

Thèse de Doctorat

Itzel GONZALEZ OJEDA

*Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université de Nantes
sous le sceau de l'Université Bretagne Loire*

École doctorale : Mathématiques et Sciences et Technologies de l'Information et de la
Communication (MathSTIC)

Discipline : Automatique, Productique et Robotique

Unité de recherche : Laboratoire des Sciences du Numériques de Nantes (LS2N)

Soutenue le 4 avril 2018

Intégration de modèles numériques réduits dans le pilotage robots possédant des flexibilités dans la dépose de fibres

JURY

Président du jury	Etienne DOMBRE, Dr Emérite, Université de Montpellier, LIRMM.
Rapporteurs :	Emmanuel DUC, Professeur des universités, SIGMA-Clermont Ferrand Saïd ZEGHLOUL, Professeur des universités, Université de Poitiers, PPRIME
Examineurs :	Etienne DOMBRE, Dr Emérite, Université de Montpellier, LIRMM Gabriel ABBA, Professeur des universités, Université de Lorraine, LCFC Cyril NOVALES, Maître de conférences, Université d'Orléans, PRISME Ricardo BUENO, Directeur l'aire de manufacture avancée, TECNALIA Research and Innovation
Directeur de Thèse :	Yannick AOUSTIN, Professeur des universités, Université de Nantes, LS2N
Co-encadrant de Thèse :	Olivier PATROUUX, Enseignant Chercheur, ESTIA, ESTIA Recherche, Bidart

Remerciements

Je voudrais remercier à CONACYT-Mexique (Conseil national de science et technologies du gouvernement mexicain) pour avoir fourni les moyens économiques qui m'ont permis de continuer mon développement personnel et professionnel.

Je tiens tout d'abord à exprimer ma profonde gratitude à mon directeur de thèse M. Yannick Aoustin, Professeur à l'université de Nantes et à mon co-encadrant M. Olivier Patroix, Enseignant-Chercheur à ESTIA, pour avoir accepté d'encadrer ce travail, pour l'implication totale et le soutien jusqu'à la fin. Sans leurs conseils, leurs nombreuses corrections, leur disponibilité, leur patience et leur confiance, l'évolution de ce travail aurait été très difficile. Merci de m'avoir appris le sens de la rigueur scientifique, mais aussi le côté pragmatique des choses. Merci pour les nombreuses relectures de ce mémoire et pour les réunions de préparation de la présentation finale qui ont été considérées pour la réussite de ce travail.

J'adresse toute ma reconnaissance aux membres du jury qui ont examiné cette thèse, merci pour leurs observations enrichissantes et leurs conseils. Je tiens à remercier chaleureusement M. Etienne Dombre, de m'avoir fait l'honneur de présider mon jury. Je remercie tout spécialement M. Emmanuel Duc et M. Saïd Zeghloul, pour avoir accepté d'examiner mon travail de thèse. Je remercie vivement M. Gabriel Abba, M. Cyril Novales et M. Ricardo Bueno, pour avoir accepté d'y participer.

Je souhaite remercier l'ensemble de mes collègues Sébastien Bottecchia, Joseph Canou, Pierre Joyot, Xavier Fischer, Stéphanie Cagin, Maylis Uhart, Denis Géral, Vincent Magimel, Thomas Dreno, Francis Sedeilhan, Simon Desseur, Geneviève Dirassar, Dimmitri Masson, Jérôme Faham, David Gomez, Regina Ticona, Denise Muñante... pour leur accueil, leur aide, leur soutien et leur discussions qui m'ont permis de passer des moments merveilleux et de grandir professionnellement, scientifiquement et personnellement. Une pensée particulière aux doctorants Michaël Hamwi, Gilberto Fontecha, et Asmmâ Agouzoul avec lesquels j'ai passé des moments aussi agréables.

Je tiens à adresser ma profonde gratitude à M. Jacques Bonnot et Mme Françoise Narbey pour leur temps passé à lire ce mémoire, pour leur aide et leurs corrections qui ont permis d'améliorer la rédaction de ce travail surtout à niveau de la langue françaises.

La science nous dit toujours que l'existence de Dieu n'est pas démontrée, par contre il existe des choses que la science ne peut pas expliquer. Je voudrais remercie Dieu pour me donner chaque jour l'opportunité d'avoir un nouveau départ, de réaliser chaque rêve, de prendre soin de moi et de me guider. Il n'y a pas une source plus grande de sagesse pour moi que celle fournie par Dieu.

Une pensée à ces partenaires inconditionnels de vie, que j'appelle amis. Merci pour les merveilleux moments partagés, pour leurs conseils, leurs expériences, leur conversation, leur compagnie, pour les souvenirs et les moments de folie ensemble, pour être toujours à mes côtés malgré la distance et pour faire de ma vie une aventure unique.

Une pensée particulière à Laurent Bonnot pour tous les moments spéciaux à tes côtés, pour ta compagnie, ta patience, ta compréhension, ton soutien, ton temps, ton amour et ta confiance.

Je tiens à adresser un remerciement spécial à ma famille éloignée par la distance durant ces années d'étude en France et qui m'a toujours soutenue. Je remercie infiniment mes parents et mon frère qui m'ont toujours fait confiance et m'ont laissé libre de mes choix. Je les remercie pour leur gratitude, leur amour, leur compréhension, leur sacrifice et pour être toujours à mes côtés malgré la distance. Je voudrais souligner que ma réussite est d'abord et avant tout leur réussite.

En fin je tiens à adresser un remerciement à toutes les personnes qui m'ont soutenue à continuer, à celles qui m'ont accompagnée et à celles qui ont participé directement et indirectement à ce travail.

Table des matières

Remerciements	I
Table des matières	III
Table des illustrations.....	VI
Table des tableaux	IX
1. Introduction générale.....	1
1.1 Contexte.....	1
1.2 Objectifs de la thèse.....	3
1.3 Problématique.....	3
1.4 Organisation du mémoire	4
2. Cadre de l'étude	6
2.1 Les solutions proposées aux problèmes de précision	6
2.1.1 Les solutions proposées par les industriels	7
2.1.2 Les solutions proposées par les centres de recherche	9
2.2 Fabrication des pièces composites.....	18
2.3 Cellule Robotisée à CompositAdour	21
2.3.1 Système de chauffe.....	21
2.3.2 Matériau	22
2.3.3 Commande de la tâche	22
2.4 Limites du système actuel	23
2.5 Conclusion.....	25
3. Flexibilités du système	28
3.1 Modélisation de robot.....	29
3.1.1 Modèles géométrique et cinématique du robot	29
3.1.1.1 Modèle géométrique	29
3.1.1.2 Modèle cinématique.....	33
3.1.2 Dynamique des robots	35
3.1.2.1 Robots rigides	35
3.1.2.2 Robots ayant des flexibilités	35
3.2 Flexibilités de l'outil.....	39
3.2.1 Outil rigide	40
3.2.2 Outil flexible	43
3.3 Modélisation des flexibilités de l'outil	49
3.3.1 Modèle expérimental.....	49

3.3.2	Modèle numérique par éléments finis	52
3.3.3	Recalage du modèle numérique	54
3.3.4	Modèle EF discrétisé	57
3.4	Conclusion	59
4.	Réduction de modèles	61
4.1	Méthodes de réduction de modèles	62
4.1.1	Méthodes de réduction de modèles <i>a priori</i>	63
4.1.2	Méthodes de réduction de modèles <i>a posteriori</i>	64
4.1.2.1	Méthode réseaux de neurones	65
4.1.2.2	Méthode de logique floue	67
4.2	Réduction du modèle EF par réseaux de neurones	72
4.3	Conclusion	76
5.	Génération de consigne des robots par capteurs externes	78
5.1	Types de commandes	79
5.2	Structure de la commande	82
5.3	Task Frame Formalism (TFF)	84
5.4	Loi de commande	86
5.4.1	Loi de commande en position des robots rigides	87
5.4.2	Loi de commande en force	88
5.4.3	Loi de commande hybride externe en position/force	88
5.4.4	Correcteur PID	89
5.4.5	Correcteur logique floue-PI	91
5.5	Sélection de la loi de commande	93
5.5.1	La loi de commande classique PID	93
5.5.2	La loi de commande en logique floue PI	96
5.5.3	Bilan	98
5.6	Conclusion	100
6.	Approche proposée	102
6.1	Commande en pression	103
6.2	Point dynamique de centrage de l'outil	104
6.3	Génération de trajectoire en position/force-pression en modifiant le TCP	105
6.4	Dispositif expérimental	106
6.4.1	Capteur d'effort	107
6.4.2	Robot	108
6.4.3	Système de communication robot capteurs externes (RSI)	109
6.4.4	Moule	110
6.4.5	Procédure de tests	112

6.5	Génération de trajectoire en pression avec prise en compte des flexibilités de l'outil	113
6.5.1	Commande hybride en position/force-pression.....	113
6.5.2	Point dynamique de centrage de l'outil.....	117
6.6	Conclusion.....	122
7.	Conclusion et perspectives.....	124
7.1	Conclusions.....	124
7.2	Perspectives.....	125
A.	Annexes.....	128
A.1	Modèle du robot.....	128
A.2	Modèle analytique par Hertz.....	129
A.2.1	Cas cylindre-plan.....	129
A.2.2	EF et contact de Hertz.....	133
A.3	Capteur d'effort.....	136
	Références.....	139

Table des illustrations

Figure 2.1 Boucle externe de commande	8
Figure 2.2 Diagramme de processus	8
Figure 2.3 Schéma de commande en position/force.....	11
Figure 2.4 Schéma de commande en position/vision	12
Figure 2.5 Schéma de commande mixte position/force/vision	12
Figure 2.6 Schéma d'asservissement méthode miroir	16
Figure 2.7 Schéma de commande prédictive de type GPC-correcteur polynomial RST à deux degrés de liberté	17
Figure 2.8 Schéma de commande H_{∞} à deux degrés de liberté avec anticipation.....	17
Figure 2.9 Schéma de l'algorithme de détection de collisions.....	18
Figure 2.10 Tissu en carbone enrobé de résine	18
Figure 2.11 Mèche en carbone enrobée de résine	18
Figure 2.12 Moule ouvert pour dépose des fibres	19
Figure 2.13 Robot utilisée dans l'ATL (Sloan, 2008)	19
Figure 2.14 Contours complexes en composites (Electroimpact, 2016).....	20
Figure 2.15 Surfaces complexes en composites (MetalWorking, 2015).....	20
Figure 2.16 Excédent des fibres pendant la dépose de fibres par les méthodes ATL et AFP. 20	
Figure 2.17 Robot utilisé dans l'AFP(Coriolis-Composites, 2016).....	21
Figure 2.18 Cellule robotisée à CompositAdour.....	21
Figure 2.19 L'orientation et la position du TCP dans le plan XZ.....	23
Figure 2.20 L'orientation et la position du TCP dans le plan YZ.....	23
Figure 2.21 L'orientation et la position du TCP dans le plan isométrique	23
Figure 2.22 Défauts de la dépose de fibres : Overlap et Gap.....	23
Figure 2.23 Rouleau de compactage	25
Figure 3.1 Sens de rotation des articulations KR 240 (KUKA, 2004) et KR6 (KUKA, 2003)31	
Figure 3.2 Architecture géométrique	31
Figure 3.3 Modélisation des flexibilités dans deux corps	36
Figure 3.4 Modélisation de la flexibilité de l'articulation	37
Figure 3.5 TCP fixe à un point.....	40
Figure 3.6 Robot KUKA KR6 avec les cordones World et le TCP	41
Figure 3.7 L'outil pour soudage attaché au poignet du robot	41
Figure 3.8 Méthode XYZ 4- points (KUKA 2012).....	42
Figure 3.9 Composants des éléments finis	44
Figure 3.10 Système ressort	45
Figure 3.11 Rouleau de compactage a) sans aucune force b) avec la force exercée par une main.....	49
Figure 3.12 Système employé pour les tests de compression	51
Figure 3.13 Rouleau soumis à une force de 1000 N	51
Figure 3.14 Résultats obtenus pour les valeurs d'effort-déformation en compression le long d'un axe grâce à la machine à compression	52
Figure 3.15 Maillage du modèle EF pour rouleau de compactage.....	53
Figure 3.16 Résultats du modèle hyper-élastique avec un module d'élasticité de 1,44 MPa... 54	
Figure 3.17 Recalage du modèle EF pour les modèles hyper-élastiques. a) Module d'élasticité égal à 1,44 MPa b) Module d'élasticité égal à 2,84 MPa c) Module d'élasticité égal à 0,72 MPa d) Module d'élasticité égal à 1,26 MPa	56

Figure 3.18 Recalage du modèle hyper-élastique Ogden (pour un module d'élasticité égale à 1,26 MPa).....	57
Figure 3.19 Résultats obtenus par le modèle EF : a) pour la pression et b) pour la déformation	58
Figure 4.1 Diagramme du réseau de neurones artificiel.....	66
Figure 4.2 Diagramme de chaque neurone.....	67
Figure 4.3 Représentation des variables linguistiques de la logique floue	69
Figure 4.4 Représentation graphique de l'opérateur logique floue « union »	69
Figure 4.5 Représentation graphique de l'opérateur logique floue « intersection ».....	70
Figure 4.6 Représentation graphique de l'opérateur logique floue « non ».....	70
Figure 4.7 Représentation de la FAM	71
Figure 4.8 Diagramme de réseaux de neurones pour le modèle réduit	74
Figure 4.9 Résultats du modèle réduit par réseau de neurones a) modèle de pression b) modèle de déformation.....	75
Figure 5.1 Commande de boucle fermée du robot	79
Figure 5.2 Commande hybride en force/position.....	82
Figure 5.3 Commande hybride externe en force/position	82
Figure 5.4 Définition du repère tâche.....	85
Figure 5.5 Transformation des forces et couples dans le repère tâche.....	85
Figure 5.6 L'implémentation d'une loi de commande de la logique floue avec des correcteurs PI	92
Figure 5.7 Résultats de la loi de commande classique PID calculés pour une surface plane avec un capteur à une dimension.....	94
Figure 5.8 Résultats de la loi de commande classique PID calculés pour un contact au point B sur une surface plane avec un capteur à six dimensions	95
Figure 5.9 Résultats de la loi de commande classique PID calculés pour un contact au point B sur une surface convexe avec un capteur à six dimensions.....	96
Figure 5.10 Représentation graphique des règles d'apprentissage et la mémoire associative floue (FAM)	96
Figure 5.11 Résultats la loi de commande en logique floue PI calculés pour une surface plane avec un capteur à une dimension.....	97
Figure 5.12 Résultats de la loi de commande en logique floue PI pour un contact au point B sur une surface plane	97
Figure 5.13 Résultats de la loi de commande en logique floue PI pour un contact au point B sur une surface convexe	98
Figure 5.14 Comparaison des résultats de la loi de commande pour une surface plane	99
Figure 5.15 Comparaison des résultats de la loi de commande pour une surface plane	99
Figure 5.16 Comparaison des résultats de la loi de commande pour une surface convexe	99
Figure 6.1 Commande hybride externe en position/force-pression	104
Figure 6.2 Commande hybride externe TCP.....	105
Figure 6.3 Commande en position/force-pression en modifiant le TCP.....	106
Figure 6.4 Dispositif expérimental.....	106
Figure 6.5 Dimensions d'un robot.....	108
Figure 6.6 Système de communication	110
Figure 6.7 Moule plan avec des variations de dimension dans la largeur.....	111
Figure 6.8 Moule convexe.....	111
Figure 6.9 Moule triangle.....	111
Figure 6.10 Procédure de test pour le moule du type plan	112
Figure 6.11 Procédure de test pour le moule du type convexe	112
Figure 6.12 Procédure de test pour le moule du type triangle.....	112

Figure 6.13 Zones du moule du type plan	114
Figure 6.14 Résultats pour la commande en position, la commande en position/force et la commande en position/force-pression.....	116
Figure 6.15 a)TCP fixe b) Rotation de l'outil autour de l'axe Y	117
Figure 6.16 TCP a) outil déformé de 0,5 mm avec un TCP fixe (cas industriel) b) outil déformé avec un TCP dynamique.	118
Figure 6.17 Résultats de la rotation de l'outil de cinq degrés autour de l'axe Y	119
Figure 6.18 Résultats pour le TCP et le DTCP sur un moule du type convexe	120
Figure 6.19 Résultats pour le TCP et le DTCP sur un moule du type coin.....	121
Figure A.1 Contact sans aucune force.....	129
Figure A.2 Représentation de la déformation de deux sphères en contact avec une force normale.....	130
Figure A.3 Déformation sur un plan	130
Figure A.4 Distribution des pressions	131
Figure A.5 Contrainte de cisaillement xz.....	132
Figure A.6 Diagramme du modèle en contact.....	135
Figure A.7 Maillage pour le modèle FE.....	135
Figure A.8 Résultats obtenus par les méthodes de Hertz et d'EF	136
Figure A.9 Capteurs à jauges de contrainte a) Capteur pour mesurer la flexion b) Capteur en « S » c) Capteur du type Pan-cake	137
Figure A.10 Capteur piézoélectrique a) Capteur à un composant b) capteur à trois composants	138

Table des tableaux

Tableau 3.1 Paramètres de D-H modifiés	31
Tableau 3.2 Types d'équations rencontrés avec la méthode de Paul.....	33
Tableau 3.3 Formulation mathématique des modèles hyper-élastiques (ANSYS).....	47
Tableau 4.1 Fonction caractéristique et fonction d'appartenance	68
Tableau 5.1 Représentation des règles d'une loi de commande logique floue-PI.....	92
Tableau 5.2 Tables de vérité du contrôleur en logique floue PI.....	96
Tableau 6.1 Paramètres dimensionnels des robots.....	108
Tableau A.1 Paramètres de D-H modifiés	128

“Impossible is just a big word thrown around by small men who find it easier to live in the world they've been given than to explore the power they have to change it. Impossible is not a fact. It's an opinion. Impossible is not a declaration. It's a dare. Impossible is potential. Impossible is temporary. Impossible is nothing.”

Muhammad Ali

CHAPITRE

1

Introduction générale

Sommaire

1.1 Contexte	1
1.2 Objectifs de la thèse	3
1.3 Problématique.....	3
1.4 Organisation du mémoire	4

1.1 Contexte

L'accroissement de la productivité et la demande d'amélioration de la qualité dans la fabrication des pièces ont conduit au développement de machines spécialisées. Ces machines sont connues pour leur très bonne performance pour la fabrication de pièces en grande série, car elles sont dédiées à une tâche spécifique. Par contre elles ont un coût élevé et leur retour sur investissement se réalise généralement sur le long terme (Fu et al., 1988; Gallet-Hamlyn, 2011). Cependant, la production en moyenne et/ou petite série demande des machines adaptables à de multiples tâches avec un coût moindre (Lucas, 2008). Grâce à la grande polyvalence des systèmes de production robotisés et à leur coût réduit par rapport à ceux des machines spécifiques (Fu et al., 1988; Craig, 2006; Ollero Baturone, 2006), ces systèmes sont de plus en plus présents dans presque toutes les industries (Barrientos & Balaguer, 2007; Scherer, 2008) comme par exemple : les industries automobile, chimique, plastique et alimentaire. En revanche, les robots restent encore très absents de l'industrie aéronautique et aérospatiale (DeVlieg & Szallay, 2009, 2010), car les normes de qualité et de précision y sont élevées.

Selon la Fédération Internationale de Robotique (IFR en anglais), à la fin de l'année 2013, de 1 332 000 à 1 600 000 robots industriels ont été mis en service (IFR, 2014). Les robots

industriels sont créés pour faciliter les procédés répétitifs. Ils ont donc remplacé l'homme dans ces tâches permettant aux industriels d'utiliser la main-d'œuvre qualifiée pour des tâches à forte valeur ajoutée (Barrientos & Balaguer, 2007). De façon générale, les robots industriels sont utilisés pour des opérations industrielles comme : l'assemblage, le soudage, la manipulation (pick and place)... Ces opérations possèdent une bonne tolérance à l'erreur, car elles peuvent supporter une déviation de quelques millimètres à quelques centimètres sans que le procédé n'en soit affecté. Tandis que l'incursion des robots dans des opérations telles que la dépose de fibres, la rectification, le ponçage, le polissage, le détournage,... est de plus en plus fréquente, les robots ne peuvent pas atteindre une bonne précision comparée à celle du drapage manuel ou d'une machine-outil (Lucas, 2008; Gallot et al., 2012).

Malgré cette faible précision, le drapage robotisé des fibres est aujourd'hui la technique la plus utilisée pour réduire les coûts (Lecompte-Boinet, 2006), car elle permet de produire des pièces complexes et de grandes dimensions (Croft et al., 2011; Gallet-Hamlyn, 2011) en réduisant le temps de fabrication et la quantité de matériau utilisé (Lecompte-Boinet, 2006). Cette technique de dépose de fibres requiert un effort de compactage qui doit être adapté au matériau utilisé. En conséquence l'outil du robot doit rester en contact avec la surface du moule et contrôler l'effort de compactage à appliquer. Les robots manipulateurs ne sont pas conçus pour travailler en contact et/ou en effort. Ainsi, des problèmes de précision et de suivi de trajectoires apparaissent. Le problème intrinsèque de précision dans les procédés robotisés de dépose de fibres peut être corrigé soit par la sélection d'un robot dédié à cette tâche qui offre une meilleure précision, soit par l'analyse et l'adaptation des flexibilités du système. Pour le premier cas, le robot possède un calibrage en usine qui permet d'affiner ses paramètres dimensionnels. Par contre ce calibrage augmente le prix et la précision n'est pas garantie, car les fabricants de robots définissent la répétabilité (capacité à reproduire une action) et non la précision (KUKA, 2013; ABB, 2015). Pour le deuxième cas, le robot est soumis à des efforts pour connaître sa flexibilité (ou sa rigidité) afin d'adapter une correction (Makarov, 2013) et/ou pour utiliser des systèmes externes qui garantissent sa précision et sa rigidité (Dumas, 2011). Le manque de rigidité entraîne, du fait des efforts liés au procédé qui peuvent être importants, des imprécisions dues au positionnement de l'outil (Dumas, 2011) et des défauts géométriques du robot (Gallot et al., 2012).

La plupart des méthodes existantes pour améliorer les performances des robots par l'analyse de flexibilités du système considèrent uniquement les flexibilités dans le robot, mais la tâche de dépose de fibres nécessite aussi la prise en compte des flexibilités de l'effecteur, comme dans les travaux de Dumas et de Makarov (Dumas, 2011; Makarov, 2013).

Afin d'améliorer la précision dans le procédé robotisé de dépose de fibres, cette recherche doit considérer l'effort de compactage et le contact entre la surface et le robot qui entraînent des flexibilités dans l'effecteur et dans le robot.

1.2 Objectifs de la thèse

Le principal objectif de ce travail est de proposer des solutions pour améliorer la précision actuelle des robots dans le procédé de dépose de fibres sans arrêter les processus de fabrication.

D'un point de vue industriel, l'objectif peut être accompli par :

- L'identification de la source des erreurs qui génèrent l'imprécision.
- La modélisation et l'analyse du système.

D'un point de vue recherche, l'objectif peut être atteint par :

- La modélisation des flexibilités des composants (l'effecteur, la surface de dépose de fibres et le robot) en temps réel et/ou hors ligne.
- Le calcul en temps réel des écarts entre la géométrie réelle de la pièce (modèle réel) et le modèle CAO.
- Le calibrage du point de centrage de l'outil (TCP) en temps réel en considérant les déformations de la géométrie de l'effecteur lorsqu'il est soumis aux efforts.
- La correction de la trajectoire du robot en temps réel et/ou hors ligne en considérant les efforts appliqués sur la pièce fabriquée, sur l'outil et/ou sur le robot.

L'objectif visé pour l'application industrielle est l'amélioration de la précision dans le respect des spécifications.

1.3 Problématique

Afin d'augmenter l'utilisation des robots en opérations industrielles qui impliquent une haute précision et une haute qualité comme le découpage, le détourage, la rectification, le ponçage, le polissage, la dépose de fibres, entre autres, l'amélioration de la précision et la réduction des défauts géométriques (Gallot et al., 2012) représentent un enjeu important face aux exigences de qualité des pièces souhaitées (Uhart et al., 2013).

Ces opérations doivent être étudiées de manière détaillée, pour identifier toutes les erreurs qui génèrent l'imprécision en vue d'améliorer la tâche. Deux caractéristiques très importantes sont présentées dans ces opérations : d'un côté, le contact constant entre l'outil et la pièce et d'un autre côté, la force exercée pour accomplir la tâche. Ces caractéristiques entraînent des efforts externes sur la pièce et sur le robot qui produisent des modifications importantes dans leurs géométries et des variations dans la trajectoire du robot. En conséquence, la commande du robot

doit identifier et prendre en compte les modifications dans l'environnement, et corriger la trajectoire en temps réel ou hors ligne pour résoudre les problèmes de précision. La tâche de dépose de fibres doit donc contrôler la force de compactage et garantir l'interaction entre l'outil et l'environnement (moule de dépose de fibres).

Le processus de dépose de fibres robotisé est fortement implémenté dans de multiples entreprises aéronautiques. Ainsi, le procédé peut être partiellement modifié sans affecter le système déjà installé, en équipant ce système avec des capteurs extéroceptifs qui fournissent des informations sur l'interaction entre l'effecteur et l'environnement et/ou sur le positionnement de l'effecteur dans l'environnement.

1.4 Organisation du mémoire

Le chapitre 2 aborde le cadre d'étude. Il expose les solutions proposées par les entreprises et les centres de recherche pour diminuer les imprécisions du robot dans des tâches d'usinage et de déplacement des fibres. Ensuite, les différentes méthodes pour la fabrication des pièces composites sont précisées. Finalement, les divers composants qui intègrent une cellule robotisée de dépose de fibres et les limitations de celle-ci sont détaillés.

Le chapitre 3 décrit les flexibilités du système. Les modélisations du robot et de l'outil sont étudiées pour définir les facteurs qui génèrent des imprécisions dans les tâches robotisées. Ensuite, les modèles du robot en prenant en compte les flexibilités des corps ou des articulations sont décrits. Finalement l'importance de la définition du point de centrage de l'outil est exposée. Le chapitre 4 introduit la réduction de modèle. Il présente les différentes méthodes pour la réduction de modèles. Ces méthodes permettent d'appliquer des modèles numériques d'un coût élevé de calcul comme les éléments finis sur des systèmes en temps réel.

Le chapitre 5 traite de la génération de consignes de robots intégrant des capteurs externes. Les différents types de commande pour corriger la trajectoire du robot en temps réel utilisant des capteurs extéroceptifs sont examinés. Ensuite, la méthode pour réaliser la translation des efforts du repère de la tâche au repère du capteur est présentée, ainsi que le calcul des correcteurs pour garantir la stabilité et le bon fonctionnement du système.

Le chapitre 6 énonce les approches proposées. Les solutions pour résoudre les problèmes d'imprécision du robot dans le procédé de dépose de fibres en maintenant un compactage homogène des fibres le long de la pièce indépendamment de la surface sont analysées, ainsi que les solutions pour redéfinir en temps réel le point de centrage dynamique de l'outil (DTCP) en prenant en compte les flexibilités de l'outil.

Le chapitre 7 précise l'expérimentation et les résultats obtenus dans ce travail. Le dispositif expérimental, les méthodes et les composants utilisés sont exposés. Les résultats de l'expérience sont expliqués et analysés en ce qui concerne la modélisation des flexibilités de l'outil, les lois de commandes, l'asservissement en position-force/pression et la redéfinition du point du TCP. Le chapitre 8 présente les conclusions générales et les perspectives de cette recherche.

CHAPITRE

2

Cadre de l'étude

Sommaire

2.1 Les solutions proposées aux problèmes de précision.....	6
2.1.1 Les solutions proposées par les industriels.....	7
2.1.2 Les solutions proposées par les centres de recherche.....	9
2.2 Fabrication des pièces composites	18
2.3 Cellule Robotisée à CompositAdour.....	21
2.3.1 Système de chauffe.....	21
2.3.2 Matériau.....	22
2.3.3 Commande de la tâche.....	22
2.4 Limites du système actuel	23
2.5 Conclusion.....	25

Ce chapitre introduit les solutions proposées par les entreprises et laboratoires de recherche pour pallier les problèmes de précision liés à la non-rigidité d'un robot. Ensuite, nous exposerons de façon générale les types de processus et les composants d'une cellule robotisée pour la fabrication de pièces en composites. Finalement, nous préciserons le type de commande et les restrictions actuelles de la tâche de dépose de fibres.

2.1 Les solutions proposées aux problèmes de précision

Dans ce paragraphe, sont présentés les travaux développés jusqu'à présent pour améliorer la précision du robot. Nous exposons également comment sont pris en compte les efforts externes et internes dans les composants de la cellule robotique (robot, surface et effecteur). Ces efforts

sont à l'origine des déformations de la structure mécanique du robot et de l'environnement en contact avec l'organe effecteur. Ces déformations provoquent des problèmes de précision.

2.1.1 Les solutions proposées par les industriels

Dans les travaux de DeVlieg (DeVlieg et al., 2002; DeVlieg, 2009; DeVlieg & Szallay, 2009; DeVlieg, 2010; DeVlieg & Szallay, 2010; DeVlieg, 2011), Saund (Saund & DeVlieg, 2013), Rathjen (Rathjen & Richardson, 2013) et Jeffries (Jeffries, 2013), les auteurs mettent en œuvre des robots pour des opérations comme le fraisage, l'assemblage et le placement de fibres. Les auteurs considèrent que les flexibilités du robot et de la surface sont dues à des efforts externes. Les robots employés dans ces recherches sont des robots industriels à six degrés de liberté (DDL) de la compagnie KUKA. Dans leurs travaux (DeVlieg, 2010; DeVlieg & Szallay, 2010; DeVlieg, 2011; Jeffries, 2013), les auteurs utilisent un contrôleur externe, afin de commander le robot et l'effecteur comme une machine-outil à commande numérique (CN). Les flexibilités des robots industriels sont prises en compte avec les informations mesurées par des capteurs proprioceptifs comme les codeurs incrémentaux et des capteurs extéroceptifs comme le télémètre laser. L'information obtenue par les capteurs de position est envoyée au modèle cinématique, afin de réaliser la correction de la position de l'outil. Dans leurs travaux (DeVlieg et al., 2002; DeVlieg, 2009), les auteurs utilisent aussi le principe de commande numérique, mais ils commandent uniquement l'outil et l'information de la correction est envoyée au contrôleur du robot via une boucle externe (figure 2.1). Le travail de (DeVlieg et al., 2002) utilise un outil logiciel spécialisé pour améliorer la précision, par contre l'auteur mesure la répétabilité. Dans leurs travaux (DeVlieg et al., 2002; DeVlieg, 2009), les auteurs proposent l'utilisation d'une caméra à l'extrémité du robot pour inspecter les pièces terminées et aussi pour définir et corriger l'écart entre les positions désirées et réelles. Ils ont installé des lampes LED pour résoudre les problèmes d'éclairage. Le point de référence de la caméra est aligné sur la ligne centrale de la broche. L'information de la correction de la position obtenue à partir de la caméra est envoyée au contrôleur (Contrôleur Numérique (CN)) et celui-ci la transmet au robot pour exécuter l'autocorrection de la trajectoire. L'auteur propose dans son travail (DeVlieg, 2009) que le traitement des images et l'envoi de l'information de la position des pièces se fassent grâce à un outil logiciel dédié (figure 2.2). Par contre dans leur travail (DeVlieg & Szallay, 2010), les auteurs proposent l'utilisation d'un capteur laser tracker qui a le même objectif que la caméra. Dans son travail (DeVlieg, 2011), l'auteur a réalisé la programmation et la simulation de la trajectoire du robot hors ligne avec le logiciel de CAO CATIA V5. Dans son travail (DeVlieg & Szallay, 2010) la programmation et la simulation sont réalisées dans

l'environnement CENIT North America qui est un composant externe du logiciel CATIA V5. Finalement dans le travail de (Jeffries, 2013) la programmation et la simulation sont réalisées avec le logiciel CGTech VERICUT Composite Programming (VCP) et VERICUT Composite Simulation (VCS). De cette programmation et cette simulation hors ligne, l'auteur extrait un code qui peut être utilisé seulement dans un système à commande numérique (CN). Les simulations permettent de connaître les possibles collisions entre le robot et l'environnement, les éventuelles singularités du robot pendant son déplacement et la meilleure trajectoire pour réaliser la tâche (Devlieg & Szallay, 2010; Devlieg, 2011).

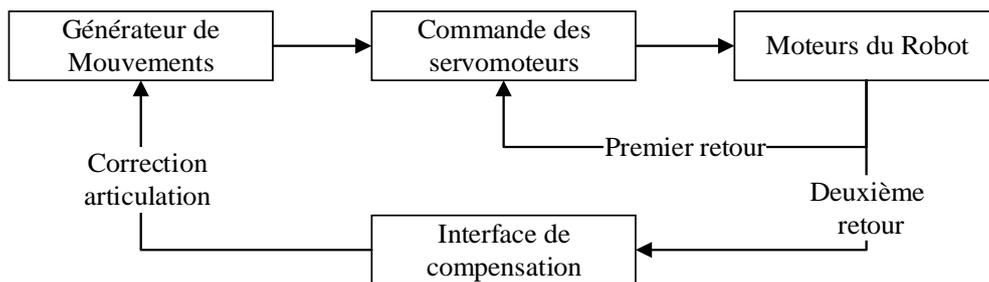


Figure 2.1 Boucle externe de commande

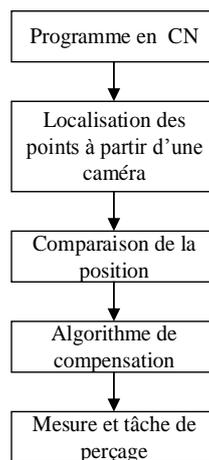


Figure 2.2 Diagramme de processus

Dans le travail présenté par Atkinson (Atkinson et al., 2007), l'auteur intègre des robots industriels considérés comme rigides pour la fabrication des ailerons d'avion. Ces robots réalisent les processus d'usinage et de perçage sur des matériaux comme le plastique renforcé en fibre de carbone (CFRP), l'acier, l'aluminium et le titane. Le robot utilisé est le KUKA 360-2 à six DDL monté sur un rail. Tous les composants du robot sauf la base de montage et de la bride de fixation de l'outil, sont fabriqués en aluminium, ce qui leur donne une bonne rigidité et un faible poids. L'effecteur est composé d'une base en aluminium et monté avec une orientation alignée sur celle des trois derniers axes du robot. Ce montage maximise la rigidité et minimise les mouvements parasites du robot. L'effecteur est un assemblage de trois

éléments : un système de pression, un axe de pression (servo-axe) et un outil de perçage. Le système de pression réagit aux forces de contact entre la surface et l'effecteur. Ce système est contrôlé par un capteur d'effort qui mesure les forces de contact. Ces mesures sont envoyées au système de commande de l'outil, afin d'être transformées en écarts de position/orientation indépendants de l'orientation du robot. Ces écarts sont ajoutés à la position actuelle du robot pour corriger la trajectoire. D'un autre côté, le servo-axe permet de définir une pression constante dans tout l'effecteur, afin de le stabiliser et de réduire les mouvements indésirables du robot pendant le perçage. Il est aussi contrôlé par un retour d'effort. Finalement un axe externe est ajouté pour concentrer les efforts sur l'effecteur et ainsi réduire les déformations sur les corps du robot. La programmation du robot est développée avec le logiciel Robotic Assembly Cell Controller (RACC) qui sert d'interface entre le robot et l'effecteur final, puisque le logiciel utilisé pour déplacer le robot est l'environnement de pilotage classique de chez KUKA (KRL). Néanmoins, l'effecteur est géré par un logiciel à commande numérique. L'auteur considère les facteurs suivants, pour déterminer la précision du système : le modèle cinématique imparfait, l'imprécision dans le calcul du TCP, le calibrage du robot et de l'effecteur, les flexibilités liées à la gravité et les flexibilités engendrées par les efforts statiques et dynamiques. Afin de corriger la trajectoire de l'outil, l'auteur propose l'intégration d'algorithmes qui compensent les erreurs liées aux facteurs décrits antérieurement. Les erreurs associées à la position du centre de gravité sont corrigées directement par le contrôleur du robot. Les flexibilités engendrées par les efforts statiques et dynamiques sont divisées en deux catégories : les erreurs liées aux efforts du contact et les erreurs associées aux flexibilités du robot. Les premières sont réduites par le système de pression. Pour réduire les deuxièmes, l'auteur propose l'implémentation d'un algorithme anti-décalage qui prédit et compense la déformation de l'outil et les efforts sur et/ou dans les axes du robot. Ces flexibilités sont prises en compte à partir de la mesure de la position par des codeurs incrémentaux et par un télémètre laser, comme dans le travail de DeVlieg (DeVlieg & Szallay, 2009). Finalement, l'auteur implémente le logiciel « KUKA Absolute Accuracy Package » (KAAP) qui corrige le modèle cinématique et le calibrage du TCP et du robot.

2.1.2 Les solutions proposées par les centres de recherche

Dans les travaux présentés par Uhart (Uhart et al., 2013; Uhart, 2014; Uhart et al., 2014), l'auteur propose une solution pour l'amélioration de la précision dans une tâche de dépose de fibres. Il utilise un robot KUKA KRC 240 de six DDL et une baie de contrôle KRC2 avec une boucle de contrôle fermée. Cette baie n'est pas accessible pour effectuer des modifications par

sécurité et pour la garantie du robot. L'auteur propose donc d'implémenter un générateur de consigne hybride externe qui a pour but de réduire l'erreur dans la commande. Ce générateur de consigne utilise des capteurs externes au robot pour obtenir des informations de leur environnement. Ceci est connecté à la baie de commande du robot via le module spécial RSI (Robot Sensor Interface). Le module RSI permet de communiquer des informations entre les systèmes externes et le robot, et de modifier certaines variables du robot sans modifier son contrôleur interne. Les capteurs extéroceptifs sont connectés à un ordinateur (PC-Capteur) qui est relié à un serveur par le protocole TCP/IP. La baie de commande du robot est également reliée au serveur par le même protocole. Le serveur transmet l'information de la baie de commande vers l'ordinateur PC-Capteur et vice et versa. Les capteurs extéroceptifs mesurent la position et la force en temps réel, cette information est envoyée au PC-Capteur qui intègre une interface développée spécialement pour la tâche de dépose de fibres. Cette interface permet de traiter l'information obtenue par les capteurs extéroceptifs. L'algorithme de calcul de l'interface permet de comparer la position souhaitée et la position réelle et puis de calculer l'écart entre les deux. Ces écarts représentent la correction de la position par rapport à l'outil. Ils doivent donc être ajoutés à la position courante du robot pour compenser la trajectoire et réduire l'erreur.

L'auteur propose trois approches pour la génération des consignes. Dans la première, il souhaite corriger une trajectoire commandée en force (figure 2.3). Il utilise alors un capteur d'effort du type ATI-gamma qui mesure la force exercée sur la surface (F_{mes}). L'algorithme de calcul de l'interface compare la force mesurée (F_{mes}) et la force désirée (F_{des}), ensuite, il calcule l'écart entre les deux forces (Δ_F). Afin que le robot puisse se déplacer selon la trajectoire commandée en force, l'écart de force (Δ_F) doit être transformé en un écart de position (Δ_{xF}). En même temps, l'algorithme de calcul de l'interface calcule l'écart entre les positions et orientations courantes (X_{mes}) et les positions et orientations souhaitées (X_{des}). Une matrice de sélection (S) permet de commander certaines directions en forces (S_f) et d'autres en positions (S_p). Les écarts en position (Δ_{xP}) et en force (Δ_{xF}) sont envoyés en un seul vecteur d'information représenté par (Δ_{xFP}). Ces écarts de position et d'orientation sont envoyés au robot en coordonnées cartésiennes par le module RSI et puis sont ajoutés à la position courante (X) du robot. Afin d'obtenir une force de compactage constante, l'auteur corrige la position courante de l'outil selon l'axe Z avec un écart de position en force (Δ_{xF}). Cette correction de l'outil permet également que les forces et les couples non désirés sur les directions commandées en position soient inférieurs à ceux qui sont mesurés sans la correction de l'outil (force de compactage

variable). Avec cette correction, les auteurs ont réduit les perturbations provenant du contact entre le robot et la pièce, pour le cas spécifique de la dépose de fibres.

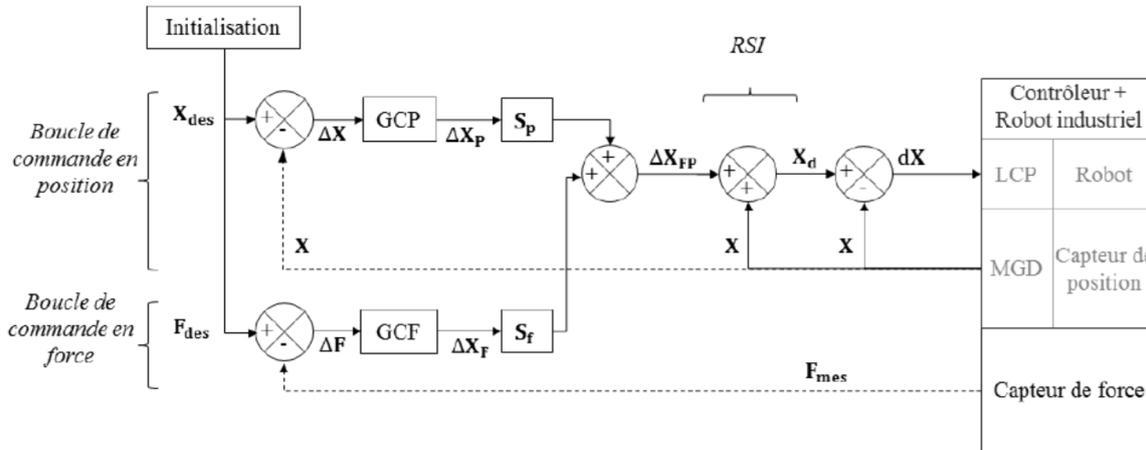


Figure 2.3 Schéma de commande en position/force

La deuxième approche que l'auteur est mise en œuvre est la correction de la commande avec un système de vision (figure 2.4). Ce système est composé d'une caméra Firefly MV FMVU-13S2C de la société Point Grey Research et d'un laser Lairis StockerYale Canada. Les deux éléments sont montés sur l'effecteur. Le système de vision consiste à projeter une ligne sur les fibres (surface) avec le laser, afin de rendre visible le bord et l'épaisseur des fibres qui sont identifiées en temps réel par la caméra. L'information obtenue par la caméra est traitée pour déterminer la position réelle de l'outil par rapport au laser et aux fibres. Une fois, la position réelle (I_{mes}) détectée, l'algorithme de calcul de l'interface la compare avec la position désirée (I_{des}) et calcul leur l'écart (Δ_v). Cet écart entre les positions est mesuré en pixels, alors cette erreur doit être transformée en un écart de position en millimètres (Δ_{xv}). Comme dans le cas du capteur d'effort, l'algorithme de calcul de l'interface calcule en même temps, l'écart (Δ_x) entre les positions et orientations courantes (X_{mes}) et les positions et orientations souhaitées (X_{des}). Pour éviter la confusion entre l'erreur de position en vision (Δ_{xv}) et l'erreur de position (Δ_x), les auteurs définissent l'erreur de position comme (Δ_{xp}). L'auteur utilise une matrice de sélection (S) pour commander certaines directions en vision (S_v) et les autres en position (S_p). Par contre les écarts en position (Δ_{xp}) et en vision (Δ_{xv}) sont envoyés dans une seule chaîne d'information représentée par (Δ_{xvp}). Ces écarts de position et d'orientation sont ajoutés à la position courante (X) du robot et sont envoyés au robot en coordonnées cartésiennes par le module RSI. Afin de maintenir le bord de la fibre détecté et aligné sur la cible, l'auteur corrige la position courante de l'outil le long de l'axe Y en corrigeant les écarts de position en vision (Δ_{xv}).

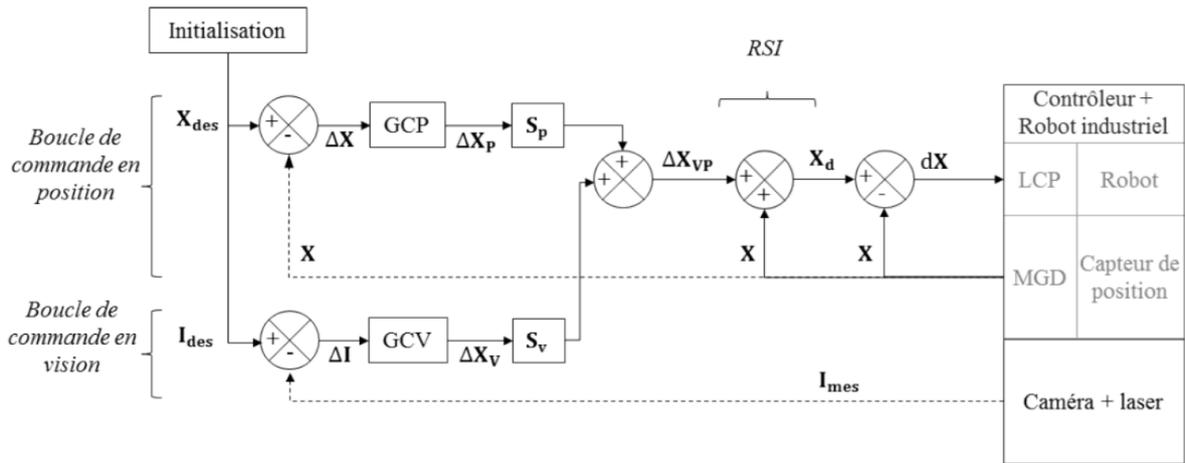


Figure 2.4 Schéma de commande en position/vision

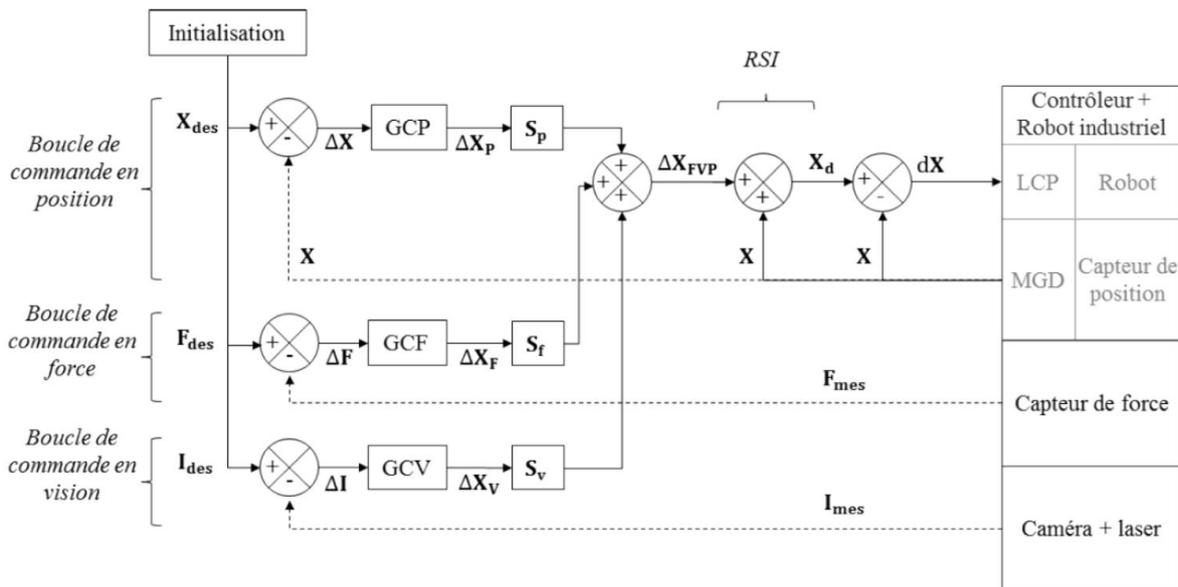


Figure 2.5 Schéma de commande mixte position/force/vision

Pour la troisième approche, l'auteur unifie les deux approches décrites antérieurement, c'est-à-dire un système force-vision (figure 2.5). L'axe Y est commandé par le système de vision et l'axe Z est commandé par le système en force. Pour mettre en œuvre, la correction du robot en force-vision, l'effecteur doit être en contact avec la surface pour déposer les fibres. Lorsque le robot ne se trouve pas en contact, toutes les directions du robot sont commandées en position. La position initiale de la tâche doit être en contact sans exercer aucune force. Lorsque le robot se trouve en position initiale de la tâche, l'effecteur est commandé en force le long de l'axe Z, pour obtenir l'effort de compactage désiré et la répartition homogène de l'effort. En même temps, la direction dans l'axe Y est commandée en vision. Cette configuration permet de garder une position latérale alignée sur la bande précédente. Les autres directions sont commandées en position. Pour déposer une bande tout en appliquant l'effort de compactage désiré avec une

distribution de l'effort et une position latérale correcte, l'effecteur se déplace le long de l'axe X. Lorsque la tâche de dépose est finalisée, toutes les directions sont commandées en position et l'effecteur est positionné en un point à l'extérieur de la surface. Dans les trois approches, l'algorithme de calcul de l'interface détermine et envoie les corrections de la trajectoire par rapport au repère de l'outil avec un période d'échantillonnage de 12 ms.

Dans les travaux présentés par (Dumas; et al., 2010; Dumas, 2011; Dumas et al., 2011), l'auteur propose d'identifier les paramètres du robot soit les forces externes, les forces internes et les vibrations pour améliorer la précision. Le système utilisé dans ces travaux est composé d'un robot KUKA KR270, d'un rétro-rélecteur, d'une masse reliée à l'extrémité de l'effecteur par un bras de levier et d'un dynamomètre. Afin d'obtenir les paramètres de flexibilité du robot, l'auteur propose deux approches.

Dans la première approche, l'auteur utilise un système de photogrammétrie 3D PONTOS de la société GOM. Celui-ci est constitué de deux caméras qui prennent des images du robot sur lequel plusieurs pastilles ont été collées. Le logiciel des fabricants du système de photogrammétrie reconnaît les pastilles et celui-ci calcule les coordonnées cartésiennes par la méthode de triangulation. Cette étape de calcul est réalisée pour chaque image enregistrée. L'auteur prend des images du robot sans appliquer aucune force et des images du robot soumis à des forces. Il calcule l'écart entre la position du robot sans aucune force et soumis à des forces. Cet écart représente la flexibilité du robot soumis à des forces externes. Dans cette première approche, l'auteur conclut que l'analyse quantitative de la flexibilité dans les corps du robot n'est pas significative, à cause de divers facteurs, tels que la précision de mesure, le calibrage du système, les problèmes de l'éclairage, la perte d'information, entre autres.

Dans leur deuxième approche, il utilise un laser poursuite FARO. Ce système est composé d'un laser et d'un système rétro-rélecteur qui est monté avec le principe de la sphère (appelé SMR, Spherically Mounted Retroreflectors). Ce principe permet de mesurer avec précision la distance entre la source et la cible, puisque le système envoie un faisceau laser vers la cible rétro-réfléchissante et le faisceau réfléchi par la cible suit le chemin inverse vers le laser. Le système « laser poursuite » intègre aussi deux codeurs orientés orthogonalement qui mesurent chacun un angle. Les coordonnées cartésiennes de la cible sont mesurées, grâce à l'information de la position du laser et aux angles mesurés par les codeurs. Le « laser poursuite » a pour inconvénient la vitesse du processus d'acquisition des images, car il est plus lent que le système de photogrammétrie, mais a pour avantage sa précision. L'auteur décide d'utiliser le système « laser poursuite », car dans son cas d'étude, la précision est un point très important.

Afin d'obtenir les flexibilités du robot, l'auteur reprend l'expérimentation réalisée dans sa première approche avec le système photogrammétrie. L'auteur intègre trois pastilles sur chaque articulation du robot pour une mesure plus précise et puis il mesure la position actuelle du robot sans aucune force. Ensuite il soumet le robot à des forces externes et puis il mesure sa position. Finalement, il calcule les écarts de position de chaque articulation. Afin de comparer ces résultats, l'auteur réalise aussi une analyse numérique par éléments finis (EF) avec le logiciel CATIA. Les résultats du modèle numérique sont comparés aux résultats obtenus par le laser poursuite, l'auteur déduit que les déformations calculées par le laser sont plus fiables que celles calculées par le modèle numérique.

Afin de calculer la flexibilité de l'effecteur lorsqu'il est soumis à une force, l'auteur utilise le modèle élasto-statique du robot dans une tâche de détournage robotisée. La flexibilité est alors calculée par la matrice de raideur cartésienne du robot (CaSM, cartesian Stiffness Matrix) qui permet de connaître les déformations linéaires et angulaires de l'effecteur. Le calcul de la flexibilité permet d'améliorer les opérations robotisées qui sont soumises à des forces externes à partir de la correction de la trajectoire du robot et/ou de l'outil. L'auteur simule la trajectoire avec un algorithme qui prend en compte les efforts de coupe et les flexibilités du robot (modèle élasto-statique). Ces résultats démontrent que la trajectoire consignée et la trajectoire simulée possèdent des variations, surtout lorsque l'outil entre ou sort du matériau. L'auteur considère les corps du robot comme flexibles, mais seulement dans leurs rotations autour de l'axe Z, et les liaisons du robot comme des ressorts de torsion à un DDL autour de l'axe géométrique de la liaison. Le robot est modélisé avec les hypothèses « corps rigides et liaisons flexibles à un DDL autour de l'axe géométrique » et « les raideurs des ressorts placés aux articulations prennent en compte les flexibilités des corps autour de l'axe Z ».

L'auteur propose deux méthodes pour faire la correction, méthodes basées sur le modèle élasto-statique.

La première méthode recherche un placement adapté de la pièce dans l'espace de travail du robot. Cette méthode est couplée à la gestion de la redondance cinématique du robot selon la tâche à réaliser. Elle permet de diminuer les défauts dimensionnels de la pièce dus aux déformations du robot, par contre l'auteur doit identifier les raideurs articulaires et connaître ou mesurer les efforts exercés sur le robot. Cette méthode est limitée par la flexibilité du robot.

Dans la deuxième méthode, l'auteur programme une adaptation de la méthode miroir, cette méthode consiste à simuler avec le logiciel MATLAB la trajectoire qui sera réellement réalisée par le robot, afin de connaître le défaut à corriger, puis à calculer la trajectoire déformée à partir de la connaissance de la trajectoire nominale du comportement élasto-statique du robot et des

efforts soumis sur l'effecteur. La méthode modifie le programme envoyé au robot en considérant l'erreur entre les trajectoires, afin que la trajectoire déformée soit la même que la trajectoire nominale. Cette trajectoire prend en compte, en même temps, l'évolution des efforts de coupe et le comportement élasto-statique du robot. La méthode miroir corrige la trajectoire de l'outil et compense les erreurs dues à la flexibilité du robot. La trajectoire avec la méthode miroir peut être programmée hors-ligne, à condition d'avoir le modèle des efforts externes. Autrement le robot doit être arrêté, pour mesurer les écarts entre les trajectoires. Les solutions proposées dans ce travail sont mises en œuvre directement sur le contrôleur du robot par la programmation en langage du robot.

