_y + p_y \frac{L_3}{2} \right] + \frac{1}{kf_{z3}} \left[p_y \frac{L_3^3}{8} + F_y \frac{L_3^2}{3} + M_z \frac{L_3}{2} \right] \\ dz = -\frac{1}{kf_{y3}} M_y \frac{L_3}{2} \end{cases} \quad \text{Eq. 163}$$

$$\begin{cases} d\alpha = \frac{1}{k_{t3}} M_x \\ d\beta = \frac{1}{kf_{y3}} M_y \\ d\gamma = \frac{1}{kf_{z3}} \left[p_y \frac{L_3^2}{6} + F_y \frac{L_3}{2} + M_z \right] \end{cases} \quad \text{Eq. 164}$$

La charge répartie p , calculée à partir de la masse m_3 du segment et de sa longueur L_3 , est bien sûr elle aussi exprimée dans le repère propre au segment, c'est-à-dire le repère de l'articulation 2.

Segment 4-5 : d'après les résultats du tableau 19 sur les déformations de cet ensemble de segments (4 et 5), il apparaît que cet ensemble est la principale source d'erreur de justesse sur l'effecteur, due aux déformations. L'élasticité des segments 4 et 5 est donc importante à modéliser. Dans notre modélisation géométrique du robot utilisant la convention DH modifiée, le segment 5 a une longueur nulle. De plus, le segment 4 de cette modélisation comprend les segments 4 et 5 réels, en plus de la longueur d_4 qui n'est pas dans la direction des segments, comme le montre la figure 57. Les segments 4 et 5 sont séparés par l'articulation 4 qui peut se déformer et dont la coordonnée articulaire peut évoluer. Pour modéliser la déformation réelle du robot, il faudrait donc bien séparer les segments 4 et 5 dans la modélisation géométrique pour pouvoir insérer leurs déformations respectives. Il faudrait également utiliser les longueurs réelles de ces segments. Mais tout comme il n'est pas possible de dissocier les longueurs réelles des segments 4 et 5 dans l'identification géométrique, il est difficile de différencier les élasticités de ces deux segments.

Tout d'abord l'axe de l'articulation 4 est dans l'alignement des segments 4 et 5, et une modification de θ_4 conserve l'alignement des segments. Les faibles déformations de l'articulation 4 permettent par ailleurs de supposer que le torseur d'efforts à l'extrémité du segment 4 est identique au torseur d'efforts du début du segment 5. Nous supposons donc une continuité en effort, comme au sein d'une poutre. Il est donc préférable de réunir les segments 4 et 5 en un seul segment 4-5 pour en calculer les déformées, car il serait difficile de distinguer les paramètres élastiques de chaque segment séparément. Cependant, les raideurs en flexion de ce segment 4-5, correspondant aux paramètres $kf_{y_{45}}$ et $kf_{z_{45}}$, dépendent des raideurs des segments 4 et 5. Or la rotation de θ_4 modifie l'orientation du segment 5 et donc la contribution en raideur de ce segment, c'est-à-dire la contribution de kf_{y_5} et kf_{z_5} sur les paramètres $kf_{y_{45}}$ et $kf_{z_{45}}$. On peut supposer que le segment 5 a une géométrie proche d'une symétrie de révolution, ce qui permet d'écrire l'égalité suivante : $kf_{z_5} = kf_{y_5}$. Avec cette hypothèse, la contribution en raideur du segment 5 est la même pour tout θ_4 , et la rotation autour de l'articulation 4 ne modifie pas les paramètres de raideur en flexion $kf_{y_{45}}$ et $kf_{z_{45}}$ du segment 4-5. Au final, pour la modélisation élastique, on peut rassembler les segments 4 et 5 en un seul segment ce qui correspond à l'ensemble {segment 4 réel ; segment 5 réel} de la figure 57 (donc sans le paramètre d_4). Dans notre modélisation élasto-géométrique ce segment 4-5, est de longueur $L_{45} = 1500$ mm.

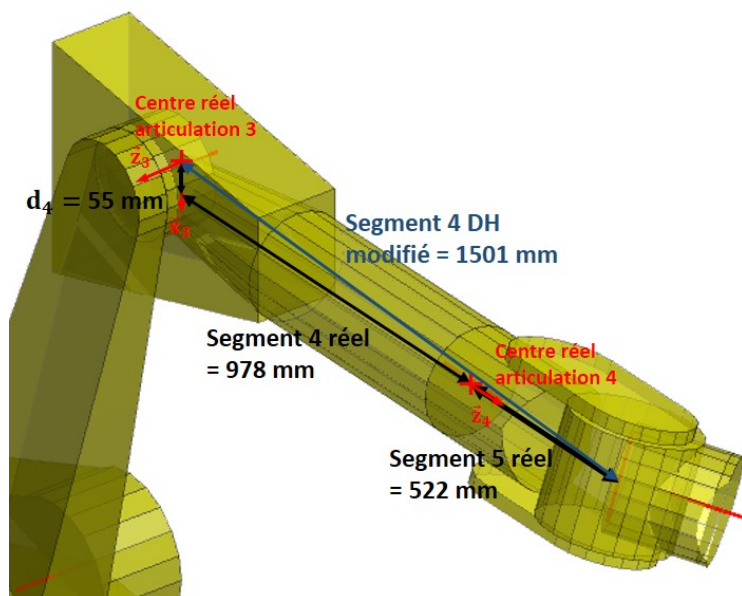


Figure 57 : Segment 4 du modèle géométrique et segments 4 et 5 réels

Pour prendre en compte ces déformations sous cette forme (un seul segment), il faut modifier l'équation 148 qui nous permet de placer l'articulation 4. Nous choisissons de placer la déformation du segment 4-5 sous forme de matrice homogène, à l'extrémité de ce segment, c'est à dire à la place de $T_{\text{elastseg5}}$:

$$\begin{aligned} T_{3,5} &= Td_{3,4}(d_4, \alpha_4, r_4)Td_{3,4}(\delta\theta_4 + \theta_4)T_{\text{elastart4}}\text{Trans}(\mathbf{Z}_4, l_5) \\ &\times T_{\text{elastseg45}}Td_{4,5}(d_5, \alpha_5, r_5)Td_{4,5}(\delta\theta_5 + \theta_5)T_{\text{elastart5}} \end{aligned} \quad \text{Eq. 165}$$

Pour rappel, r_4 vaut théoriquement 978 mm (au lieu de 1500mm), et l_5 vaut 522 mm. Le repère propre du segment 4-5, dans lequel sont exprimées les déformées du segment en son extrémité, est simplement obtenu à partir du repère de l'articulation 3 (représenté sur la figure 57) dont l'axe \mathbf{Y}_3 correspond à la direction prépondérante du segment \mathbf{X} ($\mathbf{Y}_3 = \mathbf{X}$, $\mathbf{X}_3 = \mathbf{Y}$, $\mathbf{Z}_3 = \mathbf{Z}$). L'orientation du repère du segment 4-5 dépend alors des angles θ_1 à θ_3 mais pas de l'angle θ_4 . Il faut donc modifier l'équation 165 dans laquelle les déformations du segment 4-5 $T_{\text{elastseg45}}$ apparaissent après la rotation d'angle θ_4 . Il est possible de décaler la rotation θ_4 jusqu'à l'extrémité du segment 4-5, sans altérer le modèle car $Td_{3,4}(\delta\theta_4 + \theta_4)$ influe très peu sur l'effet de l'élasticité dans l'articulation 4 $T_{\text{elastart4}}$ et n'influe pas sur la translation $\text{Trans}(\mathbf{Z}_4, l_5)$. L'équation 165 s'écrit alors :

$$\begin{aligned} T_{3,5} &= Td_{3,4}(d_4, \alpha_4, r_4)T_{\text{elastart4}}\text{Trans}(\mathbf{Z}_4, l_5)T_{\text{elastseg45}} \\ &\times Td_{3,4}(\delta\theta_4 + \theta_4)Td_{4,5}(d_5, \alpha_5, r_5)Td_{4,5}(\delta\theta_5 + \theta_5)T_{\text{elastart5}} \end{aligned} \quad \text{Eq. 166}$$

Par ailleurs, les forces de gravité ne pourront jamais créer de force suivant l'axe \mathbf{Z} du repère propre au segment (suivant \mathbf{Z}_3), pour les mêmes raisons décrites lors de la simplification de l'expression des déformées sur l'articulation 3. On peut donc simplifier l'expression des déformations en retirant la force F_z et la charge répartie p_z . Par ailleurs, le rassemblement des segments 4 et 5 en un seul segment fait que nous ne considérons plus que 5 paramètres de raideurs $\{k_{h45}; k_{c45}; k_{f_{y45}}; k_{f_{z45}}; k_{t45}\}$ et une seule masse m_{45} pour cet ensemble.

$$\left\{ \begin{aligned} dx &= \frac{1}{k_{h45}} \left[F_x + p_x \frac{L_{45}}{2} \right] \\ dy &= \frac{1}{k_{c45}} \left[F_y + p_y \frac{L_{45}}{2} \right] + \frac{1}{k_{f_{z45}}} \left[p_y \frac{L_{45}^3}{8} + F_y \frac{L_{45}^2}{3} + M_z \frac{L_{45}}{2} \right] \\ dz &= -\frac{1}{k_{f_{y45}}} M_y \frac{L_{45}}{2} \end{aligned} \right. \quad \text{Eq. 167}$$

$$\left\{ \begin{aligned} d\alpha &= \frac{1}{k_{t45}} M_x \\ d\beta &= \frac{1}{k_{f_{y45}}} M_y \\ d\gamma &= \frac{1}{k_{f_{z45}}} \left[p_y \frac{L_{45}^2}{6} + F_y \frac{L_{45}}{2} + M_z \right] \end{aligned} \right. \quad \text{Eq. 168}$$

La charge répartie \mathbf{p} , calculée à partir de la masse m_{45} du segment et de sa longueur L_{45} , est bien sûr elle aussi exprimée dans le repère propre au segment.

Segment 6 : le segment 6 est soumis à la charge de l'effecteur et à son propre poids. Mais comme pour les segments 1 et 2, les déformations sur ce segment données par le tableau 19 sont faibles. Nous choisissons donc de ne pas prendre en compte la déformation de ce segment que nous considérons rigide.

Outil : lors de l'identification des paramètres de ce modèle élasto-géométrique, nous utilisons une interface permettant de charger l'effecteur avec différentes masses. Mais lors d'une numérisation, c'est le capteur laser qui sera fixé à l'effecteur, ou alors la pièce à numériser. Et il n'est pas possible d'attacher des poids à la pièce ou au capteur afin de vérifier les élasticités des ensembles {interface, pièce} ou {interface, capteur}. Cependant, les interfaces permettant de fixer le capteur ou la pièce sont des blocs

d'aluminium pleins épousant les formes de l'effecteur du robot (figure 3). L'interface pour la pièce est une structure rigide la maintenant solidement en position par des crochets de bridage et des sauterelles. Les cibles réfléchissantes étant disposées à la base de la pièce à numériser qui est la partie la plus massive et directement maintenue par l'interface, on peut supposer que leur position n'est pas déplacée par une déformation de l'ensemble {interface, pièce}. L'interface avec le capteur laser, est dotée d'un PH10 intégré de manière permanente à la partie pleine en aluminium. Le PH10 est une interface servant à recevoir la fixation très particulière du capteur laser. L'ensemble constitué avec le capteur laser ne permet aucune déformation susceptible d'altérer la numérisation. Par ailleurs, si la partie en aluminium de l'interface est moins massive que celle de l'interface pour la pièce à numériser, c'est que le capteur est beaucoup moins lourd à porter. On peut donc là aussi supposer que l'ensemble {interface, capteur} ne subit pas de déformation à prendre en compte dans notre modélisation. Nous considérons donc qu'il n'y a aucune élasticité de l'outil et donc aucun paramètre associé.

Synthèse sur tous les segments : nous avons donc défini les différents paramètres élastiques apparaissant dans les segments du robot. Ces paramètres au nombre de 14, sont résumés dans le tableau 20 ci-dessous. Les longueurs ne font pas partie de ce tableau, car ce sont des valeurs tirées du modèle géométrique et donc de paramètres géométriques. Il ne faut pas oublier la masse du segment 6 m_6 , ainsi que la masse du segment 5 m_5 qui doit être identifiée en plus de m_{45} car elle influe sur les déformations de l'articulation 4.

	Segment 3	Segment 4-5	Segment 5	Segment 6
Raideurs	$k_{h3}, k_{c3}, k_{f_{y3}}, k_{f_{z3}}, k_{t3}$	$k_{h45}, k_{c45}, k_{f_{y45}}, k_{f_{z45}}, k_{t45}$	-	-
Masses	m_3	m_{45}	m_5	m_6

Tableau 20 : Paramètres non géométriques des segments du robot

3.2.6 Modèle élasto-géométrique complet du robot

Notre modélisation élasto-géométrique comporte donc 14 paramètres non géométriques pour les segments (tableau 20), ainsi que 28 autres paramètres pour les articulations (tableau 18), soit un total de 42 paramètres non géométriques à identifier. La formulation complète homogène contenant tous ces paramètres est donnée dans l'équation suivante :

$$\begin{aligned}
 T_{\text{base,cibles}} = & \text{Trans}(\mathbf{Z}_{CT}, r_{\text{base}}) \text{Rot}(\mathbf{Z}_{CT}, \delta\theta_{\text{base}}) T_{d_{\text{base},1}} T_{\text{elastart1}} \\
 & \times T_{d_{1,2}} T_{\text{elastart2}} T_{h_{2,3}}(d_3) T_{\text{elastseg3}} T_{h_{2,3}}(\alpha_6, \beta_6, \theta_6 + \delta\theta_6) T_{\text{elastart3}} \\
 & \times T_{d_{3,4}}(d_4, \alpha_4, r_4) T_{\text{elastart4}} \text{Trans}(\mathbf{Z}_4, l_5) T_{\text{elastseg45}} T_{d_{3,4}}(\delta\theta_4 + \theta_4) \\
 & \times T_{d_{4,5}} T_{\text{elastart5}} T_{d_{5,6}}(d_6, \alpha_6, r_6) T_{\text{elastseg6}} T_{d_{5,6}}(\theta_6 + \delta\theta_6) T_{\text{elastart6}} \\
 & \times \text{Trans}(\mathbf{Z}_6, l_6) T_{d_{6,c}} T_{\text{elastsegC}}
 \end{aligned} \tag{Eq. 169}$$

Dans la section 3.3, nous allons calculer la jacobienne et estimer l'identifiabilité des paramètres, ce qui va nous permettre d'en éliminer certains jugés peu ou non influents. Nous allons donc encore simplifier le modèle pour ne conserver que des paramètres pertinents.

3.3 Identifiabilité des paramètres et simplification

Une étape d'identification va permettre à présent, d'estimer les valeurs des paramètres élastiques de notre modèle élasto-géométrique, les plus proches possibles des valeurs réelles. Dans ce but nous allons choisir une méthode d'identification pour ces paramètres et vérifier qu'elle permet bien l'identification de tous les paramètres. C'est pourquoi après avoir défini une méthode d'identification, nous calculerons la jacobienne qui lui est associée afin d'étudier l'identifiabilité des paramètres. Nous pourrions alors observer l'influence des paramètres sur la pose du robot et ainsi retirer les paramètres peu ou non influents vis-à-vis de nos besoins sur l'opération de numérisation.

3.3.1 Méthode d'identification pour paramètres élastiques

Nous avons déjà discuté des différentes méthodes d'identification pouvant être mises en place dans la section 1.3.6. Nous avons également déjà mis en place une de ces méthodes dite « classique » lors de l'identification des paramètres géométriques, ainsi qu'une méthode que nous avons nous-mêmes développée. De la même façon, il va falloir ici sélectionner une méthode d'identification des paramètres qui se décompose en deux parties : une méthode d'acquisition de mesures puis une méthode de résolution.

Comme nous l'avons déjà mentionné, nous cherchons une méthode de mesure qui soit pratique et facile à mettre en place. C'est pourquoi, comme pour l'identification des paramètres géométriques, nous nous tournons vers une méthode de mesure sans contact. Le plus simple est d'utiliser la même méthode classique que celle mise en place dans la section 2.2. Cette méthode peut être employée pour l'identification de tout type de paramètre : elle est utilisée à la fois pour des paramètres géométriques et pour des paramètres élastiques [Khalil et Besnard 2002, Olabi 2011, Yin et Gao 2012]. La mesure de mises en position de l'effecteur du robot dans différentes configurations et sous différentes charges va permettre l'acquisition de données pour la phase de résolution. La variété des configurations et masses choisies devra solliciter les différents paramètres à identifier. Nous ne cherchons pas à mettre en place une adaptation de notre méthode ANCAR pour des paramètres élastiques. Le développement d'une telle adaptation requiert des travaux sur des études de formes complexes qui seront abordés en perspectives.

Ainsi la méthode que nous souhaitons mettre en place nécessite une sélection de configurations et de charges permettant de solliciter tous les paramètres élastiques que nous voulons identifier. Étudier le comportement élastique du robot uniquement sous son propre poids n'est pas évident. Cela demande une sélection de configurations bien précises pour générer différents torseurs d'efforts sollicitant les paramètres élastiques. De plus certains paramètres élastiques ne peuvent tout simplement pas être sollicités par le seul poids propre du robot. Il est donc plus facile d'étudier les déformations en ajoutant une charge sur l'effecteur comme sur la figure 58. Appliquer différentes charges mettra facilement en évidence le comportement linéaire des déformations élastiques. De plus comme nous l'avons déjà exprimé dans la simplification du modèle de l'articulation 6, les outils fixés sur le robot pour la numérisation génèrent des moments négligeables. Il n'est donc pas utile de déporter la charge que nous allons appliquer à l'effecteur, car cette sollicitation n'interviendra pas lors d'une opération de numérisation. Nous choisissons comme charges à appliquer, les bornes minimale et maximale possibles pour notre application, ainsi qu'une valeur intermédiaire : {0kg, 12kg, 23kg}.

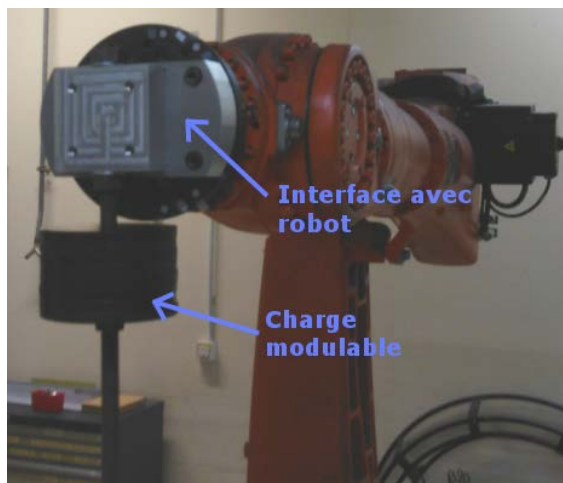


Figure 58 : Interface pour chargement du robot

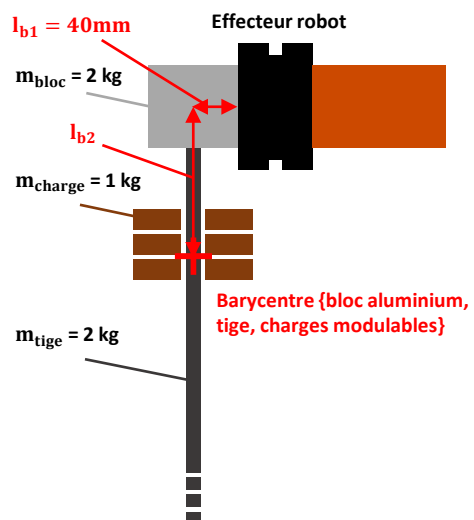


Figure 59 : Interface de chargement

L'interface de chargement permettant de générer différents efforts sur l'effecteur est présentée sur la figure 59. La position du barycentre de l'ensemble {bloc aluminium, tige, charges modulables} est calculée pour pouvoir exprimer le torseur d'efforts s'appliquant sur l'effecteur facilement, quelle que soit la configuration. Cette position l_{b2} est calculée pour un nombre n de charges fixées sur la tige à des positions $l_{chargei}$ (l_{tige} est la demi-longueur de la tige).

$$l_{b2} = \frac{l_{tige}m_{tige} + \sum_{i=1}^n l_{chargei}m_{chargei}}{m_{bloc} + m_{tige} + n m_{chargei}} \quad \text{Eq. 170}$$

Dans cette équation les masses $m_{chargei}$ sont précisées pour chaque charge, car leur masse varie légèrement autour de 1kg. Les longueurs l_{b1} et l_{b2} permettent de calculer le vecteur EB (Effecteur-Barycentre) pour déterminer le torseur des efforts s'appliquant sur l'effecteur :

$$\begin{cases} \mathbf{F} = (m_{bloc} + m_{tige} + n m_{charge})\mathbf{g} \\ \mathbf{M} = \mathbf{EB} \wedge \mathbf{F} \end{cases} \quad \text{Eq. 171}$$

Afin de rester cohérent avec les hypothèses faites sur l'articulation 6, il est nécessaire de rechercher des configurations de mesure qui ne génèrent pas de moment axial avec cette interface de chargement. En effet nos ensembles {interface, pièce} ou {interface, capteur} étaient centrés sur l'effecteur, mais ce n'est plus le cas avec cette interface de chargement. La tige de cette interface devra donc toujours se trouver dans le plan contenant la verticale et l'axe de l'articulation 6 afin de ne générer aucun moment axial.

Il est ensuite possible d'effectuer une optimisation à l'aide de la jacobienne de notre modèle, afin de rechercher les meilleures configurations pour l'identification des paramètres (section 1.3.6). Mais nous ne cherchons pas pour l'instant une optimisation sur ces configurations. Nous allons avant tout répartir les configurations de mesure dans l'espace visible par le système de mesure externe utilisé pour l'identification. Cet instrument de mesure (le C-Track) possède un champ de vision limité qui correspond à la zone de travail où la numérisation sera effectuée. Dans cette zone il faut s'efforcer de solliciter tous les paramètres élastiques mais aussi en particulier les jeux dans les articulations. Notre choix de configurations prend en compte ce besoin, mais aussi les limitations du champ de vision ainsi qu'une cohérence avec l'opération de mesure qui sera réalisée. Par exemple la numérisation d'une pièce ne va jamais requérir de retourner le robot à 180° autour de l'axe 1 comme sur la figure 60 figure 61. Cette configuration n'est donc pas considérée comme pertinente pour le contexte de notre application de numérisation.

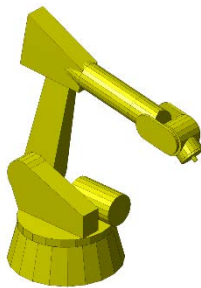


Figure 60 : Retournement à 180° de l'axe 1 et inversion du jeu $b_{\theta r1}$

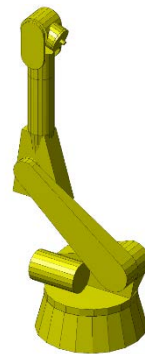


Figure 61 : Inversion du jeu $b_{\theta r1}$ dans l'articulation 1

Dans la partie concernant la modélisation des articulations, la structure même du robot nous a amené à faire des simplifications de paramètres. Ici, le contexte d'utilisation du robot (champ de vision, cohérence sur numérisation), va permettre de simplifier encore la formulation du modèle. Comme évoqué précédemment, retourner le robot autour du premier axe comme sur la figure 60 n'a pas d'intérêt

lors d'une opération de numérisation. Le robot ne sera également jamais sollicité dans la configuration de la figure 61, où l'effecteur se retrouve derrière l'axe 1. Or ce sont les deux configurations modifiant le signe du jeu b_{0r1} dans la liaison 1. Donc le jeu d'inclinaison de l'axe 1 b_{0r1} ne sera jamais inversé, sa valeur est alors prise en compte en tant que constante au travers des paramètres du modèle géométrique. Avec le même raisonnement et pour des raisons similaires, il est possible d'éliminer le jeu b_{0z2} .

Avec ces deux paramètres en moins, le modèle élasto-géométrique ne comporte plus que 40 paramètres à identifier (au lieu de 42). Étant donné que la mesure de la position de l'effecteur donne trois informations (x, y et z), il faut au minimum 14 mesures avec des configurations et charges différentes (soit 42 informations). Il est à noter que plus de 14 mesures sont effectuées afin d'avoir une meilleure robustesse des résultats obtenus. Il est également possible de mesurer l'orientation de l'effecteur (3 informations de plus), mais nous préférons utiliser uniquement la position de ce dernier dont la mesure est plus fiable avec le C-Track. Nous choisissons au total 10 configurations sollicitant tous les jeux ainsi que 3 charges différentes (0, 12 et 23kg) ce qui fait au total 30 mesures.

Une fois les mesures acquises, vient l'étape de résolution permettant de déduire les paramètres élastiques à identifier. La résolution mise en place peut être exactement la même que celle utilisée dans la section 2.2 lors de l'identification des paramètres géométriques avec une méthode classique. On reprend alors l'équation 24 où dP correspond à l'écart entre la position mesurée et la position théorique donnée par notre modèle élasto-géométrique. Il faut également recalculer la matrice jacobienne $J_{tot,w}$, par rapport aux paramètres w_i qui nous intéressent (paramètres élastiques). Il faut de plus une première estimation erronée des paramètres élastiques. La résolution permettra ensuite de calculer le vecteur dw de corrections à apporter à cette première estimation. L'étape de résolution est répétée plusieurs fois, en modifiant à chaque fois les paramètres élastiques avec leurs nouvelles valeurs identifiées, jusqu'à convergence de ces valeurs.

Après l'étape de mesure et celle de résolution, on obtient des valeurs pour les paramètres élastiques du robot. Cependant cette résolution est effectuée avec les valeurs des paramètres géométriques déjà identifiés, qui sont nécessaires pour connaître la position théorique du robot. Or ces paramètres géométriques ont été identifiés sans prendre en compte d'élasticité dans le comportement du robot, ce qui nous a d'ailleurs conduit à observer d'importantes erreurs de justesse. Ces paramètres géométriques ne sont donc pas corrects. Maintenant que nous disposons de paramètres élastiques, il faut à nouveau effectuer l'identification géométrique en tenant compte des élasticités et des jeux. Les premiers paramètres géométriques identifiés sont des paramètres initiaux au problème présenté sur la figure 62. Ensuite plusieurs itérations successives d'identification des paramètres élastiques et géométriques sont réalisées jusqu'à convergence de ces paramètres.

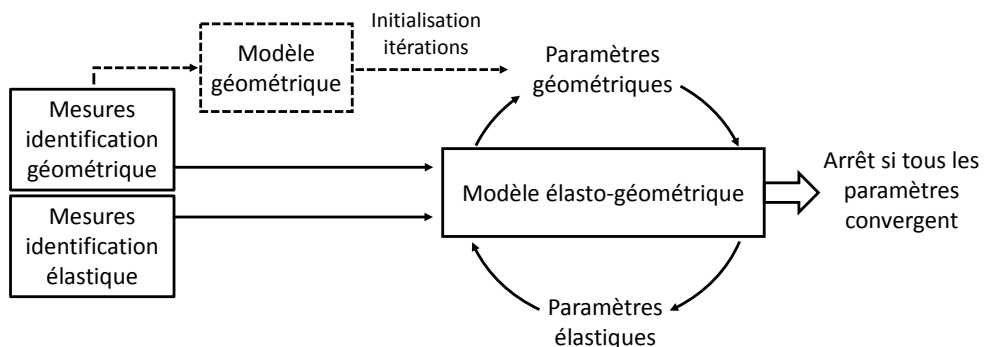


Figure 62 : Itérations pour l'identification des paramètres géométriques et élastiques

On a alors finalement une méthode permettant d'identifier les paramètres élastiques de notre modèle élasto-géométrique, mais permettant également de recalculer les paramètres géométriques pour qu'ils soient cohérents avec ce nouveau modèle plus détaillé, et ce, sans nouvelle campagne de mesure. Avant

de pouvoir mettre en œuvre cette méthode, nous avons vu que sa résolution nécessite le calcul de la jacobienne de notre modèle, à partir des paramètres élastiques. Avant de mettre en place l'identification, il faut également vérifier si cette méthode permet bien l'identification de tous les paramètres élastiques, avec une étude d'identifiabilité. L'étude d'identifiabilité requiert aussi l'expression de la matrice jacobienne de notre modèle élasto-géométrique, à partir des paramètres élastiques. Nous allons donc décrire comment obtenir cette matrice.

3.3.2 Calcul de la jacobienne du modèle élasto-géométrique

La jacobienne des modèles de comportement de robot a toujours été très utilisée. Dans notre cas son calcul est primordial afin d'une part de déterminer l'identifiabilité des paramètres, et d'autre part de résoudre l'identification des paramètres. Nous calculons donc la matrice jacobienne élastique J_{welast} associée à notre modèle élasto-géométrique, ayant autant de colonnes que de paramètres élastiques (donc 40), et 6 lignes (position et orientation). Cette matrice pourra être réduite à 3 lignes seulement si l'on utilise que la position du capteur lors de l'identification. La jacobienne J_{welast} étant calculée pour un ensemble {configuration du robot + charge}, il faudra ensuite déduire la jacobienne totale élastique $J_{\text{tot}_{\text{welast}}}$ pour les m ensembles utilisés lors de l'identification des paramètres tel que :

$$J_{\text{tot}_{\text{welast}}} = [J1_{\text{welast}}, J2_{\text{welast}}, \dots, Jm_{\text{welast}}] \quad \text{Eq. 172}$$

Une matrice jacobienne s'obtient par dérivation du modèle qui lui est associé, dans notre cas le modèle élasto-géométrique. Cependant, la dérivation ne peut être appliquée que sur des expressions continues, et l'intégration de jeux dans notre modèle rend son comportement discontinu. Afin de pouvoir tout de même calculer la jacobienne, il faut donc éliminer les discontinuités, c'est-à-dire les termes calculant les signes des efforts. Il faut savoir qu'une jacobienne se calcule pour une configuration et une charge donnée, car elle lie les paramètres à la mesure d'une position statique de l'effecteur. Or pour une configuration et une charge donnée, il est possible de connaître le sens des jeux, sachant que nous avons choisi des configurations assez éloignées des inversions de jeu. Ainsi pour chaque matrice jacobienne associée à une configuration et une charge, nous calculons avant le sens des jeux, afin d'éliminer les termes discontinus de recherche de signe. Chaque jeu est alors doté du signe qui lui correspond pour la configuration donnée.

Avec cette formulation continue, il faut à présent calculer la matrice jacobienne. Comme nous l'avons vu dans les sections 2.2 et 2.5 du chapitre 2, ce calcul peut être réalisé colonne par colonne pour plus de simplicité. Nous allons reproduire ce même raisonnement avec chacun des paramètres élastiques, pour pouvoir ensuite reconstituer la matrice jacobienne. Mais le calcul sera différent, car les paramètres élastiques ne sont pas directement des paramètres de translations ou de rotations, et on ne peut pas appliquer directement les relations cinématiques vues dans le chapitre 2. Pour obtenir les colonnes de la jacobienne, il faudra tout d'abord dériver le modèle élasto-géométrique selon les paramètres élastiques. Lors de la dérivation du modèle, ces paramètres peuvent être classés en trois catégories : les paramètres de jeux, les paramètres de masse, et les paramètres de raideurs. Afin de simplifier les dérivations à effectuer, nous préférons travailler sur la souplesse plutôt que sur la raideur, car les paramètres de raideur sont toujours au dénominateur dans le modèle. La souplesse est l'inverse de la raideur, nous définissons donc le changement de variable suivant pour chaque paramètre de raideur k ($k_h, k_c, k_f, \text{etc}$) :

$$s = \frac{1}{k} \quad \text{Eq. 173}$$

La jacobienne sera donc calculée avec des paramètres de souplesse s ($s_h, s_c, s_f, \text{etc}$), mais toute conclusion tirée de cette jacobienne pourra être traduite sur nos paramètres de raideur par inversion. Nous exprimons donc la formulation des colonnes de la jacobienne pour chaque type de paramètre non

géométrique ce qui nous donne un lot d'expressions génériques comme sur la figure 63. Puis nous appliquons ces expressions génériques aux 40 paramètres non géométriques pour obtenir la jacobienne élastique J_{welast} . Enfin nous calculons cette jacobienne pour les m ensembles {configuration, charge} afin d'en déduire $J_{\text{totwelast}}$.

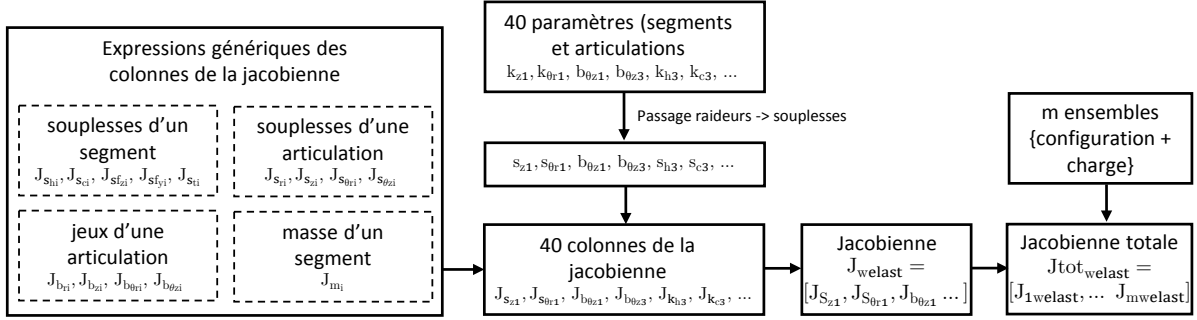


Figure 63 : Etablissement de la jacobienne totale élastique

Expression des colonnes associées aux souplesses d'un segment

Le calcul des colonnes de la jacobienne, utilisant des paramètres de souplesse pour les segments, est déjà décrit dans les travaux de Khalil [Khalil et Besnard 2002], pour des segments suivant la théorie d'Euler-Bernoulli. Nous reprenons ici une adaptation utilisant la théorie de Timoshenko, plus complète.

Les 5 paramètres de souplesse d'un segment i ($s_{hi}, s_{ci}, sf_{zi}, sf_{yi}, s_{ti}$), correspondant aux raideurs ($k_{hi}, k_{ci}, kf_{zi}, kf_{yi}, k_{ti}$), n'apparaissent que dans l'expression des déformations ($dx, dy, dz, da, d\beta, d\gamma$) d'un segment i . Il n'est donc pas utile de dériver le modèle entier, mais seulement les formules des déformations d'un segment présentées dans les équations 160 et 161.

