

**Titre:** Étalonnage de systèmes bi-robots par entraînement de réseaux de neurones  
Title:

**Auteur:** Alaïs Imbert  
Author:

**Date:** 2022

**Type:** Mémoire ou thèse / Dissertation or Thesis

**Référence:** Imbert, A. (2022). Étalonnage de systèmes bi-robots par entraînement de réseaux de neurones [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie.  
Citation: <https://publications.polymtl.ca/10782/>

 **Document en libre accès dans PolyPublie**  
Open Access document in PolyPublie

**URL de PolyPublie:** <https://publications.polymtl.ca/10782/>  
PolyPublie URL:

**Directeurs de recherche:** Lionel Birglen  
Advisors:

**Programme:** Génie aérospatial  
Program:

**POLYTECHNIQUE MONTRÉAL**  
affiliée à l'Université de Montréal

**Étalonnage de systèmes bi-robots par entraînement de réseaux de neurones**

**ALAÏS IMBERT**  
Département de génie mécanique

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise des sciences appliquées*  
Génie aérospatial

Décembre 2022

**POLYTECHNIQUE MONTRÉAL**

affiliée à l'Université de Montréal

Ce mémoire intitulé :

**Étalonnage de systèmes bi-robots par entraînement de réseaux de neurones**

présenté par **Alais IMBERT**

en vue de l'obtention du diplôme de *Maîtrise des sciences appliquées*

a été dûment accepté par le jury d'examen constitué de :

**Farbod KHAMENEIFAR**, président

**Lionel BIRGLEN**, membre et directeur de recherche

**Yuvim CHINNIAH**, membre

## DÉDICACE

*À Christophe Absi*

## REMERCIEMENTS

Je souhaite remercier Lionel Birglen et Bruno Monsarrat pour l'opportunité offerte ainsi que Gabriel Coté, Ali Maghami et John Choi pour leur aide.

J'aimerais également remercier les techniciens qui ont rendu ce projet possible, mais particulièrement Christian Fortin, dont le travail formidable m'a permis d'achever ce projet de maîtrise.

Finalement, je souhaite remercier les membres du jury Farbod Khameneifar et Yuvin Chiniah.

## RÉSUMÉ

L'objectif de recherche est de parvenir à améliorer la précision de robots soumis à des tâches coopératives. Les travaux se focalise sur l'étalonnage commun des robots et cherche à réduire l'erreur de poursuite statique entre les deux brides des deux robots.

La méthode proposée consiste à réaliser un premier étalonnage afin d'obtenir un modèle géométrique de chacun des deux robots et de situer ces modèles dans l'espace de travail. Ensuite, une série de mesure est effectuée afin de pouvoir évaluer l'erreur résiduelle de l'étalonnage pour chaque combinaison de poses du robot maître et du robot esclave. Ces données sont utilisées pour entraîner un réseau de neurones afin de prédire les erreur non-géométriques et les erreurs aléatoires résiduelles à la suite de la calibration géométrique.

L'utilisation de réseau de neurone pour la prédiction de l'erreur de poursuite a tout d'abord été mis à l'épreuve en simulation à partir de données issues d'un projet antérieur portant également sur la calibration de robots coopératifs. Les résultats obtenus ont permis de valider qualitativement la méthode, mais il fut nécessaire de récolter davantage de données expérimentales pour l'obtention de résultats quantitatifs.

Un total de mille poses ont été mesurées pour chacun des robots sur une cellule expérimentale contenant deux KUKA KR300 R2500 et un Leica LTD800. Une première série de mesure est effectuée, puis les robots sont ensuite étalonné à partir de la méthode par axe. Puis, une autre série de mesure compensée est exécutée suite à cet l'étalonnage. Un réseau de neurone a été ensuite été entraîné à partir de 900 données provenant des données compensées, prenant en entrée les positions cartésiennes (XYZABC) des deux robots et en sortie les composantes cartésienne de l'erreur de poursuite entre les deux brides. Les 100 données restantes du groupe de données compensées ont été utilisées pour évaluer la performance de l'approximation de l'erreur. Le méthode proposée permet de réduire l'erreur maximale de 1.2 mm à 0.2 mm par axe pour le positionnement et de  $0.5^\circ$  à  $0.05^\circ$  par angle pour l'orientation. En conclusion, l'utilisation de réseaux de neurones pour la calibration des erreurs non-géométriques permet d'améliorer la précision de l'erreur de poursuite entre les brides de robot travaillant en coopération.

**Mots-clés :** Robotique, Robots coopératifs, Étalonnage, Réseaux de neurones,

## ABSTRACT

The research goal of the presented project is to improve accuracy of cooperative robots. The current work focus on dual robot calibration and aim to reduce static pursuit error between the two flanges.

The proposed method is to first calibrate each robot geometrical model and to locate both bases in the workspace. A first dataset is measured before calibration and then, a second set is measured once the robots are calibrated in order to quantify the accuracy improvement of the first calibration. The second set of data also allows to compute the residual error after calibration for each combinaison of poses of the master and slave robot. This dataset is used to train a neural network to predict the residual error associated to non-geometric constraints and random error left after calibration.

Using a neural network to predict error was first simulated using data from previous work on cooperative robots. The results show that the neural network can accurately approximate error, but additionnal experimental data was necessary to evaluate accuracy.

A total of a thousand randomized poses has been measured on an experimental workspace containing two KUKA KR300 R2500 and a Leica LTD800, before and after axis calibration. A neural network was trained using 900 datapoints from the compensated dataset, using the cartesian position (XYZABC) of both robots as an input in order to predict the flange to flange error. The remaining 100 datapoints is then used to evaluate the accuracy of the method. The successive calibrations allows to reduce the maximal error from 1.2 mm to 0.2 mm by axis for the postioning and from  $0.5^\circ$  to  $0.05^\circ$  by angle for the orientation. To conclude, using neural network prediction in cooperative robots calibration allows to improve accuracy of flange to flange error.

**Keywords:** Robotic, Cooperative Robots, Calibration, Neural Network,

## TABLE DES MATIÈRES

DÉDICACE . . . . .	iii
REMERCIEMENTS . . . . .	iv
RÉSUMÉ . . . . .	v
ABSTRACT . . . . .	vi
TABLE DES MATIÈRES . . . . .	vii
LISTE DES TABLEAUX . . . . .	ix
LISTE DES FIGURES . . . . .	x
LISTE DES ANNEXES . . . . .	xii
CHAPITRE 1 INTRODUCTION . . . . .	1
1.1 Définitions et concepts de base . . . . .	2
1.1.1 Robots coopératifs . . . . .	2
1.1.2 Étalonnage en robotique . . . . .	2
1.1.3 Réseaux de neurones . . . . .	4
1.2 Éléments de la problématique et objectifs de recherche . . . . .	8
1.3 Plan du mémoire . . . . .	9
CHAPITRE 2 REVUE DE LITTÉRATURE . . . . .	11
2.1 Robots Coopératifs . . . . .	11
2.1.1 Recherche académique . . . . .	12
2.1.2 Applications à grande rigidité . . . . .	13
2.1.3 Applications statiques à joint libres . . . . .	14
2.1.4 Applications sans échanges de forces . . . . .	14
2.2 Étalonnage des robots coopératifs . . . . .	14
2.3 Réseaux de neurones pour l'étalonnage . . . . .	16
CHAPITRE 3 VALIDATION DE LA MÉTHODE PAR SIMULATION . . . . .	19
3.1 Étalonnage du processus de rivetage . . . . .	19
3.2 Données disponibles . . . . .	20

3.3	Limitations des données . . . . .	21
3.4	Résultats . . . . .	22
CHAPITRE 4 CONCEPTION DE L'OUTIL ET MÉTHODOLOGIE EXPÉRIMENTALE D'ÉTALONNAGE . . . . .		
	TALE D'ÉTALONNAGE . . . . .	26
4.1	Définition de l'erreur . . . . .	27
4.2	Étalonnage géométrique . . . . .	28
4.3	Appareils disponibles . . . . .	29
4.3.1	KUKA KR300 R2500 ultra . . . . .	29
4.3.2	Polaris Vega XT . . . . .	29
4.3.3	Leica LTD800 . . . . .	31
4.4	Conception des outils . . . . .	32
4.4.1	Cibles Optiques – Polaris . . . . .	33
4.4.2	Cibles optiques – LEICA . . . . .	36
4.5	Processus expérimental . . . . .	37
4.6	Mesures de sécurité . . . . .	40
CHAPITRE 5 OPTIMISATION DES HYPERPARAMÈTRES ET RÉSULTATS FINAUX . . . . .		
	NAUX . . . . .	41
5.1	Données obtenues . . . . .	41
5.2	Optimisation des hyperparamètres . . . . .	41
5.2.1	Répartition des pourcentages et normalisation des paramètres d'entrées	43
5.2.2	Choix du théorème de régression . . . . .	44
5.2.3	Détermination des paramètres d'entrées et de l'architecture du réseau	44
5.2.4	Convergence du réseau de neurone . . . . .	49
5.3	Résultats finaux . . . . .	50
CHAPITRE 6 CONCLUSION . . . . .		
	6.1 Synthèse des travaux . . . . .	55
	6.2 Limitations de la solution proposée et amélioration futures . . . . .	56
RÉFÉRENCES . . . . .		
		57
ANNEXES . . . . .		
		61

## LISTE DES TABLEAUX

Tableau 3.1	Répartition des données mesurées et des données approximées par panneau . . . . .	20
Tableau 3.2	Paramètres d'entraînement - Données Bombardier . . . . .	23
Tableau 3.3	Comparaison entre l'approximation du réseau de neurone et du filtre géométrique . . . . .	23
Tableau 5.1	Paramètres d'entraînement - Paramètres d'entrée . . . . .	46
Tableau 5.2	Paramètres d'entraînement - Architecture . . . . .	48
Tableau 5.3	Paramètres d'entraînement - Itérations . . . . .	49
Tableau 5.4	Paramètres d'entraînement - Résultats Finaux . . . . .	51
Tableau 5.5	Erreur de poursuite maximale et moyenne pour chaque ensemble de données . . . . .	54
Tableau A.1	Erreur moyenne obtenues par itérations des paramètres d'entrées et de l'architecture sur 5 essais . . . . .	62
Tableau A.2	Déviatiion standard de l'erreur obtenue par itérations des paramètres d'entrées et de l'architecture sur 5 essais . . . . .	63

## LISTE DES FIGURES

Figure 1.1	Erreurs de modélisation . . . . .	3
Figure 1.2	Processus d'apprentissage automatique . . . . .	5
Figure 1.3	Schéma d'un réseau de neurones . . . . .	6
Figure 1.4	Exemple de régression linéaire . . . . .	7
Figure 2.1	Différentes catégories d'applications industrielles des robots coopératifs	11
Figure 2.2	Laminage miroir ( <i>Mirror Milling</i> ) . . . . .	13
Figure 2.3	Méthodologie d'étalonnage des robots coopératifs . . . . .	15
Figure 3.1	Positionnement des points mesurés avec le Leica (rouge) et des rivets (noir) sur le panneau de la porte ( <i>Door Panel</i> ) . . . . .	20
Figure 3.2	Approximation de l'erreur par le réseau de neurone et le filtre géomé- trique pour les positions de rails E1 = 1400 et E2= -2000 sur la position XYZ . . . . .	24
Figure 3.3	Approximation de l'erreur par le réseau de neurone et le filtre géomé- trique pour les positions de rails E1 = 1400 et E2= -2000 sur l'orien- tation ABC . . . . .	24
Figure 4.1	Répartition des données expérimentales . . . . .	26
Figure 4.2	Illustration de l'erreur . . . . .	27
Figure 4.3	Robot KUKA KR300 R2500 ultra . . . . .	29
Figure 4.4	Appareil optique Polaris Vega XT, tiré de [1] . . . . .	30
Figure 4.5	Espace de travail du Polaris Vega XT, tiré de [2] . . . . .	30
Figure 4.6	Outil du robot maître . . . . .	32
Figure 4.7	Position des cibles du Polaris sur l'outil maître . . . . .	34
Figure 4.8	Visibilité des cibles du Polaris, tiré de [3] . . . . .	35
Figure 4.9	Positions des cibles du Polaris sur les outils . . . . .	36
Figure 4.10	Position des cibles du Leica sur l'outil maître . . . . .	37
Figure 4.11	Position des repères . . . . .	39
Figure 4.12	Prise de mesure avec le capteur LEICA . . . . .	40
Figure 5.1	Norme euclidienne de l'erreur en XYZ dans le repère de la bride maître	41
Figure 5.2	Erreur en position XYZ . . . . .	42
Figure 5.3	Erreur en orientation ABC . . . . .	42
Figure 5.4	Répartition des données d'entraînement et de validation . . . . .	43
Figure 5.5	Optimisation des paramètres d'entrées du réseau de neurone . . . . .	47
Figure 5.6	Optimisation de l'architecture du réseau de neurones . . . . .	48

Figure 5.7	Optimisation du nombre d'itérations du réseau de neurones . . . . .	50
Figure 5.8	Norme euclidienne de l'erreur en XYZ dans le repère de la base du robot maître . . . . .	50
Figure 5.9	Histogramme de l'erreur en position . . . . .	52
Figure 5.10	Histogramme de l'erreur en orientation ABC . . . . .	52
Figure 5.11	Prédiction de l'erreur en position XYZ . . . . .	53
Figure 5.12	Prédiction de l'erreur en orientation ABC . . . . .	53
Figure B.1	Outil du robot maître . . . . .	64
Figure B.2	Capteurs NDI, tiré de [4] . . . . .	65
Figure B.3	Capteurs NDI - Dessins Techniques . . . . .	66
Figure B.4	Positions des cibles du Polaris sur les outils . . . . .	67
Figure B.5	Capteurs NDI sur l'outil maître . . . . .	68
Figure B.6	Dessins techniques des capteurs Leica, tiré de [5] . . . . .	68
Figure B.7	Vis Pression des capteurs Leica, tiré de [6] . . . . .	69
Figure B.8	Fixation des capteurs Leica sur l'effecteur . . . . .	69
Figure B.9	Capteurs Leica sur l'outil maître . . . . .	70
Figure B.10	Dessin technique de la bride 1, fourni par le CNRC . . . . .	71
Figure B.11	Dessin technique de la bride 2, fourni par le CNRC . . . . .	71
Figure B.12	Dessin technique de la bride 3, [7] . . . . .	72
Figure B.13	Dessin technique de la plaque de maintien . . . . .	72
Figure B.14	Placement des capteurs . . . . .	73

**LISTE DES ANNEXES**

Annexe A	Détail des résultats obtenus par itérations des paramètres d'entrées et de l'architecture . . . . .	61
Annexe B	Détail de la conception de l'outil . . . . .	64

## CHAPITRE 1 INTRODUCTION

Au cours des dernières décennies, l'utilisation de matériaux composites se définit comme étant le futur de l'aérospatial. En effet, ses excellentes propriétés mécaniques, particulièrement le ratio de rigidité par densité, offre une alternative plus solide et plus légère que l'aluminium traditionnellement utilisé. Cette solution est à la fois plus performante, mais également plus écologique, car des avions plus légers requièrent moins de carburant. Dans l'industrie aéronautique, le processus général de fabrication de pièces en composite consiste premièrement à poser de la fibre de carbone sur un moule par un robot industriel. Dépendamment du procédé, les fibres sont soit préimprégnées d'un produit liant, normalement un mélange de résine et d'époxy ou alors le moule, une fois recouvert de fibre de carbone est ensuite mis sous vide puis imbibé de ce produit. La pièce est ensuite cuite dans une étuve, puis démoulée [8,9].