Dans le travail présenté par Gallot (Gallot et al., 2012), l'auteur reprend le modèle élasto-statique du robot et la méthode miroir extraits du travail réalisé par Dumas (Dumas; et al., 2010; Dumas, 2011; Dumas et al., 2011). La méthode miroir considère uniquement les flexibilités du robot, sans prendre en compte les flexibilités de l'outil. Cette méthode peut être exprimée sous la forme d'un modèle d'asservissement de la position réelle de l'effecteur à la position de consigne (figure 2.6). Les paramètres indispensables pour la mise en place de cette méthode sont : les conditions de coupe, la position et la dimension de la pièce, ainsi que tous les points de la trajectoire de consigne. L'auteur conclut que l'implémentation de cet asservissement dans une baie de commande de robot, n'est pas facile, mais son principal avantage est la prise en compte de la déviation du robot lors de l'usinage. L'auteur propose la correction dynamique de la trajectoire du robot à partir de l'utilisation d'un laser poursuite. Ce capteur permet d'obtenir en temps réel les déplacements de l'outil et de quantifier le manque de précision intrinsèque du robot. La flexibilité globale du robot est prise en compte, grâce à l'utilisation de ce capteur, sans employer le modèle élasto-statique, ni le modèle des efforts appliqués sur l'outil, mais l'auteur n'a pas connaissance des causes qui provoquent la flexibilité. Le capteur est connecté au robot par le module RSI à partir du protocole Ethernet TPC/IP, comme dans le travail effectué par Uhart (Uhart et al., 2013; Uhart, 2014; Uhart et al., 2014). Le « laser poursuite » mesure en temps réel la position de la cible qui est fixée à l'extrémité de la broche. L'information de la position mesurée de la broche est envoyée au serveur par la connexion Ethernet. En même temps, l'information de la position commandée de la broche est envoyée au serveur. Un ordinateur récupère l'information de la position actuelle et calcule l'erreur entre la trajectoire commandée et la trajectoire réelle. Cette erreur est envoyée au robot, afin de corriger la trajectoire de la broche. La correction de la mesure obtenue par le laser poursuite est du type PID.

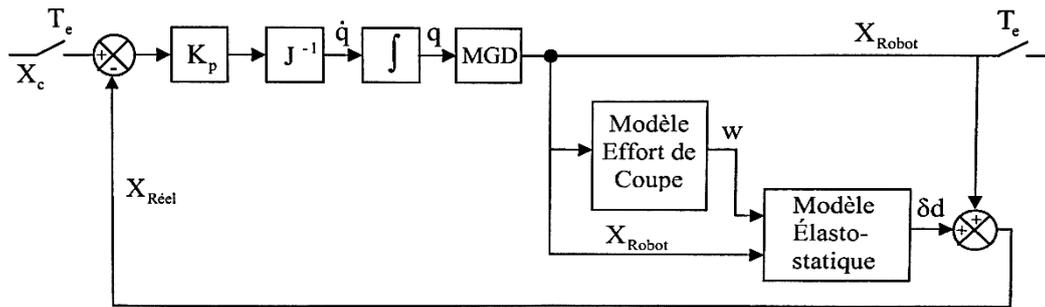


Figure 2.6 Schéma d'asservissement méthode miroir

Dans le travail présenté par Makarov (Makarov, 2013), l'auteur considère une application du type interaction homme-robot pour la réalisation collaborative de tâches complexes dans le milieu industriel. Les robots en interaction doivent assurer la sécurité des opérateurs, ainsi que ces mouvements soient précis. Ces robots possèdent une structure mécanique plus légère et flexible comparée aux robots industriels traditionnels. L'auteur doit donc considérer les paramètres de raideur du robot pour améliorer sa précision. Le robot utilisé pour l'expérimentation est un robot à sept DDL, nommé ASSIT, par contre l'auteur utilise pour son analyse seulement deux DDL. Ce robot ne dispose pas d'une loi de contrôle en boucle fermée comme ceux des sociétés KUKA, ABB, MOTOMAN... Donc le contrôle peut être modifié sans affecter la garantie du robot. Afin d'identifier les paramètres de flexibilité et les intégrer dans une loi de commande, l'auteur sélectionne une caractérisation du système flexible dans le domaine fréquentiel adaptée à la synthèse de loi de commande robuste. Les paramètres du système flexible sont modélisés par un ensemble de modèles linéaires multi-variables et résonants représentatifs des variations du système en fonction de sa configuration. L'objectif de son travail consiste à développer une loi de contrôle précise qui prend en compte les paramètres de raideur dans les articulations du robot, les fréquences naturelles du robot, la gravité, les efforts externes et internes du robot et les modèles cinématique et dynamique du robot.

L'auteur élabore deux méthodes de synthèse de commande robuste pour suivre la trajectoire et amortir les vibrations. La première méthode prend en compte le formalisme de commande prédictive GPC (Commande Prédictive Généralisée) qui permet de robustifier un correcteur initial soumis à des contraintes fréquentielles et temporelles (figure 2.7).

La deuxième méthode proposée est la technique H^∞ (figure 2.8). Cette méthode comporte deux étapes qui sont la correction à 2 DDL et l'anticipation. Ces méthodes permettent de pallier les insuffisances des correcteurs GPC et H^∞ initiaux qui se traduisent pour la commande GPC par un comportement non amortissant et non robuste, et pour la commande H^∞ par une précision

insuffisante en suivi moteur. L' auteur en conclut que les correcteurs finaux mis en place pour le robot ASSIST combinent les propriétés anticipatives, robustes et amortissantes (Makarov, 2013). Les deux méthodes proposés par Makarov (Makarov, 2013) améliorent la trajectoire de l'effecteur et la précision du robot. La commande H_∞ est légèrement plus robuste et moins sensible aux couplages que la commande GPC, par contre la commande GPC robustifiée est légèrement plus amortissante que la commande H_∞ sur le système nominal.

Les deux méthodes sont mises en œuvre via une synthèse sur des modèles simplifiés des deux axes, en conséquence, les réglages de la pré-compensation et de la boucle interne d'amortissement sont similaires à la pré-compensation et à la boucle interne d'amortissement du modèle rigide.

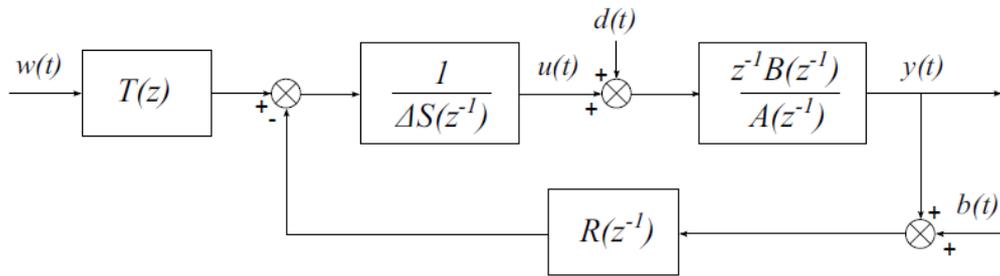


Figure 2.7 Schéma de commande prédictive de type GPC-correcteur polynomial RST à deux degrés de liberté

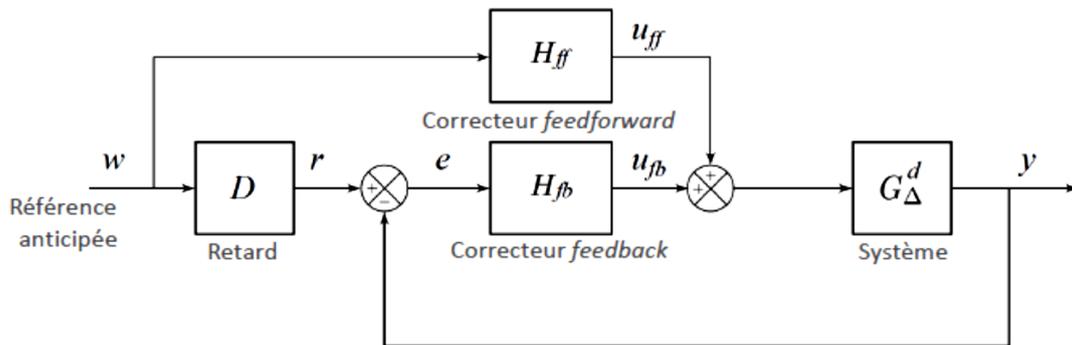


Figure 2.8 Schéma de commande H_∞ à deux degrés de liberté avec anticipation

Afin de garantir une interaction sûre, l'auteur emploie un algorithme de détection de collisions du robot avec son environnement sans capteur d'effort et en présence d'incertitudes de modélisation dans le contexte applicatif des robots et de leurs propriétés mécaniques (figure 2.9). L' algorithme de détection de collisions proposé par Makarov (Makarov, 2013) consiste en une méthode améliorée de l'évaluation du résidu qui est créé à partir de la différence entre le couple réel et le couple estimé. Cette méthode est composée d'un filtrage et d'une comparaison avec une entrée dynamique en fonction de l'état du système qui permet :

- d'isoler les effets dynamiques à moyennes et hautes fréquences de la collision.

- de différencier les variations du résidu dues aux incertitudes et liées à l'état du système des variations dues à la collision.

Une approche modulaire de type « boîte noire » permet de tenir compte de différents types d'incertitudes dominantes pour le robot, sans qu'un modèle raffiné du comportement physique (par exemple des frottements) soit nécessaire. L'approche proposée inclut un mécanisme d'adaptation en ligne, ce qui réduit considérablement la procédure de réglage et assure l'efficacité de l'algorithme dans des conditions de fonctionnement variables.

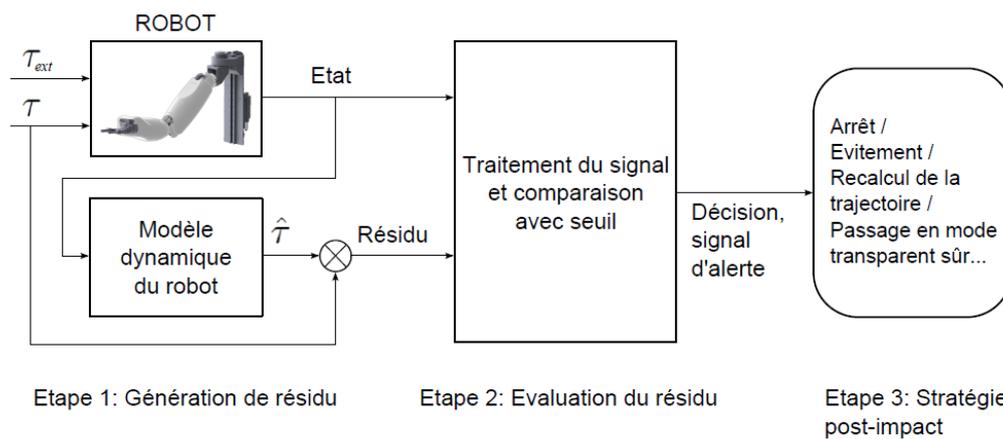


Figure 2.9 Schéma de l'algorithme de détection de collisions

2.2 Fabrication des pièces composites

Pour la fabrication des pièces en composites, deux matériaux sont généralement utilisés soit les matériaux pré-imprégnés, soit les fibres sèches (Parneix & Lucas, 2000). Les matériaux pré-imprégnés peuvent être des matrices thermodurcissables ou des matrices thermoplastiques. Ces matrices se présentent généralement sous la forme d'un tissu (figure 2.10) (Uhart, 2014) ou d'une mèche enrobée de résine pré-catalysée (figure 2.11) (Uhart, 2014). Cette résine participe à l'union du matériau entre chaque pli.



Figure 2.10 Tissu en carbone enrobé de résine

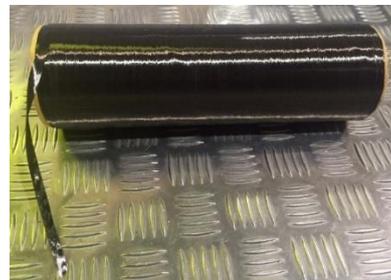


Figure 2.11 Mèche en carbone enrobée de résine

La fabrication des pièces en composites se divise principalement en deux groupes : le drapage manuel et le drapage automatisé.

Le drapage manuel consiste à appliquer les matrices ou les fibres sur un moule généralement ouvert (figure 2.12), puis les matrices sont placées suivant les spécifications de fabrication (nombres de plis, l'orientation de dépose...). Les fibres sont compactées avec un rouleau ébulleur pour enlever l'air et assurer un bon contact entre les plis et obtenir aussi une meilleure homogénéité dans la pièce. Dans certains cas, les fabricants utilisent un moule sous vide, afin d'obtenir une qualité supérieure de la finition des pièces en composites (Parneix & Lucas, 2000; Uhart, 2014). Finalement, les pièces en composites doivent être cuites dans une étuve ou en autoclave, afin d'effectuer la polymérisation de la résine.



Figure 2.12 Moule ouvert pour dépôt des fibres

Le processus de drapage suit les mêmes étapes dans les deux cas, la seule différence se situe dans la façon de déposer le matériau. Dans le drapage automatisé, il est possible de différencier deux types de drapage : le placement automatisé de bandes (ATL Automated Tape Laying) et le placement automatisé de fibres (AFP Automated Fiber Placement).

L'ATL consiste en l'application des bandes (tissus) pré-imprégnées sur des surfaces planes ou à faible courbure. Ces bandes sont déposées de façon unidirectionnelle (Sloan, 2008; Contreras, 2014). L'ATL utilise généralement un robot du type cartésien pour réaliser le drapage (figure 2.13). L'outil de drapage dans l'ATL est composé d'une bobine, des guides de bobinage, d'un rouleau de compactage et d'un élément coupant.



Figure 2.13 Robot utilisée dans l'ATL (Sloan, 2008)

L'AFP consiste en l'application automatique de fibres individuelles pré-imprégnées (Debout et al., 2011) ou de fibres sèches sur des surfaces planes avec des contours complexes (figure 2.14)

ou des surfaces avec des courbes importantes (figure 2.15). L'outil de dépôt de fibres est conçu pour effectuer différentes tâches pendant le drapage comme : la dépôt, le compactage et le découpage des fibres. Selon le fabricant, l'outil de drapage AFP est généralement composé d'un rouleau de compactage, d'un élément coupant, d'un système de chauffe (laser ou lampe infrarouge), d'un système de guidage de mèches, d'une bobine et des guides de bobinage. Les avantages de cette méthode par rapport à l'ATL sont : la dépôt des fibres individuelles et de façon multidirectionnelle, la réduction des excédents (figure 2.16), la réduction de poids dans les pièces et la réalisation des pièces avec des contours complexes et des surfaces comportant de courbes.



Figure 2.14 Contours complexes en composites
(Electroimpact, 2016)



Figure 2.15 Surfaces complexes en composites
(MetalWorking, 2015)

Les limitations de la méthode AFP pour déposer les fibres sur des surfaces comportant de courbes ou de contours complexes sont : la largeur et la longueur des fibres et le nombre de couches (Contreras, 2014).

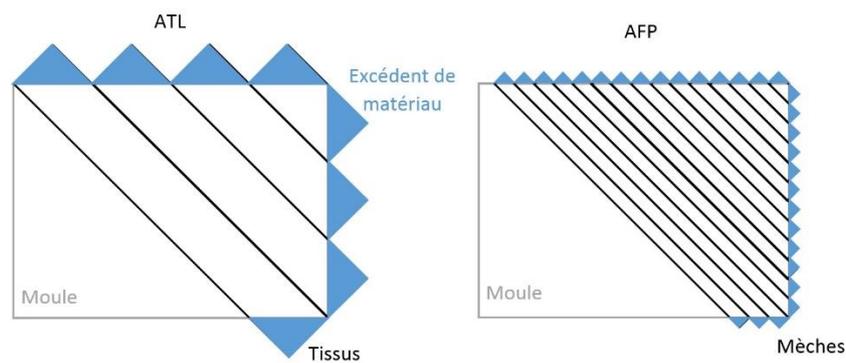


Figure 2.16 Excédent des fibres pendant la dépôt de fibres par les méthodes ATL et AFP

L'AFP utilise généralement un robot manipulateur (figure 2.17) (Uhart, 2014). La cellule robotisée AFP basique possède sept degrés de liberté : six degrés de liberté pour le bras et un pour le moule (Contreras, 2014), ces degrés de liberté peuvent augmenter lorsque le robot est monté sur un rail ou que le moule est monté sur un axe externe. Le nombre de fibres utilisées dépend de la forme de la pièce.



Figure 2.17 Robot utilisé dans l'AFP (Coriolis-Composites, 2016)

2.3 Cellule Robotisée à CompositAdour

La cellule robotisée à CompositAdour est composée d'un robot manipulateur KUKA KR 240 monté sur un rail linéaire de 16 mètres de longueur, d'un effecteur pour déposer les fibres (figure 2.18), d'une table aspirante ou d'un moule d'acier pour le drapage, d'un cantre où sont placées les bobines de mèches de carbone et d'une gaine qu'empruntent les mèches entre le cantre et l'effecteur.

L'effecteur est composé d'un système de chauffe (laser ou lampe infrarouge), un rouleau de compactage, un système de coupage et un système de guidage de mèches.

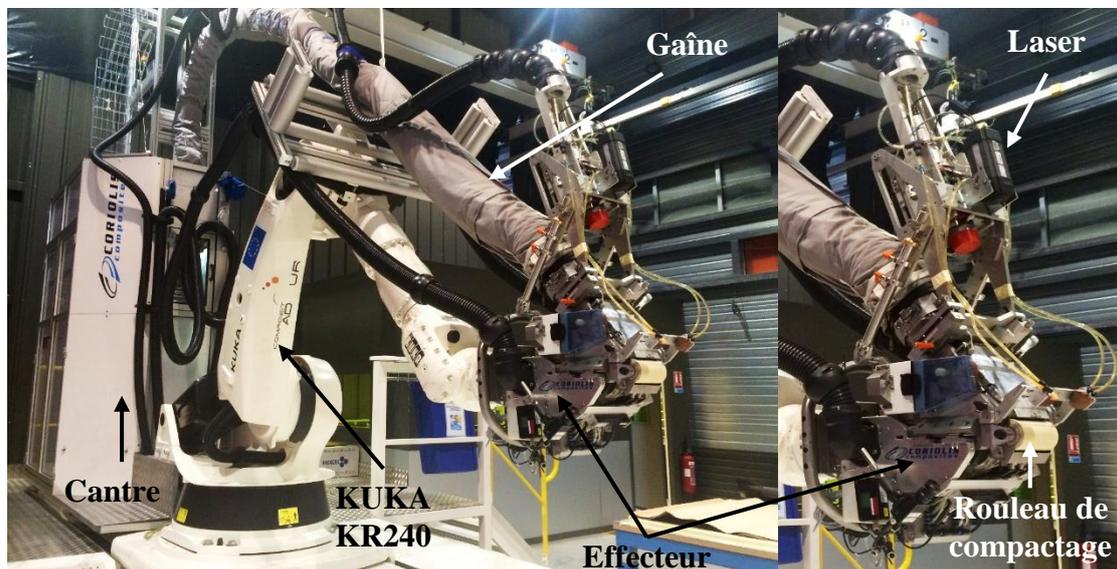


Figure 2.18 Cellule robotisée à CompositAdour

2.3.1 Système de chauffe

Le système de chauffe permet l'union des fibres entre plis grâce à la résine pré-catalysée pour éviter leurs détachements des plis précédents. Le système de chauffe est adapté en fonction du type du matériau, des fibres utilisées, ainsi que des températures appliquées. Le système de

chauffe pour les matrices thermodurcissables, consiste en une lampe infrarouge, parce que ces matrices possèdent une température de polymérisation autour de 50 °C. En revanche, pour les matrices thermoplastiques et les fibres sèches, le système de chauffe utilisé consiste en un laser, car les températures de polymérisation de ces matériaux sont entre 150 °C et 200 °C pour les matrices thermoplastiques et entre 350 °C et 400 °C pour les fibres sèches. Les fibres en forme de mèche sont chauffées juste avant leur dépose sur le moule.

2.3.2 Matériau

La largeur d'une mèche en composite (carbone) est égale à $6,35 \text{ mm} \pm 0,05 \text{ mm}$. Les fibres possèdent une épaisseur d'entre 0,1 mm et 0,3 mm selon le matériau utilisé. L'outil utilisé dans la cellule robotisée CompositAdour conçu par Coriolis possède trois têtes différentes. Selon la tête utilisée, le robot peut déposer d'une à huit mèches, d'une à seize mèches ou d'une à trente-deux mèches.

2.3.3 Commande de la tâche

Le rouleau de compactage est placé à l'extrémité de l'effecteur, afin d'exercer une pression sur les fibres déposées. Les fibres sont déposées sur le moule ou le pli précédent, grâce à cette pression et à la pré-catalysation (système de chauffe). La pression est directement affectée par l'effort de compactage et la surface de contact. L'effort de compactage est géré en ajustant ou en contrôlant la pression à l'intérieur d'un vérin pneumatique qui est positionné entre le robot et l'effecteur. La principale restriction du système appliqué est la direction de l'effort de compactage (F_z) qui doit être normale à la surface (figure 2.21). Afin de définir les déplacements de l'outil en position et en orientation, le système actuel est programmé hors-ligne, donc les déplacements du robot sont définis dans le repère du TCP (Debout et al., 2011). La position le long de l'axe X permet de suivre la trajectoire en vitesse (figure 2.19) et l'orientation autour de l'axe X permet d'appliquer l'effort de compactage de manière homogène sur toute la surface de contact (figure 2.19). La position le long de l'axe Y permet de déposer les fibres par rapport à la définition du moule ou du pli (Figure 2.20) et l'orientation autour de l'axe Y est fixée à 90 degrés (figure 2.20), car elle est contrainte par la direction de l'effort de compactage (F_z) (Debout et al., 2011; Uhart, 2014). La position le long de l'axe Z permet de positionner l'outil pour être en contact avec le moule et activer le vérin pneumatique (figure 2.19) et l'orientation autour de l'axe Z permet de positionner la bande selon les spécifications de drapage (figure 2.21).

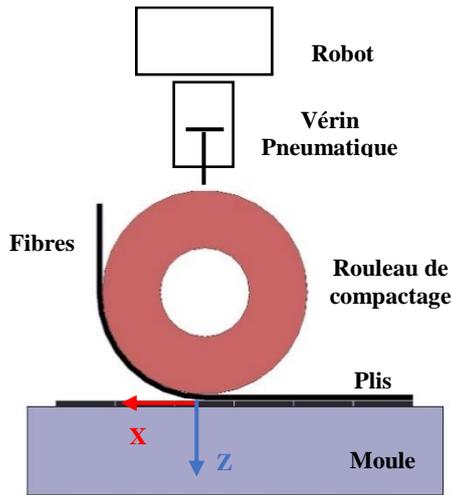


Figure 2.19 L'orientation et la position du TCP dans le plan XZ

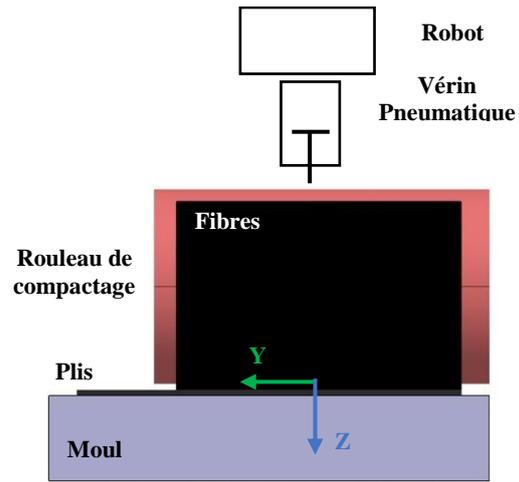


Figure 2.20 L'orientation et la position du TCP dans le plan YZ

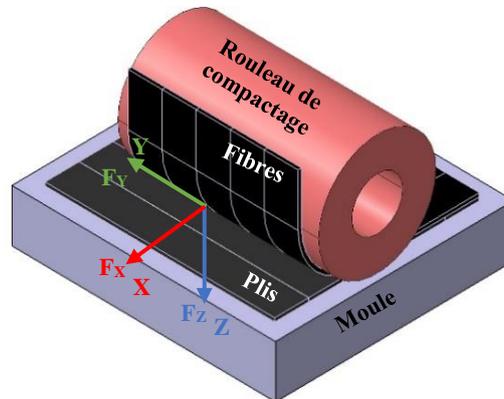


Figure 2.21 L'orientation et la position du TCP dans le plan isométrique

2.4 Limites du système actuel

Pendant la dépose de fibres dans le système actuel, divers défauts peuvent apparaître comme l'espacement (gap), le recouvrement (overlap) et la déviation angulaire entre fibres ou entre plis (figure 2.22). Ces défauts fournissent une mauvaise qualité à la pièce et sont dus à divers facteurs comme le guidage des fibres, la largeur des fibres hors de la tolérance acceptée, la configuration du robot, la position/orientation incorrecte de l'outil, l'effort de compactage non homogène sur les fibres...

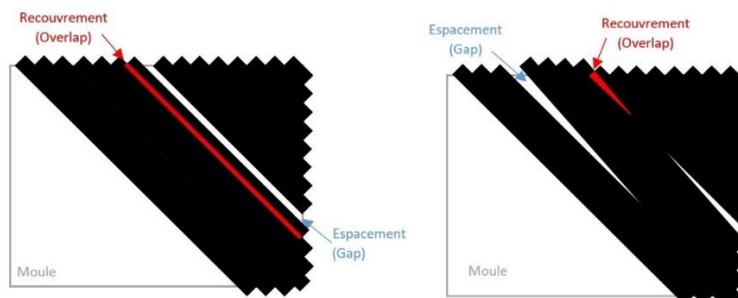


Figure 2.22 Défauts de la dépose de fibres : Overlap et Gap

Le ruban du système de guidage permet le passage des fibres de 6,4 mm. Lorsque les fibres dépassent cette dimension, les défauts d'espacement et de recouvrement peuvent apparaître. Cependant, l'ajustement des dimensions de fibres est un problème indépendant du système de dépose, donc ce problème n'est pas traité dans le cadre de ce travail.

Un autre facteur qui provoque des erreurs dans la dépose de fibres, est la configuration du robot, car selon celle-ci, la structure mécanique du robot peut être plus ou moins résistante aux efforts externes. Ensuite, la position/orientation de l'outil incorrecte entraîne une différence entre la dépose de fibres théorique et la dépose de fibres réelle.

L'effort de compactage est généré par le vérin en boucle ouverte sur l'effort, car il n'y a pas de capteur d'effort sur le système. Le travail d'Uhart (Uhart, 2014) propose l'implémentation d'un capteur d'effort entre l'outil et le robot, pour contrôler l'effort de compactage en boucle fermée. Ce contrôle permet que l'effort de compactage reste constant pendant la dépose de fibres sur des surfaces planes ou à faibles courbures avec des aires constantes, donc le compactage des fibres est homogène. Mais ce contrôle de l'effort de compactage ne peut pas garantir un compactage homogène sur des surfaces avec des courbures importantes, sur des surfaces avec des coins et sur des surfaces planes avec un changement de ces dimensions, car la surface de contact est modifiée selon les dimensions de la pièce.

- *Tâche de dépose*

Dans le travail d'Uhart (Uhart, 2014), l'effort de compactage pendant la dépose a été maîtrisé selon le matériau utilisé, ce qui a permis de limiter les efforts parasites dans les directions commandées en position et d'assurer la qualité de la pièce qui est garantie seulement en surfaces planes, car les tests présentés par l'auteur sont réalisés uniquement dans ces types de surfaces. Dans le cadre de cette recherche, l'objectif est de maîtriser la pression de compactage, afin de garantir une dépose homogène des fibres pour des pièces complexes (avec des coins, pièces avec variations de la surface de contact, des pièces courbes...). Cette pression correspond à l'effort de compactage sur la surface de contact. L'effort de compactage est défini par l'utilisateur selon le matériau des fibres. Par contre, la surface de contact est déterminée par deux éléments tels que la pièce à fabriquer et le contact du rouleau de compactage. L'approche de cette recherche est donc fondée sur la relation qu'il y a entre la pression P , la force F et la surface de contact S soit : $P = F/S$.

Dans la définition actuelle du processus de dépose, l'effort de compactage pour la fibre sèche est d'environ 5 N, pour le matériau thermoplastique d'environ 200 N et pour le matériau

thermodurcissable entre 500 N et 1200 N. L'effort de compactage doit être également réparti sur la surface de contact, pour assurer que toutes les fibres subissent le même effort.

Les efforts sont dus au contact permanent entre les éléments (le rouleau de compactage, les fibres et le moule) et à l'effort de compactage. Ces efforts peuvent provoquer des modifications importantes et irréversibles dans la géométrie des éléments en contact. Le moule utilisé pendant la dépose de fibres est en acier (figure 2.12). Grâce aux propriétés mécaniques de ce matériau, les efforts appliqués sur ce type de moule peuvent être supérieurs à ceux utilisés pour la dépose des fibres, sans impliquer des changements importants de la géométrie du moule. Ce dernier est donc considéré comme un corps rigide. Les autres éléments en interaction sont les fibres, dont l'épaisseur comprise entre 0,1 mm à 0,3 mm (figure 2.11) est inférieure à la déformation du rouleau de compactage. Dans le cadre de cette recherche, la déformation dans ces éléments est considérée comme négligeable.

Le rouleau de compactage est l'autre élément en interaction pendant la dépose de fibres. Cet élément est composé de différentes couches de Sylomer du Type P, de colle entre chaque couche et d'une couche de gaine à l'extérieur (figure 2.23). Les propriétés de comportements mécaniques de cet élément multi-matériaux ne sont pas précisées par le fabricant. Pourtant, le rouleau de compactage est un élément très souple. Il se déforme énormément sous l'effort de compactage. Dans le travail de (Uhart, 2014), l'auteur considère le rouleau de compactage comme un élément non déformable et sans glissement.



Figure 2.23 Rouleau de compactage

2.5 Conclusion

Les travaux proposés par les entreprises s'adressent principalement aux robots industriels dans lesquels la baie de commande n'est pas modifiable, par contre ces travaux considèrent que les flexibilités dans les outils et dans le robot sont dues aux efforts externes. Ces efforts peuvent être mesurés à partir des capteurs proprioceptifs (codeurs incrémentaux) et extéroceptifs

(télémètre laser, camera). Dans les travaux (DeVlieg et al., 2002; Atkinson et al., 2007; DeVlieg, 2009, 2010; DeVlieg & Szallay, 2010; DeVlieg, 2011; Jeffries, 2013) les auteurs proposent de commander le robot à partir d'un système externe programmé comme une machine-outil à commande numérique (CN). Dans ces travaux, les auteurs doivent implémenter un nouveau contrôle du robot compatible avec la commande numérique. Ce contrôle doit considérer les modèles cinématique et dynamique du robot, ainsi que les flexibilités du robot et de l'outil. Cependant, d'autres travaux présentés par les industriels proposent d'ajouter à la baie de commande du robot, un système de commande pour l'outil. Ce système utilise une boucle externe pour envoyer l'information obtenue par les capteurs externes à la baie de commande du robot. Cette information permet de corriger la trajectoire du robot sans modifier la baie de commande du robot.

Les travaux proposés par les centres de recherche utilisent des robots industriels avec une baie de commande restrictive, mais aussi des robots avec une baie de commande flexible et complètement modifiable. Ces travaux sont divisés en deux, soit l'utilisation d'une commande en boucle externe à la baie de commande du robot ou soit la création d'une commande qui modifie ou substitue la baie de commande du robot. Dans (Gallot et al., 2012; Uhart, 2014), les auteurs utilisent une boucle externe pour corriger la trajectoire de l'outil à partir des capteurs extéroceptifs, afin d'améliorer les diverses tâches exposées à des forces externes et dans lesquelles le robot est en contact permanent avec la surface. D'un autre côté, dans les travaux de (Dumas, 2011; Makarov, 2013), les auteurs intègrent les flexibilités du robot dues à des forces externes au modèle cinématique. Ces forces externes sont prédites à partir d'algorithmes de calcul, donc les auteurs ne considèrent pas les informations de capteurs extéroceptifs.

L'étude des solutions proposées permet de dresser la problématique de la prise en compte des flexibilités, ainsi que les différentes solutions pour l'intégration de la flexibilité dans la baie de commande et finalement les facteurs qui doivent être considérés pour améliorer la précision pendant l'exécution des tâches robotisées. Ces éléments sont principalement les forces externes, les modèles cinématique et dynamique du robot et les déformations des éléments en contact (l'outil, le robot, l'environnement).

L'analyse de la fabrication des pièces en composite nous permet de comprendre les différents procédés implémentés pour la dépose de fibres, les différentes étapes, les composants utilisés et les limites. Ainsi, l'approfondissement sur la cellule robotisée à CompositAdour nous permet de centrer notre cas d'étude et d'évaluer les composants, afin de poser les problèmes qui causent une dégradation de la tâche et une mauvaise qualité des pièces. Finalement, la connaissance des

limites du système actuel nous permet de réaliser une proposition qui participe à l'amélioration de la tâche de dépose des fibres.

La dépose de fibres est une tâche dans laquelle le robot et les éléments de dépose (fibres et moule) sont en contact permanent et l'outil doit exercer une pression homogène sur les fibres pour améliorer leur adhérence sur le moule et/ou le pli précédent. Le robot de la cellule robotisée pour la dépose de fibres est du type industriel avec une baie de commande non modifiable. L'outil de dépose de fibres est composé par différents matériaux. La combinaison de ces matériaux fournit à l'outil de la raideur lorsqu'il est soumis à des efforts importants.

CHAPITRE

3

Flexibilités du système

Sommaire

3.1 Modélisation de robot.....	29
3.1.1 Modèles géométrique et cinématique du robot.....	29
3.1.1.1 Modèle géométrique	29
3.1.1.2 Modèle cinématique.....	33
3.1.2 Dynamique des robots	35
3.1.2.1 Robots rigides	35
3.1.2.2 Robots ayant des flexibilités	35
3.2 Flexibilités de l'outil	39
3.2.1 Outil rigide.....	40
3.2.2 Outil flexible.....	43
3.3 Modélisation des flexibilités de l'outil.....	49
3.3.1 Modèle expérimental	49
3.3.2 Modèle numérique par éléments finis	52
3.3.3 Recalage du modèle numérique.....	54
3.3.4 Modèle EF discrétisé	57
3.4 Conclusion.....	59

Ce chapitre est consacré à l'étude et à la modélisation des robots industriels et de leur outil. Cette modélisation permet de connaître les facteurs qui génèrent des imprécisions sur les tâches robotisées. En premier lieu, nous allons exposer les modèles géométrique et cinématique du robot qui nous permettent de comprendre la dynamique du robot. Ensuite, nous présentons le modèle dynamique pour un robot rigide, pour un robot avec des articulations flexibles et des corps rigides et pour un robot avec des corps flexibles et des articulations rigides. Les modèles

dynamiques pour les deux derniers cas considèrent les flexibilités sur le robot soumis aux forces externes. Finalement, nous montrons l'importance du calibrage de l'outil lors de l'exécution de la tâche et la modélisation d'un outil lorsqu'elle est soumise aux forces externes.

3.1 Modélisation de robot

Les robots industriels sont composés de segments rigides reliés par des articulations, qui permettent le mouvement relatif des corps consécutifs. Ces articulations peuvent être des articulations rotoïdes (elles donnent un degré de liberté et elles permettent la rotation autour de l'articulation) ou prismatiques (elles donnent un degré de liberté et elles permettent la translation le long de l'articulation). Généralement, les robots industriels installés dans les entreprises possèdent six degrés de liberté, puisqu'ils ont besoin de positionner et d'orienter un corps dans l'espace. Ces six DDL sont divisés en deux groupes : trois pour le positionnement (tronc, épaule et coude) et trois pour l'orientation (poignet). Les robots industriels avec six degrés de liberté ont généralement une structure géométrique série et cela facilite la modélisation et l'identification des paramètres.

3.1.1 Modèles géométrique et cinématique du robot

3.1.1.1 Modèle géométrique

Le modèle géométrique du robot permet d'établir un lien entre la position et l'orientation du poignet et les valeurs des coordonnées des articulations.

Le modèle géométrique repose sur deux aspects, le modèle géométrique direct (MGD) et le modèle géométrique inverse (MGI). La première forme (MGD) permet de calculer la position et l'orientation du poignet, par rapport au système de coordonnées de référence. La position et l'orientation sont calculées à partir des valeurs des articulations et des paramètres géométriques des éléments du robot. La deuxième forme (MGI) détermine la ou les configurations (variables articulaires) que le robot doit prendre pour atteindre une position et une orientation données (Barrientos & Balaguer, 2007).

- **Modèle géométrique direct (MGD)**

La position et l'orientation de l'élément terminal du robot sont obtenues en déterminant la matrice de transformation T représentée par (3.1). Le MGD pour un robot à n articulations est représenté par (3.2).

$$T = {}^0T_j = {}^0T_1 \cdot {}^1T_2 \cdots {}^{j-1}T_j \quad (3.1)$$

$$X=f(q) \tag{3.2}$$

Tels que :

q étant le vecteur des variables articulaires tel que : $q = [q_1 \ q_2 \ \dots \ q_n]$,

Les coordonnées cartésiennes sont définies par le vecteur $X : X = [x_1 \ x_2 \ \dots \ x_m]$.

La matrice de passage peut être exprimée par :

$${}^{(j-1)}T_j = \begin{bmatrix} s_x & n_x & a_x & p_x \\ s_y & n_y & a_y & p_y \\ s_z & n_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

Cette matrice est composée par un vecteur de position $\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$ et une matrice de rotation

$$\begin{bmatrix} s_x & n_x & a_x \\ s_y & n_y & a_y \\ s_z & n_z & a_z \end{bmatrix}.$$

Ce travail est basé dans le modèle géométrique qui est réalisé par Uhart (Uhart, 2014). L'auteur décrit les paramètres de Denavit-Hartenberg (D-H) modifiés (W. Khalil & É. Dombre, 1999; Craig, 2006; Ollero Baturone, 2006) pour le robot KUKA KR240 utilisé par la cellule robotisée de dépose de fibres à CompositAdour. Cette cellule robotisée ne peut pas être arrêtée ou modifiée, car elle fabrique des pièces en permanence et est calibrée pour accomplir les exigences de l'industrie aéronautique. En conséquence, dans notre recherche, le robot utilisé est un robot KUKA KR6. Cependant, les deux robots (KR6 et KR240) possèdent les mêmes DDL, la même géométrie et les mêmes sens de rotations des articulations, comme le montre la figure 3.1, la différence entre eux est dans les dimensions des corps.

Les paramètres de D-H modifiés pris en compte par Uhart (Uhart, 2014) sont présentés dans la section A.1 (Tableau A.1). L'auteur prend en compte un rail à la base du robot, cet élément ajoute un degré de liberté au robot, en conséquence les paramètres de DH modifiés changent. Dans notre recherche, le robot est fixé au sol, ces paramètres de DH modifiés sont présentés dans le tableau 3.1. Ces paramètres indiquent la position de repos du robot, cette position est représentée en coordonnées articulaires par $[0, -90, 90, 0, 0, 0]$. Le tableau 3.1 ajoute trois paramètres tels que : le paramètre r_1 décrit la hauteur entre le sol et le premier degré de liberté, le paramètre r_7 correspond à la hauteur du capteur d'effort qui est fixé entre le poignet du robot et l'outil et le paramètre r_8 représente la hauteur de l'outil depuis la fixation (capteur d'effort) jusqu'au rouleau de compactage. L'architecture géométrique du robot est présentée figure 3.2.

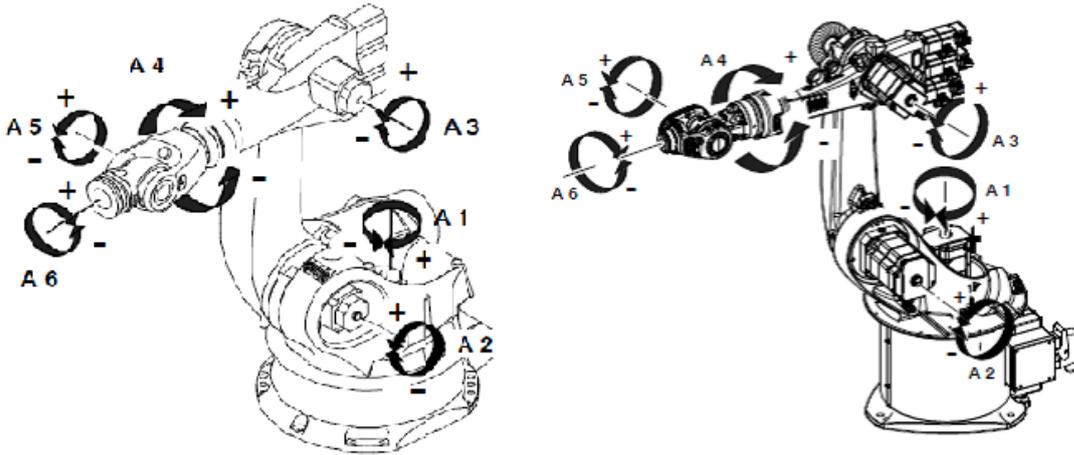


Figure 3.1 Sens de rotation des articulations KR 240 (KUKA, 2004) et KR6 (KUKA, 2003)

Tableau 3.1 Paramètres de D-H modifiés

$j^{-1}T_j$	j	Alpha (α)	d	Thêta (θ)	r
${}^{ate}T_0$	atelier	0	0	0	r_1
0T_1	1	π	0	q_1	0
1T_2	2	$\pi/2$	d_2	$q_2 - \pi/2$	0
2T_3	3	0	d_3	q_3	0
3T_4	4	$\pi/2$	$-d_4$	q_4	$-r_4$
4T_5	5	$-\pi/2$	0	q_5	0
5T_6	6	$\pi/2$	0	q_6	0
6T_7	capteur	0	0	0	$-r_7$
7T_8	outil	0	0	0	r_8

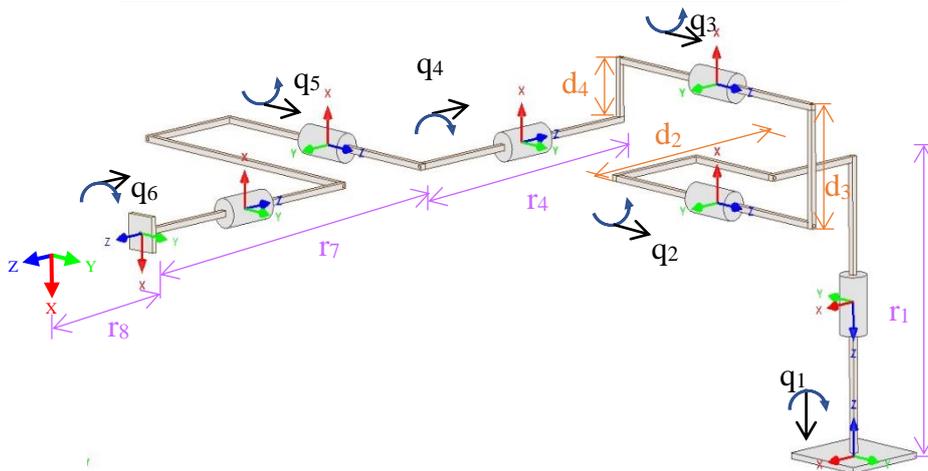


Figure 3.2 Architecture géométrique

Les matrices de transformation utilisées pour obtenir le MGD sont décrites par (3.4). Afin de garder une bonne lisibilité, les fonctions trigonométriques cosinus () et sinus () sont remplacées par C () et S () respectivement dans les matrices homogènes.

$$\begin{aligned}
 {}^0T_1 &= \begin{bmatrix} C(q_1) & -S(q_1) & 0 & 0 \\ -S(q_1) & -C(q_1) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^1T_2 &= \begin{bmatrix} S(q_2) & C(q_2) & 0 & d_2 \\ 0 & 0 & -1 & 0 \\ -C(q_2) & S(q_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^2T_3 &= \begin{bmatrix} C(q_3) & -S(q_3) & 0 & d_3 \\ S(q_3) & C(q_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^3T_4 &= \begin{bmatrix} C(q_4) & -S(q_4) & 0 & -d_4 \\ 0 & 0 & -1 & r_4 \\ S(q_4) & C(q_4) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^4T_5 &= \begin{bmatrix} C(q_5) & -S(q_5) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -S(q_5) & -C(q_5) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^5T_6 &= \begin{bmatrix} C(q_6) & -S(q_6) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ S(q_6) & C(q_6) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^6T_{7(\text{capteur})} &= \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & -r_7 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^{7(\text{capteur})}T_{8(\text{outil})} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r_8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^{\text{ate}}T_0 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{3.4}$$

Les équations de position définies à partir du MGD pour le robot KR6 sont représentées pour l'axe X par (3.5), pour l'axe Y par (3.6) et pour l'axe Z par (3.7).

$$\begin{aligned}
 X &= -\cos(q_1) \cos(q_5) \cos(q_2 + q_3) (r_7 - r_8) \\
 &+ \cos(q_1) (d_2 + d_3 \sin(q_2) - d_4 \sin(q_2 + q_3) + r_4 \cos(q_2 + q_3))
 \end{aligned} \tag{3.5}$$

$$\begin{aligned}
 &+ \sin(q_5) (\sin(q_1) \sin(q_4) + \cos(q_1) \cos(q_4) \sin(q_2 + q_3)) (r_7 - r_8) \\
 Y &= \cos(q_5) \sin(q_1) \cos(q_2 + q_3) (r_7 - r_8) \\
 &- \sin(q_1) (d_2 + d_3 \sin(q_2) - d_4 \sin(q_2 + q_3) + r_4 \cos(q_2 + q_3))
 \end{aligned} \tag{3.6}$$

$$\begin{aligned}
 &+ \sin(q_5) (r_7 - r_8) (\cos(q_1) \sin(q_4) - \cos(q_4) \sin(q_1) \sin(q_2 + q_3)) \\
 Z &= r_1 + d_3 \cos(q_2) - d_4 \cos(q_2 + q_3) - r_4 \sin(q_2 + q_3) \\
 &+ \cos(q_5) \sin(q_2 + q_3) (r_7 - r_8) + \cos(q_4) \sin(q_5) \cos(q_2 + q_3) (r_7 - r_8)
 \end{aligned} \tag{3.7}$$

- **Modèle géométrique inverse (MGI)**

Le MGD permet de connaître les valeurs des coordonnées articulaires, à partir de la position et de l'orientation de l'élément terminal du robot. Par contre, la représentation symbolique du MGI est plus complexe, car les solutions particulières doivent être calculées pour chaque position souhaitée. Afin de positionner et orienter l'effecteur dans la position souhaitée, le robot peut prendre diverses configurations en suivant ses articulations et en évitant les configurations singulières.

Dans notre recherche, le calcul du MGI est basé sur la méthode de Paul (W. Khalil & E. Dombre, 1999), car elle est adaptée à la plupart des robots industriels. Cette méthode est représentée par (3.8)(3.8) et (3.9).

$$q=f(X) \quad (3.8)$$

$$U_0={}^0T_1(q_1){}^1T_2(q_2),\dots,{}^{n-1}T_n(q_n) \quad (3.9)$$

Dans le tableau 3.2, on montre les types d'équations plus communs pour la plupart des robots industriels pour solutionner l'équation (3.9) par la méthode de Paul.

Tableau 3.2 Types d'équations rencontrés avec la méthode de Paul

Type 1	$Xr_i = Y$
Type 2	$XS\theta_i + YC\theta_i = Z$
Type 3	$X1S\theta_i + Y1C\theta_i = Z1$ $X2S\theta_i + Y2C\theta_i = Z2$
Type 4	$X1r_jS\theta_i = Y1$ $X2r_jC\theta_i = Y2$
Type 5	$X1S\theta_i = Y1 + Z1r_j$ $X2C\theta_i = Y2 + Z2r_j$
Type 6	$WS\theta_j = XC\theta_i + YS\theta_i + Z1$ $WC\theta_j = XS\theta_i - YC\theta_i + Z2$
Type 7	$W1C\theta_j + W2S\theta_j = XC\theta_i + YS\theta_i + Z1$ $W1S\theta_j - W2C\theta_j = XS\theta_i - YC\theta_i + Z2$
Type 8	$XC\theta_i + YC(\theta_i + \theta_j) = Z1$ $XS\theta_i + YS(\theta_i + \theta_j) = Z2$
Type 9	$XC\theta_i = Y$
Type 10	$XS\theta_i = Y$

3.1.1.2 Modèle cinématique

Le modèle cinématique permet de calculer les vitesses de translation et de rotation du corps terminal en fonction des vitesses articulaires. Ce modèle permet de trouver une solution au problème géométrique inverse quelle que soit la structure du robot.

Le modèle cinématique repose sur deux aspects : le modèle cinématique direct (MCD) et le modèle cinématique inverse (MCI). Le MCD décrit les vitesses des coordonnées cartésiennes en fonction des vitesses articulaires. Le MCI permet de connaître les vitesses articulaires en

fonction des vitesses des coordonnées cartésiennes. Ces modèles se basent sur le calcul de la matrice Jacobienne. Cette matrice permet aussi de formuler le modèle statique qui consisté à calculer les efforts appliquées en chaque articulation par rapport à la base, en considérant le robot fixe.

- **Modèle cinématique direct (MCD)**

Le MCD est caractérisé par (3.10) et représenté sous forme matricielle en fonction de la vitesse par (3.11) (Craig, 2006; Barrientos & Balaguer, 2007; Ansoategui et al., 2010).

$$\dot{X}=J\dot{q} \quad (3.10)$$

$$\begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_\alpha \\ \omega_\beta \\ \omega_\gamma \end{bmatrix} = J \cdot \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \quad (3.11)$$

Tel que :

- J désigne la matrice Jacobienne (m x n) du robot,
- \dot{X} indique les vitesses des coordonnées cartésiennes,
- \dot{q} soit les vitesses des coordonnées articulaires.

Le modèle des efforts statique utilise la Jacobienne pour établir un lien entre les efforts exercés sur l'effecteur et les forces et couples des actionneurs. Il est représenté par (3.12) (W. Khalil & E. Dombre, 1999)

$$\Gamma_F=J^T f_{en} \quad (3.12)$$

Tel que :

- J^T représente la matrice transposée de la Jacobienne,
- f_{en} soit le vecteur des forces de dimension (3 × 1) et le vecteur des couples de dimension (3 × 1).

- **Modèle cinématique inverse (MCI)**

Le MCI est représenté par (3.13) sous forme matricielle en fonction de la vitesse par (3.14).

$$\dot{q}=J^{-1}\dot{x} \quad (3.13)$$

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_\alpha \\ \omega_\beta \\ \omega_\gamma \end{bmatrix} \quad (3.14)$$

Le MCI dépend de la configuration du robot. En conséquence, il doit être calculé pour chaque configuration.

3.1.2 Dynamique des robots

3.1.2.1 Robots rigides

Le modèle dynamique du robot permet d'établir une relation entre les couples (et/ou forces) appliqués aux actionneurs, aux positions, aux vitesses et aux accélérations articulaires (W. Khalil & E. Dombre, 1999). L'équation générale de la dynamique du robot en termes Lagrangiens est déterminée par (3.15). Dans cette équation, le robot est considéré complètement rigide.

$$\Gamma + \Gamma_F = M(q)\ddot{q} + C(q, \dot{q}) + g(q) + D(\dot{q}) \quad (3.15)$$

Tel que :

$q = [q_1 \cdots q_n]$ soit le vecteur des coordonnées généralisées des articulations,

n soit le nombre total des articulations dans le robot,

$M(q)$ soit la matrice d'inertie,

$C(q, \dot{q})$ soit le vecteur des couples/forces de Coriolis et des forces centrifuges,

$g(q)$ soit le vecteur des couples dus à la gravité,

$D(\dot{q})$ soit le vecteur des frottements des articulations,

Γ soit le vecteur des couples moteurs.

$\Gamma_F = \mathbf{J}^T f_{en}$ soit le vecteur de forces/couples externes.

3.1.2.2 Robots ayant des flexibilités

La raideur du robot est principalement due à la flexibilité des transmissions au niveau des articulations et à la déflexion au niveau des corps. Les flexibilités au niveau des articulations sont dues à des éléments de transmission ou à des éléments de réductions du mouvement (des arbres, des câbles, des engrenages à onde de déformation « harmonic drives », d'engrenages cycloïdaux, ...). Les flexibilités au niveau des corps sont dues à leur longueur (Chedmail et al., 1991; De Luca & Book, 2008; De Luca, 2015).

- *Flexibilités dans les corps*

Le modèle Euler-Bernoulli permet de connaître les flexibilités localisées au niveau des corps. Ce modèle admet que les articulations du robot sont rigides et les flexibilités se trouvent dans les corps. Les corps sont modélisés comme des poutres. La figure 3.3 représente la flexibilité dans les corps.

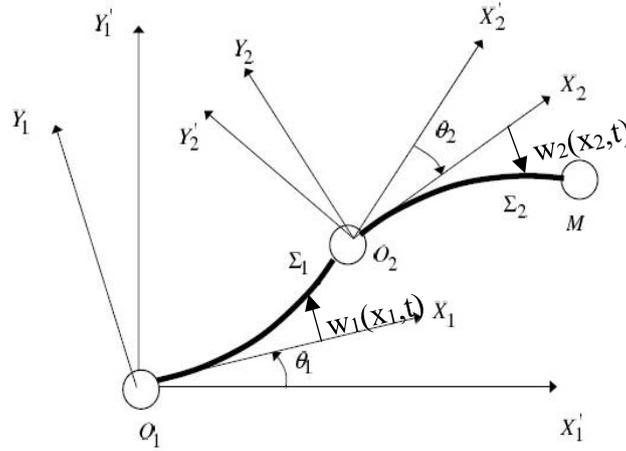


Figure 3.3 Modélisation des flexibilités dans deux corps

Le modèle Euler-Bernoulli est calculé par (3.16) pour un seul corps (Moallem, 1996; Aoustin & Formal'sky, 1997; Sooraksa & Chen, 1998; De Luca & Book, 2008; Bottega et al., 2009; Rigatos, 2009; Khairudin et al., 2011; Rigatos, 2012).

$$EI \frac{\partial^4 \omega(x,t)}{\partial x^4} + \rho \frac{\partial^2 \omega(x,t)}{\partial t^2} + \rho x \ddot{\theta}(t) = 0 \quad (3.16)$$

$$I_t \ddot{\theta}(t) + \rho \int_0^L x \frac{\partial^2 \omega(x,t)}{\partial t^2} dx = T(t)$$

Tel que :

$\omega(x, t)$ représente la flexibilité dans le corps,

$\theta(t)$ représente l'angle de l'articulation,

I_t soit le moment d'inertie du corps rigide de longueur L ,

ρ soit la densité du corps,

EI soit la rigidité à la flexion.