Le paramètre s_{hi} d'un segment i , apparaît seulement dans la déformée dx qui est une translation de l'équation 160. La dérivée de dx par rapport à la souplesse s_{hi} donne :

$$\frac{\partial dx}{\partial s_{hi}} = F_x + p_x \frac{L_i}{2} \tag{Eq. 174}$$

Cette dérivée nous donne l'influence du paramètre s_{hi} (et donc de $1/k_{hi}$) selon la direction X_i du repère du segment i . Cette influence en translation nous permet d'appliquer la relation cinématique de l'équation 50, et donc d'en déduire la colonne de la jacobienne correspondante :

$$J_{s_{hi}} = \begin{cases} \left(F_x + p_x \frac{L_i}{2} \right) X_i \\ 0_{\times 3} \end{cases} \tag{Eq. 175}$$

À partir des déformées dy et dz de l'équation 160, on obtient de la même manière la colonne de la jacobienne correspondante au paramètre s_{ci} , dont l'influence est portée par les directions Y_i et Z_i du repère du segment i :

$$J_{s_{ci}} = \begin{cases} \left(F_y + p_y \frac{L_i}{2} \right) Y_i + \left(F_z + p_z \frac{L_i}{2} \right) Z_i \\ 0_{\times 3} \end{cases} \tag{Eq. 176}$$

Le paramètre Sf_{zi} lui influence la pose de l'extrémité du segment i en translation selon Y_i (équation 160) mais aussi en rotation autour de Z_i (équation 161). L'expression de la colonne correspondante de la jacobienne fait donc intervenir le vecteur B_iC entre l'extrémité du segment B_i et la position des cibles C qui est mesurée.

$$J_{sf_{zi}} = \begin{cases} \left(p_y \frac{L_i^3}{8} + F_y \frac{L_i^2}{3} + M_z \frac{L_i}{2} \right) \mathbf{Y}_i + \mathbf{R}_J \wedge \mathbf{B}_i \mathbf{C} \\ \left(p_y \frac{L_i^2}{6} + F_y \frac{L_i}{2} + M_z \right) \mathbf{Z}_i = \mathbf{R}_J \end{cases} \quad \text{Eq. 177}$$

De la même manière, on obtient la colonne de la jacobienne correspondant au paramètre sf_{yi} , qui a une influence en translation selon \mathbf{Z}_i et en rotation autour de \mathbf{Y}_i dans le repère du segment i :

$$J_{sf_{yi}} = \begin{cases} \left(p_z \frac{L_i^3}{8} + F_z \frac{L_i^2}{3} - M_y \frac{L_i}{2} \right) \mathbf{Z}_i + \mathbf{R}_J \wedge \mathbf{B}_i \mathbf{C} \\ \left(-p_z \frac{L_i^2}{6} - F_z \frac{L_i}{2} + M_y \right) \mathbf{Y}_i = \mathbf{R}_J \end{cases} \quad \text{Eq. 178}$$

Enfin, le paramètre s_{ti} influence la pose de l'extrémité du segment i en rotation autour de \mathbf{X}_i (déformée $d\alpha$ de l'équation 161). La colonne correspondante de la jacobienne est alors :

$$J_{s_{ti}} = \begin{cases} \mathbf{R}_J \wedge \mathbf{B}_i \mathbf{C} \\ M_x \mathbf{X}_i = \mathbf{R}_J \end{cases} \quad \text{Eq. 179}$$

Expression des colonnes associées aux souplesses d'une articulation

Nous calculons ensuite les colonnes des 4 paramètres de souplesse ($s_r, s_z, s_{\theta r}, s_{\theta z}$) d'une articulation i , correspondant aux raideurs ($k_r, k_z, k_{\theta r}, k_{\theta z}$). Ces paramètres n'apparaissent que dans l'expression des déformations ($dx, dy, dz, d\alpha, d\beta, d\gamma$) d'une articulation i . Il n'est donc pas utile de dériver le modèle entier, mais seulement les formules des déformations d'une articulation présentées dans les équations 129 et 130.

Ces colonnes s'obtiennent de la même façon que pour celle associées aux souplesses d'un segment. On utilise ici le vecteur $\mathbf{O}_i \mathbf{C}$ entre le centre de la liaison \mathbf{O}_i et la position des cibles \mathbf{C} qui est mesurée.

$$J_{s_{ri}} = \begin{cases} F_x \mathbf{X}_i + F_y \mathbf{Y}_i \\ 0_{\times 3} \end{cases} \quad \text{Eq. 180}$$

$$J_{s_{zi}} = \begin{cases} F_z \mathbf{Z}_i \\ 0_{\times 3} \end{cases} \quad \text{Eq. 181}$$

$$J_{s_{\theta ri}} = \begin{cases} \mathbf{R}_J \wedge \mathbf{O}_i \mathbf{C} \\ M_x \mathbf{X}_i + M_y \mathbf{Y}_i = \mathbf{R}_J \end{cases} \quad \text{Eq. 182}$$

$$J_{s_{\theta zi}} = \begin{cases} \mathbf{R}_J \wedge \mathbf{O}_i \mathbf{C} \\ M_z \mathbf{Z}_i = \mathbf{R}_J \end{cases} \quad \text{Eq. 183}$$

Expression des colonnes associées aux jeux d'une articulation

Les expressions des colonnes de la jacobienne qui sont associées aux paramètres de jeux sont tout aussi simples à déduire, toujours à partir des équations 129 et 130 des déformées sur une articulation. Comme il a été mentionné au début de cette sous-section, les termes de signe peuvent être connus pour une jacobienne donnée. Par exemple la valeur de $\text{signe}(F_z)$ se calcule comme $F_z / \|F_z\|$ qui vaut 1 ou -1. Nous faisons apparaître ces valeurs sous forme d'un choix ± 1 qui sera fixé en fonction de la configuration de mesure. En suivant la même formulation que pour les souplesses dans une articulation, on a :

$$J_{b_{ri}} = \begin{cases} \frac{F_x}{F_r} \mathbf{X}_i + \frac{F_y}{F_r} \mathbf{Y}_i \\ 0_{\times 3} \end{cases} \quad \text{Eq. 184}$$

$$J_{b_{zi}} = \begin{cases} \pm \mathbf{Z}_i \\ 0_{\times 3} \end{cases} \quad \text{Eq. 185}$$

$$J_{b_{\theta_{ri}}} = \begin{cases} \mathbf{R}_j \wedge \mathbf{O}_i \mathbf{C} \\ \frac{M_x}{M_r} \mathbf{X}_i + \frac{M_y}{M_r} \mathbf{Y}_i = \mathbf{R}_j \end{cases} \quad \text{Eq. 186}$$

$$J_{b_{\theta_{zi}}} = \begin{cases} \mathbf{R}_j \wedge \mathbf{O}_i \mathbf{C} \\ \pm \mathbf{Z}_i = \mathbf{R}_j \end{cases} \quad \text{Eq. 187}$$

Expression des colonnes associées aux masses des segments

Pour finir, nous allons chercher l'expression des colonnes de la jacobienne correspondant aux paramètres de masse des segments. Le calcul de ces colonnes est plus complexe, car ces masses interviennent dans les torseurs d'efforts sur toutes les articulations et segments du robot. Une même masse va donc influencer les déformations de plusieurs segments et articulations. L'expression des colonnes pour les paramètres de masse est plus longue, car elle comporte plus d'éléments.

Trois influences différentes vont apparaître pour une même masse. Tout d'abord elle va influencer sur son propre segment i sous la forme de la répartition de charge p . Ensuite cette même masse va influencer sur tous les segments en amont du segment i c'est-à-dire les segments 1 à $i-1$. De même elle va influencer sur toutes les articulations en amont (articulation 1 à $i-1$). Commençons par exprimer l'influence d'une masse m_i sur son propre segment i , au travers de la charge répartie p . Cette charge en fonction de la masse m_i et de la pesanteur g (selon la verticale \mathbf{u}), s'exprime dans le repère du segment par :

$$\mathbf{p} = \frac{m_i}{L_i} g \mathbf{u} = \frac{m_i}{L_i} g V_x \mathbf{X}_i + \frac{m_i}{L_i} g V_y \mathbf{Y}_i + \frac{m_i}{L_i} g V_z \mathbf{Z}_i = p_x \mathbf{X}_i + p_y \mathbf{Y}_i + p_z \mathbf{Z}_i \quad \text{Eq. 188}$$

On retrouve les termes $\{p_x; p_y; p_z\}$ des équations 160 et 161. L'influence de la masse m_i sur la déformation du segment i est alors :

$$\begin{pmatrix} \frac{\partial dx_{segi}}{\partial m_i} \\ \frac{\partial dy_{segi}}{\partial m_i} \\ \frac{\partial dz_{segi}}{\partial m_i} \end{pmatrix} = \begin{pmatrix} \frac{s_h}{2} g V_x \\ \frac{s_c}{2} g V_y + s_f \frac{L_i^2}{8} g V_y \\ \frac{s_c}{2} g V_z + s_f \frac{L_i^2}{8} g V_z \end{pmatrix} + \frac{\partial \mathbf{Ori}_{segi}}{\partial m_i} \wedge \mathbf{B}_i \mathbf{C} \quad \text{Eq. 189}$$

$$\frac{\partial \mathbf{Ori}_{segi}}{\partial m_i} = \begin{pmatrix} \frac{\partial d\alpha_{segi}}{\partial m_i} = 0 \\ \frac{\partial d\beta_{segi}}{\partial m_i} = -s_f \frac{L_i}{6} g V_z \\ \frac{\partial d\gamma_{segi}}{\partial m_i} = s_f \frac{L_i}{6} g V_y \end{pmatrix} \quad \text{Eq. 190}$$

L'influence de cette même masse sur un segment j en amont s'exprime au travers du torseur des efforts appliqué sur l'extrémité du segment. L'expression de cette masse requiert le vecteur $\mathbf{B}_j \mathbf{G}_i$ entre l'extrémité du segment B_j et la position du centre de gravité G_i du segment i situé au milieu de celui-ci (dans le repère du segment j). La masse apparaît dans ce torseur sous la forme suivante :

$$\begin{cases} \mathbf{F}_{ji} = m_i g \mathbf{u} \\ \mathbf{M}_{ji} = \mathbf{B}_j \mathbf{G}_i \wedge m_i g \mathbf{u} = m_i M_{xmi} \mathbf{X}_j + m_i M_{ymi} \mathbf{Y}_j + m_i M_{zmi} \mathbf{Z}_j \end{cases} \quad \text{Eq. 191}$$

Dans cette équation 191, nous utilisons la notation $M_{xmi} \mathbf{X}_j + M_{ymi} \mathbf{Y}_j + M_{zmi} \mathbf{Z}_j$ pour exprimer le résultat du produit vectoriel entre $\mathbf{B}_j \mathbf{G}_i$ et \mathbf{u} , en y incluant aussi g . Ce produit vectoriel sera calculé pour une configuration donnée, c'est-à-dire pour une position de B_j et G_i donnée et une orientation de \mathbf{u} dans le repère du segment j . À partir des équations 160 et 161, dans lesquelles les efforts prennent les valeurs

données par l'équation 191, on déduit la contribution du taux de déformation du segment j due au paramètre de masse m_i sur la variation en position et en orientation de l'effecteur (point C) :

$$\begin{pmatrix} \frac{\partial dx_{segj}}{\partial m_i} \\ \frac{\partial dy_{segj}}{\partial m_i} \\ \frac{\partial dz_{segj}}{\partial m_i} \end{pmatrix} = \begin{pmatrix} s_h g V_x \\ s_c g V_y + s f_z \frac{L_j^2}{3} g V_y + s f_z \frac{L_j}{2} M_{zmi} \\ s_c g V_z + s f_y \frac{L_j^2}{3} g V_z - s f_y \frac{L_j}{2} M_{ymi} \end{pmatrix} + \frac{\partial Ori_{segj}}{\partial m_i} \wedge B_j C \quad \text{Eq. 192}$$

$$\frac{\partial Ori_{segj}}{\partial m_i} = \begin{pmatrix} \frac{\partial d\alpha_{segj}}{\partial m_i} = s_t M_{xmi} \\ \frac{\partial d\beta_{segj}}{\partial m_i} = -s f_y \frac{L_j}{2} g V_z + s f_y M_{ymi} \\ \frac{\partial d\gamma_{segj}}{\partial m_i} = s f_z \frac{L_j}{2} g V_y + s f_z M_{zmi} \end{pmatrix} \quad \text{Eq. 193}$$

Pour le calcul de l'influence de cette masse m_i sur la déformation d'une articulation j , il est nécessaire de déterminer la dérivée de fonctions composées par rapport à la masse m_i . Ces fonctions composées sont la direction du jeu radial et du jeu d'inclinaison, comme : $F_x/F_r = F_x/\sqrt{F_x^2 + F_y^2}$. Ces dérivées sont calculées à partir des valeurs d'effort données par l'équation 191 :

$$\frac{\partial (F_x/F_r)}{\partial m_i} = \frac{g V_x}{\sqrt{F_x^2 + F_y^2}} - \frac{F_x g (F_x V_x + F_y V_y)}{(F_x^2 + F_y^2)^{3/2}} \quad \text{Eq. 194}$$

$$\frac{\partial (F_y/F_r)}{\partial m_i} = \frac{g V_y}{\sqrt{F_x^2 + F_y^2}} - \frac{F_y g (F_x V_x + F_y V_y)}{(F_x^2 + F_y^2)^{3/2}} \quad \text{Eq. 195}$$

$$\frac{\partial (M_x/M_r)}{\partial m_i} = \frac{M_{xmi}}{\sqrt{M_x^2 + M_y^2}} - \frac{M_x (M_x M_{xmi} + M_y M_{ymi})}{(M_x^2 + M_y^2)^{3/2}} \quad \text{Eq. 196}$$

$$\frac{\partial (M_y/M_r)}{\partial m_i} = \frac{M_{ymi}}{\sqrt{M_x^2 + M_y^2}} - \frac{M_y (M_x M_{xmi} + M_y M_{ymi})}{(M_x^2 + M_y^2)^{3/2}} \quad \text{Eq. 197}$$

Toutes les valeurs (F_x, M_x, M_{xmi}, \dots) sont bien sûr exprimées dans le repère de l'articulation j où Z_j est la direction de l'axe de la liaison. À partir de ces équations 194 à 197 et à partir des déformées sur une articulation exprimées dans les équations 129 et 130, on peut alors déterminer l'influence de la masse m_i sur la déformation de l'articulation j :

$$\begin{pmatrix} \frac{\partial dx_{artj}}{\partial m_i} \\ \frac{\partial dy_{artj}}{\partial m_i} \\ \frac{\partial dz_{artj}}{\partial m_i} \end{pmatrix} = \begin{pmatrix} b_r \frac{\partial (F_x/F_r)}{\partial m_i} + g V_x s_r \\ b_r \frac{\partial (F_y/F_r)}{\partial m_i} + g V_y s_r \\ g V_z s_z \end{pmatrix} + \frac{\partial Ori_{artj}}{\partial m_i} \wedge O_j C \quad \text{Eq. 198}$$

$$\frac{\partial Ori_{artj}}{\partial m_i} = \begin{pmatrix} \frac{\partial d\alpha_{artj}}{\partial m_i} = b_{\theta r} \frac{\partial (M_x/M_r)}{\partial m_i} + M_{xmi} s_{\theta r} \\ \frac{\partial d\beta_{artj}}{\partial m_i} = b_{\theta r} \frac{\partial (M_y/M_r)}{\partial m_i} + M_{ymi} s_{\theta r} \\ \frac{\partial d\gamma_{artj}}{\partial m_i} = M_{zmi} s_{\theta z} \end{pmatrix} \quad \text{Eq. 199}$$

On peut à présent concaténer les influences de la masse m_i sur les déformations et jeux de toutes les articulations et de tous les segments, pour obtenir l'influence de cette masse sur la pose de l'effecteur.

Chaque terme concaténé est exprimé dans le repère de l'articulation ou du segment concerné, c'est pourquoi il est important de faire apparaître les directions du segment i (\mathbf{X}_{segi} , \mathbf{Y}_{segi} , \mathbf{Z}_{segi}) et des autres articulations et segments (\mathbf{X}_{segi} , \mathbf{X}_{artj} , ...).

$$\mathbf{J}_{m_i} = \begin{cases} \frac{\partial dx_{\text{segi}}}{\partial m_i} \mathbf{X}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial dx_{\text{segi}}}{\partial m_i} \mathbf{X}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial dx_{\text{artj}}}{\partial m_i} \mathbf{X}_{\text{artj}} + \dots \\ \frac{\partial dy_{\text{segi}}}{\partial m_i} \mathbf{Y}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial dy_{\text{segi}}}{\partial m_i} \mathbf{Y}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial dy_{\text{artj}}}{\partial m_i} \mathbf{Y}_{\text{artj}} + \dots \\ \frac{\partial dz_{\text{segi}}}{\partial m_i} \mathbf{Z}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial dz_{\text{segi}}}{\partial m_i} \mathbf{Z}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial dz_{\text{artj}}}{\partial m_i} \mathbf{Z}_{\text{artj}} \\ \frac{\partial d\alpha_{\text{segi}}}{\partial m_i} \mathbf{X}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial d\alpha_{\text{segi}}}{\partial m_i} \mathbf{X}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial d\alpha_{\text{artj}}}{\partial m_i} \mathbf{X}_{\text{artj}} + \dots \\ \frac{\partial d\beta_{\text{segi}}}{\partial m_i} \mathbf{Y}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial d\beta_{\text{segi}}}{\partial m_i} \mathbf{Y}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial d\beta_{\text{artj}}}{\partial m_i} \mathbf{Y}_{\text{artj}} + \dots \\ \frac{\partial d\gamma_{\text{segi}}}{\partial m_i} \mathbf{Z}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial d\gamma_{\text{segi}}}{\partial m_i} \mathbf{Z}_{\text{segi}} + \sum_{j=1}^{i-1} \frac{\partial d\gamma_{\text{artj}}}{\partial m_i} \mathbf{Z}_{\text{artj}} \end{cases} \quad \text{Eq. 200}$$

Calcul de la jacobienne élastique totale

On a ainsi l'expression des colonnes de la jacobienne de notre modèle élasto-géométrique, pour chaque paramètre élastique. Les colonnes sont alors assemblées pour former la matrice jacobienne élastique $\mathbf{J}_{\text{welast}}$ pour un ensemble {configuration, charge} donné. L'ordre des colonnes suit le vecteur des paramètres $d\mathbf{w}$, en remplaçant les paramètres de raideur par des paramètres de souplesse. Dans la résolution ou dans l'étude d'identifiabilité, nous n'utilisons que la partie position de la jacobienne $\mathbf{J}_{\text{welast,p}}$, et pas la partie orientation car l'instrument de mesure C-Track est meilleur en position lors de l'identification. Si nous voulions utiliser la partie inférieure de la jacobienne (partie orientation), il faudrait adimensionner la partie supérieure avec la longueur caractéristique présentée dans l'équation 22.

On constitue ensuite la matrice jacobienne totale $\mathbf{J}_{\text{tot,welast,p}}$, reprenant les matrices jacobienne $\mathbf{J}_{\text{welast,p}}$ établies pour les m ensembles {configuration, charge} de mesure. Cette matrice totale de taille $3m \times 40$, permet alors la résolution de l'identification, et l'étude d'identifiabilité des paramètres pour la méthode d'identification proposée. Bien sûr une conversion est à effectuer dans l'étude ou dans la résolution pour revenir aux paramètres de raideur à partir des souplesses.

3.3.3 Identifiabilité

Avant de passer à l'identification des paramètres élastiques de notre modèle, il est important de vérifier si ces paramètres sont bien identifiables avec la méthode que nous souhaitons mettre en place. Nous proposons dans cette sous-section, une étude originale sur l'identifiabilité qui va permettre de déterminer si un paramètre peut être identifié ou non avec cette méthode, et avec les configurations et charges choisies. Mais elle va également nous permettre de détecter des paramètres très peu influents au regard de la qualité de positionnement souhaitée sur l'effecteur. Il sera alors possible de négliger ces paramètres. Par la même occasion, notre étude d'identifiabilité nous aidera à déceler des paramètres redondants, qui ne sont donc pas identifiables et qui doivent être éliminés, car déjà pris en compte au travers d'autres paramètres.

Comme précisé dans la section précédente, nous travaillons avec la jacobienne totale élastique uniquement en position $J_{tot_welast,p}$. Nous allons « adimensionner » cette jacobienne pour compenser les différences d'ordres de grandeur entre les paramètres. Puis nous allons étudier l'identifiabilité des paramètres pour simplifier le modèle en suivant ces étapes :

- Recherche des paramètres non influents : élimination de ces paramètres
- Recherche des paramètres redondants : élimination des paramètres en surplus
- Estimation de l'influence de chaque paramètre : élimination des paramètres trop peu influents

Des méthodes d'étude de l'identifiabilité ont été présentées dans la section 1.3.5. Afin de déterminer le nombre de paramètres identifiables, ces méthodes se basent toutes sur le rang de la jacobienne, que nous utiliserons également dans notre méthode. Cependant, elles ne permettent pas de savoir quels sont les paramètres à retirer en cas de redondance. La méthode que nous proposons permet de bien identifier les ensembles de paramètres qui sont liés entre eux, afin de pouvoir sélectionner parmi ces ensembles, quels seront les paramètres jugés redondants et à éliminer.

Pour identifier les paramètres élastiques ou même pour réaliser une étude d'identifiabilité, il est nécessaire d'avoir une première estimation approximative de ces paramètres. Nous avons donc mis en place des essais afin de déterminer l'ordre de grandeur des paramètres élastiques de notre modèle élasto-géométrique, en chargeant le robot et en mesurant la déformée en différents points du robot. L'hypothèse a été faite que l'ordre de grandeur des paramètres d'articulation est le même pour chaque articulation. Nous faisons la même hypothèse pour les segments. Les ordres de grandeur estimés des paramètres d'une articulation i sont donnés dans le tableau 21 (raideurs en N/m et N.m/rad, et jeux en m et en rad). Les ordres de grandeur estimés des paramètres d'un segment i et des masses des segments sont donnés dans le tableau 22 (masses en kg, raideurs en N/m et en N.m/rad).

k_{ri}	k_{zi}	$k_{\theta ri}$	$k_{\theta zi}$	b_{ri}	b_{zi}	$b_{\theta ri}$	$b_{\theta zi}$
10^6	10^6	10^7	10^6	10^{-5}	10^{-5}	10^{-5}	10^{-4}

Tableau 21 : Ordres de grandeur des paramètres d'une articulation i

k_{hi}	k_{ci}	$k_{f_{vi}}$	$k_{f_{zi}}$	k_{ti}	m_3	m_{45}	m_5	m_6
10^9	10^8	10^7	10^7	10^7	200	170	60	30

Tableau 22 : Ordres de grandeur des paramètres d'un segment i , et masses

Nous avons décidé d'« adimensionner » la matrice jacobienne $J_{tot_welast,p}$, car le calcul du rang d'une matrice est très sensible aux différences d'ordres de grandeur dans la matrice. Or parmi les paramètres que nous cherchons à identifier, les différences d'ordre de grandeur sont très importantes entre jeux, masses et raideurs. Ces différences rendent les valeurs dans la matrice jacobienne $J_{tot_welast,p}$ très disparates. Au regard des ordres de grandeur des tableaux 21 et 22, l'utilisation d'une telle matrice jacobienne pour l'identification donnera de mauvais résultats, et le calcul de son rang va inévitablement conclure au fait que beaucoup de paramètres ne sont pas identifiables. C'est pourquoi nous voulons avoir des paramètres à identifier qui soient tous du même ordre de grandeur. Dans ce but nous utilisons les valeurs estimées des 40 paramètres élastiques w_i sous forme d'une matrice diagonale pour « adimensionner » la matrice jacobienne en suivant l'équation 201.

$$\overline{J_{tot_welast,p}} = J_{tot_welast,p} \text{diag}(w_i) \quad \text{Eq. 201}$$

Ainsi la jacobienne devient plus équilibrée et les calculs ne souffriront pas d'une grande différence d'échelle. On utilise cette jacobienne adimensionnée pour l'étude d'identifiabilité, mais elle doit aussi être utilisée lors de la résolution de l'identification pour donner de meilleurs résultats. La relation de l'équation 24 utilisée pour la résolution devient dans ce cas :

$$\overline{dw} = \left(\overline{J_{tot_welast,p}}^t \overline{J_{tot_welast,p}} \right)^{-1} \overline{J_{tot_welast,p}}^t dP \quad \text{avec} \quad dw = \text{diag}(w_i) \overline{dw} \quad \text{Eq. 202}$$

Les variations adimensionnées \overline{dw} qui seront identifiées seront toutes du même ordre de grandeur et seront utilisées pour déterminer les corrections dw à appliquer aux paramètres.

L'étude que nous proposons sur l'identifiabilité se découpe en trois grandes étapes : élimination des paramètres non influents, recherche des paramètres redondants, estimation de l'influence de chaque paramètre.

Élimination des paramètres non influents

Comme pour la plupart des méthodes d'identifiabilité, il faut commencer par éliminer les paramètres non identifiables triviaux. Ces paramètres correspondent à des colonnes de la jacobienne $\overline{J_{tot_{welast,p}}}$ nulles ou dont les valeurs sont proches de 0. Il faut donc fixer une valeur seuil, liée à la précision de calcul. Au-delà de cette limite, les valeurs seront considérées comme nulles. Avec notre logiciel (Matlab) la précision de calcul est de l'ordre de 10^{-14} . Pour notre modèle élasto-géométrique, nous ne trouvons aucun paramètre trivial à éliminer.

Pourtant, après calcul, le rang de notre matrice $\overline{J_{tot_{welast,p}}}$ est de 34 alors que nous avons 40 paramètres à identifier. Des paramètres de notre modèle sont donc liés entre eux. Nous recherchons ces paramètres dans l'étape suivante.

Recherche des paramètres redondants

Les méthodes existantes sur l'identifiabilité, utilisent des décompositions SVD ou des études des valeurs singulières de $\overline{J_{tot_{welast,p}}}$ pour déterminer s'il y a ou non des paramètres redondants. Mais elles ne permettent pas d'identifier quels sont ces paramètres. Le rang de la jacobienne est lié au rang de la famille des vecteurs colonnes de cette jacobienne. Nous proposons alors simplement ici de rechercher d'éventuelles combinaisons linéaires entre les colonnes de la jacobienne, qui expliqueraient le rang calculé de la matrice (34). Ce rang calculé laisse supposer que 6 paramètres liés doivent être éliminés.

Ainsi nous recherchons pour un paramètre w_k , si la colonne correspondante $\overline{J_{tot_{wk,p}}}$ est la combinaison linéaire d'une ou plusieurs autres colonnes ($\overline{J_{tot_{wk,p}}} = a \overline{J_{tot_{wa,p}}} + \dots + e \overline{J_{tot_{we,p}}}$), ce qui permettra de savoir si w_k est lié à d'autres paramètres. Pour vérifier s'il y a une combinaison linéaire entre n vecteurs (n colonnes), il faut calculer le rang de cette famille de vecteurs. Si le rang est égal à n , le nombre de vecteurs, alors il n'existe pas de combinaison donnant l'un de ces vecteurs à partir des autres. Il est possible de rechercher de manière empirique toutes les combinaisons possibles pour lesquelles ce rang est inférieur à n .

Le calcul doit alors être fait pour les g paramètres, pour des combinaisons linéaires à 1, 2, ... $g-1$ paramètres (dans notre cas $g=40$). Ce calcul est lourd et prend du temps, c'est pourquoi nous proposons d'utiliser une décomposition QR de la jacobienne, afin d'estimer quels sont les paramètres potentiellement liés et ainsi réduire le calcul. En premier lieu, la décomposition QR (avec R triangulaire supérieure) donne sur la diagonale de R , l'influence des paramètres dans une base propre. Une faible valeur sur cette diagonale signifie que le paramètre correspondant est peu influent sur la pose de l'effecteur. Deux raisons majeures sont alors possibles : ce paramètre est tout simplement peu influent, ou bien un autre paramètre lié a déjà pris en compte ce type de sollicitation de l'effecteur. La diagonale de R fournit donc une liste de paramètres peu influents qui peuvent être liés. Nous recherchons alors des combinaisons linéaires seulement pour ces paramètres peu influents, ce qui réduit énormément le temps de calcul.

Pour notre modélisation élasto-géométrique la matrice triangulaire R issue de la décomposition QR de la jacobienne $\overline{J_{tot_{welast,p}}}$, a fait ressortir les paramètres suivants : $k_{r2}, b_{r5}, b_{z6}, k_{c3}, k_{c45}, k_{fz45}, k_{t45}, k_{0r6}, b_{0r6}$. Parmi ces paramètres, tous ne sont pas nécessairement liés, ce sont juste les paramètres les moins

influent dans la décomposition. Après des calculs de rangs, nous trouvons des combinaisons simples ($\overline{J_{tot_{wk,p}}} = a. \overline{J_{tot_{wa,p}}}$) entre les paramètres suivants :

$$k_{r2} \leftrightarrow k_{z1} \leftrightarrow k_{r3} \quad b_{r5} \leftrightarrow \emptyset \quad b_{z6} \leftrightarrow \emptyset \quad k_{c3} \leftrightarrow \emptyset \quad b_{\theta r6} \leftrightarrow \emptyset \quad \text{Eq. 203}$$

$$k_{c45} \leftrightarrow kf_{z45} \leftrightarrow k_{\theta z3} \quad k_{\theta z4} \leftrightarrow k_{t45} \quad k_{z4} \leftrightarrow k_{h45} \quad k_{\theta r6} \leftrightarrow \emptyset \quad \text{Eq. 204}$$

À partir de ces résultats, on peut donc dire que les paramètres b_{r5} , b_{z6} , $k_{\theta r6}$, $b_{\theta r6}$ et k_{c3} sont simplement peu influents, certains seront supprimés dans l'étape suivante qui étudie cette influence. Les paramètres liés dans les équations 203 et 204 ont la même influence sur la pose des cibles suivies. Il suffit donc de n'en garder qu'un seul parmi chaque lot de paramètres liés, afin de ne plus avoir de redondance sans affecter pour autant la qualité de modélisation de la pose des cibles. Nous décidons de supprimer les 6 paramètres redondants suivants : k_{r2} , k_{r3} , kf_{z45} , k_{c45} , k_{t45} , k_{h45} . Le choix de ces 6 paramètres dans les lots est libre et n'a pas d'importance pour la modélisation. Notre modélisation élasto-géométrique ne comporte donc plus que 34 paramètres.

A la fin de cette étape, notre modèle est réduit à 34 paramètres, ce qui correspond au rang de $\overline{J_{tot_{welast,p}}}$ calculé précédemment. Un nouveau calcul du rang sur la jacobienne réduite à 34 colonnes nous donne également ce résultat. On peut donc conclure que tous les paramètres du modèle sont à présent identifiables pour notre méthode d'identification et pour les configurations et les charges choisies.

Estimation de l'influence de chaque paramètre

Enfin, la matrice jacobienne $\overline{J_{tot_{welast,p}}}$ nous permet d'estimer l'influence de chaque paramètre sur le positionnement des cibles suivies. En effet chaque colonne de cette matrice donne l'influence d'un paramètre sur la position des cibles pour toutes les configurations de mesure. Pour cette estimation nous utilisons bien sûr la version de la jacobienne réduite à 34 paramètres déduite précédemment. Pour pouvoir comparer ces influences à une référence, nous calculons l'influence moyenne des paramètres (moyenne du logarithme des valeurs de la matrice) ainsi que l'écart type associé. Cette moyenne, cet écart type ainsi que l'influence maximale de chaque paramètre sur la pose des cibles (valeur maximale de chaque colonne de la jacobienne), sont représentés sur la figure 64. Afin de bien représenter les ordres de grandeur de ces influences, cette figure est représentée avec une échelle logarithmique. Les paramètres très peu influents apparaissent sur cette figure, comme des points aberrants.

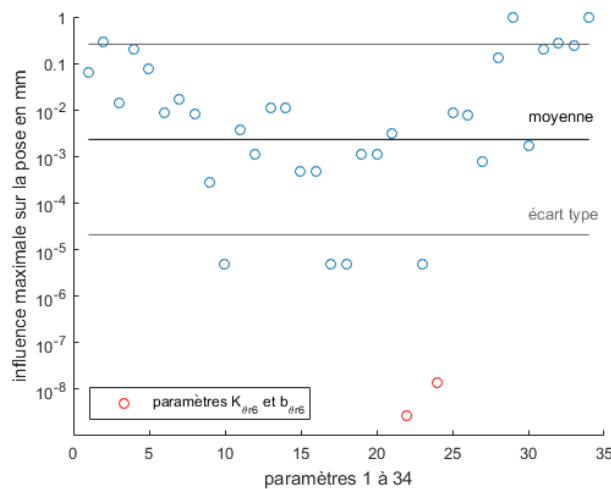


Figure 64 : Influence maximale de chaque paramètre sur la pose des cibles (échelle logarithmique)

Sur la figure 64, deux paramètres participent très peu au positionnement des cibles suivies : $k_{\theta r6}$, $b_{\theta r6}$. Cependant ces paramètres influent directement sur l'orientation de l'outil, qui n'a pas été prise en

compte dans notre méthode d'identification (uniquement position des cibles). Il est donc préférable de conserver ces paramètres dans notre modèle. Pour optimiser l'identification des paramètres, vu la faible influence de ces deux paramètres sur la position, il est possible de les identifier à part avec une autre méthode focalisée uniquement sur la raideur et le jeu radial sur l'axe 6. C'est le choix que nous faisons, car une identification avec les autres paramètres ne donnera pas une qualité d'identification satisfaisante sur ces deux paramètres peu observables.

Notre étude sur l'identifiabilité des paramètres nous a donc permis de déceler des paramètres redondants ainsi que des paramètres trop peu influents en position pour aboutir à un modèle identifiable. Notre identification des paramètres élastiques (et géométriques par itérations) suivra donc la méthodologie présentée en section 3.3.1. En résumé, les 32 paramètres élastiques à identifier avec cette méthode sont les suivants :

$$k_{z1}, k_{\theta r1}, b_{\theta z1}, k_{\theta z2}, k_{\theta z3}, b_{\theta z3}, k_{z4}, k_{\theta z4}, b_{z4}, b_{\theta r4}, b_{\theta z4}, k_{r5}, k_{z5}, k_{\theta r5}, k_{\theta z5}, b_{r5}, b_{z5}, b_{\theta r5}, b_{\theta z5}, k_{z6}, b_{z6}, k_{h3}, k_{c3}, k_{f_y2}, k_{f_z3}, k_{t3}, k_{f_y45}, m_3, m_4, m_5, m_6 \quad \text{Eq. 205}$$

Les deux paramètres radiaux $k_{\theta r6}$ et $b_{\theta r6}$ sur la liaison 6 seront identifiés à part, ce qui nous permet au final d'aboutir à une modélisation élasto-géométrique complète de 34 paramètres élastiques en plus des paramètres géométriques. Cette modélisation représente le robot, l'outil (capteur) et le positionnement du robot par rapport à un référentiel externe (C-Track).

3.4 Conclusion sur la modélisation élasto-géométrique

Le début de ce chapitre nous a permis de mettre en évidence le besoin d'une modélisation plus détaillée du robot. Nous avons tout d'abord observé que notre modélisation géométrique était suffisante pour l'étude des indices de performance du robot. Nous avons également constaté que l'erreur de justesse résultant de l'utilisation de notre modèle géométrique pour générer la trajectoire, n'impactait pas de manière significative la qualité de la numérisation réalisée. Par contre nous avons montré que cette même erreur de justesse était très impactante lorsque nous utilisons le modèle géométrique et les coordonnées du robot pour recalibrer les lignes laser numérisées. Il sera d'ailleurs effectivement nécessaire d'utiliser notre modèle pour le recalage dans les cas où les cibles suivies ne seraient plus visibles par le C-Track, ou lorsque nous souhaitons ne pas utiliser de système de suivi externe au cours de la numérisation.

Nous avons donc cherché à détailler le modèle géométrique, avec des paramètres non géométriques. Les paramètres que nous avons retenus sont des élasticités dans les segments du robot, ainsi que des élasticités et des jeux dans les liaisons. Nous avons défini les paramètres d'élasticité liés à la déformation d'un segment en suivant la théorie des poutres (Timoshenko). Nous avons également créé une modélisation pour les articulations du robot, définissant des paramètres de jeu et d'élasticité liés aux déformations et déplacements dans une articulation. Pour mettre en place notre modèle d'une articulation, nous nous sommes inspiré de la norme ISO 230-1 [ISO.230-1 2012], définissant les erreurs que l'on retrouve dans une articulation. Nous avons ensuite inclus ces paramètres dits « élastiques » dans le modèle géométrique pour donner un modèle complet élasto-géométrique. Ce dernier définit le comportement géométrique et élastique du robot, de l'outil (capteur laser) et du positionnement du robot dans un référentiel externe (C-Track). Sans aucune hypothèse simplificatrice, le nombre de paramètres élastiques de notre modèle complet est de 84 : 8 paramètres par articulations et 6 paramètres par segment pour un total de 6 segments et 6 articulations.

Dans la section 3.2, nous avons émis des hypothèses liées à la géométrie de notre robot ainsi qu'à notre application de numérisation afin de simplifier et retirer des paramètres qui ne sont pas utiles. Ces simplifications réduisent le modèle à 40 paramètres élastiques. Nous avons poursuivi cette logique de

simplification des paramètres dans la section 3.3, en proposant une étude originale sur l'identifiabilité des paramètres élastiques. Pour cette étude il a fallu définir la méthode d'identification que nous allons mettre en place, ainsi que les différentes configurations de mesure du robot utilisées dans cette identification. Notre étude sur l'identifiabilité des paramètres nous a permis de réduire notre modèle à 34 paramètres élastiques (dont 2 à identifier séparément). Nous aboutissons donc à un modèle ne conservant que les paramètres utiles et représentant les élasticités et jeux dans le robot. Ces paramètres utiles sont tous identifiables, et leur identification sera mise en place dans des travaux futurs. Ils permettront de réaliser le recalage des lignes laser numérisées, à partir de notre modèle élasto-géométrique et des coordonnées articulaires du robot.