L'usage de composites, bien qu'innovateur, comporte tout de même ses défis notamment au niveau de la cuisson à partir d'énormes étuves qui en plus de prendre énormément d'espace au sein des usines de manufacture, ont également d'important coûts énergétiques. Pour pallier cette problématique, les industries entament une progression vers les composites thermosets qui ne nécessitent pas de période de cuisson dans des étuves. Une fois la fibre de composite tissée, le matériau peut être immédiatement soudé ce qui réduit le temps de production de 80%. Le centre de recherche en aérospatial allemand (DLR) travaille au développement d'un outil permettant de fixer le composite au moment de la pose [10,11].

Cette méthode de tissage et de cuisson simultanée n'est cependant pas encore étendue car cette dernière dégrade énormément les moules utilisés pour le formage des pièces. Il s'agit d'un inconvénient de taille, compte tenu du coût important de ces moules ainsi que des requis de précision de l'industrie aérospatial, cette dégradation entraînant une perte de précision [12].

L'objectif de la collaboration entre DLR et le conseil national de recherche canadien (CNRC) cherche à se dissocier de l'usage de moule en appliquant la fibre de composite sur un second robot dont le mouvement compenserait la géométrie du moule. Les travaux qui suivent s'inscrivent donc dans l'objectif d'améliorer la précision de deux robots travaillant en coopération afin d'accomplir cette application.

## 1.1 Définitions et concepts de base

### 1.1.1 Robots coopératifs

Le terme de coopération lorsqu'appliqué à la robotique peut référer à plusieurs scénarios : un robot coopératif ou cobot peut être un robot travaillant en coopération avec un opérateur humain ou encore deux robots peuvent effectuer une tâche en collaboration, comme des applications de placement (*pick and place*) en respectant l'espace de travail de l'autre sans toutefois interagir. Lorsque sera fait référence à des robots coopératifs dans ce document, il sera attendu à ce que l'on réfère à deux robots couplés travaillant en collaboration sur une même tâche.

Les robots peuvent être couplés de manière physique, par exemple si deux robots sont fixés sur une même pièce, ou peuvent être couplés par une tâche qui requièrent un couplage géométrique et/ou une interaction des forces.

### 1.1.2 Étalonnage en robotique

Dans le domaine de la robotique, il est souvent fait mention de la programmation en ligne (*online*) versus la programmation hors ligne (*offline*). La programmation en ligne fait référence à des trajectoires implémentées manuellement par un opérateur. Par exemple, si un robot devait poser un objet dans une boîte, l'opérateur piloterait le robot vers l'objet puis vers la boîte pour le déposer. La trajectoire pourrait alors être enregistrée et effectuée de nouveau, cette fois de manière automatique. La programmation hors ligne, quant à elle, demanderait à un opérateur de programmer une trajectoire prenant en compte la position de l'objet et de la boîte ainsi que du modèle du robot. La programmation hors-ligne est grandement favorisée dans les applications industrielles, car cette méthode est beaucoup plus rapide et offre plus de flexibilité que la méthode en ligne. Cependant, l'utilisation de ce type de programmation requière une excellente connaissance du modèle du robot. Pour un robot industriel, le modèle géométrique est généralement fourni par le fabricant sous forme de paramètres. Pour s'assurer d'une précision optimale, le robot est soumis à une procédure d'étalonnage, c'est-à-dire que des mesures sont effectuées afin de compenser ce modèle pour qu'il soit le plus représentatif possible de la réalité.

Les sources d'erreur de modélisation sont divisées en deux types : les sources d'erreurs systématiques et les erreurs aléatoires. Le détail des types d'erreurs est présent à la figure 1.1.

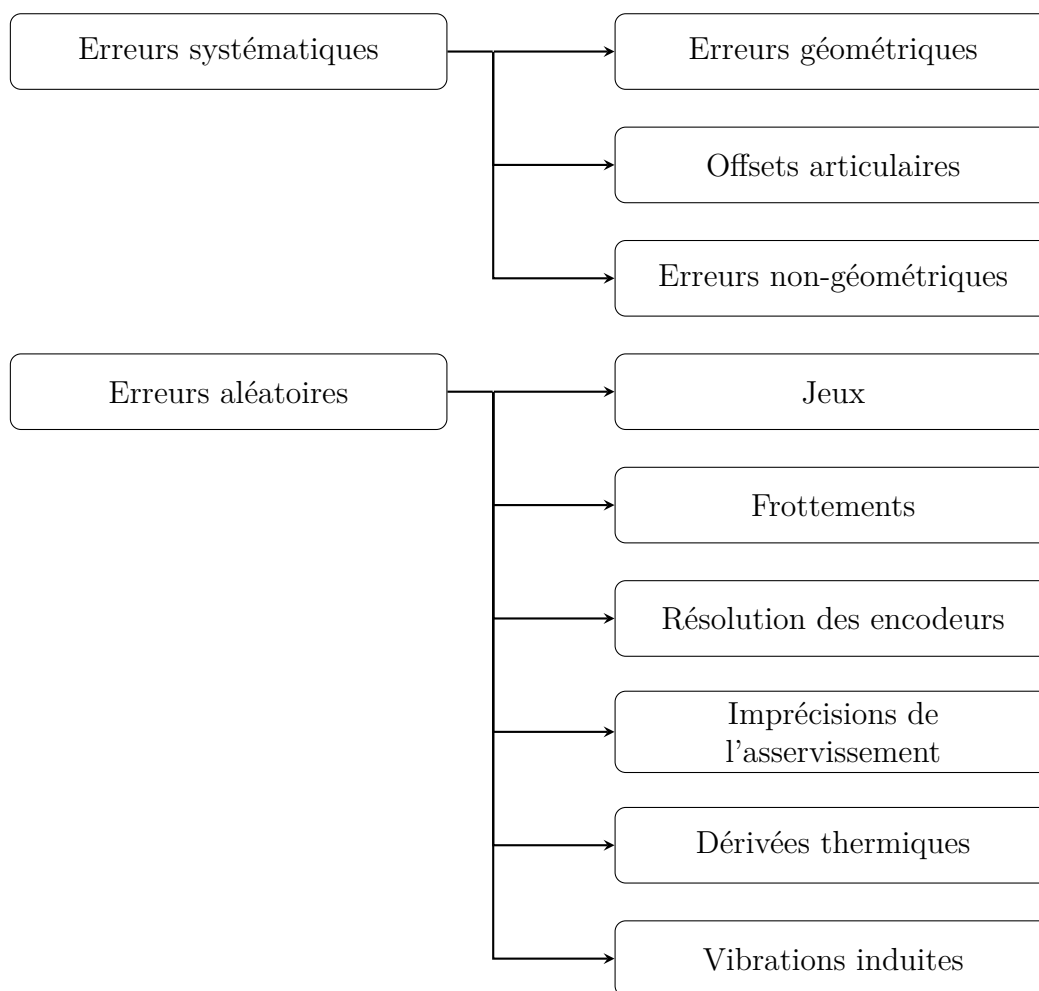


Figure 1.1 Erreurs de modélisation

L'étalonnage s'effectue généralement en quatre étapes :

- **Phase de modélisation** : Dans cette première étape on identifie les paramètres connus du modèle existant et les paramètres que l'on cherche à déterminer par la méthode d'étalonnage et/ou à ajuster.
- **Phase de mesure** : Des mesures sont effectuées à partir d'appareil de métrologie traditionnellement des théodolites, systèmes de mesure à câbles ou une combinaison de lasers de poursuite et de capteurs optiques.
- **Phase d'identification** : À partir des données obtenues précédemment, les paramètres du modèle sont identifiés et/ou corrigés.

- **Phase de compensation et de vérification** : Les changements déterminés à la phase d'identification sont appliqués sur le modèle et ces modifications sont validés soit par simulation ou par la prise de nouvelles mesures.

Pour modéliser un robot, plusieurs étalonnages successifs peuvent être effectués en se concentrant sur différentes erreurs. Plusieurs méthodes de calibration existent dont la méthode globale, la calibration par axe ou la cartographie de l'erreur.

La méthodologie globale d'étalonnage linéaire consiste à identifier une matrice  $\mathbf{A}$  qui une fois multiplié par un vecteur de position  $\mathbf{X}$  donne le vecteur d'erreur sur la position  $\boldsymbol{\delta}$ .

$$\boldsymbol{\delta} = f(\mathbf{x}) = \mathbf{A}\mathbf{X} \quad (1.1)$$

Cette matrice est obtenue par régression des mesures obtenues à partir d'une identification non-linéaire (Newton, gradient, direction conjuguée, Levenberg Maquart) ou par identification linéaire (série de Taylor). L'obtention du vecteur d'erreur permet d'identifier les compensations à apporter à la consigne de positionnement.

La méthode axe par axe sert à identifier les positions relatives liant les différents axes articulaires du robot. Des mesures sont effectuées autour de chaque articulation afin de déterminer la position de son axe en trouvant le centre de ces mesures.

La cartographie des erreurs consiste à effectuer de nombreuses mesures dans l'espace de travail et d'évaluer l'erreur d'un manipulateur en fonction de la position de son outil dans l'espace. Cette dernière méthode est particulièrement propice à l'utilisation d'apprentissage machine [13].

### 1.1.3 Réseaux de neurones

L'amélioration exponentielle des capacités de calcul des processeurs a permis l'essor de nombreuses technologies, notamment l'intelligence artificielle. Cette appellation est très large et peut se rapporter à de nombreuses applications. Dans le cadre de ce projet, l'apprentissage automatique (*machine learning*) sera utilisé. L'apprentissage automatique se qualifie par une première phase d'entraînement qui sert à déterminer un modèle  $h$ , que l'on nomme hypothèse, à partir de données observables ( $\mathbf{X}, \mathbf{Y}$ ). Ensuite, l'hypothèse  $h$ , à partir de nouvelles

données d'entrées  $\mathbf{X}$ , peut prédire les données de sortie  $\mathbf{Y}$  en suivant la même tendance que les données d'entraînement. Ce processus est illustré à la figure 1.2.

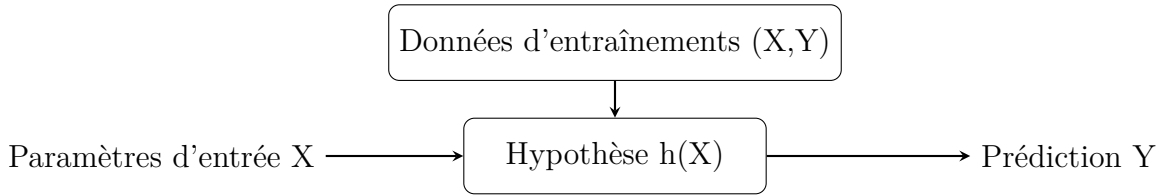


Figure 1.2 Processus d'apprentissage automatique

La prédiction  $\mathbf{Y}$  peut être un résultat quantitatif comme la prédiction du nombre d'années avant qu'une pièce soit endommagée ou encore un résultat qualitatif comme déterminer si une pièce nécessite de la maintenance ou non en fonction de ses performances. Les résultats qualitatifs sont qualifiés de catégorisation et les résultats quantitatifs de régression.

Dans le cas d'une régression linéaire, l'entraînement a pour objectif de prédire les poids  $\theta_0, \theta_1, \dots, \theta_i$  à partir de combinaisons  $\mathbf{X}$  et  $\mathbf{Y}$  connues :

$$\mathbf{Y} = h(\mathbf{X}) = \boldsymbol{\theta} \cdot \mathbf{X} = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \theta_3 X_3 + \dots + \theta_i X_i \quad (1.2)$$

La détermination des poids  $\theta_0, \theta_1, \dots, \theta_i$  est effectuée par descente progressive de gradient sur plusieurs itérations. La progression de cette régression est effectuée à partir de la fonction de perte  $J(\theta)$ . Cette fonction est calculé à partir de l'erreur moyenne carrée (*Mean Square Error*) entre la prédiction de sortie  $h(x_n)$  et la sortie  $y_n$  sur la totalité des données d'entraînement  $m$  :

$$J(\theta_0, \theta_1, \dots, \theta_i) = \frac{1}{2m} \sum_{n=1}^m (h_{\theta}(x_n) - y_n)^2 \quad (1.3)$$

Les poids  $\theta_0, \theta_1, \dots, \theta_i$  sont incrémentés à partir de la dérivée partielle  $i$  de la fonction de perte et d'un facteur  $\alpha$  qui indique le taux d'apprentissage. Ce dernier facteur permet de déterminer la vitesse de la descente de gradient. On a alors :

$$\theta_i = \theta_i - \alpha \cdot \frac{\partial J(\theta_0, \theta_1, \dots, \theta_i)}{\partial \theta_i} \quad (1.4)$$

Les règles de mise à jour des poids varient ensuite en fonction de la technique utilisée. Pour les réseaux de neurones, le principe reste le même, à l'exception que la régression s'effectue sur de multiples plans. Un exemple d'une architecture d'un réseau de neurones est illustré à la figure 1.3.