L'analyse modale définie par (3.17) est utilisée pour calculer la flexibilité du corps $\omega(x, t)$. Cette analyse suppose que la flexibilité du corps peut-être approchée par une somme pondérée de fonctions orthogonales de base.

$$\omega(x,t) = \sum_{i=1}^{n_e} \phi(x) q_i(t) \quad (3.17)$$

Tel que :

$i=[1, 2, \dots, n_e]$ soient les modes normaux de vibration du corps flexible.

$\phi(x)$ soit la fonction modale

En utilisant, l'analyse modale, un modèle dynamique de dimensions finies est dérivé pour la flexibilité du corps de robot. Un modèle dynamique général pour un robot possédant de corps flexibles peut être représenté par :

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+D(\dot{q})+K_e q+g(q)=u \quad (3.18)$$

Tel que :

$q = [\theta, \delta]^T$ soit le vecteur des coordonnées généralisées,

$M(q) = \begin{bmatrix} M_{\theta\theta} & M_{\theta\delta} \\ M_{\theta\delta}^T & M_{\delta\delta} \end{bmatrix}$ soit la matrice d'inertie,

$C(q, \dot{q}) = \begin{bmatrix} C_{\theta\theta} & C_{\theta\delta} \\ C_{\delta\theta} & C_{\delta\delta} \end{bmatrix}$ soit la matrice des couples/forces de Coriolis et des forces centrifuges,

$g(q) = \begin{bmatrix} g_{\delta}(\theta) \\ g_{\theta}(\delta) \end{bmatrix}$ soit le vecteur des couples dus à la gravité,

$D = \begin{bmatrix} 0_{\theta\theta} & 0_{\theta\delta} \\ 0_{\delta\theta} & D_{\delta\delta} \end{bmatrix}$ soit le vecteur des frottements,

$K_e = \begin{bmatrix} 0 & 0 \\ 0 & K \end{bmatrix}$ soit la matrice des coefficients de la rigidité,

$u = \begin{bmatrix} \tau \\ 0 \end{bmatrix}$ soit le vecteur de contrôle de couples.

Les sous-indices $\theta\theta, \theta\delta$ and $\delta\delta$ sont les termes qui correspondent aux corps rigides, aux articulations rigides et corps flexibles et aux corps flexibles respectivement.

- **Flexibilités dans les articulations**

Le modèle dynamique général décrit par (3.15) suppose que tous les composants du robot sont des éléments rigides. Le modèle élasto-statique permet de connaître les raideurs articulaires de chaque liaison et les flexibilités localisées dans chaque articulation. Ce modèle admet que les corps du robot sont rigides et que toutes les flexibilités se trouvent dans les articulations. Ces articulations seront modélisées avec une liaison ressort sans amortissement. La figure 3.4 représente une articulation flexible.

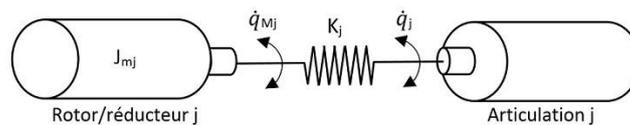


Figure 3.4 Modélisation de la flexibilité de l'articulation

Le modèle élasto-statique est calculé par (3.19) (Mark W. Spong, 1989; Rodriguez-Angeles et al., 2004; Makarov, 2013; W. Khalil & E. Dombre, 1999).

$$\begin{aligned} M(q)\ddot{q}+C(q,\dot{q})\dot{q}+D(\dot{q})+k(q-q_M)&= \Gamma_F \\ J_m\ddot{q}+B\dot{q}_m-k(q-q_M)&=\Gamma \end{aligned} \quad (3.19)$$

Tel que :

$q_M = [q_1 \cdots q_n]$ soit le vecteur des coordonnées généralisées du moteur,

$k = \text{diag}[k_1 \cdots k_n]$ soit la matrice diagonale des coefficients de la rigidité dans chaque articulation,

$J_m=[J_1 \cdots J_n]$ soit la matrice d'inertie du robot,

$B = [B_1 \cdots B_n]$ soit le vecteur de frottement cinématique du moteur.

- **Méthodes pour calculer les flexibilités**

Dans le domaine de la robotique, beaucoup de travaux sont développés pour traiter le calcul de la matrice de raideur. Les trois méthodes les plus utilisées sont : la méthode des liaisons virtuelles (VJM), l'analyse par éléments finis (EF) et l'analyse structurale matricielle (MSA) (Pashkevich et al., 2009; Ahmad et al., 2011; Klimchik et al., 2012).

- **Flexibilités structurelles**

L'EF est une méthode numérique qui permet de trouver une solution approchée du système par l'introduction de sa formulation variationnelle et l'introduction d'éléments finis pour résoudre numériquement cette formulation. Cette méthode est très utilisée pour connaître les flexibilités structurelles du robot et de l'effecteur. Pourtant, l'EF ne donne pas la relation analytique entre la raideur et les dimensions du système (Ahmad et al., 2011). Cette méthode requiert un coût élevé de calcul (Pashkevich et al., 2009; Ahmad et al., 2011). Dans le travail de (Partida et al., 2010; Duriez, 2013), les auteurs calculent les coefficients de la matrice de flexibilité à partir de l'EF. Les simulations dans le logiciel EF consistent à soumettre l'outil et le robot aux forces externes, donc ces éléments subiront des déformations. Ces déformations sur chaque axe sont égales aux coefficients de flexibilité. Ces simulations permettent alors d'obtenir les coefficients de raideur de la matrice de flexibilité.

La MSA reprend les idées principales de l'EF, néanmoins les éléments finis du système sont de taille plus importante que dans l'EF et cette méthode inclut des poutres flexibles pour décrire la structure du manipulateur (Ahmad et al., 2011).

- **Flexibilités localisées**

La VJM est basée sur le calcul de la matrice Jacobienne par rapport au déplacement des liaisons, au TCP et aux directions dans l'espace cartésien (Ahmad et al., 2011). Cette méthode considère une série de corps rigides qui sont séparés par des liaisons flexibles (Ahmad et al., 2011;

Klimchik et al., 2012) et des ressorts. Ces ressorts décrivent l'élasticité des corps et des moteurs (Klimchik et al., 2012).

Alici, Gallot, Li, Dumas, Klimchik, Pashkevich, (Alici & Shirinzadeh, 2005; Pashkevich et al., 2009; Dumas; et al., 2010; Dumas et al., 2011; Ming et al., 2011; Gallot et al., 2012; Klimchik et al., 2012) considèrent dans leurs travaux la méthode VJM. Quelques auteurs comme Spong, Rodriguez, Ozgoli et Ulrich (Mark W. Spong, 1989; Ozgoli & Taghirad, 2004; Rodriguez-Angeles et al., 2004; Ozgoli & Taghirad, 2006; Ulrich & Sasiadek, 2011) utilisent la terminologie liaison flexible du robot (FJR-flexible joint robots) pour définir une méthode similaire à la VJM.

Le modèle de raideur, qu'ils présentent, relie les efforts exercés au bout de l'effecteur et les déplacements du robot. Ce modèle est représenté par (3.20) (Tsai, 1999),

$$F = -K\Delta x \quad (3.20)$$

tel que :

F soit le vecteur 6×1 des efforts et des moments exercés sur l'effecteur du robot,

Δx soit le vecteur 6×1 des déplacements linéaires et angulaires de l'effecteur,

K soit la matrice 6×6 de raideur cartésienne du robot.

La matrice de raideur dépend de la configuration du robot, des raideurs de chaque articulation et des couples exercés sur l'effecteur. Cette matrice est calculée par :

$$K = J^{-T}(K_\theta - K_c)J^{-1} \quad (3.21)$$

Telle que :

K_θ soit une matrice diagonale qui est constituée des six raideurs articulaires

$$K_\theta = \begin{bmatrix} K_{\theta_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & K_{\theta_6} \end{bmatrix}_{6 \times 6},$$

K_c soit la matrice de raideur complémentaire. Cette matrice permet de prendre en compte

l'influence des efforts sur la raideur du robot $K_c = \left[\frac{\partial J^T}{\partial \theta_1} F \cdots \frac{\partial J^T}{\partial \theta_6} F \right]_{6 \times 6}$,

J soit la matrice Jacobienne du robot.

3.2 Flexibilités de l'outil

Les flexibilités dans l'outil modifient la trajectoire de commande du robot. Afin d'éviter, les imprécisions qui sont généralement dues aux variations entre la position de l'outil et la pièce (Lartigue et al., 2003; Shaopeng, 2010), la trajectoire du robot doit être modifiée selon les flexibilités. Ces changements doivent être réalisés rapidement et facilement sans modifier la

trajectoire de la tâche (Schrimpf, 2013). Afin d'éviter le réapprentissage de tous les points de la trajectoire du robot, les programmeurs identifient les points sur lesquels il y a des variations de position et ensuite ils modifient ces points pour compenser les écarts (Shaopeng, 2010). Différentes méthodes sont implémentées pour mesurer les variations entre l'outil et la pièce (Schrimpf, 2013), pour modifier la trajectoire automatiquement. Cette modification de la trajectoire du robot doit considérer les éléments en interaction (le robot, l'environnement et la pièce).

Généralement dans les applications industrielles, les composants du robot et de l'effecteur sont définis comme des éléments rigides (Lartigue et al., 2003). Dans ce cas, il est logique que le TCP soit fixé à une seule position/orientation, car la géométrie de l'outil est figée. Par exemple dans le procédé de « pick-place » (figure 3.5), le TCP est en principe fixé au centre des mors de la pince.

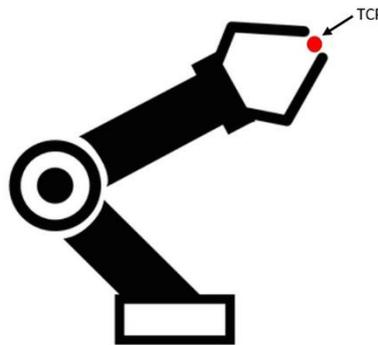


Figure 3.5 TCP fixe à un point

Dans les applications industrielles pour lesquelles les composants du robot et/ou de l'effecteur sont considérés comme des éléments flexibles, le TCP doit changer de position/orientation selon les flexibilités du robot et/ou de l'outil.

3.2.1 Outil rigide

- *Point de centrage de l'outil*

La précision du robot dépend de ses propriétés géométriques, de la stabilité du mouvement dynamique et du TCP (Bossoni, 2009; Shaopeng, 2010). Le TCP permet de connaître la position et l'orientation de l'outil pendant le mouvement du robot (Toffano, 2015). Afin de connaître la position précise du TCP, il faut procéder à une opération de calibrage qui consiste à définir un point dans l'espace cartésien. Ce point permet de connaître la position/orientation de l'outil.

Généralement, le TCP des robots manipulateurs sans outil est défini au centre du poignet du robot dans le repère de la base de montage « flange », comme le montre la figure 3.6.

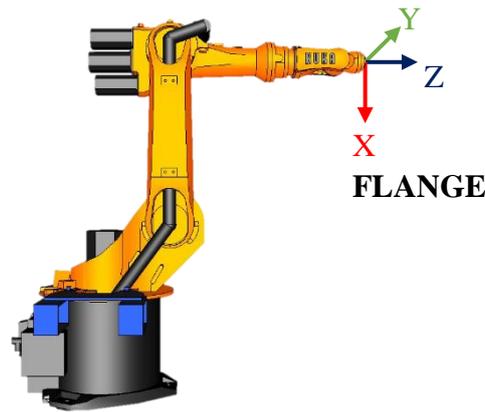


Figure 3.6 Robot KUKA KR6 avec les cordonnées World et le TCP

Avant de réaliser une tâche, le TCP doit donc être calibré et défini dans le contrôle du robot pour chaque outil (Kostal, 2014) selon ses caractéristiques géométriques (Chanal et al., 2007). Jusqu'à maintenant, le calibrage du TCP des robots manipulateurs est effectué manuellement (Hallenberg, 2007). Par conséquent un bon calibrage dépend des compétences du programmeur du robot. Le TCP est à l'origine du système de référence qui représente la position et l'orientation de l'outil (Toffano, 2015) par rapport au système de coordonnées de la base de montage.

Afin de déplacer le robot selon l'outil attaché et de maintenir une dimension fixe entre l'outil et la surface de travail, le TCP doit être calibré et défini selon le système de coordonnées de l'outil (Shaopeng, 2010; Schrimpf, 2013). La formulation mathématique de la position et l'orientation est représentée par la matrice de passage (3.3) (Shaopeng, 2010).

A titre d'exemple de la modification du TCP selon la géométrie de l'outil, la figure 3.7 montre un outil pour soudage attaché sur le poignet du robot (système de coordonnées de la base de montage) : le TCP original est positionné sur le poignet du robot sans rotation et sans translation, par contre, le TCP de l'outil de soudage a une translation dans l'axe Z et une rotation dans l'axe Y par rapport au TCP original.

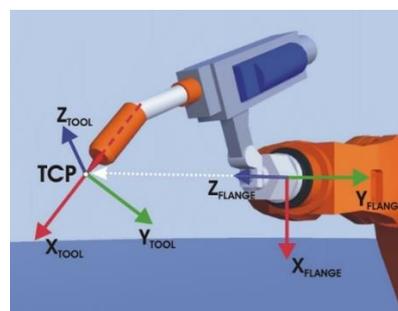


Figure 3.7 L'outil pour soudage attaché au poignet du robot

Le calibrage du TCP pour les bras manipulateur KUKA est généralement effectué en deux étapes (KUKA, 2012) :

- a) La première étape consiste à définir l'origine du système de coordonnées de l'outil. Cette définition peut être réalisée suivant deux méthodes :
- « XYZ 4-point »,
 - « XYZ référencée ».
- b) La deuxième étape consiste à définir l'orientation du système de coordonnées de l'outil. Cette orientation peut être réalisée suivant deux méthodes (KUKA, 2012).
- « ABC 2-point »,
 - « ABC world ».
- La méthode « XYZ 4-point » consiste à déplacer l'outil par rapport à une référence dans quatre directions différentes. Cette référence peut être un outil spécialement conçu ou libre. Le module de commande du robot calcule le TCP à partir des différentes positions de la base de montage (Figure 3.8).

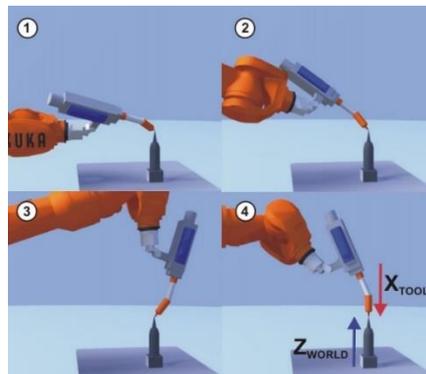


Figure 3.8 Méthode XYZ 4- points (KUKA 2012)

- La méthode « XYZ référencée » consiste à calibrer un nouvel outil à partir d'un outil antérieurement calibré. Le module de commande du robot compare les positions de la base de montage et calcule le nouveau TCP.
- La méthode « ABC 2-point » consiste à déplacer l'outil vers un point dans l'axe X et vers un point dans le plan XY et à envoyer à la baie de commande du robot le système de coordonnées.
- La méthode « ABC-world » consiste à aligner parallèlement les axes du système de coordonnées de l'outil et les axes du système de coordonnées WORLD. L'information de l'orientation du système des coordonnées de l'outil est transmise à la baie de commande du robot.
- Les axes du système de coordonnées TOOL sont alignés parallèlement aux axes du système de coordonnées WORLD. Ceci communique l'orientation du système de coordonnées TOOL à la baie de commande du robot.

Une autre façon de faire le calibrage du TCP est d'introduire numériquement les dimensions et les orientations de l'outil. Cette information est extraite de la CAO, des spécifications des fabricants ou du calibrage externe de l'outil (KUKA, 2012). Ainsi, le calibrage est effectué à partir de la définition des moments d'inertie, du centre de masse et de la masse de l'outil.

Dans le travail exposé par (Khalil & Lemoine, 1999), les auteurs développent GECARO « GEometric CALibration of Robots » un logiciel pour la simulation et la réalisation de l'étalonnage des paramètres géométriques des robots. Ce logiciel permet de réaliser le calibrage par méthodes classiques qui requièrent des capteurs externes pour mesurer la position/orientation de l'effecteur et par méthodes autonomes qui utilisent seulement des capteurs de position. Ces méthodes autonomes sont basées sur la réalisation de liaisons mécaniques (point-point, repère-repère et point-plan. Les paramètres identifiables pour chaque méthode sont déterminés par une méthode numérique fondée sur la décomposition QR de la matrice d'observation. L'algorithme d'identification utilise une méthode itérative de moindres carrés linéaire qui met à jour les paramètres géométriques après chaque itération.

3.2.2 Outil flexible

Afin de considérer les flexibilités de l'outil et modifier le TCP par rapport à l'outil, il est important d'étudier le comportement de l'outil. Dans le cas appliqué dans cette recherche et décrit dans le paragraphe 2.4, les flexibilités de l'outil, dues à la force de compactage qui lui est appliquée, seront prises en compte. Ce paragraphe est consacré à étudier le comportement du rouleau de compactage de la machine de dépose de fibre robotisée et à élaborer un modèle de l'outil qui permet de considérer la relation force-déformation.

- *Approche par éléments finis*

La méthode des éléments finis (EF) est une méthode de discrétisation des problèmes continus qui trouve une solution approchée pour simplifier le modèle (Szabo & Babuška, 1991; Cook, 2001; Hutton, 2003), de telle façon que :

- La structure continue est divisée en un nombre fini des morceaux « éléments », dont le comportement est spécifié par un nombre fini des paramètres associés à des points caractéristiques « nœuds ». Ces nœuds sont les points d'union de chaque élément avec les éléments adjacents (figure 3.9) (Hutton, 2003; Valero, 2004; Fischer, 2010).
- Les règles aux problèmes discrets doivent être appliquées à toute la solution, car le modèle est un assemblage des éléments et la solution du problème est donnée par les déplacements de ces nœuds. Ces déplacements permettent de trouver les contraintes

principales, les contraintes de cisaillement, la déformation, ... du système (Valero, 2004; Pezzotti & Antic, 2008).

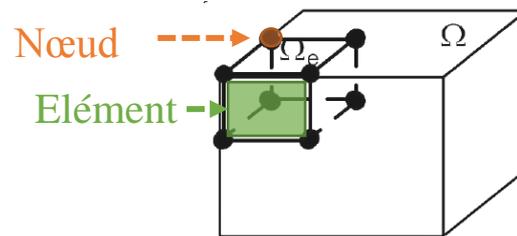


Figure 3.9 Composants des éléments finis

- Le système continu (de dimension infinie) modélisé par des équations aux dérivées partielles devient un système avec un nombre fini de degrés de liberté et modélisé par des équations linéaires ou non linéaires (Szabo & Babuška, 1991; Cook, 2001; Fischer, 2010). C'est-à-dire le MEF transforme un corps continu en un modèle discret approché.
- Les fonctions et les inconnues du système deviennent les fonctions et les inconnues des nœuds (Hutton, 2003; Valero, 2004).
- Le comportement de chaque élément est défini par le comportement des nœuds à partir des fonctions d'interpolation (Cook, 2001; Hutton, 2003; Valero, 2004). C'est-à-dire la connaissance à l'intérieur du modèle discret approché est obtenue par l'interpolation des valeurs sur les nœuds.

La méthode EF approche des valeurs d'une fonction à partir de la connaissance d'un nombre défini et fini des points (Hutton, 2003; Valero, 2004).

Les propriétés des éléments associés aux nœuds, sont données par le matériau assigné à la géométrie. La rigidité des éléments est définie par le matériau assigné. Les déplacements et les forces externes sont liés par la rigidité, le nombre des éléments et nœuds de la géométrie (Pezzotti & Antic, 2008).

Le principe de base est d'approcher la solution par des fonctions de formes (Huebner, 2001; Moharos et al., 2012). Ces fonctions de forme représentent l'assemblage de systèmes complexes. La méthode EF plus utilisée dans les logiciels est basée sur la méthode de déplacement (Bathe, 2006; Moharos et al., 2012). La procédure de discrétisation divise la géométrie en un nombre fini des éléments. Cette division génère un nombre fini d'inconnues qui sont représentées en fonctions d'approximation (Huebner, 2001). La solution de ces fonctions d'approximation satisfait les équations d'équilibre. La méthode de déplacement peut être représentée par un ressort (figure 3.10).

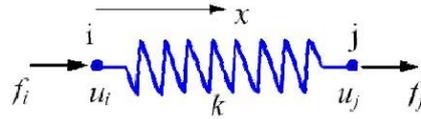


Figure 3.10 Système ressort

Tel que :

- x soit l'axe de coordonnées locale,
- i et j soient les nœuds,
- u_i et u_j soient les déplacements des nœuds (degré de liberté),
- f_i et f_j soient les forces internes,
- k soit la rigidité de l'élément.

Afin de prendre en compte les forces internes du ressort en fonction des déplacements des nœuds et la rigidité de l'élément, la contribution des ressorts est :

$$\begin{aligned} f_i &= -K_e(u_i - u_j) \\ f_j &= -K_e(u_j - u_i) \end{aligned} \quad (3.22)$$

La représentation sous forme matricielle est formulée par :

$$\begin{bmatrix} f_i \\ f_j \end{bmatrix} = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \begin{bmatrix} u_i \\ u_j \end{bmatrix} \quad (3.23)$$

Pour chaque ressort ajouté au système, les dimensions de la matrice augmentent. Par exemple, pour un système de deux ressorts la représentation matricielle est définie par :

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (3.24)$$

La méthode EF considère que la matrice de rigidité tient compte des propriétés du matériau et la géométrie. Cette matrice est représentée par :

$$k = \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (3.25)$$

Tel que :

- E soit le module de Young,
- A soit la surface transversale,
- L soit la longueur de l'élément.

La matrice de rigidité est substituée dans la forme matricielle du ressort (3.23). La représentation mathématique du ressort est réécrite et représentée par :

$$\begin{bmatrix} f_i \\ f_j \end{bmatrix} = \begin{bmatrix} \frac{EA}{L} & -\frac{EA}{L} \\ -\frac{EA}{L} & \frac{EA}{L} \end{bmatrix} \begin{bmatrix} u_i \\ u_j \end{bmatrix} \quad (3.26)$$

Pour un élément en deux dimensions, la position des éléments doit être considérée. Pourtant pour un élément à trois dimensions, le moment d'inertie (I), le module de Young (E), la surface (A) et la longueur d'élément (L) doivent être considérés.

- **Modélisation des matériaux hyper-élastique**

Les lois de comportement hyper-élastiques sont utilisées pour modéliser la réponse élastique des matériaux lorsqu'ils sont soumis à des grandes déformations. Ces lois sont applicables à des matériaux non-linéaires et avec des grands changements de forme (Bower, 2009). Les applications de cette théorie sont :

- Modéliser le comportement plastique d'un matériau polymère.
- Modéliser les matériaux de mousses de polymère.

Dans le cas de cette recherche, le rouleau de compactage monté sur le robot chez CompositAdour est composé en grande partie d'un matériau Sylomer type P (paragraphe 2.4). Ce matériau est un élastomère du type polyuréthane, donc la modélisation de son comportement plastique est similaire à celui d'un matériau polymère.

Les caractéristiques du comportement du polymère (caoutchoucs) solide sont :

- Un comportement proche de celui des matériaux élastiques idéaux, car l'effort est déterminé par la déformation actuelle et la vitesse de chargement, indépendamment de la déformation historique (Bower, 2009; Jakel, 2010).
- Un comportement proche de celui des matériaux élastiques idéaux lorsqu'ils sont déformés à une température constante (Bower, 2009; Jakel, 2010).
- Un comportement élastique, car ils reviennent à leur forme originale lorsque les forces ne sont plus exercées (Jakel, 2010).
- Une résistance aux variations de leur volume, car le module d'élasticité isostatique est comparable à celui des métaux (Bower, 2009).
- La contrainte de cisaillement est très élevée et dépend de la température. Le matériau devient plus rigide quand il est chauffé (Bower, 2009).
- Le matériau hyper-élastique est isotrope (Bower, 2009). La relation effort-déformation provient d'une fonction de densité d'énergie de déformation (Jakel, 2010).
- Une émission de chaleur quand ils sont étirés (Bower, 2009).

Les élastomères généralement subissent de grandes déformations avec des petites charges. Ces matériaux sont près de l'incompressibilité, alors leur coefficient de Poisson est proche de 0,5. Les courbes effort-déformation dépendent de facteurs comme : le temps, la charge statique ou dynamique, la fréquence dynamique... (Jakel, 2010).

Les modèles hyper-élastiques sont définis à partir de :

- La contrainte de tension-déformation définie par la densité d'énergie de déformation en fonction de la déformation du gradient de tenseur.
- Le matériau est généralement supposé isotrope.
- La contrainte de tension en fonction de la déformation est calculée par le différentiel de la densité d'énergie de déformation.

La formulation mathématique des modèles hyper-élastiques utilisés par le logiciel d'éléments finis ANSYS est présentée dans le tableau 3.3.

Tableau 3.3 Formulation mathématique des modèles hyper-élastiques (ANSYS)

Neo-Hookean (Bower, 2009; Jakel, 2010; ANSYS, 2015)	$W = \frac{\mu}{2}(\bar{I}_1 - 3) + \frac{1}{d}(J - 1)^2 \quad (3.27)$
Mooney-Rivlin (Bower, 2009; Jakel, 2010; ANSYS, 2015)	$W = C_{10}(\bar{I}_1 - 3) + C_{01}(\bar{I}_2 - 3) + \frac{1}{d_{MR}}(J - 1)^2 \quad (3.28)$
Polynomial (Bower, 2009; Jakel, 2010; ANSYS, 2015; Rackl, 2015)	$W = \sum_{i+j-1}^N C_{ij}(\bar{I}_1 - 3)^i(\bar{I}_2 - 3)^j + \sum_{k-1}^N \frac{1}{d_k}(J - 1)^{2k} \quad (3.29)$
Gent (ANSYS, 2015)	$W = -\frac{\mu J_m}{2} \ln\left(1 - \frac{\bar{I}_1 - 3}{J_m}\right) + \frac{1}{d}\left(\frac{J^2 - 1}{2} - \ln J\right) \quad (3.30)$
Arunda-Boyce (Bower, 2009; ANSYS, 2015)	$W = \mu \left[\frac{1}{2}(\bar{I}_1 - 3) + \frac{1}{20\lambda_L^2}(\bar{I}_1^2 - 9) + \frac{11}{1050\lambda_L^4}(\bar{I}_1^3 - 27) + \frac{19}{7050\lambda_L^6}(\bar{I}_1^4 - 81) + \frac{519}{673750\lambda_L^8}(\bar{I}_1^5 - 243) \right] + \frac{1}{d}\left(\frac{J^2 - 1}{2} - \ln J\right) \quad (3.31)$
Odgen (Bower, 2009; ANSYS, 2015; Rackl, 2015)	$W = \sum_{i-1}^N \frac{\mu_0}{\alpha_i} (\bar{\lambda}_1^{\alpha_i} + \bar{\lambda}_2^{\alpha_i} + \bar{\lambda}_3^{\alpha_i} - 3) + \sum_{k-1}^N \frac{1}{d_k}(J - 1)^{2k} \quad (3.32)$
Yeoh (Jakel, 2010; ANSYS, 2015; Rackl, 2015)	$W = \sum_{i-1}^N C_{i0} (I_1 - 3)^i + \sum_{k-1}^N \frac{1}{d_k}(J - 1)^{2k} \quad (3.33)$

Telle que:

W soit l'énergie potentielle de la déformation par unité de volume,

\bar{I}_1 soit le premier invariant du tenseur des déformations,

\bar{I}_2 soit le deuxième invariant du tenseur des déformations,

J soit le gradient de déformation élastique F ,

μ soit le module de rigidité $\mu = \frac{E}{2(1+\nu)}$,

d soit le paramètre d'incompressibilité $d = \frac{2}{k}$,

k soit le module d'élasticité isostatique $k = \frac{E}{3(1-2\nu)}$,

ν soit le coefficient de Poisson,

E soit le module d'Young,

C_{10} C_{01} soient des constantes du matériau qui caractérisent les tenseurs de la déformation (éléments de la déformation du tenseur Cauchy-Green),

μ_{MRP} soit le module de rigidité $\mu_{MRP} = 2(C_{10} + C_{01})$,

d_{MR} soit le paramètre d'incompressibilité $d_{MR} = \frac{2}{k}$ ou $d_{MR} = \frac{(1-2\nu)}{(C_{10}+C_{01})}$,

J_m soit la limite de valeur de $\bar{I}_1 - 3$,

λ_L soit la limite maximale de déformation (premier coefficient de Lamé),

λ_p ($p = 1,2,3$) soit le tenseur principal $\bar{\lambda}_p = \frac{\lambda_p}{J^{1/3}}$,

λ_p soit le tenseur gauche de Cauchy-Green,

$N, \mu_i, \alpha_i, d_k, C_{i0}$ soient des constantes, caractéristiques du matériau,

μ_0 soit le module de rigidité $\mu_0 = \frac{1}{2} \sum_{i=1}^N \alpha_i \mu_i$,

μ_Y soit le module de rigidité $\mu_Y = 2C_{i0}$.

Si le modèle polynomial répond aux conditions :

$N = 1$ et $C_{01} = 0$, le modèle est le même que le modèle Neo-Hookean,

$N = 1$, le modèle est le même que le modèle Mooney-Rivlin à deux paramètres,

$N = 2$, le modèle est le même que le modèle Mooney-Rivlin à cinq paramètres,

$N = 3$, le modèle est le même que le modèle Mooney-Rivlin à neuf paramètres.

Si le modèle Gent répond à la condition :

J_m est proche de l'infini, le modèle est le même que le modèle Neo-Hookean.

Si le modèle Arunda-Boyce du tableau 3.3 répond à la condition :

λ_L est proche de l'infini, le modèle est le même que le modèle Neo-Hookean.

Si le modèle Odgen répond aux conditions :

$N=1$, $\alpha_1 = 2$ le modèle est le même que le modèle Neo-Hookean,

$N=2$, $\alpha_1 = 2$ le modèle est le même que le modèle Mooney-Rivlin.

Si le modèle Yeoh répond à la condition :

$N=1$, le modèle est le même que le modèle Neo-Hookean.

3.3 Modélisation des flexibilités de l'outil

Afin de considérer les flexibilités de l'outil et modifier le TCP, l'étude du comportement de l'outil est très importante, car ce comportement génère d'importantes modifications dans la trajectoire du robot pendant la dépose de fibres. Pour le cas pratique de cette recherche, décrit dans les paragraphes 2.4 et 3.2, les flexibilités de l'outil sont dues à la force de compactage appliquée sur l'outil et sont observées principalement sur le rouleau de compactage. Pour mettre en évidence l'élasticité du rouleau de compactage, la figure 3.11a) montre le rouleau de compactage sans aucune force et la figure 3.11b) montre la déformation du rouleau de compactage soumis à la force exercée par une main.

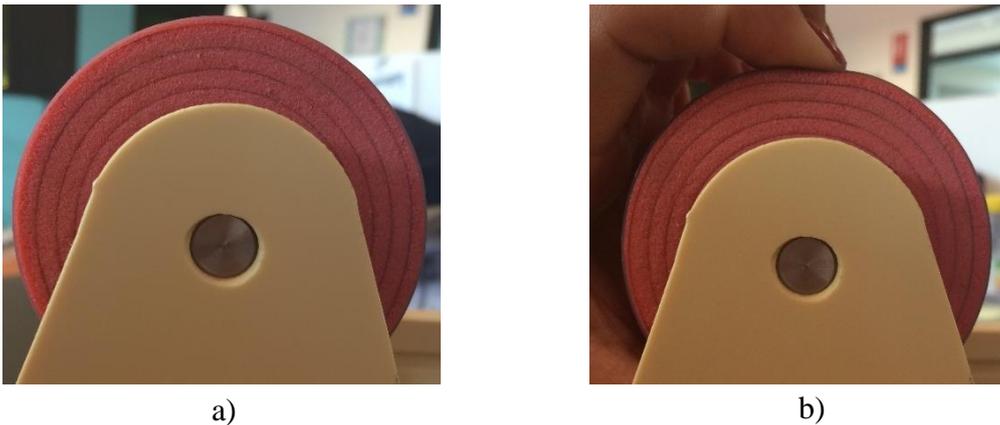


Figure 3.11 Rouleau de compactage a) sans aucune force b) avec la force exercée par une main

3.3.1 Modèle expérimental

Les éléments utilisés pour le modèle expérimental sont le rouleau de compactage et la surface de contact avec les caractéristiques géométriques suivantes :

- Le rouleau de compactage monté sur le robot de CompositAdour, est représenté par un cylindre de diamètre extérieur 0,068 m, un diamètre intérieur 0,03 m et de longueur 0,118 m.
- L'axe de rotation d'un diamètre 0,03 m et de longueur 0,16 m.
- La surface de contact (moule) est représentée par une plaque plane de longueur 0,18 m, de largeur 0,1 m et d'épaisseur 0,05 m.

Les éléments utilisés possèdent les caractéristiques mécaniques de matériau suivantes :

- Le rouleau de compactage est composé de couches superposées en Sylomer type P, de colle entre chaque couches et d'une gaine extérieure. Comme nous l'avons dit dans le paragraphe 2.4, les caractéristiques mécaniques du matériau du rouleau de compactage par les différentes couches sont inconnues. Pour réaliser le calcul du modèle numérique, le rouleau de compactage est considéré comme un élément composé par un matériau homogène. Ce matériau est le Sylomer du type P qui est un matériau hyper-élastique. Il peut prendre différentes valeurs de module d'Young selon la pression appliquée (W. Getzner, 2004) et différents coefficients de Poisson compris entre 0,3 et 0,5 (W. G. Getzner, 2011). Le module d'Young de la pression statique limite considéré est de 1,44 MPa et le coefficient de Poisson considéré est de 0,499.
- L'axe et le moule sont en acier. Ce matériau a un module d'Young de 200 000 MPa et un coefficient de Poisson de 0,3.
- ***Machine tension/compression***

Pour définir les flexibilités réelles du rouleau de compactage, nous réalisons des tests en compression du rouleau de compactage sur une machine à traction/compression. Dans le cas appliqué dans cette recherche et décrit dans le paragraphe 2.4, la force de compactage appliquée dans le procédé robotisé pour déposer les fibres est comprise entre 5 N et 1200 N. En raison des caractéristiques du capteur d'effort monté à l'intérieur de la machine à traction/compression, la force maximale à laquelle le rouleau de compactage est soumis est de 1000 N. La figure 3.12 montre le système utilisé pour cette expérimentation qui est composé d'une plaque plane en acier (moule) montée au-dessus d'une mâchoire mobile (mors), d'un capteur d'effort, d'un rouleau de compactage et d'un télémètre laser monté sur le mors fixe.

Le test en compression est effectué avec des incréments de déplacement de $1.667e-5$ m/sec, ce qui représente la vitesse minimale de la machine de traction/compression. La fréquence d'acquisition des capteurs (force et télémètre laser) est de 20 Hz.

Le test consiste à déplacer la mâchoire mobile, afin de comprimer le rouleau de compactage et de créer un effort. Cet effort est mesuré par le capteur d'effort. En même temps, le télémètre laser mesure la distance qui le sépare de la plaque d'acier, pour obtenir les valeurs de la déformation du rouleau de compactage selon la force appliquée. La figure 3.13 montre la déformation du rouleau lorsqu'il est soumis à une force de 1000 N.

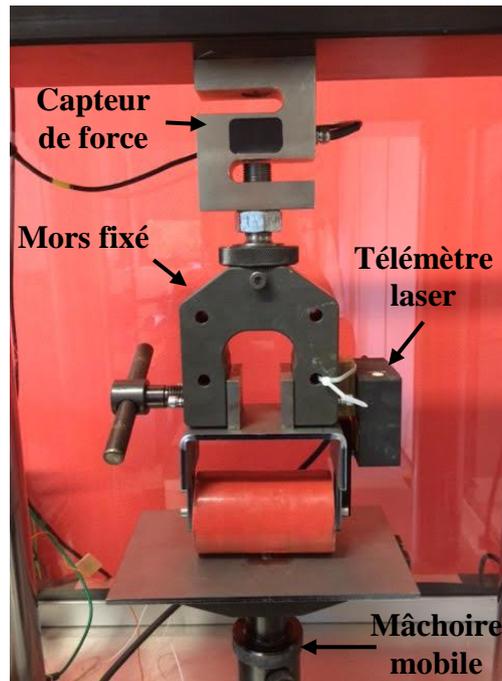


Figure 3.12 Système employé pour les tests de compression



Figure 3.13 Rouleau soumis à une force de 1000 N

Cette expérimentation permet d'obtenir les valeurs d'effort-déformation en compression le long d'un axe pour le rouleau de compactage avec les caractéristiques décrites dans le paragraphe 3.3.1. L'expérimentation consiste en quatre tests, afin de réaliser une comparaison entre eux pour définir le comportement du rouleau. Les résultats obtenus dans cette expérimentation sont représentés dans la figure 3.14. Nous pouvons constater que le comportement du rouleau est similaire dans les quatre tests, l'erreur entre eux variant entre 0,5 % et 1 %. Le déplacement du rouleau de compactage à 1000 N est d'environ de 0,0047 m.

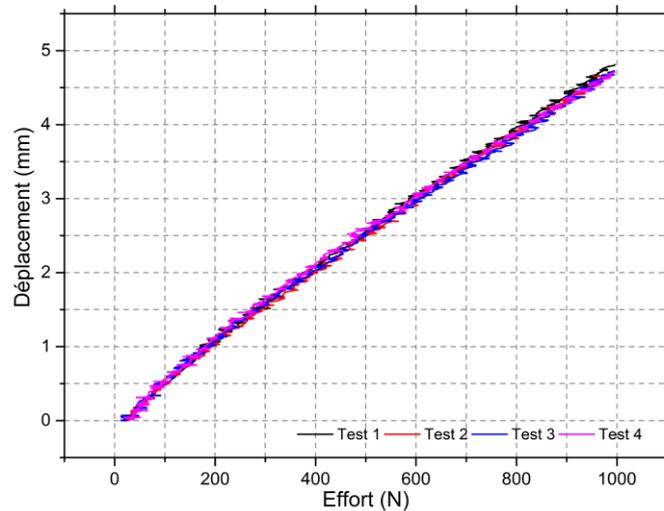


Figure 3.14 Résultats obtenus pour les valeurs d'effort-déformation en compression le long d'un axe grâce à la machine à compression

3.3.2 Modèle numérique par éléments finis

Afin de reproduire le comportement réel du rouleau de compactage et d'établir un modèle de ce comportement exploitable numériquement, nous considérons un modèle numérique par éléments finis (EF). Ce modèle doit prendre en compte le comportement du rouleau de compactage en contact avec une surface, ainsi que le comportement de son matériau supposé hyper-élastique. Le paragraphe 3.2.2 décrit les lois de comportement hyper-élastique utilisées pour modéliser la réponse élastique des matériaux lorsqu'ils sont soumis à de grandes déformations, comme dans notre cas d'étude. Le tableau 3.3 présente différents modèles mathématiques pour l'analyse des matériaux hyper-élastiques.

Dans (J. Li et al., 2013), l'auteur analyse un rouleau de compactage utilisé pour le placement de fibres. Ce rouleau est composé d'un matériau du type caoutchouc considéré comme un élément hyper-élastique. L'auteur réalise l'analyse par EF avec le logiciel ABAQUS et utilise le modèle mathématique de Mooney-Rivlin (équation (3.28)). Les paramètres mécaniques du matériau (coefficient de Poisson et module d'élasticité) sont définis grâce à un test uni-axial de force-déformation réalisé avec une machine à tension/compression. Les paramètres mécaniques du matériau utilisé sont : le coefficient de Poisson de 0.475 et le module d'élasticité de 2.22 MPa. La force maximale appliquée est de 400 N avec des intervalles de 50 N chacun. Les résultats obtenus par EF sont vérifiés avec les résultats obtenus par l'expérimentation, l'erreur entre les deux résultats étant comprise entre 3,88% et 12.96%.

L'exactitude du modèle numérique dépend principalement des composants géométriques de la pièce, des propriétés mécaniques, de la forme et de la taille du maillage et de la méthode de résolution (interactive ou directe). Afin d'établir un modèle numérique plus proche de la réalité

et avec une bonne précision, nous réalisons un modèle avec des composants qui respectent les conditions des équations de Hertz (paragraphe A.2) et nous réalisons une comparaison entre le modèle numérique EF et le modèle théorique calculé selon le modèle de Hertz. Dans le cas d'étude de cette recherche, le rouleau de compactage est considéré comme un cylindre et la surface de contact est considérée comme plane, car la force appliquée doit être normale à la surface, donc nous prenons en compte les équations cylindre-plan de Hertz (paragraphe A.2.1). Les résultats obtenus pour ces deux modèles sont exposés dans le paragraphe A.2.2. Ils nous permettent de valider le modèle EF. Malgré cela, le type de matériau du rouleau de compactage ne peut pas être modélisé par les équations de Hertz, car il présente de grandes flexibilités.

Pour l'analyse par EF de notre cas d'étude, les caractéristiques géométriques et les caractéristiques mécaniques du matériau sont considérées pour le rouleau de compactage décrit dans le paragraphe 3.3.1. Le modèle EF validé et le modèle pour le rouleau de compactage considèrent le même type de maillage. La surface plane dans la zone de contact est découpée avec les mêmes nombre de couches et la réduction de la taille de maillage. La figure 3.15 montre le maillage de forme majoritairement quadrilatère utilisé pour le rouleau de compactage. Nous utilisons la fonction remaillage automatique de « workbench ». La taille de maillage est de 0,004 m pour tous les composants et la surface plane est découpée en sept couches.

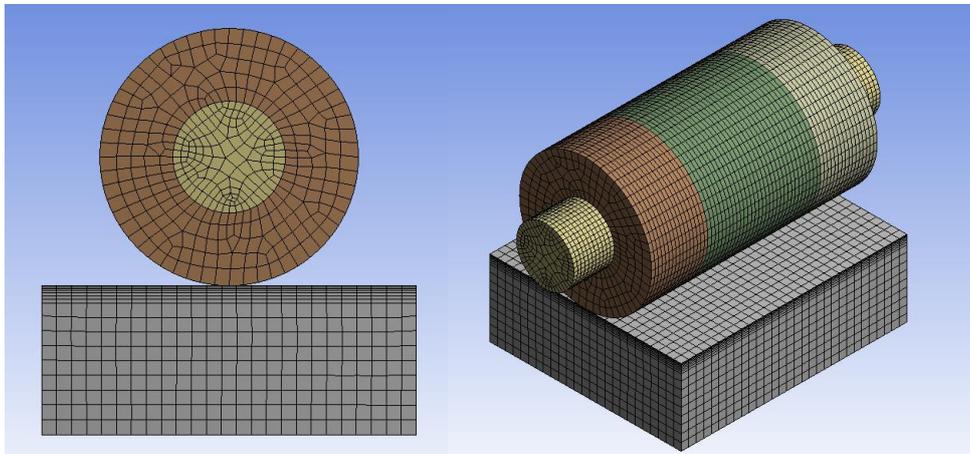


Figure 3.15 Maillage du modèle EF pour rouleau de compactage

Afin de vérifier les résultats entre les modèles hyper-élastiques et les résultats obtenus par la machine à traction/compression, ainsi que la prise en compte du modèle hyper-élastique plus performant, nous sélectionnons les méthodes de Neo-Hookean, Ogden et Yeoh, parmi les méthodes hyper-élastiques exposées dans le tableau 3.3.

La méthode de Neo-Hookean (équation (3.27)) est calculée à partir de deux constantes : la constante d'incompressibilité et le module de rigidité initial.

La méthode d'Ogden (équation (3.32)) est calculée à partir de trois constantes : la constante d'incompressibilité et les constantes du matériau (μ_i et α_i) définies par le module de rigidité.

La méthode de Yeoh (équation (3.33)) est calculée à partir de deux constantes : la constante d'incompressibilité et la constante du matériau (C_{10}) définie par le module de rigidité.

La valeur de la constante d'incompressibilité est proche à zéro, car le coefficient de Poisson est de 0,4999. Cette constante est la même pour les trois méthodes. Les autres constantes sont calculées pour chaque méthode. Pour la méthode de Neo-Hookean le module de rigidité initiale est de 0,48 MPa, pour la méthode d'Ogden la constante α_i est fixée à un (paramètre par défaut) et la constante μ_i du matériau est de 0,96 MPa et la constante du matériau est de 0,24 MPa pour la méthode de Yeoh

La figure 3.16 montre les résultats obtenus par les trois modèles hyper-élastiques en comparaison au modèle réel (modèle obtenu par la machine à tension/compression) pour un module d'élasticité de 1,44 MPa.

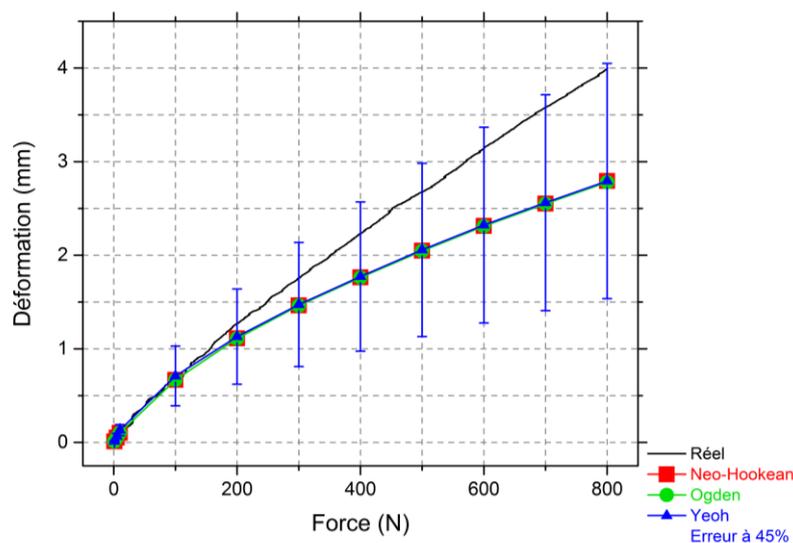


Figure 3.16 Résultats du modèle hyper-élastique avec un module d'élasticité de 1,44 MPa

Les trois modèles sont équivalents, car les constantes de chaque modèle dépendent du module d'élasticité et le coefficient de Poisson. Toutefois ces deux paramètres évoluent au cours du temps pour le matériau considéré. Cependant ils restent constantes dans la limite de charge statique. En conséquence, les trois modèles ont la même convergence et produisent des erreurs d'environ 45% par rapport au modèle réel. Afin de réduire l'erreur, nous devons effectuer un recalage du modèle.

3.3.3 Recalage du modèle numérique

Afin d'obtenir les paramètres mécaniques du matériau les plus proches de la réalité et de réduire l'erreur du modèle, nous devons recalibrer le modèle EF en modifiant les paramètres mécaniques.

Pour déterminer la plage de modification, nous utilisons un module d'élasticité dont la force est nulle. Selon la « data sheet », du Sylomer pur du type P, ce module d'élasticité est de 2,84 MPa. Donc l'intervalle initial de modification du module d'élasticité est compris entre 1.44 MPa et 2.84 MPa.

A partir de la méthode de dichotomie, il est possible de définir une valeur du module d'élasticité en suivant les résultats réels. La méthode de dichotomie consiste à diviser un intervalle en deux jusqu'à trouver la solution. Les conditions pour appliquer cette méthode sont les suivantes :

- Deux nombres réels a et b (intervalle [a, b]).
- Une fonction réelle continue (f).
- $f(a)f(b) \leq 0$.

La méthode de dichotomie cherche à trouver une solution approchée de la fonction ($f(x)=0$), cette méthode affirmant qu'il existe au moins une solution de l'équation ($f(x)=0$) dans l'intervalle [a, b]. Alors, la méthode de dichotomie évalue la fonction dans la valeur intermédiaire ($m = \frac{a+b}{2}$) de l'intervalle [a, b] et deux solutions sont envisageables :

- Si $f(a)f(m) \leq 0$, la solution $f(x)=0$ se trouve dans l'intervalle [a, m]
- Si $f(m)f(b) > 0$, la solution $f(x)=0$ se trouve dans l'intervalle [m, b]

Cette méthode de dichotomie est dite récursive. Le processus d'évaluation s'arrête lorsque l'erreur de la méthode (erreur = $\frac{b-a}{2}$) est inférieure à la précision souhaitée.

La figure 3.17a et la figure 3.17b montrent les résultats obtenus par le modèle EF dans l'intervalle initial (module d'élasticité de 1,44 MPa et 2,84 MPa respectivement). La solution trouvée pour la valeur de 2,84 MPa est éloignée de la fonction réelle. Donc la solution ne se trouve pas dans cet intervalle. Afin de définir un nouvel intervalle, la valeur 1,44 MPa est divisée à la moitié de sa valeur pour définir la limite inférieure, alors le nouvel intervalle prend les valeurs comprises entre 0,72 MPa et 1,44 MPa. La figure 3.17c montre la solution trouvée pour la valeur de 0,72 MPa. Cette solution s'approche de la fonction réelle, nous pouvons donc conclure que la solution se trouve dans cet intervalle. La meilleure solution approchée (obtenue après 5 interactions) par la méthode de dichotomie est pour un module d'élasticité de 1,26 MPa (figure 3.17d).

La solution obtenue pour un module d'élasticité de 1,26 MPa est la même pour tous les modèles, par contre le modèle d'Ogden dispose d'une deuxième constante qui peut être modifiée pour obtenir un modèle plus proche du modèle réel. Dans le premier recalage, la constante d'incompressibilité est égale à zéro, la constante α_i est fixée à un (paramètre par défaut) et la constante μ_i est de 0,96 MPa. Comme nous pouvons le constater dans la figure 3.17d, l'erreur

entre le modèle réel et le modèle d'Odgen est réduite, cette erreur représente le 20%. Afin de réduire cette erreur, nous réalisons un recalage du modèle d'Odgen en modifiant la constante α_i par la méthode de dichotomie.

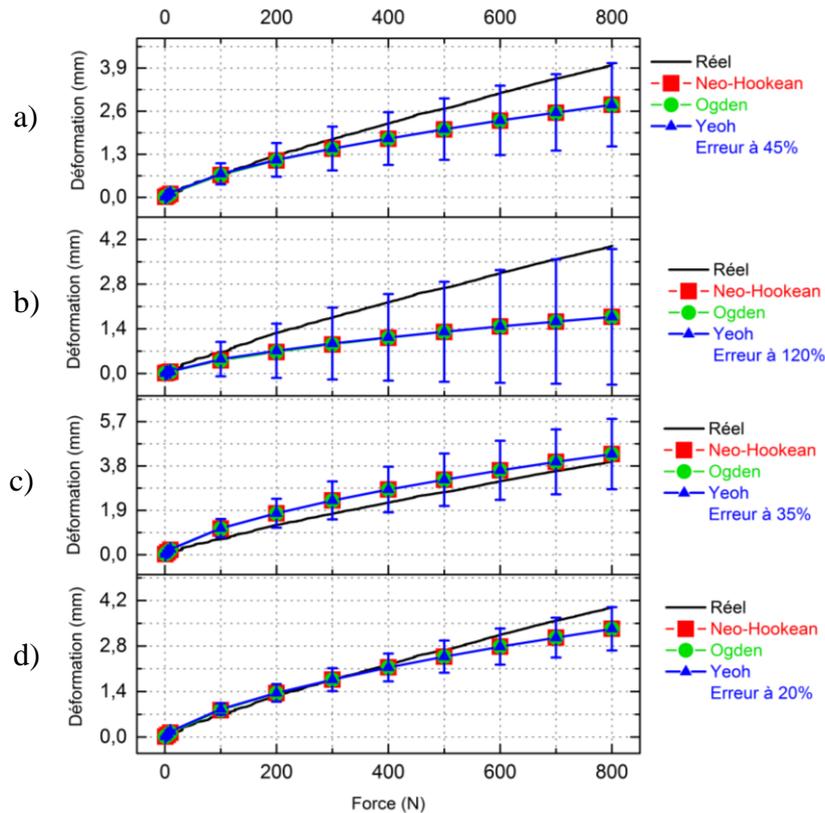


Figure 3.17 Recalage du modèle EF pour les modèles hyper-élastiques. a) Module d'élasticité égal à 1,44 MPa b) Module d'élasticité égal à 2,84 MPa c) Module d'élasticité égal à 0,72 MPa d) Module d'élasticité égal à 1,26 MPa

Pour déterminer la plage de modification de la constante α_i , nous considérons le double de la valeur initiale pour définir la limite supérieure de l'intervalle et la valeur initiale. Nous obtenons donc un intervalle compris entre un et deux. La figure 3.18a montre les résultats obtenus pour la limite inférieure de l'intervalle. La figure 3.18b montre les résultats obtenus pour la limite supérieure, comme nous pouvons l'observer la fonction trouvée est éloignée de la fonction réelle, alors la solution ne se trouve pas dans cet intervalle. Afin de définir un nouvel intervalle, nous prenons la moitié de la valeur initiale pour définir la limite inférieure de l'intervalle et la valeur initiale, alors le nouvel intervalle prend les valeurs comprises entre 0,5 et un. La figure 3.18c montre les résultats obtenus pour la limite inférieure de l'intervalle. Comme nous pouvons l'observer la fonction trouvée est localisée au-dessus de la fonction réelle, alors la solution ne se trouve pas dans cet intervalle. La figure 3.18d montre la meilleure solution obtenue par la méthode de dichotomie pour la valeur de α_i égale à 0,68.

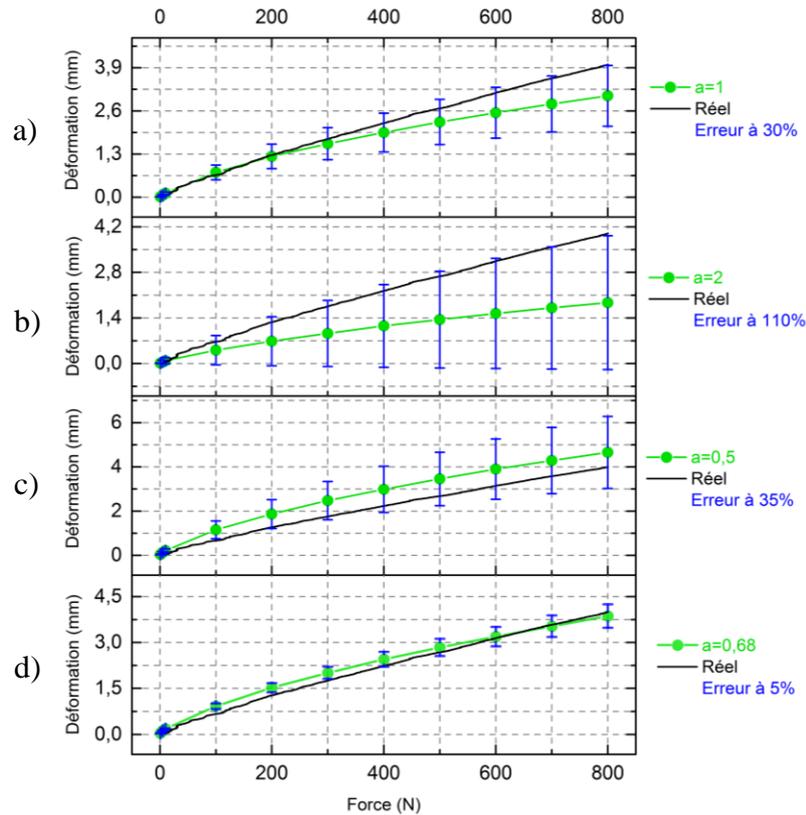


Figure 3.18 Recalage du modèle hyper-élastique Ogden (pour un module d'élasticité égale à 1,26 MPa)

L'erreur entre le modèle réel et le modèle d'Odgen avec de constantes $\alpha_i = 0,68$ et $\mu_i = 0,84$ est inférieur à 5%. Nous considérons que cette approche du modèle numérique représente le comportement plus proche du comportement réel pour le rouleau de compactage.