Chapitre 4

Indices de performances pour la qualification du robot

Le chapitre 4 se focalise essentiellement sur la sélection et la proposition d'indices de performance, qui seront utilisés dans le processus de génération de trajectoire. Ces indices devront permettre d'identifier les configurations du robot qui satisfont des contraintes sur la trajectoire en termes de vitesse, qualité de pose et autres capacités du robot (domaine accessible et potentiel de mouvement), et d'optimiser suivant ces mêmes critères.

Dans un premier temps, après avoir montré l'insuffisance des indices de vitesse usuels en robotique, nous proposons deux nouveaux indices de vitesse, l'un portant sur le respect de contraintes et l'autre servant à optimiser. La caractéristique essentielle de ces indices réside dans leur aptitude à représenter la capacité en vitesse du robot au niveau de l'effecteur suivant une direction désirée et ceci, tout le long d'une trajectoire. Dans un deuxième temps, en vue d'applications exigeantes en termes de qualité de pose, nous sommes amenés à proposer un nouvel indice donnant la meilleure répétabilité que peut assurer de manière sûre le robot le long d'une trajectoire désirée. Par ailleurs, afin de générer une trajectoire dans le domaine de travail et de conserver un potentiel de mouvement, nous sélectionnons un indice portant sur l'accessibilité de l'effecteur du robot. Enfin, nous regroupons tous les indices retenus au sein du générateur de trajectoire. Un cas servant d'exemple (trajectoire en 3 passes) est utilisé pour générer des cartographies 2D permettant de mieux illustrer les différents indices durant tout le chapitre.

4.1 Indices de performance et contrôle de la vitesse

Dans notre présentation de la stratégie de génération de trajectoire au chapitre 1, nous avons vu l'importance de connaître les capacités des différents éléments constituant la cellule de numérisation, et notamment du robot. Il faut d'une part savoir si le robot pourra satisfaire les contraintes fixées sur la numérisation, et d'autre part exploiter au mieux les capacités du robot en vue d'une optimisation. Or ces capacités sont dépendantes de la configuration du robot, car elles sont hétérogènes dans son espace de travail. Il existe donc des indices (ou critères) permettant de calculer les performances du robot pour une configuration donnée. Ces indices sont des fonctions prenant 6 coordonnées articulaires en entrée (configuration donnée) pour calculer une valeur de capacité du robot. Le générateur de trajectoire utilisera ces fonctions pour vérifier des respects de contraintes et rechercher des optimums. Dans ce chapitre nous illustrons ces indices à 6 paramètres par des cartographies 2D, en fixant l'orientation de l'effecteur (-3D) et en exploitant la symétrie de révolution donnée par l'axe 1 (-1D).

Dans la section 1.4, nous avons déjà décrit quelques-uns de ces indices très présents dans le domaine de la robotique sérielle, et nous allons à présent sélectionner ou proposer des indices cohérents avec notre besoin de génération de trajectoire. Nous allons dans un premier temps regarder du côté des performances en vitesse du robot. Puis nous verrons dans les sections suivantes comment qualifier la qualité de pose du robot et son domaine accessible, avant de chercher à rassembler tous les indices retenus.

La notion de vitesse est primordiale dans un contexte de numérisation, afin de maîtriser la densité du nuage de points. De plus, nous souhaitons que notre application soit compatible avec un contexte industriel (chaîne de production), et la maîtrise de la vitesse permet une optimisation du temps de cycle. La connaissance des capacités en vitesse du robot va donc nous permettre de respecter une vitesse maximale ou minimale dictée par l'opération de numérisation, ainsi que d'optimiser l'utilisation du robot en vitesse pour minimiser le temps de l'opération. Nous allons donc sélectionner des indices de vitesse adaptés à ces besoins.

4.1.1 Insuffisance des indices de vitesse usuels

La vitesse du système support de l'opération de numérisation doit être maîtrisée, ce qui nous amène à sélectionner un critère décrivant le comportement en vitesse du robot. Parmi les indices de performance décrits au chapitre 1, nous avons vu la manipulabilité [Yoshikawa 1985] qui donne une estimation de la capacité du robot à générer une vitesse. Cet indice est basé sur un produit des valeurs singulières de la matrice jacobienne articulaire du robot, comme décrit dans l'équation 30. Cette matrice jacobienne est obtenue à partir de notre modélisation du robot. À partir de cette formulation, il est possible de dresser des cartographies 2D des capacités en vitesse du robot pour des orientations de l'effecteur fixées. Deux cartographies de l'indice de manipulabilité sont présentées sur les figures 65 et 66. Pour générer ces cartographies, nous fixons également la posture du robot, car pour une même pose (position et orientation) 8 postures du robot sont possibles.

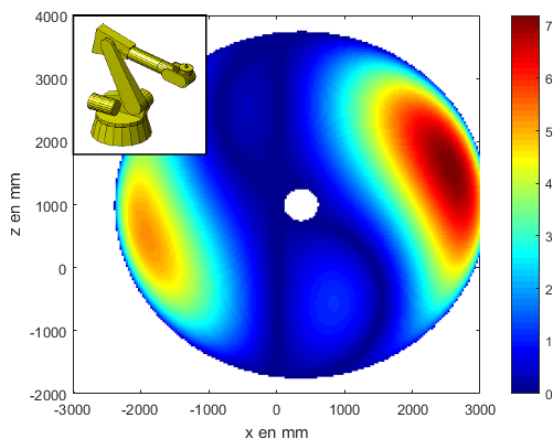


Figure 65 : Cartographie de la manipulabilité, orientation vers le haut et coude haut

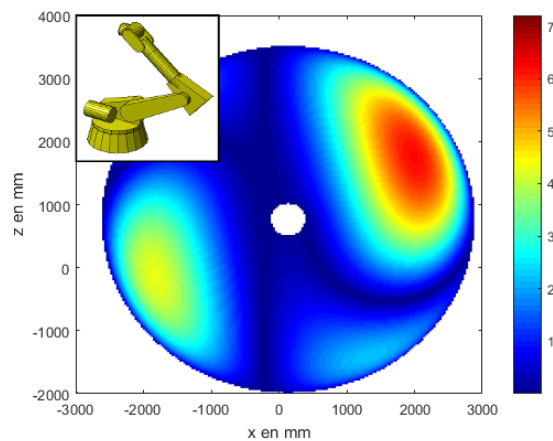


Figure 66 : Cartographie de la manipulabilité, orientation vers la gauche et coude bas

Grâce à ces cartographies, il est facile de sélectionner une zone de travail dans laquelle le robot aura la meilleure capacité en vitesse, pour réaliser la numérisation. On peut voir des zones inaccessibles pour l'effecteur en blanc, des zones de faible capacité en vitesse en bleu, et des zones de très bonne capacité en rouge. On peut voir que la posture et l'orientation de la figure 65 donnent la meilleure capacité en vitesse, dans la zone rouge la plus foncée. En comparant ainsi de multiples postures et orientations, on peut déterminer la meilleure zone pour effectuer l'opération de numérisation (en termes de vitesse). Dans notre générateur de trajectoires, l'indice de manipulabilité peut donc être utilisé pour sélectionner une zone de travail pour la numérisation.

La valeur de l'indice de manipulabilité n'est pas une valeur chiffrée directement exploitable comme le serait une vitesse suivant une direction en mètres par seconde. Elle sert uniquement à effectuer des comparaisons avec d'autres valeurs de manipulabilité, que ce soit pour des configurations différentes ou même pour d'autres robots. Cet indice permet donc de sélectionner des configurations, mais ne permet pas de vérifier le respect de contraintes (valeur chiffrée non exploitable).

Le calcul de la manipulabilité repose sur la jacobienne articulaire et donc sur le modèle du robot. La figure 65 a été réalisée à partir du modèle géométrique théorique du robot. Nous souhaitons à présent

vérifier si un changement de modèle (pour un meilleur) aurait un quelconque effet sur l'indice. Dans ce but nous avons modifié les paramètres du modèle théorique aléatoirement, de l'ordre de deux fois les corrections maximales identifiées dans les tableaux 11 et 12 du chapitre 2 (soit 2mm et 1°). On obtient alors la figure 67, quasiment identique à la figure 65. La différence entre les deux figures est donnée sur la figure 68.

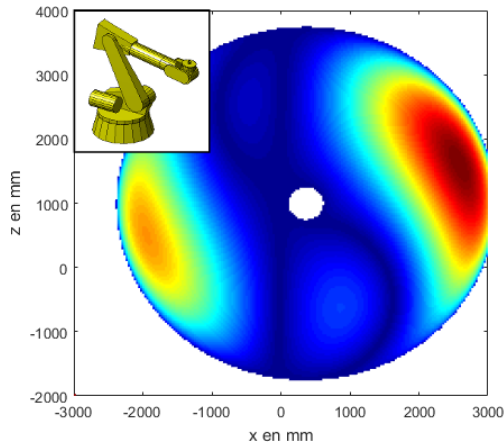


Figure 67 : Cartographie de manipulabilité, modèle théorique modifié

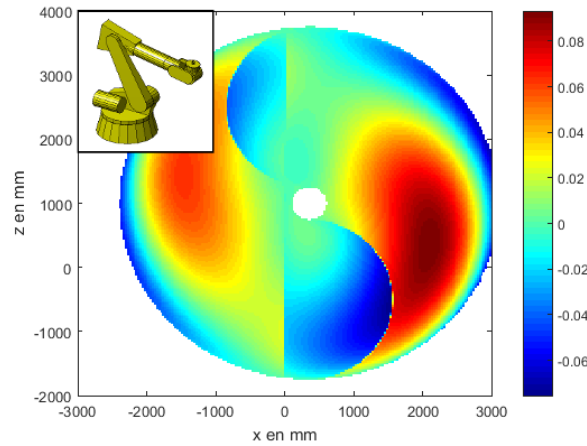


Figure 68 : Différence entre les figures 65 et 67.

Sur la figure 68, la différence moyenne est de 0,04 soit 0,5% (pourcentage effectué avec les extremums des échelles des figures 65 et 67), et la différence maximale est de 0,085 soit 1,5%. La différence est donc très faible, surtout si l'on rappelle que l'indice de la manipulabilité ne permet que de faire du comparatif pour de l'optimisation.

Nous pouvons donc conclure qu'il est inutile de chercher à utiliser un meilleur modèle que le modèle théorique, pour générer l'indice de manipulabilité. La matrice jacobienne étant à la base de la manipulabilité, nous supposons qu'elle est également peu dépendante des erreurs sur les paramètres. Nous faisons alors l'hypothèse que tous les autres indices utilisant cette jacobienne, n'ont pas besoin d'un modèle plus abouti que le modèle théorique, hypothèse qui se vérifie pour chaque indice.

L'indice de manipulabilité permet de sélectionner des zones de meilleure capacité en vitesse par comparaison, mais ne permet pas de vérifier une contrainte. Or le générateur de trajectoire requiert également de pouvoir vérifier des contraintes imposées. De plus, en chaque point de la trajectoire, la vitesse qui importe est celle suivant la direction d'avance. Or la manipulabilité est un indice global, dont la valeur peut être importante si les capacités en vitesse du robot sont bonnes suivant une direction qui ne nous intéresse pas. Ainsi même pour effectuer une sélection de configuration, la manipulabilité peut être un mauvais indice. La manipulabilité est avant tout un indice servant à la conception des robots industriels et à la sélection d'une zone de travail donnant globalement de bonnes vitesses de déplacement. Ce type d'indice n'est donc pas adapté pour l'optimisation d'une opération rigoureuse comme une numérisation. On préfère alors un indice donnant une valeur chiffrée de la vitesse de l'effecteur, pour une configuration donnée du robot et suivant une direction d'avance.

L'ellipsoïde de vitesses est un autre indice présenté au chapitre 1, qui utilise un ellipsoïde pour décrire les vitesses de l'effecteur suivant 3 directions prépondérantes qui sont les demi-axes de l'ellipsoïde. Pour une description des vitesses de translation et de rotation, il y a donc deux ellipsoïdes. On peut donc utiliser cet indice pour avoir une estimation de la vitesse de l'effecteur suivant une direction d'avance. L'ellipsoïde de vitesses donne alors les capacités du robot pour une configuration donnée et suivant une direction donnée. Mais cet ellipsoïde comme la manipulabilité ne donne pas une valeur chiffrée exploitable en mètres par seconde. Nous n'avons donc toujours pas accès à une valeur de vitesse utilisable pour la validation des contraintes en vitesse sur la trajectoire.

Cet ellipsoïde exploite aussi la jacobienne articulaire, qui lie les déplacements des articulations aux déplacements de l'effecteur. Mais plutôt que d'utiliser uniquement la jacobienne, il est plus intéressant d'utiliser également les valeurs de vitesse de chaque articulation pour en déduire celle de l'effecteur en mètres par seconde (ou radians par seconde). Ainsi la connaissance de la vitesse maximale atteignable sur chaque articulation permet ensuite de déduire la vitesse maximale atteignable par l'effecteur, suivant n'importe quelle direction. Nous présentons dans la suite deux indices de vitesse basés sur ce principe.

4.1.2 Proposition d'indices de vitesse pour la génération de trajectoire

Afin de maîtriser et d'optimiser la vitesse de l'effecteur dans le générateur de trajectoire, nous proposons de construire deux indices : un indice dit de vitesse minimale et un indice de vitesse moyenne.

L'indice de vitesse minimale sert à sélectionner une zone de travail où placer la pièce à numériser, et dans laquelle le robot respectera des contraintes en vitesse imposée à la trajectoire. Il faut donc pouvoir garantir une vitesse minimale tout le long de la trajectoire. Cet indice donne la plus petite vitesse maximale calculée sur la trajectoire. Ainsi nous sommes assurés de pouvoir respecter cette vitesse d'avance sur la trajectoire.

L'indice de vitesse moyenne est utile pour une optimisation de la durée de l'opération de numérisation. Pour cela, il faut être rapide sur l'ensemble de la trajectoire. Cet indice permet donc de rechercher une vitesse moyenne le long de la trajectoire, qui soit maximale. Il est calculé comme la valeur moyenne des vitesses maximales sur tous les points de la trajectoire.

Tout d'abord il faut calculer la vitesse maximale de l'effecteur suivant une direction et pour une configuration. Pour ce calcul nous utilisons la jacobienne articulaire associée au modèle géométrique théorique J_θ , ainsi que les vitesses maximales atteignables sur chaque articulation $\dot{\theta}_{i\max}$. Le calcul est donné dans l'équation 206. Le résultat est un vecteur décrivant les vitesses de l'effecteur dans le repère cartésien de la base du robot, ce qui permet facilement d'en déduire une vitesse suivant une direction. Par contre l'ellipsoïde de vitesses pourra tout de même servir dans le générateur de trajectoire, pour trouver facilement une direction d'avance prépondérante en vitesse. Ce qui peut alors fournir une bonne direction générale pour le balayage de la pièce, et servir d'initialisation au problème de génération de trajectoire.

$$[\dot{X}_{\max} \quad \dot{Y}_{\max} \quad \dot{Z}_{\max} \quad \dot{\alpha}_{\max} \quad \dot{\beta}_{\max} \quad \dot{\gamma}_{\max}]^t = J_\theta [\dot{\theta}_{1\max} \quad \dots \quad \dot{\theta}_{6\max}]^t \quad \text{Eq. 206}$$

Nous avons donc un vecteur décrivant la vitesse maximale de l'effecteur pour une configuration. Mais nos indices doivent représenter les capacités du robot en vitesse tout le long d'une trajectoire et pas seulement en un point. Nos indices vont donc concaténer en une seule valeur, les informations de vitesse maximale calculées pour chaque point de passage de la trajectoire.

Afin de calculer nos indices, il faut alors disposer d'une trajectoire et de ses points de passage. Comme décrit dans la section 1.2, les différents indices de performances seront utilisés dans la stratégie de génération de trajectoire afin tout d'abord de sélectionner des zones de travail à partir de trajectoires locales entre capteur laser et pièce. Ainsi, au moment de calculer ces indices, une ébauche de trajectoire sera disponible sous la forme d'une trajectoire locale exprimée dans le repère de la pièce à numériser.

Nos indices sont calculés à partir d'une trajectoire exprimée dans le repère du robot, car ils qualifient les capacités du robot et sont basés sur la vitesse de l'effecteur. Il faut donc placer l'ébauche de trajectoire initialement exprimée dans le repère de la pièce, dans ce repère du robot. Une fois la trajectoire placée en position et en orientation dans le repère du robot, les indices peuvent être calculés. Pour établir ce placement de la trajectoire, nous définissons : la position du point de départ de la trajectoire dans l'espace du robot, et l'orientation de la direction d'avance en ce point.

Nous pouvons calculer ensuite les vitesses maximales en chaque point du trajet, puis les valeurs des indices qui en découlent. Pour toute position du point de départ et orientation de la direction d'avance, nous pouvons donc calculer les indices de vitesse moyenne et de vitesse minimale, qui permettront au générateur de trajectoire de respecter des contraintes imposées et d'optimiser la durée de l'opération de numérisation. Nous allons voir à présent comment calculer ces indices et donc comment concaténer les informations de vitesse maximale des points de passage.

4.1.3 Exemple de trajectoire d'ébauche et cas singulier

La présence d'un cas singulier dans le comportement du robot nécessite d'établir un exemple de trajectoire ébauche afin de l'illustrer. Cette singularité est une discontinuité de la posture du robot au cours de la trajectoire se produisant dans certaines configurations. Nous utilisons donc l'exemple de trajectoire présenté sur la figure 69, avec lequel nous illustrerons également les indices de vitesse minimale et de vitesse moyenne.

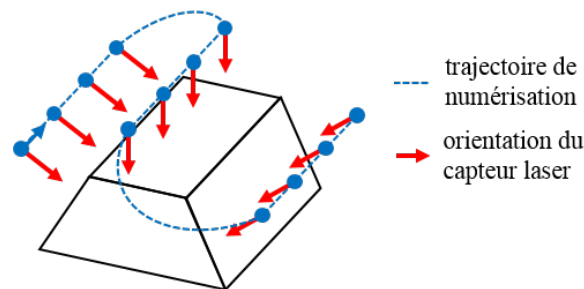


Figure 69 : Exemple de trajectoire en 3 passes pour numérisation

Cette trajectoire correspond à une trajectoire locale décrite dans le repère pièce, il faut donc fixer la position et l'orientation du point de départ dans l'espace de travail du robot. Puis, la position des points de passage et l'orientation du capteur laser vont permettre de déterminer les configurations du robot correspondantes (avec un modèle géométrique inverse). La direction d'avance elle change le long de la trajectoire et dépend donc du point considéré. Pour l'instant nous n'utilisons que des lignes droites pour passer d'un point de passage à l'autre, ce qui donne des directions d'avance simples. À partir de cette trajectoire en 3 passes, on peut calculer les vitesses maximales en tout point de passage (couple {configuration, direction}), et alors calculer des valeurs pour nos deux indices de vitesse.

Si l'on souhaite utiliser une trajectoire plus complexe pour passer d'un point à l'autre, il faudra prendre en compte plus que la seule direction d'avance au niveau des points de passage. Par exemple, pour les trajectoires courbes de la figure 69, et pour la recherche d'une vitesse maximale garantie, il faudrait rechercher le couple {configuration, direction} le long de la courbe qui donne la plus faible vitesse maximale et l'utiliser comme valeur pour le point de passage au début de cette courbe. D'une manière générale, nous avons simplifié le calcul des vitesses maximales sur les points de passage. Mais il est possible de prendre en compte les capacités d'accélération du robot et son comportement autour d'un point de passage pour obtenir des valeurs plus proches de ce que l'on peut réellement espérer. On peut même aller jusqu'à avoir une description continue de la vitesse maximale atteignable tout le long de la trajectoire et pas seulement sur les points de passage.

Lors du choix de posture pour le robot sur un point de passage, un cas particulier doit être traité avec attention. Chaque point de passage va imposer la position et l'orientation de l'effecteur du robot, mais le choix de posture pour les atteindre reste libre parmi 8 postures. Afin de ne pas perdre de temps, il est préférable de choisir une posture qui sera la même pour tous les points de passage. Une trajectoire simple avec une posture coude haut est représentée sur la figure 70.

Cependant, lorsque la trajectoire à effectuer est autour de l'axe 1 du robot, il n'est plus possible de conserver la même posture sans s'éloigner de la trajectoire initiale et perdre énormément de temps. Ce

cas particulier de trajectoire où la posture ne peut pas être conservée sans dégrader fortement la continuité de la trajectoire est présenté sur la figure 71.

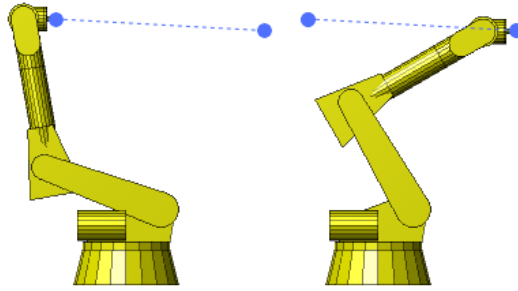


Figure 70 : Trajectoire simple, même posture

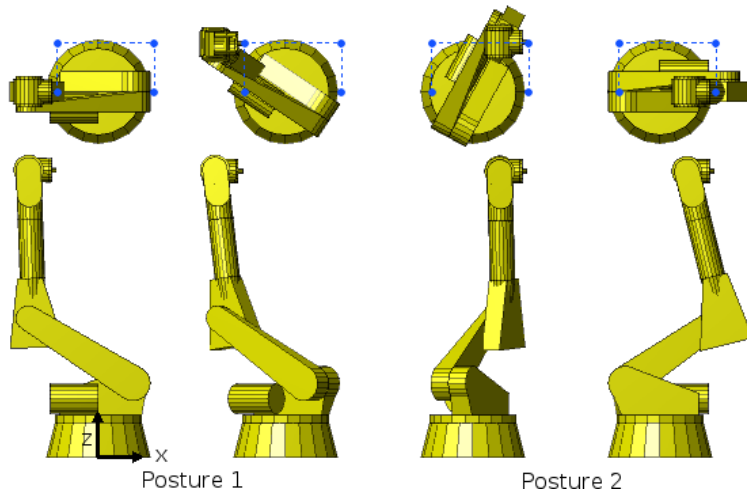


Figure 71 : Trajectoire avec changement de posture

Sur cette figure, on peut observer un changement de posture du robot lors d'un changement de signe suivant la direction X , avec un déport suivant la direction Y ($y \neq 0$), ce qui traduit le passage par une configuration singulière. La posture du robot est nécessairement différente afin de suivre la trajectoire. La posture 2 sur cette figure est alors complètement retournée par rapport à la posture 1 initiale coude haut. D'autres cas particuliers plus rares existent, où la posture du robot ne peut pas non plus être conservée. Lors du choix de posture du robot pour un point de passage, il faut donc être attentifs aux cas singuliers et vérifier la continuité de mouvement entre tous les points de passage.

4.1.4 Calcul des indices de vitesse moyenne et minimale

Les valeurs des indices sont calculées à partir des vitesses maximales atteignables sur tous les points de la trajectoire. Nous utilisons l'équation 206 afin de connaître ces vitesses maximales. Les vitesses maximales sur chaque articulation, requises dans cette équation, peuvent être obtenues sur une documentation du robot ou par des essais. La jacobienne peut ensuite être calculée pour chaque point de passage. Pour un point, on obtient alors un vecteur des vitesses maximales sur l'effecteur, dans le repère de la base du robot. De ce vecteur, on déduit par projection la vitesse maximale pouvant être générée par l'effecteur suivant la direction d'avance fixée Z_{dir} . Seule la vitesse en translation du vecteur nous intéresse ici.

$$V_{max} = [\dot{X}_{max} \quad \dot{Y}_{max} \quad \dot{Z}_{max}] Z_{dir} \quad \text{Eq. 207}$$

Cependant, chaque articulation peut être sollicitée à la vitesse maximale dans deux sens de rotation, ce qui change son influence sur l'effecteur. Il faut alors faire en sorte que chaque vitesse articulaire contribue à la vitesse de l'effecteur dans le même sens que la direction demandée, et ce, afin de vraiment obtenir la vitesse maximale. En effet les mouvements générés par chaque articulation pourraient s'opposer. Dans l'équation 207 le sens de l'influence de chaque articulation n'est pas maîtrisé ce qui amène dans certains cas à un résultat qui n'est pas la vitesse maximale pouvant être obtenue sur l'effecteur. Il faut donc reformuler cette équation et considérer à part l'influence de chaque articulation sur la direction d'avance. Pour cela il faut séparer chaque colonne J_{θ_i} de la jacobienne comme dans l'équation 208. Pour simplifier l'écriture, on ne prend que les 3 premières lignes de la jacobienne associées à la translation.

$$V_{\max} = |(J_{\theta_1} \cdot \dot{\theta}_{1\max}) \mathbf{Z}_{\text{dir}}| + |(J_{\theta_2} \cdot \dot{\theta}_{2\max}) \mathbf{Z}_{\text{dir}}| + \dots + |(J_{\theta_6} \cdot \dot{\theta}_{6\max}) \mathbf{Z}_{\text{dir}}| \quad \text{Eq. 208}$$

Ainsi nous nous assurons que chaque articulation contribue à la vitesse maximale dans le même sens. Nous allons à présent utiliser cette vitesse maximale pour calculer l'indice de vitesse moyenne et l'indice de vitesse minimale.

■ Dans un premier temps il faut vérifier si le robot est apte à respecter les contraintes en vitesses imposées sur la trajectoire. Pour cela nous utilisons l'indice de vitesse minimale avec lequel nous pourrions déterminer les configurations et directions d'avance de départ donnant des trajectoires qui respectent les contraintes en vitesse. L'indice de vitesse minimale est simplement la plus petite vitesse maximale observée sur les points de passage de la trajectoire, ce qui assure de respecter au moins cette vitesse sur tous les points. Pour une trajectoire de n points, l'expression de cet indice est donc :

$$I_{V_{\min}} = \min(V_{\max 1}, V_{\max 2}, \dots, V_{\max n}) \quad \text{Eq. 209}$$

Afin de calculer et d'illustrer cet indice nous utilisons la trajectoire ébauche de la figure 69. À partir de cette trajectoire simple et de la formule de l'indice de vitesse minimale, nous pouvons dresser des cartographies. Nous choisissons de représenter des cartographies 2D des positions de départ possibles pour la trajectoire. Dans les figures 72 et 73, deux orientations différentes sont choisies pour le point de départ. Les orientations de départ sont donc toujours les mêmes sur les deux cartographies, tout comme la posture du robot (coude haut) et la direction d'avance du point de départ.

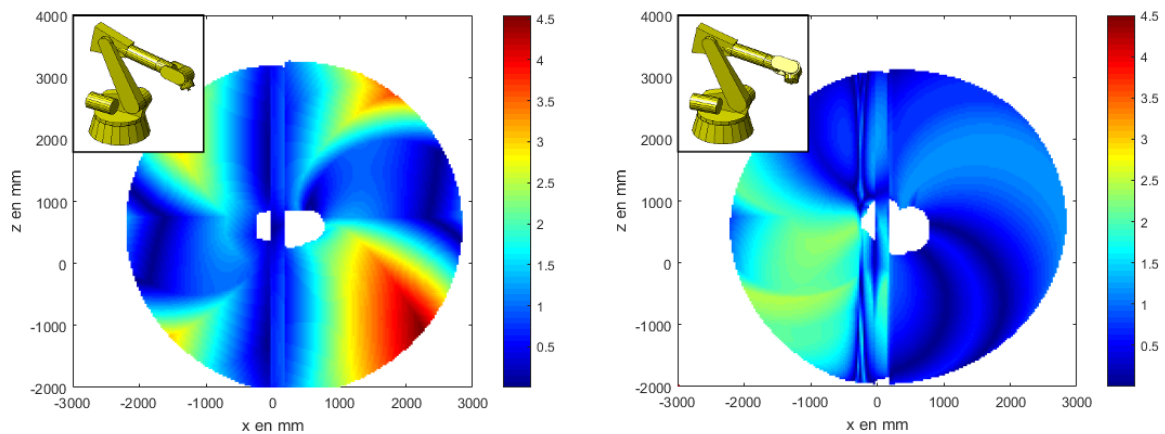


Figure 72 : Indice de vitesse minimale orientation 1, en m/s Figure 73 : Indice de vitesse minimale orientation 2, en m/s

On peut remarquer qu'un faible changement de l'orientation de la configuration de départ, et par conséquent de la configuration liée à chaque point de passage, amène de grandes différences sur les valeurs de cet indice. On peut également observer une zone rouge sur la figure 72 permettant d'atteindre une vitesse minimale jusqu'à 4 fois supérieure à d'autres zones. Cette zone disparaît avec la deuxième orientation de la figure 73. Il faudra alors choisir avec soin la position et l'orientation de la pièce à numériser afin de pouvoir respecter des contraintes sur la vitesse de la trajectoire. Par ailleurs, même si

cet indice semble très sensible, l'utilisation d'un autre modèle que le théorique a une influence minime. Comme pour l'essai avec l'indice de manipulabilité, nous avons modifié les valeurs des paramètres de l'ordre de 2mm pour les distances et de 1° pour les angles. La différence moyenne obtenue est de 0,04 m/s et la différence maximale observée est de 0,19 m/s. Il est donc inutile de chercher une modélisation complexe et précise pour l'expression de cet indice.

Par ailleurs, on peut observer des discontinuités verticales pour des abscisses x proches de 0, sur toute la hauteur des cartographies. Elles sont dues à la prise en compte du cas particulier cité précédemment, et correspondent à une position de la pièce telle que les points de passage de la trajectoire de la figure 72 n'ont pas tous le même signe suivant X . Pour ces positions de pièce, la posture du robot ne peut rester coude haut, ce qui donne des performances en vitesse différentes. En prenant en compte les cas singuliers, on s'assure d'avoir toujours une trajectoire fluide et continue, mais l'ensemble des valeurs que peut prendre l'indice de vitesse est alors discontinu.

■ Dans un deuxième temps, nous allons chercher à exploiter au maximum les capacités en vitesse du robot afin de réduire le temps de numérisation. Nous utilisons ici l'indice de vitesse moyenne que nous avons développé dans ce but. Cet indice repose sur la moyenne des vitesses maximales calculées sur les points de passage de la trajectoire. Cela donne alors une bonne estimation de la rapidité d'une trajectoire donnée, et permet de sélectionner la trajectoire minimisant le temps de numérisation.

Mais la distance entre les points de passages n'est pas toujours la même. Certaines portions du trajet doivent être détaillées par de nombreux points tandis que d'autres portions simples ne nécessitent qu'un point de départ et d'arrivée. Ainsi pour de nombreuses trajectoires, la densité des points de passages n'est pas homogène, et une simple moyenne des vitesses sur chaque point ne donne pas la vitesse moyenne le long de la trajectoire. On peut facilement prendre en compte cette densité en pondérant la valeur de vitesse maximale $V_{\max j}$ d'un point de passage j par la distance d_j qui sépare ce point du suivant, comme dans l'équation 210. Dans cette équation d_{tot} représente la distance totale parcourue le long de la trajectoire.

$$I_{V_{\text{moy}}} = \frac{d_1}{d_{\text{tot}}} V_{\max 1} + \frac{d_2}{d_{\text{tot}}} V_{\max 2} + \dots + \frac{d_n}{d_{\text{tot}}} V_{\max n} \quad \text{Eq. 210}$$

Comme pour l'indice de vitesse minimale, nous dressons les cartographies des points de départ pour la trajectoire, affichant la valeur de notre indice. Nous utilisons également la trajectoire en 3 passes de la figure 69 afin de générer les cartographies des figures 74 et 75 représentant l'indice de vitesse moyenne pour deux orientations différentes. Nous prenons toujours en compte le cas particulier de positionnement de la trajectoire qui rend les cartographies discontinues.

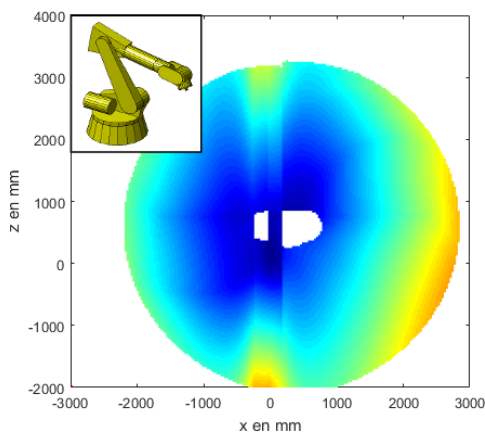


Figure 74 : Indice de vitesse moyenne avec discontinuité et orientation 1, en m/s

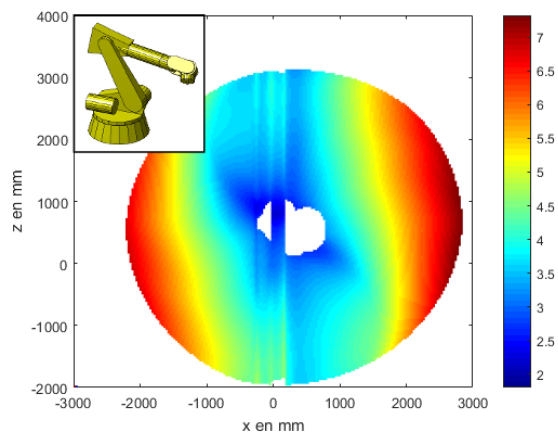


Figure 75 : Indice de vitesse moyenne avec discontinuité et orientation 2, en m/s

Ces cartographies permettent de bien illustrer la différence de répartition qu'il y a entre la vitesse minimale et la vitesse moyenne. Optimiser la vitesse minimale ne permettra pas d'obtenir la trajectoire la plus rapide. En effet, contrairement aux résultats sur la vitesse minimale, cette fois-ci c'est la deuxième orientation qui présente la génération de vitesse la plus avantageuse allant jusqu'à 7m/s en position bras tendu.

Nous avons donc deux indices bien distincts, qui donnent lieu à deux sélections différentes. L'indice de vitesse minimale permet de trouver les zones de travail garantissant une vitesse minimale, qui respecte les contraintes d'un cahier des charges. Puis dans ces zones, l'indice de vitesse moyenne permet de trouver la meilleure position et orientation de la pièce pour une vitesse minimale de l'opération.

4.1.5 Perspectives d'évolution

Les définitions que nous avons données pour nos indices sont simples afin de les mettre en place rapidement. Pour rendre ces indices plus représentatifs des performances réelles du robot, il faudrait mieux prendre en compte les trajets entre les points de passage ainsi que les capacités d'accélération des différentes articulations du robot. Pour l'instant seuls des calculs simples sur les points de passage sont effectués pour obtenir la vitesse maximale, mais cette vitesse n'est pas constante entre deux points et l'accélération des articulations ne permet pas forcément de l'atteindre. Une modélisation plus juste, mais aussi plus lourde devra prendre en compte ces accélérations, les trajets entre les points de passage, et le comportement du robot quand il approche les points de passage. La vitesse maximale au cours de la trajectoire ne sera donc pas seulement exprimée sur les points de passage, mais sur la totalité du trajet sous forme d'une fonction continue. Dans ce cas l'indice de vitesse minimale cherchera le minimum sur cette fonction et l'indice de vitesse moyenne sera calculé à l'aide d'une intégrale. Mais à ce stade nos études se focalisent sur nos définitions des indices de vitesse, donnant des informations suffisamment fiables et permettant une intégration et des calculs rapides dans le générateur de trajectoire.

4.2 Contrôle de la qualité de pose

Une autre caractéristique que nous devons maîtriser dans l'opération de numérisation est la qualité de positionnement du capteur laser. Nous nous intéressons donc aux capacités en positionnement du système porteur de l'opération, c'est-à-dire du robot. La qualité de pose va influencer la qualité de la numérisation et du nuage de points résultant. En effet une optimisation est réalisée sur le positionnement du capteur par rapport à la pièce à numériser, et s'écarter de cet optimum dégradera nécessairement le résultat. Mais la qualité de pose aura surtout une influence lorsque le modèle du robot sera utilisé pour le suivi de l'effecteur lors de l'opération de numérisation (au lieu du système de suivi externe C-Track). Lors de la numérisation, tout écart entre notre modèle et le comportement réel du robot engendra une erreur sur le recalage des lignes laser. Il faut donc bien maîtriser la qualité de positionnement de l'effecteur du robot c'est-à-dire : assurer une qualité de pose respectant la contrainte de qualité imposée sur tous les points de la trajectoire, puis pouvoir optimiser cette qualité pour une meilleure numérisation. Dans ce but nous allons créer un indice de qualité de pose, basé sur un indice existant, puis réaliser des cartographies afin d'illustrer cet indice.