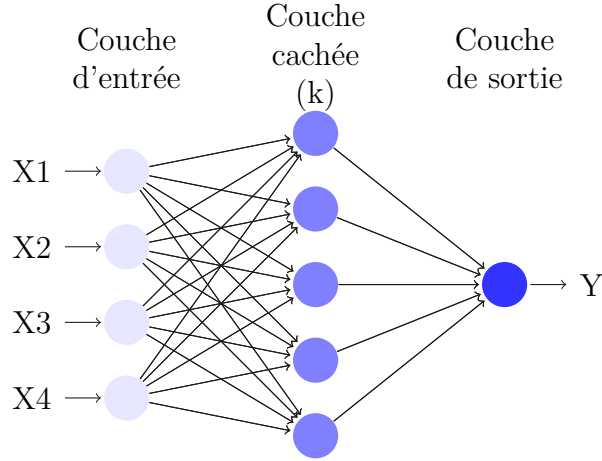


Figure 1.3 Schéma d'un réseau de neurones

L'architecture se constitue du nombre de couche  $k$  et du nombre de neurones par couche  $j$  du réseau de neurones. Les paramètres d'entrées  $X_1, X_2, \dots, X_n$  sont introduits dans la couche d'entrée, les entrées sont alors utilisées pour calculer les valeurs de chaque neurones ou perceptron à l'aide d'une fonction d'activation  $a_i^k$  pour chaque neurone  $j$  d'une couche  $k$ . Chaque fonction d'activation comporte son propre vecteur de poids  $\theta_j^k$ . Les valeurs de ces neurones sont ensuite utilisés pour le calcul de la prochaine couche et ainsi de suite jusqu'à l'obtention d'un résultat de sortie. Chaque neurone est calculé de manière à ce que :

$$a_i^k = g(\theta_j^k X) \quad (1.5)$$

Par exemple, la valeur du premier neurone  $i=1$  de la couche  $k=1$  de la figure 1.3 serait calculée ainsi :

$$a_1^1 = g(\theta_{10}^1 X_0 + \theta_{11}^1 X_1 + \theta_{12}^1 X_2 + \theta_{13}^1 X_3) \quad (1.6)$$

La prédiction finale se calcule à partir des résultats des dernières couches, de manière à ce que :

$$h_{\theta}(X) = a_1^2 = g(\theta_{10}^1 a_1^1 + \theta_{11}^1 a_2^1 + \theta_{12}^1 a_3^1 + \theta_{13}^1 a_4^1) \quad (1.7)$$

De la même manière que pour la régression linéaire, les poids  $\theta_i$  sont incrémentés à chaque itération de l'entraînement selon la loi de régression choisie. Lors de l'entraînement, la majorité des données sont utilisées pour la régression (normalement autour de 70%), mais un certain pourcentage est mis de côté pour l'évaluation de la performance, soit le pourcentage de test (normalement autour de 15%) ainsi que pour l'évaluation de la convergence (normalement autour de 15%). Ces groupes permettent d'évaluer la performance de l'entraînement et de donner les conditions d'arrêts. À chaque itération, la régression est effectuée à partir de la totalité des données d'entraînements. Une nombre fixe d'itérations est défini au début de l'entraînement, si ce nombre d'itérations est atteint sans interruption, l'entraînement prend fin. Le pourcentage de données de test calcule la valeur de la fonction de perte à chaque itérations, si la valeur obtenue est inférieure à l'objectif de performance fixé, l'entraînement prends fin. Finalement, le pourcentage de données consacré à la validation évalue la convergence de l'entraînement du réseau de neurones. Après un certain nombre de validation où les itérations sont consécutivement divergentes, l'entraînement prend fin.

Le choix de ces hyperparamètres est important pour s'assurer que le réseau soit en mesure de prédire correctement les résultats. Un réseau sous-entraîné ne donnera pas les bonnes données de sortie et un réseau trop entraîné sera surspécialisé sur les données d'entraînement (*overfit*) et aura de mauvaises performances sur les autres données. Ces phénomènes sont illustrés à la figure 1.4 pour une régression linéaire.

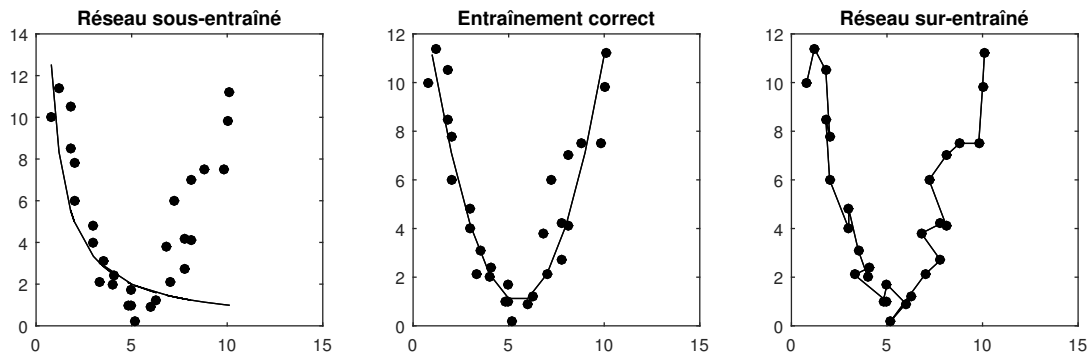


Figure 1.4 Exemple de régression linéaire

Afin d'éviter le dépassement, il est possible de normaliser les données pour que éviter que

certaines paramètres dont l'ordre est plus élevé prenne le dessus sur d'autres paramètres. Une méthode couramment utilisée est la normalisation par la moyenne des données. Chaque paramètre  $X_i$  est normalisé en fonction de la moyenne des données  $\mu_i$ , ainsi que la valeur maximale  $X_{max}$  et la valeur minimale  $X_{min}$  pour chaque paramètre  $i$ , soit :

$$X_i = \frac{X_i - \mu_i}{X_{max} - X_{min}} \quad (1.8)$$

Ce traitement des données permet d'améliorer les performances d'un réseau de neurones cependant, l'élément le plus important pour s'assurer de l'obtention de bons résultats est la qualité ainsi que la quantité des données, un réseau de neurones ne peut avoir des performances qui vont au delà de celles des données qui lui sont fournies.

Il est également à noter que de nombreuses topologies existent lors de la détermination de l'architecture d'un réseau de neurones tel que les réseaux convolutifs, les chaînes de Markov, etc. Ces topologies cherchent à adapter le fonctionnement des réseaux afin d'améliorer leurs performances de calculs. Par exemple, les réseaux convolutifs sont beaucoup utilisés en visionique car les paramètres d'entrées, souvent des images, contiennent énormément d'informations. Le réseau est qualifié de convolutif car une partie du réseau va compresser les données d'entrée avant de passer à l'entraînement. Dans le cadre de ce projet, il ne sera pas nécessaire d'utiliser de topologie particulière. Le réseau sera constitué de une à trois couches cachées en fonction des résultats de l'optimisation.

## 1.2 Éléments de la problématique et objectifs de recherche

Le maintien de la précision lors de tâches collaboratives entre deux robots relève du défi car la présence de deux manipulateurs entraîne une plus grande propagation de l'erreur. Également, une approche multirobot est plus complexe à situer dans l'espace de travail, car elle comporte plus d'éléments.

Pour remédier à cette problématique est proposée une méthode comportant deux étalonnages successifs : un étalonnage cherchant à identifier et compenser les erreurs géométriques et articulaires, suivi d'un second étalonnage ayant pour objectif de représenter l'erreur résiduelle dans l'espace. Pour cette seconde étape sera utilisé une approche par cartographie de l'erreur via un réseau de neurones. Le premier objectif de recherche est de valider qu'un réseau de neurone peut prédire l'erreur résiduelle d'un étalonnage géométrique. Avant de progresser

dans l'établissement d'une méthodologie expérimentale, des tests seront effectués à partir de données existantes.

Ensuite, la conception de la méthodologie expérimentale sera établie en prenant compte la calibration géométrique non seulement des modèles des deux robots, mais aussi de leur position relative dans l'espace de travail. Le choix des capteurs, des appareils de mesures ainsi que de leur placement devra être effectué en conséquence. Le second objectif est donc d'établir une méthodologie d'étalonnage complète simple et abordable faisant utilisation des réseaux de neurones qui pourrait ensuite être introduite dans le milieu industriel.

Finalement, suite à l'obtention de données expérimentales, le troisième et dernier objectif est d'évaluer la performance de la méthode proposée en représentant la progression de l'erreur à chaque étape d'étalonnage.

### 1.3 Plan du mémoire

Le présent mémoire comporte les chapitres suivants :

- **CHAPITRE 2 REVUE DE LITTÉRATURE** : La revue de littérature met en contexte l'utilisation des robots coopératifs dans le milieu industriel, les méthodes d'étalonnage utilisées pour des tâches collaboratives ainsi que des études portant sur l'utilisation des réseaux de neurones dans l'étalonnage.
- **CHAPITRE 3 VALIDATION DE LA MÉTHODE PAR SIMULATION** : Afin de valider la méthode proposée, des premiers tests seront effectués sur des données provenant d'études antérieures sur l'étalonnage de robots coopératifs. Un réseau de neurones sera entraîné à partir des données disponibles. Cette section présente la méthodologie employée, les résultats obtenus ainsi que les limitations de ces premiers tests.
- **CHAPITRE 4 CONCEPTION DE L'OUTIL ET MÉTHODOLOGIE EXPÉRIMENTALE D'ÉTALONNAGE** : Cette section détaille la démarche de conception de l'outil ainsi que les informations relatives à la métrologie. L'obtention des données expérimentales est également détaillée dans ce chapitre.
- **CHAPITRE 5 OPTIMISATION DES HYPERPARAMÈTRES ET RÉSULTATS FINAUX** : Cette section montre le détail de l'optimisation des hyperparamètres du réseau de neurone ainsi que les résultats finaux obtenus à la suite de l'entraînement du réseau.

- **CHAPITRE 6 CONCLUSION** : Ce dernier chapitre effectue une synthèse des travaux et comporte une discussion sur les limitations de l'étude ainsi que des améliorations futures.
- **ANNEXE 1** : Cette annexe comporte le détail des résultats obtenus lors de l'itération des paramètres d'entrée et d'architecture du réseau de neurones.
- **ANNEXE 2** : Cette annexe contient le détail des technicalités entourant la conception des outils.

## CHAPITRE 2 REVUE DE LITTÉRATURE

### 2.1 Robots Coopératifs

Bien que l'implémentation de tâches coopératives sur des robots comporte de nombreux défis, cette approche est loin d'être novatrice. L'approche anthropomorphique, soit deux bras robotisés travaillant de concert, a fait sa première apparition dans l'industrie en 1940 avec le maniement de produits radioactifs, puis de nouveau en 1950 pour l'exploration sous-marine. La NASA a ensuite développé sa propre itération en 1969 qui a plus tard mené à des entités telles que Robonaut et Dextre. L'objectif de ces applications est de remplacer l'intervention humaine dans des zones dangereuses ou difficile d'accès comme les fonds marins ou l'espace. Ensuite, l'utilisation de robots coopératifs s'est davantage répandu avec la progression de l'industrialisation et ses avancées technologiques. Ce développement se justifie par la flexibilité du mécanisme, car une cellule coopérative peut effectuer des tâches aussi variées que complexes. Cette flexibilité, bien qu'économique, représente tout de même un défi technique en terme de contrôle et maintien de la précision [14]. L'apparence anthropomorphique de l'utilisation de robots coopératifs est attrayant pour ses applications domestiques. Néanmoins, dans le cadre de cette recherche le focus sera fait sur les applications industrielles. Les différentes applications industrielles sont détaillées à la figure 2.1.

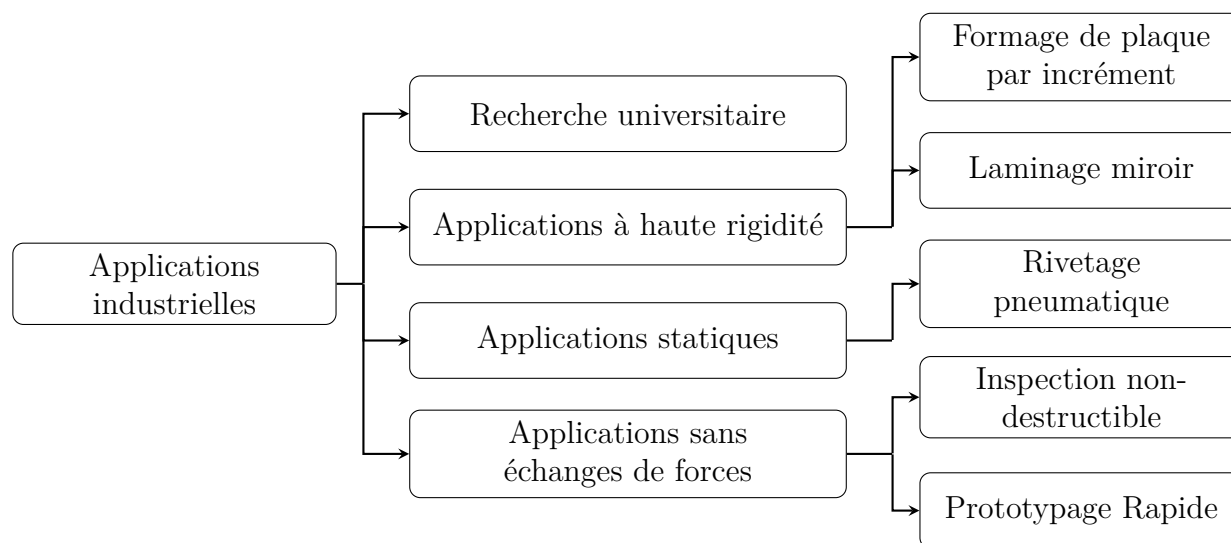


Figure 2.1 Différentes catégories d'applications industrielles des robots coopératifs

La recherche disponible permet de catégoriser les applications industrielles en quatre catégories. La première catégorie est la recherche académique, qui ne cherche pas à implémenter une tâche en particulier, mais qui tente de regrouper de l'information sur la modélisation ainsi que le comportement physique des mécanismes ainsi que l'impact des différents modes de contrôle et d'étalonnage. La seconde catégorie regroupe les applications comprenant des mécanismes à haute rigidité. Cette catégorie utilise davantage des cadres solides plutôt que des robots commerciaux. La troisième catégorie comporte des applications statiques à articulations libres qui impliquent des interactions de forces. Finalement, la dernière catégorie réfère aux applications sans échanges de forces, ces applications impliquent un mouvement dynamique, mais aucun échange de forces entre les deux outils.