Une approximation linéaire peut être effectuée, dans le cas où le comportement du matériau travaille dans le limite de charge statique, par contre si le matériau travaille dans la zone de charge dynamique, le comportement du matériau devient non linéaire. En conséquence une approximation non linéaire doit être mise en place pour représenter le comportement réel du matériau à tout moment.

3.3.4 Modèle EF discrétisé

Actuellement, la dépose de fibre considère que la force de compactage est constante le long de la pièce. Pourtant si les dimensions de la pièce sont réduites, la force de compactage n'est pas constante car la surface de contact est modifiée. Cette relation peut être déterminée par (3.34).

$$P = \frac{F}{A} \quad (3.34)$$

Tel que :

P soit la pression de compactage,

F soit la force de compactage,

A soit la surface de contact.

Afin d'établir la pression, il faut déterminer le comportement du rouleau, lorsqu'il est complètement ou partiellement en contact avec la surface (moule).

Le modèle EF proche du modèle réel obtenu dans le paragraphe 3.3.3, permet d'extraire des informations du comportement du rouleau soumis à des forces, comme la surface de contact, la déformation, la pression, les contraintes de cisaillement, les contraintes principales, entre autres. En conséquence, du modèle EF nous extrayons l'information de la pression et de la déformation. Par contre, nous pouvons obtenir la pression et la déformation lorsque le rouleau de compactage se trouve en contact total du modèle du paragraphe 3.3.3. Afin d'obtenir un modèle qui puisse représenter le comportement du rouleau de compactage lorsque la surface de contact est modifiée, nous réalisons de simulations avec le même modèle EF du paragraphe 3.3.3 en modifiant la surface de contact. Les simulations consistent à appliquer des forces comprises entre 0 et 800 N sur le rouleau pour différentes valeurs d'interaction rouleau/surface. Les valeurs d'interaction sont représentées en pourcentage et elles vont de 20% à 100%, soit 100% pour le contact total entre la surface et le rouleau de compactage, dont la surface doit être supérieure ou égale à la longueur du rouleau de compactage (118 mm) et soit 20 % pour le contact partiel entre la surface et le rouleau de compactage, dont la surface est égal à 23,6 mm de la longueur du rouleau de compactage. Les résultats obtenus par le modèle EF pour définir la pression sont montrés figure 3.19a, cette figure montre que le comportement du rouleau de compactage est modifié selon la surface de contact et la force appliquée. De la même manière, nous obtenons les résultats de la déformation du rouleau de compactage par rapport à la force de compactage et la surface de contact, ces résultats sont montrés figure 3.19b.

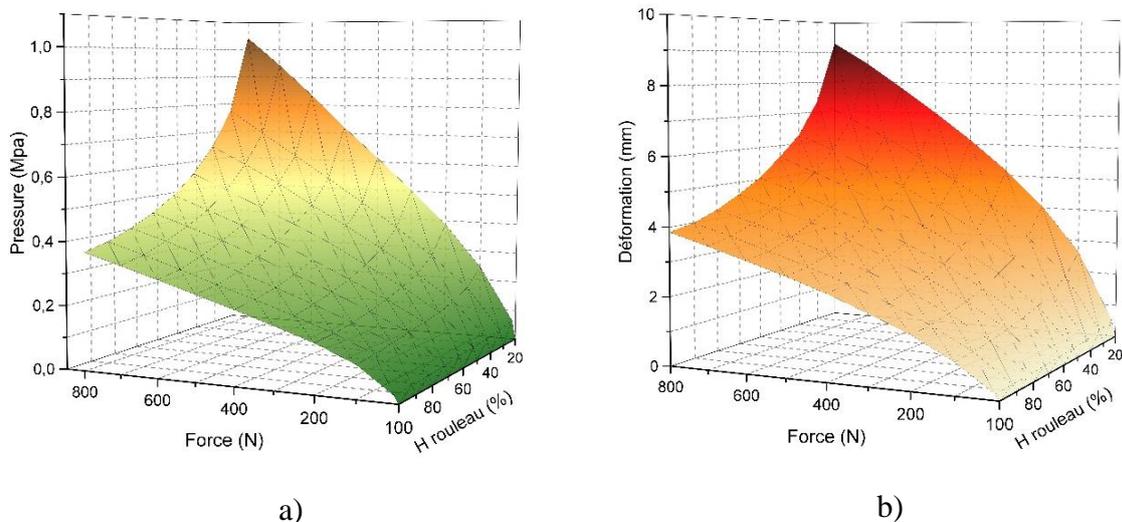


Figure 3.19 Résultats obtenus par le modèle EF : a) pour la pression et b) pour la déformation

3.4 Conclusion

L'étude de la modélisation des robots permet d'approfondir les paramètres à prendre en compte dans la modélisation et la paramétrisation du contrôle du robot, afin de réduire les erreurs de précision dans la trajectoire du robot. Ces paramètres sont : la vitesse, la matrice de raideur, la position et les efforts/couples externes.

Les paramètres de compensation sont obtenus par le modèle dynamique du robot (3.15). Ce modèle considère que le robot est parfaitement rigide, que les forces externes ne génèrent pas de déformations sur le robot et que les déviations sur la trajectoire sont négligeables. Les erreurs de précision sont de l'ordre du millimètre. Ces erreurs sont donc insignifiantes pour les tâches de « pick-place », par contre, elles ne sont pas admissibles pour des tâches comme le détournage, la dépose des fibres, le fraisage..., car les pièces fabriquées exigent une grande précision. Nous devons alors considérer le robot comme un assemblage d'éléments flexibles. Les modèles dynamiques définis paragraphe 3.1.2.2 permettent de prendre en compte les flexibilités sur le robot. Les modèles dynamiques incorporent des flexibilités sont de deux sortes : le modèle dynamique du robot aux articulations flexibles représenté par (3.19) et le modèle dynamique du robot aux corps flexibles représenté par (3.16). Le premier considère que les flexibilités du robot se trouvent seulement dans les articulations (flexibilités localisées), le deuxième considère que les flexibilités du robot se trouvent sur les corps (flexibilités structurelles).

Afin de calculer les modèles dynamiques, les modèles géométrique et cinématique du robot doivent être calculés précédemment. Ces derniers modèles sont calculés grâce aux paramètres de Denavit-Hartenberg (paragraphe 3.1.1.1). Les modèles dynamiques permettent de prendre en compte les flexibilités du robot lorsqu'il est soumis aux forces externes et de corriger sa trajectoire. Par contre, ces modèles sont applicables si le robot permet de modifier le contrôle interne du robot. Dans les robots industriels, les fabricants ne permettent aucune modification pour des questions de garantie et sécurité.

D'un autre côté, l'étude du comportement de l'outil lorsqu'il est soumis à des forces extérieures et le calibrage de l'outil permettent de définir les flexibilités et le point de centrage de l'outil (TCP). La définition correcte du TCP et la correction de la trajectoire par rapport aux flexibilités de l'outil réduisent les erreurs d'imprécision.

Afin de découvrir le comportement de l'outil, nous proposons de réaliser une modèle numérique par éléments finis. Ce modèle permet d'approcher une solution qui traduit le comportement réel. L'exactitude de notre modèle numérique dépend des composants comme la taille du

maillage (la distance entre chaque nœud), les propriétés du matériau, la force appliquée, les dimensions de la pièce et la méthode de solution du logiciel.

Pour définir le modèle numérique, nous avons étudié le comportement de l'outil soumis à une force, puisque le rouleau de compactage est un élément très flexible. Les flexibilités dans le rouleau de compactage entraînent des modifications de la trajectoire du robot et de la définition du point de centrage de l'outil (TCP).

Le comportement réel du rouleau de compactage le long d'un axe Z est obtenu par des tests réalisés sur la machine de traction/compression et le modèle numérique par éléments finis (EF) permet d'approcher une solution qui ressemble au comportement réel. Le modèle numérique comparé avec le modèle réel possède une erreur maximale inférieure à 5%. Ces modèles sont montrés dans la figure 3.18d.

A partir du modèle numérique, nous obtenons des informations du comportement du rouleau de compactage en contact total ou partiel. Parmi l'information acquise, nous trouvons la pression, la surface de contact, la déformation, la force,.... Comme la figure 3.19 le montre.

CHAPITRE

4

Réduction de modèles

Sommaire

4.1 Méthodes de réduction de modèles	62
4.1.1 Méthodes de réduction de modèles a priori.....	63
4.1.2 Méthodes de réduction de modèles <i>a posteriori</i>	64
4.1.2.1 Méthode réseaux de neurones	65
4.1.2.2 Méthode de logique floue	67
4.2 Réduction du modèle EF par réseaux de neurones.....	72
4.3 Conclusion.....	76

Malgré l'impressionnant développement réalisé dans la modélisation mathématique, l'analyse numérique, les techniques de discrétisation et l'informatique, de nombreux problèmes restent insolubles par des techniques numériques directes et traditionnelles, en raison de leur complexité numérique ou de leurs contraintes individuelles (Chinesta et al., 2013). Les différents problèmes restant difficiles pour ces techniques sont :

- La modélisation de systèmes de grande dimension ou de maillage très fin. Ce type de modélisation est lié au nombre de degrés de liberté (Chinesta et al., 2013; Cagin, 2015).
- De nombreux problèmes de modélisation paramétrique, d'identification inverse et d'optimisation. Ce type de problèmes nécessite le calcul direct d'un très grand nombre de solutions du modèle avec des paramètres particuliers et nombreux (Chinesta et al., 2013; Cagin, 2015).
- Le contrôle en ligne d'un système complexe. Les modèles physiques de systèmes complexes sont généralement formulés par un ensemble d'équations aux dérivées partielles non linéaires et fortement couplées. Ces systèmes peuvent être modélisés avec

des techniques de simulation numérique, mais cela nécessite un coût élevé de ressources informatiques. La modélisation par simulation numérique fournit des informations détaillées sur le système, mais elle reste inapplicable pour les applications de contrôle en temps réel. En conséquence, cette modélisation est seulement utilisée pour formuler des modèles simplifiés qui peuvent être implémentés en temps réel (Chinesta et al., 2013; Cagin, 2015).

- Le contrôle de systèmes dynamiques complexes. Les systèmes d'application dynamiques pilotés par les données (DDDAS) sont un ensemble de techniques qui permettent le couplage d'outils de simulation avec des dispositifs de mesure pour le contrôle en temps réel de systèmes dynamiques complexes (Chinesta et al., 2013).
- La réalité augmentée. Ce domaine doit fournir en temps réel des informations de la réalité perçue par l'utilisateur (Chinesta et al., 2013).

Le besoin de développer un modèle multi-variable en temps réel qui représente le comportement du rouleau de compactage à partir des simulations numériques, ainsi que d'obtenir une loi de commande modifiable selon les caractéristiques de la surface, nous permet de résoudre ces problèmes par des techniques indirectes comme les méthodes de réduction de modèles.

4.1 Méthodes de réduction de modèles

La réduction du modèle est une technique fréquemment utilisée dans l'analyse multi-physique pour réduire la complexité des modèles (Schilders, 2008; Bonithon et al., 2011; Bur, 2015) et/ou le temps de calcul (Bonithon et al., 2011; Heyberger, 2014) en préservant au mieux la relation entrées-sorties (Guillet, 2011) et les caractéristiques essentielles du modèle initial (Chen, 1999; Bonithon et al., 2011). Les propriétés fondamentales du modèle initial peuvent être représentées par un modèle approximatif (Heres, 2005).

Les principales raisons d'obtenir un modèle réduit sont (Schilders, 2008) :

- Simplifier l'étude du système.
- Diminuer les temps de calcul lors de la simulation.
- Réduire les temps de calcul de manière à rendre la conception de commandes plus efficaces.
- Obtenir des lois de commande simplifiées.
- Réduire le coût computationnel et la complexité du modèle.

Les méthodes de réduction de modèles peuvent être classifiées dans deux groupes :

- Méthodes de réduction de modèles *a priori*

- Méthodes de réduction de modèles *a posteriori*

L'approche à mettre en place est généralement choisie en fonction du domaine d'application, de l'utilisation du modèle réduit et de l'erreur induite par la réduction. La volonté de réduire la dimension d'un modèle revient souvent à identifier les domaines numériques portant les informations les plus pertinentes (Cagin, 2015).

4.1.1 Méthodes de réduction de modèles *a priori*

Les méthodes de réduction de modèles *a priori* n'ont pas besoin d'une connaissance préalable de la solution recherchée (Allery et al., 2011; Dumon, 2011). Ces méthodes effectuent la réduction avant le développement du modèle (Cagin, 2015). Parmi les méthodes présentées dans la littérature, nous retiendrons les méthodes APR et PGD. Ces méthodes sont les plus étudiées, les plus développées et les plus utilisées pour la modélisation et la commande de systèmes complexes. Les méthodes APR et PGD consistent à construire une base de la fonction pour représenter la solution. A chaque itération cette base de représentation est améliorée pour approcher le plus possible la solution. Les autres méthodes *a priori* développées dans la littérature sont basées dans les principes des méthodes APR et PGD.

- **Méthode APR**

La méthode de réduction *a priori* (APR) est basée sur une méthode interactive. Cette méthode consiste à projeter le problème discret sur une base définie pour obtenir un modèle réduit. Ce modèle doit être résolu de façon rapide et précise avec des schémas numériques standards. Chaque interaction permet d'améliorer la base. Cette amélioration est divisée en deux phases : la première améliore la base précédente à partir de la décomposition de Karhunen-Loève et la deuxième amplifie la base améliorée avec les vecteurs de Krylov (Dumon, 2011).

L'APR permet donc de calculer une solution du problème sous la forme de décomposition de fonctions spatiales et temporelles de dimension très faible comparée à la taille du problème discrétisé de sortie.

- **Méthode PGD**

La méthode Proper Generalized Decomposition (PGD) est une méthode de réduction de modèle *a priori*. La PGD permet de résoudre un problème transitoire de trois manières :

- 1) La première consiste à trouver une solution sous forme d'une décomposition de variables d'espace et de temps (Boucinha, 2013; Bur, 2015) en utilisant une approche non-incrémentale.
- 2) La seconde consiste à trouver une solution sous la forme d'une décomposition de chacune des variables de l'espace physique (Heyberger, 2014). Cette décomposition

conduit à une augmentation significative du nombre de termes de la somme des termes de décomposition.

- 3) La troisième consiste à garder une approche incrémentale en temps et d'effectuer la séparation, à chaque pas de temps, uniquement sur les variables spatiales (Dumon, 2011; Metoui, 2015).

La PGD a également été appliquée pour résoudre des équations aux dérivées partielles stochastiques non linéaires.

4.1.2 Méthodes de réduction de modèles *a posteriori*

Les méthodes de réduction de modèles *a posteriori* doivent avoir une connaissance préalable de la solution (Allery et al., 2011; Dumon, 2011). Ces méthodes effectuent la réduction à partir d'un premier modèle déjà établi (Dumon, 2011; Cagin, 2015). Parmi les méthodes présentées dans la littérature, nous retiendrons les méthodes POD, SVD et approximation. Ces méthodes sont les plus développées et plus utilisées pour la modélisation. La méthode à mettre en place est choisie en fonction de l'application, de l'utilisation, de la représentation qui doit être plus proche du modèle réel et de l'erreur induite par la réduction. En effet, les méthodes POD, SVD et approximation permettent un calcul algébrique en un nombre fini de point, cela engendre une réduction du temps de calcul pour obtenir une approximation de la solution du modèle complet dont nous avons une connaissance préalable.

- **Méthode POD**

La méthode « Proper Orthogonal Decomposition » (POD) est une méthode *a posteriori* de projection linéaire (Gu, 2011) qui permet de séparer les variables d'espace et de temps (Boucinha, 2013; Cagin, 2015). Cette méthode prend en compte plusieurs données/trajectoires des variables d'état, afin de construire un sous-espace linéaire qui minimise certaines erreurs entre le modèle réduit projeté et le modèle complet (Gu, 2011). En d'autres termes, la méthode consiste à réduire la dimension d'un ensemble de données formé par un grand nombre de variables, à partir de la transformation et de la classification des variables dans un ensemble sans aucune corrélation avec des nouvelles variables. En conséquence, les premières composantes contiennent la plupart des variations contenues dans les variables originelles (Dumon, 2011). La POD est une des techniques classiques les plus utilisées pour les comportements eulériens (Bergmann, 2004; Dumon, 2011; Tissot, 2014) et est étendue au cas des matrices non carrées.

- **Méthode SVD**

Le modèle Singular Value Decomposition (SVD) est une méthode de factorisation de matrices (Dumon, 2011). Il est devenu très important suite à l'augmentation du nombre d'applications et à l'accroissement de la capacité de mémoire et de la vitesse de calcul des ordinateurs (Martin & Porter, 2012). Cette méthode peut être interprétée comme une extension de la décomposition aux valeurs propres pour les matrices rectangulaires (Dumon, 2011).

- **Méthodes d'approximation**

Les méthodes d'approximation permettent de trouver la solution à partir de l'approximation de la fonction de transfert du modèle (Schilders, 2008; Salah, 2017). La fonction de transfert est considérée comme une boîte noire, car la structure interne et/ou la représentation du modèle sont inaccessibles (Gu, 2011). En conséquence, seulement les entrées contrôlables et les sorties observables sont abordables. Ces méthodes utilisent des techniques d'ajustement de courbes pour rapprocher les données (Schutt-Ainé & Goh, 2013) comme la régression, le réseau de neurones, l'apprentissage automatique,... (Gu, 2011). Parmi les algorithmes d'approximation pour la réduction de modèles, cette recherche considère les réseaux de neurones et la logique floue.

4.1.2.1 Méthode réseaux de neurones

Les réseaux de neurones artificiels (dont les sigles en anglais sont ANN) est un modèle mathématique et/ou informatique inspiré par la structure du cerveau humain spécifiquement dans la structure et/ou le fonctionnement des réseaux des neurones biologiques (Renders, 1994; Turban et al., 2004; Baş & Boyacı, 2007; Akintunde et al., 2015; Astray et al., 2016; Durodola et al., 2017). Le cerveau humain est composé de cellules appelées « neurones », ces neurones sont connectés entre eux « réseau », afin d'obtenir des informations d'autres systèmes et d'envoyer cette information, au système nerveux central qui se charge de l'évaluation.

Au cours des dernières années, la méthode des réseaux de neurones artificiels est très utilisée pour modéliser, approcher et optimiser les problèmes non linéaires (Desai et al., 2008; Condon & Grahovski, 2010b; San & Maulik, 2017), pour fournir des solutions adaptatives et pour réévaluer les paramètres du modèle dans des formes relativement simples (Pappu & Gummadi, 2017; Patra et al., 2017). Le réseau de neurones artificiels est une technique de modélisation considérée comme une méthode d'identification du système (Everett & Dubay, 2017), donc il permet de réaliser la réduction de modèles à partir de l'approximation du modèle. Cette réduction doit conserver les propriétés fondamentales du modèle original.

La méthode de réseaux de neurones artificiels est implémentée pour prédire les éléments inconnus du modèle d'ordre réduit. La prédiction de cette méthode est basée sur la minimisation

de la fonction obtenue par la différence entre le comportement actuel et le comportement souhaité du système (Adel & Salah, 2016)

Les réseaux de neurones artificiels possèdent des avantages comme l'apprentissage de données incomplètes, la capacité à traiter l'information en parallèle et la prédisposition à travailler avec des données non linéaires, bruitées, complexes et imprécises (Perpetuo et al., 2012; Akintunde et al., 2015). Ils permettent une grande marge de généralisation non linéaire et une capacité de réaction avec un grand nombre de variables d'entrée dans les méthodes d'optimisation (Farley et al., 2012; Durodola et al., 2017).

Le fonctionnement et la méthode d'apprentissage des réseaux de neurones artificiels sont similaires à ceux du cerveau humain (Astray et al., 2016). La structure du réseau de neurones artificiels est composée de trois couches : une couche d'entrée, une couche cachée et une couche de sortie (Patra et al., 2017). Les neurones de la couche cachée sont interconnectés aux neurones des couches précédentes et successives par des liens pondérés (Patra et al., 2017). La configuration du réseau de neurones artificiels dépend du système, du nombre de couches, du nombre de neurones requis pour décrire un système et de la complexité du système (Everett & Dubay, 2017).

La figure 4.1 montre une structure du réseau de neurones artificiels telle que : les entrées sont représentées par E, les neurones par N et les sorties par S.

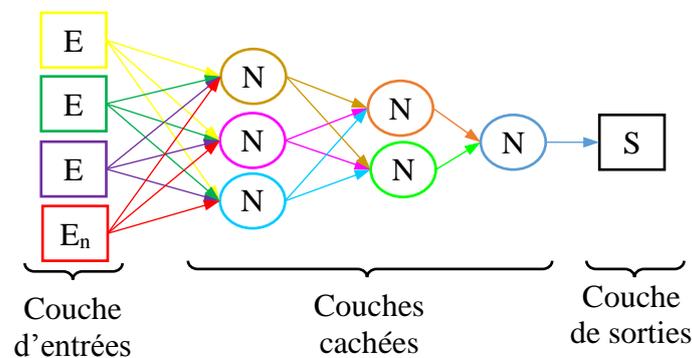


Figure 4.1 Diagramme du réseau de neurones artificiel

Dans chaque neurone existent des fonctions mathématiques (sigmoïde, tangente hyperbolique (Tanh), gaussienne, linéaire et bipolaire (Akintunde et al., 2015)) qui traitent le signal d'entrée et obtiennent le signal résultant envoyé à tous les neurones connectés. Ainsi chaque neurone est composé de deux paramètres importants : le biais et le poids. Ces paramètres sont trouvés par formation en référence à la valeur de la sortie par rapport à la valeur de l'entrée. La figure 4.2 montre le diagramme de chaque neurone et les équations réalisées pour traiter le signal d'entrée.

connaissances inexactes pour des données incomplètes, incompatibles ou même manquantes (Negnevitsky, 2005). La logique classique est décrite par :

Si (IF) A est vrai, alors (THEN) A n'est pas faux,
 Si (IF) B est faux, alors (THEN) B n'est pas vrai

La logique floue utilise les règles «IF-THEN» pour intégrer les connaissances humaines, mais ces règles sont floues telles que :

Si (IF) la température ambiante est de 30 °C, alors (THEN) il fait chaud
 Si (IF) la température ambiante est de 20 °C, alors (THEN) il fait tempéré
 Si (IF) la température ambiante est de 10 °C, alors (THEN) il fait froid

Les variables de la logique floue sont considérées comme des variables linguistiques, car elles sont décrites plutôt par des mots que par des nombres. Ces mots peuvent être utilisés directement dans des règles. Les valeurs floues de l'exemple décrit antérieurement sont : « chaud », « tempéré » et « froid ».

La théorie des ensembles flous est un moyen de spécifier à quel point un objet satisfait à une description vague (Russell & Norvig, 1995). La théorie des ensembles flous repose sur la notion d'appartenance partielle : chaque élément appartient partiellement ou graduellement aux ensembles flous déjà définis (Chevrie & Guély, 1998; Démoncourt, 2011). Un ensemble flou est défini par sa « fonction d'appartenance » qui peut être partiellement vraie ou partiellement fautive. La fonction d'appartenance peut prendre une valeur réelle dans l'intervalle [0,1]. Le tableau 4.1 montre les expressions mathématiques qui représentent la fonction caractéristique de la logique classique et la fonction d'appartenance de la logique floue.

Tableau 4.1 Fonction caractéristique et fonction d'appartenance

Fonction caractéristique (logique classique)	Fonction d'appartenance (logique floue)
$f_A(x) : X \rightarrow 0,1$ Telle que : $f_A(x) = \begin{cases} 1, & \text{Si } X \in A \\ 0, & \text{Si } X \notin A \end{cases}$	$\mu_A(x) : X \rightarrow [0,1]$ Telles que : $\mu_A(x) = 1$ Si X est complètement dans A $\mu_A(x) = 0$ Si X n'est pas dans A $0 < \mu_A(x) < 1$ Si X est partiellement dans A

Les fonctions d'appartenance peuvent théoriquement prendre n'importe quelle forme, pourtant les plus utilisées sont les triangles et trapézoïdes, car ces figures sont simples et permettent de délimiter les zones de vérité et d'erreur. Les fonctions d'appartenance dans l'exemple décrit antérieurement sont : « chaud », « tempéré » et « froid » (figure 4.3).

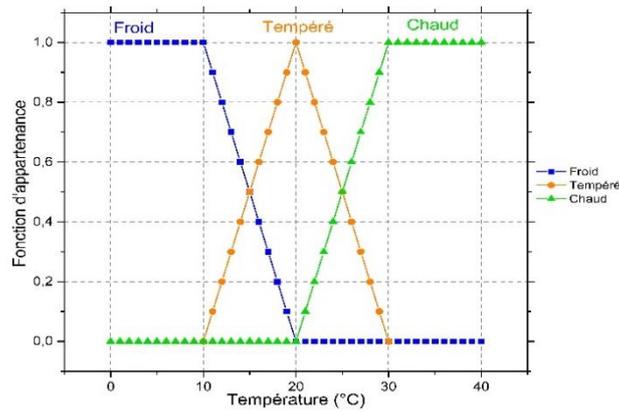


Figure 4.3 Représentation des variables linguistiques de la logique floue

Les opérateurs de la logique floue permettent d'écrire des combinaisons logiques entre notions floues et de calculer les degrés de vérité.

- Union. Cet opérateur logique correspondant à la liaison « ou » est représenté mathématiquement par « \cup ». Le degré de vérité de la proposition « A OU B ou $A \cup B$ » est le maximum des degrés de vérité de A et de B. Cet opérateur est représenté par (4.2).

$$\mu_{A \cup B}(x) = \text{MAX}(\mu_A(x), \mu_B(x)) \quad (4.2)$$

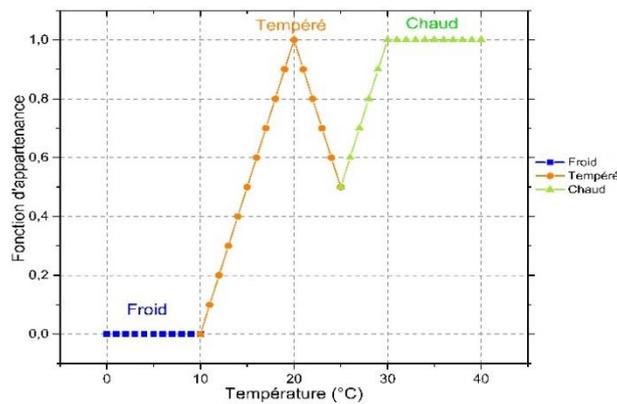


Figure 4.4 Représentation graphique de l'opérateur logique floue « union »

- Intersection. Cet opérateur logique correspondant à la liaison « ET » est représenté mathématiquement par « \cap ». Le degré de vérité de la proposition « A ET B ou $A \cap B$ » est le minimum des degrés de vérité de A et de B. Cet opérateur est représenté par (4.3).

$$\mu_{A \cap B}(x) = \text{MIN}(\mu_A(x), \mu_B(x)) \quad (4.3)$$

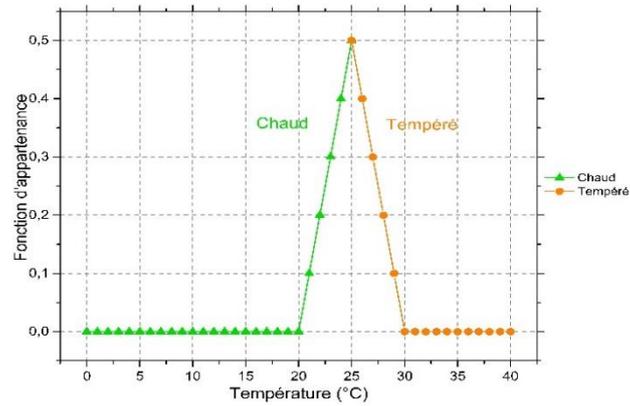


Figure 4.5 Représentation graphique de l'opérateur logique flou « intersection »

- Complément. Cet opérateur logique correspondant à « NON » est représenté mathématiquement par « \neg ». L'opérateur logique correspondant au complément d'un ensemble est la négation. Cet opérateur est représenté par (4.4).

$$\mu_{\neg A}(x) = 1 - \mu_A(x) \quad (4.4)$$

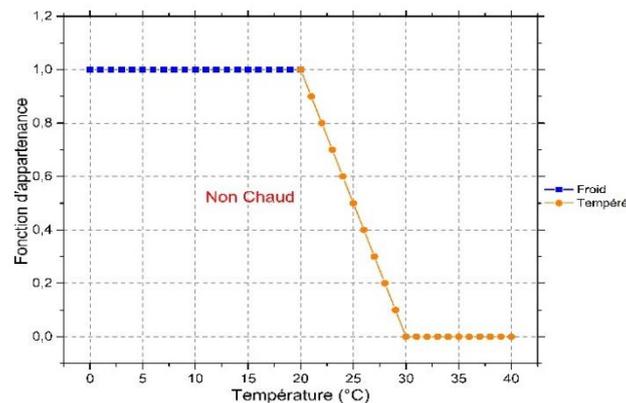


Figure 4.6 Représentation graphique de l'opérateur logique flou « non »

L'implémentation de logique floue doit principalement respecter les étapes suivantes :

- Fuzzification. Cette étape permet de convertir les valeurs réelles (domaine réel) en valeurs floues (domaine flou) et consiste à définir le degré d'appartenance d'une valeur réelle à un ensemble flou.

Pour une température ambiante de 27 °C de l'exemple décrit antérieurement, la logique floue se traduit par :

La température ambiante est de 27 °C

« Il fait chaud » son degré d'appartenance est de 85%

« il fait tempère » son degré d'appartenance est de 15%

« il fait froid » son degré d'appartenance est de 0%

- Définition et évaluation des règles floues. Cette étape permet de décrire la façon dont le problème est résolu en utilisant les variables de la logique floue.

Les règles floues peuvent être représentées par une matrice. Le système de deux entrées et une sortie est représenté sous la forme d'une matrice de variables d'entrée $M \times N$. Les valeurs linguistiques d'une variable d'entrée forment l'axe horizontal et les valeurs linguistiques de l'autre variable d'entrée forment l'axe vertical. À l'intersection d'une ligne et une colonne se trouve la valeur linguistique de la variable de sortie. Un système trois entrées et une sortie, est représenté par un cube $M \times N \times K$. Cette forme de représentation s'appelle mémoire associative floue (FAM en anglais).

La FAM pour l'exemple antérieurement décrit, est représentée sur la figure 4.7.

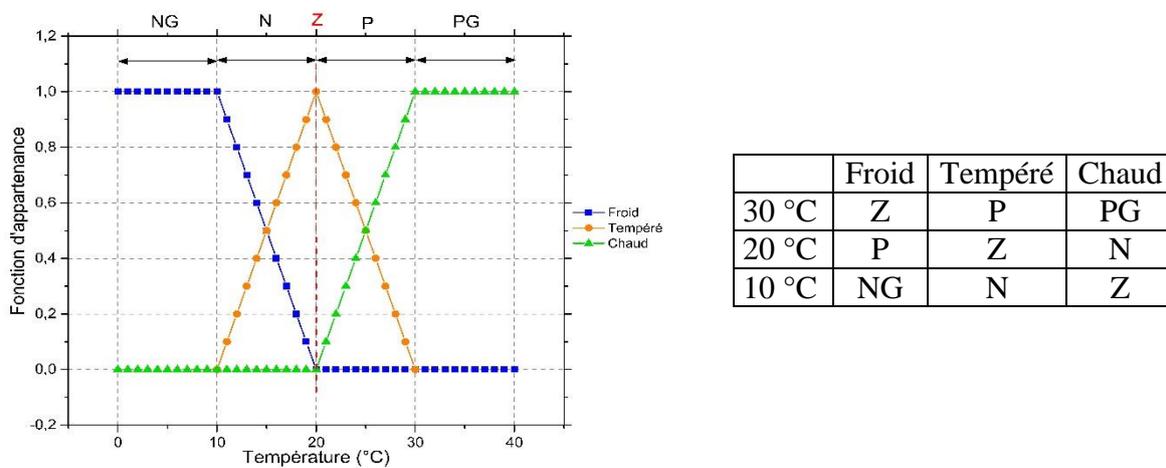


Figure 4.7 Représentation de la FAM

Tel que :

Z représente zéro,

P représente positive,

N représente négative,

PG représente positive grande,

NG représente négative grande,

Les règles de l'exemple antérieurement décrit sont :

PG= Si (IF) la température ambiante est de 30 °C, ou (OR) la température ambiante est supérieure à 30 °C, alors (THEN) « sortie élevée »

P= Si (IF) la température ambiante est supérieure à 20 °C et (AND) la température ambiante est inférieure à 30 °C, alors (THEN) « sortie moyenne positive »

Z= Si (IF) la température ambiante est de 20 °C, alors (THEN) « sortie moyenne »

N= Si (IF) la température ambiante est supérieure à 10 °C et (AND) la température ambiante est inférieure à 20 °C, alors (THEN) « sortie moyenne négative »

NG= Si (IF) la température ambiante est de 10 °C, ou (OR) la température ambiante est inférieure à 10 °C alors (THEN) « sortie négative »

- Défuzzification. Cette étape permet de convertir les valeurs floues (domaine flou) en valeurs réelles (domaine réel). Il existe différentes méthodes pour réaliser la défuzzification, la plus courante dans la littérature étant la méthode du centre de gravité. Elle est représentée mathématiquement par (4.5) :

$$\text{Sortie} = \frac{\int x\mu(x)dx}{\int \mu(x)dx} \quad (4.5)$$

4.2 Réduction du modèle EF par réseaux de neurones

Le comportement du rouleau de compactage peut être obtenu par la méthode d'EF avec une bonne approximation du modèle réel, comme il est indiqué dans les paragraphes 3.3.2 et 3.3.3. Par contre, cette méthode est incompatible pour la plupart des systèmes en temps réel, puisque le temps de calcul est très élevé. En conséquence, dans ce paragraphe, nous développons le modèle réduit du modèle EF pour l'appliquer expérimentalement en temps réel.

Les différentes méthodes pour la réduction de modèles sont exposées dans ce chapitre. Les méthodes d'approximation représentent la meilleure option pour réduire notre modèle. Ces méthodes permettent d'obtenir un modèle multi-variable approché du modèle réel à partir de l'approximation à la courbure de solution. Parmi ces modèles, nous choisissons la méthode de réseau de neurones indiquée dans le paragraphe 4.1.2.1, afin d'approcher le modèle du comportement du rouleau de compactage obtenu par le modèle EF (figure 3.19). La méthode de réduction de réseau de neurones a été utilisée par (Bellamine & Elkamel, 2008; Chetouani, 2009; Condon & Grahovski, 2010a; Alsmadi et al., 2011; Adel & Salah, 2016; Pater, 2016; San & Maulik, 2017). Cette méthode permet de réaliser la réduction de modèles à partir de l'approximation du modèle. Les réseaux des neurones est une méthode qui essaie d'imiter le fonctionnement du cerveau humain. Ces neurones séparément n'ont pas une tâche spécifique,

mais à partir de l'expérience, les neurones établissent des synapses et renforcent des connexions pour obtenir un algorithme d'apprentissage. Les réseaux de neurones nous permettent alors de trouver des solutions à un problème particulier qui ne peut pas être résolu par des algorithmes traditionnelles. Ces solutions sont obtenues à partir de la combinaison des paramètres d'entrée et de l'ajustement de l'apprentissage du réseau de neurones. Par contre pour que les réseaux de neurones donnent une solution précise, ils ont besoin de cas réels servant d'exemples pour leur apprentissage. Les réseaux de neurones, grâce à leur capacité d'être des « approximateurs universels », ont prouvé leur capacité à extraire de données d'expérimentation des modèles performants, sans avoir à effectuer d'hypothèses sur la forme générale de ces derniers.

Parmi les avantages des réseaux de neurones, nous pouvons mentionner les capacités pour apprendre à partir de l'expérience (le fait d'assimiler sous une forme dense des exemples éventuellement nombreux) ou d'une généralisation (le fait d'être capable, grâce aux exemples appris, de traiter des exemples distincts, encore non rencontrés, mais similaires). Ils permettent aussi d'organiser leur même apprentissage, cette auto-organisation consiste à modifier complètement le réseau de neurones pour accomplir l'objectif. Les réseaux des neurones ont la capacité de continuer à travailler même s'il existe un problème dans une partie du système. Le grand avantage de cette méthode, cette méthode est de pouvoir travailler en temps réel à partir des connexions en parallèle. Une connexion réalise une transmission de haute débit. Les moyens technologiques pour l'implémentation selon les tâches sont accessibles sur les ordinateurs actuels.

Les réseaux de neurones nous permettent aussi de travailler avec des données incomplètes, non linéaires, bruitées, complexes ou imprécises, et de travailler avec un grand nombre de variables d'entrée et de sortie. Outre la facilité d'implémentation, l'erreur et le temps de calcul de cette méthode sont minimales, ce qui est compatible avec la période d'échantillonnage du robot qui est de 12 ms.

Grace à ce type de fonctionnement des réseaux de neurones, les conditions limites se trouvent dans une zone quasi-linéaire ou une zone zéro, c'est-à-dire si le neurone est active sa valeur est supérieure à zéro et inférieure au seuil, si il est inactive son valeur est inférieure ou égale à zéro. On peut considérer que dans cette zone le neurone fonctionne comme un dispositif analogique. Mais en présence de saturations très proches de la zone quasi-linéaire, le neurone fonctionne comme un dispositif numérique (binaire). Donc le neurone a la possibilité d'ajuster ses poids pour fonctionner en analogique ou en numérique suivant les besoins. C'est indispensable que les poids initiaux des neurones soient adaptés à l'application, dans le cas contraire ils resteraient

inutilisés durant les premières étapes de l'apprentissage, et un grand nombre de neurones satureraient, d'où un gradient faible et un apprentissage très lent.

Représentation du modèle par réseau de neurones

L'approximation du modèle reprend les propriétés basiques du modèle original. Pourtant ce modèle doit contenir suffisamment d'information, afin de garantir que le modèle est proche du modèle réel. Le modèle par réseau de neurones peut interpoler les points d'apprentissage intermédiaires.

Dans le cadre de cette recherche, nous utilisons trois réseaux de neurones. Chaque réseau utilise une couche d'entrée avec deux paramètres, trois couches cachées dont la première et la deuxième couches possèdent quatre neurones et la troisième seulement un neurone et finalement une couche de sortie avec un seul paramètre, comme le montre la figure 4.8. Un réseau de neurones résout la pression dont les paramètres d'entrée sont la force de compactage (F) et la surface de contact (H). Le deuxième réseau de neurones résout la déformation dont les paramètres d'entrée sont aussi la force de compactage (F) et la surface de contact (H). Finalement, le dernier réseau de neurones résout le modèle réduit inverse dont le paramètre de sortie est la force avec la pression (P) et la surface de contact (H) comme les paramètres d'entrée.

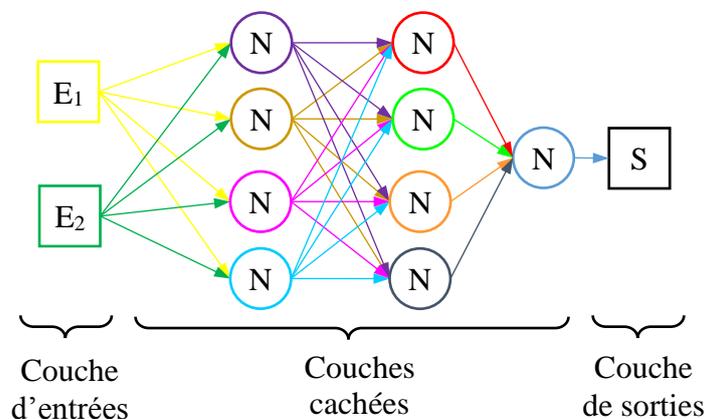


Figure 4.8 Diagramme de réseaux de neurones pour le modèle réduit

Les fonctions mathématiques de chaque neurone sont : pour les premières couches une fonction sigmoïde tangente hyperbolique ($f_1, f_2 = \text{« tansig »}$) et pour la dernière couche une fonction linéaire ($f_3 = \text{« pureline »}$).

Les modèles du rouleau de compactage par rapport à leur pression et à leur déformation sont fonction de la force et de la surface de contact et peuvent être représentés mathématiquement par (4.6) et par (4.7) respectivement.

$$P = f(F, H) \quad (4.6)$$

$$\Delta = f(F, H) \quad (4.7)$$

L'apprentissage du modèle réduit par réseau de neurones dans cette recherche, est effectué par l'outil logiciel « Matlab neural fitting tool », tel que le modèle initial ou réel est représenté par le modèle réalisé dans le paragraphe 3.3.4. Les paramètres de sortie (pression, déformation, force) sont calculés par (4.5). L'algorithme utilisé pour l'apprentissage des points intermédiaires du modèle est celui de Levenberg-Marquardt qui permet d'obtenir une solution numérique au problème de minimisation d'une fonction, souvent non linéaire et dépendant de plusieurs variables. Les résultats obtenus du modèle réduit par réseau de neurones sont montrés dans la figure 4.9a pour la pression et dans la figure 4.9b pour la déformation. Ces résultats montrent une bonne approximation au modèle par éléments finis avec 405 interactions pour la pression et 183 interactions pour la déformation. Le modèle par éléments finis possède seulement 90 points pour réaliser l'approximation, par contre le nombre de points obtenus par les réseaux de neurones est égal à 1681. Ce qui permet de calculer les valeurs intermédiaires du modèle. L'erreur quadratique moyenne (MSE) est de 7×10^{-6} MPa pour le modèle à pression et 5×10^{-6} mm pour le modèle à déformation.

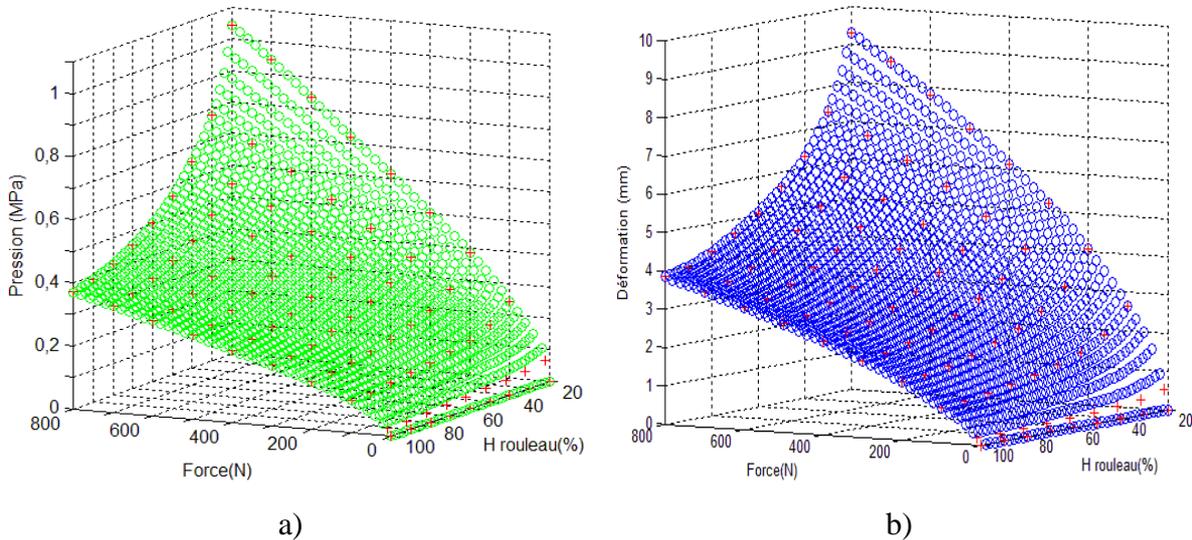


Figure 4.9 Résultats du modèle réduit par réseau de neurones a) modèle de pression b) modèle de déformation

Le modèle réduit inverse représenté par (4.8) permet de calculer une force selon la surface de contact et la pression.

$$F = f(P, H) \quad (4.8)$$

Le modèle par réseau de neurones est programmé en langage C#, afin d'être compatible avec le dispositif expérimental implémenté pour la génération de trajectoires.

4.3 Conclusion

L'étude de la réduction de modèle nous permet de modéliser divers types de problèmes comme :

- Les systèmes complexes de grande dimension ou de maillage très fin.
- Le contrôle des modèles complexes en temps réel.
- Le contrôle de systèmes dynamiques complexes en temps réel.
- La modélisation paramétrique, d'identification et d'optimisation.
- La réalité augmentée.

Ces problèmes possèdent les mêmes contraintes comme l'impossibilité de travailler en temps réel ou de trouver une solution au problème, car ces modèles sont des systèmes non linéaires et ils possèdent un grand coût informatique, une grande difficulté de calcul et un grand nombre de solutions avec des multiples paramètres.

Le besoin toujours croissant d'une prédiction précise de nombreux processus non linéaires complexes conduit à des systèmes dynamiques à très grande échelle dont les simulations et les analyses imposent des exigences écrasantes et ingérables sur les ressources de calcul. Cette volonté de réduire la dimension d'un modèle revient souvent à identifier les domaines numériques portant les informations les plus pertinentes.

Ces méthodes de réduction nous ont permis donc de développer un modèle multi-variables du comportement du rouleau de compactage exploitable en temps réel, ainsi qu'une loi de commande modifiable selon les exigences souhaitées.

Pour la réduction de modèles *a priori*, nous devons prédire le comportement du rouleau de compactage et les valeurs précises des correcteurs de la loi de commande. Par contre, la prédiction de ces éléments est difficile car le rouleau de compactage est composé de différents matériaux et les paramètres de la loi de commande sont calculés selon les caractéristiques de la surface en contact.

Pour la réduction de modèles *a posteriori*, nous modélisons le comportement du rouleau de compactage et la loi de commande en connaissant les modèles préalables. Ces modèles préalables peuvent être définis par l'expérimentation ou par la simulation.

Parmi les méthodes *a posteriori*, les méthodes d'approximation permettent de trouver une solution par approximation de la fonction de transfert. Les deux problèmes présentent une structure interne, cette structure ne peut pas être modifiable. Ainsi ces problèmes possèdent de multiples variables d'entrée et de sortie. Le modèle du comportement du rouleau de compactage

est obtenu par simulation, dont nous obtenons des solutions qui peuvent être représentées par une courbe. La méthode de réseau de neurones représente la meilleure façon d'approcher la courbe de solution du comportement du rouleau de compactage, car l'erreur et la vitesse de calcul de cette méthode sont minimales. Cette méthode permet de calculer la solution dans le temps d'échantillonnage du robot.

D'autre part, la méthode de logique floue permet de calculer en temps réel les paramètres correcteurs de la loi de commande, car un algorithme de logique floue peut calculer les valeurs dans l'intervalle $[0,1]$ selon les paramètres souhaitées.

Le temps de calcul du modèle numérique est incompatible avec la plupart des systèmes en temps réel. De ce fait, nous proposons l'utilisation d'un modèle réduit pour interagir avec le robot en temps réel. Le modèle réduit est effectué par la méthode de réseaux de neurones. Les résultats sont montrés dans la figure 4.9.

CHAPITRE

5

Génération de consigne des robots par capteurs externes

Sommaire

5.1 Types de commandes	79
5.2 Structure de la commande	82
5.3 Task Frame Formalism (TFF)	84
5.4 Loi de commande	86
5.4.1 Loi de commande en position des robots rigides	87
5.4.2 Loi de commande en force	88
5.4.3 Loi de commande hybride externe en position/force	88
5.4.4 Correcteur PID.....	89
5.4.5 Correcteur logique floue-PI.....	91
5.5 Sélection de la loi de commande.....	93
5.5.1 La loi de commande classique PID	93
5.5.2 La loi de commande en logique floue PI.....	96
5.5.3 Bilan.....	98
5.6 Conclusion.....	100

La plupart des robots font des mouvements répétitifs et dans un environnement limité. Cet environnement est composé des éléments en contact et des éléments autour du robot. Ces éléments sont prédéfinis dans la commande pour éviter des collisions ou des erreurs dans les trajectoires. Par contre, si des changements dues à l'environnement et/ou à l'orientation et la

position de la pièce et/ou à un événement externe sont présents, la réaction du robot est très limitée (Gangloff & Poignet, 2007), en conséquence les éléments en contact (la pièce et/ou le robot) peuvent subir des dégâts irréversibles. Dans certaines tâches, la définition des mouvements et de l'environnement est programmable hors-ligne. Cependant il y a d'autres tâches qui ont besoin que le robot puisse travailler dans un environnement inconnu ou dynamique (Lahouar et al., 2006; Mezouar et al., 2007) en temps réel. D'autres tâches nécessitent que le robot exerce des forces dans un environnement rigide (Degoulange et al., 1993) avec des mouvements libres et/ou des mouvements limités, par exemples : l'assemblage, l'usinage et le suivi des surfaces (polissage) (W. Khalil & E. Dombre, 1999). En conséquence, beaucoup des recherches portent sur la commande du robot, afin de considérer un environnement dynamique.

5.1 Types de commandes

Divers types de commandes peuvent être employés pour commander la position et/ou l'orientation du robot. Elles sont divisées principalement en deux groupes, soit la commande classique, soit la commande en interaction avec l'environnement.

Si l'environnement est parfaitement connu, la tâche commandée sera répétitive et fonctionnelle, donc la commande classique est utilisée. La figure 5.1 montre le diagramme de commande d'un robot industriel traditionnel en boucle fermée. Ce diagramme montre le contrôle interne du robot défini par la commande des mouvements du robot en position, en orientation ou en vitesse.

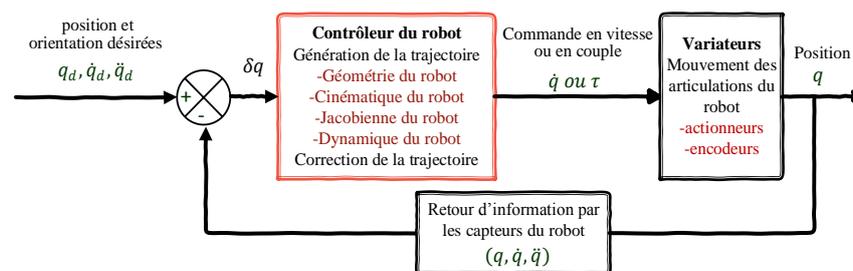


Figure 5.1 Commande de boucle fermée du robot

Si le robot trouve un obstacle non prévu ou non connu dans l'environnement, les mouvements commandés ne sont pas respectés et le robot, et/ou l'environnement et/ou la pièce pourront donc être détériorés. Ces problèmes sont dus à la connaissance limitée de l'environnement (Ferguene & Toumi, 2009) et aux limitations de la baie de commande qui possède une boucle fermée. La solution donnée à cette problématique est l'implémentation de capteurs externes (extéroceptifs) permettant de réaliser l'interaction du robot avec son environnement, en temps réel (Smits et al., 2006). Jusqu'à maintenant les robots conçus par de grandes entreprises comme KUKA,

ABB, TOYOTA... et mis en œuvre dans l'industrie ne disposent pas, par défaut, des capacités de recevoir des informations par des capteurs externes dans leur contrôle interne, sauf l'information interne du robot (position, orientation, vitesse) transmise par les capteurs proprioceptifs (Smits et al., 2006). Dans le paragraphe suivant, les différents types de commande en interaction avec l'environnement sont expliqués.

Types de commande du robot en interaction avec l'environnement

Les robots industriels sont de plus en plus en interaction avec leur environnement, par exemple pour des tâches qui demandent d'être en contact constant (robot et surface de contact) ou pour des tâches qui exigent d'être en contact avec des humains ou des robots, ou pour des tâches qui requièrent un changement constant de la surface de travail ou d'environnement. Ces tâches exposent les robots aux forces externes dues aux contraintes de l'environnement (Katsura et al., 2006). Les avancées technologiques (la disponibilité et le coût de la puissance de calcul) et l'évolution de la commande des robots (Whitney, 1985) ont permis l'évolution des robots versé interaction avec leur environnement. La commande employée pour l'interaction entre le robot et l'environnement utilise l'information des capteurs externes principalement de vision et de force, pour modifier la trajectoire du robot.

L'intégration des capteurs de vision ou de force dans la baie de commande du robot est très similaire. Dans ce paragraphe, l'implémentation du capteur d'effort est expliquée. La commande en force a pour objet de maîtriser les contacts mécaniques entre le robot et l'environnement. Dans sa formulation dite explicite, la commande en efforts consiste à asservir une valeur désirée des efforts de contact appliqués par un robot sur son environnement, cependant elle s'attache à programmer l'impédance du robot (Morel & Gangloff, 2005). La commande en effort peut avoir différentes architectures (Degoulange et al., 1993) qui peuvent être classées en deux approches : la commande sans consigne en effort et la commande avec consigne en effort (W. Khalil & E. Dombre, 1999).

La commande sans consigne en effort consiste à contrôler le robot en position. Mais un capteur d'effort est intégré pour mesurer les efforts dans l'effecteur. La commande sans consigne en effort est définie comme : compliance passive, commande par raideur active ou commande en impédance (W. Khalil & E. Dombre, 1999).

La compliance passive diminue les forces de contact entre le robot et l'environnement. Cette solution consiste à utiliser un dispositif déformable (jauge de déformation) entre l'effecteur et le robot. L'information obtenue du dispositif déformable est envoyée et ajoutée à l'information du robot, afin de corriger les erreurs dans la position (Drake, 1978).

La commande par raideur active consiste à modéliser le comportement du robot comme un ressort à raideur programmable (Salisbury, 1980). La matrice de raideur et la valeur du gain sont modifiables en fonction des directions commandées en position et en effort.