La qualité de positionnement du robot n'est pas la même partout. Cette qualité regroupe plusieurs notions en robotique [ISO.9283 2015] : l'exactitude de pose et la répétabilité de pose. Tout d'abord l'exactitude de pose est l'écart entre la position souhaitée de l'effecteur et sa position réellement atteinte.

L'exactitude de pose peut être cartographiée de manière empirique par des mesures dans tout l'espace de travail du robot. Cette cartographie permet alors de corriger les erreurs d'exactitude sur la pose du robot. Il n'est ensuite pas utile de rechercher de meilleure zone d'exactitude avec la cartographie, car les erreurs sont corrigées. Par ailleurs, dans notre cas, nous n'utilisons pas de cartographie empirique mais

une modélisation élasto-géométrique qui permet également de compenser les erreurs d'exactitude. C'est pourquoi nous n'utilisons pas la notion d'exactitude en tant qu'indice de performance dans notre générateur de trajectoire.

L'autre notion associée à la qualité de positionnement est la répétabilité sur la pose de l'effecteur. Contrairement à l'exactitude, la répétabilité ne donne pas une valeur d'erreur à corriger, mais un écart type traduisant la partie aléatoire de l'erreur. Une cartographie de la répétabilité ne permettra donc pas de corriger une erreur, mais de visualiser et d'éviter les zones de travail où la position de l'effecteur est la plus incertaine.

On peut également citer l'inverse de la manipulabilité comme indice sur la qualité de pose, qui donne la sensibilité du robot dans une configuration donnée. Cette sensibilité exprime la variation de la pose de l'effecteur en fonction de la variation du positionnement angulaire de chaque articulation. Mais l'indice souffre toujours des mêmes défauts présentés dans la section 4.1.1. De plus on retrouve cette notion de sensibilité avec l'ellipsoïde de répétabilité présenté dans la section suivante. Le lien étroit présent entre la manipulabilité et l'ellipsoïde de répétabilité est présenté en annexe 4. Pour contrôler la qualité de pose du robot, nous nous focalisons donc essentiellement sur la répétabilité du robot.

4.2.1 Définition d'un ellipsoïde de répétabilité

Une approche permet de décrire la répétabilité sur l'effecteur, en utilisant la répétabilité de chaque articulation et la sensibilité transcrite par la matrice jacobienne. Cette approche est proposée par Brethé, qui met en œuvre un ellipsoïde de répétabilité pour décrire la répétabilité sur l'effecteur [Brethé et al. 2006]. Nous utilisons cet ellipsoïde pour définir notre indice de répétabilité du robot, dans la section 4.2.2 qui suit. La notion de répétabilité correspond à l'écart type résultant de multiples positionnements, et dont la répartition est souvent proche d'une gaussienne. On peut donc utiliser une loi normale pour modéliser ce phénomène. La méthode de Brethé est basée sur le cumul de lois normales correspondant aux répétabilités de chaque articulation. Le cumul de n lois normales, de variable aléatoire a_i et d'écart type σ_i , est décrit de la manière suivante :

$$f(a_1, a_2, \dots, a_n) = \frac{1}{\sigma_1 \sigma_2 \dots \sigma_n (2\pi)^{\frac{n}{2}}} e^{-\frac{1}{2} \left[\left(\frac{a_1}{\sigma_1}\right)^2 + \left(\frac{a_2}{\sigma_2}\right)^2 + \dots + \left(\frac{a_n}{\sigma_n}\right)^2 \right]} \quad \text{Eq. 211}$$

Dans notre cas nous étudions le cumul des répétabilités de chaque articulation d'un robot 6 axes. En s'appuyant sur l'équation précédente, et à partir des répétabilités σ_i des 6 articulations, Brethé écrit la densité de probabilité sur le positionnement de l'effecteur dans l'espace articulaire.

$$f(d\theta) = \rho \exp\left(-\frac{1}{2} [d\theta^t D^{-1} d\theta]\right) \quad \text{Eq. 212}$$

$$\text{avec } d\theta^t = [d\theta_1 \quad \dots \quad d\theta_6] \quad ; \quad \rho^{-1} = (2\pi)^3 \sigma_1 \dots \sigma_6 \quad ; \quad D = \text{diag}(\sigma_1^2, \dots, \sigma_6^2) \quad \text{Eq. 213}$$

où les variables aléatoires sont alors les variations angulaires $d\theta_i$ autour de chaque articulation. L'objectif est ensuite d'obtenir cette densité sur le positionnement de l'effecteur, mais dans l'espace des tâches. Il faut donc exprimer la densité dans le repère de base du robot, qui est cartésien. La matrice jacobienne articulaire J_θ permet le passage du domaine articulaire au domaine cartésien du repère de base. À l'aide de la jacobienne, l'équation 212 peut-être reformulée avec cette fois-ci, dX comme variable aléatoire regroupant les variations en position et orientation sur l'effecteur.

$$f(dX) = \rho \exp\left(-\frac{1}{2} [dX^t C^{-1} dX]\right) \quad \text{Eq. 214}$$

$$\text{avec } dX = J_\theta d\theta \quad ; \quad C = J_\theta D J_\theta^t \quad \text{Eq. 215}$$

À partir de l'équation 214, Brethé définit l'ellipsoïde de répétabilité comme une isodensité de probabilité. L'ellipsoïde va donc représenter une probabilité de retrouver l'erreur aléatoire de pose de

l'effecteur dans cet ellipsoïde. Nous allons chercher à retrouver la formule d'un ellipsoïde à partir de la définition de l'isodensité. Ce cheminement nous aidera dans la suite à définir notre indice de répétabilité. La densité de probabilité $f(d\mathbf{X})$ doit donc être constante, nous notons cette constante cst . Comme ρ est une constante, on peut définir la contrainte d'isodensité à partir de l'équation 214 avec la formulation suivante :

$$d\mathbf{X}^t C^{-1} d\mathbf{X} = cst \quad \text{Eq. 216}$$

On peut ensuite chercher les valeurs propres de C et diagonaliser cette matrice. On appelle alors K la matrice diagonale regroupant les valeurs propres λ_i de C , et P la matrice de passage associée. L'équation 216 devient :

$$d\mathbf{X}^t P K^{-1} P^{-1} d\mathbf{X} = cst \quad \text{Eq. 217}$$

En exploitant la matrice de passage, on change la variable aléatoire de base telle que $P^{-1} d\mathbf{X} = d\mathbf{X}'$. De plus K est diagonale, et développer cette équation nous donne finalement :

$$\frac{dx'^2}{\lambda_1} + \frac{dy'^2}{\lambda_2} + \frac{dz'^2}{\lambda_3} + \frac{d\alpha'^2}{\lambda_4} + \frac{d\beta'^2}{\lambda_5} + \frac{d\gamma'^2}{\lambda_6} = cst \quad \text{Eq. 218}$$

Ce qui est l'équation d'un ellipsoïde de dimension 6. Ici $[dx', dy', dz', d\alpha', d\beta', d\gamma']$ sont les variables en position et orientation dans la base de l'ellipsoïde. Nous avons donc la formulation de l'ellipsoïde de répétabilité de Brethé à partir de la définition de l'isodensité de probabilité. La longueur de chaque demi-axe de cet ellipsoïde est donnée par les valeurs propres λ_i de la matrice C et par la valeur constante cst . La valeur de cette constante permet donc de fixer la probabilité pour une valeur $d\mathbf{X}$ d'être dans l'ellipsoïde, et donc la probabilité pour l'erreur aléatoire sur la pose de l'effecteur d'être dans l'ellipsoïde.

À partir de cette définition donnée par Brethé, nous allons donner une formulation simple de la répétabilité de l'effecteur, pour pouvoir calculer notre indice de performance. Tout d'abord, la valeur constante cst de l'équation 216, est également valable dans l'espace articulaire, ce qui implique la formulation suivante :

$$d\theta^t D^{-1} d\theta = \left(\frac{d\theta_1}{\sigma_1}\right)^2 + \left(\frac{d\theta_2}{\sigma_2}\right)^2 + \dots + \left(\frac{d\theta_6}{\sigma_6}\right)^2 = cst \quad \text{Eq. 219}$$

Bien sûr l'ellipsoïde défini par cette équation est alors différent de l'ellipsoïde E , car dans le domaine articulaire. Dans ce domaine, chaque demi-axe est défini par la valeur de l'écart type σ_i d'une articulation et par la constante. Cet écart type étant dans notre cas une répétabilité, nous souhaitons que chaque demi-axe ait comme longueur cette valeur de répétabilité. Cela permet de simplifier l'interprétation des demi-axes de l'ellipsoïde E . Nous fixons donc la constante à 1 et les longueurs des demi-axes de l'ellipsoïde E correspondent donc aux valeurs de répétabilité dans le repère cartésien (position et orientation) :

$$(\sqrt{\lambda_1} \quad \sqrt{\lambda_2} \quad \sqrt{\lambda_3} \quad \sqrt{\lambda_4} \quad \sqrt{\lambda_5} \quad \sqrt{\lambda_6}) \quad \text{Eq. 220}$$

Ces valeurs de répétabilité ne sont donc pas données dans le repère de base du robot, mais dans celui de l'ellipsoïde E . Elles permettent tout de même de nombreux calculs sur les capacités de répétabilité du robot en une configuration. Elles servent notamment à la mise en place de notre indice de performance sur les capacités du robot en répétabilité que nous allons présenter.

4.2.2 Définition d'un indice pour la répétabilité

Nous souhaitons définir un indice permettant de comparer les capacités de répétabilité du robot sur plusieurs configurations. Contrairement à l'étude en vitesse, la qualité de mise en position de l'effecteur est recherchée suivant toutes les directions et pas seulement suivant une direction d'avance. Il faut donc prendre en considération les répétabilités suivant toutes les directions pour notre indice. La direction donnant la moins bonne répétabilité sera donc le facteur limitant. La répétabilité étant la valeur d'un écart type, une grande valeur de répétabilité implique une mauvaise répétabilité. Nous choisissons donc un indice qui recherche la valeur maximale de répétabilité pouvant être observée sur le positionnement de l'effecteur (pour une configuration donnée), et parmi les demi-axes de l'ellipsoïde de répétabilité.

$$\sigma_{\max} = \max(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \dots, \sqrt{\lambda_6}) \quad \text{Eq. 221}$$

Comme pour la création d'indices sur la vitesse, il faut prendre en compte non pas une configuration, mais tous les points de passage d'une trajectoire complète. Notre indice reposera donc sur l'équation 221, que nous appliquons pour chaque point de passage afin de déterminer la valeur de répétabilité maximale sur le positionnement de toute la trajectoire. Notre indice $I_{\sigma_{\max}}$ est nommé indice de répétabilité maximale, et est formulé de la manière suivante pour une trajectoire de n points :

$$I_{\sigma_{\max}} = \max(\sigma_{\max}(P_1), \sigma_{\max}(P_2), \dots, \sigma_{\max}(P_n)) \quad \text{Eq. 222}$$

Afin d'illustrer cet indice, nous traçons des cartographies de ses valeurs pour différentes configurations du robot, et pour la trajectoire en 3 passes de la figure 69. Dans ce but nous avons tout d'abord dû identifier les répétabilités de chaque articulation du robot. Nous avons donc réalisé 6 séries de 30 répétitions de mise en position de l'effecteur. Pour chaque série, une seule articulation est sollicitée afin de tester sa répétabilité. Les répétabilités mesurées pour chaque articulation sont présentées dans le tableau 23.

Axe 1	Axe 2	Axe 3	Axe 4	Axe 5	Axe 6
0,19	0,44	0,17	4,38	0,48	5,70

Tableau 23 : Répétabilités brutes des axes du robot, en 10^{-3}°

Mais il faut prendre en compte le moyen de mesure dans cette estimation. En effet le système de suivi externe utilisé (C-Track) a lui aussi une répétabilité lors du suivi d'une cible. Il faut donc recalculer la répétabilité de chaque axe du robot à partir de la connaissance que nous avons de la répétabilité du C-Track est des répétabilités brutes du tableau 23. Ces répétabilités brutes sont donc la combinaison de la répétabilité du C-Track et des répétabilités articulaires recherchées. Sachant que nous observons l'erreur autour de chaque axe, la variable aléatoire utilisée dans les lois normales est un angle. Or dans la section 1.5.2, nous avons mesuré et décrit la répétabilité du C-Track dans un repère cartésien $[\sigma_x, \sigma_y, \sigma_z]$. Nous avons donc converti ces écarts types du C-Track, en une répétabilité angulaire pour chaque axe, en fonction de la configuration du robot utilisée lors des mesures. Nous avons finalement un cumul de lois normales ayant la même variable aléatoire : une variation d'angle autour de l'articulation. L'écart type angulaire ou répétabilité angulaire, résultant du cumul des répétabilités du C-Track et du robot, peut être obtenu avec la formule de cumul de lois normales suivante :

$$\sigma_{\text{tot}} = \sqrt{\sigma_{\text{C-Track}}^2 + \sigma_{\text{robot}}^2} \quad \text{Eq. 223}$$

A l'aide de cette équation, des valeurs brutes du tableau 23, et des répétabilités angulaires calculées pour le C-Track variant entre 0,08 et $1,57 \cdot 10^{-3}^\circ$, on estime les valeurs de répétabilité des axes du robot. Ces nouvelles valeurs sont données dans le tableau 24.

Axe 1	Axe 2	Axe 3	Axe 4	Axe 5	Axe 6
0,07	0,43	0,13	4,27	0,09	5,48

Tableau 24 : Répétabilité identifiées des axes du robot, en $10^{-3\sigma}$

On peut noter une grande disparité entre les répétabilités de chaque axe du robot. Cela souligne bien l'importance d'un indice de répétabilité prenant en compte la répétabilité propre à chaque articulation. Nous pouvons ensuite, à partir de ces valeurs identifiées, tracer des cartographies de l'indice de répétabilité maximale. A des fins de comparaison, les configurations utilisées sont les mêmes que celles des cartographies des indices de vitesse.

Par ailleurs nous ne nous intéressons ici qu'à la répétabilité sur le positionnement de l'effecteur et pas sur l'orientation. Nous n'utilisons donc que la partie supérieure de la jacobienne et nous n'obtenons que 3 valeurs propres pour notre indice de répétabilité maximale. Si nous voulions également exploiter la répétabilité sur l'orientation de l'effecteur, il faudrait alors adimensionner la partie supérieure de la jacobienne avec une longueur caractéristique (à cause des différentes unités). Les cartographies suivantes représentent donc la répétabilité maximale en position sur toute la trajectoire en 3 passes, pour des orientations et postures fixes identiques à celles utilisées pour les indices en vitesse.

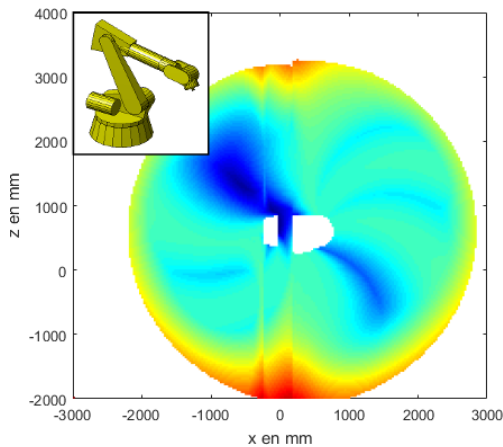


Figure 76 : Indice de répétabilité maximale pour 1^{ère} orientation, en mm

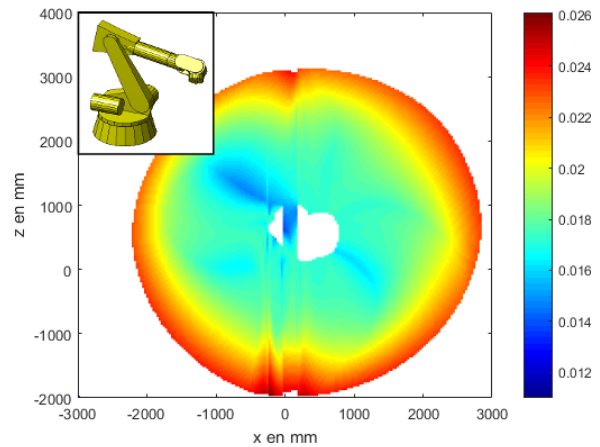


Figure 77 : Indice de répétabilité maximale pour 2^{ème} orientation, en mm

Les différences entre ces deux cartographies ne sont pas très importantes. La moins bonne répétabilité semble être atteinte pour une configuration bras tendu tandis que l'on va avoir de meilleurs résultats pour un effecteur plus proche du centre de l'axe 2 du robot. Mais dans l'étude en vitesse, c'est la configuration bras tendu qui semblait fournir les meilleurs résultats. Il faudra donc conjuguer l'étude de répétabilité et l'étude en vitesse afin d'une part de respecter des contraintes imposées en vitesse ou en qualité, puis d'autre part pour une optimisation de l'opération de numérisation en qualité et en vitesse. Nous verrons dans l'avant-dernière section de ce chapitre comment combiner vitesse et qualité. Par ailleurs sur ces figures, on peut remarquer les valeurs de notre indice sont plutôt éloignées de la répétabilité affichée par le constructeur qui est de 0,12mm. Après une vérification par des essais de répétabilité sur le robot, l'ordre de grandeur de nos figures est bien le bon et la valeur donnée par le constructeur doit contenir un fort coefficient de sécurité.

Par ailleurs, nous avons là aussi vérifié l'influence de l'utilisation d'un autre modèle que le théorique. Comme pour les essais avec les indices de vitesse et de manipulabilité, nous avons modifié les valeurs des paramètres de l'ordre de 2mm pour les distances et de 1° pour les angles. La différence moyenne obtenue est de 0,002mm et la différence maximale observée est de 0,009mm. Cette valeur maximale sur la différence de répétabilité peut sembler importante compte tenu des échelles des figures 76 et 77. Cependant c'est une valeur rarement atteinte, car la répartition de la différence a un très faible écart type de 0,002mm (comme la moyenne). Là encore une modélisation complexe et précise est inutile.

4.2.3 Identification des répétabilités des articulations et perspectives

Lors de nos essais de répétabilité sur les articulations du robot, nous avons pu constater que certains axes du robot avaient une faible influence sur la répétabilité observable de l'effecteur. En effet, pour les axes 1 et 5, la répétabilité du robot est masquée par celle du moyen de mesure. Il faudrait alors un moyen de mesure plus précis pour obtenir des valeurs plus fines pour ces axes. Mais ces axes sont peu influents sur la répétabilité finale de l'effecteur, qui intègre toutes les répétabilités. Notre étude ne requiert donc pas de mesure plus précise de ces valeurs de répétabilité.

Cependant nous avons pu remarquer d'autres phénomènes qui nécessiteraient des appareils de mesures plus précis pour être pris en compte. Durant nos essais nous avons par exemple noté l'influence de la vitesse du robot et de la décélération de l'axe avant d'atteindre la position ciblée. De grandes vitesses, des trajectoires courtes, et des décélération brusques ont tendance à détériorer la répétabilité lors des essais. Pour modéliser plus finement le phénomène de répétabilité, il serait donc intéressant d'étudier l'influence de la trajectoire sur la répétabilité. La répétabilité d'une articulation ne serait plus un simple écart type σ_i , mais une fonction dépendant de la vitesse, des accélérations et décélérations le long de la trajectoire ou même de la direction d'arrivée au point ciblé.

4.3 Limites en mouvement

Pour tous les indices définis précédemment, nous utilisons, d'une manière ou d'une autre, la modélisation du robot, soit avec le modèle géométrique direct ou inverse, soit avec la jacobienne articulaire. Ces indices transcrivent donc la modélisation du comportement du robot, que l'on essaye de rapprocher au mieux du comportement réel via nos travaux sur la modélisation. Mais il manque des contraintes importantes sur les possibilités de mouvement. L'utilisation seule d'une modélisation fournit certes le domaine accessible théorique du robot, mais ne donne en aucun cas les limites articulaires des axes du robot. Les courses réellement possibles sur chaque articulation vont grandement limiter le domaine accessible du robot. D'autre part la modélisation du robot ne donne pas non plus d'information sur les limites physiques dans la cellule de mesure comme le sol, le plafond ou l'enceinte. Ainsi un grand nombre de zones ne sont pas atteignables dans les cartographies générées avec les indices en vitesse et de répétabilité.

Afin de prendre en compte ces limites sur l'accessibilité des configurations du robot, il faut les intégrer aux indices précédents. Les indices ne retourneront alors plus de valeurs pour des configurations non atteignables.

4.3.1 Disponibilité articulaire sur trajectoire entière

Il est possible de formuler les limites sur les courses des articulations avec un indice de performance. Ces indices basés sur ces limites sont souvent appelés indices de disponibilité articulaire. En plus de prendre en compte les butées sur les articulations, ils donnent une valeur image de la quantité de déplacement encore possible avant d'atteindre une butée. Ils sont ainsi utilisés pour se tenir éloigné des butées des axes, et conserver une bonne liberté dans le mouvement. Parmi les nombreux indices traitant des limites articulaires, nous avons choisi de nous baser sur l'indice de disponibilité globale que nous rappelons ici [Moreno et al. 2010] :

$$\Delta\theta_i = \left| \theta_i - \frac{\theta_{imax} + \theta_{imin}}{2} \right| \quad \text{Eq. 224}$$

$$I_{dg} = \max_i \frac{\Delta\theta_i}{\Delta\theta_{imax}} \quad \text{Eq. 225}$$

Cet indice est simple à calculer en se basant sur la connaissance des limites articulaires de chaque axe

θ_{imax} et θ_{imin} . Lorsque la valeur de I_{dg} est proche de 1, c'est que le robot atteint une butée sur l'un des axes. En s'assurant d'avoir toujours $I_{dg} \leq 1$, nous avons l'assurance que les configurations ciblées restent dans le domaine accessible. Il peut être intéressant dans certains cas, de chercher à conserver du potentiel de mouvement en minimisant la valeur de I_{dg} , ce qui permet entre autres, de pouvoir appliquer des modifications mineures à la trajectoire ou de la prolonger, sans pour autant sortir du domaine accessible. Dans notre contexte de génération de trajectoire, nous n'avons pas besoin de conserver un potentiel de mouvement, la trajectoire générée devant être la trajectoire optimale pour une numérisation ciblée. Mais il faut tout de même ne pas sélectionner de configuration trop proche d'une butée, car un dépassement de la consigne nous ferait alors atteindre celle-ci. L'inégalité sur la valeur I_{dg} doit donc intégrer un coefficient de sécurité. Au vu des bonnes répétabilités des axes du robot pour les faibles vitesses de notre opération de numérisation (donc faibles dépassements), nous choisissons une faible sécurité de 0,1%. La valeur de I_{dg} devra donc respecter $I_{dg} \leq 99,9\%$, ce qui ne fait perdre qu'une petite partie du domaine accessible (moins de $0,3^\circ$ sur chaque axe). Pour des applications sollicitant de plus grandes vitesses sur l'effecteur et donnant de l'importance aux effets d'inertie, un coefficient de sécurité plus grand sera requis.

Comme pour les indices de vitesse et de qualité, nous définissons l'indice d'accessibilité sur une trajectoire de n points de passage P_i liés à une trajectoire. Cet indice d'accessibilité, que nous appelons encore indice de disponibilité sur trajectoire I_{dt} , est alors donné par l'équation 226 :

$$I_{dt} = \max(I_{dg}(P_1), I_{dg}(P_2), \dots, I_{dg}(P_n)) \quad \text{Eq. 226}$$

Nous pouvons à présent représenter cet indice d'accessibilité dans une configuration du robot et pour une trajectoire donnée. Nous utilisons la même trajectoire en 3 passes, ainsi que les mêmes postures et orientations utilisées dans les cartographies de vitesse et de qualité. Et comme pour ces cartographies, nous estimons cet indice I_{dt} pour l'ensemble des points de départ possible pour la trajectoire. Sur les cartographies des figures 78 et 79 l'indice devra vérifier $I_{dt} \leq 99,9\%$ pour qu'une trajectoire soit jugée acceptable (coefficient de sécurité).

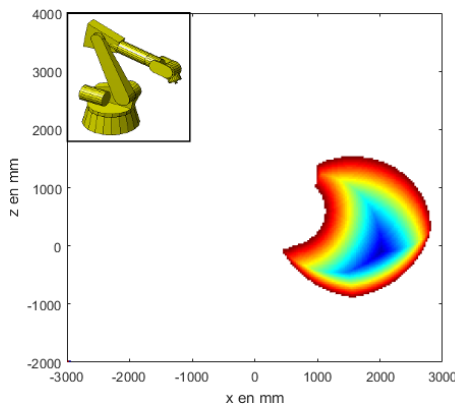


Figure 78 : Indice d'accessibilité pour 1^{ère} orientation, entre 0 et 1

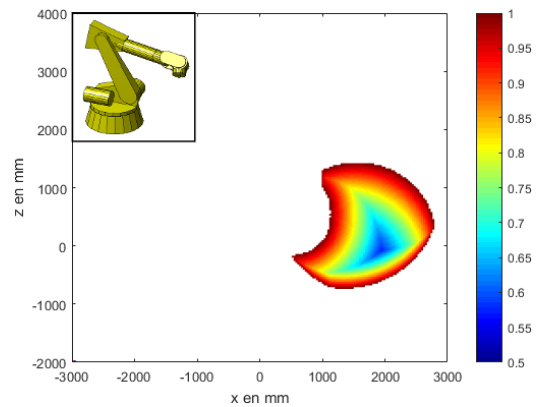


Figure 79 : Indice d'accessibilité pour 2^{ème} orientation, entre 0 et 1

Sur les figures 78 et 79 on peut voir que les limites articulaires restreignent énormément le domaine accessible. Les zones exploitables des cartographies de vitesse et de répétabilité sont alors très réduites. Sur ces figures on note également que l'orientation 1 laisse plus de liberté sur le choix de positionnement de la pièce à numériser. Le robot est donc plus à l'aise pour suivre une trajectoire avec la 1^{ère} orientation.

Afin de bien visualiser l'impact de la réduction du domaine accessible sur nos choix en vitesse et en répétabilité, nous allons appliquer les domaines trouvés avec l'indice d'accessibilité sur les cartographies de vitesse et de répétabilité on obtient alors les figures 80 et 81. Sur ces figures, on comprend bien l'importance de l'association des différents indices. Ces indices ne peuvent pas être

utilisés seuls, il faut tous les combiner, que ce soit pour respecter les contraintes imposées sur la trajectoire, ou pour l'optimisation de cette dernière. La combinaison des indices sera abordée dans l'avant-dernière section de ce chapitre.

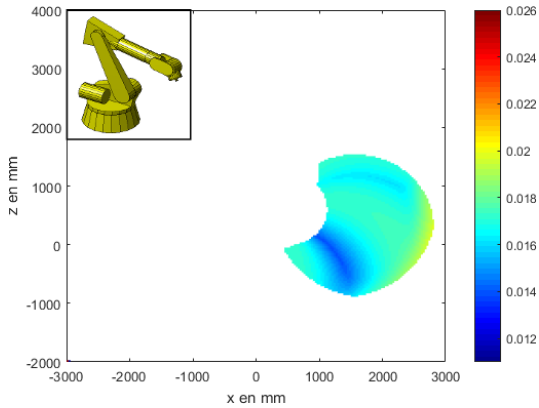


Figure 80 : Répétabilité maximale pour 1^{ère} orientation et domaine limité, en mm

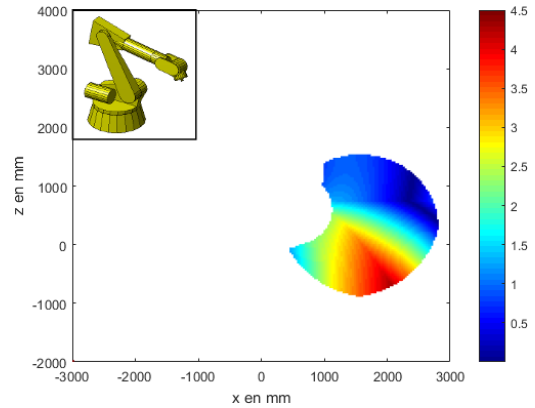


Figure 81 : Indice de vitesse minimale pour 1^{ère} orientation et domaine limité, en m/s

4.3.2 Barrières physiques et immatérielles

Il faut prendre en compte une autre limite au domaine accessible, qui n'est pas propre au robot, mais à la cellule de numérisation. L'environnement de la cellule, que ce soit le sol, le plafond, ou l'enceinte, crée des zones qui ne peuvent pas être utilisées même si le robot peut les atteindre. Ces limites physiques de l'environnement sont prises en compte par une limite immatérielle enregistrée dans la commande du robot afin de protéger ce dernier de collisions. Cette sécurité est souvent constituée d'équations de plans, qui ne peuvent être franchis et servant de barrières immatérielles. Ces plans sont définis manuellement, car ils dépendent de la mise en place du robot dans la cellule. Cette fois-ci la limite est donc décrite dans l'espace des tâches (cartésien) et non pas dans l'espace articulaire.

Il est donc possible de récupérer les équations de ces plans afin de créer une dernière limitation du domaine accessible lors de la génération de trajectoire. Mais cette limitation ne peut pas être traduite en un indice de performance, car elle ne porte pas sur les capacités du robot lui-même. L'ajout de cette limitation modifie une dernière fois les cartographies de vitesse et de répétabilité pour donner le résultat présenté sur les figures 82 et 83. Sur ces figures déjà bien tronquées par la disponibilité sur la trajectoire, seule la limite due au sol est apparente.

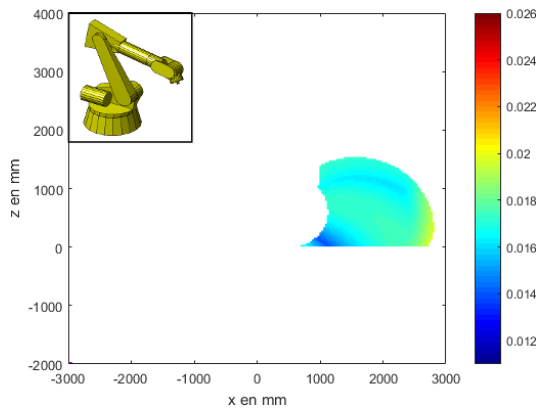


Figure 82 : Répétabilité maximale pour 1^{ère} orientation et domaine plus limité, en mm

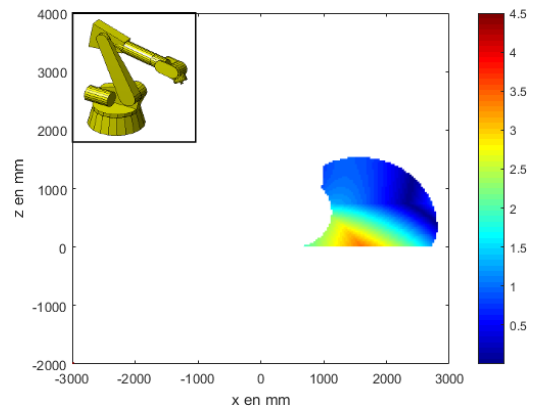


Figure 83 : Indice de vitesse minimale pour 1^{ère} orientation et domaine plus limité, en m/s

Les choix sur le positionnement de la pièce à numériser sont à présent très restreints. On peut voir que le sol élimine une partie intéressante du domaine accessible, que ce soit pour la vitesse ou pour la

répétabilité. Ces figures font ressortir une autre problématique non traitée ici qui est le placement du robot dans la cellule de numérisation. Si le robot était placé sur une base plus haute, il serait possible d'exploiter les zones intéressantes qui sont éliminées ici. Mais ces cartographies ne présentent qu'une seule posture et une seule orientation. Il y a beaucoup d'autres configurations possibles pour la numérisation de la pièce, qui peuvent être plus intéressantes en vitesse et en répétabilité.

4.4 Regroupement des indices de performance

Avant de chercher à exploiter les différents indices de performance du robot, il faut rappeler quelle est leur utilité dans le processus de génération de trajectoire. Tout d'abord, la CAO de la pièce à numériser, les capacités du capteur laser et d'éventuelles contraintes sur la vitesse ou la qualité (données par un cahier des charges), donnent une trajectoire ébauche dans le repère pièce. Les indices décrivant les capacités du robot doivent ensuite permettre de trouver les positionnements de la pièce dans l'espace des tâches du robot, dans lesquels cette trajectoire ébauche peut être suivie. Le premier rôle des indices de performance est d'estimer si la trajectoire ébauche est réalisable, pour un point de départ donné (position et orientation). Le résultat de cette étape doit être un ensemble de zones répondant à ce besoin.

Pour mener à bien cette étape dans la génération de trajectoire, les indices qui importent sont ceux assurant au minimum la vitesse imposée par le cahier des charges et/ou assurant la qualité minimale sur le positionnement requise pour la numérisation, le tout en vérifiant toujours l'accessibilité des points de passage. Il est donc nécessaire ici d'utiliser l'indice de vitesse minimale, l'indice de répétabilité maximale et les limitations du domaine accessible données par l'indice d'accessibilité et les barrières immatérielles. Et comme nous l'avons vu, tous ces indices doivent être combinés, car ils sont complémentaires, en particulier pour le domaine accessible.

Soit une trajectoire devant respecter une vitesse d'au moins V_c et une répétabilité d'au plus σ_c . L'ensemble E du domaine respectant les contraintes d'accessibilité, de vitesse et de qualité est obtenu par l'équation 227.

$$E = \{I_{\sigma_{\max}} < \sigma_c\} \cap \{I_{V_{\min}} > V_c\} \cap \{I_{dt} \leq 0,999\} \quad \text{Eq. 227}$$

L'ensemble E est alors l'ensemble des configurations de départ, c'est-à-dire positions, postures et orientations, permettant la numérisation d'une pièce avec des contraintes fixées et pour une trajectoire ébauche donnée. Afin d'illustrer cet ensemble, nous cherchons à le déterminer pour la trajectoire en 3 passes et les orientations et postures utilisées précédemment. Dans ce cas servant d'exemple, nous fixons comme contrainte une répétabilité maximale sur la mise en position de 0,018mm, ainsi qu'une vitesse minimale de 1m/s. Les figures 84 et 85 représentent l'ensemble E pour ce cas exemple.

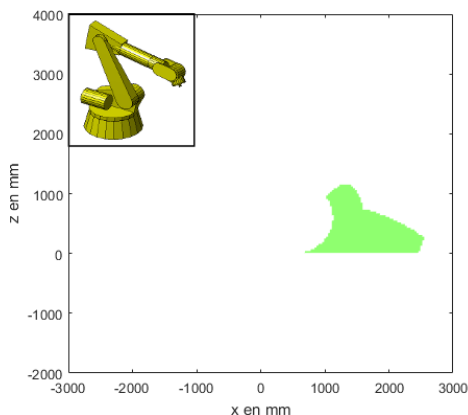


Figure 84 : Zone admissible pour la numérisation sous contrainte de la pièce, 1^{ère} orientation

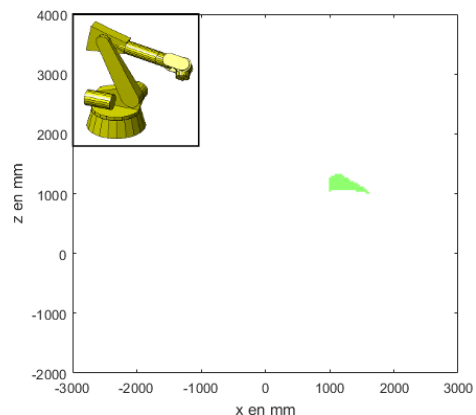


Figure 85 : Zone admissible pour la numérisation sous contrainte de la pièce, 2^{ème} orientation

Chaque orientation et posture différente donne une cartographie comme celles-ci, résultant en une définition des zones admissibles dans l'espace de travail du robot. Nous rappelons que ces cartographies 2D servent ici à illustrer les indices et la méthode, mais le générateur utilisera directement l'équation 227 pour déterminer l'ensemble E pour toute configuration et posture. Lors de cette étape de la génération de trajectoire, il est donc possible de déterminer si une pièce peut être numérisée avec notre cellule de mesure et pour les contraintes fixées.