### 2.1.1 Recherche académique

La recherche académique est très intéressante, car au lieu de se concentrer sur une tâche, l'emphase est entièrement portée sur la compréhension du comportement physique du mécanisme. La modélisation entre deux robots travaillant en coopération est complexe car la position géométrique, le mouvement dynamique et/ou les forces appliquées par chaque robot doivent être prises en compte, mais également les interactions entre les deux entités.

La modélisation entre deux robots travaillant en coopération peut être effectuée en partageant les forces de l'application entre les deux robots, cette méthode peut s'avérer très complexe lors de l'interaction des forces. Une autre méthode de modélisation est la chaîne cinématique, cette technique modélise les deux robots comme faisant partie d'une chaîne cinématique fermée. Cette approche se porte particulièrement aux robots coopératifs couplés physiquement.

Plusieurs stratégies de contrôle sont étudiées :

- **Contrôle maître/esclave** : Cette approche considère le contrôle d'un seul robot qualifié d'esclave en fonction du second robot, qualifié de maître.
- **Contrôle hybride force/position** : Cette méthode fait usage d'un contrôleur hybride qui mesure à la fois la force et la position de chaque robot.
- **Contrôle de l'impédance** : Les robots sont asservis par un contrôleur d'impédance situé sur chacune des articulations.

L'utilisation de ces stratégies de contrôle est complexe car elles sont plus difficiles à implémenter, et requièrent un plus grand nombre de données. Pour pallier à cette problématique,

des études introduisent l'utilisation de structures moins conventionnelles telles que la logique floue, les réseaux de neurones ou les algorithmes génétiques [15–18].

### 2.1.2 Applications à grande rigidité

Le formage de plaque par incrément (*Incremental Sheet Forming*) est une application des robots coopératifs qui consiste à former une plaque de métal entre deux effecteurs afin d'obtenir une pièce usinée. Cette technique permet de former des pièces à géométrie complexes avec des outils simples demandant moins de forces tout en réduisant les délais pour des pièces produites en faible volume [19].

Dans cette optique existe également le laminage miroir (*mirror milling*) qui consiste à amincir une feuille de métal à l'aide de deux outils placés de chaque côté de la pièce à usiner, un outil maintenant la pièce en place et un autre qui enlève de la matière tel qu'illustré à la figure 2.2

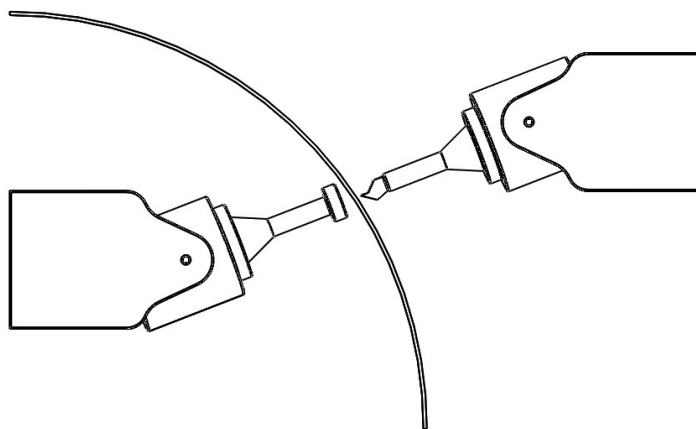


Figure 2.2 Laminage miroir (*Mirror Milling*)

Cette application gagne en popularité dans l'industrie automobile et aéronautique car elle permet de réduire le poids des pièces de manière précise et sans risquer de les endommager [20–22].

Ces applications requièrent une grande rigidité car le procédé nécessite l'application constante de force par les outils. En industrie, les tâches sont effectuées à partir de mécanismes montés sur des portiques comme les solutions proposées par Duffieux Technologies.

### 2.1.3 Applications statiques à joint libres

Une application de rivetage pneumatique a été implémenté par Dassault sur leurs avions Rafale et Falcon ainsi que Bombardier sur leur CJR700. Les robots sont contrôlés à l'aide d'une approche maître-esclave, le robot maître étant équipé de l'outil insérant les rivets et le robot esclave, de la barre de retour (*bucking bar*). Pour cette tâche, les deux robots doivent être précisément étalonnés par rapport à leur environnement. En effet, les deux outils doivent être placé précisément sur la pièce afin de placer le rivet correctement [23,24].

### 2.1.4 Applications sans échanges de forces

Finalemnt, il existe également des applications qui nécessitent un couplage géométrique entre les robots sans toutefois nécessiter d'échanges de forces. Un exemple où la précision de la position relative des robots est importante est le prototypage rapide collaboratif où deux robots sont munies d'outil pour l'usinage d'une pièce. Les cellules bi-robots sont avantageuses dans le prototypage rapide, car elles permettent de réduire le temps d'usinage tout en offrant une plus grande flexibilité et fiabilité [25].

Un autre exemple est l'inspection non-destructive, les robots coopératifs sont utilisés pour détecter des défaut dans des pièces à géométrie complexe. Chaque robot est équipé d'une sonde, chaque outil est ensuite disposé d'un coté de la pièce, une sonde émet des ultra sons et l'autre sonde reçoit le signal qui est ensuite analysé afin de détecter les anomalies entre les deux surfaces. Pour que ce dispositif soit fonctionnel, les sondes doivent être disposées au même endroit de la pièce. Il est donc primordial que le couplage géométrique des deux robots soit très précis [26].

## 2.2 Étalonnage des robots coopératifs

L'étalonnage d'une cellule coopérative est beaucoup plus complexe, car la présence de nombreux éléments entraîne une propagation de l'erreur. Il faut tout d'abord savoir étalonner la position de chaque élément dans l'espace de travail, soit pouvoir localiser avec précision la position de chacun des robots par rapport aux instruments de mesure et ensuite localiser les positions de chacun des robots par rapport à l'autre. Cette tâche peut s'avérer très complexe selon l'espace de travail ainsi que des instruments de mesure mis à disposition. En plus d'étalonner la position des éléments, il est également nécessaire d'effectuer les compensations

nécessaires à la modélisation de chacun des mécanismes afin de réduire la propagation de l'erreur au maximum.

Une des premières approches rencontrées dans la lecture de travaux portant sur ce sujet est la calibration base à base, c'est à dire que l'on souhaite situer des robots l'un par rapport à l'autre. Pour ce faire, les robots sont munis d'un outil d'étalonnage montré à la figure 2.3. Les deux robots doivent mettre la pointe de chacun des outils en contact dans une configuration qualifiée de poignée de main (*handclasp*) tel que montré également à la figure 2.3.

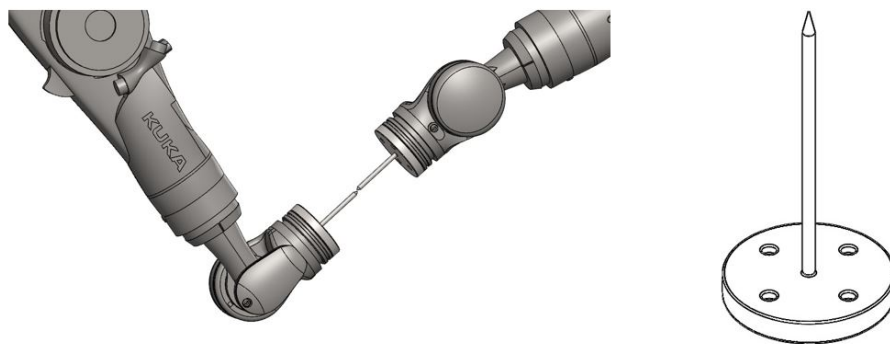


Figure 2.3 Méthodologie d'étalonnage des robots coopératifs

La mesure de quatre points à des endroits différents dans l'espace de travail seront ensuite effectuées dans cette configuration, à l'aide d'appareil de mesure laser. Une regression est ensuite effectuée afin de déterminer la transformation qui lie les bases entre les robots [27]. L'erreur résiduelle de cette approche est de l'ordre de quelques mm, malgré l'utilisation de robots industriels et d'appareils de mesure très précis [28]. Une première étude résulte en une erreur entre la pose mesurée et la pose commandée de 8mm et 1 deg sur un Motoman VA1400 et HP200 ayant une répétabilité de 0,08 mm et 0,06 mm, respectivement, avec l'utilisation d'un appareil de mesure (3D CMM ACtris350) ayant une répétabilité de 0,025 mm et une précision de 0,015 mm. [29]. Un autre article rapporte une erreur de 11 mm et 1,5 deg sur un KUKA KR 16 L6 et un KUKA KR 5 Arc ayant tous les deux une répétabilité de 0.1 mm et un appareil de mesure (CMM API-T3) ayant une résolution de 0.015 mm [30].

D'autres études ont également été exécutées à partir d'outils différents. Un article fait notamment mention d'une approche faisant usage de deux outils munis d'un système d'alignement laser qui émettent un signal sonore lorsque les deux outils sont correctement alignés [31]. Néanmoins, les performances n'ont pas été évaluées de manière quantitative.

Une autre publication effectue la même manipulation, mais à partir de deux caméras et de cibles optiques situées sur les poignets des deux robots. La précision résultante est de 0.1 mm dans le plan X-Y, 0.12 mm dans le plan XZ et 0.12 mm dans le plan YZ avec deux robots ABB IRB ayant une répétabilité de 0.03 mm et deux caméras (MER-231-41GC-P) [32]. La précision est bien meilleure, mais, il est à noter que l'espace de travail des caméras est très restreint.

Une autre approche consiste à utiliser une méthode semblable soit l'utilisation de la méthode globale sur quelques points, mais cherche à étalonner la transformation entre la bride et l'appareil de mesure, ce qui est référencé dans la littérature comme étant la méthode *Hand-Eye Calibration*. Le principe est le même que pour les méthodes décrites précédemment, des points sont mesurés dans l'espace et la transformation est établie à partir de la régression des données obtenues. Les données expérimentales montrent que l'erreur résiduelle de placement suite à cette méthode est de l'ordre d'un mm. Toutefois, l'espace de travail est une fois de plus très réduit [33, 34].

Certains articles proposent également de poser directement un appareil de mesure optique sur l'un des outils afin d'effectuer le suivi de l'erreur entre les brides [35]. Les résultats médians obtenus rapportent des résultats d'erreur médiane de l'ordre de 1.5 mm [36]. Cette méthode peut s'avérer complexe à implémenter car le centre optique n'est pas toujours connu.

L'état de l'art montre que la principale préoccupation dans l'étalonnage de robots coopératifs est de correctement situer les éléments dans l'espace, soit l'appareil de mesure, les bases des robots ainsi que la position des brides ou de l'outils, dépendamment de l'application. Les résultats sont très variables en fonction de l'appareil de mesure utilisé et de l'espace de travail, ce qui montre que dans le processus d'étalonnage le choix de la métrologie et de la disposition des capteurs est tout aussi important que la stratégie de modélisation. Également, ces études montrent que les méthodologies d'étalonnage font toutes usage de régression, ce qui justifie l'utilisation de réseaux de neurones qui s'intégrerait facilement aux processus décrits précédemment.

### 2.3 Réseaux de neurones pour l'étalonnage

Des travaux ont précédemment été effectués sur l'utilisation de réseaux de neurones dans l'étalonnage. Les études détaillées dans les paragraphes suivants se portent néanmoins seulement sur la calibration d'un robot unique. La quasi totalité des publications portent sur la prédiction de compensation à apporter, il existe quelques exceptions qui tentent d'effectuer une

modélisation. Par exemple, une étude utilise un réseau de neurone qui effectue directement une cinématique inverse. Cette approche prend en entrée les position cartésiennes de la pose et donne en sortie les valeurs articulaires prédites à partir d'une couche de 20 neurones. Cette approche permet de réduire l'erreur moyenne de positionnement de 0.852 mm à 0.230 mm [37].

D'autres travaux relevés dans la littérature cherchent à identifier les compensations à apporter à la commande du robot plutôt que d'identifier les paramètres d'un modèle ou, dans le cas de l'étude précédente, de remplacer un modèle. Une seconde étude cherche à améliorer la précision d'un robot à six degrés de liberté en effectuant deux étalonnages successifs soit un premier étalonnage à l'aide de l'optimisation par essaims de particules et un second étalonnage par réseau de neurone afin de prédire l'erreur résiduelle. Le réseau de neurone constitué de 4 couches de 10 neurones prend en entrée la pose du robot en coordonnées cartésienne et prédit les compensations à appliquer sur la commande du robot. Cette technique a été validée sur Webbot à l'aide d'un jumeau digital (*Digital Twin*) et l'erreur moyenne passe de 31 mm à 8 mm lors de leur premier essai en simulation [38]. Une étude similaire, prenant en entrée la pose et calculant en sortie la compensation à appliquer, fait passer l'erreur moyenne de positionnement de 3.522 mm à 0.192 mm. Le réseau de neurone comporte deux couches, respectivement de 64 et 32 neurones et nécessite la mesure de 13 919 poses pour son entraînement [39].