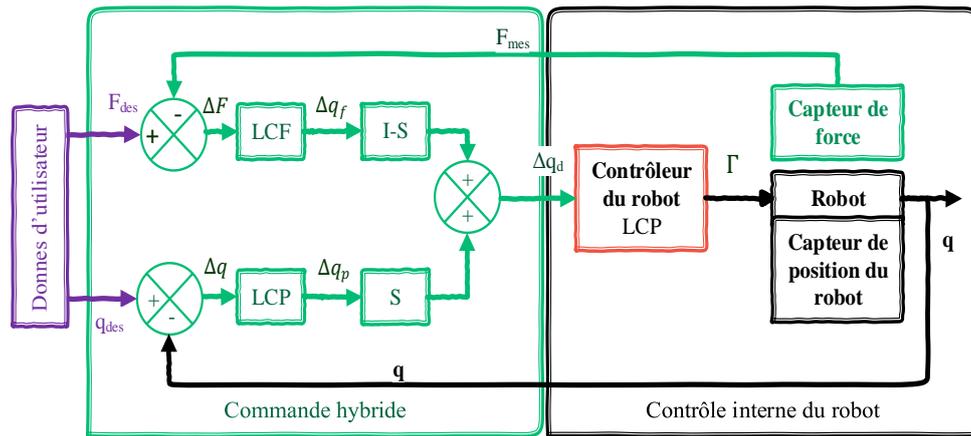
La commande en impédance consiste à établir une relation entre la position de l'effecteur et les efforts exercés par le robot sur son environnement (Hogan, 1987). Le robot est modélisé comme un système masse- amortissement- ressort.

Dans les commandes sans consigne en effort, les forces en contact ne sont pas directement mesurées par un capteur d'effort (Degoulange et al., 1993).

La commande avec consigne en effort utilise un capteur d'effort pour mesurer les forces en contact. L'asservissement de cette commande est en position et en force. La commande avec consigne en effort peut se décomposer en deux classes : la commande hybride et la commande hybride externe (W. Khalil & E. Dombre, 1999).

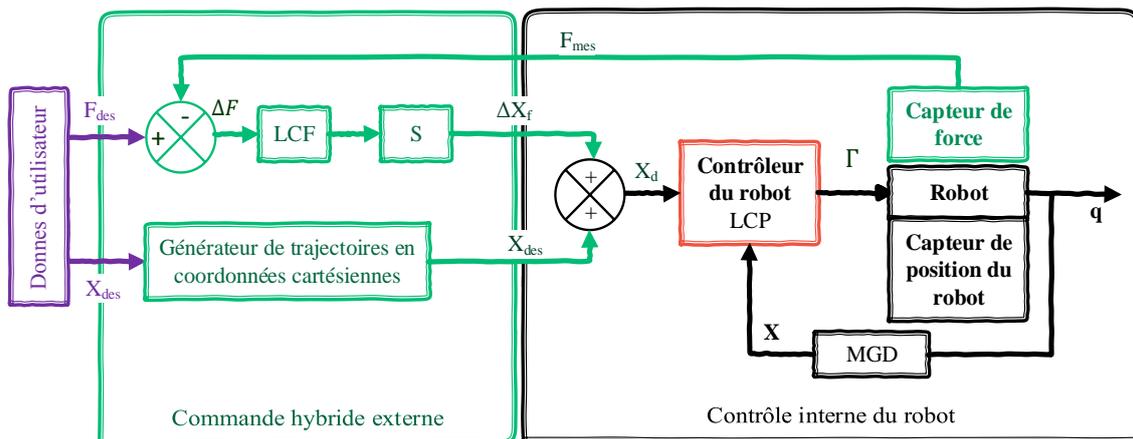
La commande hybride consiste à combiner les informations des capteurs externes et de la position, pour satisfaire les contraintes simultanées de la trajectoire (Raibert & Craig, 1981; Degoulange et al., 1993). Cette commande définit deux espaces orthogonaux et deux boucles de commande (Degoulange et al., 1993). Une matrice est utilisée pour sélectionner les directions commandées en position et celles commandées en effort. La boucle qui commande les directions en positions est implémentée par défaut dans la baie de commande du robot et la boucle qui commande les directions en force doit être implémentée en coordonnées articulaires (Degoulange et al., 1993). Les informations des sorties des capteurs sont sommées et envoyées directement au contrôleur du robot (Uhart, 2014; Uhart et al., 2014). Cet algorithme a un grand coût de calcul (Degoulange et al., 1993). La figure 5.2 montre le diagramme d'une commande hybride en force-position.

La commande hybride externe permet d'exécuter, en même temps, l'asservissement en position et l'asservissement basé sur les données des capteurs extéroceptifs. Cette commande a deux boucles de commande imbriquées (De Schutter & Van Brussel, 1988) : la boucle interne pour l'asservissement en position et la boucle externe qui correspond à l'asservissement basé sur les capteurs externes. La boucle interne utilise le modèle géométrique direct pour travailler en coordonnées cartésiennes (Degoulange et al., 1993). La sortie de la boucle externe est transformée en consigne de position, grâce à une loi de commande (W. Khalil & E. Dombre, 1999) et les axes du robot sont commandés, grâce au contrôleur interne du robot (Uhart, 2014). La figure 5.3 montre le diagramme d'une commande hybride externe en force.



LCP : loi de contrôle en position $\Delta q = \Delta q_{des} - q$: Position-orientation en coordonnées articulaires. S et I-S : Matrices de sélection. $\Delta q = \Delta q_{des} - q$: Position-orientation en coordonnées articulaires. $\Delta F = F_{des} - F_{mes}$: Force Γ : Vecteur de couples. q : Vecteur de position-orientation en coordonnées articulaires.

Figure 5.2 Commande hybride en force/position



LCP : loi de contrôle en position $X_d = \Delta X_f + X_{des}$: Position/orientation désirée en coordonnées cartésiennes. S : Matrice de sélection. Γ : Vecteur de couples. q : Vecteur de position-orientation en coordonnées articulaires. MGD : modèle géométrique direct

Figure 5.3 Commande hybride externe en force/position

5.2 Structure de la commande

Les baies de commande des robots industriels sont considérées comme des boîtes noires, puisque leurs caractéristiques ne sont pas connues (Uhart, 2014). La commande de type classique ne peut pas être implémentée sur une baie de commande du robot proposée par les fabricants. Elle peut être implémentée si une loi de contrôle est développée et adaptée pour commander le robot. Elle demande de déterminer tous les paramètres de chaque axe du robot (vitesse, correcteurs, position et orientation, capteurs de position...), et les paramètres de l'interaction. Ce type de commande est présenté dans (Mark W. Spong, 1989; Zheng, 1991;

Rodriguez-Angeles et al., 2004; Ollero Baturone, 2006; Peza Solís et al., 2009; Reyes Cortes, 2011; Makarov, 2013). La plupart de ces travaux utilisent des robots fabriqués par des laboratoires scientifiques, dont le contrôle interne est modifiable selon la spécification des tâches.

La commande hybride est compatible avec les baies de commande des robots. Elle impose de modifier le contrôleur et de réaliser la commande du robot avec le contrôleur de bas niveau. La commande hybride consiste à appliquer le principe de découpage des degrés de liberté du robot, afin que certains nombres de degrés de liberté agissent en position et que les autres suivent l'information des mesures des capteurs externes (Tan et al., 2002; Nganga-Kouya & Okou, 2009). Cette commande travaille en parallèle avec la commande en position, les informations d'effort et de position sont donc utilisées simultanément (Gangloff & Poignet, 2007). Cette commande présente un certain nombre d'inconvénients comme par exemple si le robot est commandé suivant une direction en effort, la position n'est pas asservie et une collision en amont du capteur ne pourra pas être compensée (Gangloff & Poignet, 2007). L'inconvénient de cette commande est le coût élevé de calcul (Degoulange et al., 1993).

La commande de type hybride externe est aussi compatible avec les baies de commande des robots. Elle permet d'ajouter des capteurs externes dans une boucle externe pour commander le robot sans modification du contrôleur interne du robot. Cette commande permet de travailler avec le contrôleur à plus haut niveau (Uhart, 2014). Le coût de calcul est réduit, car cette commande travaille en série ou en cascade (Gangloff & Poignet, 2007). Si la commande est basée sur les capteurs externes, la position peut être asservie en même temps, en conséquence les éléments dans l'environnement sont compensés (Mark W. Spong, 1989; Degoulange et al., 1993; Baeten et al., 2003; Ollero Baturone, 2006; Smits et al., 2006; Albu-Schäffer et al., 2007; Gangloff & Poignet, 2007; Mezouar et al., 2007; Prats et al., 2007; Reyes Cortes, 2011; Gallot et al., 2012; Gazeau & Zegloul, 2012; Long et al., 2013; Uhart, 2014; W. Khalil & E. Dombre, 1999), les auteurs expliquent dans leurs travaux, la commande des robots par dispositifs externes, comme les capteurs de force, de proximité, de vision, les capteurs laser, entre autres. Les robots utilisés dans ces recherches sont de type industriel, dans lesquels le contrôle interne du robot est non modifiable et inaccessible.

Dans le cadre de cette recherche, la tâche robotisée de dépose de fibres utilise un robot de type industriel, donc la baie de commande est non modifiable. En conséquence, seulement deux commandes peuvent être développées : la commande hybride et la commande hybride externe. Après l'analyse des avantages et des inconvénients de ces commandes, la commande hybride

externe représente la meilleure option dans ce cas d'étude. Parmi les avantages de la commande hybride externe, on relève que :

- L'asservissement en position articulaire est toujours actif, évitant ainsi des problèmes d'instabilité (Gangloff & Poignet, 2007). Cet asservissement se trouve à l'intérieur du contrôleur interne du robot.
- La définition des directions commandées en position et des directions commandées par des capteurs externes, est gérée par la matrice de sélection S (Tan et al., 2002)
- La commande hybride externe est composée de deux boucles de commande (Dombre et al., 2003) : la boucle en position et la boucle externe définie par les capteurs extéroceptifs.
- La boucle externe est hiérarchiquement supérieure à la boucle en position, donc la consigne d'interaction est toujours prioritaire par rapport à la consigne de position.
- La boucle externe traduit en déplacement les variations entre l'information désirée et l'information mesurée par des capteurs.
- Le robot maintient sa position (Uhart, 2014), sauf si la trajectoire du robot est modifiée ou si le robot reste en contact avec l'environnement (Gangloff & Poignet, 2007; Uhart, 2014)
- La loi de commande fonctionne avec des correcteurs de commande classique du type PID pour l'asservissement articulaire et du type Intégral pour la boucle externe en effort (Gangloff & Poignet, 2007).
- La facilité de mise en œuvre sur des contrôleurs industriels (Gangloff & Poignet, 2007).

5.3 Task Frame Formalism (TFF)

Le capteur d'effort est généralement fixé sur le poignet du robot entre l'effecteur et le bras manipulateur (Kröger et al., 2004; Uhart, 2014).

Le « Task Frame Formalism » (TFF) permet de travailler dans le repère de la tâche lorsque il existe une interaction physique (un contact entre l'effecteur et la pièce) (Bruyninckx & Schutter, 1996; Prats et al., 2012). Ce formalisme spécifie les actions souhaitées séparément pour chaque direction du repère orthogonal lié à l'outil (Baeten et al., 2003; Prats et al., 2012).

Dans le cas du placement de fibres robotisé, la tâche est définie avec une contrainte souhaitée, car l'effort de compactage (F_z) le long de l'axe Z doit être contrôlé, ainsi que la répartition des fibres sur le moule correspondant au couple (M_x) autour de l'axe X . Afin d'observer l'influence des corrections sur la dynamique du système, le capteur d'effort mesure toutes les composantes en position et en orientation. Le repère de la tâche (figure 5.4) est défini hors-ligne par la

génération de la trajectoire, en conséquence, le repère de l'effecteur et de la tâche sont redéfinies constamment, pendant toute la trajectoire.

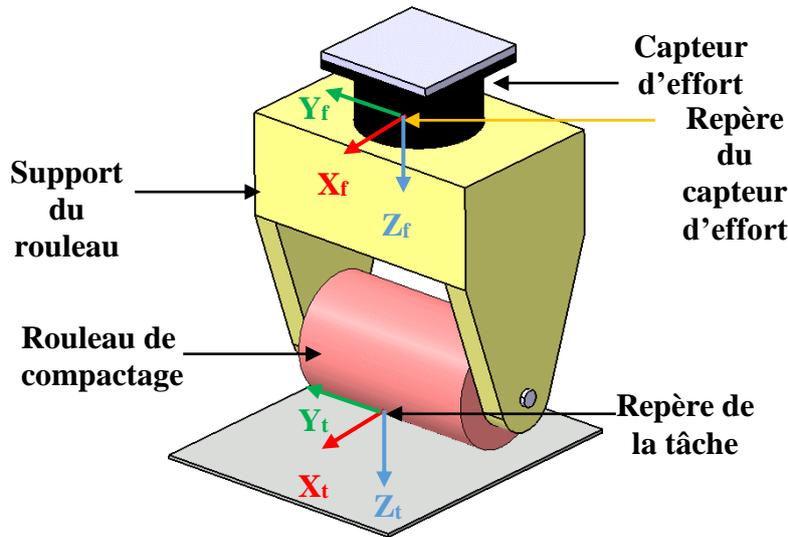


Figure 5.4 Définition du repère tâche

Grâce à la définition du repère de la tâche, la trajectoire peut être suivie en appliquant une force de compactage (F_z) et un couple (M_x) contrôlés. Les mouvements à suivre par la trajectoire sont calculés à partir de ce repère. Le contrôle du couple (M_x) est très important en raison du contact linéaire entre le rouleau de compactage et la surface du moule. Si l'orientation est incorrecte un couple peut apparaître autour de l'axe X , en conséquence, une mauvaise distribution de la force de compactage peut en résulter. En conséquence, les efforts et les couples mesurés au niveau du capteur d'effort doivent être exprimés au centre du contact linéaire (figure 5.5), pour connaître les efforts et les couples réels en utilisant la formule de transformation des efforts et des couples.

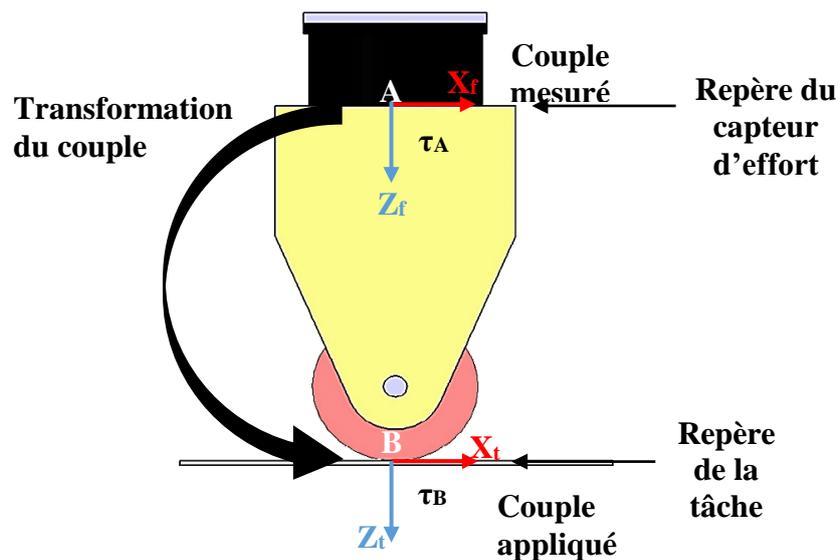


Figure 5.5 Transformation des forces et couples dans le repère tâche

La transformation du couple mesuré (A) au couple appliqué (B) est représentée par (5.1)

$$\tau_A = \begin{bmatrix} F_A \\ M_A \end{bmatrix}_A = \begin{bmatrix} F_A \\ M_A + BA \times F_A \end{bmatrix}_B \quad \tau_B = \begin{bmatrix} F_B \\ M_B \end{bmatrix}_B \quad (5.1)$$

Le principe fondamental de la statique appliqué se définit par (5.2)

$$\begin{bmatrix} F_A \\ M_A + BA \times F_A \end{bmatrix}_B + \tau_B = \begin{bmatrix} 0 \\ 0 \end{bmatrix}_B \quad (5.2)$$

Le couple appliqué peut s'écrire par (5.3)

$$\tau_B = \begin{bmatrix} F_B \\ M_B \end{bmatrix}_B = - \begin{bmatrix} F_A \\ M_A + BA \times F_A \end{bmatrix}_B \quad (5.3)$$

5.4 Loi de commande

La loi de commande est généralement définie par une expression mathématique, dont la mise en œuvre permet d'assurer le fonctionnement souhaité d'un système, afin de réaliser une tâche précise (Nganga-Kouya & Okou, 2009; Reyes Cortes, 2011). La connaissance du modèle dynamique du robot et de son expression mathématique est importante pour obtenir une bonne précision dans des tâches spécifiques (Reyes Cortes, 2011).

Dans le cas de cette recherche, nous utilisons une commande hybride externe, car elle est très utilisée pour son efficacité et son implémentation aisée (Nganga-Kouya & Okou, 2009). L'objectif de la commande hybride externe force-position consiste à séparer en deux sous problèmes la loi de commande : la loi de commande en position et la loi de commande en force (Paul, 1987; Hosoda et al., 1998; Tan et al., 2002; Dombre et al., 2003; Katsura et al., 2006; Villani & De Schutter, 2008; Nganga-Kouya & Okou, 2009; Reyes Cortes, 2011) et à linéariser les équations par pré-compensation (Di et al., 2000; M'Sirdi, 2012; Zhu et al., 2013). La conception de ce type de commande est donc basée sur une boucle de contrôle interne (« inner control loop ») qui compense les non linéarités dynamiques du robot et une boucle de contrôle externe (« outer control loop ») qui corrige la position en suivant la trajectoire et la force souhaitée et qui rejette les perturbations dues à l'interaction (contact) (Villani & De Schutter, 2008; Nganga-Kouya & Okou, 2009). Cette configuration permet de préserver les propriétés essentielles de la baie de commande du robot (Ozgoli & Taghirad, 2004, 2005). Ces boucles travaillent dans un espace de coordonnées cartésiennes (Paul, 1987) pour échanger l'information de position/orientation, surtout dans la sortie de la boucle externe qui est transformée en consigne de position et ajoutée à la boucle interne (W. Khalil & É. Dombre, 1999). La commande en position est régie par une expression mathématique qui résulte du modèle dynamique du robot (Nganga-Kouya & Okou, 2009), les erreurs dans la commande en position sont réduites par le contrôleur interne du robot. L'objectif de cette commande est de

positionner l'effecteur du robot dans un point précis et qui lui reste indéfiniment dans cette position (Reyes Cortes, 2011). Par contre, la commande en force doit garantir l'interaction et réguler la force de contact souhaitée (M.W. Spong et al., 2005; Craig, 2006; Reyes Cortes, 2011). Afin de minimiser l'erreur de force, la commande en force intègre des correcteurs de types intégral (I), proportionnel-dérive (PD), proportionnel-intégral (PI), proportionnel-intégral-dérive (PID),... (Paul, 1987; Duchemin, 2002; Duchemin et al., 2005; Reyes Cortes, 2011; Zhu et al., 2013).

5.4.1 Loi de commande en position des robots rigides

La loi de commande est développée à partir de (3.15) de la section 353.1.2.1, cette équation expose la dynamique générale d'un robot rigide en termes lagrangiens. Ce modèle dynamique décrit un système de n équations différentielles de second ordre non linéaires et couplées. La plupart des robots industriels sont asservis axe par axe avec une commande classique de type PID (Youbin et al., 1996; W. Khalil & É. Dombre, 1999; Uhart, 2014). La commande classique modélise les robots comme un système linéaire.

La loi de commande en coordonnées articulaires est définie par (5.4)

$$\Gamma_P = K_P(q^d - q) + K_D(\dot{q}^d - \dot{q}) + K_I \int_{t_0}^t (q^d - q) d\tau \quad (5.4)$$

Tel que :

\dot{q}^d, q^d soient les vitesses et les positions désirées dans l'espace articulaire,

K_P, K_D, K_I soient des matrices diagonales définies positives, de dimension $(n \times n)$ d'éléments génériques respectivement les gains Proportionnels (K_{Pj}), Dérivés (K_{Dj}) et Intégraux (K_{Ij}).

Lorsque les effets de la gravité sont compensés par construction mécanique, une loi de type PD (5.5) est stable pour une position désirée q^d fixe (Arimoto, 1984; Tomei, 1991; Nganga-Kouya & Okou, 2009).

$$\Gamma_P = K_P(q^d - q) + K_D(\dot{q}^d - \dot{q}) \quad (5.5)$$

Le modèle cinématique inverse défini par (3.13) est dérivé pour obtenir le terme d'accélération en coordonnées articulaires en fonction des coordonnées cartésiennes (5.6). Ce terme est substitué en (3.15), afin d'obtenir la loi de commande en coordonnées cartésiennes.

$$\ddot{q} = J^{-1}(\ddot{x} + \dot{J}\dot{x}) \quad (5.6)$$

Tel que :

\ddot{q} soit l'accélération en coordonnées articulaires,

J^{-1} soit l'inverse de Jacobienne,

$\dot{j} = \frac{dq}{dt}$ soit la dérivée temporelle de la Jacobienne,

\ddot{x}, \dot{x} soient l'accélération et la vitesse en coordonnées cartésiennes.

La loi de commande en coordonnées cartésiennes est représentée par (5.7).

$$\Gamma_P = K_P(X^d - X) + K_D(\dot{X}^d - \dot{X}) + K_I \int_{t_0}^t (X^d - X) d\tau \quad (5.7)$$

5.4.2 Loi de commande en force

La loi de commande en force est développée à partir de (3.12) et basée sur la linéarisation de la dynamique du robot (Jiang, 2008). La loi de commande en force en coordonnées cartésiennes est :

$$\Gamma_F = J^T \left[f^d + K_f(f^d - f) + K_{fD}\dot{X}^d + K_{fI} \int_{t_0}^t (f^d - f) d\tau \right] \quad (5.8)$$

J^T étant la matrice transposée de la Jacobienne.

Le terme dérivé de la commande en force est la vitesse cartésienne plutôt que l'effort, à cause du bruit de mesure associé au capteur.

5.4.3 Loi de commande hybride externe en position/force

La loi de commande hybride externe peut être exprimée en suivant une loi de commande par découplage non linéaire qui modélise le contact du robot avec l'environnement comme pour la commande en impédance. La commande en impédance indique la relation entre l'effort et la position de l'effecteur et régule l'impédance de sortie du robot. L'impédance en fonction de transfert Z en termes de Laplace est :

$$F(s) = Z(s)dX(s) \quad (5.9)$$

Tel que :

dX soit un déplacement opérationnel à partir d'une position nominale X_0 ,

F soit la force produite en réponse au déplacement dX .

Le robot est généralement considéré comme un système masse-ressort-amortisseur de deuxième ordre. Il se comporte comme un système mécanique d'impédance Z variable selon les phases de la tâche. La relation d'impédance du système mécanique masse-ressort (robot) est représentée par (5.10).

$$Z(s) = \Lambda s^2 + Bs + K \quad (5.10)$$

Tel que :

- Λ soit une matrice d'inertie,
- B soit une matrice d'amortissement,
- K soit une matrice de raideur.

En conséquence, la loi de commande découplée est représentée par (5.11).

$$\Gamma = J^T [M(q)w(t) + C(q, \dot{q}) + g(q) + f] \quad (5.11)$$

Tel que :

$w(t)$ soit le vecteur linéaire du contrôle.

Le vecteur linéaire du contrôle dans la commande hybride externe est défini par la somme de la loi de commande en position et de la loi de commande en force $w(t) = \Gamma_P + \Gamma_F$.

La commande hybride externe est représentée en coordonnées cartésiennes par (5.12).

$$\Lambda(\ddot{X}^d - \ddot{X}) + K_D(\dot{X}^d - \dot{X}) + K_P(X^d - X) + K_f(f^d - f) + K_{fi} \int_{t_0}^t (f^d - f) d\tau = 0 \quad (5.12)$$

Le gain K_{fi} représente le terme intégral de force. Ce terme force l'erreur en effort à zéro quelle que soit la consigne en position et permet aussi à la boucle en effort d'être hiérarchiquement supérieure à la boucle en position.

La loi de commande en position est régulée par la baie de commande du robot, mais la loi de commande en force doit être transformée du repère du capteur ou repère de la tâche, afin de minimiser les perturbations.

5.4.4 Correcteur PID

Dans le travail de Duchemin (Duchemin, 2002), l'auteur utilise un correcteur du type intégral (I) qui permet de corriger de façon simple et correcte la boucle en force de la commande hybride externe. Cette recherche est focalisée sur la tâche de dépose de fibres. En conséquence ce travail s'appuie dans la recherche précédent réalisée par Uhart (Uhart, 2014), l'auteur utilise un correcteur du type PID pour la même tâche.

La loi de commande en force peut être aussi représentée avec une expression générale du signal de sortie à temps continu d'un contrôleur PID :

$$u(t) = K_P e(t) + K_I \int_{t_0}^t e(\tau) d\tau + K_D \frac{de(t)}{dt} \quad (5.13)$$

Tel que :

- $u(t)$ soit la signal de sortie,
- $e(t)$ soit l'erreur entre la mesure et la consigne,
- K_P soit le gain proportionnel,

$K_I = \frac{K_P}{T_i}$ soit le gain intégral,

T_i soit la constante de temps intégrale,

$K_D = K_P T_D$ soit le gain dérivé,

T_D soit la constante de temps de dérivation.

La forme continue de (5.13) doit être discrétisée, car la mise en œuvre de ce système est réalisée par ordinateur et le temps d'échantillonnage doit être considéré. La discrétisation peut être calculée par différentes méthodes d'approximation dont :

- la dérivée des méthodes des rectangles, des trapèzes ou de Simpson,
- l'estimation de la dérivée première par une droite,
- l'approximation des trois dernières erreurs par une courbe parabolique ou cubique.

Dans le travail d'Uhart (Uhart, 2014), l'auteur utilise l'algorithme incrémental (velocity algorithm) pour calculer les gains du PID. Cet algorithme consiste à intégrer la vitesse de la variable à contrôler, donc le signal de contrôle est directement piloté par un intégrateur (Aström & Hagglund, 1995; Astrom, 2002). La forme incrémentale de l'algorithme PID est calculée par les variations du temps de sortie du contrôleur et les écarts sont additionnés (Astrom, 2002). Cet algorithme est :

$$\Delta u(t_k) = u(t_k) - u(t_{k-1}) = \Delta P(t_k) + \Delta I(t_k) + \Delta D(t_k) \quad (5.14)$$

Les écarts proportionnels, intégraux et dérivés sont calculés par (5.15), (5.16) et (5.17).

$$\Delta P(t_k) = P(t_k) - P(t_{k-1}) = K(e(t_k) - e(t_{k-1})) \quad (5.15)$$

$$\Delta I(t_k) = I(t_k) - I(t_{k-1}) = b_i(e(t_k) + e(t_{k-1})) \quad (5.16)$$

$$\Delta D(t_k) = D(t_k) - D(t_{k-1}) = -b_d(e(t_k) - 2e(t_{k-1}) + e(t_{k-2})) \quad (5.17)$$

Tels que :

$$b_i = \frac{K\Delta}{T_i},$$

$$b_d = \frac{KT_d}{\Delta}.$$

En remplaçant les écarts proportionnels (5.15), intégraux (5.16) et dérivés (5.17) dans (5.14), la solution obtenue est donnée par (5.18).

$$\begin{aligned} \Delta u(t_k) = u(t_k) - u(t_{k-1}) &= K \left[(e(t_k) - e(t_{k-1})) + \frac{\Delta}{T_i} e(t_k) + \frac{T_d}{\Delta} (e(t_k) - 2e(t_{k-1}) + e(t_{k-2})) \right] \\ &= K_0 e(t_k) + K_1 e(t_{k-1}) + K_2 e(t_{k-2}) \end{aligned} \quad (5.18)$$

Tels que :

$$K_0 = K_P \left(1 + \frac{\Delta}{T_i} + \frac{T_d}{\Delta} \right),$$

$$K_1 = -K_P \left(1 + \frac{2T_d}{\Delta} \right),$$

$$K_2 = K_P \left(\frac{T_d}{\Delta} \right).$$

A chaque instant de temps, l'écart de commande $\Delta u(t_k)$ est calculé et envoyé au robot. Les erreurs $e(t_k)$, $e(t_{k-1})$ et $e(t_{k-2})$ aux instants t_k , t_{k-1} , t_{k-2} sont indispensables pour le calculs des paramètres K_0 , K_1 , K_2 . Ces paramètres peuvent être réglables à partir des valeurs K_P , K_I et K_D .

5.4.5 Correcteur logique floue-PI

La loi de commande de logique floue produit une bonne performance lorsqu'elle est conçue pour une tâche spécifique, car elle traite l'information systématiquement pour trouver des solutions précises (Pires et al., 2002). La loi de commande hybride en force-position est employée principalement pour contrôler les limites de la force du robot (Huang et al., 2003), car elle peut s'ajuster à toutes les surfaces, à condition de connaître leur rigidité de la surface. Si cette rigidité n'a pas été prise en compte ou varie significativement, la performance est dégradée. La loi de commande de la force par logique floue possède la capacité d'être adaptable en ajustant les facteurs d'échelle de la loi de commande (Neto et al., 2010).

La loi de commande de logique floue est basée sur un système de connaissance qui consiste en règles « IF... THEN » (paragraphe 4.1.2.2) (Pires et al., 2002; Neto et al., 2010). Les règles de la logique floue sont conçues à partir du comportement réel du système, afin d'améliorer la robustesse du contrôleur (Morel & Bidaud, 1996). L'implémentation d'une loi de commande de logique floue doit respecter les étapes (Mendes et al., 2010; Neto et al., 2010) décrits dans le paragraphe 4.1.2.2.

Dans le cadre de cette recherche, l'entrée de la loi de commande en force est l'information collectée par le capteur d'effort et la sortie doit indiquer un déplacement en millimètres qui permet de déplacer le robot en respectant la force désirée.

Les variables d'entrée du système données par le capteur d'effort sont représentées par (5.19).

$$\begin{aligned} e_k &= f_{d_k} - f_{a_k} \\ de_k &= e_k - e_{k-1} \end{aligned} \quad (5.19)$$

Tel que :

e_k soit l'erreur des forces/couples,

de_k soit la dérivée de l'erreur,

f_{a_k} soit la force actuelle,

f_{d_k} soit la force désirée,

La loi de commande PI classique est représenté par (5.20)

$$u(t) = K_P e(t) + K_I \int_{t_0}^t e(\tau) d\tau \quad (5.20)$$

Tel que :

u soit le déplacement du robot,

K_P et K_I soient les coefficients proportionnel et intégral.

La figure 5.6 montre l'implémentation pratique d'une loi de commande en force par logique floue.

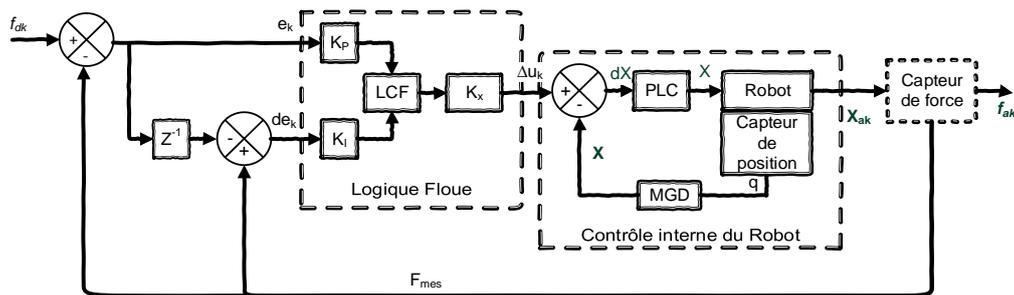


Figure 5.6 L'implémentation d'une loi de commande de la logique floue avec des correcteurs PI

La définition des groupes de règles « knowledge base » est composée de deux éléments : la base de données et les règles de la loi de commande. Ces règles sont interprétées par des étiquettes linguistiques. Dans les travaux de Neto et Mendes (Mendes et al., 2010; Neto et al., 2010), les auteurs utilisent les règles montrées dans le tableau 5.1 pour une loi de commande de logique floue PI.

Les règles de la loi de commande sont extraites de l'expérience pratique qui permet de contrôler une tâche spécifique (Neto et al., 2010) et elles peuvent être différentes selon la tâche à commander.

Tableau 5.1 Représentation des règles d'une loi de commande logique floue-PI

de \ e	NG	NM	NP	ZR	PP	PM	PG
PG	NG	NM	NP	ZR	PM	PG	PG
PM	NG	NG	NM	ZR	PM	PG	PG
PP	NG	NG	NP	ZR	PP	PG	PG
ZR	NG	NM	NP	ZR	PP	PM	PG
NP	NG	NG	NP	ZR	PP	PG	PG
NM	NG	NG	NM	ZR	PM	PG	PG
NG	NG	NG	NM	ZR	PP	PM	PG

Telle que :

PG représente la règle Positive Grande,

PM représente la règle Positive Moyenne,

PP représente la règle Positive Petite,
 Z représente la règle Zéro,
 NG représente la règle Négative Grande,
 NM représente la règle Négative Moyenne,
 NP représente la règle Négative Petite.

La défuzzification est réalisée à partir de la méthode du centre de la surface représentée par (5.21).

$$\Delta U = \frac{\sum_{i=1}^n \mu_i \Delta U_i}{\sum_{i=1}^n \mu_i} \quad (5.21)$$

Tel que :

μ_i soit un membre de la fonction qui prend des valeurs comprises entre zéro et un.

5.5 Sélection de la loi de commande

La sélection des correcteurs de la loi de commande est fondamentale pour garantir le fonctionnement souhaité et améliorer la précision du système. La loi de commande utilisée dans cette recherche est la loi de commande pour un système hybride externe en position/force. Cette commande est décrite dans le paragraphe 5.4. Elle intègre généralement des correcteurs de type PD, PI, PID, car la plupart des robots industriels sont asservis axe par axe avec une commande classique.

Les correcteurs de la loi de commande classique (PD, PI, PID) sont utilisés principalement dans des systèmes régulateurs dont la consigne est constante dans le temps. Les méthodes de calcul de ces correcteurs considèrent une solution unique. Cependant si la consigne est modifiée régulièrement dans le temps, la loi de commande classique possède des problèmes de stabilisation. Des lois de commande en logique floue sont proposées comme solutions à ces problèmes de stabilisation, car la logique floue possède la capacité d'être adaptable en ajustant les correcteurs à chaque modification de la consigne. Afin de considérer la meilleure régulation des correcteurs, nous réalisons la régulation pour la commande classique du type PID et la régulation pour la commande en logique floue du type PI.

5.5.1 La loi de commande classique PID

Les correcteurs PID de la loi de commande classique sont calculés par (5.18). Ces valeurs doivent être calculées selon la rigidité du système sous-amorti.

Dans un premier temps, nous calculons les valeurs des correcteurs PID de la loi de commande classique par (5.18), ensuite les valeurs sont implémentées dans la boucle d'effort avec un capteur à une dimension. Les résultats obtenus sont montrés dans la figure 5.7, ces résultats

démontrent que la loi de commande classique PID arrive à stabiliser le signal avec une erreur minimale pour différents forces appliquées.

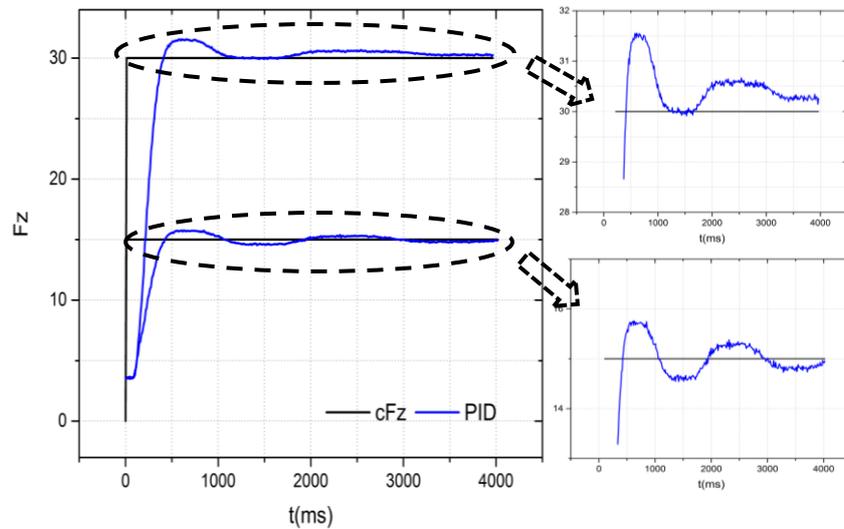


Figure 5.7 Résultats de la loi de commande classique PID calculés pour une surface plane avec un capteur à une dimension

Ensuite à l'expérimentation avec un capteur à dimension, nous vérifions le comportement du PID avec le rouleau de compactage et un capteur à six dimensions. La loi de commande reste instable, donc les valeurs des correcteurs PID calculées pour la première expérimentation ne sont pas valables pour notre système, à cause de la raideur rouleau de compactage. Les correcteurs de PID sont modifiés en prenant en compte la raideur du rouleau de compactage. Les valeurs pour les correcteurs PID de la loi de commande classique sont : $K_0=0,00884$, $K_1=0,017167$ et $K_2=0,008333$ pour une surface plane. Les résultats obtenus sont montrés dans la figure 5.8 pour le contact du point B sur une surface plane avec une force de consigne de 20 N et de 40 N. La figure 5.8 montre que la loi de commande est stabilisée dans le temps pour une force de consigne de 20 N, mais que pour une force de consigne de 40 N, la loi de commande possède une erreur comprise entre 1 N et 5 N sans arriver à la stabilisation.

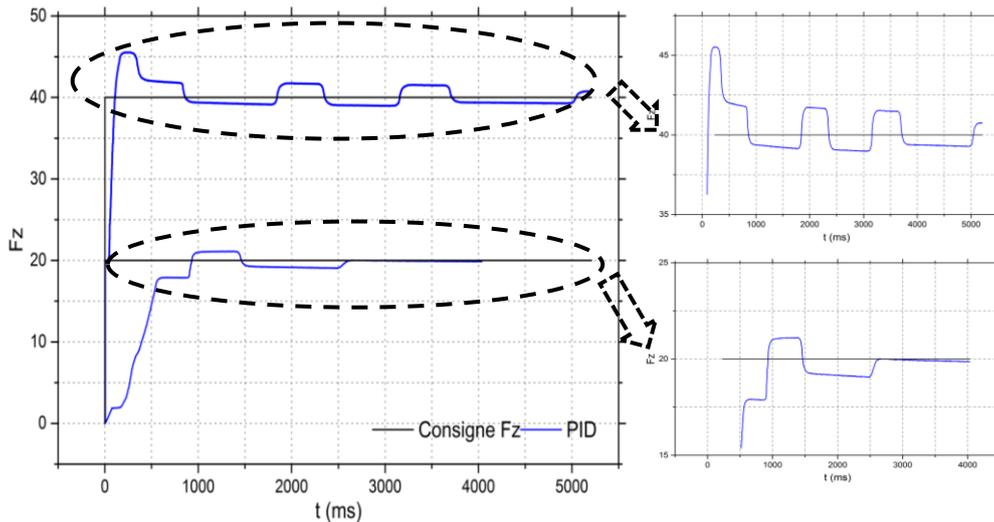


Figure 5.8 Résultats de la loi de commande classique PID calculés pour un contact au point B sur une surface plane avec un capteur à six dimensions

Nous réalisons un test pour évaluer le comportement du PID lorsque la surface de dépose présente des courbures. Nous utilisons pour ce test, le rouleau de compactage, le capteur à six dimensions et le surface considère comme convexe. La loi de commande reste instable, donc les valeurs des correcteurs PID calculées pour la surface plane ne sont pas satisfaisantes car la rigidité de la pièce est différente. Alors les paramètres pour les correcteurs PID de la loi de commande classique sont : $K_0=0,01$, $K_1=0,0175$ et $K_2=0,00751$ pour une surface convexe. Les résultats obtenus pour le contact du point B sur une surface convexe avec une force de consigne de 20 N et de 40 N sont montrés figure 5.9. Contrairement à l'expérimentation sur la surface plane, la loi de commande sur une surface convexe avec une force de consigne de 20 N possède une erreur comprise entre 0,25 N et 0,5 N sans arriver à la stabilisation. Par contre la loi de commande est stabilisée sur une surface convexe avec une force de consigne de 40 N. Ces résultats sont expliqués par les correcteurs de la commande classique qu'ils sont calculés selon la rigidité du système. Cette rigidité dépend de la force appliquée, des caractéristiques mécaniques du matériau et des caractéristiques géométriques de la pièce. Pour les deux surfaces nous avons appliqué les mêmes forces, par contre, les caractéristiques mécaniques et géométriques des moules sont différentes. Le matériau de la surface plane est en bois et de la surface convexe est en plastique, en conséquence la rigidité varie significativement et la performance de la loi de commande est dégradée.

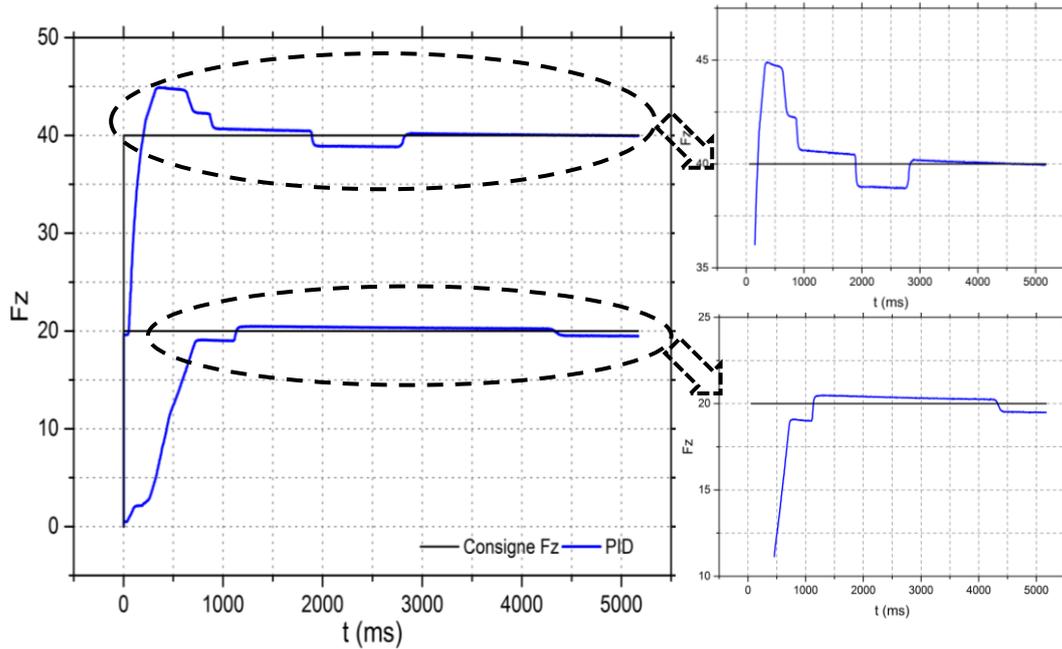


Figure 5.9 Résultats de la loi de commande classique PID calculés pour un contact au point B sur une surface convexe avec un capteur à six dimensions

Les correcteurs PID de la loi de commande classique doivent être calculés pour chaque consigne de force et pour chaque surface.

5.5.2 La loi de commande en logique floue PI

La loi de commande en logique floue PI est réalisée à partir des étapes décrites dans la section 5.4.5. Les entrées sont définies par le capteur d'effort à partir de (5.19) et les tables de vérité sont montrées par le tableau 5.2.

Tableau 5.2 Tables de vérité du contrôleur en logique floue PI

PG	PM	P	Z	N	NM	NG
0,05	0,01	0,007	0,0	0,007	0,01	0,05

Les règles d'apprentissage de la loi de commande en logique floue sont représentées graphiquement par la figure 5.10.

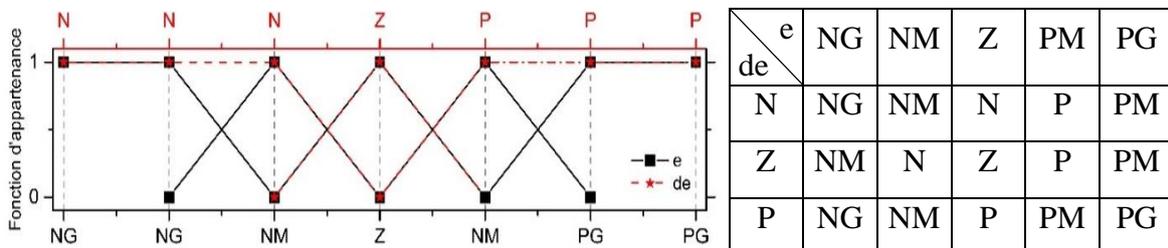


Figure 5.10 Représentation graphique des règles d'apprentissage et la mémoire associative floue (FAM)

La défuzzification est réalisée à partir de la méthode du centre de la surface et représentée par (5.21).

Nous réalisons les mêmes tests que pour la loi de commande classique, alors pour le premier test, les valeurs des correcteurs PI sont calculées par la méthode de logique floue et ensuite les valeurs sont implémentées dans la boucle d'effort avec un capteur à une dimension. Les résultats obtenus sont montrés dans la figure 5.11, ces résultats démontrent que la loi de commande en logique floue PI arrive à stabiliser le signal avec une erreur minimale pour différents forces appliquées.

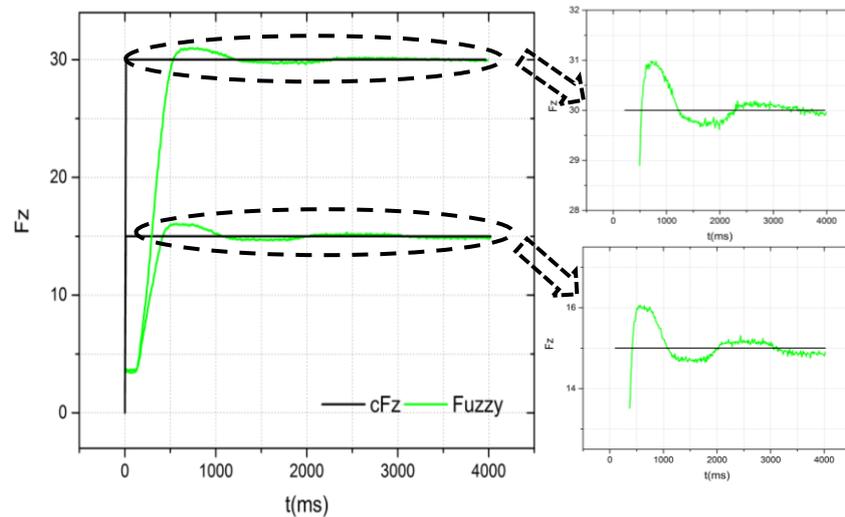


Figure 5.11 Résultats la loi de commande en logique floue PI calculés pour une surface plane avec un capteur à une dimension

Ensuite, nous réalisons un test dans notre système qui inclut le rouleau de compactage et un capteur à six dimensions. Les résultats obtenus sont montrés dans la figure 5.12 pour le contact du point B sur une surface plane avec une force de consigne de 20 N et de 40 N. La loi de commande en logique floue PI possède une erreur comprise entre 1 N et 2 N avant la stabilisation.

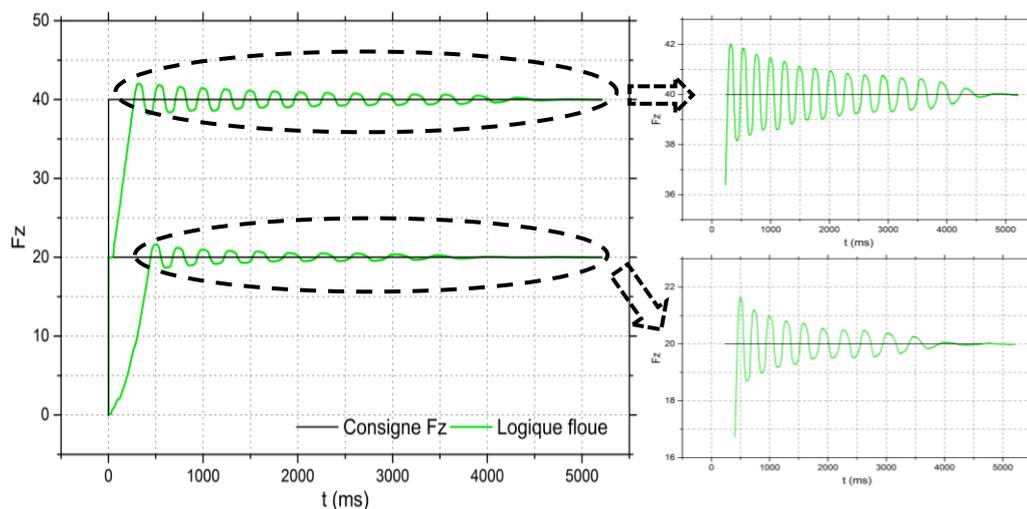


Figure 5.12 Résultats de la loi de commande en logique floue PI pour un contact au point B sur une surface plane

Finalement, nous réalisons un test pour évaluer le comportement sur la surface de dépose convexe avec le capteur à six dimensions. Les résultats obtenus pour le contact du point B sur une surface convexe avec une force de consigne à 20 N et à 40 N sont exposés dans la figure 5.13. La loi de commande en logique floue PI possède une erreur comprise entre 0,5 N et 1 N avant de la stabilisation. La loi de commande en logique floue PI possède une bonne performance, car elle se stabilise pour les deux forces de consigne et pour les deux surfaces.

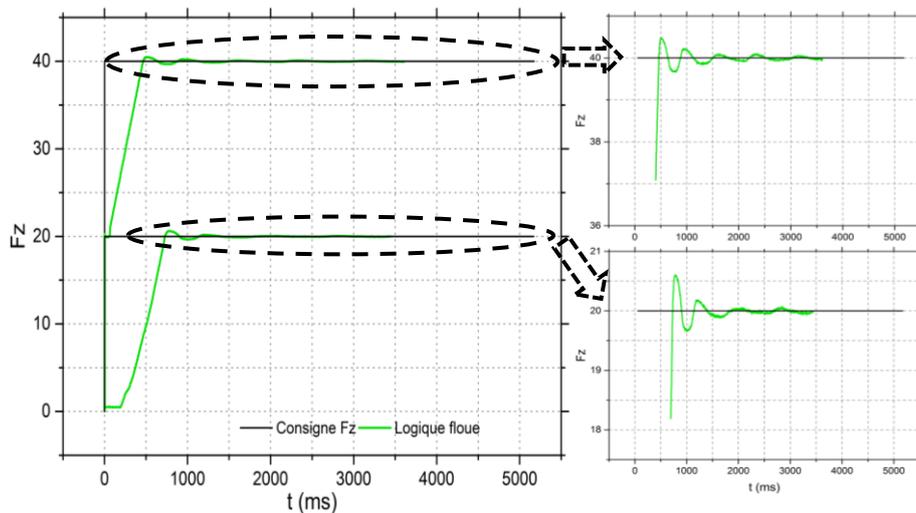


Figure 5.13 Résultats de la loi de commande en logique floue PI pour un contact au point B sur une surface convexe

5.5.3 Bilan

Les résultats obtenus pour la loi de commande classique PID et en logique floue PI sont comparés et montrés dans la figure 5.14 pour le capteur à une dimension, dans la figure 5.15 pour une surface plane avec un capteur à six dimensions, dans la figure 5.16 pour une surface convexe avec un capteur à six dimensions. Ces résultats montrent que la loi de commande en logique floue PI se stabilise rapidement pour tous les cas exposés, car elle réalise un réglage dynamique des correcteurs selon la force de consigne et la surface de contact. Par contre, avec la loi de commande classique PID, on doit calculer et régler les correcteurs pour chaque consigne de force et chaque surface pour se stabiliser.

La loi de commande classique PID est idéale lorsque la consigne en force et la surface sont constantes. Dans le cas contraire, une commande qui calcule les correcteurs dynamiquement doit être implémentée.

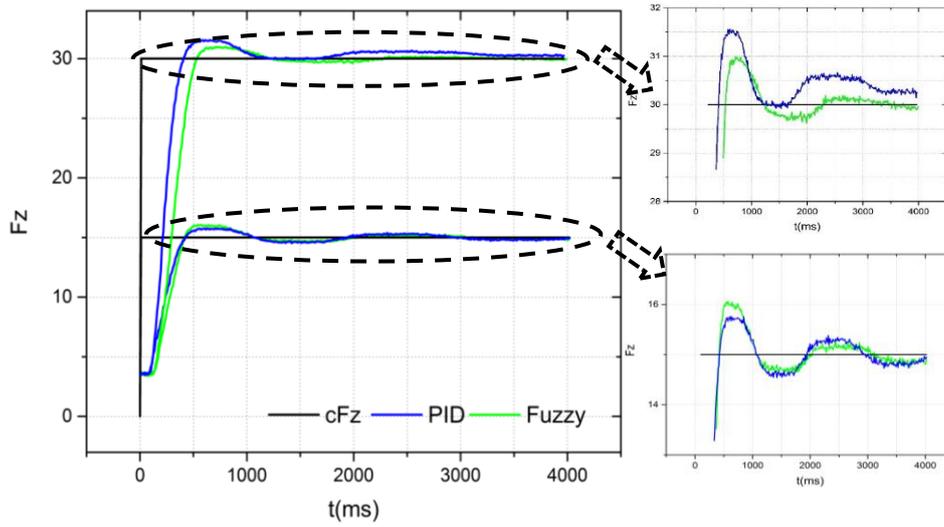


Figure 5.14 Comparaison des résultats de la loi de commande pour une surface plane

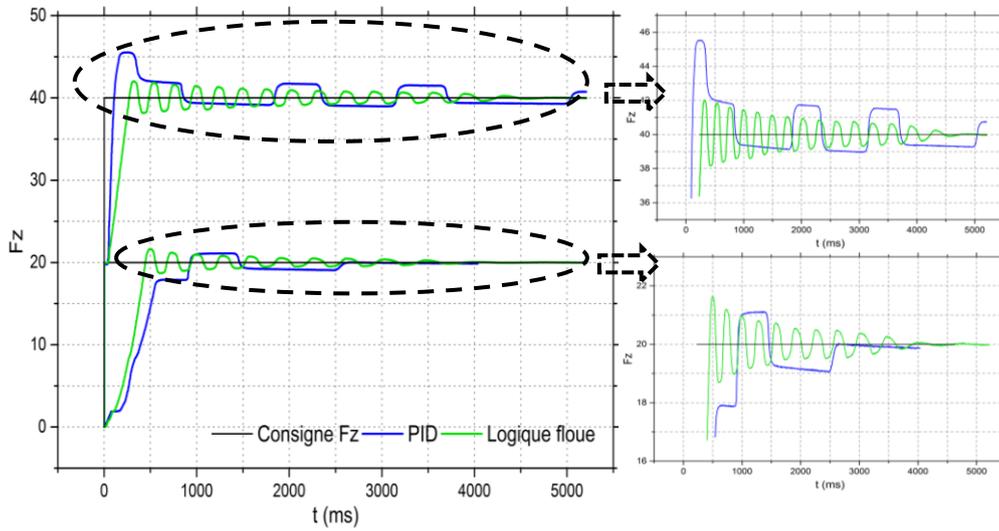


Figure 5.15 Comparaison des résultats de la loi de commande pour une surface plane

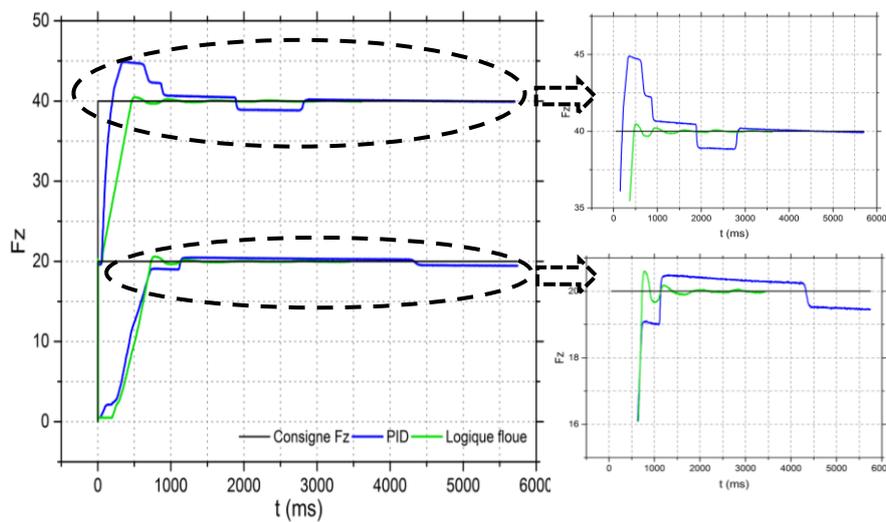


Figure 5.16 Comparaison des résultats de la loi de commande pour une surface convexe

5.6 Conclusion

L'analyse des différents types de commande nous a permis de connaître les commandes utilisées fréquemment pour commander la position/orientation du robot dont la commande classique ou la commande en interaction entre le robot et l'environnement. La commande classique qui considère les efforts externes et qui peut corriger la position en suivant l'interaction entre le robot et l'environnement doit développer une loi de contrôle adaptée au robot pour commander la position, donc cette commande exclut ou modifie la baie de commande du robot. Par contre, la commande en interaction ajoute une loi de commande externe à la baie de commande du robot.