Dans un deuxième temps, le générateur de trajectoire doit déterminer la trajectoire optimale pour la numérisation, en fonction d'un ou plusieurs critères d'optimisation. Cette phase d'optimisation recherchera une meilleure vitesse le long de la trajectoire, nous utilisons alors l'indice de vitesse moyenne. Nous utiliserons également l'indice de répétabilité maximale pour minimiser l'incertitude sur le positionnement de l'effecteur au cours de la trajectoire. Ces indices sont regroupés pour ne former qu'un seul et même critère dont nous chercherons la valeur minimale.

Pour créer ce critère, il faut tout d'abord normer les différents indices utilisés, car ceux-ci n'ont pas les mêmes unités. Tout d'abord il faut que les optimums des indices soient les mêmes. Pour la répétabilité, la meilleure valeur correspond bien à un minimum, mais pour la vitesse moyenne c'est un maximum. Il faut donc inverser l'indice de vitesse moyenne. Ensuite, l'identification des valeurs minimales et maximales pour chaque indice permet de changer l'échelle des valeurs de chaque indice pour qu'elles soient comprises entre 0 et 1. Ces valeurs minimales et maximales sont calculées sur l'ensemble E de tous les points de départ retenus par l'équation 227. Les indices rectifiés sont donc :

$$I'_{\sigma_{\max}} = \frac{I_{\sigma_{\max}} - \min_E(I_{\sigma_{\max}})}{\max_E(I_{\sigma_{\max}}) - \min_E(I_{\sigma_{\max}})} \quad \text{Eq. 228}$$

$$I'_{V_{\text{moy}}} = \frac{\frac{1}{I_{V_{\text{moy}}}} - \min_E\left(\frac{1}{I_{V_{\text{moy}}}}\right)}{\max_E\left(\frac{1}{I_{V_{\text{moy}}}}\right) - \min_E\left(\frac{1}{I_{V_{\text{moy}}}}\right)} \quad \text{Eq. 229}$$

Enfin il ne reste plus qu'à combiner ces indices rectifiés $[I'_{\sigma_{\max}}, I'_{V_{\text{moy}}}]$ en leur donnant une pondération en fonction de l'optimisation recherchée. On obtient finalement la configuration optimale par minimisation de cette combinaison, sur l'ensemble E déjà trouvé précédemment.

$$\min_E(c_{\sigma} I'_{\sigma_{\max}} + c_V I'_{V_{\text{moy}}}) \quad \text{Eq. 230}$$

Les pondérations c_{σ} et c_V de l'équation 230 sont libres. Bien sûr il est possible d'ajouter d'autres indices de performance à la formule si le besoin se présente. Par exemple il peut être nécessaire de prendre en compte le potentiel de mouvement ou même la consommation énergétique du robot. Pour ajouter un indice, il faudra cependant être capable d'évaluer sa valeur pour toute configuration et posture du robot.

En reprenant la trajectoire en 3 passes utilisée jusqu'à présent, il est maintenant possible de déterminer une configuration optimale. Nous allons par exemple chercher une configuration donnant la numérisation la plus rapide. Les pondérations choisies sont alors les suivantes :

$$c_{\sigma} = 0 \quad ; \quad c_V = 1 \quad \text{Eq. 231}$$

La configuration alors retenue dans l'ensemble E et pour ces pondérations se trouve suivant la première orientation (figure 84), pour une position du point de départ de $[2430 \ 0 \ 0]$. Cette configuration est donc celle minimisant la durée de la numérisation, et permettant la numérisation de la pièce pour les contraintes fixées.

4.5 Synthèse sur les indices exploités

La mise en place de notre cellule de mesure robotisée requiert une stratégie pour la génération de trajectoire de numérisation. Nous avons alors décrit dans le chapitre 1, la méthode à suivre pour obtenir cette trajectoire. À partir de la CAO de la pièce à numériser, des capacités du capteur laser et du robot, et de contraintes en vitesse ou en qualité sur la trajectoire la stratégie permet : d'estimer une trajectoire ébauche dans le repère de la pièce, d'en déduire l'ensemble des configurations robot pouvant exécuter cette trajectoire ébauche, de sélectionner dans cet ensemble, les configurations amenant à une numérisation optimale. Il est donc nécessaire de connaître les capacités du robot pour estimer la faisabilité de la trajectoire ébauche, puis pour trouver une trajectoire optimale. Dans ce quatrième chapitre, nous avons donc élaboré des indices permettant cette estimation et cette optimisation.

Les principales performances à chercher sur le robot pour une estimation de faisabilité et une optimisation sont : la vitesse de déplacement et la qualité de positionnement. Il faut également inclure l'accessibilité des configurations, qui est par principe indispensable. Nous avons alors sélectionné des indices décrivant ces capacités du robot, et permettant une étude de faisabilité de la trajectoire ainsi qu'une optimisation de trajectoire. Dans ce contexte nous avons proposé de nouveaux indices à partir d'indices existants, afin qu'ils puissent non pas qualifier une configuration du robot, mais une trajectoire entière placée dans l'espace de travail du robot. Les indices retenus sont :

- Indice de vitesse minimale : pour estimer la meilleure vitesse pouvant être respectée sur tous les points de la trajectoire. Utile pour vérifier la faisabilité de la trajectoire vis-à-vis de contraintes en vitesse.
- Indice de vitesse moyenne : pour estimer la vitesse moyenne le long de la trajectoire. Utile pour sélectionner la trajectoire la plus rapide dans l'espace de travail du robot.
- Indice de répétabilité maximale : permettant de déterminer la pire répétabilité de pose croisée au cours de la trajectoire. Ce qui permet de vérifier la faisabilité de la trajectoire vis-à-vis de la qualité de pose requise. Utile également pour sélectionner la trajectoire assurant la meilleure répétabilité dans l'espace de travail du robot.
- Indice d'accessibilité : afin d'estimer si tous les points de la trajectoire sont accessibles par le robot. Cet indice permet donc de savoir si une trajectoire donnée est réalisable par le robot. Il est également utile pour ne pas trop approcher les butées articulaires et permettre plus tard des modifications mineures de la trajectoire.

Ces indices sont ensuite combinés tout d'abord pour déterminer les domaines de travail desquels le robot sera apte à suivre la trajectoire ébauche. Puis pour sélectionner la trajectoire de ces domaines donnant les meilleurs résultats en vitesse ou en qualité. Par ailleurs, comme nous l'avons déjà souligné, d'autres indices peuvent être intégrés à cette stratégie en fonction du besoin, pour une optimisation différente. La mise en place de ces indices dans le générateur de trajectoire est simple. Cependant le temps de calcul de ces indices pour chaque configuration possible du robot peut être très long. Lors de cette mise en place, l'espace de travail du robot sera discrétisé de manière à réduire le nombre de configurations dans lesquelles calculer ces indices. Il faudra prendre soin de discrétiser avec un pas permettant de ne rien rater dans le comportement du robot, mais suffisamment large pour diminuer le temps de calcul.

Chapitre 5

Calibration du capteur laser plan dans la cellule robotisée

Dans ce dernier chapitre, la question de la mise en place concrète de la cellule de numérisation robotisée est traitée. Cette mise en place soulève le besoin de calibrer le capteur laser plan qui est au cœur de la cellule. La méthode proposée par le constructeur pour la calibration de notre capteur Zephyr n'est pas adaptée à notre application, dans laquelle le capteur doit être réorienté de manière continue au cours de la trajectoire. C'est pourquoi nous proposons dans ce chapitre une méthodologie adaptée à la calibration d'un capteur laser plan pour ce contexte.

Nous commençons par présenter les différents éléments intervenant dans la cellule robotisée ainsi que les communications requises entre ces éléments. Le besoin d'une calibration pour le capteur Zephyr ressort alors naturellement de cette présentation, mais il s'avère que la méthode existante et proposée par le fabricant donne des nuages de points incohérents dans notre contexte. Nous proposons donc deux méthodes de calibration complémentaires à la calibration du fabricant : une méthode rapide et pratique, et une méthode plus complexe mais de meilleure qualité. Ces méthodes se basent sur la numérisation de pièces de référence comme un plan ou un cylindre afin d'en déduire une matrice dite complémentaire permettant de compenser les défauts de la méthode de calibration existante. Ces deux matrices sont ensuite utilisées pour effectuer une qualification de la numérisation obtenue à l'aide de notre cellule robotisée. Durant cette qualification on peut observer de bons résultats en termes de bruit de mesure sur le nuage de points et de justesse.

5.1 Matériel et communication entre composants

La cellule de mesure robotisée que nous mettons en place est représentée sur la figure 86. L'opération de numérisation est réalisée par un capteur laser plan : le Zephyr (4). Dans le cas présent, qui est utilisé pour cette thèse, ce capteur est porté par notre robot sériel (1) qui assure sa mise en position et en orientation. La pièce à numériser est fixe sur le support de pièce (3). Un autre cas possible non traité ici, est d'avoir le capteur laser fixe, et la pièce à numériser portée par le robot. Une interface appelée KCP (2) permet de suivre et programmer les mouvements du robot. Des cibles fixées sur le Zephyr permettent de suivre ses mouvements par notre système de stéréovision : le C-Track (5). Chaque instrument de mesure (Zephyr et C-Track) dispose de son contrôleur (6), qui le relie à un ordinateur (7) et à un logiciel d'acquisition qui lui est propre : Polygonia pour le Zephyr et VXelements pour le C-Track.

Tous les éléments de la figure 86 sont utiles à l'opération de numérisation et à l'obtention d'un nuage de points représentatif de la pièce à numériser. L'élément le plus important est le capteur laser plan, qui permet de relever la trace laser issue de l'intersection entre le plan laser qu'il génère et la pièce à numériser. À l'aide de paramètres internes calibrés par le constructeur, la chaîne d'acquisition Zephyr – contrôleur - Polygonia fait l'acquisition de cette trace laser sous forme d'un ensemble de points 3D exprimés dans un repère interne au Zephyr. Ainsi grâce au système de déplacement (le robot), cette chaîne va pouvoir faire l'acquisition de différentes traces laser issues de différentes intersections, le but étant de couvrir toute la surface à numériser. Cependant, les différents ensembles de points 3D alors

obtenus sont exprimés dans le repère interne au Zephyr qui est mobile. C'est pourquoi le logiciel Polygonia requiert l'information $T_{R_{ext},R_{int}}$ sur la pose (position et orientation) du Zephyr dans un repère externe fixe, afin de réexprimer les ensembles de points dans ce repère. Ce passage du repère interne à un repère externe, pour un ensemble de n points 3D, est décrit par l'équation 232 (en notations homogènes).

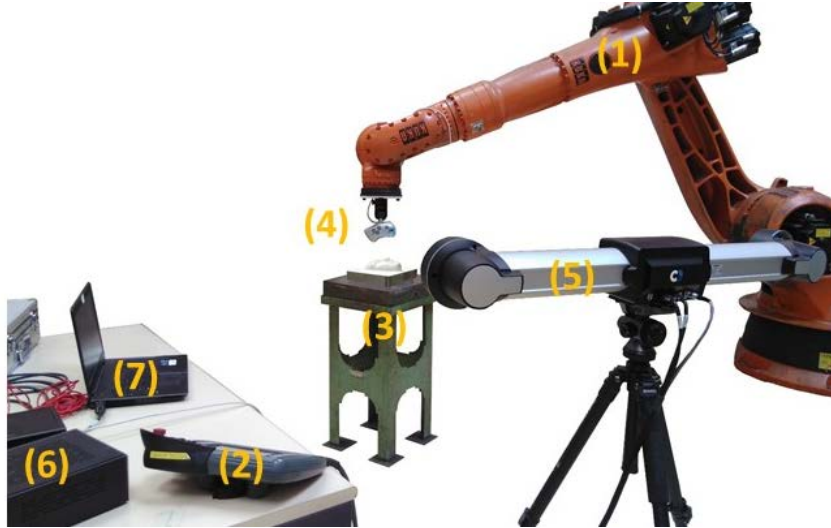


Figure 86 : Éléments de la cellule de numérisation

$$[pt_1 \dots pt_n]_{R_{ext}} = T_{R_{ext},R_{int}} [pt_1 \dots pt_n]_{R_{int}} \quad \text{Eq. 232}$$

À l'aide de cette équation, les points sont alors tous exprimés dans un même repère, puis réunis pour obtenir le nuage de points complet représentant la pièce à numériser. L'équation 233 donne l'expression d'un nuage à partir de m ensembles de points 3D.

$$\text{nuage} = \bigcup_{i=1}^m \{pt_{i,1} \dots pt_{i,n}\}_{R_{ext}} \quad \text{Eq. 233}$$

Dans notre contexte, le repère externe est le repère défini par le C-Track, et la pose du capteur Zephyr est fournie par le C-Track. Un autre mode de fonctionnement permet d'avoir cette pose à partir des coordonnées articulaires du robot et du modèle du robot. Ce mode permet de poursuivre la numérisation en cas de perte des cibles suivies par le C-Track, ou même d'effectuer une numérisation sans système de suivi externe.

Cette pose doit comprendre à la fois la position et l'orientation du capteur. En effet, le système de déplacement ici va modifier la position et l'orientation du capteur laser, contrairement à une MMT qui ne va agir que sur la position pour une orientation fixe. Cela permet d'effectuer une numérisation avec réorientation continue du capteur au cours de l'opération, afin de s'adapter aux contours de la pièce à numériser et d'obtenir un meilleur résultat de numérisation.

Ainsi l'information sur la pose complète du capteur doit être disponible à chaque acquisition d'un ensemble de points, pour pouvoir l'ajouter au nuage complet dans le logiciel Polygonia. Nous avons donc dû mettre en place une interface permettant la communication de cette information du C-Track vers Polygonia (figure 87). La communication entre le robot et Polygonia est traitée en perspective.

L'interface que nous avons développée, permet de mettre en relation le logiciel VXelements fournissant l'information sur la pose du Zephyr, avec le logiciel Polygonia effectuant la fusion des ensembles de points acquis, dans le repère externe. Les détails sur le développement de cette interface sont donnés dans l'annexe 5.

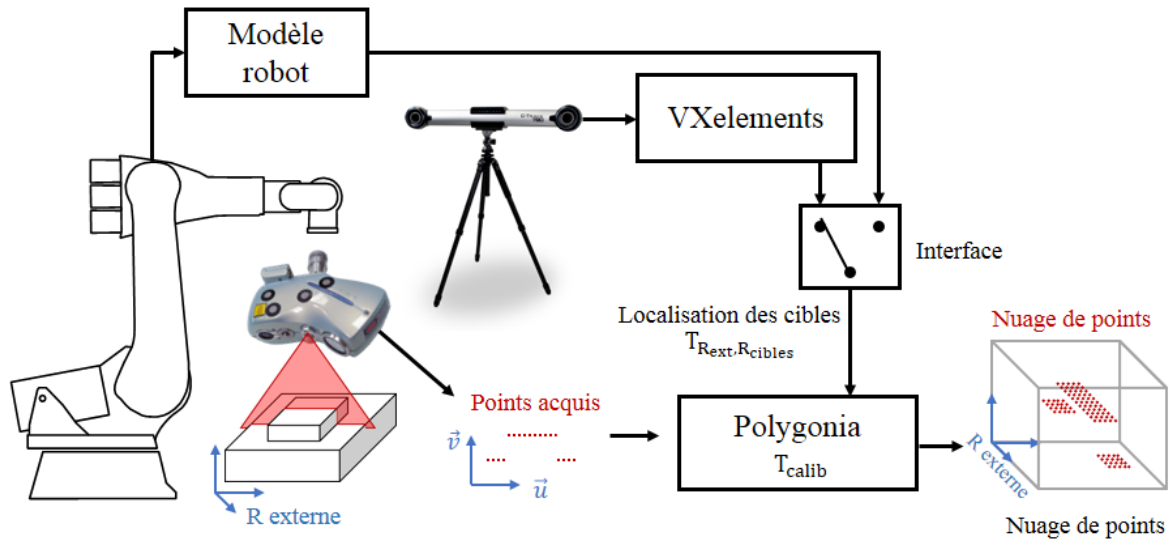


Figure 87 : Interface transmettant la position et l'orientation de l'ensemble de cibles, donnée par le C-Track ou le robot

Cependant, le C-Track ou l'ensemble {coordonnées articulaire + modèle robot} ne fournissent que la pose de l'ensemble des cibles disposées sur le Zephyr, c'est-à-dire une matrice $T_{R_{ext}, R_{cibles}}$ au lieu de la matrice $T_{R_{ext}, R_{int}}$. Or Polygonia utilise la pose du repère interne au Zephyr dans lequel sont exprimés les points acquis. Il est donc nécessaire de connaître la matrice permettant de passer du repère associé à l'ensemble des cibles au repère interne au Zephyr. Une méthode de calibration proposée dans le logiciel Polygonia permet d'identifier cette matrice de passage et de l'intégrer directement dans le logiciel. Cette matrice de passage T_{calib} est appelée matrice de calibration, et l'équation 232 permettant de réexprimer les points acquis devient :

$$[pt_1 \dots pt_n]_{R_{ext}} = T_{R_{ext}, R_{cibles}} T_{calib} [pt_1 \dots pt_n]_{R_{int}} \quad \text{Eq. 234}$$

La mise en place de cette calibration et une première numérisation d'une pièce test sont présentés dans l'annexe 6. Le nuage résultant de cette première numérisation est présentée sur la figure 88, et montre que la méthode de calibration proposée présente des défauts.

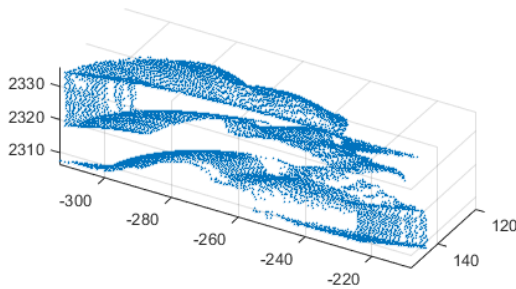


Figure 88 : Numérisation du visage, vue isométrique



Figure 89 : Visage usiné servant de pièce test

L'étude de l'annexe 6 permet de conclure que la partie orientation de la matrice de calibration T_{calib} n'est pas correcte, mais il n'est pas possible de la modifier car elle est comprise dans le logiciel Polygonia et n'est pas accessible. Nous choisissons donc d'ajouter une matrice T_{com} complémentaire à cette matrice de calibration, que nous intégrons dans notre interface juste avant l'envoi de l'information sur la pose de l'ensemble de cibles à Polygonia. Cet ajout est décrit sur la figure 90 qui reprend la figure 87. Cette matrice complémentaire est une matrice de rotation qui va compenser les défauts d'orientation de la matrice de calibration.

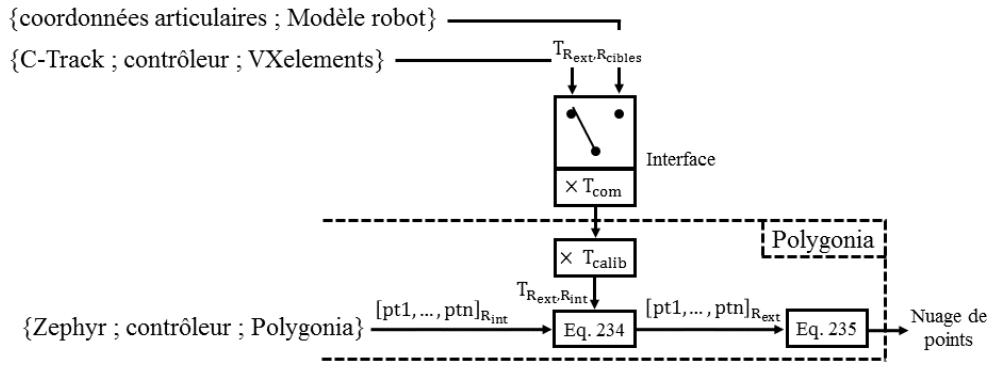


Figure 90 : Matrice complémentaire dans notre interface et calculs internes à Polygonia

La pose du Zephyr dans le repère externe s'exprime finalement par l'équation suivante :

$$T_{R_{ext},R_{int}} = T_{R_{ext},R_{cibles}} T_{com} T_{calib} \quad \text{Eq. 235}$$

Nous avons donc établi une liaison entre notre système de suivi et le Zephyr au travers de notre interface. L'information sur la pose de l'ensemble des cibles disposées sur le Zephyr peut donc être utilisée pour réexprimer les points numérisés dans un repère commun. Cependant la calibration de base du capteur Zephyr, faisant le lien entre ces cibles et le repère interne au Zephyr, donne un résultat aberrant sur cette fusion des points acquis. Nous devons donc identifier une matrice complémentaire permettant de rectifier ce défaut lié à l'information sur l'orientation du Zephyr. Dans la section suivante, nous présentons 2 méthodes différentes permettant d'obtenir cette matrice complémentaire.

5.2 Calibration entre modèle de cibles et capteur laser

5.2.1 Énoncé du problème et compensation

La section précédente conclue sur un défaut dans la fusion des lignes laser dans le repère externe due à une mauvaise connaissance de la véritable orientation du capteur Zephyr dans le repère externe par le logiciel Polygonia. Nous choisissons alors d'appliquer une matrice complémentaire, modifiant l'information sur l'orientation du Zephyr qui est envoyée au logiciel. Notre objectif est alors de créer une méthode de calibration permettant d'identifier cette matrice complémentaire. Dans les faits, deux méthodes sont développées : une méthode plus rapide et une méthode de meilleure qualité.

La seule donnée observable et utilisable pour nos méthodes de calibration est le nuage de point résultant d'une numérisation. Nous générons donc des nuages de points avec notre cellule de mesure comprenant la matrice de calibration de Polygonia et à partir d'une pièce étalon de géométrie connue. Les nuages ainsi générés présentent les défauts déjà mentionnés, et une étude comparative entre ces nuages et la géométrie théorique de l'étalon permet d'identifier les valeurs de notre matrice complémentaire. Les deux méthodes développées sont génériques et permettent de déterminer l'orientation de tout capteur laser plan dans un repère externe.

Il sera également intéressant, dans cette phase de calibration et d'identification, de répondre à un second objectif en rapport avec la modélisation de la cellule. Les chapitres 2 et 3 ont permis de mettre en place une modélisation donnant la pose de l'ensemble de cibles associé au Zephyr dans le repère externe. Mais la donnée essentielle est plutôt la pose du plan laser du Zephyr par rapport à la pièce à numériser afin de pouvoir optimiser l'angle et la distance de numérisation. Il faut donc ajouter dans la modélisation, la configuration de ce plan laser et de ses limites dans le repère de l'ensemble de cibles. L'identification de la configuration du plan laser par rapport aux cibles est effectuée à la suite de l'identification de la matrice complémentaire, car elle requière des informations qui seront alors disponibles.

5.2.2 Calcul de la matrice complémentaire

La matrice complémentaire à identifier T_{com} , de dimensions 4×4 , est une matrice uniquement de rotation. Nous recherchons donc sa partie en orientation Q_{com} de dimensions 3×3 . Cette matrice complémentaire est calculée à partir de la numérisation d'une pièce servant de référence, dont la géométrie et la disposition sont connues. L'objectif est d'avoir une bonne orientation de chaque ensemble de points acquis, avant de les fusionner en un nuage de points dans le repère externe. L'étude sur l'orientation n'est donc pas faite sur le nuage final entier, mais sur chaque ensemble de points. Nous cherchons donc à recaler l'orientation des ensembles de points, sur une orientation de référence connue grâce à la pièce de référence numérisée. Deux pièces de référence sont utilisées : un plan étalon et un cylindre étalon.

Tout d'abord l'intersection entre le plan laser du capteur Zephyr et le plan étalon à numériser correspond à une trace laser de forme linéaire. L'acquisition de cette ligne laser donne un ensemble de points formant une ligne. Puis, malgré les défauts sur l'information d'orientation du Zephyr, cette ligne reste une ligne dans le nuage de points final, que l'on peut facilement isoler. Il est alors possible pour chaque ensemble de points i , d'en déduire un vecteur directeur Z_{Li} par minimisation. Avec une numérisation à orientation constante, on peut alors calculer la moyenne de ces directions Z_{Li} , pour obtenir le vecteur $Z_{LigneAcquise}$ donnant la direction moyenne des lignes acquises avec défaut, dans le repère externe. On peut ensuite comparer cette direction acquise, avec la véritable direction de la trace laser (ligne laser) laissée sur le plan étalon, dans le repère externe et notée $Z_{LigneRef}$. L'obtention de cette direction de référence est détaillée plus loin, pour chacune de nos deux méthodes.

D'autre part, l'intersection entre le plan laser du Zephyr et le cylindre étalon à numériser correspond à une portion d'ellipse. L'acquisition de cette trace laser donne un ensemble de points dont la forme importe peu, mais dont l'étendue permet de calculer un plan des moindres carrés de bonne qualité. Ce plan des moindres carrés issu d'un ensemble de points acquis correspond donc au plan laser du Zephyr. Pour chaque ensemble de points i du nuage final, on peut donc calculer la direction de ce plan Z_{Pi} par minimisation (moindres carrés). Par une numérisation à orientation constante, on peut calculer la moyenne de ces directions, pour obtenir le vecteur $Z_{PlanAcquis}$ donnant la direction moyenne du plan laser acquis avec défaut, dans le repère externe. Ici aussi nous cherchons ensuite à comparer cette direction avec la véritable direction du plan laser $Z_{PlanRef}$ utilisée au cours de la numérisation du cylindre étalon.

L'orientation du capteur Zephyr doit donc être fixe durant nos identifications. Nous travaillons uniquement sur la comparaison entre une direction acquise et une direction de référence. Soit $Q_{passage}$ la matrice de passage entre une direction acquise Z_{Acquis} et une direction de référence Z_{Ref} :

$$Z_{Ref} = Q_{passage} Z_{Acquis} \quad \text{Eq. 236}$$

L'équation 236 est exprimée sans notation homogène (vecteurs de dimension 3). Cette matrice de passage est donc une matrice de rotation qui ramène la direction acquise avec défauts sur la direction de référence. En supposant que la direction de référence corresponde à la direction d'une ligne acquise sans défaut, c'est-à-dire acquise avec une matrice complémentaire parfaite, on peut l'exprimer à l'aide des équations 232 et 235 :

$$Z_{Ref} = Q_{R_{ext}, R_{cibles}} Q_{com} Q_{calib} Z_{int} \quad \text{Eq. 237}$$

Dans cette équation, Z_{int} est la direction d'un ensemble de points $[pt_1 \dots pt_n]_{R_{int}}$ acquis dans le repère interne du Zephyr. Ici, Z_{int} correspond soit à la direction de la ligne laser dans le cas de la numérisation du plan étalon, soit à la normale du plan laser dans le cas du cylindre étalon. Pour rappel, Z_{Ref} et Z_{Acquis}

sont eux exprimés dans le repère externe. De même on a l'expression de la direction Z_{Acquis} , sans Q_{com} qui est la partie orientation de la matrice complémentaire :

$$Z_{Acquis} = Q_{R_{ext}, R_{cibles}} Q_{calib} Z_{int} \quad \text{Eq. 238}$$

Les équations 236, 237 et 238 nous permettent d'écrire la formule simplifiée suivante :

$$Q_{com} = Q_{R_{ext}, R_{cibles}} {}^t Q_{passage} Q_{R_{ext}, R_{cibles}} \quad \text{Eq. 239}$$

On se permet ici d'écrire directement la transposée au lieu de l'inverse car la matrice $Q_{R_{ext}, R_{cibles}}$ est une matrice de rotation (donc orthogonale). Nous obtenons ainsi l'expression de Q_{com} définissant la rotation de la matrice complémentaire. La rotation permettant le passage de la direction Z_{Acquis} , à la direction Z_{Ref} , est simplement définie par un axe de rotation (direction) et un angle. La matrice de rotation $Q_{passage}$ correspondante est alors obtenue avec la formulation suivante, qui décrit une matrice de rotation $M(\theta, \mathbf{u})$ de dimensions 3x3, à partir d'un axe \mathbf{u} de rotation (normé) et d'un angle θ :

$$M(\theta, \mathbf{u}) = \begin{bmatrix} u_x^2(1-c) + c & u_x u_y(1-c) - u_z s & u_x u_z(1-c) + u_y s \\ u_x u_y(1-c) + u_z s & u_y^2(1-c) + c & u_y u_z(1-c) - u_x s \\ u_x u_z(1-c) - u_y s & u_y u_z(1-c) + u_x s & u_z^2(1-c) + c \end{bmatrix} \quad \text{Eq. 240}$$

$$c = \cos \theta \quad ; \quad s = \sin \theta \quad ; \quad \mathbf{u} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$$

Il est cependant souvent plus pratique de manipuler la direction d'un axe (3 données) et un angle (1 donnée) plutôt qu'une matrice de rotation (9 données), notamment lors d'une implémentation dans du code. On peut donc exprimer Q_{com} sous forme d'un axe de rotation et d'un angle. Il faut alors adapter l'équation 239 en une autre formulation équivalente donnée dans l'équation 241. La démonstration de cette équivalence peut être faite en développant les expressions des deux équations.

$$\text{Pour } Q_{passage} = M(\theta, \mathbf{u}) \text{ on a } Q_{com} = M(\theta, Q_{R_{ext}, R_{cibles}} {}^t \mathbf{u}) \quad \text{Eq. 241}$$

Ainsi, nous estimons tout d'abord la direction de l'axe de rotation et l'angle permettant le passage de Z_{Acquis} à Z_{Ref} . La direction de l'axe de rotation s'obtient par un produit vectoriel de ces vecteurs, et l'angle s'obtient par un cosinus inverse sur le produit scalaire des deux vecteurs. Puis, avec l'équation 241 et la connaissance de l'orientation fixe des cibles dans le repère externe au cours de notre identification ($Q_{R_{ext}, R_{cibles}}$), nous déduisons l'ensemble {direction axe ; angle} de la rotation Q_{com} . Enfin, nous intégrons la matrice complémentaire T_{com} dans notre programme interface pour modifier la partie orientation de la pose du Zephyr envoyée au logiciel Polygonia.

Cependant, une deuxième identification doit être effectuée, car la direction Z_{Ref} utilisée comme référence ne permet pas de fixer la rotation autour de cette même direction. La définition de l'orientation de la matrice complémentaire est donc incomplète. Il faut alors identifier une deuxième rotation Q_{com2} définie également par un ensemble {direction axe2 ; angle2}, qui s'ajoutera à la première rotation déjà identifiée.

Afin d'identifier cette deuxième rotation, on suppose la matrice complémentaire T_{com} déjà en place dans notre interface, comprenant la rotation Q_{com} . Les défauts de numérisation pour cette deuxième identification sont donc déjà partiellement corrigés. Cette deuxième rotation Q_{com2} à identifier doit fixer la rotation laissée libre autour de Z_{Ref} . Afin de solliciter cette rotation libre et de ne pas altérer la modification d'orientation déjà effectuée par Q_{com} , l'axe de rotation de $M_{passage2}$ est fixé comme étant Z_{Ref} . La référence Z_{Ref2} utilisée pour cette deuxième identification est bien sûr différente de Z_{Ref} . Afin d'avoir Z_{Ref} comme axe de rotation de $M_{passage2}$, nous choisissons Z_{Ref2} orthogonal à Z_{Ref} . De même

la direction $Z_{Acquis2}$ doit elle aussi être orthogonale à Z_{Ref} (une projection sera faite si besoin). Ainsi, avec Z_{Ref2} et $Z_{Acquis2}$, l'axe de rotation u_2 est nécessairement Z_{Ref} , et l'angle θ_2 est calculé comme précédemment à l'aide d'un produit scalaire. Le calcul de la rotation Q_{com2} est alors défini comme suit :

$$\text{Pour } M_{\text{passage}2} = M(\theta_2, u_2) \text{ on a } Q_{com2} = M(\theta_2, Q_{com} {}^t Q_{R_{\text{ext}}, \text{cibles}} {}^t u_2) \quad \text{Eq. 242}$$

On redéfini alors la matrice complémentaire avec la formulation suivante, pour prendre en compte les deux rotations Q_{com} et Q_{com2} :

$$T_{com} = \begin{bmatrix} Q_{com} \cdot Q_{com2} & 0_{\times 3} \\ 0_{\times 3} & 1 \end{bmatrix} \quad \text{Eq. 243}$$

Nous décrivons dans la suite 2 méthodes de calibration utilisant ce raisonnement, chacune utilisant des références différentes permettant de calculer la matrice complémentaire finale. Pour répondre au second objectif, nous verrons également comment déterminer le positionnement et les limites du plan laser du Zephyr dans le repère des cibles.

5.2.3 1^{ère} méthode : calibration rapide sur robot

Le but de cette première méthode est de pouvoir réaliser notre calibration avec le capteur monté sur le robot pour plus de rapidité et de praticité. Le capteur laser doté de cibles est donc fixé sur l'effecteur du robot. La méthode consiste ensuite à trouver les rotations de la matrice complémentaire, permettant de corriger les défauts visibles sur le nuage de point résultant.

Cette première méthode utilise uniquement un plan comme référence. Ce plan est disposé dans deux configurations différentes présentées sur les figures 91 et 92, pour définir les directions de référence Z_{Ref} et Z_{Ref2} , et pour effectuer l'identification des deux rotations de la matrice complémentaire Q_{com} et Q_{com2} .

Cependant dans cette première méthode, aucune information n'est disponible sur la configuration du plan étalon car il n'est pas possible de mettre en position ce plan de façon connue dans le repère externe. La seule information disponible est le nuage de points issu d'une numérisation. Il est donc impossible d'avoir la direction théorique Z_{Ref} de la ligne laser (intersection plan laser et plan étalon). L'alternative pour laquelle nous optons est de travailler avec la normale du plan de étalon Z_{Ref}' , car il est possible d'obtenir la direction de cette normale à partir du nuage de points avec défauts. Pour cela il faut numériser le plan étalon en plusieurs passes, puis effectuer une minimisation au sens des moindres carrés pour déduire des points acquis le plan de référence. Avec suffisamment de passes et un plan de grande taille, la normale identifiée est alors bien représentative du plan étalon malgré les défauts. Cette normale identifiée à partir de la numérisation sert alors de direction de référence Z_{Ref}' .

Avec la normale du plan comme référence, il n'est alors plus question de chercher la matrice de passage entre la direction moyenne des lignes acquises Z_{Acquis} et cette normale Z_{Ref}' . Par contre, il est certain que tous les points acquis devraient se trouver dans le plan de normale Z_{Ref}' de même que les lignes acquises. Dans ce cas, la matrice de passage recherchée est la matrice qui va ramener la direction Z_{Acquis} dans le plan de normale Z_{Ref}' . La direction u de l'axe de rotation de M_{passage} est alors le produit vectoriel entre Z_{Acquis} et Z_{Ref}' , et l'angle θ est obtenu à l'aide du produit scalaire :

$$\theta = \cos^{-1}(Z_{Acquis} \cdot Z_{Ref}') - 90 \quad \text{Eq. 244}$$

À l'aide de θ et u , on calcul la première rotation Q_{com} de la matrice complémentaire. On applique alors la matrice complémentaire dotée de cette première rotation dans notre interface, afin d'identifier ensuite la deuxième rotation Q_{com2} . Et comme le montre la figure 91, cette deuxième rotation est d'un angle φ autour de la normale Z_{Ref}' pour que les points acquis correspondent à la trace laser.

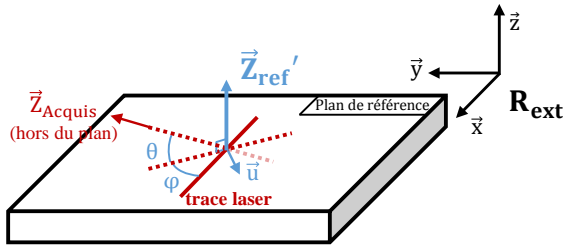


Figure 91 : Points acquis et trace laser dans le repère externe, identification de Q_{com}

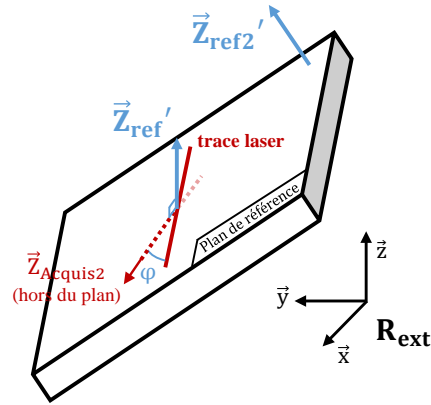


Figure 92 : Points acquis et trace laser dans le repère externe, identification de Q_{com2}

Dans les faits, chercher la rotation pour qu'un vecteur soit dans un plan, fixe une orientation de moins dans la matrice complémentaire que de chercher la rotation d'un vecteur vers un autre vecteur. Ici la rotation Q_{com} laisse deux rotations libres : la rotation autour de Z_{Ref}' , et la rotation autour de la trace laser elle-même. Mais la rotation autour de la ligne laser laisse cette trace laser invariante. Si la trace laissée n'était pas sous forme d'une ligne, nous pourrions observer des variations, mais la pièce de référence utilisée est un plan avec lequel nous ne générons que des lignes laser. Finalement, seule une deuxième rotation est requise autour de Z_{Ref}' .