Une autre étude exécute une méthodologie similaire sur un bras mécanique à 8 degrés de liberté en faisant un premier étalonnage articulaires à l'aide des encodeurs installés sur les articulations du mécanisme. Les erreurs cartésiennes sur la bride sont ensuite prédites à partir d'un réseau de neurones comportant une couche de quatre neurones et prenant en entrée les positions de chacune des articulations. L'erreur de calibration moyenne passe de 9 cm à 7 cm à la suite de la calibration articulaire, puis à 4.5 cm suite à la compensation par réseaux de neurones. Ces valeurs sont d'un ordre plus élevées car le bras mécanique est actionné à partir de pistons, ce qui réduit la précision de ce dernier [40].

Les dernières études ont été effectuées sur un robot HH800 et, une fois de plus, un étalonnage géométrique est effectué, suivi d'un second étalonnage par réseau de neurone. Le réseau prend cette fois en entrée les positions des joints lues sur l'encodeur et prédit l'erreur résultante sur la bride du robot à partir d'une couche de 5 neurones. Les résultats sont mesurés à partir d'un appareil de mesure laser ayant une précision de 0.01 mm et une répétabilité de 0.006 mm/m. Les données obtenues montrent que l'erreur moyenne passe de 4.07 mm à 0.70 mm après l'étalonnage géométrique, puis à 0.33 mm suite à l'utilisation d'un réseau de neurone [41–43].

Ces études montrent qu'il est possible d'augmenter la précision d'un robot à l'aide de réseaux de neurones. Les résultats sont variables en fonction de l'appareil de mesure et de l'espace de travail. Toutes les méthodes présentées plus tôt convergent, cependant, certains processus semblent plus faciles à implémenter à une stratégie de contrôle que d'autres. Finalement, les résultats quantitatifs évoqués plus tôt ont été validés par simulation, par conséquent, les performances de ces méthodes ne sont pas exactement ce que l'on retrouverait dans un contexte expérimental.

## CHAPITRE 3 VALIDATION DE LA MÉTHODE PAR SIMULATION

Avant d’entreprendre une démarche expérimentale, l’utilisation de réseaux de neurones pour l’étalonnage de robots coopératifs a été testée sur des données disponibles provenant d’une expérience ultérieure comprenant également des robots travaillant en coopération. En 2004, l’équipe de robotique du CNRC a entrepris une collaboration avec Bombardier Aerospace dans le développement d’une cellule de rivetage automatisée.

Cette cellule est composée de deux robots KUKA KR 500 fixés sur un rail, d’une station de rivetage fixe et d’un système de poursuite laser Leica LTD600. Des panneaux sont soutenus par les deux robots qui travaillent de concert afin de placer et d’orienter la structure correctement dans une riveteuse.

Chaque robot comporte six degrés de liberté et est placé sur un rail qui ajoute un mouvement de translation supplémentaire afin de satisfaire toutes les positions et orientations requises pour le placement de rivets sur le panneau.

### 3.1 Étalonnage du processus de rivetage

Le rail ainsi que les deux robots sont tout d’abord étalonnés de manière indépendante selon les recommandations de leur fabricant respectif. Ensuite, un second étalonnage est effectuée sur les deux robots afin de quantifier leur erreur de poursuite. Pour ce faire, des altères sont montées sur chacune des brides afin de simuler les cas de charge de chacun des panneaux. La position des robots maître et esclave est ensuite mesurée à l’aide de l’appareil de mesure optique Leica afin de déterminer l’erreur de poursuite entre les deux robots. Le panneau de la porte (*Door Panel*) est illustré dans l’espace à la figure 3.1, les points en rouge représentent les points mesurés par le Leica. À partir de ces points, un filtre géométrique est développé par triangulation de l’erreur pour estimer l’erreur de poursuite en fonction de la position géométrique du rivet. La position des rivets est illustré en noir à la figure 3.1.

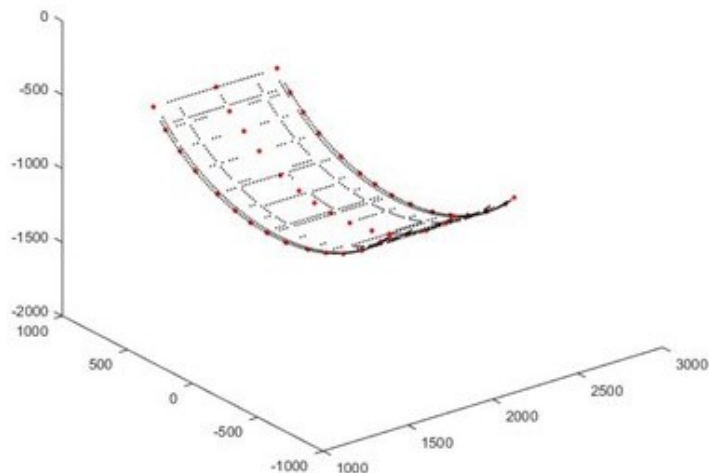


Figure 3.1 Positionnement des points mesurés avec le Leica (rouge) et des rivets (noir) sur le panneau de la porte (*Door Panel*)

La figure 3.1 illustre la répartition physique entre les mesures effectuées par le laser ainsi que les positions des rivets pour le panneau de la porte. L'erreur de poursuite sur les points rouges est mesurée par un appareil de mesure tandis que l'erreur de poursuite sur les points noirs est approximée par le filtre géométrique. Le nombre de points mesurés et le nombre de points approximés par panneau sont recensés dans le tableau 3.1.

Tableau 3.1 Répartition des données mesurées et des données approximées par panneau

Nom du panneau	Nombre de données mesurées	Nombre de données approximées
Door Panel	60	974
Door Surround	184	2562
Lower Panel	119	2606
Upper Panel	119	2301
Belly Panel	94	1419

### 3.2 Données disponibles

Les données disponibles pour l'entraînement du réseau de neurones sont :

- Les positions cartésiennes nominales  $(X, Y, Z)$  [mm]  $(A, B, C)$  [deg] des brides du robot

maître et du robot esclave, chacune dans le repère de leurs bases respectives pour les valeurs mesurées et approximées

- Les charges [kg] sur chacune des brides pour chaque cas de chargement associé à chaque panneau
- La position des robots sur le rail [mm]
- L'erreur de poursuite (X,Y,Z) [mm] (A,B,C) [deg] entre les deux brides dans le repère de la bride du maître mesurées par le Leica LTD600 pour les données mesurées
- L'approximation de l'erreur de poursuite (X,Y,Z) [mm] (A,B,C) [deg] entre les deux brides dans le repère de la bride du maître par le filtre géométrique pour les données approximées

Les paramètres de Denavit-Hartenberg des robots ainsi que les valeurs nominales des transformées bride à bride pour chacun des panneaux sont également disponibles.

### 3.3 Limitations des données

Bien que les données permettent de tester la convergence de l'utilisation de réseaux de neurones en calibration, le contexte de la prise de ces valeurs ne permet pas de statuer sur la précision de la méthode ni d'optimiser correctement le réseau de neurones, dû aux limitations suivantes :

- **Présence de rails** : Tout d'abord, les deux robots sont fixés à un rail. Bien que la position de ce rail soit calibrée, il s'agit d'une source d'erreur additionnelle qui complexifie énormément le problème de calibration. En effet, en plus d'ajouter une erreur de position, le mouvement sur le rail modifie également la position des systèmes de référence soit les bases des robots.
- **Cas sous charge** : Toutes les données sont également exprimées pour des cas sous charge. Il faut donc également prendre en compte les forces et moments additionnels appliqués sur les outils. Pour les mesures effectuées avec le capteur Leica, les poids des panneaux ont été simulés avec des haltères, par conséquent, cela n'adresse pas le changement de position du centre de gravité du cas de charge associé au mouvement des robots
- **Manque de données** : Les données mesurées sont peu nombreuses, par conséquent, cela réduit le nombre d'itérations possibles pour l'entraînement du réseau de neurones.

Le réseau de neurone est entraîné à partir des données mesurées et les résultats obtenus sont comparés aux approximations du filtre.

- **Données approximatives** : La majeure partie des données disponibles sont obtenues à partir des données approximées par le filtre géométrique. Par conséquent, les approximations faites par le réseau de neurones sont comparées à des données qui ont également été approximées ce qui entraine une évaluation de l'erreur difficilement quantifiable.

### 3.4 Résultats

Tout d'abord, les données disponibles détaillées à la section 3.2 ont été normalisées afin de niveler les entrées d'entraînement, car l'ordre de grandeur des valeurs d'entrées est très différent dépendamment s'il s'agit d'une position [mm] ou d'une orientation [deg], par exemple. Chaque paramètre  $X_i$  est ainsi normalisé en fonction de la moyenne des données, afin de prévenir les dépassements (*overfit*) et donc d'obtenir des résultats optimaux. L'étude des paramètres d'entrée par itération a permis de déterminer que les meilleures performances étaient obtenues lorsque toutes les données provenaient des mêmes coordonnées de rail. Par conséquent, au lieu d'utiliser la position des rails comme paramètre d'entrée pour le réseau de neurones, un réseau de neurones a été entraîné pour chaque combinaison de positions de rail, simulant ainsi deux robots à base fixe. Il a également été établi qu'il n'était pas nécessaire de prendre en compte la charge appliquée sur les brides en entrée. L'introduction des paramètres de positionnement des rails ainsi que les charges causaient un dépassement (*overfit*) des données.

Un réseau de neurones a été entraîné pour chacune des combinaisons de rails :

- **Paramètre d'entrée** : Les positions cartésiennes nominales (X,Y,Z) [mm] (A,B,C) [deg] des brides du robot maître et du robot esclave, chacun dans le repère de leurs bases respectives.
- **Paramètre de sortie** : L'erreur de poursuite  $(\delta_X, \delta_Y, \delta_Z)$  [mm]  $(\delta_A, \delta_B, \delta_C)$  [deg] [mm/deg] entre les deux brides dans le repère de la bride du maître

Le théorème de régression utilisé est le théorème de Bayes. S'agissant d'un modèle basé sur l'étude statistique, ce théorème était celui qui se prêtait le mieux au nombre limité de données d'entraînement. L'architecture, soit le nombre de couches et de neurones par couche est déterminé de manière itérative.

La faible quantité de données disponible rend l'optimisation de l'entraînement difficile et la manipulation des paramètres d'entrée cause rapidement des dépassements (*overfit*) dans les résultats. Néanmoins, les résultats obtenus montrent, tel qu'illustré aux figures 3.2 et 3.3 pour les positions des robots sur les rails à 1400 mm (E1) et -2000 mm (E2), que le réseau de neurones parvient à prédire l'erreur de poursuite. Les paramètres d'entraînement sont standards.

Tableau 3.2 Paramètres d'entraînement - Données Bombardier

Paramètres d'entraînement	
Paramètres d'entrées	$X_M, Y_M, Z_M \quad A_M, B_M, C_M \quad X_S, Y_S, Z_S \quad A_S, B_S, C_S$
Paramètres de sortie	$\delta_X, \delta_Y, \delta_Z, \delta_A, \delta_B, \delta_C$
Pourcentage d'entraînement	70%
Pourcentage de validation	15%
Pourcentage de test	15%
Architecture	16
Nombre d'itérations	5000
Objectif de performance	0.001
Nombre de validation	6
Théorème de régression	<i>bayes</i>

Ce réseau de neurone a été entraîné à partir de 104 données mesurées et vérifiées auprès de 1900 données approximées. La comparaison avec les approximations du filtre est présente à la table 3.3 à titre indicatif.