Dans cette recherche, la tâche robotisée de déplacement des fibres est en contact avec le moule de dépose. Nous pouvons utiliser les deux commandes. Par contre, tous les composants externes de cette tâche (système de chauffe, système de coupage) sont intégrés à la baie de commande du robot, en conséquence, la commande à utiliser pour cette tâche est celle en interaction. Ce type de commande intègre des capteurs extéroceptifs principalement de vision et force pour corriger la trajectoire du robot selon les flexibilités du robot ou de l'environnement. Dans le cas de cette recherche, les flexibilités se trouvent principalement dans l'outil, comme nous les avons définies dans le chapitre 3. Nous nous sommes intéressés au rapport force-déformation de l'outil, le capteur utilisé est un capteur d'effort. Les solutions proposées pour la commande en force sont divisées en deux approches : la commande sans consigne en effort et la commande avec consigne en effort. Pour les deux commandes un capteur en effort est intégré. Dans la première approche, le capteur modifie la valeur de référence en force. Cette valeur de référence peut être modifiée par trois méthodes : la compliance passive, la commande par raideur active et la commande en impédance. Dans la deuxième approche, le capteur mesure les forces en contact pour modifier la trajectoire du robot selon la valeur de référence en force. Cette approche est classée en deux types de commande : la commande hybride et la commande hybride externe. Ces commandes combinent les informations des capteurs extéroceptifs et la position du robot, la différence entre ces commandes est principalement le temps de calcul. Pour la commande hybride, les informations du capteur en force travaillent en parallèle avec l'information de la position/orientation du robot, cette configuration augmente le temps de calcul. Pour la commande hybride externe, les informations du capteur en force et l'information de la position/orientation du robot s'exécutent en série ou en cascade, ce type de configuration réduit le temps de calcul.

Parmi les types de commande développés dans ce chapitre, nous avons choisi de travailler avec une commande hybride externe, car elle permet de travailler avec le robot sans modifier la baie de commande, d'intégrer des capteurs pour réduire les erreurs dans la trajectoire dues aux flexibilités, de commander certaines directions en position et d'autres en force et de travailler avec une loi de commande qui fonctionne avec des correcteurs de commande classique.

Le capteur d'effort est souvent positionné entre le robot et l'outil, afin de mesurer les forces d'interaction. Les forces mesurées se trouvent dans le repère du capteur, donc ces forces ne représentent pas les forces en interaction ou en contact. Afin d'obtenir les forces en interaction, les forces mesurées doivent être transportées au centre du contact linéaire à partir de (5.1), (5.2) et (5.3).

Pour garantir le fonctionnement souhaité de la boucle de commande externe (capteur externe) et la boucle de commande interne (robot), et ainsi pour obtenir une bonne précision dans le déplacement des fibres, l'analyse des lois de commandes est effectuée. La commande hybride est découpée en deux sous-systèmes : la loi de commande en position et la loi de commande en force. Ces lois utilisent des correcteurs de la commande classique du type PID, afin de réduire les perturbations. La loi de commande en position est définie par (5.7) et la loi de commande en force est représentée par (5.8). En conséquence, la loi de commande hybride externe est représentée par (5.13). Néanmoins, la loi de commande en position est régulée par la baie de commande du robot. Par conséquent, nous devons surveiller la loi de commande en force pour minimiser les perturbations dues à l'interaction.

Deux méthodes sont présentées pour le calcul des correcteurs de la loi de commande en force soit : les correcteurs de la commande classique PID (section 5.4.4) et les correcteurs de logique floue PI (section 5.4.5). Pour la commande classique, la régulation est réalisée à partir du calcul de ces correcteurs, par contre cette régulation est effectuée pour une seule solution. Pour la commande en logique floue, la régulation est réalisée à partir des règles linguistiques qui apportent des solutions singulières.

L'expérimentation réalisée nous permet de considérer la meilleure loi de commande en force. La loi de commande qui permet de garantir une stabilisation pour toutes surfaces est la loi de commande en logique floue PI, comme les résultats sont montrés dans la figure 5.15 et la figure 5.16.

CHAPITRE

6

Approche proposée

Sommaire

6.1 Commande en pression	103
6.2 Point dynamique de centrage de l'outil	104
6.3 Génération de trajectoire en position/force-pression en modifiant le TCP	105
6.4 Dispositif expérimental	106
_Toc523084735	
6.4.2 Robot	108
6.4.3 Système de communication robot capteurs externes (RSI)	109
6.4.4 Moule.....	110
6.4.5 Procédure de tests	112
6.5 Génération de trajectoire en pression avec prise en compte des flexibilités de l'outil ...	113
6.5.1 Commande hybride en position/force-pression.....	113
6.5.2 Point dynamique de centrage de l'outil	117
6.6 Conclusion.....	122

Dans ce chapitre nous décrivons, les solutions proposées pour résoudre les problèmes d'imprécision du robot lorsqu'il réalise le procédé de dépose de fibres.

La dépose fibre robotisée doit garantir la cohésion entre couches et respecter le processus de fabrication. Pour réaliser une bonne cohésion entre plis, cette tâche a besoin d'appliquer une force homogène sur toute la pièce. Actuellement, ce procédé considère que le rouleau de compactage est complètement en contact avec la surface et que la force de compactage est constante le long de la pièce. Ces hypothèses sont seulement valables lorsque la pièce est plane ou dispose de faibles courbures sans variation dans la surface. Malgré cela, le procédé de dépose de fibres doit fabriquer des pièces avec une importante courbure perceptible lors des passages

de coin et avec une importante variation de la surface appréciable dans la réduction de la surface lors de la dépose des fibres à 45 degrés. Afin de maintenir un compactage homogène des fibres le long de la pièce, nous proposons de réaliser une commande en pression de compactage, au lieu d'une commande en force de compactage. Ainsi, nous proposons l'application d'un point de centrage dynamique de l'outil (DTCP) en suivant les flexibilités de l'outil, afin de positionner correctement l'outil lors de passage de zones anguleuses ou de grandes courbes.

6.1 Commande en pression

La commande en pression considère principalement les variations dans la surface. Afin de maintenir un compactage homogène, la force doit être modifiée selon ces variations. En conséquence, la force de compactage est fonction de la surface de contact. Cette relation est représentée par une pression de compactage qui est symbolisée par (6.1).

$$P = \frac{F}{A} \quad (6.1)$$

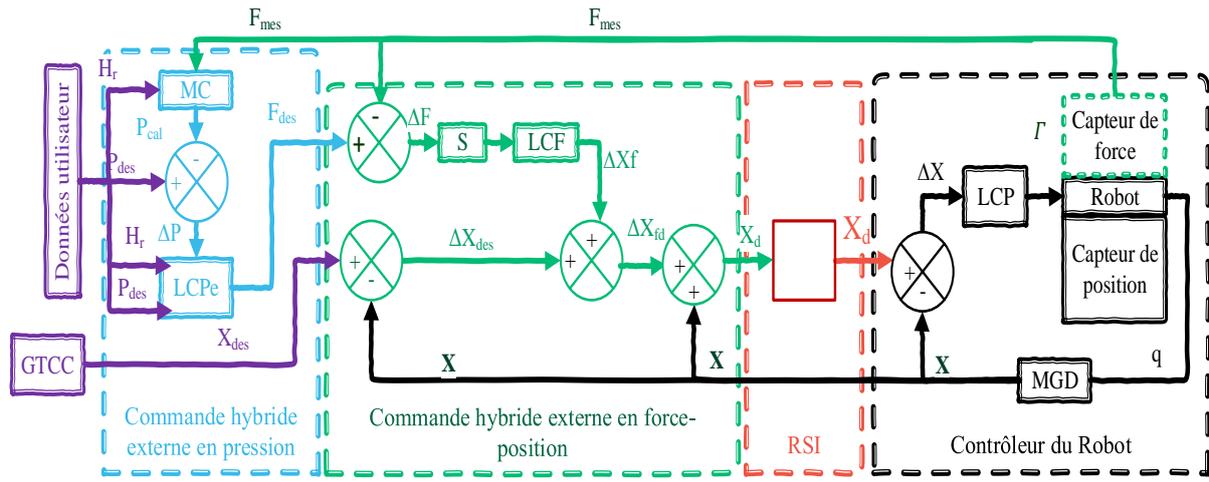
Tel que :

P soit la pression de compactage,

F soit la force de compactage,

A soit la surface de contact.

Cette recherche propose de prendre en compte la pression de compactage et de l'intégrer au travail précédemment développé par (Uhart, 2014). Ce travail considère un générateur de consigne en effort qui améliore la tâche de dépose sur des surfaces planes, néanmoins il ne considère pas les changements de la surface du moule. Pour établir un asservissement en pression, nous proposons l'utilisation d'une commande hybride externe (figure 6.1). La génération de consigne en pression est basée sur le générateur d'effort développé (Uhart, 2014). Ce dernier utilise un retour du capteur d'effort pour calculer la consigne en pression. Le capteur d'effort mesure les forces exercées sur et par le robot (principe d'action/réaction) (F_{mes}) qui sont les forces résultantes du contact entre l'outil (rouleau de compactage) et le moule pendant la dépose des fibres. La génération d'effort suppose que l'interaction reste constante, mais le générateur en pression définit la surface de contact du rouleau (H_r) grâce à la CAO de la pièce et à la programmation hors ligne. En conséquence le module de calcul (MC) détermine la pression à partir de la force mesurée (F_{mes}) et la surface de contact (H_r).



H_r : Surface de contact du rouleau, P_{des} : Pression désirée. MC : module de calcul $P_{cal}(F_{mes}, H_r)$. $\Delta P = P_{des} - P_{cal}$: variation de pression. LCPe : Loi de contrôle en pression. $\Delta F = F_{des} - F_{mes}$: variation de force. LCF : loi de contrôle en force. $\Delta X_{des} = X_{des} - X$: variation de la position-orientation désirées. LCP : loi de contrôle en Position. $\Delta X_{fp} = \Delta X_f + \Delta X_p$ variation de la position-orientation désirées pour la commande hybride. S : Matrice de décision. Γ : Vecteur de couples. q : Vecteur de position-orientation

Figure 6.1 Commande hybride externe en position/force-pression

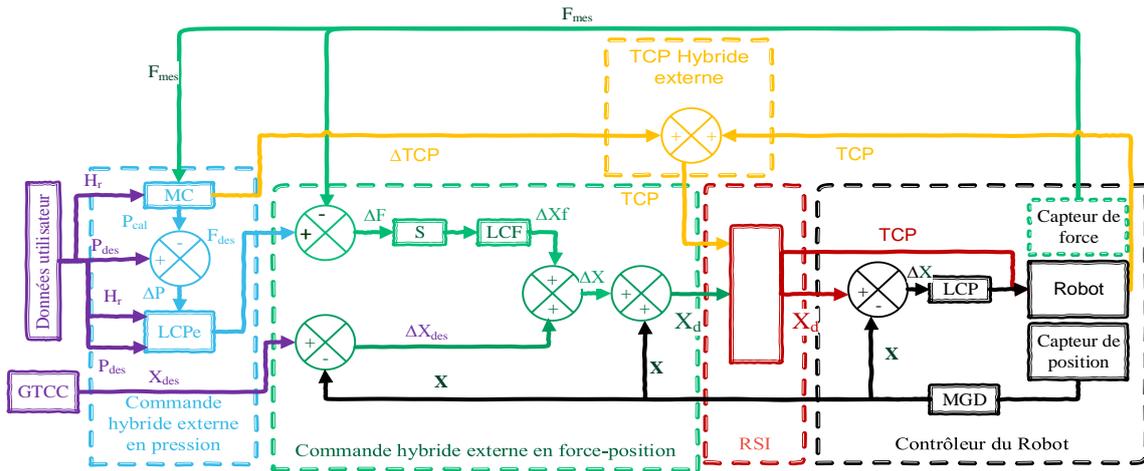
6.2 Point dynamique de centrage de l'outil

La plupart des tâches industrielles robotisées considèrent que le robot et l'effecteur sont des composants rigides. Dans ce cas, le point de centrage de l'outil (TCP) est fixé à une seule position/orientation, car la géométrie du robot et de l'outil ne change pas. Par contre, la tâche de dépose de fibres requiert que l'outil de dépose soit en contact avec le moule (l'environnement). Ce contact provoque des flexibilités au niveau de l'outil (rouleau de compactage) et/ou de la chaîne cinématique (articulations du robot principalement).

Comme nous l'avons décrit dans le paragraphe 3.2, les flexibilités de l'outil génèrent des imprécisions dans la trajectoire du robot, principalement pour le mauvais calibrage du TCP et les changements dans la géométrie de l'outil lors d'un contact avec l'environnement.

Afin de prendre en compte la déformation du rouleau de compactage, nous proposons la modification du TCP en temps réel que nous appelons point de centrage dynamique de l'outil (DTCP pour les sigles en anglais).

Le TCP du robot est défini à partir des diverses méthodes de calibrage montrées dans le paragraphe □. La définition du TCP peut être réalisée aussi manuellement à condition de connaître les dimensions précises de l'outil. Dans le cas de cette recherche, la définition du TCP initial à partir des méthodes de calibrage, considère que la force de contact doit être nulle, de ce fait l'outil est considéré comme rigide. D'un autre côté, la définition du TCP initial est



H_r : Surface de contact du rouleau, P_{des} : Pression désirée. MC : module de calcul Δ_z (H_r, F_{mes}) et P_{cal} (H_r, F_{mes}). $\Delta P = P_{des} - P_{cal}$: variation de pression. LCPe : Loi de contrôle en pression. $\Delta F = F_{des} - F_{mes}$: variation de force. LCF : loi de contrôle en force. $\Delta X_{des} = X_{des} - X$: variation de la position-orientation désirée. LCP : loi de contrôle en Position. $\Delta X_{fp} = \Delta X_f + \Delta X_p$ variation de la position-orientation désirée pour la commande hybride. S : Matrice de décision. Γ : Vecteur de couples. q : Vecteur de position-orientation. $\Delta TCP_z = TCP_z - \Delta_z$: variation de TCP dans l'axe Z. DTCP = $\Delta TCP_z + TCP$: TCP calculé

Figure 6.3 Commande en position/force-pression en modifiant le TCP

Dans cette approche, le module de calcul (MC) évalue la déformation dans l'outil et estime la pression de compactage. Cette déformation permet de modifier le TCP selon les flexibilités de l'outil. La pression permet de corriger la trajectoire de l'outil selon les efforts mesurés par le capteur d'effort et la surface de contact extraite du modèle CAO.

6.4 Dispositif expérimental

Le dispositif expérimental est composé d'un robot, d'un capteur d'effort, d'un outil de dépôt, d'un système de communication entre le robot et les capteurs et d'un moule (figure 6.4). Ces éléments sont décrits dans les paragraphes suivantes.

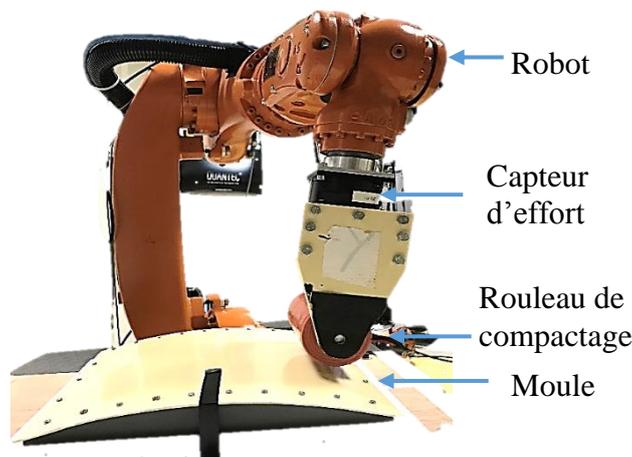


Figure 6.4 Dispositif expérimental

6.4.1 Capteur d'effort

La mesure des efforts permet de connaître l'interaction entre le robot et la surface. Cette mesure permet de réaliser au robot une action (Craig, 2006) selon la tâche. Les capteurs d'effort sont fréquemment appelés cellules de charge. Celles-ci sont capables de mesurer des forces en compression, en traction ou en cisaillement. Pour réaliser la sélection de ce capteur nous avons étudié le fonctionnement des capteurs d'effort à jauges de contraintes et des capteurs d'effort piézo-électriques qui est exposé dans la section A.3.

- **Sélection de capteur**

Nous sélectionnons un capteur d'effort équipés jauges de contraintes pour notre travail, car nous avons besoin de mesurer principalement trois composants de force d'axes différents (F_x , F_y , F_z). En conséquence, le capteur à utiliser doit être un capteur multi-axes.

Le capteur disponible dans la cellule robotisée de l'ESTIA est un capteur ATI FT-Gamma. Les caractéristiques techniques de ce capteur sont extraites de la documentation d'ATI Industrial Automation (Automation, 2016) et sont :

- Les plages de mesure pour les forces sont : F_x , $F_y = \pm 32$ N et $F_z = \pm 100$ N,
- Les plages de mesure pour les couples sont : M_x , M_y et $M_z = \pm 2,5$ Nm,
- Ce capteur est du type pan-cake et son fonctionnement est basé sur des jauges de contrainte intégrées dans un pont de Wheatstone,
- Le transducteur est une structure monolithique en acier. Selon le type de transducteur, la plage de la température d'opération est comprise entre 0 °C et 60 °C, entre 0 °C et 70 °C ou entre -5 °C et 105 °C,
- La fréquence de résonance pour F_x , F_y et M_z est de 1400 Hz et pour F_z , T_x et T_y est de 2000 Hz.
- Les caractéristiques physiques sont :
Masse = 0,255 kg,
Diamètre = 75,4 mm,
Épaisseur = 33,3 mm.

Les logiciels disponibles pour le développement de notre application sont :

- ATICombinedDAQFT,
- ATIDAQDontnetDemo,
- VisualC-Sample.

Ce capteur est déjà utilisé pour différentes applications, telles que les processus productifs comme la dépose de fibres (Gangloff & Poignet, 2007; Zengxi & Hui, 2008; Long et al., 2013, 2014; Uhart, 2014) et dans le cadre médical comme la chirurgie (Zemiti, 2005).

Ce capteur est fixé sur le poignet du robot entre l'effecteur et le bras manipulateur (figure 6.4). L'intégration du capteur d'effort dans la baie de commande du robot est mise en œuvre par le module RSI (Robot Sensor Interface).

6.4.2 Robot

Comme décrit dans le paragraphe 3.1.1.1, le robot à disposition pour réaliser notre expérimentation est le robot KUKA KR6 arc de la cellule robotisée à l'ESTIA. Ce robot est plus petit que le robot KUKA KR240 de la cellule robotisée à CompositAdour, par contre, ils possèdent tous deux la même architecture géométrique. Ces robots se différencient par les dimensions de leur corps (tableau 6.1) et de leurs paramètres inertiels. Le robot de la cellule robotisée à l'ESTIA ne comporte pas de rail linéaire. La modélisation de cette cellule en termes symboliques est décrite dans le chapitre 3. La modélisation du robot avec un rail linéaire est décrite dans (Uhart, 2014). Ces robots possèdent aussi la même baie de commande (KRC2). Ces baies de commande permettent l'incorporation du module RSI qui intègre des capteurs externes et réalise la communication avec eux.

Tableau 6.1 Paramètres dimensionnels des robots

Dimensions (mm)	KR240	KR6
A	230	115
B	1300	670
C	55	35
D	1250	680
E	750	675
F	350	260

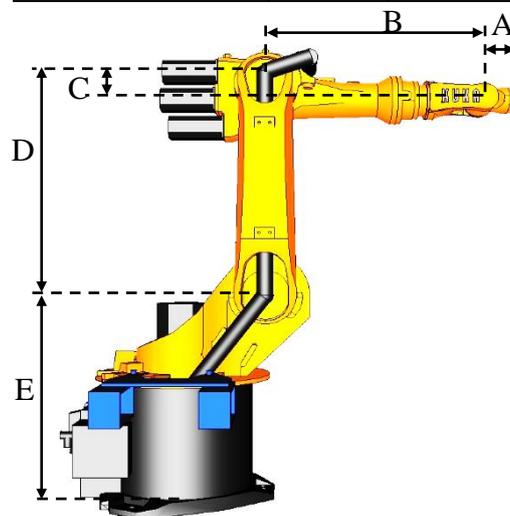


Figure 6.5 Dimensions d'un robot

6.4.3 Système de communication robot capteurs externes (RSI)

La programmation des robots KUKA est réalisée dans un langage dédié connu comme KUKA Robot Language (KRL) (KUKA, 2003; Nyman & Edberg, 2010; Y. Li, 2011; Tovar-Arriaga et al., 2012; Kostal, 2014; Sanfilippo et al., 2014). Ce langage permet de programmer et de définir les mouvements d'un robot (Y. Li, 2011; Kostal, 2014), la boucle du programme, les fonctions spéciales, les sous-programmes (Kostal, 2014), les variables, l'interaction avec des outils et les entrées/sorties des capteurs proprioceptifs (Sanfilippo et al., 2014). Par contre, l'intégration des capteurs extéroceptifs à ce langage reste difficile, car il travaille avec des opérations mathématiques basiques (Sanfilippo et al., 2014).

Afin de réaliser l'intégration des capteurs extéroceptifs à la baie de commande du robot, le fabricant propose l'utilisation du module RSI. Le module RSI est un logiciel qui permet de modifier l'exécution du programme ou le mouvement du robot selon les données des capteurs extéroceptifs (Y. Li, 2011). Ce module RSI travaille en temps réel. L'échange des données est transmis par des trames du type XML via le protocole TCP/IP (Transmission Control Protocol/Internet Protocol) pour la baie de commande KRC2 et via le protocole UDP/IP (User Datagram Protocol/ Internet Protocol) pour la baie de commande KRC4 (Nyman & Edberg, 2010). Cet échange des données est réalisée de façon cyclique (Hirzinger et al., 2005; Majdani et al., 2009) avec un cycle d'interpolation de 4 ms pour la baie de commande KRC4 et 12 ms pour la baie de commande KRC2. Ce cycle est appelé IPO (Input Process Output).

La baie de commande du robot envoie au système externe les informations du robot (la position, les angles des axes, le mode d'opération...) par le module RSI et reçoit les informations du système externe (la commande de contrôle, les paramètres de mouvements, la correction de la trajectoire...) chaque 4 ms ou 12 ms par le même module.

Dans le cadre de cette recherche, nous utilisons le module RSI avec une communication Ethernet, car le robot peut modifier ses mouvements par rapport au capteur externe de force avec une commande en pression. La baie de commande utilisée est de type KRC2. La transmission des données entre celle-ci et les capteurs externes doit être réalisée par protocole TCP/IP avec une trame XML dans une période d'échantillonnage de 12 ms. Si le cycle IPO dépasse les 12 ms, les données sont perdues. Après un nombre important de données perdues, le contrôleur du robot reprend la commande.

La communication par protocole TCP/IP entre la baie de commande et les capteurs externes est du type Client-Serveur-Client, dont les clients sont la baie de commande du robot et le système externe (capteurs) et dont le serveur est un intermédiaire entre les clients qui permet l'échange de l'information. Afin de réduire la perte des données (Uhart, 2014) entre les clients et le

serveur, le serveur envoie une trame d'information à un des clients et il attend un accusé de réception pour envoyer la trame suivante. Ce système de communication du type Client-Serveur-Client est représenté dans la figure 6.6.

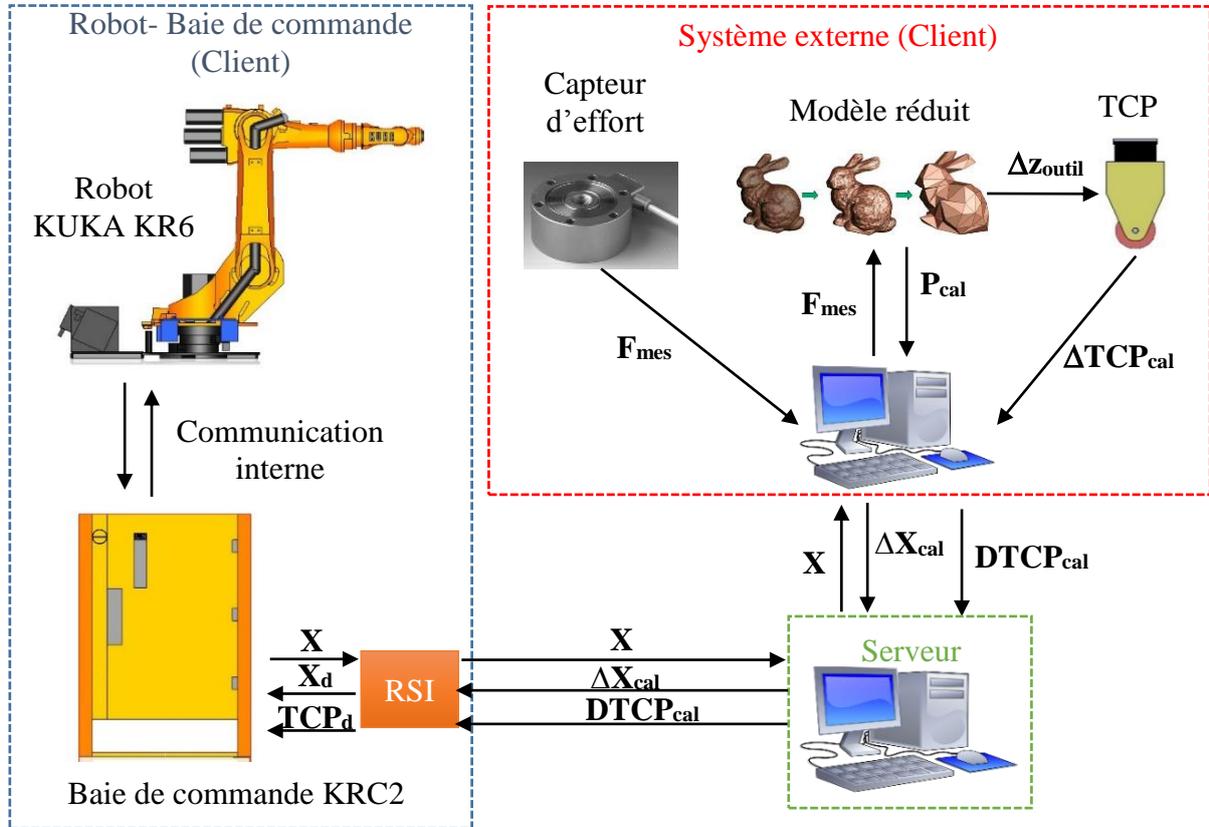


Figure 6.6 Système de communication

6.4.4 Moule

Les moules employés dans cette recherche sont le moule plan, le moule convexe et le moule triangle. Ils présentent les caractéristiques suivantes :

- Le moule plan comprend une surface plane avec des variations de dimensions dans la largeur (figure 6.7). Ses dimensions sont : une longueur de 0,44 m (axe X), une largeur de 0,15 m (axe Y) et une hauteur de 0,05 m (axe Z). Cependant, ce moule est divisé en quatre zones égales selon la longueur. La première (surface un) et la dernière (surface quatre) possèdent la même largeur 0,15 m, la deuxième (surface deux) dispose d'une variation dans sa largeur comprise entre 0,18 m et 0,047 m et la troisième (surface trois) garde une largeur de 0,047 m.

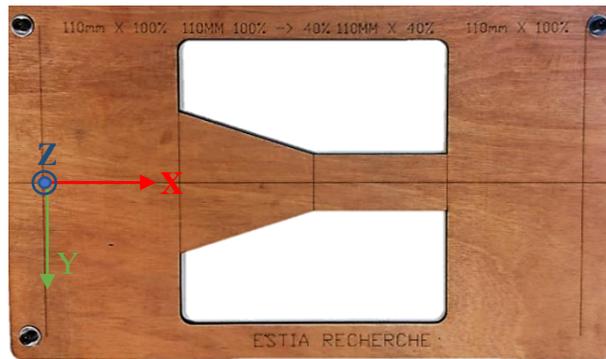


Figure 6.7 Moule plan avec des variations de dimension dans la largeur

- Le moule convexe (figure 6.8). Ses dimensions sont : une longueur (axe X) de 0,3 m, une largeur (axe Y) de 0,3 m, une hauteur (c) de 0,027 m et rayon (d) de 0,05 m (axe Z).

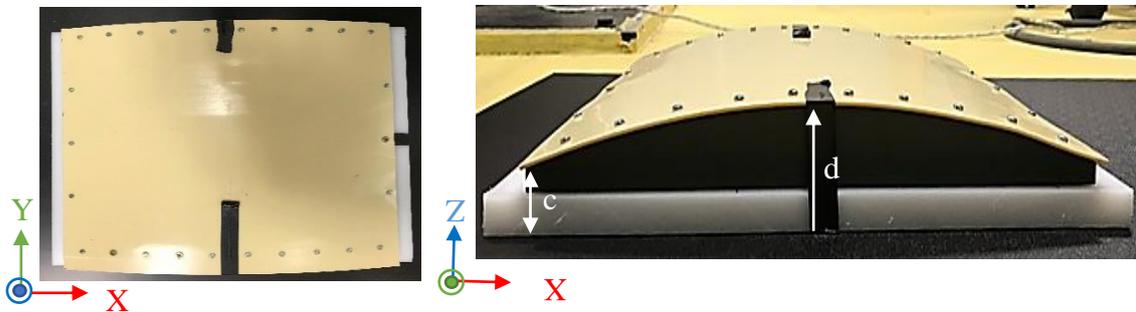


Figure 6.8 Moule convexe

- Le moule triangle comprend une surface en triangle rectangle isocèle (figure 6.9). Ces dimensions sont : une longueur (axe X) de 0,1414 m (la distance entre le point initial et la hauteur maximale (b) est de 0,0707 m), largeur (axe Y) de 0,52 m, une hauteur maximale (axe Z) de 0,0707 m et un rayon (r) de 0,005 m.

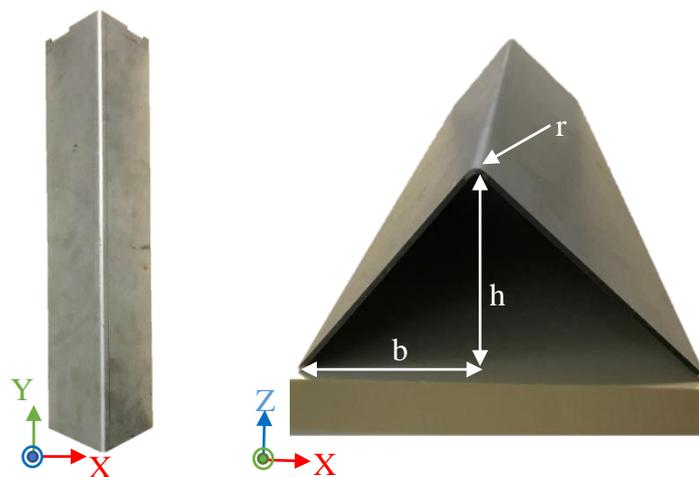


Figure 6.9 Moule triangle

6.4.5 Procédure de tests

La procédure pour la réalisation de l'expérimentation est similaire pour les trois moules. La trajectoire est décrite dans le plan XZ. Afin d'approcher le robot du moule, le robot est positionné au point A. Ensuite il descend au point B et est déplacé au point C. Entre les points B et C, le robot est en contact avec la surface. La trajectoire suivie par le robot pour le moule du type plan est exposée à la figure 6.10, pour le moule du type convexe à la figure 6.11 et pour le moule du type coin à la figure 6.12.

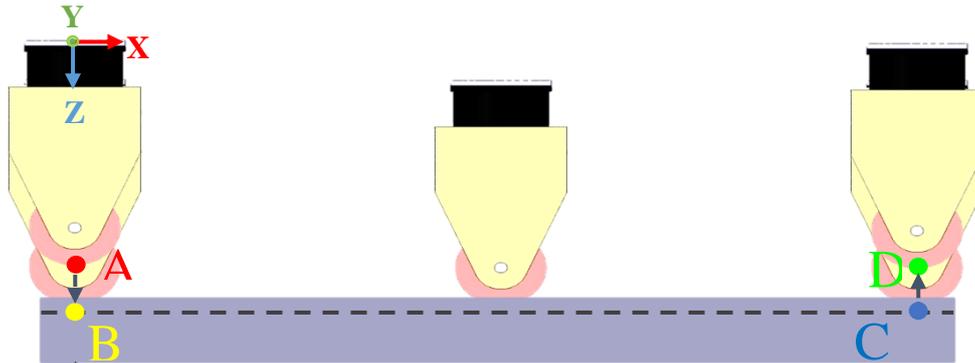


Figure 6.10 Procédure de test pour le moule du type plan

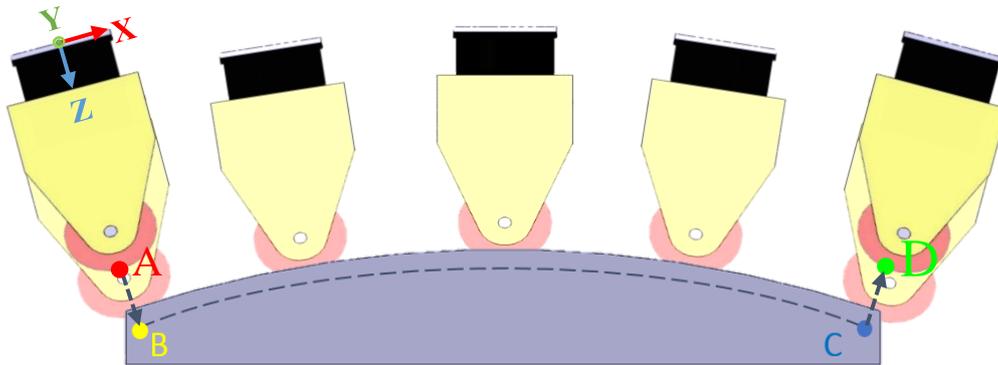


Figure 6.11 Procédure de test pour le moule du type convexe

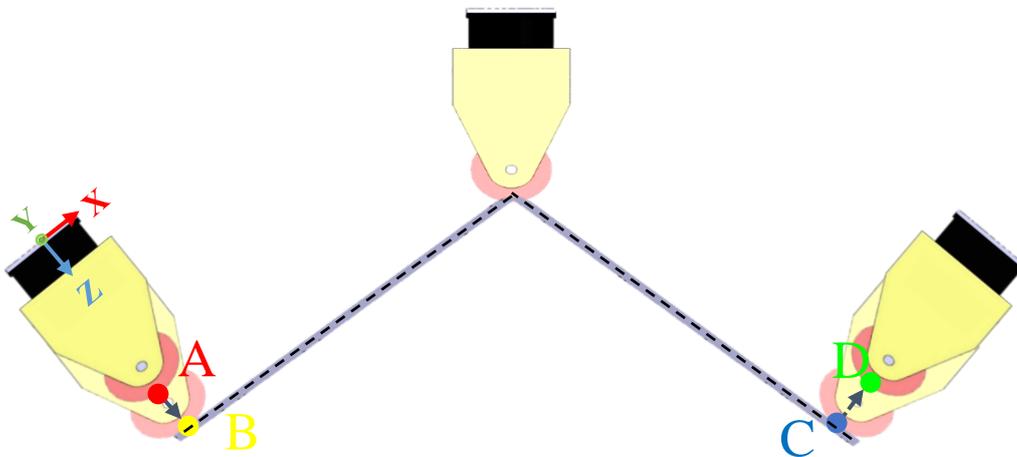


Figure 6.12 Procédure de test pour le moule du type triangle

6.5 Génération de trajectoire en pression avec prise en compte des flexibilités de l'outil

La solution proposée dans le paragraphe 6.1 considère que la force de compactage est modifiée lorsque la surface de contact change. Cette proposition reprend les principes basiques de la physique, tel que la pression est la relation d'une force appliquée par unité de surface et elle est représentée par (6.1). Cependant les flexibilités de l'outil provoquent un effet négatif sur la géométrie, lorsqu'une force est appliquée (paragraphe 6.2). Donc la génération de trajectoires en pression doit prendre en compte la force appliquée, la surface de contact et les flexibilités de l'outil, afin d'obtenir un compactage homogène (paragraphe 6.3). Ces éléments sont calculés dans les paragraphes 3.3.4 et 4.2 dans le cadre de cette recherche.

La génération de trajectoire qui considère ces éléments ne peut pas être réalisée en langage KRL, donc la trajectoire est générée à partir d'une boucle de commande externe via le module RSI (paragraphe 6.4.3) qui est l'intermédiaire entre la boucle de position interne et la boucle de commande externe. La dualité de commande force/position impose qu'une direction soit commandée par un seul type de commande. Afin d'éviter cette dualité, une matrice de sélection est intégrée dont la matrice en position est représentée par S_p et la matrice de sélection en force-pression est représentée par S_{fp} . Lorsque le système est commandé en force-pression, la pression de compactage est asservie selon la direction Z, les matrices de sélection sont représentées par :

$$S_p = (1; 1; 0; 1; 1; 1)$$

$$S_{fp} = (0; 0; 1; 0; 0; 0)$$

6.5.1 Commande hybride en position/force-pression

La commande hybride externe en position/force-pression utilise le capteur d'effort ATI FT-Gamma (défini dans le paragraphe 6.4.1) pour mesurer la force de compactage en temps réel et la surface de contact est définie hors-ligne par le modèle CAO (Conception Assistée par Ordinateur) du moule. De ce fait, le capteur d'effort est figé entre le poignet du robot et l'outil, les forces mesurées sont différentes de celles au contact, en conséquence, le transport de force défini par (5.3), doit être implémenté avant tout calcul.

La commande hybride externe en position/force-pression consiste à ajouter la commande en pression à la commande hybride externe en position/force, comme il est décrit dans le paragraphe 6.1.

Le moule plan (décrit dans le paragraphe 6.4.4) utilisé dans ce test, est divisé en quatre zones. Afin de rendre compatible l'information de ces zones avec les entrées du RN, les zones sont définies en termes de pourcentage. En conséquence, la première (surface un) zone et la dernière

(surface quatre) zone sont en contact total, ce qui représente que le rouleau de compactage est à 100% en contact avec la surface (contact A de la figure 6.13). La deuxième zone (surface deux) possède une variation dans sa largeur au long de l'axe Y de 0,18 m à 0,047 m, ce qui représente que le rouleau de compactage varie son contact avec la surface de 100% à 40%. La troisième zone (surface trois) reste constante à 0,047 m, ce qui indique que le rouleau de compactage est à 40% en contact avec la surface (contact B de la figure 6.13). La procédure des tests est définie dans le paragraphe 6.4.5.

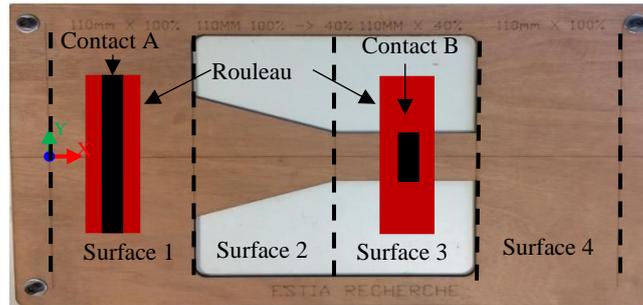


Figure 6.13 Zones du moule du type plan

Les résultats obtenus pour la commande hybride externe en position/force-pression sont comparés à ceux obtenus par la commande en position et à ceux obtenus par la commande hybride externe en position/force, pour vérifier la performance de la commande proposée. La commande en position est la commande intégrée par défaut dans le robot. Pour cette commande, le robot est positionné sur le point initial du contact (point B de la figure 6.7) avec une force initiale de 20 N et le robot se déplace jusqu'au point final de contact (point C de la figure 6.7). Toutes les forces sont mesurées pendant son déplacement. La commande hybride en position/force, le déplacement du robot en contact (B→C, figure 6.7) est commandé avec une force de consigne à 20 N. Finalement, la commande hybride en position/force-pression est commandée avec une pression de 0,078 MPa pour le déplacement du robot en contact (B→C, figure 6.7). Cette pression correspond à une force de 19,97 N lorsque le rouleau est en contact à 100%.

Les résultats de cette comparaison sont montrés figure 6.14. Ils permettent de vérifier l'homogénéité du compactage des fibres sur toute la surface pendant la dépose. La pression est calculée pour les trois commandes, grâce aux forces mesurées par le capteur d'effort. La figure 6.14 montre aussi les forces X et Y et les couples X, Y et Z mesurés. Les forces X et Y représentent les glissements du robot pendant la translation et les couples X, Y et Z représentent les rotations du robot.

La commande en position ne considère ni la force ni la surface, donc le robot est déplacé selon les mouvements appris. La figure 6.14a) montre les résultats obtenus pour la pression calculée.

Celle-ci est variable pendant le déplacement du robot en contact (point B→C, figure 6.7) et prend des valeurs entre 0,05 MPa et 0,150 MPa. La figure 6.14b) montre la force de compactage (F_z) qui est aussi variable pendant le déplacement du robot en contact. Ces valeurs sont comprises entre 5 N et 37 N. La force non désirée dans l'axe Y (figure 6.14d) et le couple non désiré dans l'axe Z (figure 6.14g) sont proche de zéro. Par contre, la force non désiré dans l'axe X est d'environ 0,865 N (figure 6.14c), le couple non désiré dans l'axe Y est de 0,205 N-m (figure 6.14f) et le couple non désiré dans l'axe X est variable pendant tout le déplacement, ces valeurs sont comprises entre - 0,0776 N-m et - 3,185 N-m (figure 6.14e). Par conséquent, le robot a un glissement non désiré dans l'axe X et des rotations non désirées dans les axes X et Y pendant son déplacement en contact.

La commande hybride en position/force montre une bonne performance sur les surfaces sans changement de dimension (surfaces un et quatre), car la force de compactage reste constante ($F_z=20$ N), ainsi que la pression calculée ($P_{cal}\approx 0,078$ MPa). Pourtant la performance est très mauvaise sur les surfaces avec changement de dimension (surface deux et trois), puisque la force de compactage reste constante sans tenir en compte de la modification de la surface (figure 6.14b). La pression calculée est perturbée par la force et par la modification de la surface. Considérant que la force reste constante, la pression augmente lorsque la surface se réduit et vice-versa. La pression calculée est comprise entre 0,078 MPa et 0,14 MPa (figure 6.14a). La force non désirée dans l'axe X est variable autour de 0,84 N (figure 6.14c), la force non désirée dans l'axe Y est variable avec des valeurs comprises entre 0,99 N et 5,5 N (figure 6.14d), les couples non désirés dans l'axe Y (figure 6.14f) et l'axe Z (figure 6.14g) sont variables avec des valeurs autour de 0,32 N-m et de 0,19 N-m respectivement, le couple non désiré autour de l'axe X est variable avec des valeurs comprises entre - 2,012 N-m et 0,527 N-m (figure 6.14e). En conséquence, le robot a des glissements non désirés dans les axes X et Y et des rotations non désirées autour des axes X et Y pendant son déplacement en contact.

La commande hybride en position/force-pression consiste à utiliser une pression constante pour toute la surface. La force de compactage est une variable d'état interne basée sur la pression souhaitée et la forme de la surface. Cette commande prend en compte la surface et la pression au même temps. La commande hybride en position/force-pression présente une bonne performance pour toutes les surfaces. Pour les surfaces un et quatre, la force de compactage (F_z) est proche de 20 N (figure 6.14a) et la pression calculée est proche de 0,078 MPa (figure 6.14b). Pour les surfaces deux et trois, la pression calculée reste constante ($P_{cal}\approx 0,078$ MPa) (figure 6.14a), tandis que la force de compactage (F_z) modifie sa valeur de 20 N à 9 N pour la surface

deux et reste constante à 9 N pour la surface trois (figure 6.14a). La force non désirée dans l'axe X est variable avec une valeur maximale de 0,8 N (figure 6.14c), la force non désirée dans l'axe Y est variable suivant la surface avec des valeurs comprises entre 1,67 N et 5,42 N (figure 6.14d), les couples non désirés dans l'axe Y (figure 6.14f) et l'axe Z (figure 6.14g) sont variables avec des valeurs maximales de 0,24 N-m et de 0,08 N-m respectivement. Le couple autour de l'axe X est variable avec des valeurs comprises entre 0,5 N-m et 1,2 N-m (figure 6.14e). Donc le robot a des glissements non désirés le long des axes X et Y et des rotations non désirées autour des axes X et Y pendant son déplacement.

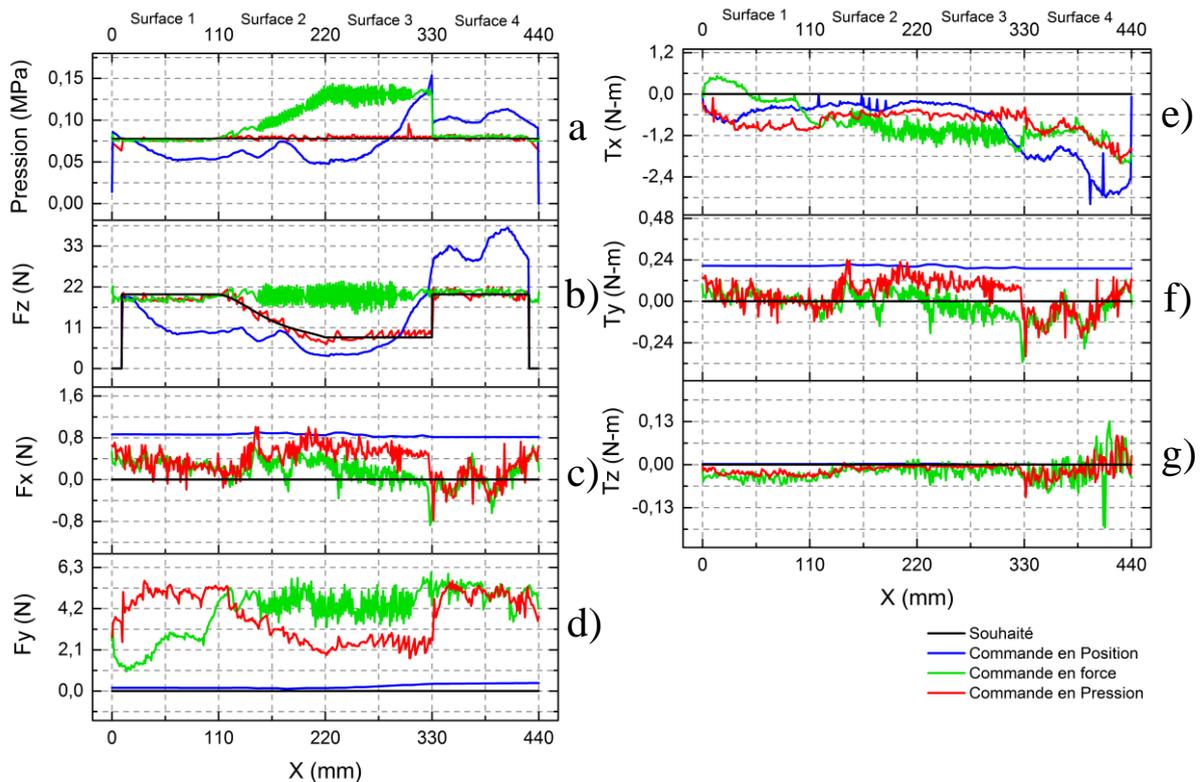


Figure 6.14 Résultats pour la commande en position, la commande en position/force et la commande en position/force-pression

Les forces et les couples non désirés dans la commande en position sont moindres pour certains axes comparés aux forces et couples non désirés des commandes en force et en pression. En dépit de ces résultats, la commande en position ne peut pas garantir un compactage homogène. Dans le cas des commandes en force et en pression, les forces et les couples non désirés sont très similaires, mais pour la commande en pression sont inférieurs que ceux de la commande en force. Par conséquent, la commande en pression réduit les glissements et les rotations non désirés. La commande en force permet de garantir un compactage homogène sur des pièces planes, dans des pièces complexes comme le montre la figure 6.13, mais l'homogénéité de la

pièce n'est pas respectée. Tandis que pour la commande en position/force-pression un compactage uniforme sur les pièces planes et complexes est assuré.

6.5.2 Point dynamique de centrage de l'outil

La plupart des applications industrielles robotisées considèrent l'outil comme rigide (Lartigue et al., 2003), de ce fait les tâches dont l'outil et la pièce sont en contact, le TCP et le point de contact sont généralement mis à la même position (figure 6.15 a).

Dans cette recherche, le TCP avec une déformation nulle dans l'outil à partir du poignet du robot est déterminé par (6.2).

$$T_{TCP} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0,2301 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.2)$$

Néanmoins dans notre recherche, l'outil est considéré comme un élément flexible. La non prise en compte des déformations, dans la définition du TCP génère des problèmes importants, principalement pendant la rotation de l'outil. Pour illustrer ce problème, la figure 6.15b) montre l'outil utilisé pour la dépose de fibres tourné à 45 degrés autour de l'axe Y : dans ce cas le TCP reste constant, tandis que le rouleau de compactage se déforme et le point de contact change de position.

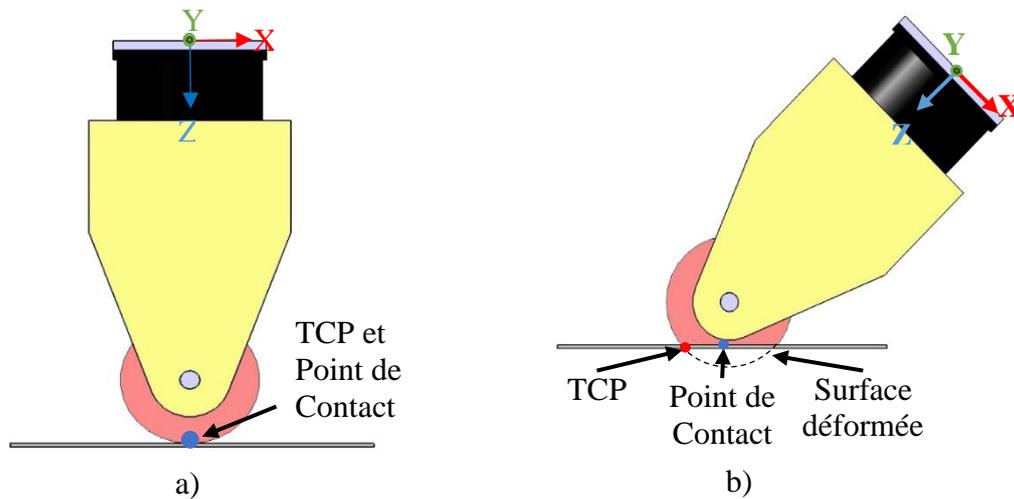


Figure 6.15 a) TCP fixe b) Rotation de l'outil autour de l'axe Y

La déformation du rouleau augmente selon les degrés de rotation autour de l'axe Y et le point/surface de contact de l'outil est modifié à cause de cette déformation. En conséquence les forces et couples mesurés par le capteur d'effort sont modifiés de manière importante. Afin de vérifier l'impact d'un changement de géométrie de l'outil sur le comportement du robot, des expérimentations sont réalisées. Ces tests consistent en une rotation de l'outil autour de l'axe Y avec des valeurs comprises entre zéro et cinq degrés, car l'outil doit respecter la restriction de

la tâche (paragraphe 2.3.3) d'être normale à la surface (90 degrés). Ces valeurs de rotation sont prises car si la rotation dépasse cinq degrés le capteur d'effort dépasse ses capacités de mesure et selon le matériau composite et la largeur des fibres la rotation admissible est limitée à un degré (Debout et al., 2011), l'outil est normal à la surface.

Dans le cas idéal, les forces et couples sont nuls et la modification du TCP est négligeable, par contre le contact entre l'outil et la pièce entraîne des forces et des couples qui produisent des glissements et des rotations de l'outil. Ces forces et couples modifient la position et l'orientation de l'outil. Dans un premier temps, l'outil est déformé de 0,5 mm en suivant l'axe Z et le TCP étant fixe et défini par (6.2) (cas industriel) (figure 6.16 a), ce qui conduit à l'apparition des efforts principalement dans l'axe Z. Pour le deuxième cas, l'outil est déformé de 0,5 mm en suivant l'axe Z (figure 6.16 b) et le TCP est calculé dynamiquement (proposition). Alors, le TCP étant défini par (6.2), il prend en compte une distance selon l'axe Z depuis le poignet du robot de 0,2296 m. Les résultats sont montrés figure 6.17. Les forces selon l'axe X (figure 6.17a), Y (figure 6.17b) et Z (figure 6.17c) augmentent avec la rotation. La force selon l'axe X est plus faible pour le première cas (TCP à 230,1 mm avec l'outil déformé de 5 mm le long de l'axe Z), selon l'axe Y, elle est minimale pour le deuxième cas (TCP à 229,6 mm) et pour les deux cas, la force selon l'axe Z est similaire, à cause du contact exercé avec la surface. Par ailleurs, les couples selon l'axe X (figure 6.17d), Y (figure 6.17e) et Z (figure 6.17f) sont proches de zéro, étant donné que la rotation est autour de l'axe Y et que le déplacement est nul. Par contre, les valeurs mesurées (TCP à 229.6 mm) sont considérablement plus faibles pour le deuxième cas, en conséquence le dernier cas est le plus proche de l'idéal.