Avec la configuration du plan de la figure 91, il n'est pas possible de rechercher l'angle ϕ de cette deuxième rotation autour de Z_{Ref}' . En effet, la direction de la trace laser n'est pas connue, de plus notre méthode ici consiste à ramener les points acquis dans le plan de référence ce que la rotation Q_{com} a déjà effectué. Nous choisissons donc une nouvelle configuration pour le plan étalon, présentée sur la figure 92, où l'angle ϕ peut être identifié en cherchant la rotation amenant le vecteur $Z_{Acquis2}$ dans ce nouveau plan de normale Z_{Ref2}' .

L'angle ϕ est donc calculé avec l'équation 244, les directions $Z_{Acquis2}$ et Z_{Ref2}' . La deuxième rotation Q_{com2} de la matrice complémentaire peut donc être déduite de cette rotation d'angle ϕ autour de Z_{Ref}' , et de l'équation 242. Puis l'équation 243 donne la matrice complémentaire totale T_{com} à partir de Q_{com} et Q_{com2} .

Nous mettons en place l'identification de cette matrice complémentaire et de ces deux rotations Q_{com} et Q_{com2} dans notre cellule de numérisation robotisée. Tout d'abord un plan étalon (dimensions 100x50x10 mm) est placé sur le support de pièce de la cellule, avec sa normale verticale ascendante pour l'identification de Q_{com} . La numérisation du plan étalon, avec notre capteur laser plan non incliné comme sur la figure 91, sans matrice complémentaire mais avec la matrice de calibration, donne le résultat présenté sur les figures 93 et 94. La numérisation du plan est réalisée en deux passes décalées de 15 mm, que l'on reconnaît sur la figure 93 représentant le nuage de points en vue de dessus.

Sans notre matrice complémentaire modifiant l'information sur l'orientation du Zephyr, on peut voir sur la figure 94 que les points acquis ne forment pas un plan. En appliquant la méthode présentée précédemment, on peut rechercher l'axe u et l'angle θ de la rotation ramenant les points 3D acquis dans le plan de référence. Le calcul de la rotation Q_{com} est ensuite effectué avant de l'intégrer au travers de la matrice complémentaire dans notre interface.

La numérisation du plan étalon en deux passes, toujours dans la même configuration, et avec la prise en compte de Q_{com} , est donnée sur les figures 95 et 96, où tous les points numérisés sont bien compris dans un même plan.

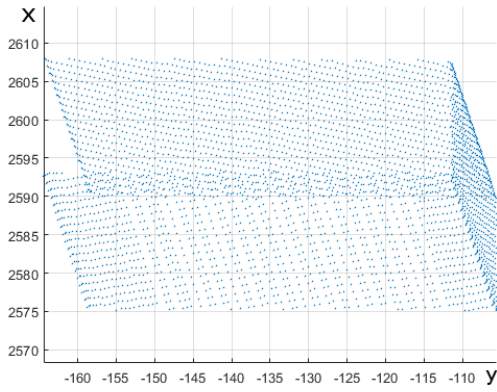


Figure 93 : Numérisation du plan étalon sans notre matrice complémentaire, vue XY en mm

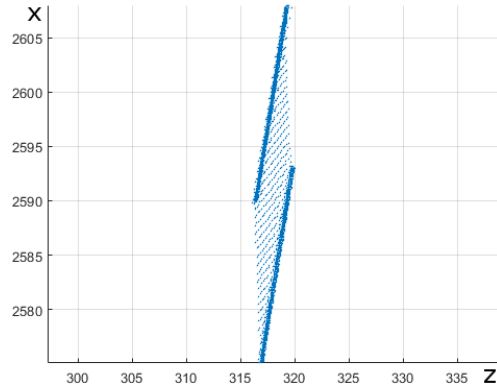


Figure 94 : Numérisation du plan étalon sans notre matrice complémentaire, vue ZY en mm

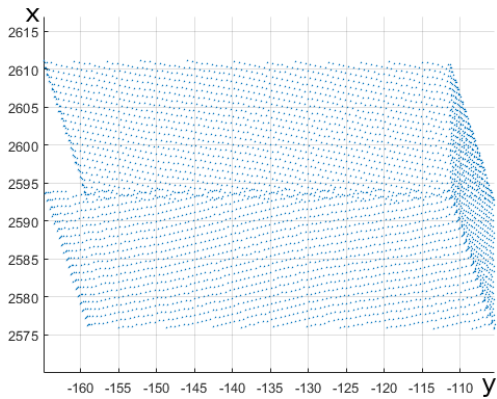


Figure 95 : Numérisation du plan étalon avec notre première rotation complémentaire Q_{com} , vue XY en mm

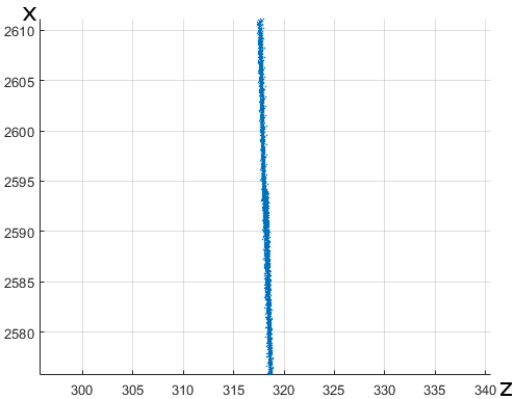


Figure 96 : Numérisation du plan étalon avec notre première rotation complémentaire Q_{com} , vue ZY en mm

La deuxième étape consiste à trouver la rotation Q_{com2} à partir d'une autre configuration du plan étalon. Ce plan étalon est donc incliné comme sur le schéma de la figure 92, puis en appliquant notre méthode, nous identifions Q_{com2} . La numérisation de ce plan incliné pour l'identification de Q_{com2} , avec prise en compte de la rotation Q_{com} , donne le résultat visible sur les figures 97 et 98. Puis la numérisation de ce même plan incliné, en prenant compte de Q_{com} et de Q_{com2} , est donnée sur les figures 99 et 100, où les points numérisés sont bien compris dans un même plan.

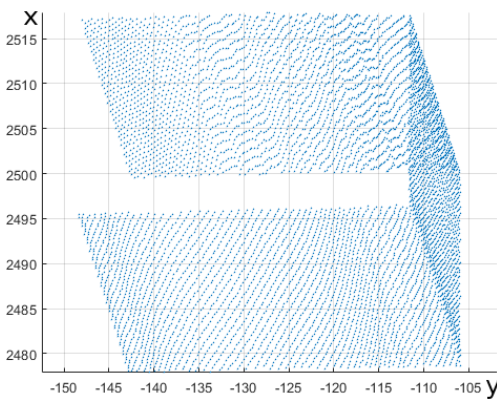


Figure 97 : Numérisation du plan étalon incliné à 45° avec Q_{com} , vue XY en mm

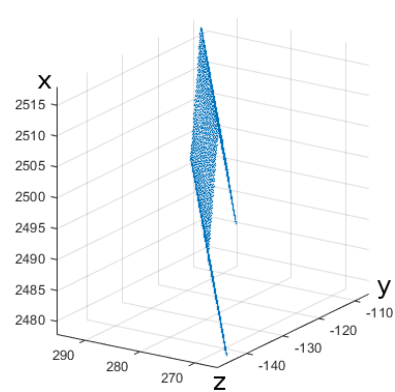


Figure 98 : Numérisation du plan étalon incliné à 45° avec Q_{com} , vue isométrique en mm

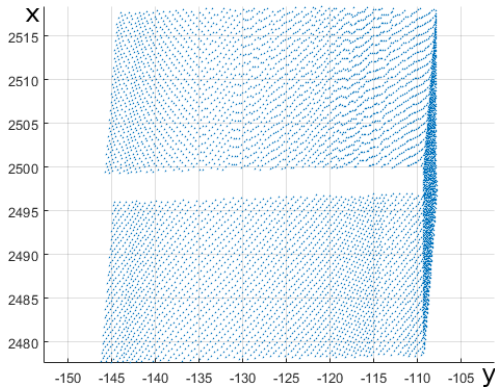


Figure 99 : Numérisation du plan étalon incliné à 45° avec T_{com} complet, vue XY en mm

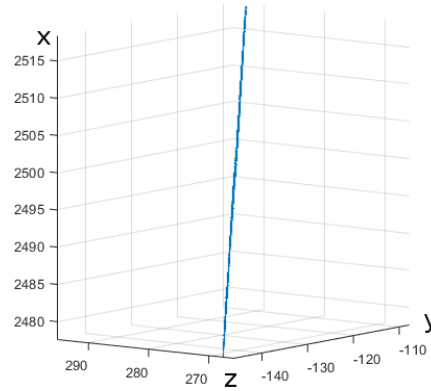


Figure 100 : Numérisation du plan étalon incliné à 45° avec T_{com} complet, vue isométrique en mm

Afin de tester la matrice complémentaire ainsi identifiée, comprenant les matrices de rotation Q_{com} et Q_{com2} , nous numérisons une pièce test. La pièce test utilisée est celle de la figure 89 déjà employée dans l'annexe 6 et représentant un visage. Le résultat final sur la numérisation en trois passes de cette pièce est visible sur les figures 101 et 102. Notre matrice complémentaire devant opérer correctement pour toute orientation du capteur, le visage est numérisé dans une configuration différente que celle utilisée lors de notre identification c'est-à-dire : visage incliné sur le support de pièce, et orientation Zephyr/visage également incliné.

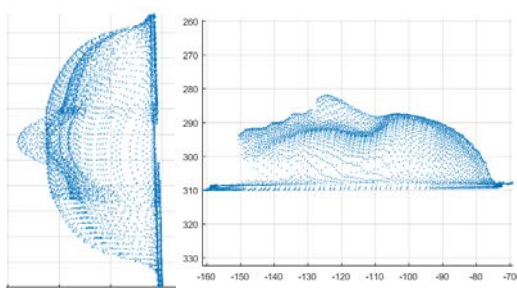


Figure 101 : Visage numérisé avec notre matrice complémentaire, vue de côté en mm

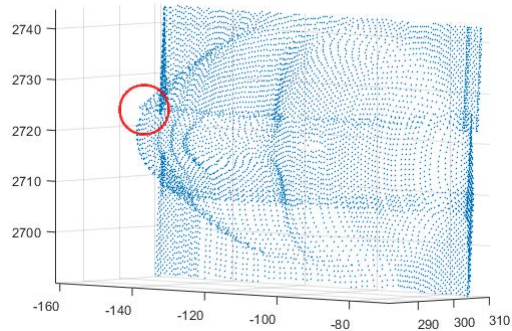


Figure 102 : Visage numérisé avec notre matrice complémentaire, vue isométrique en mm

Notre méthode d'identification et notre matrice complémentaire amènent à une orientation satisfaisante des ensembles de points 3D acquis, et donc à un nuage de points satisfaisant. Cette méthode est à la fois pratique et rapide à mettre en place, car elle ne nécessite qu'un plan étalon et le capteur reste sur l'effecteur du robot. Cependant pour une configuration tout autre que celle utilisée pour la calibration, on peut encore observer des décalages entre deux passes comme sur la figure 102 au niveau du menton. La troisième rotation non prise en compte autour de la ligne laser peut donc s'avérer importante et n'est pas négligeable pour des configurations très différentes de la configuration de calibration. Pour la numérisation de forme ayant peu de volume comme un plan, c'est-à-dire des formes pour lesquelles la trace laser reste à peu près une ligne, cette première méthode de calibration fonctionne très bien. Cependant lors de la numérisation d'une forme avec plus de volume comme un visage, des erreurs autour de la troisième rotation influent sur la numérisation dans la profondeur de mesure du plan laser.

Dans cette première méthode, la seule information disponible est la forme de la pièce de référence (plan étalon), rendant impossible l'estimation de la troisième rotation plus complexe. De plus, notre deuxième objectif est d'obtenir le positionnement du plan laser et de ses limites dans le repère des cibles, afin de compléter la modélisation de la cellule. Ce positionnement est impossible à obtenir sans avoir plus d'informations sur la pièce de référence numérisée et sur la configuration du capteur lors de la

calibration. Pour ces raisons une deuxième méthode est proposée, plus lourde à mettre en place, mais plus complète.

Cette première méthode de calibration est tout de même parfaitement valable pour des numérisations simples où l'orientation du capteur n'est pas trop éloignée de l'orientation de calibration, et où la pièce à numériser dispose de formes simples avec peu de profondeur.

5.2.4 2^{ème} méthode : calibration sur MMT

Cette deuxième méthode a pour objectif de se baser sur des références plus complètes, afin d'identifier une matrice complémentaire comprenant la rotation négligée avec la méthode précédente. De plus cette calibration n'est pas effectuée sur le robot mais sur une Machine de Mesure Tridimensionnelle (MMT), afin d'obtenir une matrice complémentaire plus juste et une meilleure réorientation des ensembles de points 3D acquis. Ainsi la calibration est effectuée sur une MMT, pour une utilisation du capteur monté ensuite sur le robot. Le second objectif est d'avoir également le positionnement du plan laser et de ses limites dans le repère des cibles.

Le capteur laser doté de cibles est donc mis en place sur une MMT, qui est un système porteur conçu pour la mesure. Cette deuxième méthode est donc plus complexe à mettre en place et nécessite plus de matériel. La MMT peut déplacer le capteur suivant 3 directions X , Y et Z , et les règles de mesure sur chacun de ces axes permettent au contrôleur du Zephyr une acquisition simple d'un nuage de point après une calibration. En effet le logiciel Polygonia est conçu pour fonctionner avec ce type de système porteur, et le résultat de numérisation en passant par une MMT est donc optimal. Il faut pour cela réaliser une calibration proposée par Polygonia à l'aide d'une sphère étalon. Dans cette deuxième méthode de calibration, deux modes de fonctionnement du capteur sont utilisés pour le capteur monté sur la MMT. Soit l'information sur la position du Zephyr est fournie par la MMT, et Polygonia utilise la calibration sphère mentionnée pour donner un nuage de points de très bonne qualité. Soit l'information sur la pose (position et orientation) du Zephyr est fournie par le C-Track, et Polygonia utilise la matrice de calibration T_{calib} . La numérisation effectuée avec les coordonnées de la MMT sert de référence pour identifier notre matrice complémentaire.

Tout d'abord la calibration classique du capteur avec la MMT est réalisée à l'aide d'une sphère étalon. Puis un plan étalon est numérisé en utilisant les coordonnées de la MMT et le logiciel Polygonia. On obtient alors un nuage de points correspondant à ce plan dans le repère de la MMT, ce nuage servira de référence. Or dans notre méthodologie, la référence doit être exprimée dans le repère externe qui est le repère C-Track. Nous cherchons donc à estimer l'orientation du repère MMT dans le repère CTrack (car seule l'orientation nous intéresse). Pour cela nous suivons le déplacement des cibles (sur le Zephyr) en sollicitant un à un les axes de la MMT. Avec ces mesures il est ensuite possible de déduire l'orientation des axes X Y et Z de la MMT dans le repère CTrack. Cela nous permet d'exprimer ce premier nuage de points de référence dans le repère C-Track.

Contrairement à la première méthode, on peut tirer de ce nuage non pas la normale d'un plan, mais la direction moyenne de la ligne laser au cours de la numérisation Z_{Ref} . En effet ici nous avons plus qu'une simple information sur la forme que devrait avoir le nuage, mais un nuage de référence. Chaque ensemble de points étant acquis séparément, nous calculons la direction correspondante à chaque ligne laser acquise, puis la direction moyenne de référence Z_{Ref} . Cependant, pour que cette direction de référence reste valide, l'orientation du capteur doit rester identique durant toute la calibration.

La numérisation de ce même plan en utilisant la pose fournie par le C-Track nous permet d'obtenir Z_{Acquis} . On peut alors appliquer la méthodologie présentée dans la section 5.2.2, et calculer la première rotation Q_{com} de la matrice complémentaire. Contrairement à la première méthode de calibration, la rotation Q_{com} ici correspond bien au passage d'un vecteur vers un vecteur et non vers un plan, ce qui ne laisse que la rotation autour de la ligne laser de libre. La rotation $Q_{\text{com}2}$ prend en compte cette dernière

rotation laissée libre. La correction de l'information sur l'orientation du Zephyr avec la matrice complémentaire est donc complètement définie (pas de rotation négligée).

Pour pouvoir identifier Q_{com2} il faut réussir à mettre en évidence les défauts causés par la rotation autour de la ligne laser (autour de Z_{Ref}). Or, cette rotation ne modifie pas la trace laser qui est une ligne. Une rotation autour de Z_{Ref} signifie que le plan laser change d'orientation, donc il faut pouvoir mesurer l'orientation des plans lasers servant de référence et acquis. Pour cela il faut donner à la trace laser une forme non rectiligne, couvrant autant que possible le plan laser du Zephyr. La numérisation d'un cylindre incliné permet de maximiser cette couverture dans la fenêtre laser (plan laser et ses limites) comme sur la figure 103. Nous numérisons donc un cylindre incliné de 10° , cette numérisation donne le résultat visible sur la figure 104.

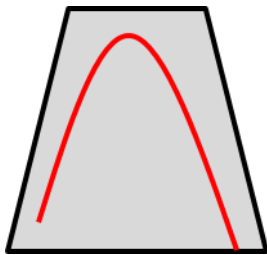


Figure 103 : Remplissage de la fenêtre par la ligne laser

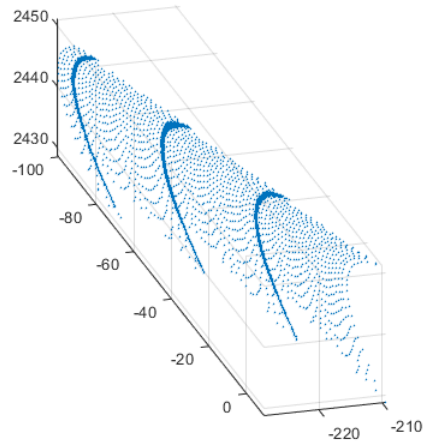


Figure 104 : Numérisation du cylindre incliné, mm

Deux numérisations sont effectuées pour ce cylindre : une numérisation de référence utilisant les coordonnées de la MMT, et une autre avec défauts utilisant le C-Track (avec notre matrice complémentaire incomplète). La mesure de référence nous donne des ensembles de points suivant une ligne courbée. Pour chacune de ces courbes nous calculons le plan correspondant. Une moyenne de ces plans nous donne la direction de référence de la normale du plan laser Z_{Ref2} . Puis il faut faire de même avec l'autre numérisation : acquisition des ensembles de points, calcul des plans, calcul de la normale moyenne des plans laser acquis $Z_{Acquis2}$. On peut alors suivre la méthodologie de la section 5.2.2, en cherchant l'angle θ_2 de la rotation autour de Z_{Ref} , permettant le passage entre $Z_{Acquis2}$ et Z_{Ref2} . Les deux directions $Z_{Acquis2}$ et Z_{Ref2} doivent être orthogonale à Z_{Ref} pour pouvoir estimer θ_2 , les normales de référence et acquise identifiées sont donc projetées pour correspondre à cette contrainte. On obtient donc au final la deuxième matrice de rotation Q_{com2} .

Cette deuxième méthode de calibration est mise en place avec le C-Track et la MMT disponible au sein du laboratoire. La matrice Q_{com} est identifiée puis appliquée dans notre interface au travers de la matrice complémentaire afin d'identifier la matrice Q_{com2} . Ces deux matrices de rotations sont finalement incluses dans la matrice complémentaire, ce qui clôt cette méthode de calibration.

Le capteur Zephyr est alors monté sur le robot, pour effectuer une numérisation de validation avec la pièce test de la figure 89 représentant un visage. Le résultat obtenu est présenté sur les figures 105 et 106, le capteur étant dans une configuration différente de celle utilisée pour la calibration afin de s'assurer de la validité de notre identification pour d'autres configurations.

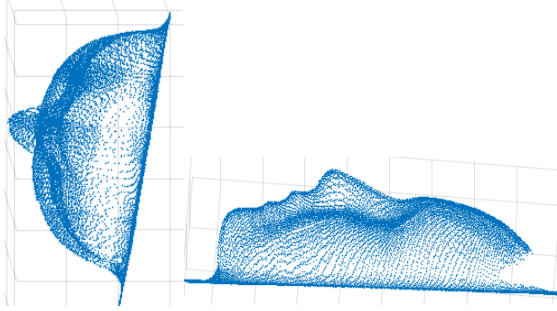


Figure 105 : Visage numérisé avec 2^{ème} méthode de calibration, vue de côté en mm

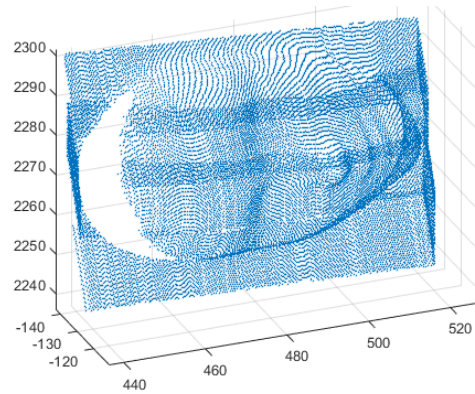


Figure 106 : Visage numérisé avec 2^{ème} méthode de calibration, vue isométrique en mm

Cette deuxième méthode de calibration est plus fastidieuse à mettre en place, nécessitant plus de temps et de matériel. Mais les résultats après calibration sont meilleurs, le visage numérisé ne présente cette fois-ci aucune imperfection visible. Le protocole est plus rigoureux, utilisant une référence plus fiable et identification complète de la matrice complémentaire. De plus cette méthode nous permet également de répondre à notre deuxième objectif qui est d'obtenir le positionnement du plan laser et de ses limites (fenêtre laser) dans le repère des cibles. Il faut cependant des essais supplémentaires, car la méthode en l'état ne nous permet que de connaître le positionnement du plan laser dans le repère des cibles.

En utilisant le plan étalon comme pièce à numériser, nous positionnons le capteur laser à sa limite haute de visibilité de la pièce. Nous faisons une mesure sans déplacer le capteur pour acquérir la ligne laser correspondante dans le repère du C-Track, ainsi que la pose du repère de cibles également dans ce repère. Il est alors possible d'en tirer le segment faisant la borne inférieure de la fenêtre laser dans le repère des cibles. De même nous plaçons le capteur à sa limite basse de visibilité de la pièce pour faire les mêmes mesures et en déduire le segment faisant la borne supérieure de la fenêtre laser. Les segments étant à la bonne longueur, on en déduit le trapèze de la fenêtre de mesure du capteur Zephyr délimitant la visibilité dans le plan laser, et qui est exprimé dans le repère des cibles.

Même si cette deuxième méthode semble donner de meilleurs résultats de numérisation, nous mettons en place un protocole de qualification déterminant la qualité de la numérisation réalisée avec les différentes méthodes présentées. Cette étude est réalisée dans la section suivante.

5.3 Evaluation qualitative des méthodes de calibration présentées

Afin de qualifier la numérisation réalisée par notre cellule de mesure robotisée, nous mettons en place une série d'essais focalisés sur la justesse et le bruit dans le nuage de point résultant. Cette qualification permet d'évaluer l'ensemble {système porteur ; capteur laser}, et prend en compte également l'impact du système de suivi externe et de notre calibration du Zephyr. La qualification est réalisée en appliquant des méthodologies présentes dans le protocole QualiPSO qui a été développé dans le but de qualifier des systèmes de numérisation [Mehdi-Souzani et al. 2016]. Nous n'appliquons ce protocole qu'en partie, car le capteur Zephyr a déjà été qualifié sur un autre système porteur (MMT). Or certains résultats ne changent pas d'un porteur à un autre et sont alors réutilisables. Nous nous focalisons donc sur une étude du bruit de numérisation sur un plan étalon numérisé, ainsi que sur la justesse dans la numérisation d'un artéfact calibré. Ces notions de bruit et de justesse ont déjà été définies dans la section 1.5.1.

Tout d'abord nous réalisons plusieurs numérisations d'un plan étalon pour estimer le bruit. Des essais sont mis en place pour chacune de nos deux méthodes de calibration du capteur laser que nous avons proposées. Nous employons également des vitesses différentes et plusieurs configurations du capteur laser. Utiliser différentes vitesses permet d'observer l'impact de la vitesse de numérisation sur la qualité

du nuage de points. En effet diminuer le temps de l'opération de numérisation diminue également la densité du nuage et impacte la qualité de ce dernier. Les deux vitesses choisies pour le déplacement de l'effecteur du robot sont de 15 mm/sec et de 40 mm/sec. D'autre part nous sollicitons plusieurs configurations du capteur afin de tester l'efficacité de notre calibration en orientation. Les quatre configurations différentes utilisées sont présentées sur la figure 107. Sur cette figure la flèche rouge indique l'orientation du capteur laser, dans le repère du C-Track. Pour la première configuration les cibles disposées sur le capteur laser font face au C-Track, dans les autres configurations elles sont donc inclinées de 45° suivant l'axe correspondant, comme le plan. Le but est d'identifier l'influence de l'inclinaison des cibles par rapport aux caméras du C-Track. Or l'inclinaison du Zephyr par rapport à la pièce à numériser influence également sur la qualité du nuage de point. C'est pourquoi, pour minimiser l'effet de l'inclinaison pièce/Zephyr, mais tout de même incliner les cibles dans le repère du C-Track, nous fixons l'orientation pièce/Zephyr, et inclinons cet ensemble comme sur la figure 107. Les numérisations sont réalisées en plusieurs passes comme dans nos méthodes de calibration, afin de rendre influents les défauts restants sur l'orientation des ensembles de points acquis.

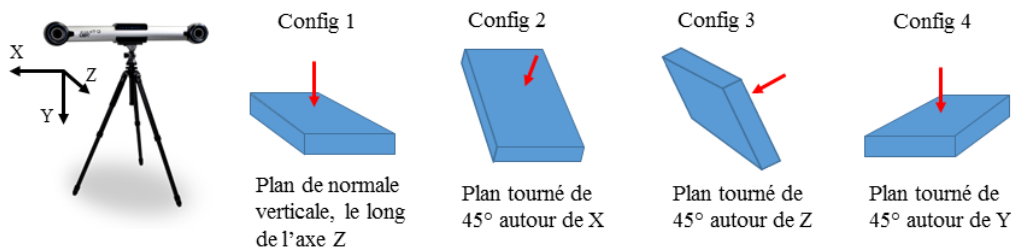


Figure 107 : Orientations du plan étalon (et du Zephyr) pour numérisation

Après avoir numérisé le plan en jouant sur chaque variable, nous estimons le bruit de mesure sur les nuages de points résultants. Pour ce calcul nous utilisons la formule déjà introduite dans l'équation 39 de la section 1.5.1. La figure 108 présente les résultats obtenus sur le bruit, en fonction de la configuration du capteur et du plan, et pour chaque couple {méthode de calibration ; vitesse}. La méthode 1 correspond à la méthode de calibration effectuée avec le capteur sur le robot, tandis que la méthode 2 correspond à celle où le capteur est sur une MMT. Les vitesses 1 et 2 correspondent respectivement aux vitesses de 15 mm/sec et de 40 mm/sec.

L'étude du bruit permet de constater qu'une plus grande vitesse tend à dégrader la qualité du nuage de points. Par contre il ne ressort pas de meilleure méthode, même si l'association de la première méthode avec la deuxième vitesse donne les plus mauvais résultats. On peut observer une influence importante des configurations sur le bruit, mais aucune tendance ne se dégage des résultats. Il faut donc tester les configurations choisies pour la numérisation d'une pièce si l'on souhaite minimiser le bruit. À titre de comparaison, le bruit de numérisation maximal obtenu lorsque le capteur Zephyr est porté par la MMT du laboratoire et non par le robot est de l'ordre de 0,035 mm [Mehdi-Souzani et al. 2016]. Par ailleurs, lorsqu'un capteur laser plan est porté par un robot, le bruit maximal atteint est également de cet ordre de grandeur (0,01mm à 0,03mm) [Phan et al. 2018, Sharifzadeh et al. 2018]. Mais cette valeur est obtenue pour la numérisation d'un plan en une passe, tandis que la figure 108 est obtenue pour la numérisation d'un plan en 4 passes afin de plus solliciter les défauts d'orientation des ensembles de points évoqués dans les sections précédentes. En effet on retrouve dans [Phan et al. 2018], que le bruit maximal sur une numérisation en plusieurs passes monte à 0,07 mm au lieu de 0,02 mm pour une seule passe. Si nous réalisons une numérisation en une seule passe avec notre cellule robotisée, le bruit ne dépasse pas 0,043 mm quelle que soit la méthode, la vitesse ou la configuration, ce qui est très proche de la valeur obtenue avec une MMT.

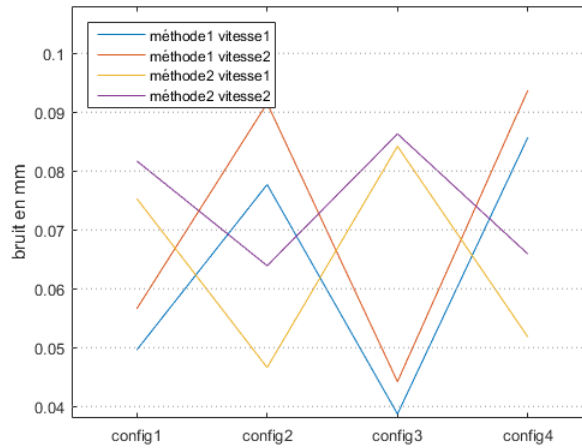


Figure 108 : Bruit de numérisation, fonction de la configuration, de la vitesse et de la méthode

L'autre caractéristique recherchée pour qualifier notre système de numérisation est la justesse. La numérisation d'un simple plan ne permet pas de qualifier cette caractéristique, il faut un artéfact disposant de distances calibrées. C'est pourquoi nous employons l'artéfact présenté dans les figures 109 et 110, représentant des marches calibrées. Sur cet objet la différence de hauteur entre chaque marche est étalonnée, et les marches sont suffisamment fines pour pouvoir en numériser 2 à la fois. Il sera donc possible de comparer les valeurs de référence et des valeurs calculées après numérisation. L'étroitesse de l'artéfact et des marches, impose d'effectuer les numérisations en une seule passe contrairement à l'étude sur le bruit.

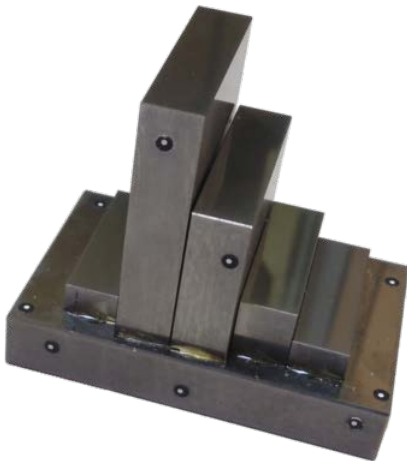


Figure 109 : Marches calibrées : artéfact pour l'estimation de la justesse

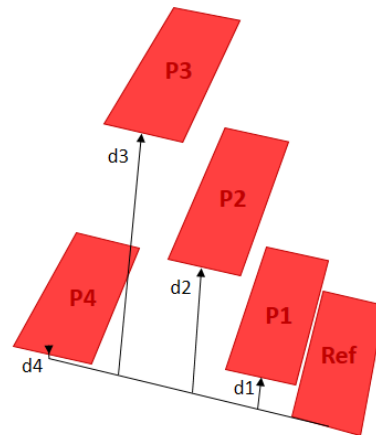


Figure 110 : Marches 1 à 4 et distances au plan de référence

Sur la figure 110, les marches 1 à 4 se situent respectivement à une hauteur de : 5 mm, 20 mm, 35 mm et 0 mm, par rapport à la première marche de référence. Ces marches sont alors numérisées pour les 4 configurations, les 2 méthodes et les 2 vitesses déjà utilisées dans l'étude sur le bruit. Tout d'abord la figure 111 présente la justesse de la hauteur de chacune de ces marches pour les différentes méthodes et vitesses. Puis la figure 112 présente également la justesse sur les marches, mais pour les 4 configurations du capteur. Les calculs de justesse sont réalisés en utilisant l'équation 40 déjà présentée dans la partie 1.5.1.

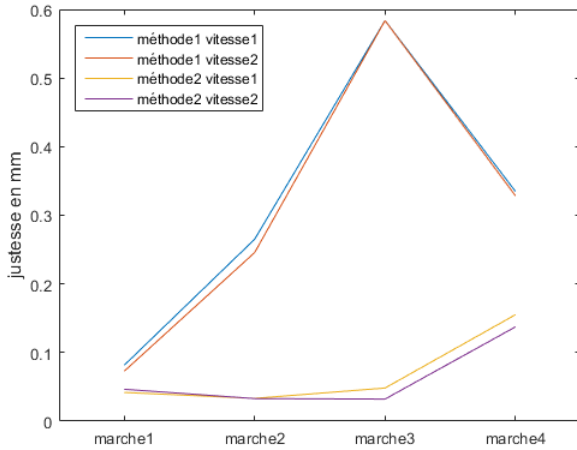


Figure 111 : Justesse sur la hauteur des marches de l'artéfact, pour les différentes méthodes et vitesses

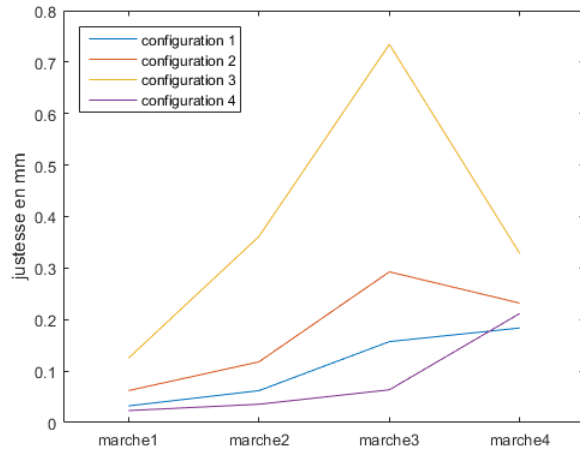


Figure 112 : Justesse sur la hauteur des marches de l'artéfact, pour les différentes configurations

On remarque cette fois-ci, à partir de la figure 111, que la vitesse n'a que très peu d'influence sur la justesse. Par contre on différencie très bien les méthodes 1 et 2. La deuxième méthode de calibration effectuée avec la MMT présente une justesse plus de deux fois meilleure que celle atteinte avec la première méthode. On peut donc conclure sur le faible impact de la différence de vitesse que nous avons fixée, et sur l'avantage en qualité apporté par la deuxième méthode de calibration. La figure 112 permet de montrer l'impact important de la configuration sur la justesse, surtout pour la configuration 3. L'inclinaison du capteur autour de l'axe Z du C-Track sera donc à limiter afin de ne pas solliciter cette source d'erreur. À titre de comparaison, la valeur de justesse maximale atteinte pour la numérisation d'un plan étalon avec le Zephyr porté par une MMT est de l'ordre de 0,02 mm [Mehdi-Souzani et al. 2016]. Les résultats présentés sur ces figures sont en moyenne au-dessus, mais utiliser la deuxième méthode de calibration et faire attention aux configurations employées (orientation des cibles), permet d'approcher cet ordre de grandeur et de travailler au centième sur la justesse.

5.4 Synthèse du chapitre

Dans ce chapitre, la mise en place de l'opération de numérisation dans notre cellule robotisée a été présentée. Dans ce contexte il a fallu mettre en relation le capteur laser avec le système de suivi externe. En effet le capteur a besoin de connaître sa position et son orientation dans un repère externe afin de recalibrer les différentes lignes laser acquises. Le C-Track envoie cette pose au capteur au travers d'un programme interface que nous avons développé. Cette même interface permettra plus tard la transmission de la pose non pas par le C-Track, mais par le modèle du robot.