Tableau 3.3 Comparaison entre l'approximation du réseau de neurone et du filtre géométrique

	Position [mm]			Orientation [deg]		
	X	Y	Z	A	B	C
Erreur maximale	1.6006	2.1318	1.1172	0.0451	0.1035	0.0877
Erreur moyenne	0.5029	0.4331	0.1869	0.0114	0.0248	0.0208

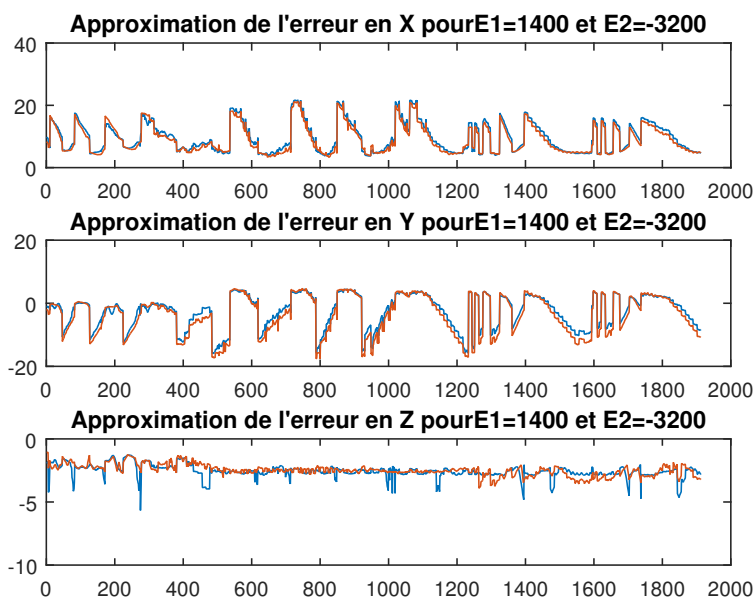


Figure 3.2 Approximation de l'erreur par le réseau de neurone et le filtre géométrique pour les positions de rails  $E1 = 1400$  et  $E2 = -2000$  sur la position XYZ

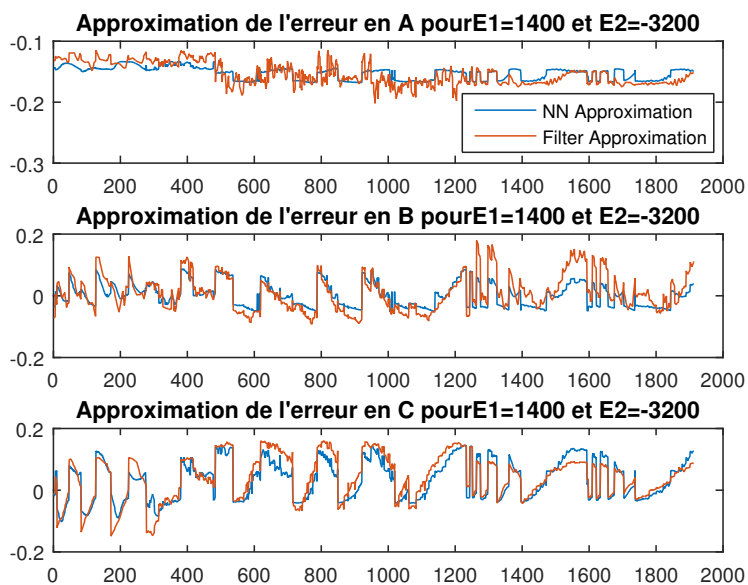


Figure 3.3 Approximation de l'erreur par le réseau de neurone et le filtre géométrique pour les positions de rails  $E1 = 1400$  et  $E2 = -2000$  sur l'orientation ABC

Les résultats obtenus permettent de valider qualitativement la convergence de la méthode de calibration par réseau de neurones. Cependant, il n'est pas possible de statuer sur les performances quantitatives de la méthode proposée considérant qu'il n'y a pas assez de données pour effectuer à la fois l'entraînement et l'évaluation des résultats et que les approximations faites par le filtre comportent également leur propre erreur de prédiction.

## CHAPITRE 4 CONCEPTION DE L'OUTIL ET MÉTHODOLOGIE EXPÉRIMENTALE D'ÉTALONNAGE

La convergence de l'utilisation des réseaux de neurones dans l'étalonnage a été précédemment validée par simulation sur les données du projet de Bombardier. Néanmoins, tel que mentionné plus tôt, le contexte des informations ne permet pas d'évaluer la performance de la méthode proposée. Des données expérimentales ont donc été obtenues dans un cadre contrôlé afin d'obtenir des résultats quantifiables. La répartition des données expérimentales est illustrée à la figure 4.1

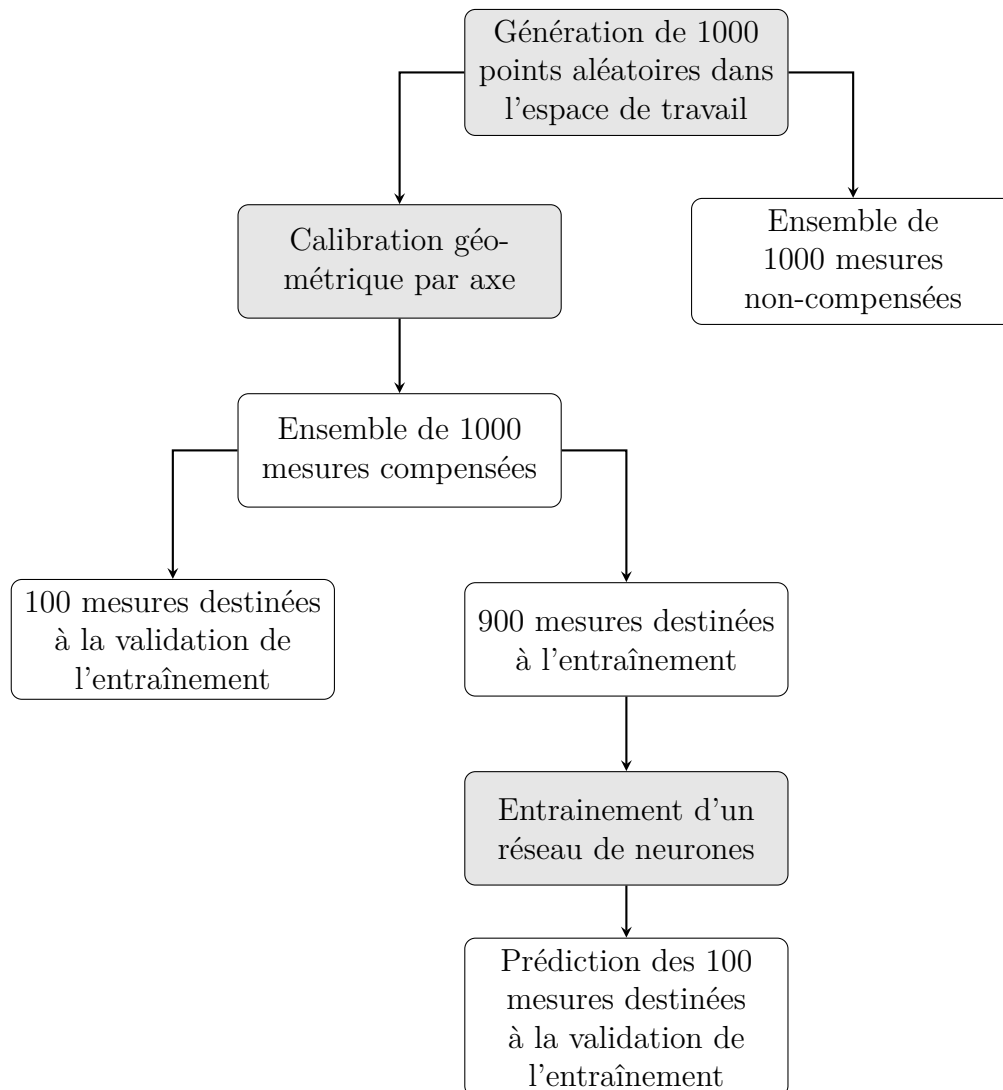


Figure 4.1 Répartition des données expérimentales

Une première série de donnée est obtenue sans aucun type d'étalonnage, soit les données non-compensées. Les robots sont ensuite calibrés à l'aide du module d'étalonnage par axe inclus dans Robot DK. Une seconde série de données est ensuite mesurée, cette fois en appliquant les corrections résultant de l'étalonnage, soit les données compensées. Cette dernière série de données est ensuite utilisée pour entraîner un réseau de neurones à prédire l'erreur résiduelle entre la position nominale et la position mesurée à la suite de la compensation. Ce réseau permet ensuite de déterminer la compensation à appliquer à la commande pour réduire l'erreur de poursuite. Le réseau entraîné est ensuite utilisé sur une fraction des données non compensées qui n'a pas été utilisée dans l'entraînement. Les valeurs obtenues sont ensuite comparées aux valeurs mesurées afin d'évaluer la performance de méthode. La précision de la méthode est donc obtenue à partir de résultats simulés.

#### 4.1 Définition de l'erreur

Pour commencer, il est important de caractériser le contexte expérimental. La cellule de test comporte deux robots KUKA KR300 R2500 fixés au sol dont les bases sont munies de marqueurs de sol permettant d'identifier avec précision la position du repère base de chacun des robots ainsi que la position des bases les unes par rapport aux autres. Il est donc possible pour l'instrument de mesure de déterminer la position relative des brides par rapport à chacune des bases. Ces positions sont référencées à la figure 4.2.

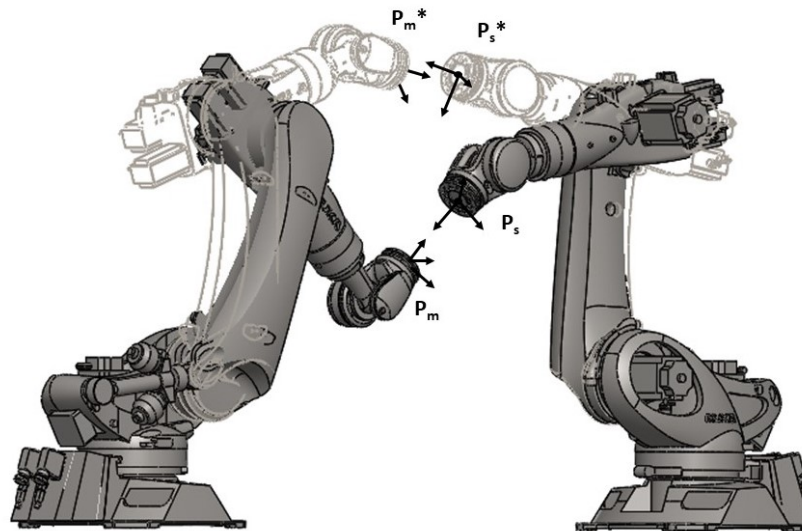


Figure 4.2 Illustration de l'erreur

Les positions mesurées et nominales des deux robots sont exprimées dans un repère commun et fixe soit la base du maître. L'erreur de poursuite ( $\delta_p$ ) est définie comme étant l'erreur de position cumulée soit la différence entre la position nominale du robot esclave ( $\mathbf{P}_s$ ) et du robot maître ( $\mathbf{P}_m$ ) et la position mesurée du robot esclave ( $\mathbf{P}_s^*$ ) et du robot maître ( $\mathbf{P}_m^*$ ) dans le repère de la base du maître.

$$\delta_p = (\mathbf{P}_s - \mathbf{P}_s^*) + (\mathbf{P}_m - \mathbf{P}_m^*) \quad (4.1)$$

Les résultats sont exprimés dans le repère de la bride maître en utilisant la position mesurée exprimée dans le repère maître afin d'avoir une référence absolue dans un repère qui reste fixe. Les commandes sont effectuées dans le repère des bases des robots, il est donc pertinent d'obtenir les compensations à appliquer dans ce même repère.

## 4.2 Étalonnage géométrique

L'étalonnage géométrique est effectué à partir du module intégré au logiciel Robot DK. La méthode utilisée est l'étalonnage par axe qui consiste à mesurer les positions de l'outil lors de la rotation autour d'une articulation pour identifier la position de l'axe de rotation. En procédant articulation par articulation, il est possible de déterminer des relations géométriques entre chacun de ces axes. Le centre de la base étant déjà précisément calibré, seuls les trois derniers axes du robot, soit les axes J4, J5 et J6 ont été étalonnés.

Cette méthode, bien que très précise, comportent certains désavantages. Tout d'abord, la technique offre seulement un modèle relatif, cependant, dans le cas de la présente cellule, il ne s'agit pas d'un obstacle compte tenu de la présence de marqueurs de sol qui permettent de situer les mesures dans l'espace de la cellule. Ensuite, un autre désavantage est le temps requis, en effet, effectuer des mesures axe par axe peut être très long lorsque comparé à d'autres méthodes, surtout avec un capteur laser tel que Leica qui requière des manipulations d'un opérateur à chaque pose pour effectuer le suivi de ses cibles. L'utilisation du capteur Polaris tente de palier à cet obstacle en automatisant les manipulations. Finalement, cette méthode a tendance à induire des erreurs associées à l'effet de pesanteur. L'inclinaison ainsi que la position des articulations en aval peut introduire des variations de positions due à l'effet de la gravité qui sont difficilement modélisable par cette technique.

Le choix s'est finalement porté sur cette méthode de part ses performances de précision et

répétabilité, mais aussi de part sa facilité d'intégration au logiciel Robot DK.

### 4.3 Appareils disponibles

#### 4.3.1 KUKA KR300 R2500 ultra

La cellule comporte deux robots KUKA KR300 R2500 ultra en configuration maître-esclave. Il s'agit de deux manipulateurs industriels comportant six degrés de liberté ainsi qu'une répétabilité de  $\pm 0.06$  mm.

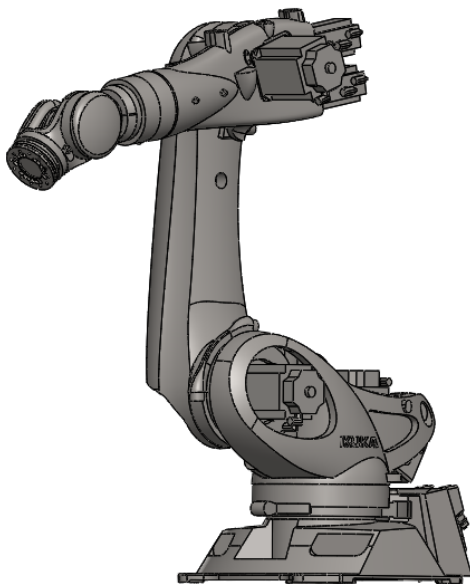


Figure 4.3 Robot KUKA KR300 R2500 ultra

#### 4.3.2 Polaris Vega XT

L'appareil de mesure à l'aide duquel le processus d'étalonnage sera implémenté est le Polaris Vega XT, appareil optique fourni par Northern Digital Inc (NDI). Originellement conçu pour des applications biomédicales, cet appareil possède une haute précision de mesure, une grande vitesse de localisation ainsi qu'un faible temps de latence. Comme ce modèle a la capacité de localiser jusqu'à 25 cibles, il s'agit d'une solution économique pour l'étalonnage simultané de multiples robots.



Figure 4.4 Appareil optique Polaris Vega XT, tiré de [1]

Le Polaris Vega comporte deux ensembles d'illuminateurs et de capteurs situés sur la gauche et la droite de l'appareil. Les illuminateurs sont deux séries de diodes qui émettent de la lumière infrarouge (IREDs). La lumière est ensuite soit réfléchi sur des cibles dites passives ou provoque une réponse par des cibles dites actives qui seront ensuite lues par les capteurs. Les données lues sont ensuite compilées par un algorithme géométrique qui permet d'identifier les positions des cibles. Le Polaris Vega XT possède également un laser de positionnement au centre du système ainsi que deux indicateurs lumineux pour indiquer l'alimentation ainsi que la communication d'erreur. L'appareil est alimenté par Power over Ethernet (PoE) ce qui signifie qu'un seul fil Ethernet est requis comme source d'alimentation et pour le transfert de données vers un ordinateur.