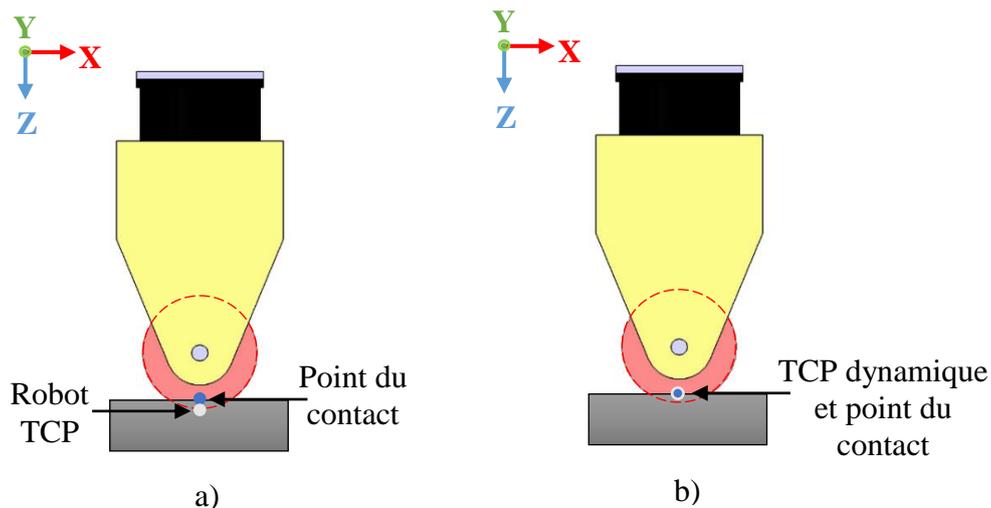


Figure 6.16 TCP a) outil déformé de 0,5 mm avec un TCP fixe (cas industriel) b) outil déformé avec un TCP dynamique.

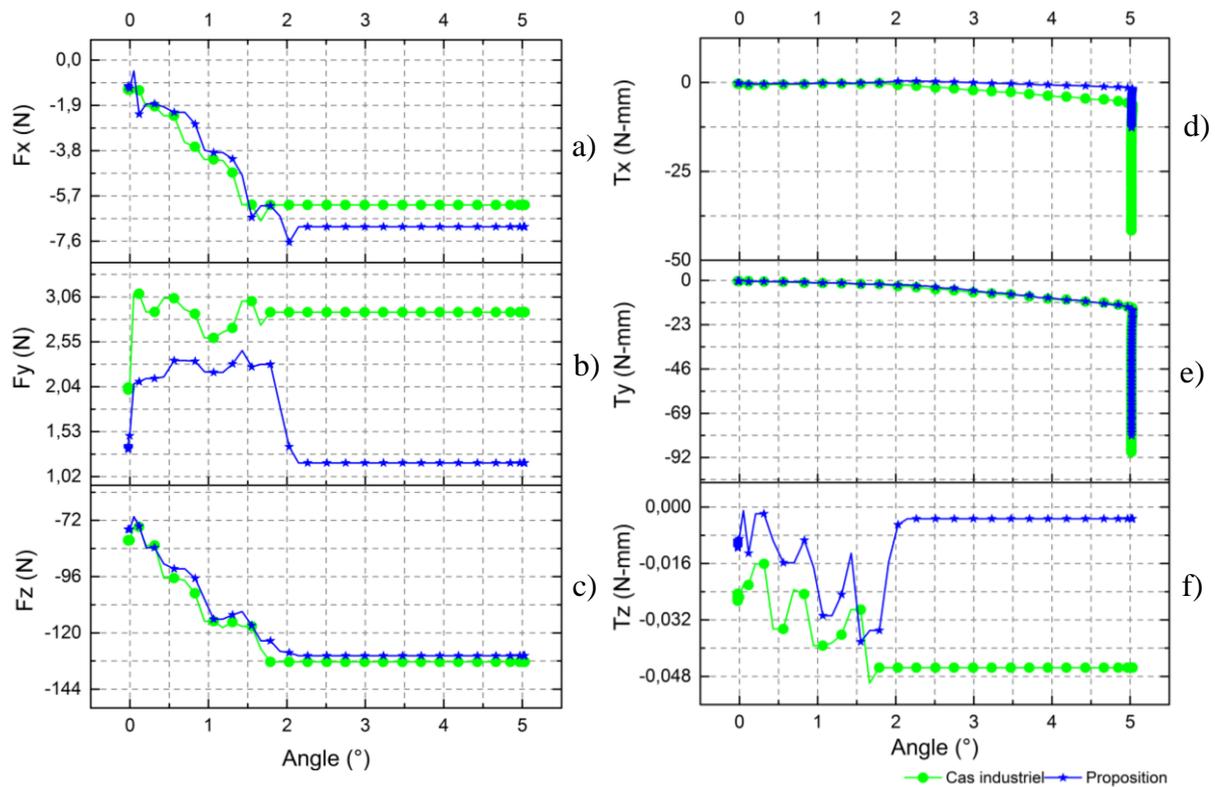


Figure 6.17 Résultats de la rotation de l'outil de cinq degrés autour de l'axe Y

Ces tests permettent de vérifier le comportement des forces et couples, lorsque l'outil est soumis à une rotation et/ou une déformation pour un TCP fixe et pour un TCP prenant en compte la déformation de l'outil. Les forces et couples sont généralement diminués en considérant les déformations de l'outil et en modifiant le TCP.

Afin de diminuer les efforts non désirés qui entraînent des déviations dans la trajectoire du robot pendant le drapage de fibres, nous réalisons les expérimentations dans les moules convexe et triangle (décrits dans le paragraphe 6.4.4) en considérant un TCP fixe et un TCP modifié en temps réel (TCP dynamique décrit dans le paragraphe 6.2). Ces moules présentent d'importantes courbures, donc le comportement de l'outil durant les mouvements de rotation reste importante à analyser.

La déformation de l'outil dans l'axe Z modifie le TCP. Cette déformation est calculée par le modèle réduit (paragraphe 4.2) et elle est fonction de la force appliquée et de la surface en contact. Ainsi, le capteur utilisé est un ATI-gamma (paragraphe 6.4.1) dans la plage de mesure maximale selon l'axe Z est de 100 N. En conséquence, la force mesurée selon cet axe doit être inférieure à 100 N pour nos tests. La commande en position/force-pression décrite dans le paragraphe 6.3 est utilisée pour les expérimentations. La surface de contact (H_r) est de 100%, car le rouleau de compactage est en contact total avec le moule et la force de consigne (F_{des}) est de 40 N selon l'axe Z.

Pour la première expérimentation, nous considérons le moule convexe. La procédure de test décrite dans le paragraphe 6.4.4 consiste à approcher le robot du moule (point A figure 6.11), ensuite le robot est positionné au contact (point B figure 6.11), enfin il est déplacé jusqu'à la fin du moule (point C figure 6.11). La trajectoire doit être modifiée en suivant la courbure du moule, afin que l'outil reste normal à la surface. Pour définir la trajectoire automatiquement, l'information des dimensions du moule est extraite des données de la CAO. La trajectoire entre les points B à C est calculée avec le logiciel MATLAB. La trajectoire commence (point B figure 6.11) avec un angle de -12° et se termine (point C figure 6.11) avec un angle de 12° , la distance de translation entre ces points est de 0,210 m. Pendant le déplacement du robot du point B au point C, l'outil tourne pour rester en contact et respecter sa contrainte. Les résultats obtenus sont montrés à la figure 6.18. Les courbes sont presque identiques, mais les résultats sont légèrement meilleurs pour le DTCP que pour le TCP fixe. La distance de déplacement du robot pour chaque degré étant d'environ 0,0087 m, et la rotation étant minimale par rapport à la distance parcourue, la surface peut être considérée comme plane, en conséquence la déformation de l'outil est considérée comme nulle. La différence entre les résultats pour le DTCP et le TCP fixe n'est pas significative.

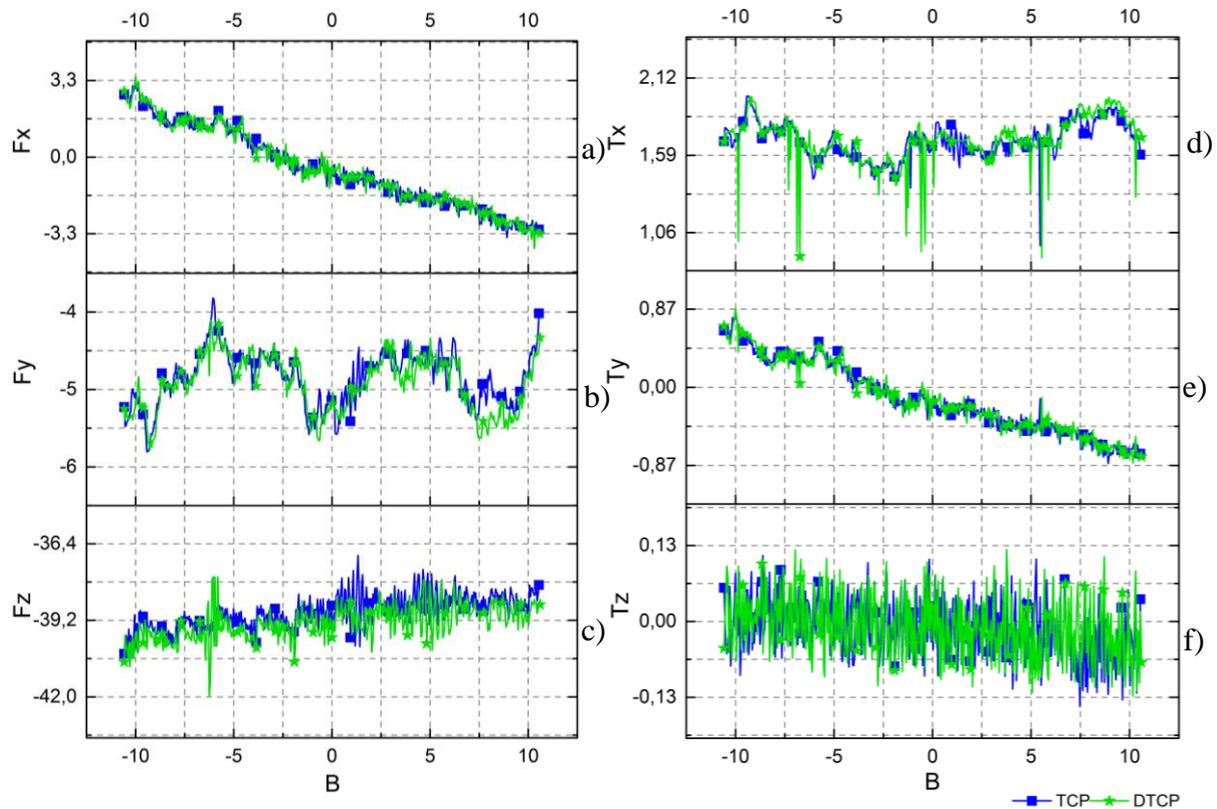


Figure 6.18 Résultats pour le TCP et le DTCP sur un moule du type convexe

Pour la deuxième expérimentation, nous considérons le moule triangle. La procédure de test est similaire à celle employée pour le moule de type convexe. Le moule triangle possède un angle

de 90° et l'outil doit respecter sa contrainte (normale à la surface), donc la trajectoire dépend aussi de l'angle. Comme pour le moule convexe, la trajectoire est générée automatiquement à partir de l'information des dimensions du moule extraite de la CAO. Les points de la trajectoire pour le déplacement au contact (B \rightarrow C) sont calculés par le logiciel MATLAB. La trajectoire commence (point B) avec un angle de -45° et se termine (point C) avec un angle de 45° , la distance parcourue entre la rotation est de 0,005 m. Les résultats obtenus sont montrés à la figure 6.19. Les trajectoires pour le TCP et DTCP doivent suivre une force de consigne de 40 N selon l'axe Z (figure 6.19c). La force de consigne reste constante à 40 N pendant que le robot suit la trajectoire dans l'axe X. Cependant la force varie d'environ 3 N, lorsque l'outil tourne autour de l'axe Y. Ces erreurs sont dues à la vitesse de mesure du capteur d'effort et au temps de stabilisation de la loi de commande en logique floue PI (paragraphe 5.5.2). Le comportement du rouleau montre des résultats similaires pour la force mesurée selon l'axe X (figure 6.19a), le couple Y (figure 6.19e) et le couple Z (figure 6.19f) avec une légère amélioration dans la courbe du DTCP. Le glissement selon l'axe Y (figure 6.19b)) et la rotation non désirée selon l'axe X (figure 6.19d) du robot pendant la rotation de l'outil selon l'axe Y sont réduits grâce à l'utilisation du DTCP.

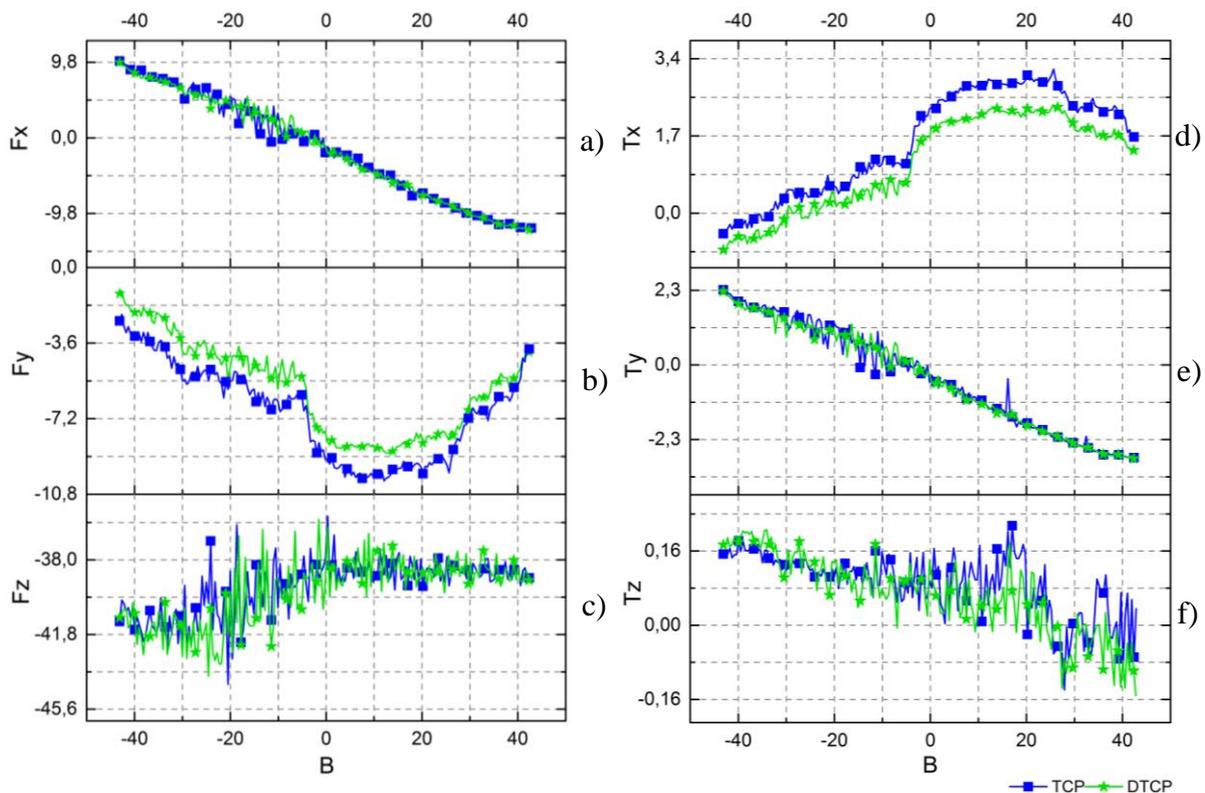


Figure 6.19 Résultats pour le TCP et le DTCP sur un moule du type coin

La force appliquée dans les deux expérimentations représente une déformation d'environ de 0,1 mm, donc les écarts entre le TCP (cas industriel) et TCP dynamique (proposition) seront plus

significatifs en appliquant des forces plus importantes. Dans le cadre de ce travail, les écarts seront significatifs lorsqu'on dépose des fibres thermoplastiques ou thermodurcissables, dont la force appliquée est d'entre 200 N et 1200 N. La déformation de l'outil pour ces forces est comprise entre 1 mm et cinq mm.

6.6 Conclusion

Ce chapitre nous permet de présenter les approches à implémenter pour l'amélioration de la tâche de dépose de fibres robotisée. Ces approches utilisent les commandes hybrides externes pour leur fonctionnement, dont l'implémentation dans des robots industriels est fiable, car la baie de commande n'est pas modifiée. Ces commandes permettent de corriger la trajectoire du robot à partir des capteurs extéroceptifs et des modules de calcul. Ainsi, la modification des variables internes du robot est réalisée en temps réel ce qui permet de prendre en compte les flexibilités de l'outil et de réduire les problèmes d'imprécision dus aux flexibilités.

La première approche permet uniquement de maintenir un compactage homogène des fibres dans toute la pièce. La force de compactage est modifiable en temps réel selon les dimensions de la surface de la pièce. La deuxième approche permet uniquement de modifier le TCP en suivant les flexibilités de l'outil, afin de positionner correctement l'outil et d'éviter des rotations ou glissements. La troisième approche permet de maintenir un compactage homogène des fibres et de positionner correctement l'outil en modifiant le TCP selon les flexibilités de l'outil. Cette dernière approche réalise, au même temps, la correction de la trajectoire de l'outil et la modification du TCP pour réaliser pièces avec des surfaces complexes et avec des grandes courbes.

Nous décrivons aussi l'expérimentation réalisée pour améliorer la précision de la tâche de dépose de fibres robotisée, à partir de la correction de la trajectoire du robot et la définition dynamique du point de centrage de l'outil (TCP).

Dans cette expérimentation, nous appliquons les commandes hybrides externes qui permettent de corriger la trajectoire du robot à partir des capteurs extéroceptifs. Le procédé de dépose de fibre exige un compactage des fibres homogènes pour garantir la qualité de la pièce. La loi de commande en pression est employée pour la correction de la trajectoire du robot. Cette commande garantit un compactage homogène sur les pièces planes et les surfaces complexes et réduit les forces et les couples non désirés. Ces résultats sont montrés dans la figure 6.14.

Les flexibilités de l'outil créent des différences entre le TCP et le point de contact sur des pièces avec une importante rotation de l'outil comme des pièces avec des coins et courbes complexes. Nous implémentons une commande hybride en position-force/pression en comptant les

flexibilités de l'outil par la définition dynamique du TCP. Cette commande réduit les forces et des couples non désirés qui génèrent des imprécisions pendant le déplacement du robot. Les résultats sont montrés dans la figure 6.19.

L'accroissement de la force provoque des modifications plus importantes sur la définition du TCP, car la relation entre la force et les flexibilités est non linéaire.

CHAPITRE

7

Conclusion et perspectives

Sommaire

7.1 Conclusions	124
7.2 Perspectives	125

7.1 Conclusions

Conformément aux objectifs poursuivis dans cette recherche, nous concluons que l'objectif général d'améliorer la précision actuelle des robots dans la tâche de dépose de fibres sans arrêter les processus de fabrication peut être atteint par une commande référencée en position/force-pression qui prend en compte les flexibilités de l'outil. La commande référencée en position/force-pression et la définition dynamique du TCP, sont proposées dans le chapitre 6 et évaluées séparément dans le paragraphe 6.5, puis elles sont intégrées conjointement.

Les contraintes de l'application industrielle ont été respectées pour l'implémentation de notre système, car la baie de commande n'est pas modifiée dans cette proposition. Le système industriel étudié ne possède pas de retour d'information sur l'interaction entre le robot et l'environnement, ce qui implique que le compactage homogène de fibres n'est pas garanti comme nous l'avons constaté dans le paragraphe 6.5. Afin d'éviter le décollage des fibres entre les plis, le procédé de dépose de fibres robotisé est commandé en boucle ouverte par un vérin pneumatique. Par contre la rigidité de l'interaction entre le rouleau de compactage et le moule de dépose entraîne des efforts parasites pendant la dépose. Ces efforts créent des glissements et rotations qui rendent impossible un drapage directionnel idéal. Le travail précédent proposé par (Uhart, 2014) pour améliorer la précision de cette tâche considère l'utilisation d'une commande avec une génération de consigne à partir de l'erreur en position, en force et en vision. Ce travail

améliore la régulation de l'effort pour les fibres, la distribution de l'effort sur le rouleau de compactage, la régulation de l'intervalle de bande et la diminution des forces non désirées. Lorsqu'il expose ces améliorations, l'auteur travaille uniquement avec des pièces planes et sans modification dans leurs dimensions, en conséquence le compactage homogène est uniquement garanti dans ce type de pièces, comme nous l'avons observé dans le paragraphe 6.5. En comparaison avec le système industriel et le travail précédent, le nouveau procédé proposé dans cette recherche dispose de plusieurs améliorations :

- Une régulation de la pression de compactage en temps réel.
- Un calcul dynamique des correcteurs de la loi de commande qui permet d'adapter cette loi à divers types de pièces.
- Un compactage homogène des fibres sur divers types de pièces.
- Une réduction des glissements et des rotations de l'outil grâce à la définition dynamique du TCP pendant la dépose de fibres.

D'autre part, les objectifs établis du point de vue industriel sont atteints, grâce à l'étude de la tâche de dépose de fibres, à l'analyse et à la modélisation du comportement des composants de la cellule robotisée exposées dans les chapitres 2 et 3. Ainsi, les objectifs établis du point de vue la recherche sont obtenus par l'utilisation d'un modèle réduit étudié dans le chapitre 4, par la génération de trajectoire avec des capteurs externes examinés dans les chapitres 5 et 6 et par un loi de commande adaptable à toutes types de surfaces présentée dans le chapitre 5. Ces éléments permettent de calculer les flexibilités de l'outil en temps réel, de réaliser le calibrage dynamique du TCP, d'évaluer les modifications de la géométrie de la pièce, et finalement, de corriger la trajectoire du robot en temps réel en considérant les efforts appliqués.

Les objectifs fixés sont alors bien accomplis en respectant les spécifications industrielles de la tâche de dépose de fibres.

7.2 Perspectives

A l'issue de ce travail de thèse, plusieurs problèmes demeurent ouverts. Nous présentons ici, ce qui nous semble être les perspectives envisageables pour améliorer les résultats de cette recherche.

- Transfert de technologie vers l'industrie : les résultats obtenus dans cette recherche ont été évalués à une échelle inférieure au niveau des efforts d'interaction et de la taille des composants de la cellule robotisée. Afin de réaliser ce transfert, la solution proposée doit être implémentée sur le robot industriel KR240 avec un capteur d'effort capable de

mesurer un effort de compactage au moins de 1500 N pendant la dépose de fibres. Ces changements peuvent influencer sur les améliorations attendues montrées dans cette thèse.

- Vitesse de transmission d'information : la commande implémentée dans cette recherche travaille avec une baie de commande du type KRC2 qui transmet les données par le module RSI avec une période d'échantillonnage de 12 ms. Il serait intéressant de travailler avec une baie de commande plus moderne comme la KRC4 qui communique l'information avec une période d'échantillonnage de 4 ms. En conséquence la commande pourrait être exécutée trois fois plus vite.
- Vitesse de dépose de fibres : les résultats obtenus dans le cadre de cette recherche considèrent une vitesse de drapage entre 10mm/sec et 40 mm/sec. Afin de réduire le temps de drapage, la vitesse de drapage doit augmenter, en conséquence, la commande et le capteur d'effort devraient transmettre l'information plus rapidement.
- Détection de l'environnement : il serait intéressant de pouvoir détecter en temps réel les conditions du travail du robot pour éviter des collisions, les configurations singulières et/ou des trajectoires impossibles à réaliser.
- Degrés de liberté externes : la modélisation de cette recherche est effectuée pour le robot KR6 en excluant le rail fixe du robot industriel. Ce système présente des défauts géométriques importants, amplifiés par la longueur de ces axes linéaires qui peut atteindre plusieurs dizaines de mètres. Certaines pièces utilisent un axe externe rotatif pour fabriquer des pièces complexes avec des courbes importantes. Cet axe présente aussi des variables comme la vitesse de rotation et des défauts géométriques importants comme les flexibilités de l'axe. Les défauts géométriques et les variables de ces systèmes externes peuvent entraîner des défauts de précision. De ce fait, le modèle (géométrique, cinématique et dynamique) de ces systèmes externes en intégrant les défauts géométriques et les variables devraient être comptés au modèle du robot.
- Intégration des flexibilités : dans le cadre de cette recherche, nous avons étudié les erreurs provoquées par les flexibilités des divers composants, mais nous avons seulement choisi de travailler avec la flexibilité de l'outil qui est la plus importante en comparaison des autres flexibilités du robot. Cependant il serait important d'intégrer les flexibilités des corps et des articulations du robot, ainsi que les flexibilités dans les fibres et le moule, afin de réduire au maximum les problèmes de précision causés par leurs interactions.

- Configuration du robot : il serait intéressant d'étudier les configurations du robot dans lesquelles il est susceptible d'être plus flexible.
- Modèle réduit : l'utilisation d'une méthode de réduction a priori comme le PGD (Proper Generalized Decomposition) permettrait de construire un modèle réduit du comportement du système physique pour explorer très rapidement, et de façon cohérente, son comportement selon le principe d'abaques numériques multiparamétriques. Cette méthode présente l'avantage de ne pas imposer de simplification du modèle de comportement du matériau ce qui est indispensable pour traiter le cas des matériaux composites.

Finalement, nous recherchons à implémenter dans les entreprises des solutions robotisées innovantes, aux coûts réduits et simples à mettre en place pour améliorer leur compétitivité.

Annexes

A. Annexes

Sommaire

A.1 Modèle du robot	128
A.2 Modèle analytique par Hertz	129
A.2.1 Cas cylindre-plan	129
A.2.2 EF et contact de Hertz.....	133
A.3 Capteur d'effort.	136

A.1 Modèle du robot

Le Tableau A.1 présente les paramètres de Denavit Hartenberg modifiés qui sont calculés par Uhart (Uhart, 2014), pour un robot de dépose de fibres.

Tableau A.1 Paramètres de D-H modifiés

j	α	d	θ	r	${}^{j-1}T_j$
1	0	0	0	q_1	0T_1
2	$-\pi/2$	0	$q_2 + \pi/2$	$-r_2$	1T_2
3	$-\pi/2$	$-d_3$	q_3	0	2T_3
4	0	$-d_4$	$q_4 - \pi/2$	0	3T_4
5	$-\pi/2$	d_5	$q_5 + \pi$	$-r_5$	4T_5
6	$-\pi/2$	0	$q_6 + \pi$	0	5T_6
7	$-\pi/2$	0	q_7	0	6T_7
8 – capteur	π	0	0	r_8	7T_8
9	0	0	$\pi/2$	0	8T_9
10	α_{10}	0	0	0	${}^9T_{10}$
11	0	0	$-\pi/2$	0	${}^{10}T_{11}$
12 – outil	0	0	0	r_{12}	${}^{11}T_{12}$

A.2 Modèle analytique par Hertz

La théorie du contact de Hertz (Puttock & Thwaite, 1969; Shigley & Mitchell, 1997; Norton & García, 1999; Harris, 2001; Utah, 2004) permet de décrire le comportement du rouleau au niveau de la surface du contact. Cette analyse établit que deux éléments en contact ont une déformation en rapport avec les forces appliquées. La déformation dépend des propriétés mécaniques du matériau, du type de surface en contact (concave, convexe ou plan (figure a.1)) et de la force exercée.

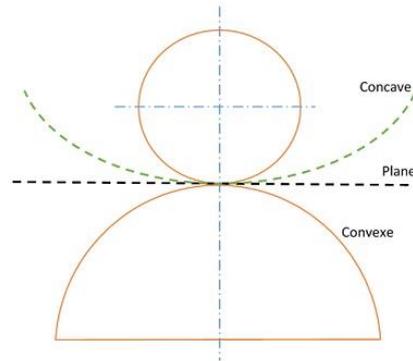


Figure A.1 Contact sans aucune force

Les équations de Hertz sont applicables si :

- la déformation du matériau se trouve dans la zone plastique (Williams & Dwyer-Joyce, 2000; Harris, 2001),
- la force est perpendiculaire à la surface,
- le rayon de la courbure des aires du contact est trop large par rapport aux dimensions des aires de contact (Tordesillas, 1992; Williams & Dwyer-Joyce, 2000; Harris, 2001),
- la dimension de l'aire de contact est plus petit que la taille des corps (Tordesillas, 1992; Williams & Dwyer-Joyce, 2000),
- les corps sont des matériaux isotropies, homogènes et linéaires (Tordesillas, 1992)
- le type de contact est sans friction (Tordesillas, 1992; Williams & Dwyer-Joyce, 2000).

A.2.1 Cas cylindre-plan

La figure a.2 montre la déformation de deux surfaces circulaires en contact, cette déformation est provoquée par une force de compression qui est perpendiculaire à la surface. La figure a.3 représente la déformation d'une surface circulaire en contact et une surface plane. Les figures sont maximisées pour amplifier la déformation. Le déplacement transverse représenté par δ (figure a.3) est :

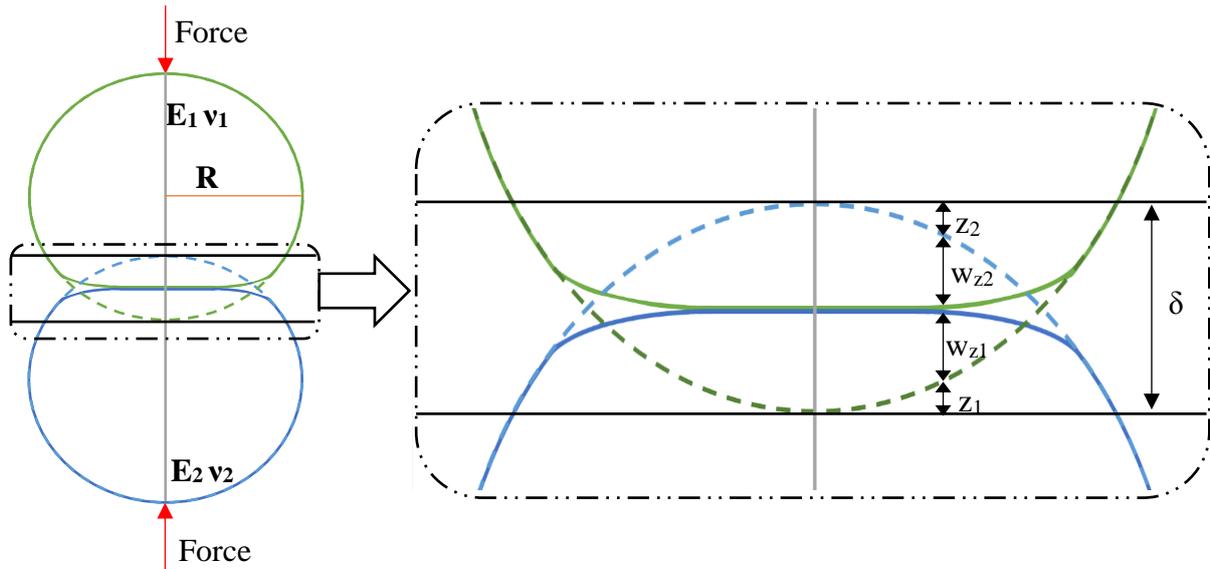


Figure A.2 Représentation de la déformation de deux sphères en contact avec une force normale

$$(z_1 + w_1) + (z_2 + w_2) = \delta \tag{A.1}$$

Telle que :

z_1 et z_2 soient les distances entre les surfaces sans déformation par rapport aux rayons,
 w_1 et w_2 soient les distances des déformations entre les surfaces par rapport aux surfaces sans déformation.

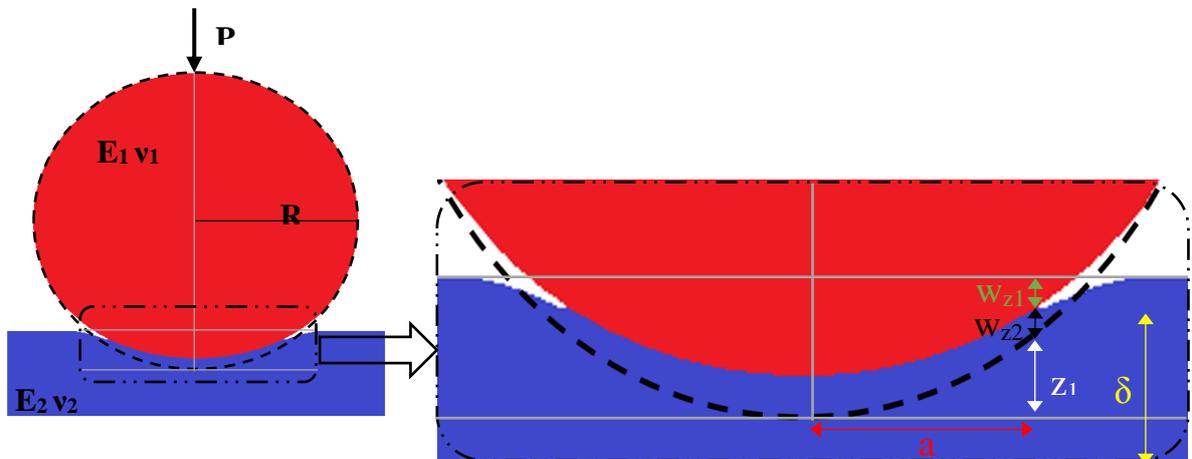


Figure A.3 Déformation sur un plan

Dans le cas d'un cylindre en contact avec une surface plane, la distance z_1 est infinie, car le rayon R_2 est égal à zéro. Le déplacement δ représenté par (A.1) est réécrit pour les surfaces planes par :

$$w_{z1} + w_{z2} = \delta - z_1 \tag{A.2}$$

Telle que :

h soit la distance donnée par $z_1 = \frac{r^2}{2R}$.

La théorie indique que le contact entre un cylindre et une surface plane crée une trace de contact avec une surface de largeur égale à zéro. En conséquence, toutes les forces appliquées sur le cylindre créeront un effort infini. Par contre, cette théorie n'est pas valide, car sinon tous les matériaux échoueraient instantanément (Norton & García, 1999). Normalement, les matériaux soumis aux efforts et en contact créent une trace de contact avec une surface supérieure à zéro, due à la déflexion des matériaux. Cette surface de contact créée par les efforts permet de supporter la charge appliquée. Les efforts génèrent dans la surface de contact une pression dont la distribution de pression est semi-ellipsoïdale. Dans le cas du cylindre, la trace de contact est semi-ellipsoïdale rectangulaire (figure a.4).

Les équations considérées dans cette section sont les équations exposées par Shigley, Norton et Utah (Shigley & Mitchell, 1997; Norton & García, 1999; Utah, 2004).

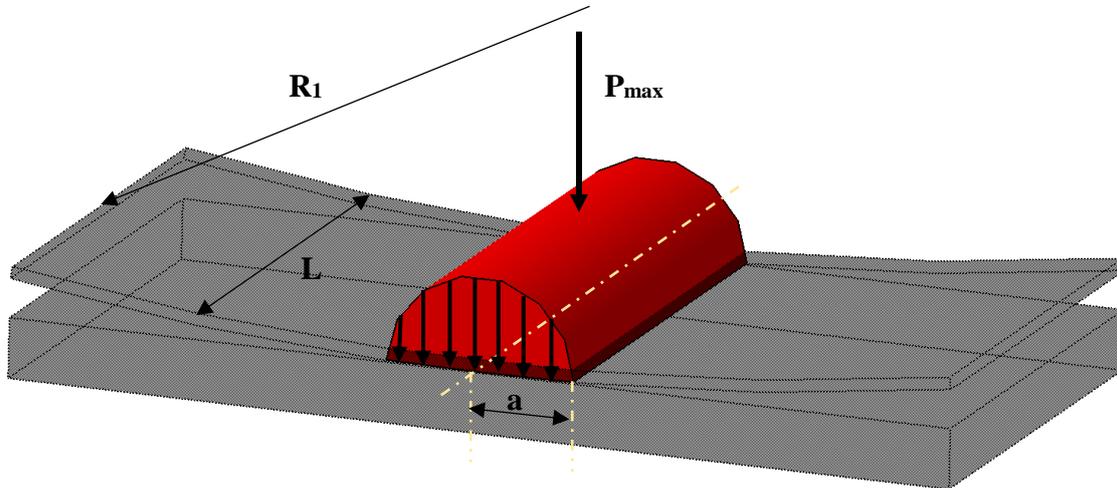


Figure A.4 Distribution des pressions

La trace de contact semi-ellipsoïdale rectangulaire du cylindre est décrite par (A.3).

$$P = P_{\max} \sqrt{1 - \frac{r^2}{a^2}} \quad r \leq a \quad (\text{A.3})$$

Tel que :

$$P_{\max} \text{ soit la pression maximale. } P_{\max} = \frac{2F}{\pi a L},$$

$$a \text{ soit la surface de la zone de contact. } a = \sqrt{\frac{2F(m_1 + m_2)}{\pi L B}},$$

$$m_1 \text{ et } m_2 \text{ soient des constantes du matériau. } m_{12} = \frac{1 - \nu_{12}^2}{\varepsilon_{12}},$$

ν_1 et ν_2 soient les coefficients de Poisson de chaque matériau,

ε_1 et ε_2 soient les Modules du Young,

$$B \text{ soit la constante géométrique des rayons. } B = \frac{1}{2} \left(\frac{1}{R_1} + \frac{1}{R_2} \right).$$

Dans le cas du cylindre en contact avec une surface plane, le rayon R_2 sera infini, donc $1/R_2=0$.

La pression de contact de la section transversale du cylindre provoque un état de contrainte (plan xz, figure a.5) caractérisé par σ_x , σ_y , σ_z , dont les valeurs maximales sont données par :

$$\begin{aligned}\sigma_{z_{\max}} &= -P_{\max} \\ \sigma_{x_{\max}} &= \sigma_{z_{\max}} \\ \sigma_{y_{\max}} &= 2\nu P_{\max}\end{aligned}\tag{A.4}$$

Les valeurs w_{z1} et w_{z2} sont calculées par :

$$w_z(x, y) = \frac{1 - \nu^2}{\pi E} + \iint_A \frac{p(x'y')}{r} dx' dy'\tag{A.5}$$

Tel que : $r^2 = (y - y')^2 + x'^2$

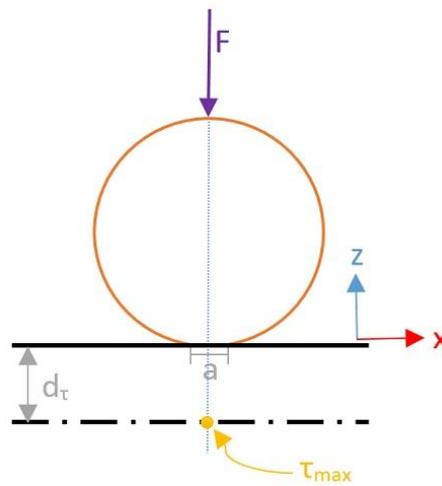


Figure A.5 Contrainte de cisaillement xz

Les valeurs calculées par (A.5) sont la représentation pour le cas général en contact. Par contre, pour le cas de contact cylindre-plan, le système est réduit au plan y. Il est représenté par :

$$\delta - h = \frac{1 - \nu^2}{\pi E} + \iint_A \frac{p(y')}{r} dx' dy'\tag{A.6}$$

La valeur de w_z est obtenue lors de la résolution de l'intégrale de (A.6). Ce qui permet d'obtenir la valeur de la déformation dans l'axe z. Dans leurs travaux (Puttock & Thwaite, 1969; Truten, 1971; Norden, 1973; Hamrock et al., 1983; Hamrock et al., 1984; Horng et al., 1999; Williams & Dwyer-Joyce, 2000; Harris, 2001; Utah, 2004; Crețu & Benchea, 2012; Dolo, 2017) reprennent la solution de l'équation de la déformation. Ces travaux représentent cette équation de manières diverses. Cette recherche considère l'équation de la déformation décrite par Hamrock (Hamrock et al., 1983; Hamrock et al., 1984). La déformation du cylindre est donc représentée par :

$$\delta_z = \frac{2F}{\pi E^* L} \left[\ln \left(\frac{4R}{a} \right) - 1 \right]\tag{A.7}$$

Tel que :

$$E^* \text{ module d'élasticité équivalent tel que } \frac{1}{E^*} = \frac{1-\nu_1^2}{E_1} + \frac{1-\nu_2^2}{E_2}$$

Certains auteurs (Shigley & Mitchell, 1997; Norton & García, 1999; Scarcella, 2008) ont démontré que la contrainte de cisaillement maximale se trouve à une profondeur (d_τ) sur l'axe z (figure a.5). Les contraintes de cisaillement sont normalisées par rapport à l'aire de la surface de contact (a), donc la profondeur (d_τ) est calculée par (A.8). La magnitude de la contrainte de cisaillement maximale est déterminée par :

$$d_\tau = 0,75a \quad d_\tau = 0,7865a \text{ (Norton)} \quad (\text{A.8})$$

$$\tau_{\max} = 0,30P_{\max} \quad \tau_{\max} = 0,304P_{\max} \text{ (Norton)} \quad (\text{A.9})$$

A.2.2 EF et contact de Hertz

Dans cette section, nous considérons uniquement le comportement élastique sous l'hypothèse de petites perturbations (HPP). Le but est de valider notre modèle éléments finis (EF) à partir des résultats du modèle analytique de type Hertz. Comme nous avons décrit dans la section A.2, les équations de Hertz sont applicables aux matériaux isotropes, homogènes et linéaires et dont leur déformation se trouve dans la zone élastique. Ces caractéristiques mécaniques sont particulières pour les matériaux comme l'acier.

Dans les travaux de Scarcella, Darji et Xia (Scarcella, 2008; Darji & Vakharia, 2012; Xia et al., 2012), les auteurs utilisent les équations de Hertz pour modéliser le contact entre deux solides (cylindre-plan) et la méthode EF pour valider leur modélisation.

Dans le travail de Scarcella (Scarcella, 2008), l'auteur emploie les équations de Hertz entre une sphère et un plan pour analyser le comportement entre un rouleau et un support. Pour l'analyse, l'auteur considère un rouleau de rayon de 0,01m, une force appliquée de 100 N (par mm d'épaisseur) et un matériau du type acier « AMS6490 Bearing Steel » pour le deux composants (le rouleau et le support). Ce matériau possède un module d'élasticité de 100 000 MPa et un coefficient de Poissons de 0,29. L'auteur réalise la validation par éléments finis avec le logiciel « ANSYS mechanical 11,0 ». L'analyse est réalisée en deux dimensions. Les résultats numériques obtenues par l'auteur pour la pression maximale avec la méthode par EF est de 410 MPa et avec la méthode théorique (les équations de Hertz) est de 417 MPa, l'erreur entre les deux méthodes est de 1,7 %. Un autre résultat obtenu par l'auteur est la surface de contact qui est de 0,000176 m pour la méthode EF et de 0,000153 m pour la méthode théorique, cette différence est négligée car l'espace entre nœuds est de 0,00003 m.

Dans le travail de Darji (Darji & Vakharia, 2012), l'auteur utilise les équations du contact de Hertz pour analyser un roulement mécanique avec cinq rouleaux internes. L'auteur utilise six rouleaux cylindriques de la série NU 22 composés d'un matériau du type acier AISI 52 100 avec un module d'élasticité de 201 330 MPa et un coefficient de Poissons 0,277. Les cinq rouleaux sont de même longueur soit 0,012 m et de différents diamètres soient de 0,00662 m, de 0,00858 m, de 0,01282 m, de 0,01714 m et de 0,02056 m. L'analyse est réalisée en trois dimensions puisque l'auteur analyse les rouleaux comme des éléments cylindriques. La force appliquée dépend de la position de chaque rouleau, car chaque rouleau est affecté par un angle. L'auteur effectue l'analyse théorique pour toutes les positions des rouleaux. La validation est réalisée par la méthode EF avec le logiciel ANSYS 9.0. Les résultats obtenus pour les deux méthodes (EF et théorique) varient de 5.8% pour la contrainte de Von Misses, entre 5% et 10% pour les autres résultats comme la pression maximale, la déformation, la surface de contact,.... La variation entre les résultats est due à la taille du maillage (la distance entre chaque nœud).

Dans le travail de Xia (Xia et al., 2012), l'auteur réalise l'analyse d'un roulement mécanique avec 22 rouleaux internes, l'objectif de sa recherche est de définir la relation entre la contrainte de tension et l'erreur conique du support par la méthode EF avec le logiciel ANSYS. Les rouleaux internes sont de type N1015. Les dimensions de chaque rouleau sont : diamètre de 0,010 m et longueur de 0,011 m. Les caractéristiques mécaniques du matériau sont : un module d'élasticité de 208 000 MPa et un coefficient de Poissons de 0,3. La force appliquée sur chaque rouleau dépend de sa position. L'auteur réalise de tests avec une machine à compression/traction pour extraire l'information du comportement réel du roulement mécanique et valider les résultats obtenus par la méthode EF. L'auteur conclut que les résultats donnés par ANSYS sont similaires aux résultats obtenus par la machine à compression/traction.

Dans notre cas d'étude, nous choisissons de faire une modélisation en trois dimensions, car nous souhaitons connaître le comportement du rouleau de compactage lors de l'application d'une force et suivre l'évolution de la surface de contact. Afin de valider la méthode EF, nous devons corroborer les résultats avec le modèle théorique, pour cette modélisation nous considérons un rouleau de compactage et une surface de contact qui respectent les contraintes des équations de Hertz (section A.2). Les caractéristiques géométriques de ces éléments sont : un cylindre de diamètre de 0,0068 m et de longueur de 0,0118 m et une surface plane (surface de contact) avec une longueur de 0,0118 m, une largeur de 0,008 m et une hauteur de 0,002 m. Les deux éléments (le cylindre et la surface de contact) sont composés d'acier, dont les caractéristiques mécaniques sont : un module de Young de 200 000 MPa et un coefficient de

Poisson de 0,3. La force maximale appliquée pour les deux modèles est de 8000 N avec des intervalles de 1000 N chacun.

- **Description du modèle**

La figure a.6 montre le diagramme que nous utilisons pour modéliser le contact entre le cylindre et la surface de contact.

Pour respecter les contraintes des équations de Hertz, l'analyse est considérée comme statique et le contact sans frottement, le rouleau de compactage possèdent des petites déformations avec des valeurs nécessairement inférieures à 1 %. Ce modèle accomplit alors l'hypothèse des petites perturbations (HPP).

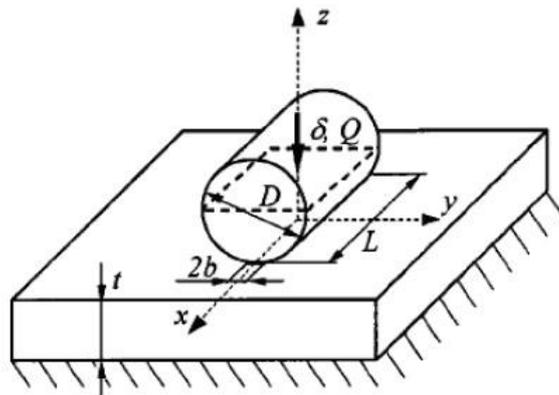


Figure A.6 Diagramme du modèle en contact

- **Maillage**

Le maillage pour les deux éléments est de forme majoritairement quadrilatère. Pour assurer la convergence des champs, nous utilisons la fonction remaillage automatique de « workbench » (WB) et un redimensionnement du maillage dans chaque objet. La taille de maillage permise par la licence académique du logiciel est de 0,0002 m. Dans la surface plane proche du contact avec le cylindre (figure a.7) nous découpons cette surface en sept couches (layers), afin de redimensionner les éléments en contact et diminuer l'erreur de calcul.

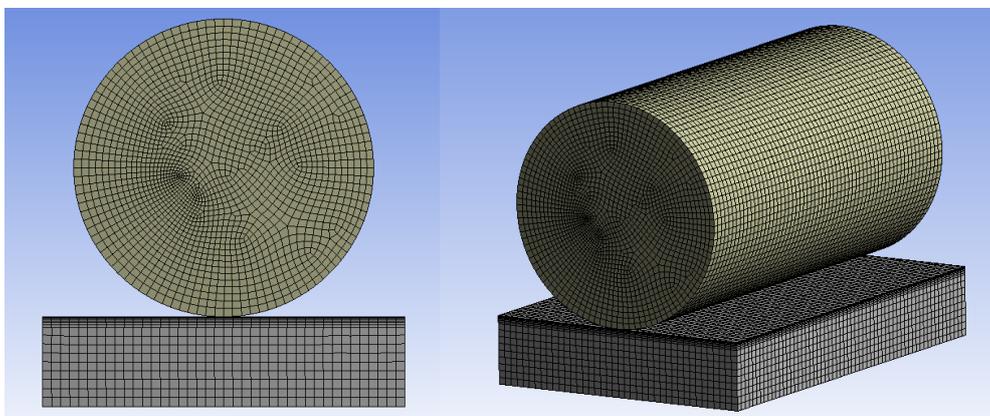


Figure A.7 Maillage pour le modèle FE

La

figure a.8 montre les résultats obtenus pour le modèle théorique (méthode de Hertz) et le modèle EF, l'erreur relative entre les méthodes est comprise entre le 5 % et 15 %. Cette erreur relative est calculée par (A.10) pour une grandeur notée g. Néanmoins l'erreur entre les deux méthodes est acceptable.

$$\frac{|g_{EF} - g_{théorique}|}{|g_{théorique}|} = \text{erreur} \quad (\text{A.10})$$

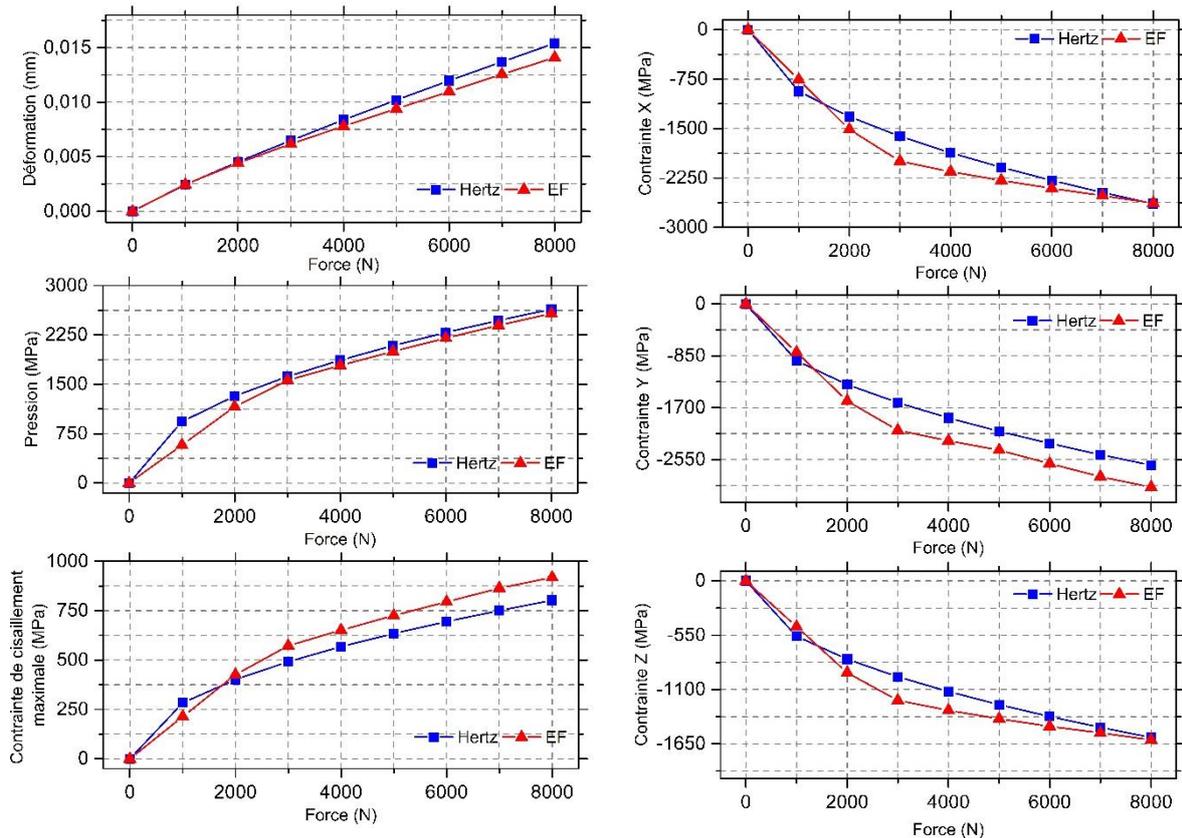


Figure A.8 Résultats obtenus par les méthodes de Hertz et d'EF

A.3 Capteur d'effort

- **Fonctionnement**

Les capteurs d'effort et/ou de couple emploient généralement un élément déformable. Le fonctionnement consiste à mesurer la déviation proportionnelle (déplacement) de l'élément déformable selon la force et/ou le couple appliqué (Bolton, 2001; Thomas R. Kurfess, 2005). Les efforts et/ou couples sont liés à cette déviation et leurs valeurs sont représentées par des signaux électriques (Lewis et al., 1999; Ollero Baturone, 2006). Afin d'éviter des mesures incorrectes, la fréquence naturelle de la structure élastique (élément déformable) du capteur

d'effort (Thomas R. Kurfess, 2005) est considérée. Les capteurs d'efforts fréquemment utilisés sont les capteurs d'efforts à jauges de contrainte et les capteurs d'efforts piézo-électriques.