Nous avons ensuite proposé plusieurs méthodes de calibration du capteur laser avec les cibles qui lui sont associées. Ces méthodes pratiques et simples, permettent un recalage de l'orientation du plan laser dans le repère des cibles qui sont suivies par le C-Track. Chaque méthode est générique et présente des avantages et des inconvénients. Afin de les comparer, nous avons conduit une campagne de qualification de la numérisation réalisée avec chaque méthode, mais aussi pour des vitesses différentes, et des configurations différentes. Il ressort que notre deuxième méthode de calibration apporte une meilleure justesse sur la mesure. De plus, ces résultats sur la justesse et le bruit ne sont pas très éloignés de ceux obtenus dans des travaux antérieurs sur la qualification du Zephyr porté par une MMT, qui nous servent de référence.

L'opération de numérisation robotisée est donc fonctionnelle. De plus amples essais devront être réalisés sur des pièces plus larges afin de qualifier cette numérisation sur des pièces encombrantes. Par ailleurs la même qualification sera effectuée lorsque la pose de l'effecteur pourra être également donnée par notre modèle du robot à partir d'une acquisition des coordonnées articulaires de celui-ci. En effet les

méthodes de calibrations présentées peuvent être utilisées également dans le cas où le robot s'occupe du suivi de l'effecteur lors du recalage des points. Il sera donc possible de numériser sans système de suivi externe, avec une qualité de numérisation différente de celle identifiée dans ce chapitre, car dépendante de la qualité du modèle proposé au chapitre 3.

Chapitre 6

Conclusion et perspectives

Dans un contexte industriel ou de laboratoire, l'état d'une pièce doit être maîtrisé tout au long de la production afin d'identifier au plus tôt d'éventuels défauts sur la pièce. Détecter un défaut tôt dans le procédé de fabrication permet d'une part d'appliquer des étapes de rectification avant de poursuivre le procédé, et d'autre part d'identifier plus facilement la source du défaut pour corriger le procédé (un réglage machine, une mauvaise trajectoire, une usure d'outil, etc). Dans ce cadre, l'utilisation d'un robot sériel associé à un instrument de mesure de type scanner apporte de nombreux avantages comme un temps court pour l'opération de mesure ou de la flexibilité (adaptation à différents procédés et pièces).

Ainsi les travaux de ce mémoire se sont attachés à l'élaboration d'une cellule de numérisation robotisée, qui doit permettre en priorité la numérisation de pièces complexes ou encombrantes pour une qualité de mesure maîtrisée et prédéfinie dans un contexte de contrôle qualité. Cellule pour laquelle nous avons également pris en compte l'optimisation de la vitesse d'exécution de l'opération tout en assurant une mise en place pratique (simple et peu intrusive) du dispositif de numérisation. Afin de répondre à ces besoins, une cellule de numérisation robotisée a été mise en place, constituée d'un robot anthropomorphe 6 axes, d'un capteur laser plan et d'un système de suivi externe. Le développement de la cellule a été guidé par différentes étapes clés, qui ont été traitées dans les chapitres de ce mémoire.

Tout d'abord dans un premier chapitre, l'analyse de l'état de l'art et du contexte autour des cellules robotisées, nous a permis de cerner la problématique majeure liée à la mise en place d'une telle cellule : la qualité de positionnement du robot. Afin de maîtriser cette qualité et d'autres aspects au cours de l'opération de numérisation, nous avons proposé une stratégie de génération de trajectoire basée sur la connaissance des performances du robot et des instruments de mesure utilisés. Cette stratégie vise à exploiter au mieux ces performances afin d'obtenir une trajectoire durant laquelle la qualité de numérisation sera assurée et la vitesse de l'opération optimisée.

Dans ce contexte, le développement de cette stratégie requiert de connaître les performances du capteur laser plan et du système de suivi, qui ont déjà été évaluées au préalable et qui sont rappelées dans le premier chapitre. Il est également nécessaire de connaître les performances du robot, qui sont étudiées dans le chapitre 4. Nous avons vu aussi qu'un modèle du robot et de son outil sont nécessaires pour pouvoir générer une trajectoire adaptée au comportement de ce robot et de cet outil. Dans le premier chapitre, un état de l'art des différentes méthodes de modélisation connues a été présenté. Mais afin de répondre aux exigences liées à notre contexte, nous avons proposé de nouvelles idées dans les chapitres 2, 3 et 5, tant sur les méthodes d'identification que sur les modèles du robot et de l'outil porté (capteur laser plan).

À partir de l'état de l'art exposé dans le premier chapitre, nous avons élaboré dans le second chapitre un modèle géométrique basé sur la convention DH modifiée, correspondant à la structure de notre robot. Il a fallu alors créer une méthode originale d'identification des paramètres de ce modèle, afin de respecter notre besoin d'une mise en place simple et peu intrusive du dispositif de numérisation, relativement aux contraintes d'un contexte industriel. La méthode d'identification que nous proposons, et que nous avons baptisée méthode ANCAR, se base sur l'étude d'arcs de cercle générés par l'effecteur du robot, permettant de trouver le positionnement des axes du robot, puis les paramètres géométriques. De plus nous avons également imaginé une méthode de résolution plus robuste, inspirée de méthodes classiques utilisant une jacobienne, mais qui peut exploiter des mesures plus riches que de simples positionnements de l'effecteur, comme par exemple la localisation des axes du robot. L'association de la méthode

ANCAR et de notre résolution par jacobienne donne une identification des paramètres géométriques très pratique et très adaptable, parfaitement en accord avec les besoins fixés.

Une série d'essais a permis de vérifier que les paramètres obtenus à partir de nos méthodes d'identification et de résolution, améliorent effectivement le positionnement absolu du robot (en comparaison avec le modèle déjà présent dans l'armoire du robot ou un modèle identifié par une méthode classique). Cependant nous avons constaté que malgré cette amélioration, l'erreur de positionnement était toujours de l'ordre du dixième de millimètre, ce qui peut nuire à la qualité de numérisation. Nous avons donc étudié au début du troisième chapitre, l'impact d'une telle erreur sur différents aspects de la numérisation robotisée. Cette étude conclut sur le besoin d'une modélisation plus détaillée car cette erreur est trop influente sur l'expression des points numérisés dans un repère de référence commun, et donc sur la qualité du nuage de points final. Avec cette étude nous avons également pu conclure que le simple modèle géométrique est suffisant pour l'étude des performances du robot ou pour la génération de la trajectoire dans le domaine articulaire (à partir du domaine cartésien).

Pour améliorer la qualité du nuage de points, nous avons donc cherché à détailler le modèle géométrique avec des paramètres non géométriques. Le chapitre 3 poursuit alors avec la définition de paramètres d'élasticité pour les segments du robot, puis de paramètres d'élasticité et jeu pour les articulations en s'inspirant de la définition normée d'une articulation [ISO.230-1 2012]. Avant d'intégrer ces paramètres au modèle géométrique existant, il a fallu les étudier pour ne conserver que les paramètres utiles. Nous avons donc proposé une méthode de sélection de paramètres non géométriques, basée en premier lieu sur un raisonnement qualitatif d'élimination des paramètres non influents sur notre structure de robot, puis sur une étude d'identifiabilité des paramètres restant. Grâce à notre méthode de sélection, nous avons réduit le nombre de paramètres non géométriques à prendre en compte de 84 à 34 paramètres. L'intégration de ces paramètres dans le modèle existant résulte en une modélisation élasto-géométrique du robot, dont les paramètres non géométriques sont encore à identifier.

Afin de connaître les capacités du robot, nous avons étudié dans le chapitre 4 différents indices de performance de robotique. Après une présentation des indices les plus utilisés dans le domaine, nous avons conclu qu'ils ne convenaient pas aux besoins de notre stratégie de génération de trajectoire. En effet notre stratégie requiert de connaître les capacités du robot sur toute une trajectoire et non pour une seule configuration donnée. C'est pourquoi nous avons introduit de nouveaux indices de performances globaux sur toute une trajectoire afin d'obtenir :

- la meilleure vitesse pouvant être respectée sur tous les points de la trajectoire : l'indice de vitesse minimale.
- la vitesse moyenne le long de la trajectoire : l'indice de vitesse moyenne.
- la pire répétabilité de pose croisée au cours de la trajectoire : l'indice de répétabilité maximale.

Nous avons également ajouté à ces indices décrivant les capacités en vitesse et en qualité de pose, un indice décrivant l'accessibilité du robot. Ces indices seront intégrés dans le générateur de trajectoire qui est encore à développer, et ils contribueront à la sélection et à l'optimisation de trajectoire de numérisation.

Enfin dans le dernier chapitre de ce mémoire, un dernier point sensible dans la mise en place de la cellule robotisée a été traité : la calibration du capteur laser plan. Le grand intérêt d'une telle cellule robotisée est de pouvoir réorienter librement le capteur laser. Mais cette liberté rend la calibration de ce capteur dans la cellule plus complexe. En effet, en temps normal l'usage de ce capteur (par exemple sur MMT) impose de définir une calibration pour chaque orientation utilisée. Lors de l'installation du capteur dans la cellule, nous avons établi deux protocoles de calibration permettant cette réorientation libre sans dégradation du nuage de points résultant de la numérisation. Ces deux protocoles de calibration sont

génériques et permettent une calibration simple et pratique de tout capteur laser plan. Après application d'un de ces protocoles, nous obtenons une cellule de numérisation robotisée fonctionnelle.

Dans ce même chapitre, nous avons mis en place des essais testant la justesse et le bruit de nuages de points obtenus après application de l'un ou l'autre protocole. Il ressort naturellement que le protocole le plus complet donne de meilleurs résultats. De plus ces résultats sont proches de résultats obtenus par des numérisations réalisées sur MMT qui servent de référence, ce qui atteste de la qualité de notre cellule.

L'ensemble des travaux menés dans le cadre de cette thèse contribuent à la mise en place rigoureuse d'une numérisation robotisée pouvant être réalisée dans un contexte industriel. Nous sommes efforcés d'établir des méthodes pratiques adaptées à ce contexte, tout en cherchant à maîtriser au mieux l'opération de numérisation en termes de vitesse et de qualité. De plus tout au cours des différents développements, nous avons validé nos démarches par des essais expérimentaux, et avons tâché de rester cohérents sur les niveaux de détails choisis en fonction des ordres de grandeurs identifiés.

D'un point de vue plus général, nous cherchons à travers ces travaux, à contribuer à rendre compatible l'usage d'un robot pour des opérations de mesure sans contact. L'association d'un robot (comme système de déplacement) et d'un système de mesure sans contact rend les opérations de numérisation plus flexibles, plus rapides et permet ainsi d'en effectuer un plus grand nombre au cours de la fabrication d'une pièce.

Les principales contributions proposées dans ce mémoire sont :

- Une stratégie de génération de trajectoire visant à exploiter au mieux les performances du robot et des instruments de mesure.
- La proposition d'une méthode d'identification (ANCAR), pour une identification rapide et pratique des paramètres du modèle géométrique.
- Une méthode de sélection de paramètres pour un modèle élasto-géométrique, basée sur la notion d'identifiabilité.

Deux méthodes de calibration pour le capteur laser plan lors de sa mise en place dans la cellule de mesure robotisée.

Perspectives

Deux articles ont déjà été publiés sur notre stratégie de génération de trajectoire [Bordron et al. 2018a, Bordron et al. 2018b], et un autre article plus focalisé sur notre méthode ANCAR est en cours d'écriture.

Les principales perspectives se trouvent dans la continuité directe des travaux de cette thèse. Il reste notamment à développer le générateur de trajectoire à partir des différents éléments établis dans ce mémoire. Il manque de plus, la connaissance du positionnement de la pièce à numériser dans le repère de la cellule, ce qui requiert également des modèles et méthodes d'identification. Par ailleurs, nous souhaitons avoir le choix entre une numérisation avec ou sans système de suivi externe, afin de pouvoir se libérer des contraintes apportés par un tel suivi. Il reste donc également à réaliser la connexion entre le capteur laser plan et le robot pour pouvoir suivre le capteur à partir du modèle et des angles du robot. Mais comme nous l'avons démontré dans une étude sur l'influence des erreurs du modèle géométrique, cette étape requiert une modélisation plus complète élasto-géométrique du robot, qui est établie dans le chapitre 3. Il reste donc de plus à identifier les paramètres non géométriques de cette modélisation.

Outre les perspectives propres à la mise en place de la cellule de numérisation, nous avons évoqué des évolutions possibles sur de nombreux points au cours de ce mémoire. Tout d'abord, l'introduction d'une résolution utilisant une jacobienne avec notre méthode ANCAR ouvre de nombreuses possibilités en termes d'identification des paramètres et pas seulement géométriques. De multiples combinaisons d'informations en entrée de notre résolution sont envisageables, exploitant la mesure de la localisation

des axes de rotation, du positionnement de l'effecteur, et potentiellement d'autres entités géométriques (mouvement à contrainte plane, ou sphérique, ou droite, etc). Toutes ces entités sont formalisées simplement avec des points et des directions dans la résolution. Ainsi, l'identification des paramètres ne se reposera plus seulement sur un ensemble de points bien choisis, mais sur des informations plus variées, amenant probablement à des résultats plus robustes. Par ailleurs, nous pensons qu'il est possible d'adapter notre méthode ANCAR et la résolution par jacobienne afin d'identifier les paramètres non géométriques de notre modèle complet.

On peut noter également en perspective, la possibilité de mieux modéliser la répétabilité du robot comme souligné dans le chapitre 4. En effet la répétabilité sur chaque articulation n'est pas une constante et dépend de paramètres comme le torseur des efforts sur l'articulation ou la dynamique du robot à l'approche du point d'arrêt (vitesse, décélération). Nous envisageons donc sur ce point, un plan d'expérience permettant de cibler les paramètres les plus impactants pour modéliser les répétabilités articulaires.

Enfin, de plus amples essais devront être réalisés pour qualifier notre cellule de mesure sur la numérisation de pièces encombrantes ou complexes. Dans le même temps ces essais devront permettre de valider l'apport de notre générateur de trajectoires sur les nuages acquis en termes de qualité et de rapidité.

Après avoir qualifié cette cellule de mesure, cette dernière pourra être utilisée dans le contexte du laboratoire pour numériser des pièces complexes réalisées en fabrication additive de métal. À partir de cette numérisation il sera possible de déduire des procédés de rectification de ces pièces sur centre d'usinage 5 axes, afin d'exploiter à la fois la flexibilité et la liberté de l'impression 3D, et la qualité de surface d'un usinage traditionnel. D'autre part, cette cellule de numérisation permettra également de mesurer des pièces d'aéronautique type panneau d'avion. Les panneaux manipulables par le robot, pourront être fixés sur celui-ci (configuration capteur fixe et pièce sur effecteur), ce qui permettra de les numériser avec différentes orientations pour observer l'effet de la gravité et en déduire la géométrie à l'état libre de ces pièces. Enfin il sera possible avec cette cellule de numériser les panneaux plus larges, posés sur des supports fixes, afin d'obtenir des nuages de points importants en une seule numérisation.

Bibliographie

- Abele, E., Bauer, J., Rothenbücher, S., Stelzer, M. et Von Stryk, O. (2008), Prediction of the tool displacement by coupled models of the compliant industrial robot and the milling process, in 'Proceedings of the International Conference on Process Machine Interactions, September 2008', Citeseer, pp. 223–230.
- Abele, E., Weigold, M. et Rothenbücher, S. (2007), 'Modeling and identification of an industrial robot for machining applications', *CIRP Annals-Manufacturing Technology* 56(1), 387–390.
- Alici, G. et Shirinzadeh, B. (2005), 'Enhanced stiffness modeling, identification and characterization for robot manipulators', *IEEE Transactions on Robotics* .
- ASME.B89.4.22-2004 (2014), 'Methods for performance evaluation of articulated arm coordinate measuring machines'.
- Barbero, B. R. et Ureta, E. S. (2011), 'Comparative study of different digitization techniques and their accuracy', *Computer-Aided Design* 43(2), 188–206.
- Becquet, M. (1987), Analysis of flexibility sources in robot structure, in 'IMACS/IFAC. Symp. Modeling and Simulation of Distributed Parameters, Hiroshima, Japan', pp. 419–424.
- BiPM, I., IFCC, I., ISO, I. et IUPAP, O. (2008), 'International vocabulary of metrology—basic and general concepts and associated terms, 2008', *JCGM* 200, 99–12.
- Bordron, M., Mehdi-Souzani, C. et Bruneau, O. (2018a), 'Inline measurement strategy for additive manufacturing', *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* p. 0954405418755826.
- Bordron, M., Mehdi-Souzani, C. et Bruneau, O. (2018b), 'An overview of an enhanced multi-systems robotized digitizing', *Computer-Aided Design and Applications* 15(4), 601–609.
- Borm, J.-H. et Meng, C.-H. (1991), 'Determination of optimal measurement configurations for robot calibration based on observability measure', *The International Journal of Robotics Research* 10(1), 51–63.
- Bouzgarrou, B., Fauroux, J., Gogu, G. et Heerah, Y. (2004), Rigidity analysis of t3r1 parallel robot with uncoupled kinematics, in 'Proc. of the 35th International Symposium on Robotics (ISR), Paris, France'.
- Brethé, J.-F., Vasselin, E., Lefebvre, D., Dakyo, B. et HAVRE, F. (2006), 'Modélisation spatio-temporelle de la répétabilité des robots manipulateurs industriels par les ellipsoïdes stochastiques', *CIFA06, Bordeaux* 30.
- Brosed, F. J., Aguilar, J. J., Guillomá, D. et Santolaria, J. (2010), '3d geometrical inspection of complex geometry parts using a novel laser triangulation sensor and a robot', *Sensors* 11(1), 90–110.
- Caenen, J.-L. (1993), Contribution à l'identification de paramètres géométriques et non géométriques d'un modèle de robot. Application à l'amélioration de la précision de positionnement statique, PhD thesis, Université de Valenciennes.
- Conrad, K. L., Shiakolas, P. S. et Yih, T. (2000), Robotic calibration issues: Accuracy, repeatability and calibration, in 'Proceedings of the 8th Mediterranean Conference on Control and Automation (MED2000), Rio, Patras, Greece'.

Bibliographie

- Coorevits, T., David, J. et Bourdet, P. (1991), 'Elimination of geometrical errors by permutations-application to a rotary table', *CIRP Annals-Manufacturing Technology* 40(1), 531–534.
- Damak, M. (1996), Théorie et instrumentation pour l'étalonnage statique des robots: Vers une programmation hors-ligne industriellement plus efficace, PhD thesis, ENSAM.
- Daney, D. (2002), Optimal measurement configurations for gough platform calibration, in 'Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on', Vol. 1, IEEE, pp. 147–152.
- Deblaise, D. (2008), Contribution à la modélisation et à l'étalonnage élasto-géométriques des manipulateurs à structure parallèle, PhD thesis, INSA de Rennes.
- Denavit, Jacques; Hartenberg, R. S. (1955), 'A kinematic notation for lower-pair mechanisms based on matrices.', *Trans. of the ASME. Journal of Applied Mechanics* 22, 215–221.
- Driels, M. R. et Pathre, U. S. (1990), 'Significance of observation strategy on the design of robot calibration experiments', *Journal of Robotic Systems* 7(2), 197–223.
- Dubreuil, L. (2017), Mesure In-situ par moyens optiques, PhD thesis, Université Paris-Saclay.
- Dumas, C., Caro, S., Garnier, S. et Furet, B. (2011), 'Joint stiffness identification of six-revolute industrial serial robots', *Robotics and Computer-Integrated Manufacturing* 27(4), 881–888.
- Elatta, A., Gen, L. P., Zhi, F. L., Daoyuan, Y. et Fei, L. (2004), 'An overview of robot calibration', *Information Technology Journal* 3(1), 74–78.
- Everett, L., Driels, M. et Mooring, B. (1987), Kinematic modelling for robot calibration, in 'Robotics and Automation. Proceedings. 1987 IEEE International Conference on', Vol. 4, IEEE, pp. 183–189.
- Flores, P. et Ambrósio, J. (2004), 'Revolute joints with clearance in multibody systems', *Computers & structures* 82(17-19), 1359–1369.
- Gong, C., Yuan, J. et Ni, J. (2000), 'Nongeometric error identification and compensation for robotic system by inverse calibration', *International Journal of Machine Tools and Manufacture* 40(14), 2119–2137.
- González-Madruga, D., Barreiro, J., Cuesta, E., González, B. et Martínez-Pellitero, S. (2014), 'Aacmm performance test: Influence of human factor and geometric features', *Procedia Engineering* 69, 442–448.
- Goswami, A., Quaid, A. et Peshkin, M. (1993), Complete parameter identification of a robot from partial pose information, in 'ICRA (1)', pp. 168–173.
- Guillo, M. et Dubourg, L. (2016), 'Impact & improvement of tool deviation in friction stir welding: Weld quality & real-time compensation on an industrial robot', *Robotics and Computer-Integrated Manufacturing* 39, 22–31.
- Guo, Y., Dong, H. et Ke, Y. (2015), 'Stiffness-oriented posture optimization in robotic machining applications', *Robotics and Computer-Integrated Manufacturing* 35, 69–76.
- Guyot, G. (1995), Contributions à l'étalonnage géométrique des robots manipulateurs, PhD thesis, Nice.
- Hayati, S. A. (1983), Robot arm geometric link parameter estimation, in 'Decision and Control, 1983. The 22nd IEEE Conference on', IEEE, pp. 1477–1483.
- He, R., Zhao, Y., Yang, S. et Yang, S. (2010), 'Kinematic-parameter identification for serial-robot calibration based on poe formula', *IEEE Transactions on Robotics* 26(3), 411–423.

Bibliographie

- Hollerbach, J. M. (1989), A survey of kinematic calibration, in 'The robotics review 1', MIT Press, pp. 207–242.
- Hollerbach, J. M. et Lokhorst, D. M. (1995), 'Closed-loop kinematic calibration of the rsi 6-dof hand controller', *IEEE Transactions on Robotics and Automation* 11(3), 352–359.
- Hollerbach, J. M. et Wampler, C. W. (1996), 'The calibration index and taxonomy for robot kinematic calibration methods', *The international journal of robotics research* 15(6), 573–591.
- ISO.230-1 (2012), 'Code d'essai des machines-outils – partie 1: Exactitude géométrique des machines fonctionnant à vide ou dans des conditions quasi-statiques'.
- ISO.9283 (2015), 'Robots manipulateurs industriels – critères de performance et méthodes d'essai correspondantes'.
- JCGM.200 (2012), 'Vocabulaire international de métrologie - concepts fondamentaux et généraux et termes associés (vim)'.
- Judd, R. P. et Knasinski, A. B. (1990), 'A technique to calibrate industrial robots with experimental verification', *Robotics and Automation, IEEE Transactions on* 6(1), 20–30.
- Khalil, W. et Besnard, S. (2002), 'Geometric calibration of robots with flexible joints and links', *Journal of Intelligent and Robotic systems* 34(4), 357–379.
- Khalil, W. et Dombre, E. (2004), *Modeling, identification and control of robots*, Butterworth-Heinemann.
- Khalil, W., Gautier, M. et Enguehard, C. (1991), 'Identifiable parameters and optimum configurations for robots calibration', *Robotica* 9(01), 63–70.
- Khalil, W. et Kleinfinger, J. (1986), A new geometric notation for open and closed-loop robots, in 'Robotics and Automation. Proceedings. 1986 IEEE International Conference on', Vol. 3, IEEE, pp. 1174–1179.
- Khan, W. A. et Angeles, J. (2006), 'The kinetostatic optimization of robotic manipulators: the inverse and the direct problems', *Journal of mechanical design* 128(1), 168–178.
- Kim, T. et Sarma, S. E. (2002), 'Toolpath generation along directions of maximum kinematic performance; a first cut at machine-optimal paths', *Computer-Aided Design* 34(6), 453–468.
- Klimchik, A., Caro, S., Wu, Y., Chablat, D., Furet, B. et Pashkevich, A. (2014), Stiffness modeling of robotic manipulator with gravity compensator, in 'Computational Kinematics', Springer, pp. 185–192.
- Klimchik, A., Furet, B., Caro, S. et Pashkevich, A. (2015), 'Identification of the manipulator stiffness model parameters in industrial environment', *Mechanism and Machine Theory* 90, 1–22.
- Klimchik, A., Pashkevich, A., Caro, S. et Chablat, D. (2012), Stiffness modeling of robotic-manipulators under auxiliary loadings, in 'ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference', American Society of Mechanical Engineers, pp. 469–476.
- Larsson, S. et Kjellander, J. (2006), 'Motion control and data capturing for laser scanning with an industrial robot', *Robotics and Autonomous Systems* 54(6), 453–460.
- Lee, K.-M. et Shah, D. K. (1988), 'Dynamic analysis of a three-degrees-of-freedom in-parallel actuated manipulator', *IEEE Journal on Robotics and Automation* 4(3), 361–367.

Bibliographie

- Li, R. et Zhao, Y. (2016), 'Dynamic error compensation for industrial robot based on thermal effect model', *Measurement* 88, 113–120.
- Liegeois, A. (1977), 'Automatic supervisory control of the configuration and behavior of multibody mechanisms', *IEEE transactions on systems, man, and cybernetics* 7(12), 868–871.
- Lin, Y., Zhao, H. et Ding, H. (2017), 'Posture optimization methodology of 6r industrial robots for machining using performance evaluation indexes', *Robotics and Computer-Integrated Manufacturing* 48, 59–72.
- Lu, T.-F., Lin, G. C. et He, J. R. (1997), 'Neural-network-based 3d force/torque sensor calibration for robot applications', *Engineering Applications of Artificial Intelligence* 10(1), 87–97.
- Mahmud, M., Joannic, D., Roy, M., Isheil, A. et Fontaine, J.-F. (2011), '3d part inspection path planning of a laser scanner with control on the uncertainty', *Computer-Aided Design* 43(4), 345–355.
- Marie, S., Courteille, E. et Maurine, P. (2013), 'Elasto-geometrical modeling and calibration of robot manipulators: Application to machining and forming applications', *Mechanism and Machine Theory* 69, 13–43.
- Meggiolaro, M. A., Dubowsky, S. et Mavroidis, C. (2005), 'Geometric and elastic error calibration of a high accuracy patient positioning system', *Mechanism and Machine Theory* 40(4), 415–427.
- Mehdi-Souzani, C., Quinsat, Y., Lartigue, C. et Bourdet, P. (2016), 'A knowledge database of qualified digitizing systems for the selection of the best system according to the application', *CIRP Journal of Manufacturing Science and Technology* 13, 15–23.
- Mineo, C., Pierce, S. G., Nicholson, P. I. et Cooper, I. (2016), 'Robotic path planning for non-destructive testing—a custom matlab toolbox approach', *Robotics and Computer-Integrated Manufacturing* 37, 1–12.
- Mooring, B. (1984), An improved method for identifying the kinematic parameters in a six axis robots, in 'Computers in Engineering, Proceedings of the International Computers in Engineering Conference and Exhibit, 1984', Vol. 1, pp. 79–84.
- Mooring, B. W., Roth, Z. S. et Driels, M. R. (1991), *Fundamentals of manipulator calibration*, Wiley New York.
- Moreno, H. A., Saltaren, R., Carrera, I., Puglisi, L. et Aracil, R. (2012), 'ndices de desempeño de robots manipuladores: una revisión del estado del arte', *Revista Iberoamericana de Automática e Informática Industrial RIAI* 9(2), 111–122.
- Moreno, H. A., Saltaren, R., Pamanes, A. et Aracil, R. (2010), Motion strategy for the trepa climbing robot on a metallic orthogonal structure, in 'Emerging Trends In Mobile Robotics', World Scientific, pp. 1161–1168.
- Nahvi, A. et Hollerbach, J. M. (1996), The noise amplification index for optimal pose selection in robot calibration, in 'Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on', Vol. 1, IEEE, pp. 647–654.
- Nahvi, A., Hollerbach, J. M. et Hayward, V. (1994), Calibration of a parallel robot using multiple kinematic closed loops, in 'ICRA', pp. 407–412.
- Neuman, C. et Khosla, P. (1985), 'Parameter identification for robot control', *Dynamic Systems: Modelling and control* pp. 213–223.

Bibliographie

- Nguyen, H.-N., Zhou, J. et Kang, H.-J. (2013), 'A new full pose measurement method for robot calibration', *Sensors* 13(7), 9132–9147.
- Nubiola, A. (2011), Calibration of a serial robot using a laser tracker, PhD thesis, École de technologie supérieure.
- Nubiola, A. et Bonev, I. A. (2013), 'Absolute calibration of an abb irb 1600 robot using a laser tracker', *Robotics and Computer-Integrated Manufacturing* 29(1), 236–245.
- Olabi, A. (2011), Amélioration de la précision des robots industriels pour des applications d'usinage à grande vitesse, PhD thesis, Arts et Métiers ParisTech.
- Olabi, A., Damak, M., Bearee, R., Gibaru, O. et Leleu, S. (2012), Improving the accuracy of industrial robots by offline compensation of joints errors, in 'Industrial Technology (ICIT), 2012 IEEE International Conference on', IEEE, pp. 492–497.
- Pashkevich, A. (2001), Computer-aided generation of complete irreducible models for robotic manipulators, in 'Int. Conference of Modelling and Simulation', pp. 293–298.
- Pashkevich, A., Klimchik, A. et Chablat, D. (2011), 'Enhanced stiffness modeling of manipulators with passive joints', *Mechanism and machine theory* 46(5), 662–679.
- Phan, N. D. M., Quinsat, Y., Lavernhe, S. et Lartigue, C. (2018), 'Scanner path planning with the control of overlap for part inspection with an industrial robot', *The International Journal of Advanced Manufacturing Technology* pp. 1–15.
- Piras, G., Cleghorn, W. et Mills, J. (2005), 'Dynamic finite-element analysis of a planar high-speed, high-precision parallel manipulator with flexible links', *Mechanism and machine theory* 40(7), 849–862.
- Posada, J. D., Schneider, U., Pidan, S., Geravand, M., Stelzer, P. et Verl, A. (2016), High accurate robotic drilling with external sensor and compliance model-based compensation, in 'Robotics and Automation (ICRA), 2016 IEEE International Conference on', IEEE, pp. 3901–3907.
- Rolland, L. H. (2003), Outils algébriques pour la résolution de problèmes géométriques et l'analyse de trajectoire de robots parallèles prévus pour des applications à haute cadence et grande précision, PhD thesis, Nancy 1.
- Roth, Z., Mooring, B. et Ravani, B. (1987), 'An overview of robot calibration', *IEEE Journal on Robotics and Automation* 5(3), 377–385.
- Sansoni, G., Trebeschi, M. et Docchio, F. (2009), 'State-of-the-art and applications of 3d imaging sensors in industry, cultural heritage, medicine, and criminal investigation', *Sensors* 9(1), 568–601.
- Santolaria, J., Aguilar, J.-J., Yagüe, J.-A. et Pastor, J. (2008), 'Kinematic parameter estimation technique for calibration and repeatability improvement of articulated arm coordinate measuring machines', *Precision Engineering* 32(4), 251–268.
- Santolaria, J., Conte, J. et Ginés, M. (2013), 'Laser tracker-based kinematic parameter calibration of industrial robots by improved cpa method and active retroreflector', *The International Journal of Advanced Manufacturing Technology* 66(9-12), 2087–2106.
- Santolaria, J. et Ginés, M. (2013), 'Uncertainty estimation in robot kinematic calibration', *Robotics and Computer-Integrated Manufacturing* 29(2), 370–384.

Bibliographie

- Santolaria, J., Guillomá, D., Cajal, C., Albajez, J. A. et Aguilar, J. J. (2009), 'Modelling and calibration technique of laser triangulation sensors for integration in robot arms and articulated arm coordinate measuring machines', *Sensors* 9(9), 7374–7396.
- Savio, E., De Chiffre, L. et Schmitt, R. (2007), 'Metrology of freeform shaped parts', *CIRP annals* 56(2), 810–835.
- Schneider, U., Posada, J. R. D. et Verl, A. (2015), Automatic pose optimization for robotic processes, in 'Robotics and Automation (ICRA), 2015 IEEE International Conference on', IEEE, pp. 2054–2059.
- Sharifzadeh, S., Biro, I., Lohse, N. et Kinnell, P. (2018), 'Abnormality detection strategies for surface inspection using robot mounted laser scanners', *Mechatronics* 51, 59–74.
- Sklar, M. E. (1989), Geometric calibration of industrial manipulators by circle point analysis, in 'Proceedings of the 2nd Conference on Recent Advances in Robotics', Vol. 178202.
- Stone, H., Sanderson, A. et Neuman, C. (1986), Arm signature identification, in 'Robotics and Automation. Proceedings. 1986 IEEE International Conference on', Vol. 3, IEEE, pp. 41–48.
- Sun, Y. et Hollerbach, J. M. (2008), Observability index selection for robot calibration, in 'Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on', IEEE, pp. 831–836.
- Tian, Q., Flores, P. et Lankarani, H. M. (2018), 'A comprehensive survey of the analytical, numerical and experimental methodologies for dynamics of multibody mechanical systems with clearance or imperfect joints', *Mechanism and Machine Theory* 122, 1–57.
- Timoshenko, S. P. (1922), 'X. on the transverse vibrations of bars of uniform cross-section', *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 43(253), 125–131.
- Van Gestel, N., Cuypers, S., Bleys, P. et Kruth, J.-P. (2009), 'A performance evaluation test for laser line scanners on cmms', *Optics and lasers in engineering* 47(3-4), 336–342.
- Vergnano, A., Thorstensson, C., Lennartson, B., Falkman, P., Pellicciari, M., Leali, F. et Biller, S. (2012), 'Modeling and optimization of energy consumption in cooperative multi-robot systems', *IEEE Transactions on Automation Science and Engineering* 9(2), 423–428.
- Vosniakos, G.-C. et Matsas, E. (2010), 'Improving feasibility of robotic milling through robot placement optimisation', *Robotics and Computer-Integrated Manufacturing* 26(5), 517–525.
- Vosniakos, G. et Chronopoulos, A. (2009), 'Industrial robot path planning in a constraint-based computer-aided design and kinematic analysis environment', *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 223(5), 523–533.
- Wang, G., Dong, H., Guo, Y. et Ke, Y. (2017), 'Finite element simulation and experimental analysis of robotic boring based on an approach of equivalent stiffness', *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* p. 0954405416683430.
- Weill, R. et Shani, B. (1991), 'Assessment of accuracy of robots in relation with geometrical tolerances in robot links', *CIRP Annals-Manufacturing Technology* 40(1), 395–399.
- Whitney, D. E., Lozinski, C. A. et Rourke, J. M. (1985), *Industrial robot calibration method and results*, Department of Mechanical Engineering, MIT.
- Wu, Y. (2014), Optimal pose selection for the identification of geometric and elastostatic parameters of machining robots, PhD thesis, Ecole des Mines de Nantes.

Bibliographie

- Wu, Y., Klimchik, A., Caro, S., Furet, B. et Pashkevich, A. (2015), 'Geometric calibration of industrial robots using enhanced partial pose measurements and design of experiments', *Robotics and Computer-Integrated Manufacturing* 35, 151–168.
- Xuan, J.-Q., Xu, S.-H. et al. (2014), 'Review on kinematics calibration technology of serial robots', *International Journal of Precision Engineering and Manufacturing* 15(8), 1759–1774.
- Yang, Z. et Sadler, J. (1992), 'Finite element analysis of revolute manipulators with link and joint compliance by joint beam elements', *ASME Design Engineering Division, Robotics, Spatial Mechanisms, and Mechanical Systems* 45, 619–625.
- Yin, J. et Gao, Y. (2012), 'Pose accuracy calibration of a serial five dof robot', *Energy Procedia* 14, 977–982.
- Yoshikawa, T. (1985), 'Manipulability of robotic mechanisms', *The international journal of Robotics Research* 4(2), 3–9.
- Zargarbashi, S., Khan, W. et Angeles, J. (2012), 'The jacobian condition number as a dexterity index in 6r machining robots', *Robotics and Computer-Integrated Manufacturing* 28(6), 694–699.
- Zeng, Y., Tian, W. et Liao, W. (2016), 'Positional error similarity analysis for error compensation of industrial robots', *Robotics and Computer-Integrated Manufacturing* 42, 113–120.
- Zha, X. F. (2002), 'Optimal pose trajectory planning for robot manipulators', *Mechanism and Machine Theory* 37(10), 1063–1086.
- Zhang, H., Wang, J., Zhang, G., Gan, Z., Pan, Z., Cui, H. et Zhu, Z. (2005), Machining with flexible manipulator: toward improving robotic machining performance, in 'Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on', IEEE, pp. 1127–1132.
- Zhang, J., Fang, X. et Qi, L. (2016), 'Sensitivity-analysis based method in single-robot cells cost-effective design and optimization', *Robotics and Computer-Integrated Manufacturing* 38, 9–15.
- Zhuang, H. (1989), Kinematic modeling, identification and compensation of robot manipulators, PhD thesis, Florida atlantic university.
- Zhuang, H. et Roth, Z. S. (1992), 'Robot calibration using the cpc error model', *Robotics and computer-integrated manufacturing* 9(3), 227–237.
- Zhuang, H., Wang, L. K. et Roth, Z. S. (1993), 'Error-model-based robot calibration using a modified cpc model', *Robotics and computer-integrated manufacturing* 10(4), 287–299.
- Ziegert, J. et Datseris, P. (1988), Basic considerations for robot calibration, in 'Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on', IEEE, pp. 932–938.