Le champ de vision de l'appareil, dont les dimensions sont détaillées à la figure 4.5, permet un espace de travail propice à l'étalonnage en robotique.

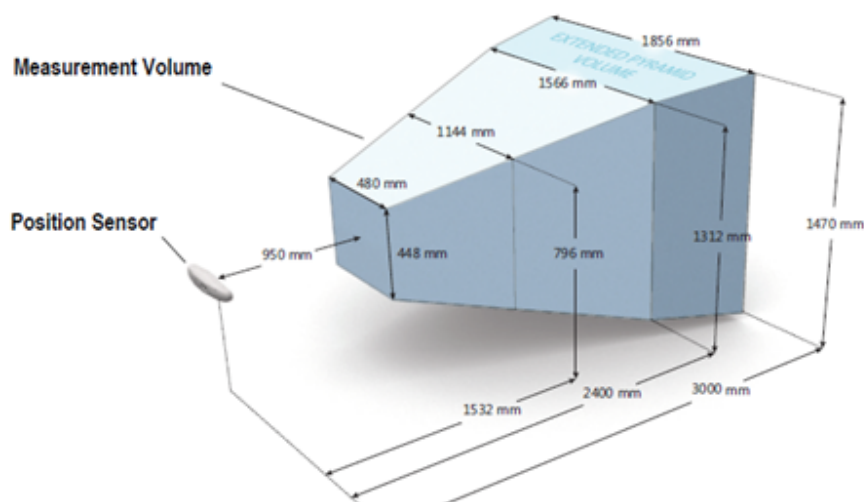


Figure 4.5 Espace de travail du Polaris Vega XT, tiré de [2]

Il est important de noter que le Polaris Vega ne permet pas d'obtenir de résultats absolus. En

effet, le centre optique à partir duquel les données sont prises est ‘flottant’ dû à un mécanisme d’amortissement, par conséquent, il n’est pas possible d’obtenir de positions exactes des cibles par rapport à l’appareil de mesure. Les données obtenues, au même titre que l’étalonnage par axe, seront relatives [2, 44].

Le Polaris Vega a déjà été utilisé dans l’étalonnage robotique, néanmoins, tel que mentionné précédemment, l’utilisation était davantage destinée à des applications médicales telles que le robot chirurgical Da Vinci. La consultation de travaux préalables permet de prévenir les possibles complications et d’identifier les meilleures pratiques tout en fournissant davantage d’informations sur l’utilisation de l’appareil de mesure dans un contexte d’étalonnage.

Les articles étudiés recommandent premièrement d’effectuer une moyenne de plusieurs mesures lors de l’obtention des positions afin de réduire le bruit, l’appareil étant sensible aux vibrations [45]. L’erreur est plus élevée sur l’axe Z, qui est parallèle à l’action de la gravité. Une étude de la répartition des erreurs montre que l’appareil perd en précision plus les mesures sont prises vers la pyramide étendue, plus particulièrement sur les coins de la pyramide [46, 47].

Comme il s’agit d’un capteur optique, il est important de s’assurer que les cibles n’aient aucune occlusions par rapport aux illuminateurs et aux capteurs. Il est également recommandé de couvrir toute surface réfléchissante dans la cellule de travail autre que les cibles [48]. L’expérimental qui suit tente d’établir une méthodologie d’étalonnage complète avec seul le Polaris Vega XT, car ce système permet de prendre des mesures en continu tout en étant beaucoup moins cher que d’autres appareils de mesure optiques normalement utilisés en calibration. Un second appareil de mesure est tout de même introduit dans le but de valider la précision des données obtenues par le Polaris.

### 4.3.3 Leica LTD800

Afin de valider les résultats obtenus par le Polaris Vega XT, le modèle LTD800 de la gamme d’appareil optique de marque Leica a été utilisé comme référence (*Gold Standard*). Effectivement, cet appareil de mesure offre des performances bien supérieures à celles du Polaris, soit une répétabilité de 0.01 mm pour le Leica et 0.12 mm pour le produit de NDI. L’ordre de prix passe de quelques centaines de milliers de dollars canadien pour un appareil Leica à quelque dizaines de milliers pour le Polaris. L’objectif de l’utilisation du Leica dans cette démarche est de valider les performances du Polaris Vega XT afin d’offrir une solution qui

ne nécessite que le Polaris.

#### 4.4 Conception des outils

Pour permettre la démarche expérimentale, deux outils ont été conçus pour le support des marqueurs métrologique, un pour le robot maître et un pour le robot esclave. Chaque outil se compose de deux faces inclinées comportant onze marqueurs, trois pour le système Leica et huit pour le système Polaris. L'outil maître est illustré à la figure 4.6.

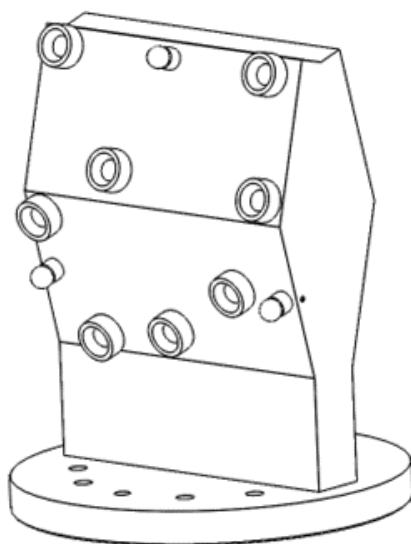


Figure 4.6 Outil du robot maître

Afin de réduire le poids et le coût des outils, il a été déterminé que la longueur de l'outil ne devait pas excéder 240 mm, ce qui s'apparente à l'ordre de grandeur de l'outil développé par DLR au moment de la démarche expérimentale. Pour ce qui est de la largeur des outils, ce paramètre a été limité au diamètre des brides, soient 200 mm pour en faciliter l'usinage.

La position des deux faces a été déterminée de manière à ce que la bride ne bloque pas le champ de vision des appareils de mesure lorsque la partie inférieure de l'outil est inclinée. Les faces sont inclinées à  $15^\circ$  également afin d'augmenter la visibilité des cibles. Une symétrie des deux cotés de l'outil a été réalisée pour pouvoir réutiliser l'outil avec soit d'autres cibles ou avec des motifs différents.

Étant donné le contexte expérimental et investigateur de la démarche, il n'a pas été possible

de déterminer un cahier des charges précis pour la conception des outils étant très limité par la disponibilité du matériel et des matériaux ainsi que du personnel qualifié pour l'usinage.

Pour des raisons économiques, tout devait être fabriqué à l'interne, ce qui donnait accès à une CNC trois axes, une CNC cinq axes, une machine de coupe au jet d'eau ainsi qu'une imprimante 3D. Également, il fallait être attentif au délai de livraison des pièces et/ou matériau commandé, compte tenu de la durée du projet, il n'aurait pas été possibles d'avoir des mois de délais imposés par la commande de matériel.

Le choix du matériau fut effectué sur trois critères : premièrement le matériau devait se conformer à une utilisation d'appareil optique soit une surface opaque et non-réfléchissante ou un matériau permettant le fini de surface ainsi que d'un recouvrement de peinture pour rencontrer ces critères. Ensuite, le matériel devait être assez rigide pour permettre de prendre des mesures en réduisant le bruit dans les données induit par les vibrations des effecteurs. Finalement, le matériel devait être usinable par les machines disponibles et les délais de livraison devaient être raisonnable en fonction de la durée du projet.

En se basant sur ces critères, l'aluminium a été sélectionné comme matériau étant facilement usinable par CNC et facile d'obtention. Il sera également possible d'effectuer une finition de surface *sand blast* ainsi que de peindre le matériau à l'issue de la fabrication afin de le rendre opaque et propre à l'utilisation attendue. L'aluminium a été commandé sous forme de cubes, les effecteurs mesurant 240 par 240 mm, deux blocs de 265 par 265 mm pour permettre 25 mm de matière supplémentaire.

#### 4.4.1 Cibles Optiques – Polaris

Pour détecter la position de l'outil, le Polaris Vega XT requièrent des cibles infrarouges. Chaque outil comporte huit cibles infrarouges, leur positionnement sur l'effecteur maître est illustré à la figure 4.7.

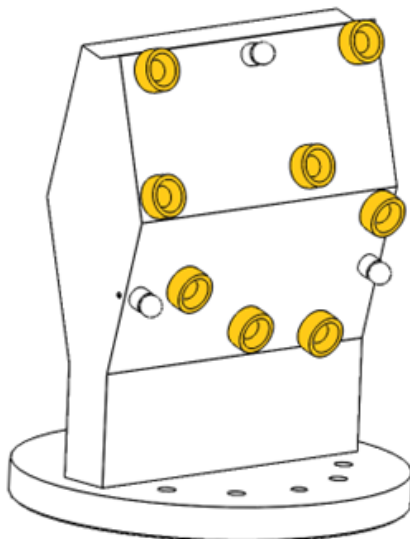


Figure 4.7 Position des cibles du Polaris sur l'outil maître

NDI offre trois types de cibles pour ses appareils optiques : des cibles passives sous forme de disques ou de sphères ainsi que des cibles actives avec ou sans fil. Cependant, le modèle Vega XT prend seulement en charge les cibles passives et les cibles avec fils.

Compte tenu que les cibles seront montées sur des manipulateurs, l'utilisation de cibles actives avec fil auraient été complexe à gérer. Pour l'application proposée, NDI recommande l'utilisation de cibles passives, plus particulièrement sous forme de disques de diamètre de 11 mm ou plus. Ces marqueurs sont souvent présents sous forme adhésive à usage unique. Néanmoins pour l'application présente, les cibles doivent être placées avec précision. C'est pour cela qu'il a également été recommandé d'utiliser des cibles cylindriques montées sur des arbres afin de pouvoir connaître leur position avec l'usinage précis de la CNC.

La technologie requière trois à six cibles pour fonctionner et NDI recommande d'en utiliser au moins quatre. Dans l'optique de conserver une solution économique, chaque face a été munie d'un total de quatre cibles.

Afin d'assurer une précision conforme aux spécifications fournies par le fabricant, la distance minimale entre les cibles doit être de 40 mm dans la pyramide du Polaris Vega et 50 mm dans la pyramide étendue. Par conséquent, la distance entre les marqueurs doit être d'au minimum 50 mm en plus de la tolérance d'usinage des cibles.

Pour que l'appareil puisse distinguer les différents outils, la géométrie du placement des cibles doit être unique sur chacune des pièces. La différence minimum requise entre les segments est de 3.5 mm et la différence minimum requise entre une paire de segment est de 2.0°. Par conséquent, le placement des cibles d'un effecteur ne peut pas être le miroir de l'autre car il s'agit d'une géométrie similaire. La géométrie des deux outils est donc très différente. Puis, selon la documentation du Polaris, pour que les positions des cibles soient directement utilisées dans le logiciel de localisation, la tolérance de placement du centre mécanique de la cible ne doit pas excéder 0.05 mm. Également, afin de localiser les cibles dans le repère du Leica, les positions des cibles doivent permettre le passage d'un disque de 25 mm de diamètre autour des cibles.

Les deux faces sur lesquelles se trouvent les cibles ont une inclinaison de 15° afin de maximiser la visibilité de ces dernières. En effet, pour maximiser la précision du Polaris, les marqueurs doivent se trouver à un angle maximum de 60° comme montré à la figure 4.8. Pour fonctionner, seulement une des deux faces doit être visible par l'appareil de mesure. L'inclinaison des faces permet donc d'augmenter le champ d'action de 60° à 90° [3].

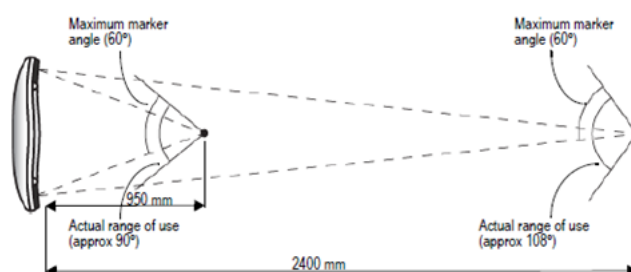


Figure 4.8 Visibilité des cibles du Polaris, tiré de [3]

Basé sur la géométrie des outils et les requis de positionnement listé ci-dessus ainsi que la position des cibles LEICA et la taille de la cible mobile, les motifs suivants illustrés à la figure 4.9 ont été déterminés pour chacune des deux faces pour les deux effecteurs.

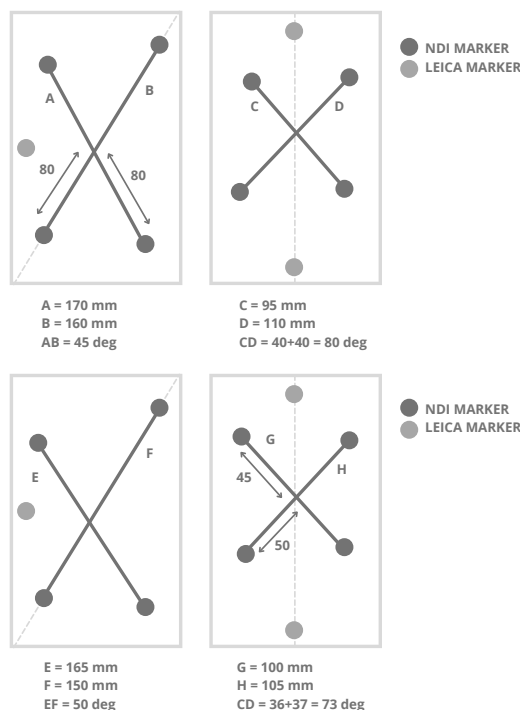


Figure 4.9 Positions des cibles du Polaris sur les outils

#### 4.4.2 Cibles optiques – LEICA

Dans le but de corroborer les positions obtenues par le Polaris Vega XT, des cibles compatibles avec l'appareil de mesure Leica seront également ajoutées aux outils. Les cibles Leica sont des cibles circulaires dont le centre creux est indiqué à l'aide de trois miroirs qui ont pour objectif de diriger le rayon laser émis par l'appareil. Ces sphères sont montées sur des socles magnétiques. Comme il est possible de les orienter sur ces socles, ces cibles n'ont pas les mêmes considérations de champ de vision que pour le Polaris. Chaque outil est muni de trois cibles comme montré à la figure 4.10 sur l'outil maître.