- **Capteurs d'efforts à jauge de contrainte**

Les capteurs à jauge de contrainte sont des capteurs élastiques résistifs, dont leur résistance est en fonction de la déformation appliquée (Lewis et al., 1999; Ollero Baturone, 2006). La jauge est collée à la surface pour mesurer les efforts appliqués. Lorsqu'un effort est appliqué, la jauge se déforme et elle subit une variation de sa longueur (Uhart, 2014). La variation de sa longueur est proportionnelle à la variation de la résistance. Cette relation est connue comme l'effet piézo-résistif (Lewis et al., 1999; Ollero Baturone, 2006; Uhart, 2014). Les capteurs à jauge de contrainte sont utilisés pour mesurer des efforts et/ou de couples statiques à basse fréquence (Thomas R. Kurfess, 2005). Ces capteurs sont divisés en deux groupes :

- Les capteurs pour mesurer la flexion (figure a.9 a))
- Les capteurs pour mesurer la traction et la compression. Ceux-ci sont divisés en : « S » (figure a.9 b)) et « pan-cake » ou « galette » (figure a.9 c))

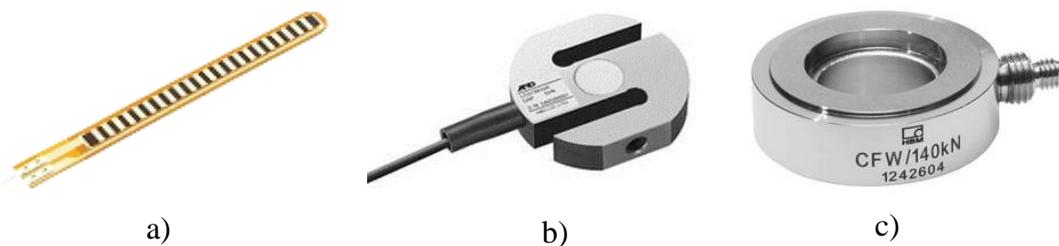


Figure A.9 Capteurs à jauges de contrainte a) Capteur pour mesurer la flexion b) Capteur en « S » c) Capteur du type Pan-cake

Parmi les inconvénients des jauges de déformation, nous pouvons mentionner la non linéarité, la sensibilité à la température, l'hystérésis, le fluage, ... (GoËR & Avril, 1992; Toulmnet, 2002-2003). La réalisation d'un étalonnage permet d'éliminer les inconvénients le plus fréquents de ces capteurs (la non linéarité et la sensibilité à la température) (Uhart, 2014).

- **Capteurs d'efforts piézo-électriques**

Les capteurs d'effort piézo-électrique transforment les efforts mécaniques en un signal électrique (Godefroy, 2000; Toulmnet, 2002-2003; Thomas R. Kurfess, 2005). Ce signal électrique est transformé de signal en courant à signal en voltage (Lewis et al., 1999; Thomas R. Kurfess, 2005). Le signal en voltage est proportionnel à la force appliquée (Lewis et al., 1999; Uhart, 2014). Ces capteurs sont inadéquates pour la mesure des efforts statiques (Thomas R. Kurfess, 2005), car le signal électrique se détériore rapidement après l'application d'un effort

(Uhart, 2014). Les capteurs d'effort piézo-électrique les plus utilisés sont divisés en deux catégories :

- Les capteurs d'efforts piézo-électriques à un composant (figure a.10 a)). Ce composant se présente sous la forme d'un disque dont l'épaisseur est plus ou moins égale à son diamètre (Godefroy, 2000). Les électrodes sont situées dans les faces planes métallisées, pour obtenir les mesures de la variation de tension (Godefroy, 2000).
- Les capteurs d'efforts piézo-électriques à trois composants (figure a.10 b)). Ces composants se présentent sous la forme de plusieurs rondelles, ils sont sensibles à la pression (F_z) et au cisaillement (F_x , F_y) (Uhart, 2014).

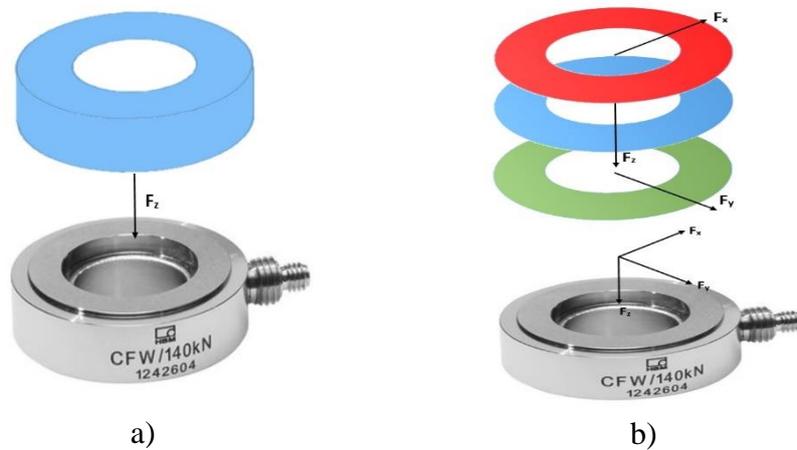


Figure A.10 Capteur piézoélectrique a) Capteur à un composant b) capteur à trois composants

Références

- ABB. (2015). Robotics (datasheet). Retrieved 24/05/2015, from ABB <http://new.abb.com/products/robotics/robot-selector>
- Adel, A., & Salah, K. (2016, 11-14 Dec. 2016). *Model order reduction using artificial neural networks*. Paper presented at the 2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS).
- Ahmad, A., Anderson, K., & Sellgren, U. (2011, 5-7 Oct. 2011). *An approach to stiffness analysis methodology for haptic devices*. Paper presented at the Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2011 3rd International Congress on.
- Akintunde, A. M., Ajala, S. O., & Betiku, E. (2015). Optimization of Bauhinia monandra seed oil extraction via artificial neural network and response surface methodology: A potential biofuel candidate. *Industrial Crops and Products*, 67, 387-394. doi: <http://dx.doi.org/10.1016/j.indcrop.2015.01.056>
- Albu-Schäffer, A., Ott, C., & Hirzinger, G. (2007). A Unified Passivity Based Control Framework for Position, Torque and Impedance Control of Flexible Joint Robots. In S. Thrun, R. Brooks & H. Durrant-Whyte (Eds.), *Robotics Research: Results of the 12th Int. Symposium ISRR* (pp. 5-21). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Alici, G., & Shirinzadeh, B. (2005). Enhanced Stiffness Modeling, Identification and Characterization for Robot Manipulators. *Robotics, IEEE Transactions on*, 21(4), 554-564. doi: 10.1109/tro.2004.842347
- Allery, C., Hamdouni, A., Ryckelynck, D., & Verdon, N. (2011). A priori reduction method for solving the two-dimensional Burgers' equations. *Applied Mathematics and Computation*, 217(15), 6671-6679. doi: <http://dx.doi.org/10.1016/j.amc.2011.01.065>
- Alsmadi, O. M. K., Abo-Hammour, Z. S., & Al-Smadi, A. M. (2011). Artificial neural network for discrete model order reduction with substructure preservation. *Applied Mathematical Modelling*, 35(9), 4620-4629. doi: <https://doi.org/10.1016/j.apm.2011.03.028>
- Ansoategui, Lasmikiz, & Aguirrebeitia. (2010). *Estimación de errores de posición y velocidad en centros de mecanizado de cinco ejes*. Paper presented at the XVIII Congreso Nacional de Ingeniería Mecánica. <http://www.uclm.es/actividades/2010/CongresoIM/cdarticulos.aspx>
- ANSYS. (2015). Theory Reference- Hyperelasticity Documentation for Release 15.0-Mechanical APDL (15.0 ed.). Retrieved from
- Aoustin, Y., & Formal'sky, A. (1997). On the Synthesis of a Nominal Trajectory for Control Law of a One-Link Flexible Arm. *The International Journal of Robotics Research*, 16(1), 36-46. doi: 10.1177/027836499701600103
- Arimoto, S. a. M., F. (1984). Stability and robustness of PID feedback control for robot manipulators of sensory capability. In M. B. a. P. Paul (Ed.), *The first International Symposium* (pp. 18). Cambridge: MIT Press.
- Astray, G., Gullón, B., Labidi, J., & Gullón, P. (2016). Comparison between developed models using response surface methodology (RSM) and artificial neural networks (ANNs) with the purpose to optimize oligosaccharide mixtures production from sugar beet pulp. *Industrial Crops and Products*, 92, 290-299. doi: <http://dx.doi.org/10.1016/j.indcrop.2016.08.011>

- Astrom, K. J. (2002). PID Control Control System Design Lecture Notes for ME 155A. Retrieved from <https://www.cds.caltech.edu/~murray/courses/cds101/fa02/caltech/astrom.html>.
- Aström, K. J., & Hagglund, T. (1995). *PID Control PID Controllers - Theory, Design, and Tuning* (2 ed ed.). USA: ISA The Instrumentation, Systems, and Automation Society.
- Atkinson, J., Hartmann, J., Jones, S., & Gleeson, P. (2007). *Robotic Drilling System for 737 Aileron*. <http://dx.doi.org/10.4271/2007-01-3821>
- Automation, A. I. (2016). F/T Sensor: Gamma Retrieved 02/09/2016, 2016, from http://www.ati-ia.com/products/ft/ft_models.aspx?id=Gamma
- Baeten, J., Bruyninckx, H., & De Schutter, J. (2003). Integrated Vision/Force Robotic Servoing in the Task Frame Formalism. *The Int. J. of Robotics Research*, 22(10-11), 941-954. doi: 10.1177/027836490302210010
- Barrientos, A., & Balaguer, C. (2007). *Fundamentos de robótica, 2a Ed*: McGraw-Hill Interamericana de España S.L.
- Baş, D., & Boyacı, İ. H. (2007). Modeling and optimization II: Comparison of estimation capabilities of response surface methodology with artificial neural networks in a biochemical reaction. *Journal of Food Engineering*, 78(3), 846-854. doi: <http://dx.doi.org/10.1016/j.jfoodeng.2005.11.025>
- Bathe, K. J. (2006). *Finite Element Procedures*: Prentice Hall.
- Bellamine, F. H., & Elkamel, A. (2008). Model order reduction using neural network principal component analysis and generalized dimensional analysis. *Engineering Computations*, 25(5), 443-463. doi: doi:10.1108/02644400810881383
- Bergmann, M. (2004). *Optimisation aérodynamique par réduction de modèle POD et contrôle optimal: application au sillage laminaire d'un cylindre circulaire*. Vandoeuvre-les-Nancy, INPL.
- Bolton, W. (2001). *Mecatronica. Sistemas de control electrónico en ingeniería* (2e ed.): Alfaomega
- Bonithon, G., Joyot, P., Chinesta, F., & Villon, P. (2011). Non-incremental boundary element discretization of parabolic models based on the use of the proper generalized decompositions. *Engineering Analysis with Boundary Elements*, 35(1), 2-17. doi: 10.1016/j.enganabound.2010.07.007
- Bossoni, S. (2009). *Geometric and Dynamlic Evaluation and Optimization of Machining Centers*. Doctor of Science Scientific and Technologie, ETH, Zurich. Retrieved from (18382)
- Bottega, V., Pergher, R., & Fonseca, J. S. O. (2009). Simultaneous control and piezoelectric insert optimization for manipulators with flexible link. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 31, 105-116.
- Boucinha, L. (2013). *A priori model order reduction based on space-time separated representation : Applications to transient dynamics*. INSA de Lyon. Retrieved from <https://tel.archives-ouvertes.fr/tel-01175500>
- Bower, A. F. (2009). Hyperelasticity – time independent behavior of rubbers and foams subjected to large strains *Applied Mechanics of Solids* (pp. 820). USA: CRC Press. Retrieved from <http://solidmechanics.org/contents.php>
- Bruyninckx, H., & Schutter, J. D. (1996). Specification of force-controlled actions in the “task frame formalism”;-a synthesis. *IEEE Transactions on Robotics and Automation*, 12(4), 581-589. doi: 10.1109/70.508440
- Bur, N. (2015). *Reduced order model algorithms for Automated Tape Placement optimisation*. Laboratoire Roberval UMR 7337 Université de Technologie de Compiègne Centre de Recherches de Royallieu CS 60319 60203 Compiègne Cedex FRANCE. Retrieved from <https://hal.archives-ouvertes.fr/tel-01262269>

- Cagin, S. p. (2015). *Reduced and separated meta-models of the scavenging by ports in 2-stroke Diesel engines to use evolutionary algorithms in search space*. Université de Bordeaux. Retrieved from <https://tel.archives-ouvertes.fr/tel-01255129>
- Chanal, H., Duc, E., & Ray, P. (2007). Calibration Accuracy of a Parallel Structure Machine Tool with Respect to Machined Part Quality. In S. Tichkiewitch, M. Tollenaere & P. Ray (Eds.), *Advances in Integrated Design and Manufacturing in Mechanical Engineering II* (pp. 245-257). Dordrecht: Springer Netherlands.
- Chedmail, P., Aoustin, Y., & Chevallereau, C. (1991). Modelling and control of flexible robots. *International Journal for Numerical Methods in Engineering*, 32(8), 1595-1619. doi: 10.1002/nme.1620320806
- Chen, Y. (1999). *Model order reduction for nonlinear systems*. Massachusetts Institute of Technology.
- Chetouani, Y. (2009). Using Artificial Neural networks for the modelling of a distillation column. *International Journal of Computer Science & Applications vol 4*(issue 3), 119-133.
- Chevrie, F., & Guély, F. (1998). Cahier technique No. 191 La logique floue (D. S. e. Technique, Trans.) (1 ed.): Groupe Schneider.
- Chinesta, F., Keunings, R., & Leygue, A. (2013). *The Proper Generalized Decomposition for Advanced Numerical Simulations: A Primer*. Springer International Publishing.
- Condon, M., & Grahovski, G. G. (2010a). *On Model Order Reduction of Perturbed Nonlinear Neural Networks with Feedback*, Berlin, Heidelberg.
- Condon, M., & Grahovski, G. G. (2010b, 2010/). *On Model Order Reduction of Perturbed Nonlinear Neural Networks with Feedback*. Paper presented at the Scientific Computing in Electrical Engineering SCEE 2008, Berlin, Heidelberg.
- Contreras, A. (2014, 3/11/2014). Tecnología de laminado automatizado en materiales compuestos. *ENGINEERING&MATERIALS* Retrieved 28/07/2016, 2016, from <https://materialsbreakthroughs.wordpress.com/2014/11/03/tecnologia-de-laminado/>
- Cook, R. D. (2001). *Concepts and applications of finite element analysis*: Wiley.
- Coriolis-Composites. (2016). TECHNICAL ADVANTAGES Retrieved 18/08/2016, 2016, from <http://www.coriolis-composites.com/products/technical-advantages.html>
- Craig, J. J. (2006). *Robótica*: Pearson Educación de México.
- Crețu, S. S., & Benchea, M. I. (2012). An Elastic-Plastic Analysis of Profile Evolution in Cylindrical Roller Bearings. (44878), 565-574. doi: 10.1115/esda2012-82491
- Croft, K., Lessard, L., Pasini, D., Hojjati, M., Chen, J., & Yousefpour, A. (2011). Experimental study of the effect of automated fiber placement induced defects on performance of composite laminates. *Composites Part A: Applied Science and Manufacturing*, 42(5), 484-491. doi: <https://doi.org/10.1016/j.compositesa.2011.01.007>
- Darji, P. H., & Vakharia, D. P. (2012). *Development of Graphical Solution to Determine Optimum Hallowness of Hollow Cylindrical Roller Bearing Using Elastic Finite Element Analysis*.
- De Luca, A. (2015). Flexible Robots. In J. Baillieul & T. Samad (Eds.), *Encyclopedia of Systems and Control* (pp. 451-458). London: Springer London.
- De Luca, A., & Book, W. (2008). Robots with Flexible Elements. In B. Siciliano & O. Khatib (Eds.), *Springer Handbook of Robotics* (pp. 287-319). Berlin, Heidelberg: Springer Berlin Heidelberg.
- De Schutter, J., & Van Brussel, H. (1988). Compliant Robot Motion II. A Control Approach Based on External Control Loops. *The Int. J. of Robotics Research*, 7(4), 18-33. doi: 10.1177/027836498800700402

- Debout, P., Chanal, H., & Duc, E. (2011). Tool path smoothing of a redundant machine: Application to Automated Fiber Placement. *Computer-Aided Design*, 43(2), 122-132. doi: <https://doi.org/10.1016/j.cad.2010.09.011>
- Degoulange, E., Dauchez, P., & Pierrot, F. (1993, 17-20 Oct 1993). *Determination of a force control law for an industrial robot in contact with a rigid environment*. Paper presented at the Systems, Man and Cybernetics, 1993. 'Systems Engineering in the Service of Humans', Conf. Proceedings., Int. Conf. on.
- Dernoncourt, F. (2011, 28/04/2011). Introduction à la logique floue Retrieved 15/09/2017, 2017, from <http://franck-dernoncourt.developpez.com/tutoriels/algo/introduction-logique-floue/>
- Desai, K. M., Survase, S. A., Saudagar, P. S., Lele, S. S., & Singhal, R. S. (2008). Comparison of artificial neural network (ANN) and response surface methodology (RSM) in fermentation media optimization: Case study of fermentative production of scleroglucan. *Biochemical Engineering Journal*, 41(3), 266-273. doi: <http://dx.doi.org/10.1016/j.bej.2008.05.009>
- DeVlieg, R. (2009). *Robotic Trailing Edge Flap Drilling System*. <http://dx.doi.org/10.4271/2009-01-3244>
- Devlieg, R. (2010). Expanding the Use of Robotics in Airframe Assembly Via Accurate Robot Technology. *SAE Int. J. Aerosp.*, 3(1), 198-203. doi: <http://doi.org/10.4271/2010-01-1846>
- Devlieg, R. (2011). High-Accuracy Robotic Drilling/Milling of 737 Inboard Flaps. *SAE Int. J. Aerosp.*, 4(2), 1373-1379. doi: <http://doi.org/10.4271/2011-01-2733>
- DeVlieg, R., Sitton, K., Feikert, E., & Inman, J. (2002). *ONCE (ONe-sided Cell End effector) Robotic Drilling System*. <http://dx.doi.org/10.4271/2002-01-2626>
- DeVlieg, R., & Szallay, T. (2009). Improved Accuracy of Unguided Articulated Robots. *SAE Int. J. Aerosp.*, 2(1), 40-45. doi: <http://doi.org/10.4271/2009-01-3108>
- Devlieg, R., & Szallay, T. (2010). Applied Accurate Robotic Drilling for Aircraft Fuselage. *SAE Int. J. Aerosp.*, 3(1), 180-186. doi: <http://doi.org/10.4271/2010-01-1836>
- Di, X., Ghosh, B. K., Ning, X., & Tarn, T. J. (2000). Sensor-based hybrid position/force control of a robot manipulator in an uncalibrated environment. *IEEE Transactions on Control Systems Technology*, 8(4), 635-645. doi: 10.1109/87.852909
- Dolo, G. (2017). *Étude expérimentale et modélisation du procédé de placement de fibres avec chauffe laser*. Available from <http://www.theses.fr/2017LORIS435>
- Dombre, E., Duchemin, G., Poignet, P., & Pierrot, F. (2003). Dermarob: A safe robot for reconstructive surgery. *Robotics and Automation, IEEE Transactions on*, 19(5), 876-884. doi: 10.1109/tra.2003.817067
- Drake, S. H. (1978). *Using compliance in lieu of sensory feedback for automatic assembly*. PhD, Massachusetts Institute of Technology. Retrieved from <http://hdl.handle.net/1721.1/16194>
- Duchemin, G. (2002). *Commande et programmation d'un robot d'assistance au geste médical pour des tâches de suivi au contact de tissus mous*. Available from <http://www.theses.fr/2002MON20120>
- Duchemin, G., Maillet, P., Poignet, P., Dombre, E., & Pierrot, F. (2005). A Hybrid Position/Force Control Approach for Identification of Deformation Models of Skin and Underlying Tissues. *IEEE Transactions on Biomedical Engineering*, 52(2), 160-170. doi: 10.1109/tbme.2004.840505
- Dumas, C. (2011). *Développement de méthodes robotisées pour le parachèvement de pièces métalliques et composites*. Docteur, Université de Nantes, France. Retrieved from <https://tel.archives-ouvertes.fr/tel-00712458>

- Dumas, C., Caro, S., Garnier, S., & Furet, B. (2011). Joint stiffness identification of six-revolute industrial serial robots. *Robotics and Computer-Integrated Manufacturing*, 27(4), 881-888. doi: <http://dx.doi.org/10.1016/j.rcim.2011.02.003>
- Dumas, C., Caro, S., Chérif, M., Garnier, S., & Furet, B. (2010, 18-22 Oct. 2010). *A methodology for joint stiffness identification of serial robots*. Paper presented at the Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on.
- Dumon, A. (2011). *Dimensional reduction of type PGD for solving incompressible flows*. Université de La Rochelle. Retrieved from <https://tel.archives-ouvertes.fr/tel-00644565>
- Duriez, C. (2013, 6-10 May 2013). *Control of elastic soft robots based on real-time finite element method*. Paper presented at the Robotics and Automation (ICRA), 2013 IEEE International Conference on.
- Durodola, J. F., Li, N., Ramachandra, S., & Thite, A. N. (2017). A pattern recognition artificial neural network method for random fatigue loading life prediction. *International Journal of Fatigue*, 99, Part 1, 55-67. doi: <http://dx.doi.org/10.1016/j.ijfatigue.2017.02.003>
- Electroimpact. (2016). Composite Manufacturing Retrieved 18/08/2016, 2016, from <https://www.electroimpact.com/Products/Composites/Overview.aspx>
- Everett, S. E., & Dubay, R. (2017). A sub-space artificial neural network for mold cooling in injection molding. *Expert Systems with Applications*, 79, 358-371. doi: <http://dx.doi.org/10.1016/j.eswa.2017.03.013>
- Farley, S. J., Durodola, J. F., Fellows, N. A., & Hernández-Gómez, L. H. (2012). High resolution non-destructive evaluation of defects using artificial neural networks and wavelets. *NDT & E International*, 52, 69-75. doi: <http://dx.doi.org/10.1016/j.ndteint.2012.08.011>
- Ferguene, F., & Toumi, R. (2009). Dynamic External Force Feedback Loop Control of a Robot Manipulator Using a Neural Compensator-Application to the Trajectory Following in an Unknown Environment. *Int. J. Appl. Math. Comput. Sci.*, 19(1), 113-126. doi: 10.2478/v10006-009-0011-9
- Fischer, X. (2010, 2010). [Mécanique des structures et comportements lineaires. Introduction au dimensionnement].
- Fu, K. S., González, R. C., & Lee, C. S. G. (1988). *Robótica: control, detección, visión e inteligencia artificial*: McGraw-Hill.
- Gallet-Hamlyn, C. (2011). multiple-use robots for composite part manufacturing. *JEC Composites*, 4.
- Gallot, G., Dumas, C., Garnier, S., Caro, S., & Furet, B. (2012). *Correction dynamique de trajectoires pour l'usinage robotisé*. Paper presented at the 13ème Colloque National AIP PRIMECA, Le Mont-Dore, France.
- Gangloff, J., & Poignet, P. (2007). Intégration Robot-Capteur. *Techniques de l'Ingenieur*, S7780.
- Gazeau, J.-P., & Zegloul, S. (2012, 19-21 March 2012). *The artist robot: A robot drawing like a human artist*. Paper presented at the 2012 IEEE International Conference on Industrial Technology.
- Getzner, W. (2004). Sylomer SR220 (Datasheet). Retrieved 03/2016 <https://www.getzner.com/en/products/sylomer>
- Getzner, W. G. (2011). Material properties and vibration isolation. Technical information. Retrieved 03/2016 http://www.zvucnaizolacija.com/materijali/02_Werkstoffeigenschaften_und%20Schwingungsisolierung_en.pdf
- Godefroy, J.-C. (2000). Pressions rapidement variables. [Article de base documentaire]. *Techniques de l'ingénieur Métrologie relative aux fluides - Niveaux et pressions, base documentaire : TIB401DUO*(ref. article : r2090).

- GoËR, J.-L. L., & Avril, J. (1992). Capteurs à jauges extensométriques. [Article de base documentaire]. *Techniques de l'ingénieur Mesures de longueurs et d'angles, base documentaire : TIB408DUO*(ref. article : r1860).
- Gu, C. (2011). *Model Order Reduction of Nonlinear Dynamical Systems*. PhD, University of California, Berkeley. Retrieved from <http://www.escholarship.org/uc/item/8mx8b17h> (No. UCB/EECS-2012-217)
- Guillet, J. (2011). *Study and order reduction of linear structured models : application to vehicle dynamics*. Université de Haute Alsace - Mulhouse. Retrieved from <https://tel.archives-ouvertes.fr/tel-00807199>
- Hallenberg, J. (2007). *Robot Tool Center Point Calibration using Computer Vision*. Master Computer Vision, Linköping University, Linköping, Sweden. Retrieved from <http://www.diva-portal.org/smash/get/diva2:23964/fulltext01> (LITH - ISY - EX - - 07/3943 - - SE)
- Hamrock, B. J., Aeronautics, U. S. N., Scientific, S. A., & Branch, T. I. (1984). *Lubrication of Machine Elements*: National Aeronautics and Space Administration, Scientific and Technical Information Branch.
- Hamrock, B. J., Anderson, W. J., Aeronautics, U. S. N., Scientific, S. A., & Branch, T. I. (1983). *Rolling-element Bearings*: National Aeronautics and Space Administration, Scientific and Technical Information Office.
- Harris, T. A. (2001). *Rolling bearing analysis*: Wiley.
- Heres, P. J. (2005). *Robust and efficient Krylov subspace methods for Model Order Reduction*. PhD, Technische Universiteit Eindhoven, Eindhoven.
- Heyberger, C. (2014). *Time-space PGD for solving parameterized problems*. École normale supérieure de Cachan - ENS Cachan. Retrieved from <https://tel.archives-ouvertes.fr/tel-01048636>
- Hirzinger, G., Bals, J., Otter, M., & Stelter, J. (2005). The DLR-KUKA success story: robotics research improves industrial robots. *IEEE Robotics & Automation Magazine*, 12(3), 16-23. doi: 10.1109/mra.2005.1511865
- Hogan, N. (1987, Mar 1987). *Stable execution of contact tasks using impedance control*. Paper presented at the Robotics and Automation. Proceedings. 1987 IEEE International Conference on.
- Hornig, T. L., Ju, S. H., & Cha, K. C. (1999). A Deformation Formula for Circular Crowned Roller Compressed Between Two Flat Plates. *Journal of Tribology*, 122(2), 405-411. doi: 10.1115/1.555376
- Hosoda, K., Igarashi, K., & Asada, M. (1998). Adaptive hybrid control for visual and force servoing in an unknown environment. *IEEE Robotics & Automation Magazine*, 5(4), 39-43. doi: 10.1109/100.740464
- Huang, L., Ge, S. S., & Lee, T. H. (2003). Fuzzy unidirectional force control of constrained robotic manipulators. *Fuzzy Sets and Systems*, 134(1), 135-146. doi: [http://dx.doi.org/10.1016/S0165-0114\(02\)00234-8](http://dx.doi.org/10.1016/S0165-0114(02)00234-8)
- Huebner, K. H. (2001). *The Finite Element Method for Engineers*: Wiley.
- Hutton, D. V. (2003). *Fundamentals of Finite Element Analysis*: McGraw-Hill.
- IFR, I. F. o. R. (2014). Industrial Robot Statistics Retrieved 09/12/2014, 2014, from <http://www.ifr.org/industrial-robots/statistics/>
- Jakel, R. (2010). *Analysis of Hyperelastic Materials with MECHANICA – Theory and Application Examples –*. Paper presented at the Presentation for the 2nd SAXSIM, Technische Universität Chemnitz.
- Jeffries, K. A. (2013). *Enhanced Robotic Automated Fiber Placement with Accurate Robot Technology and Modular Fiber Placement Head*. Paper presented at the SAE 2013

- AeroTech Congress & Exhibition Montreal, Canada.
<https://www.electroimpact.com/WhitePapers/2013-01-2290.pdf>
- Jiang, Z. H. (2008). *Impedance Control of Flexible Robot Manipulators*: INTECH Open Access Publisher.
- Katsura, S., Matsumoto, Y., & Ohnishi, K. (2006). Analysis and experimental validation of force bandwidth for force control. *Industrial Electronics, IEEE Transactions on*, 53(3), 922-928. doi: 10.1109/tie.2006.874262
- Khairudin, M., Mohamed, Z., & Husain, A. R. (2011). Dynamic model and robust control of flexible link robot manipulator. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 9(2), 279-286. doi: <http://dx.doi.org/10.12928/telkomnika.v9i2.698>
- Khalil, W., & Dombre, E. (1999). *Modélisation, identification et commande des robots* (2e édition revue et augmentée ed.). Paris, France: Hermes science.
- Khalil, W., & Dombre, É. (1999). *Modélisation, identification et commande des robots*: Hermes Science Publications.
- Khalil, W., & Lemoine, P. (1999). GECARO: A system for the GEometric CALibration of RObots. *Journal Européen des Systèmes Automatisés (JESA)*, 33(5-6), pp. 717-739.
- Klimchik, A., Pashkevich, A., Caro, S., & Chablat, D. (2012). Stiffness Matrix of Manipulators With Passive Joints: Computational Aspects. *Robotics, IEEE Transactions on*, 28(4), 955-958. doi: 10.1109/tro.2012.2187395
- Kostal, I. (2014, 3-5 Dec. 2014). *A .NET application searching for data in a log file of the KUKA industrial welding robot*. Paper presented at the Mechatronics - Mechatronika (ME), 2014 16th International Conference on.
- Kröger, T., Finkemeyer, B., Thomas, U., & Wahl, F. M. (2004). *Compliant Motion Programming: The Task Frame Formalism Revisited*. Paper presented at the In Mechatronics & Robotics.
- KUKA, R. G. (2003). *KR C2 / KR C3 Expert Programming* (Vol. Release 5.2).
- KUKA, R. G. (2004). Specifications robots KR150, KR180, KR210, KR240, KR270. In K. R. GmbH (Ed.), (04/10/2004 ed., pp. 68). Germany.
- KUKA, R. G. (2012). *Operating and Programming Instructions for System Integrators* (KSS 8.2 SI V4 en ed.). Germany: KUKA Roboter GmbH.
- KUKA, R. G. (2013). KUKA Aktiengesellschaft Retrieved 28/05/2013, 2013, from http://www.kuka-robotics.com/es/company/group/kuka_ag/
- Lahouar, S., Zeghloul, S., & Romdhane, L. (2006). Real-time path planning for multi-dof manipulators in dynamic environment. *International Journal of Advanced Robotic Systems*, 3(2), 20.
- Lartigue, C., Duc, E., & Affouard, A. (2003). Tool path deformation in 5-axis flank milling using envelope surface. *Computer-Aided Design*, 35(4), 375-382. doi: [https://doi.org/10.1016/S0010-4485\(02\)00058-1](https://doi.org/10.1016/S0010-4485(02)00058-1)
- Lecompte-Boinet, G. (2006). Les composites cherchent à réduire leurs coûts. *L'usinenouvelle*.
- Lewis, F. L., Fitzgerald, J. M., Walker, I. D., Cutkosky, M. R., Lee, K.-M., Bailey, R., . . . Liu, K. (1999). Robotics. In F. Kreith (Ed.), *Mechanical Engineering Handbook* (pp. 115): CRC Press LLC. Retrieved from <http://www.itiomar.it/pubblica/dispense/MECHANICAL%20ENGINEERING%20HANDBOOK/ch14.pdf>.
- Li, J., Song, C., Wang, X., & Xiao, J. (2013). *Study on pressure control of automated fiber placement process*. Paper presented at the The 19th International Conference on Composite Materials, Montreal, Canada. <http://confsys.encs.concordia.ca/ICCM19/AllPapers/FinalVersion/LIJ80033.pdf>

- Li, Y. (2011). *Development of a robot-based magnetic flux leakage inspection system*. Saarländische Universitäts- und Landesbibliothek, Saarbrücken. Retrieved from <http://scidok.sulb.uni-saarland.de/volltexte/2011/4420>
- Long, P., Khalil, W., & Martinet, P. (2013, 25-29 Nov. 2013). *Modeling & control of a meat-cutting robotic cell*. Paper presented at the Advanced Robotics (ICAR), 2013 16th International Conference on.
- Long, P., Khalil, W., & Martinet, P. (2014, 14-18 Sept. 2014). *Force/vision control for robotic cutting of soft materials*. Paper presented at the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems.
- Lucas, T. (2008). La dynamique des nouveaux procédés. *L'usine Nouvelle*.
- M'Sirdi, N. K. (2012). *Modelisation et Commande des Robots à Articulations Flexibles. Systèmes Automatiques à Structure Variable*. Marseille, France: LSIS , CNRS UMR - 6168.
- Majdani, O., Rau, T. S., Baron, S., Eilers, H., Baier, C., Heimann, B., . . . Leinung, M. (2009). A robot-guided minimally invasive approach for cochlear implant surgery: preliminary results of a temporal bone study. [journal article]. *International Journal of Computer Assisted Radiology and Surgery*, 4(5), 475-486. doi: 10.1007/s11548-009-0360-8
- Makarov, M. (2013). *Contribution to modeling and robust control of flexible-joint robot manipulators - Applications to interactive robotics*. Doctorat Sciences et technologies, Supélec, France. Retrieved from <https://tel.archives-ouvertes.fr/tel-00844738>
- Martin, C. D., & Porter, M. A. (2012). The Extraordinary SVD. *American Mathematical Monthly*, 119(10), 838-851. doi: 10.4169/amer.math.monthly.119.10.838
- Mendes, N., Neto, P., Norberto Pires, J., & Paulo Moreira, A. (2010). Fuzzy-PI Force Control for Industrial Robotics. In P. Vadakkepat, J.-H. Kim, N. Jesse, A. A. Mamun, T. K. Kiong, J. Baltes, J. Anderson, I. Verner & D. Ahlgren (Eds.), *Trends in Intelligent Robotics: 13th FIRA Robot World Congress, FIRA 2010, Bangalore, India, September 15-17, 2010. Proceedings* (pp. 322-329). Berlin, Heidelberg: Springer Berlin Heidelberg.
- MetalWorking, F. (2015, 04/11/2016). Automated Composite Structure Inspection Moves Forward Retrieved 18/08/2016, 2016, from <http://www.fabricatingandmetalworking.com/2015/11/automated-composite-structure-inspection-moves-forward/>
- Metoui, S. (2015). *Separated representations for the multiscale simulation of the mechanical behavior and damages of composite materials*. PhD, Ecole nationale supérieure d'arts et métiers - ENSAM. Retrieved from <https://pastel.archives-ouvertes.fr/tel-01311101>
- Mezouar, Y., Prats, M., & Martinet, P. (2007). *External Hybrid Vision/Force Control*. Paper presented at the the 13th International Conference Advanced Robotics, Jeju, Korea.
- Ming, L., Huapeng, W., & Handroos, H. (2011, 7-10 Nov. 2011). *Stiffness-maximum trajectory planning of a hybrid kinematic-redundant robot machine*. Paper presented at the IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society.
- Moallem, M. (1996). *Control and design of flexible-link manipulators*. Concordia University. Retrieved from <http://clues.concordia.ca/search/c?SEARCH=TJ%20211.35%20M63%201996>
- Moharos, I., Oldal, I., & Szekrényes, A. (2012). *Finite element methode*. Budapest: Typotex Publishing House.
- Morel, G., & Bidaud, P. (1996, 22-28 Apr 1996). *A reactive external force loop approach to control manipulators in the presence of environmental disturbances*. Paper presented at the Proceedings of IEEE International Conference on Robotics and Automation.

- Morel, G., & Gangloff, J. (2005). *Commande Référencée Capteur en Robotique Médicale et Chirurgicale*. Paper presented at the 5èmes Journées Nationales de la Recherche en Robotique, Guidel, France.
- Negnevitsky, M. (2005). *Artificial Intelligence: A Guide to Intelligent Systems*: Addison-Wesley.
- Neto, P., Mendes, N., Pires, J. N., & Moreira, A. P. (2010, 21-24 Aug. 2010). *CAD-based robot programming: The role of Fuzzy-PI force control in unstructured environments*. Paper presented at the 2010 IEEE International Conference on Automation Science and Engineering.
- Nganga-Kouya, D., & Okou, F. a. (2009). Méthodologie hybride de conception de la commande force-position des robots manipulateurs *Science Sud*, 5(3), 13.
- Norden, N. (1973). On the compression of a cylinder contact with a plane surface. In n. NISTresearchlibrary (Ed.), (Vol. NBSIR 73-243, pp. 74). Washintin D.C.: National Bureau of Standards.
- Norton, R. L., & García, G. S. (1999). *Diseño de máquinas*: Pearson educación, Prentice Hall.
- Nyman, P., & Edberg, M. (2010). *Implementing Multi-Touch Screen for Real-time Control of a Robotic cell*. Master, Universidad Politécnica de Valencia / Chalmers University of Technology, Göteborg, Sweden.
- Ollero Baturone, A. (2006). *Robótica: manipuladores y robots móviles*: Marcombo, S.A.
- Ozgoli, S., & Taghirad, H. D. (2004, 7-9 sept 2004). *Desing of Composite Control For Flexible Joint Robots With Saturing Actuators*. Paper presented at the 5th Iranian Conference on Fuzzy Systems, Tehran.
- Ozgoli, S., & Taghirad, H. D. (2005). *Robust Control for Flexible Joint Robots with a Supervisory Control to Remedy Actuator Saturation*. Paper presented at the 13th ICEE Zanjan , Iran. <http://saba.kntu.ac.ir/eecd/aras/papers/ICEE05Ozgoli.pdf>
- Ozgoli, S., & Taghirad, H. D. (2006). A SURVEY ON THE CONTROL OF FLEXIBLE JOINT ROBOTS. *Asian Journal of Control*, 8(4), 332-344. doi: 10.1111/j.1934-6093.2006.tb00285.x
- Pappu, S. M. J., & Gummadi, S. N. (2017). Artificial neural network and regression coupled genetic algorithm to optimize parameters for enhanced xylitol production by *Debaryomyces nepalensis* in bioreactor. *Biochemical Engineering Journal*, 120, 136-145. doi: <http://dx.doi.org/10.1016/j.bej.2017.01.010>
- Parneix, P., & Lucas, D. (2000). Les matériaux composites en construction navale militaire. [Article de base documentaire]. *Techniques de l'ingénieur Applications des composites, base documentaire : TIB140DUO*(ref. article : am5660).
- Partida, G., Jaramillo, V. H., & Feliú, V. (2010). *Validación experimental de la dinámica de un robot flexible de 3GDL*. Paper presented at the XVIII Congreso Nacional de Ingeniería Mecánica. <http://www.uclm.es/actividades/2010/congresoIM/pdf/cdarticulos/187.pdf>
- Pashkevich, A., Klimchik, A., Chablat, D., & Wenger., P. (2009). Stiness Analysis Of Multi-Chain Parallel Robotic Systems. *Journal of Automation, Mobile Robotics and Intelligent Systems*, 3(3), 75-82.
- Pater, L. (2016). Application of artificial neural networks and genetic algorithms for crude fractional distillation process modeling. *arXiv preprint arXiv:1605.00097*.
- Patra, K., Jha, A. K., Szalay, T., Ranjan, J., & Monostori, L. (2017). Artificial neural network based tool condition monitoring in micro mechanical peck drilling using thrust force signals. *Precision Engineering*, 48, 279-291. doi: <http://dx.doi.org/10.1016/j.precisioneng.2016.12.011>

- Paul, R. (1987, 0-0 March 1987). *Problems and research issues associated with the hybrid control of force and displacement*. Paper presented at the Proceedings. 1987 IEEE International Conference on Robotics and Automation.
- Perpetuo, E. A., Silva, D. N., Avanzi, I. R., Gracioso, L. H., Baltazar, M. P. G., & Nascimento, C. A. O. (2012). Phenol biodegradation by a microbial consortium: application of artificial neural network (ANN) modelling. *Environmental Technology*, *33*(15), 1739-1745. doi: 10.1080/09593330.2011.644585
- Peza Solís, J. F., Silva Navarro, G., & Castro Linares, R. (2009). Modeling and Tip Position Control of a Flexible Link Robot: Experimental Results. *Computación y Sistemas*, *12*, 421-435.
- Pezzotti, S., & Antic, F. (2008). Introducción a la Teoría de Elementos Finitos (F. d. Ingeniería, Trans.). In E. III (Ed.), (Juan Pablo Durruty ed., pp. 30): Universidad Nacional de La Plata.
- Pires, J. N., Ramming, J., Rauch, S., & Araújo, R. (2002). Force/torque sensing applied to industrial robotic deburring. *Sensor Review*, *22*(3), 232-241. doi: doi:10.1108/02602280210433070
- Prats, M., Martinet, P., del Pobil, A. P., & Sukhan, L. (2007, Oct. 29 2007-Nov. 2 2007). *Vision force control in task-oriented grasping and manipulation*. Paper presented at the Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on.
- Prats, M., Pobil, Á. P., & Sanz, P. J. (2012). *Robot Physical Interaction through the combination of Vision, Tactile and Force Feedback: Applications to Assistive Robotics*: Springer Berlin Heidelberg.
- Puttock, M. J., & Thwaite, E. G. (1969). *Elastic Compression of Spheres and Cylinders at Point and Line Contact*. Technical Paper (National Standards Laboratory Technical Paper No. 25). Commonwealth Scientific and Industrial Research Organization, Melbourne, Australia.
- Rackl, M. (2015). *Material testing and hyperelastic material model curve fitting for ogden, polynomial and yeoh models*. Paper presented at the ScilabTEC (7th International Scilab Users Conference), Paris.
- Raibert, M. H., & Craig, J. J. (1981). Hybrid Position/Force Control of Manipulators. *Journal of Dynamic Systems, Measurement, and Control*, *103*(2), 126-133. doi: 10.1115/1.3139652
- Rathjen, S., & Richardson, C. (2013). *High Path Accuracy, High Process Force Articulated Robot*. <http://dx.doi.org/10.4271/2013-01-2291>
- Renders, J. M. (1994). *Algorithmes génétiques et réseaux de neurones: applications à la commande de processus*: Hermès.
- Reyes Cortes, F. (2011). *Robótica: Control de Robots Manipuladores*: Marcombo.
- Rigatos, G. G. (2009). Model-based and model-free control of flexible-link robots: A comparison between representative methods. *Applied Mathematical Modelling*, *33*(10), 3906-3925. doi: https://doi.org/10.1016/j.apm.2009.01.012
- Rigatos, G. G. (2012). A robust nonlinear control approach for flexible-link robots using Kalman filtering. *CYBERNETICS AND PHYSICS 1*(2), 9.
- Rodriguez-Angeles, A., Nijmeijer, H., & van Essen, H. A. (2004). Coordination of Rigid and Flexible Joint Robot Manipulators. In H. Irschik & K. Schlacher (Eds.), *Advanced Dynamics and Control of Structures and Machines* (Vol. 444, pp. 195-215): Springer Vienna.
- Russell, S. J., & Norvig, P. (1995). *Artificial intelligence: a modern approach*: Prentice-Hall, Inc.

- Salah, K. (2017). A novel model order reduction technique based on artificial intelligence. *Microelectronics Journal*, 65, 58-71. doi: <http://dx.doi.org/10.1016/j.mejo.2017.04.007>
- Salisbury, J. K. (1980, 10-12 Dec. 1980). *Active stiffness control of a manipulator in cartesian coordinates*. Paper presented at the Decision and Control including the Symposium on Adaptive Processes, 1980 19th IEEE Conference on.
- San, O., & Maulik, R. (2017). *Neural network closures for nonlinear model order reduction*. Department of Mechanical and Aerospace Engineering. Oklahoma State University. USA. Retrieved from <https://arxiv.org/pdf/1705.08532.pdf>
- Sanfilippo, F., Hatledal, L. I., Zhang, H., Fago, M., & Pettersen, K. Y. (2014, 28-30 July 2014). *JOpenShowVar: An open-source cross-platform communication interface to Kuka robots*. Paper presented at the Information and Automation (ICIA), 2014 IEEE International Conference on.
- Saund, B., & DeVlieg, R. (2013). High Accuracy Articulated Robots with CNC Control Systems. *SAE Int. J. Aerosp.*, 6(2), 780-784. doi: <http://doi.org/10.4271/2013-01-2292>
- Scarcella, J. A. (2008). *Contact Stress Concentration due to Surface Irregularity in Cylindrical Rolling Element Bearing*. Master Technology, Rensselaer Polytechnic Institute, Hartford, CT. Retrieved from <http://www.ewp.rpi.edu/hartford/~ernesto/Su2008/EP/scarcj/Scarcella%20Engineering%20Project%20Final%202008-12-03.pdf>
- Scherer, M. (2008). Les robots plus utiles que jamais. *Industrie et Technologie*, 44-57.
- Schilders, W. (2008). Introduction to Model Order Reduction. In W. H. A. Schilders, H. A. van der Vorst & J. Rommes (Eds.), *Model Order Reduction: Theory, Research Aspects and Applications* (pp. 3-32). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Schrimpf, J. (2013). *Sensor-based Real-time Control of Industrial Robots*. PhD, Norwegian University of Science and Technology, Norway. Retrieved from <http://urn.kb.se/resolve?urn=urn:nbn:no:ntnu:diva-23456> (1503-8181 ; 2013:225)
- Schutt-Ainé, J. E., & Goh, P. (2013, 12-15 Dec. 2013). *Comparing fast convolution and model order reduction methods for S-parameter simulation*. Paper presented at the 2013 IEEE Electrical Design of Advanced Packaging Systems Symposium (EDAPS).
- Shaopeng, F. (2010). Advanced Techniques of Industrial Robot Programming. In E. Hall (Ed.), *Advances in Robot Manipulators* (pp. 79-98): InTech. Retrieved from
- Shigley, J. E., & Mitchell, L. D. (1997). *Diseño en ingeniería mecánica*: McGraw-Hill.
- Sloan, J. (2008, 09/09/2008). ATL and AFP: Signs of evolution in machines process control. *High Performance Composites* Retrieved 28/07/2016, 2016, from <http://www.compositesworld.com/articles/atl-and-afp-signs-of-evolution-in-machine-process-control>
- Smits, R., Bruyninckx, H., Meeussen, W., Baeten, J., Slaets, P., & De Schutter, J. (2006, 3-6 Sept. 2006). *Model Based Position-Force-Vision Sensor Fusion for Robot Compliant Motion Control*. Paper presented at the Multisensor Fusion and Integration for Intelligent Systems, 2006 IEEE International Conference on.
- Sooraksa, P., & Chen, G. (1998). Mathematical modeling and fuzzy control of a flexible-link robot arm. *Mathematical and Computer Modelling*, 27(6), 73-93. doi: [https://doi.org/10.1016/S0895-7177\(98\)00030-2](https://doi.org/10.1016/S0895-7177(98)00030-2)
- Spong, M. W. (1989). *Robot Dynamics and Control*: John Wiley & Sons, Inc.
- Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2005). *Robot Modeling and Control*: Wiley.
- Szabo, B. A., & Babuška, I. (1991). *Finite Element Analysis*: Wiley.
- Tan, J., Xi, N., & Wang, Y. (2002). *Hybrid force/position control of redundant mobile manipulators*. Paper presented at the 15th Triennial World Congress, Barcelona, Spain.
- Thomas R. Kurfess. (2005). *Robotics and Automation Handbook*: CRC Press Taylor & Francis.

- Tissot, G. (2014). *Réduction de modèle et contrôle d'écoulements*. Poitiers.
- Toffano, F. (2015). *An Autonomous System for the Evaluation of a Robot Tool Center Point using Computer Vision*. Bachelor Engineering, Università Degli Studi di Padova Padova, Italy. Retrieved from
- Tomei, P. (1991). A simple PD controller for robots with elastic joints. *IEEE Transactions on Automatic Control*, 36(10), 1208-1213. doi: 10.1109/9.90238
- Tordesillas, A. (1992). *Mathematical approaches to cylindrical elastostatic contact problems with applications in roller coating technology*. Doctor of Philosophy thesis, University of Wollongong. Retrieved from <http://ro.uow.edu.au/theses/1550>
- Toulmnet, G. (2002-2003). Chapitre 4 Les capteurs de force from <http://asi.insa-rouen.fr/enseignement/siteUV/capteur/>
- Tovar-Arriaga, S., Vargas, J. E., Ramos, J. M., Aceves, M. A., Gorrostieta, E., & Kalender, W. A. (2012). A Fully Sensorized Cooperative Robotic System for Surgical Interventions. *Sensors*, 12(7), 9423.
- Truten, V. A. (1971). Deformation in the contact area in measurements by the rolling method, and effect on precision. [journal article]. *Measurement Techniques*, 14(8), 1156-1161. doi: 10.1007/bf00981950
- Tsai, L.-W. (1999). *Robot Analysis The Mechanics of Serial and Parallel Manipulators*. USA: Wiley Interscience.
- Turban, E., Aronson, J. E., & Liang, T.-P. (2004). *Decision Support Systems and Intelligent Systems (7th Edition)*: Prentice-Hall, Inc.
- Uhart, M. (2014). *Amélioration de la précision du placement de fibres robotisé en utilisant un schéma de commande hybride externe force/vision*. Doctorat Science, Université de Nantes, France. Retrieved from
- Uhart, M., Patrouix, O., & Aoustin, Y. (2014). *Controlling the tape's gap in robotized fiber placement process using a visual servoing external hybrid control scheme*. Paper presented at the The International symposium Aircraft Materials (ACMA2014), Morocco. Oral retrieved from <https://hal.archives-ouvertes.fr/hal-01060699>
- Uhart, M., Patrouix, O., Aoustin, Y., & Canou, J. (2013). *Improving Accuracy in Robotized Fiber Placement*. Paper presented at the The 19th International Conference on Composite Materials, Montreal, Canada. <https://halshs.archives-ouvertes.fr/ESTIA/hal-00916362v1>
- Ulrich, S., & Sasiadek, J. Z. (2011, June 29 2011-July 1 2011). *Extended Kalman filtering for flexible joint space robot control*. Paper presented at the American Control Conference (ACC), 2011.
- Utah, T. U. o. (2004). *Topic 7 Contact Stresses and Deformations*. ME EN 7960 - Precision Machine Design Department of Mechanical Engineering.
- Valero, E. F. (2004). *Aportaciones al estudio de las maquinas eléctricas de flujo axial mediante la aplicación del método de los elementos finitos*. PhD, Universidad Politecnica de Cataluña. Retrieved from <http://www.tdx.cat/handle/10803/6294> (8468905755)
- Villani, L., & De Schutter, J. (2008). Force Control. In B. Siciliano & O. Khatib (Eds.), *Springer Handbook of Robotics* (pp. 161-185). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Vladareanu, L., Vladareanu, V., & Schiopu, P. (2013). Hybrid Force-Position Dynamic Control of the Robots Using Fuzzy Application. *Applied Mechanics and Materials*, 245, 15-23. doi: 10.4028/www.scientific.net/AMM.245.15
- Whitney, D. E. (1985, Mar 1985). *Historical perspective and state of the art in robot force control*. Paper presented at the Robotics and Automation. Proceedings. 1985 IEEE International Conference on.
- Williams, J. A., & Dwyer-Joyce, R. S. (2000). Contact Between Solid Surfaces *Modern Tribology Handbook, Two Volume Set*: CRC Press.

-
- Xia, X., Zhu, S., Jia, C., & Niu, R. (2012). Analysis of Contact Stress between Cylindrical Roller and Outer Ring Raceway with Taper Error Using ANSYS. *Modern Applied Science*, 6(12), 5. doi: 10.5539/mas.v6n12p86
- Youbin, P., Vrancic, D., & Hanus, R. (1996). Anti-windup, bumpless, and conditioned transfer techniques for PID controllers. *IEEE Control Systems*, 16(4), 48-57. doi: 10.1109/37.526915
- Zemiti, N. (2005). *Force Control in Minimally Invasive Robotic Surgery*. Université Pierre et Marie Curie - Paris VI. Retrieved from <https://tel.archives-ouvertes.fr/tel-00011767>
- Zengxi, P., & Hui, Z. (2008, 25-27 June 2008). *Robotic machining from programming to process control*. Paper presented at the Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on.
- Zheng, Y. F. F., Yuka. (1991). Robot force sensor interacting with environments. *IEEE Transactions on Robotics and Automation*, 7, 156-164.
- Zhu, G., Duan, X., & Deng, H. (2013). Hybrid Force-Position Fuzzy Control for a Prosthetic Hand. In J. Lee, M. C. Lee, H. Liu & J.-H. Ryu (Eds.), *Intelligent Robotics and Applications: 6th International Conference, ICIRA 2013, Busan, South Korea, September 25-28, 2013, Proceedings, Part I* (pp. 415-426). Berlin, Heidelberg: Springer Berlin Heidelberg.

Thèse de Doctorat

Itzel GONZALEZ OJEDA

Intégration de modèles numériques réduits dans le pilotage de robots possédant des flexibilités dans la dépose de fibres

Numerical model reduction integration into the control of a robot with flexibility in the fiber lay up

Résumé

Ces dernières années, les robots manipulateurs ont été utilisés pour des tâches, telles que le polissage, le fraisage, le placement des fibres, entre autres, auparavant réalisées par des machines spécialisées. Cependant, l'utilisation des robots pour ces tâches en contact, présente des problèmes de précision. Ce contact provoque des flexibilités dans le robot, l'outil et l'environnement.

Cette thèse cible l'amélioration de la précision de la tâche robotisée de dépose de fibres qui nécessite un compactage homogène selon le matériau et la surface. La précision du drapage peut être améliorée par la correction en ligne de la trajectoire du robot et du point de centrage dynamique de l'outil (DTCP).

Cette thèse propose l'asservissement en position/force-pressure en modifiant le TCP, pour commander le compactage selon la surface de contact et prend en compte les flexibilités dues au contact. La pression et les flexibilités sont calculées à partir d'un modèle mécanique numérique réduit grâce à des réseaux de neurones. Cet asservissement a besoin d'une sélection appropriée des correcteurs pour garantir un bon fonctionnement et pour travailler avec divers types de moules. La sélection est réalisée dynamiquement par logique floue. Le DTCP est calculé dynamiquement grâce aux flexibilités de l'outil et il est redéfini lors de l'exécution de la trajectoire du robot pendant le drapage.

Un dispositif expérimental est mis en œuvre en utilisant des éléments similaires aux cellules robotisées de placement des fibres. Une application logicielle intègre l'asservissement proposé dans la baie de commande. Cet asservissement permet d'améliorer la précision de dépose de fibres.

Mots clés

Placement de fibres Robotisé (AFP), modèle numérique réduit, réseau de neurones, commande hybride en position/force-pressure, contrôle en logique floue PI, point de centrage de l'outil (TCP), point dynamique de centrage de l'outil DTCP.

Abstract

In the last years, the robots have been used in the tasks that were previously performed by specialized machines such as polishing, milling, drilling, fibers lay up, among others. However, the use of industrial robots for these contact tasks presents problems of precision. This contact causes flexibility in the robot, in the tool and in the environment.

This thesis is aimed at improving the accuracy of the robotic fiber placement task. This task requires a homogeneous and adapted compaction according to the material and the surface. The lay-up precision can be improved by online robot trajectory correction and the dynamic tool center point (DTCP) definition.

This thesis presents a position/force-pressure control by modifying the TCP, to control the compaction according to the contact surface and which takes into account the flexibility due to the compaction force. Pressure and flexibility are calculated from a mechanical numerical model reduced using neural networks. The force-pressure control needs a selection of corrector to ensure a good performance and to work with several types of molds. This dynamic correction selection is performed by a fuzzy logic approach. The DTCP has been calculated dynamically thanks to the tool flexibility and it is redefined when the robot's trajectory is running during the draping.

An experimental setup has been implemented using industrial elements similar to robotic fiber placement cells. A software application has been developed which integrates the control loop into the robot controller. This loop improves the accuracy of fiber placement.

Key Words

Automated fiber placement (AFP), numerical model reduction, neural network, position/force-pressure control, PI fuzzy logic controller, tool center point (TCP), dynamic tool center point (DTCP).