Annexes

Annexe 1 – Défauts dans une articulation (section 1.3.3)

La norme ISO 230-1 permet de fixer le vocabulaire à employer dans les articulations ainsi que les différents défauts de position et d'orientation par rapport à la ligne moyenne de l'axe de liaison pouvant occurrer [ISO.230-1 2012] :

- d_r : Erreur de mouvement radial (perpendiculaire à la ligne moyenne)
- d_z : Erreur de mouvement axial (coaxial à la ligne moyenne)
- δ_r : Erreur de mouvement d'inclinaison (angle par rapport à la ligne moyenne)
- δ_z : Erreur de positionnement angulaire du mouvement

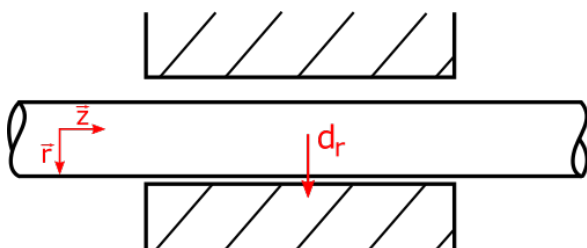


Figure 113 : Erreur de mouvement radial

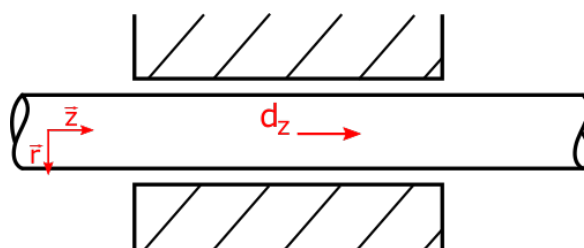


Figure 114 : Erreur de mouvement axial

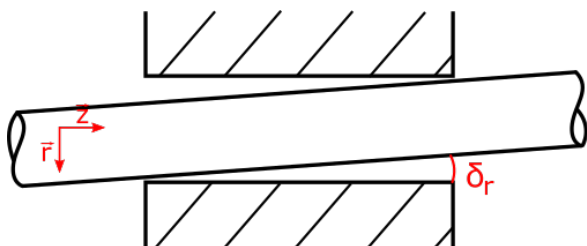


Figure 115 : Erreur de mouvement d'inclinaison

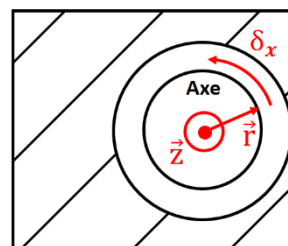


Figure 116 : Erreur de positionnement angulaire

Annexe 2 - Nombre de points minimum sur un arc pour une bonne identification par CPA modifiée (section 2.3.1)

Lors de l'étape de mesure pour l'identification des paramètres par notre méthode CPA modifiée, la longueur des arcs est limitée par le champ de vision du moyen de mesure (le C-Track). Il faut cependant déterminer la vitesse du robot lorsque ce dernier suivra ces arcs. Si le robot est rapide, l'étape de mesure sera courte, mais le nombre de points sera trop faible pour une bonne estimation du positionnement des axes du robot. Mais il faut tout de même chercher à avoir une étape de mesure la plus rapide possible. C'est dans ce cadre que nous estimons le nombre de points nécessaires sur un arc pour une bonne estimation du positionnement des axes et une bonne vitesse d'exécution. Connaissant la fréquence d'acquisition du C-Track (29 Hz) il sera alors possible de déterminer la vitesse du robot.

Une qualification du C-Track a été effectuée en section 1.5.2 et ses performances sont connues. L'incertitude ou la répétabilité sur la mesure d'un point est de $3,8\mu\text{m}$ suivant l'axe X, de $6,3\mu\text{m}$ suivant Y et de $21,4\mu\text{m}$ suivant Z. À partir de ces informations il faut à présent remonter à l'incertitude sur l'estimation du positionnement des axes du robot. Cependant cette estimation passe par l'utilisation de la méthode des moindres carrés, il est donc impossible d'avoir une formule donnant l'incertitude sur les axes à partir de l'incertitude du C-Track sur chaque point mesuré. Une étude statistique est donc mise en place, où un nombre suffisant de simulations sont faites pour avoir un résultat fiable sur la propagation de l'incertitude. En suivant le protocole de la figure 117, il est alors possible d'avoir l'incertitude sur le

positionnement d'un axe en renseignant le nombre de points sur l'arc de cercle, et en fixant d'autres paramètres sur l'arc liés à l'axe choisi (1 à 6) c'est-à-dire :

- La longueur de l'arc, qui va grandement influencer l'incertitude. Il y a en réalité le rayon du cercle qui dépend de l'axe étudié, ce rayon est grand pour l'axe 1 et petit pour l'axe 6, et il y a également la proportion d'arc par rapport au cercle, un arc faisant 1/4 du cercle donnera de meilleurs résultats qu'un arc faisant 1/8 du cercle.
- L'orientation de l'arc dans le repère C-Track, car ce dernier n'a pas la même incertitude suivant X, Y ou Z.

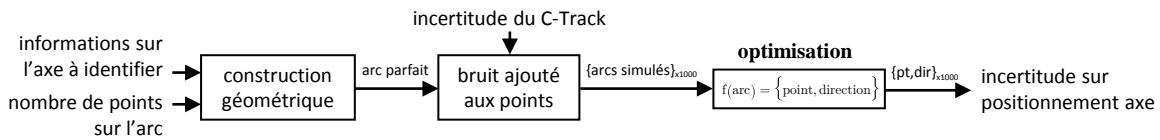


Figure 117 : Calcul de l'incertitude sur l'estimation d'un axe du robot

Dans notre cas nous déterminons un nombre de points identiques sur chaque arc, quel que soit l'axe considéré, ce nombre est déterminé à partir de l'axe le plus contraignant. Il est bien sûr possible de déterminer un nombre de points différent et adapté à chaque axe. Afin d'obtenir une faible incertitude sur l'estimation de l'axe de rotation et de ne pas ajouter de points superflus dans l'étude, le nombre de points est fixé par rapport à la convergence de la courbe présentée sur la figure 118. Sur cette figure, qui représente l'incertitude sur l'estimation du centre du cercle, deux droites permettent de visualiser les bornes de la convergence de la courbe. Les bornes sont dues au bruit aléatoire de notre étude statistique, elles délimitent donc la convergence de la courbe au bruit près. Dès que la courbe entre dans ces bornes elle n'en ressort pas et oscille avec le bruit, à partir de ce moment-là il n'est plus utile d'ajouter des points, on voit que la qualité avec 3000 points est la même qu'avec 4000. Notre critère de sélection est donc le moment où la courbe rentre dans ces bornes, c'est-à-dire autour de 2000 points, toute différence étant négligeable passé ce cap vu le bruit dans notre étude statistique.

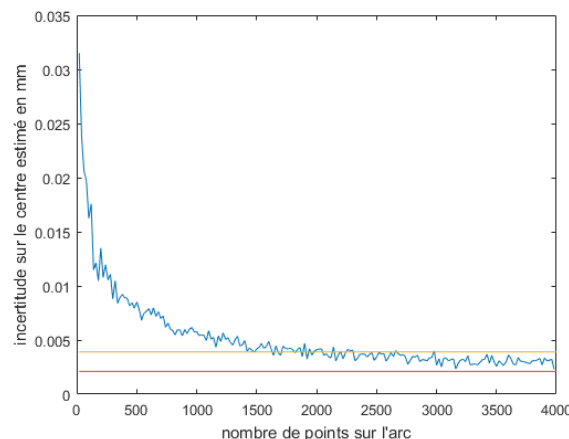


Figure 118 : Convergence de l'incertitude sur l'estimation du centre, par rapport au nombre de points sur l'arc

Annexe 3 - Etude d'une amélioration de la méthode des moindres carrés appliquée aux arcs de cercle (section 2.4)

Dans le cadre de l'estimation de centre du cercle associé à un arc, et afin d'identifier des paramètres par méthode CPA, la méthode des moindres carrés est appliquée à l'estimation d'un cercle à partir d'un arc. Il s'avère qu'étant donné la petite taille des arcs, cette méthode apporte de l'incertitude que l'on peut chercher à réduire. Dans ce but une autre méthode est investiguée ici. Pour visualiser cette incertitude, une simulation de 1000 essais a été réalisée, estimant le centre du cercle associé à un arc. Le nuage de

point constituant l'arc est généré à partir d'un arc parfait, puis chaque point est modifié aléatoirement en se basant sur l'incertitude du moyen de mesure, et en suivant une répartition gaussienne. L'identification statistique des centres est présentée sur la figure 119 où l'on retrouve les centres identifiés, l'orientation de l'arc par rapport aux centres, et en rouge le centre théorique de départ.

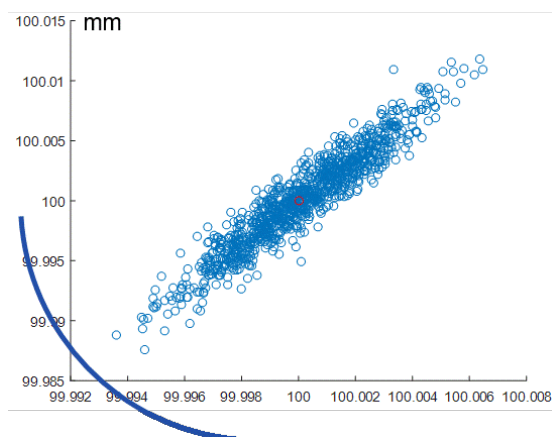


Figure 119 : Centres identifiés pour 1000 simulations

La figure nous montre que la méthode des moindres carrés pour identifier un centre à partir d'un arc donne un bon résultat suivant une direction. En effet l'ensemble des centres estimés par simulation se trouve sur la médiatrice de l'arc. Ainsi si l'on estime le centre d'un arc avec cette méthode, on peut calculer une droite Δ passant par ce centre et par le milieu de l'arc, et cette droite Δ sera très proche de la médiane réelle. On peut donc supposer que le vrai centre se trouve sur cette droite Δ (ou proche). Maintenant, si l'on sépare l'arc en 2 arcs et que l'on applique ce raisonnement aux 2 arcs, on obtient alors Δ_1 et Δ_2 deux droites proches des médianes des arcs 1 et 2 comme sur la figure 120. L'intersection de Δ_1 et Δ_2 devrait donc logiquement donner un point très proche du centre réel. Nous avons appliqué cette technique sur 1000 simulations et obtenons les résultats de la figure 121.

Malheureusement on voit sur cette figure que cette technique n'apporte aucune amélioration sur l'estimation du centre à partir d'un arc. L'écart moyen des points sur les premières simulations est de 0.0043mm, et l'écart moyen sur les 1000 simulations avec cette nouvelle méthode est de 0.0044mm. La méthode en elle-même permet d'obtenir le centre plus précisément, mais le fait de séparer l'arc en deux donne deux médiatrices moins représentatives des médiatrices réelles, ce qui éloigne finalement leur intersection du centre théorique. De nombreuses autres méthodes ont été testées sans succès, à base de polynômes et de médiatrices des arcs. La méthode des moindres carrés classique appliquée au cercle reste la plus appropriée.

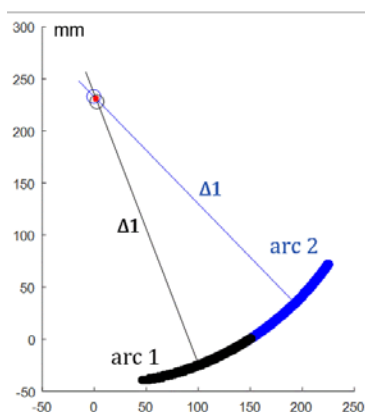


Figure 120 : Intersection des médiatrices de 2 arcs

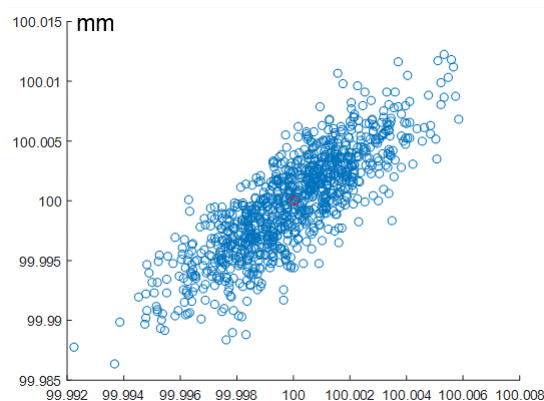


Figure 121 : Centres identifiés pour 1000 simulations, à partir de l'intersection des médiatrices

Annexe 4 - Lien entre l'indice de manipulabilité et l'ellipsoïde de répétabilité de Brethé (section 4.2)

Nous allons à présent démontrer le lien étroit qu'il y a entre l'ellipsoïde proposé par Brethé et l'indice de manipulabilité. De la même façon que la manipulabilité est proportionnelle au volume de l'ellipsoïde de vitesse, voici le volume de l'ellipsoïde de répétabilité :

$$V = \sqrt{\lambda_1} \cdot \sqrt{\lambda_2} \cdot \dots \cdot \sqrt{\lambda_6} \quad \text{Eq. 245}$$

Nous utilisons l'équation 220 donnant les valeurs des demi-axes de l'ellipsoïde. Sachant que le déterminant d'une matrice est égal au produit de ses valeurs propres, on peut réécrire l'équation précédente :

$$V = \sqrt{\det(C)} = \sqrt{\det(J_\theta \cdot D \cdot J_\theta^t)} = \sqrt{\det(J_\theta) \cdot \det(D) \cdot \det(J_\theta^t)} \quad \text{Eq. 246}$$

Or les déterminants d'une matrice et de sa transposée sont les mêmes. De plus la manipulabilité w étant le produit des valeurs propres de la jacobienne articulaire, on la voit apparaître dans l'équation.

$$V = \sqrt{w^2 \cdot \sigma_1^2 \cdot \sigma_2^2 \cdot \dots \cdot \sigma_6^2} = w \cdot \sigma_1 \cdot \sigma_2 \cdot \dots \cdot \sigma_6 \quad \text{Eq. 247}$$

Le volume de l'ellipsoïde de répétabilité V est donc proportionnel à la manipulabilité w . On peut donc calculer V à partir de la formule de la manipulabilité et des répétabilités sur chaque articulation. Ce volume peut donc être utilisé comme un indice simple représentant la répétabilité globale sur l'effecteur dans une configuration donnée. Mais nous avons préféré créer un indice nous donnant la valeur de répétabilité la plus contraignante, afin d'être certain d'assurer une qualité de pose.

Annexe 5 – Développement de la communication entre le C-Track et le Zephyr

L'objectif ici est de faire parvenir au capteur laser plan, l'information sur sa pose (position et orientation) dans le repère de numérisation, qui pour nous sera le repère externe (repère C-Track). Le contrôleur du capteur laser Zephyr dispose d'entrées permettant de recevoir l'information de position de la part du système porteur du capteur, dans un format spécifique (signal TTL). Habituellement le Zephyr est monté sur une MMT qui sert de système porteur, et les positions relevées sur les axes $\{X,Y,Z\}$ de la MMT sont compatibles avec les entrées du contrôleur. Dans le contexte de notre cellule de numérisation, le système porteur est un robot et la pose du Zephyr (fixé sur l'effecteur) est fournie par le système de suivi externe C-Track. Cependant, la pose fournie par le C-Track n'est pas compatible avec les entrées du contrôleur du Zephyr. Nous avons donc mis en place une interface logicielle permettant la communication de cette pose du C-Track au Zephyr.

Notre interface interagit avec les logiciels de gestion du C-Track et du Zephyr qui s'appellent respectivement VXelements et Polygonia. L'information sur la pose du capteur laser est donc fournie par le logiciel VXelements, puis transmise au logiciel Polygonia qui génère le nuage de point résultant de la numérisation. Du côté de VXelements, une bibliothèque de fonction (une API) peut être chargée par notre interface pour avoir accès à toutes les fonctionnalités de VXelements. De l'autre côté, Polygonia propose un toolkit qui est un exécutable remplaçant le logiciel de base et dont les fonctionnalités peuvent être modifiées. Dans notre cas, des modifications permettent la prise en compte de l'orientation du capteur laser en plus de sa position, et permettent de recevoir l'information sur cette pose par entrée logicielle et non plus par les entrées du contrôleur du Zephyr.

Le toolkit de Polygonia doit alors charger une bibliothèque de fonctions (une dll) lui donnant accès à l'information sur la pose du Zephyr, mais l'API de VXelements n'est pas compatible. En effet, bien que cette API soit une librairie (dll), elle est managée (gérée par le Common Language Runtime), mais le

toolkit ne peut charger que des dll non managées. C'est pourquoi nous avons développé en plus, notre programme servant d'interface appelé communément un wrapper. Cette interface est capable de charger l'API de VXelements et est elle-même une dll non managée, que le toolkit va pouvoir utiliser. De plus notre interface permet d'adapter les fonctions chargées de l'API pour qu'elles correspondent aux fonctions appelées par le toolkit. La communication mise en place entre le C-Track et le Zephyr est résumée sur la figure 122.

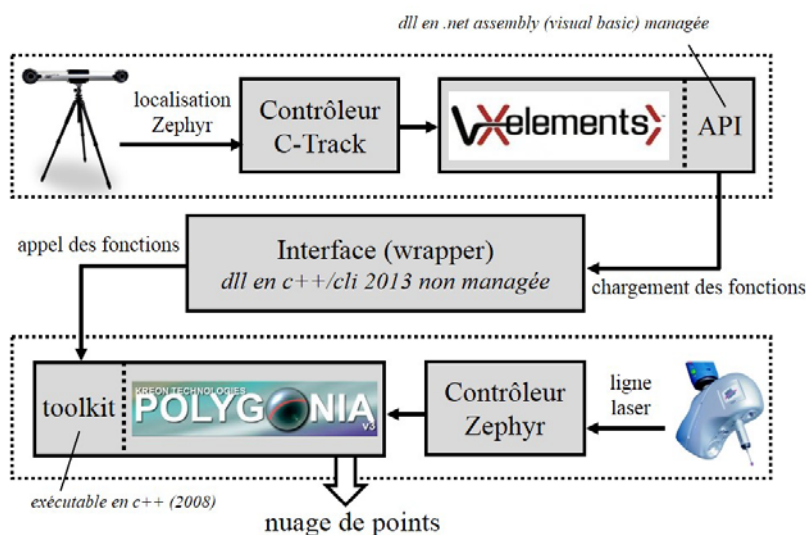


Figure 122 : Communication VXelements, Polygonia

Afin de prendre en charge le cas où l'information sur la pose du capteur laser plan est donnée par l'ensemble {modèle du robot, coordonnées articulaires}, notre interface va évoluer pour pouvoir réceptionner les coordonnées fournies par les codeurs sur les axes moteurs et incorporer le modèle du robot. Il sera ainsi possible de fonctionner de 3 façons différentes : uniquement avec le C-Track, uniquement avec les retours codeurs, avec le C-Track et les retours codeurs quand les cibles ne sont plus visibles.

Dans les travaux de cette thèse, les numérisations sont réalisées à faible vitesse, mais lorsque la méthode sera complète, la vitesse de la numérisation sera optimisée. Il faudra alors prendre en compte le décalage entre la mesure d'une ligne laser et la réception de l'information sur la pose du capteur laser plan. Pour cela un trigger sera mis en place entre les contrôleurs.

Annexe 6 – Calibration existante et 1^{ère} numérisation

Les informations transmises au capteur laser plan sur sa pose dans le repère externe (repère de la numérisation), ne donnent en vérité pas la position du capteur lui-même, mais celle de l'ensemble de cibles qui lui est associé. Une calibration est nécessaire pour faire le lien entre cette information de pose fournie et la pose du capteur dans le repère externe. Une méthode proposée par le logiciel Polygonia permet d'effectuer cette calibration en position et en orientation du capteur Zephyr en calculant une matrice de passage situant le Zephyr par rapport à l'objet réellement suivi (ici l'ensemble de cibles). Dans cette méthode une sphère fixe doit être numérisée par notre cellule de numérisation (Zephyr porté par robot et pose fournie par C-Track), sous différentes orientations et position. La matrice de passage est alors calculée par le logiciel, afin d'approcher le plus possible la forme théorique de la sphère avec le nuage de points résultants. Cette matrice de passage est ensuite automatiquement incluse dans le logiciel après calibration.

Le but de la calibration étant de connaître la position mais aussi l'orientation entre l'ensemble de cibles et le capteur Zephyr (repère interne au capteur), l'opérateur doit numériser la sphère avec un grand nombre d'orientations variées. Nous avons appliqué cette calibration pour notre cellule de numérisation

où le déplacement du capteur est assuré par notre robot. Cette procédure de calibration en position et orientation proposée par Polygonia est adaptée pour des systèmes porteurs 6 axes qui peuvent être manipulés à la main (systèmes polyarticulés) mais s'avère peu pratique dans le cas d'un robot. En effet, la programmation des nombreuses orientations autour de la sphère est longue et méticuleuse (la trace laser doit être bien positionnée). C'est une des raisons qui nous pousse à développer nos propres méthodes de calibration, plus adaptées à notre contexte de cellule de numérisation robotisée.

A la suite de cette calibration nous devrions avoir une cellule de numérisation opérationnelle, utilisant un robot comme support du capteur laser plan (Zephyr) et un système de suivi externe (C-Track) pour localiser le Zephyr. Afin de vérifier simplement la validité de notre cellule, nous choisissons de numériser une pièce test. Cette même pièce pourra être numérisée avec différentes orientations du capteur afin de tester la qualité de calibration en orientation effectuée. La pièce choisie est un visage présenté sur la figure 123.



Figure 123 : Pièce test, visage

La numérisation de ce visage en 3 passes avec notre cellule de mesure, et la calibration proposée par Polygonia, donne les résultats visibles sur les figures figure 124 et figure 125.

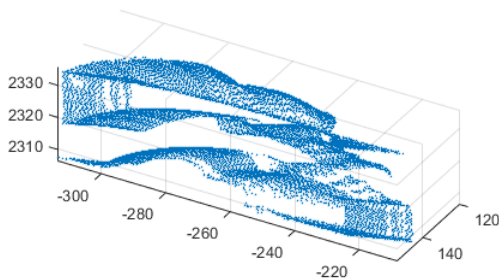


Figure 124 : Numérisation du visage, vue isométrique

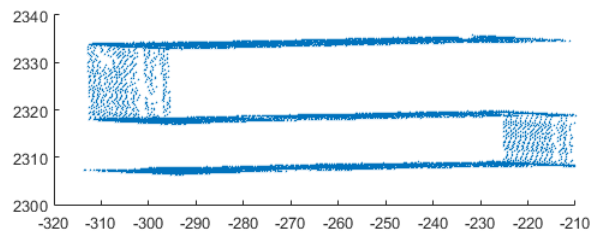


Figure 125 : Numérisation du visage, vue de dessus (plan XZ)

Sur les figures précédentes, le visage est difficilement reconnaissable. Il y a donc une erreur dans l'opération de numérisation. Après vérification, la qualité de suivi du C-track n'est pas en cause. Par ailleurs, les lignes laser, si elles sont prises séparément les unes des autres, sont correctes. L'erreur apparaît donc lorsque ces lignes sont réexprimées dans un repère commun (repère externe). La matrice de passage entre l'ensemble de cibles et le capteur Zephyr (repère interne) est donc incorrecte, ce qui fausse la position et l'orientation des lignes laser dans le repère externe. Une erreur en position dans cette matrice, ne peut engendrer qu'un mouvement de corps rigide du nuage complet sans déformer la numérisation. L'erreur visible ici vient donc plutôt de la calibration en orientation du capteur par rapport aux cibles ce qui engendre un problème d'orientation des lignes laser.

La méthode de calibration proposée dans le logiciel Polygonia, présente donc des défaillances qui ne permettent pas, dans notre cas, une calibration correcte du passage de l'ensemble de cibles suivi au capteur Zephyr. On peut en parti expliquer le problème de compatibilité entre cette méthode et notre contexte : l'inclinaison maximale des cibles pour qu'elles restent visibles, a limité la variété des

orientations pouvant être atteinte lors de la calibration. La matrice de passage qui est incorrecte ici, n'est pas accessible, et ne peut être modifiée que par la calibration de Polygonia. Ainsi pour pouvoir corriger cette erreur, nous choisissons d'appliquer une matrice de correction (en orientation), avant l'envoi de l'information de pose au Zephyr. Cette matrice de correction est donc placée dans notre interface (wrapper) qui reçoit la pose par le C-Track pour l'envoyer au Zephyr.

Afin de vérifier notre hypothèse sur la source d'erreur (calibration en orientation), nous mettons en place notre matrice de correction avec des valeurs appropriées. Ici, à partir de la figure 124, il est possible de déduire que l'erreur principale d'orientation est une rotation de 90° autour de l'axe Y (axes X et Z sur figure 125). Cette rotation de 90° autour de l'axe Y est appliquée via notre matrice de correction, juste avant l'envoi de la pose au Zephyr. La figure 126 montre le résultat de la numérisation effectuée après cette correction.

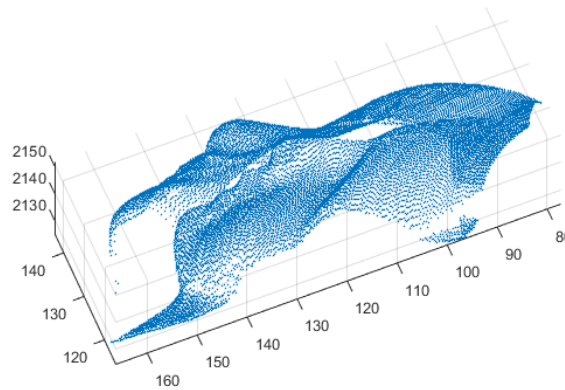


Figure 126 : Numérisation visage, avec correction grossière de la pose du capteur par rapport aux cibles

Cette correction nous permet de conclure qu'effectuer la rotation corrective adéquate permettra d'obtenir le bon nuage de points. Sur la figure précédente, le nuage n'est pas encore correct, car d'autres rotations correctives sont à identifier pour obtenir la bonne matrice de correction. Une méthode de calibration doit alors être mise en place pour identifier ces rotations. C'est pourquoi, nous proposons des méthodes de calibration permettant d'identifier les valeurs de notre matrice de correction (en orientation). De plus, ces méthodes sont pratiques car adaptées à notre contexte (cellule robotisée) contrairement à la méthode proposée par le logiciel Polygonia.

Abstract

Initially serial robots were designed for repetitive tasks requiring good dexterity in wide workspaces, while providing flexibility to the production line. New restrictive applications such as complex parts digitizing can make use of those attractive benefits. However, digitizing needs on accuracy require to work carefully on the quality of the robot end-effector positioning.

In this context, a digitizing cell using a 6 axis robot as displacement system for a laser plane sensor (KZ25) was developed. In this cell all calibrations are ensured by an external measurement system (C-Track Creaform) following the end-effector. The goal is to get to a convenient and quick set up for the cell (few equipment, installation and constraints), in order to suit a production line context. Moreover, in order to master the digitizing quality and to optimize the operating speed we have to finely qualify the robot and measurement systems performances.

The posing quality of the end-effector, and thus the digitizing quality, rely on the desired trajectory given as an input to the robot control system. Therefore we worked on a path generation strategy taking into account the robot behavior and its performances based on a model and performance indexes. But mastering the digitizing quality also requires a perfect knowledge of the measurement systems (KZ25 and C-Track). For this reason we identified their performances to integrate them into our path generation strategy.

A geometric model is selected to describe the robot behavior. Then the model parameters are identified with a quick and convenient method we developed, based on the existing circle point analysis method (CPA). A comparative study shows the advantages of our identification method over classic methods. Yet the model needs to be completed with non-geometric parameters such as flexibilities or backlashes in order to meet the quality requirements of the application. Therefore we propose a selective method to pick the most relevant parameters for our model, based on their “identifiability”.

Robot performances in terms of speed and posing quality are then studied through performance indexes. Since existing performance indexes are made for robot-design, they cannot help us in describing robot performances through an entire path. For this reason we developed our own indexes that can be directly incorporated into our path generation strategy.

Using a robot as a displacement system for the laser plane sensor allows to orientate it continuously during the path, and to fit the shape of the part to digitize. But it makes the scanning process more complex and forces us to elaborate a calibration method for the sensor in order to be able to exploit this unrestricted orientation. The digitizing of a test part made it possible to validate this sensor calibration.

All the main elements constituting our path generation strategy were developed and the different calibrations needed were conducted by following the new methods we proposed. In our work prospects, the strategy will be operated on several complex and large parts, in order to validate the resulting digitizing quality and to quantify the time saved on the digitizing process.

Résumé

Les robots sériels de grandes dimensions ont été développés pour réaliser des opérations répétitives dans de larges domaines de travail et avec une grande dextérité, le tout en apportant de la flexibilité dans les chaînes de production. De nouvelles applications contraignantes telles que la mesure de pièces complexes sur chaîne de production peuvent exploiter ces avantages très attractifs. Il est cependant impératif de mieux maîtriser le positionnement de l'effecteur de ces robots, pour répondre aux exigences de la mesure 3D.

Dans ce contexte, une cellule de numérisation robotisée a été développée, exploitant un robot sériel 6 axes comme support d'un capteur laser plan (KZ25 Kreon), et utilisant un système de stéréovision externe pour le suivi de l'opération et la calibration de la cellule (C-Track Creaform). Le but est d'aboutir à une mise en place rapide et pratique (peu de contraintes et de matériel) de cette cellule, afin de répondre à un contexte de chaîne de production. Dans le même temps, nous cherchons à qualifier les capacités de tous les éléments de la cellule afin d'une part de maîtriser le positionnement du robot pour une numérisation à qualité donnée, et d'autre part d'optimiser la vitesse de l'opération.

La maîtrise du robot, et donc de la numérisation, passe par la trajectoire donnée en consigne au robot. C'est pourquoi nous proposons une stratégie de génération de trajectoire prenant en compte les capacités et le comportement du robot (avec des indices de performance et un modèle). Mais une numérisation à qualité donnée requiert également une parfaite connaissance des capacités des systèmes de mesure utilisés (KZ25 et C-Track). Un travail de qualification a donc été mené afin d'identifier ces capacités pour les intégrer à notre stratégie.

Un modèle géométrique est choisi pour représenter le comportement du robot. Puis, afin d'identifier les paramètres de ce modèle de manière rapide et pratique, nous proposons une nouvelle méthode d'identification généralisant le concept d'étude d'arcs de cercle proposé dans la méthode CPA (Circle Point Analysis). Une étude comparative démontre les avantages de cette nouvelle méthode par rapport aux méthodes classiquement utilisées. Les exigences de l'application de numérisation imposent de compléter le modèle géométrique avec des paramètres non géométriques (flexibilités, jeux). À cet effet, nous avons élaboré une méthode de sélection de paramètres pertinents basée sur leur « identifiabilité ». Les capacités du robot en vitesse et en qualité de positionnement sont ensuite étudiées au travers d'indices de performance. La plupart des indices utilisés en robotique servent à la conception de robots et ne répondent pas à notre besoin de caractériser une trajectoire entière. Nous avons donc proposé de nouveaux indices directement intégrables dans notre stratégie de génération de trajectoire.

L'utilisation d'un robot comme système de déplacement permet une réorientation continue du capteur laser plan au cours de la trajectoire, afin d'épouser les formes de la pièce à numériser. Mais l'acquisition du nuage de point est alors plus complexe et nous impose de développer une méthode de calibration de notre capteur adaptée à cette liberté d'orientation. La numérisation d'une pièce test a permis de valider cette calibration.

Les différents éléments clés constituant notre stratégie de génération de trajectoire ont été élaborés, et les différentes calibrations requises ont été effectuées à l'aide des nouvelles méthodes proposées. En perspective, notre stratégie sera employée sur plusieurs pièces complexes et encombrantes afin de valider la qualité de numérisation résultante et de quantifier le gain sur la durée de l'opération.

Titre : Modélisation et calibration pour une numérisation robotisée

Mots clés : robotique, numérisation, indices de performance, calibration, choix de modèle

Résumé : Les robots sériels de grandes dimensions apportent dextérité, répétabilité et flexibilité dans les chaînes de production. Sur ces chaînes, des opérations telles que la mesure de pièces peuvent exploiter ces avantages très attractifs. Il est cependant impératif de mieux maîtriser le positionnement de l'effecteur de ces robots, pour répondre aux exigences de la mesure 3D. Dans ce contexte, une cellule de numérisation robotisée a été développée, exploitant un robot sériel 6 axes comme support d'un capteur laser plan (KZ25 Kreon), et utilisant un système de stéréovision externe pour le suivi de l'opération et la calibration de la cellule (C-Track Creaform). La calibration que nous proposons pour cette cellule permet de maîtriser la qualité et d'optimiser la vitesse de numérisation, et se veut à la fois rapide et pratique (peu de contraintes et de matériel) pour répondre à un contexte industriel.

Cette calibration passe par l'identification des paramètres d'un modèle géométrique pour le robot, à l'aide d'une nouvelle méthode que nous proposons, généralisant le concept d'étude d'arc de cercle proposé dans la méthode CPA (Circle Point Analysis). Une étude comparative démontre les avantages de cette nouvelle méthode par rapport aux méthodes classiquement utilisées. Une méthode de sélection que nous avons développée permet ensuite de compléter le modèle du robot avec des paramètres non-géométriques pertinents (flexibilités, jeux). Au cours de la calibration, nous étudions également les capacités du robot en vitesse et en qualité de positionnement au travers d'indices de performance originaux. Enfin nous avons élaboré et validé une méthode de calibration en position et orientation du KZ25 sur son support (le robot) ce qui permet une numérisation à 6 DDL. En perspective, un générateur de trajectoires, à donner en consigne au robot, devra utiliser cette calibration de la cellule entière pour maîtriser la qualité de la numérisation et optimiser sa durée.

Title : Modeling and calibration for a robotized digitizing

Keywords : robotics, digitizing, performance indexes, calibration, model selection

Abstract : Serial robots are designed for repetitive tasks in wide workspaces, and provide good dexterity and flexibility to production lines. Restrictive applications such as parts digitizing can make use of those attractive benefits. However, digitizing needs on accuracy require to work carefully on the quality of the robot end-effector positioning. In this context, a digitizing cell using a 6 axis robot as displacement system for a laser plane sensor (KZ25) was developed. In this cell all calibrations are ensured by an external measurement system (C-Track Creaform) following the end-effector. The goal is to master the digitizing quality and to optimize the operating speed thanks to quick and convenient calibrations (few equipment, installation and constraints).

First, the parameters of the robot geometric model are identified with a quick and convenient method we developed, based on the existing circle point analysis method (CPA). A comparative study shows the advantages of our identification method over classic methods. Then a selective method we proposed allows us to complete this geometric model with relevant non-geometric parameters such as flexibilities or backlashes. Robot performances in terms of speed and posing quality are also studied through new performance indexes. Finally, we had to create and validate a calibration method for the position and orientation of the KZ25 sensor in order to exploit the unrestricted orientation provided by the robot end-effector. In our work prospects, a path generation strategy will use those calibrations to create paths for the robot, with mastered digitizing quality and optimized speed.