Trois cibles au minimum doivent nécessairement être présentes sur l'outil afin de créer un plan et ainsi obtenir la position ainsi que l'orientation de la bride car l'appareil de mesure donne seulement la position XYZ des cibles.

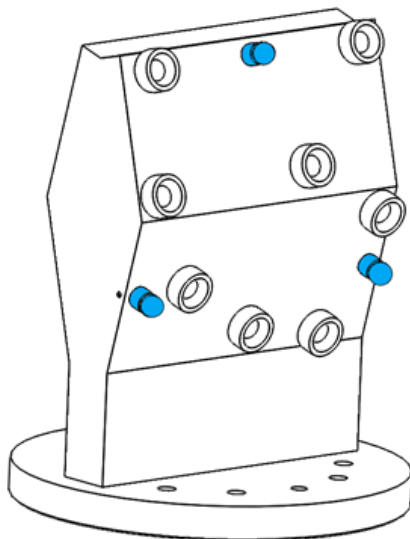


Figure 4.10 Position des cibles du Leica sur l'outil maître

#### 4.5 Processus expérimental

Les objectifs ont été modifiés au cours de la démarche expérimentale lorsque confronté à de nombreux obstacles : tout d'abord, à la suite de circonstances inévitables, il n'a pas été possible d'usiner les outils avant janvier 2022. Pour pallier cet inconvénient, une version en plastique des outils a été imprimée à partir d'impression 3D. L'imprimante 3D disponible étant de petite taille, les outils ont été imprimés en deux morceaux qui ont par la suite été assemblés par des vis. Des tests préliminaires ont été effectués sur ces outils, mais l'option du prototypage rapide a été rapidement écartée car la faible rigidité du plastique introduisait trop de bruit dans les données.

L'objectif initial de la démarche expérimentale était d'utiliser l'appareil de mesure Polaris et de valider les résultats obtenus avec le capteur Leica. La prise de données aurait entièrement été automatisée et prise par le Polaris et un groupe de 100 mesures aurait ensuite été prises avec le Leica manuellement pour évaluer la performance du Polaris. Malheureusement, il n'a pas été possible d'utiliser le Polaris dans les temps. Premièrement, l'obtention de l'appareil de mesure fut très longue à cause de procédures légales et administratives avec NDI. Ensuite, les cibles commandées pour le Polaris Vega sont arrivées endommagées. Les délais de livraison de nouvelles cibles auraient occasionné un délai additionnel de quelques mois, ce qui n'était pas envisageable considérant la durée du projet ainsi que les délais déjà occasionnés. Par

conséquent, l'utilisation du Polaris a été écartée et la totalité des mesures ont été prise à partir du Leica.

Afin d'obtenir des résultats expérimentaux, un millier de poses ont été générées sur le logiciel Robot DK pour le robot maître et le robot esclave. Ces poses doivent en premier lieu respecter la géométrie des manipulateurs, c'est-à-dire que les positions demandées n'engendrent pas de singularités lors des trajectoires. Le logiciel vérifie également que toutes les positions générées se retrouvent dans l'espace de travail du Polaris Vega XT et du Leica LTD 800 ainsi que les trajectoires ne provoquent aucune collision. Pour s'assurer de cette dernière précaution, des poses intermédiaires plus éloignées de l'espace de travail ont été ajoutées entre chaque pose afin que chacun des robots puissent prendre l'orientation requise sans singularités ni collisions.

Une fois que toutes les poses ont été validées par le logiciel, deux scripts sont générés en langage KUKA, un pour chacun des robots. Les manipulateurs sont ensuite directement commandés à partir de la télécommande KUKA. Les robots bougent en simultané vers leurs poses intermédiaires puis vers la pose à mesurer. Les deux robots se mettent ensuite en position d'arrêt et les freins sont enclenchés afin que l'opérateur puisse entrer dans l'espace de travail pour prendre en mesure la position de chacun des socles en déplaçant la cible Leica. Les deux robots sont de calibre industriel et sont situés dans une cellule robotique fermée afin de s'assurer que personne ne se trouve dans l'espace de travail lorsque ces derniers sont en marche. Les robots ne peuvent pas fonctionner si la porte n'est pas fermée. Un opérateur se charge de la commande des robots ainsi que de la prise de mesure et un second opérateur se charge d'ouvrir et fermer la porte de la cellule robotique et de déplacer les cibles Leica. Les robots ne peuvent pas être en mouvement à moins que la porte de la cellule robotique soit fermée et que personne ne se trouve dans l'espace de travail. La position des cibles est enregistrée sur le logiciel PolyWorks à partir d'un ordinateur connecté au Leica.

Avant de procéder à la prise de mesure, une seconde validation de la collision a été effectuée en addition de la vérification faite par simulation sur Robot DK. Cette validation consiste à effectuer chaque configuration lentement en s'assurant qu'il n'y ait aucun accrochage.

Avant chaque séance de mesure, la position du centre des marqueurs de sol doit être étalonnée afin de situer les différents repères dans l'espace. Les marqueurs de sol sont des sphères métalliques fixées aux bases des robots. Pour déterminer leur position, une cible Leica sphérique est glissée sur la surface des marqueurs et le logiciel Polyworks détermine ensuite la position de leur centre. Par géométrie, il est ensuite possible de déterminer avec précision la position exacte du repère des bases illustrée à la figure 4.11.

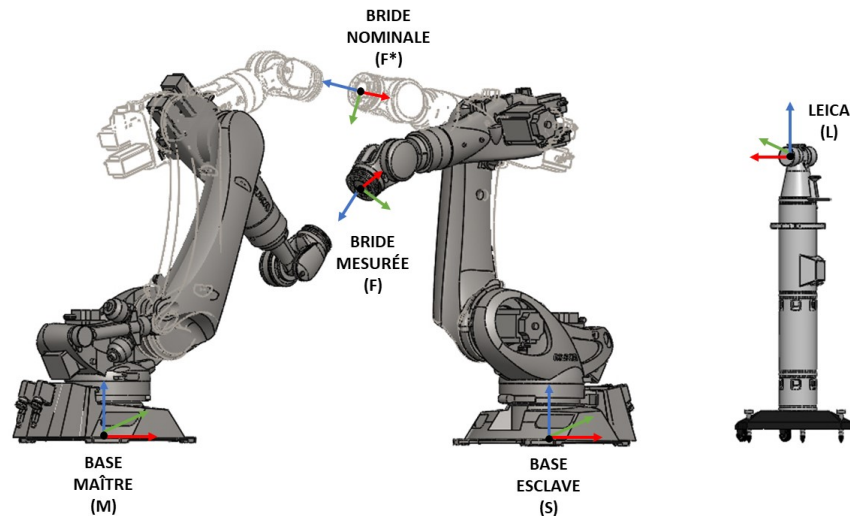


Figure 4.11 Position des repères

La première série de données est mesurée sans étalonnage. Pour chacune des 1000 configurations, la position XYZ de chacune des six cibles est enregistrée, soit trois par outil. Il est à noter qu'il n'aurait pas été nécessaire pour cette première étape d'obtenir 1000 données considérant qu'il ne s'agit pas de données d'entraînement. Cette quantité d'information était requise dans le cadre d'un autre projet. Chaque série de 1000 données prends environ une semaine à effectuer.

Les trois dernières articulations ( $J_4$ ,  $J_5$ ,  $J_6$ ) sont ensuite calibrées à partir de la méthode des axes. Pour la présente cellule, il est seulement nécessaire de calibrer ces trois axes car la base du robot est déjà précisément étalonnée avec les marqueurs de sols. Cette étalonnage est effectuée à partir du capteur Leica et du module de calibration de Robot DK. Une fois les deux robots étalonnés, les positions sont régénérées par le logiciel en appliquant la correction modélisée par l'étalonnage sur les données nominales soit en appliquant une fausse cible (fake target). Les positions intermédiaires ne sont pas modifiées. Deux nouveaux scripts KUKA sont alors de nouveau générés à partir des poses compensées. La deuxième série de données est ensuite mesurée à la suite de l'étalonnage de la même manière que pour la première série de données. Pour conclure, à l'issue des manipulations expérimentales sont obtenues deux séries de 6 000 positions XYZ des cibles Leica dans le repère de l'appareil de mesure, une avec compensation et une sans compensation géométrique. Il est possible de situer ces positions dans le repère de la base du robot maître à partir de la position des marqueurs de sol. Les positions des brides des deux robots sont ensuite exprimées dans le repère maître à partir de la relation géométrique entre les brides et les positions des cibles Leica sur les outils.



Figure 4.12 Prise de mesure avec le capteur LEICA

#### 4.6 Mesures de sécurité

Le processus expérimental étant effectué sur des robots industriels pouvant supporter des charges utiles très élevées, il est primordial de suivre les mesures de sécurité. Les robots doivent être situés dans une cellule fermée et verrouillée. Aucun opérateur ne peut se retrouver dans la cellule lorsque les robots sont en mouvement. La porte de la cellule se déverrouille seulement lorsque les freins d'urgence sont enclenchés. À ce moment, l'opérateur peut rentrer dans la cellule pour effectuer les mesures sur les robots.

## CHAPITRE 5 OPTIMISATION DES HYPERPARAMÈTRES ET RÉSULTATS FINAUX

### 5.1 Données obtenues

Les coordonnées des cibles fixées aux outils ainsi que la position des marqueurs de sol, permettent de déterminer les positions des brides des deux robots pour les deux séries de données, les données compensées et non-compensées.

Les figures 5.1, 5.2 et 5.3 permettent d'illustrer la progression de l'erreur de poursuite à la suite de l'étalonnage par arc.

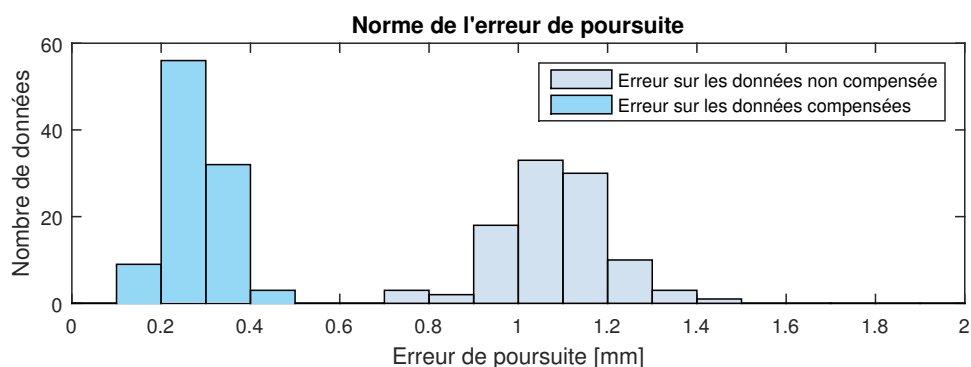


Figure 5.1 Norme euclidienne de l'erreur en XYZ dans le repère de la bride maître

Sur la figure 5.1, 5.2 et 5.3, il est possible de constater une nette amélioration de l'erreur de poursuite. Ce progrès est cependant plus faible en orientation comme indiqué à la figure 5.3, notamment pour l'angle C. Il subsiste une erreur résiduelle à la suite de l'étalonnage géométrique. C'est cette erreur, attribuée aux erreurs non-géométriques et aux erreurs aléatoires qui sera approximée à l'aide d'un réseau de neurones qui prend en entrée les positions des robots maître et esclave.

### 5.2 Optimisation des hyperparamètres

Les données obtenues de manière expérimentales étant très précises, il fut possible d'optimiser le réseau de neurones en ajustant ses paramètres et hyperparamètres, ce qui n'avait pas été

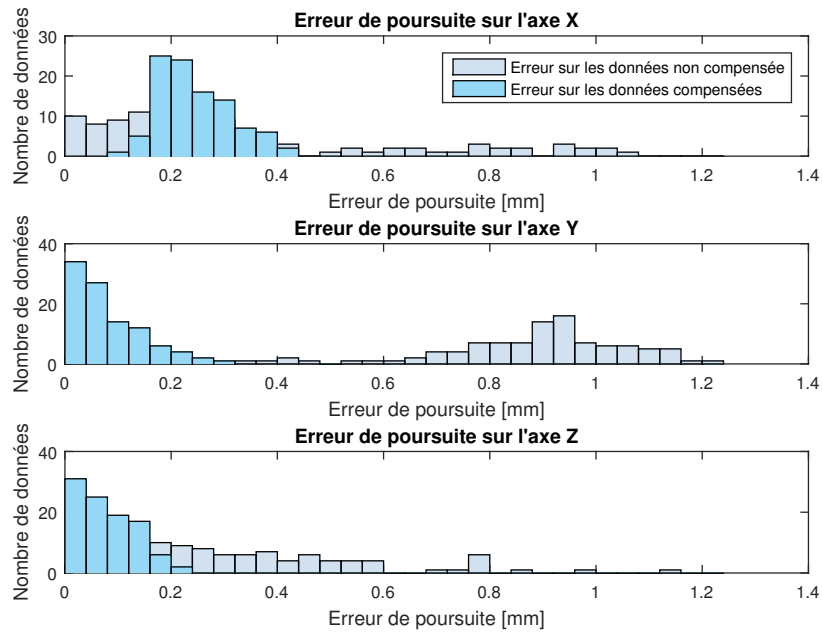


Figure 5.2 Erreur en position XYZ

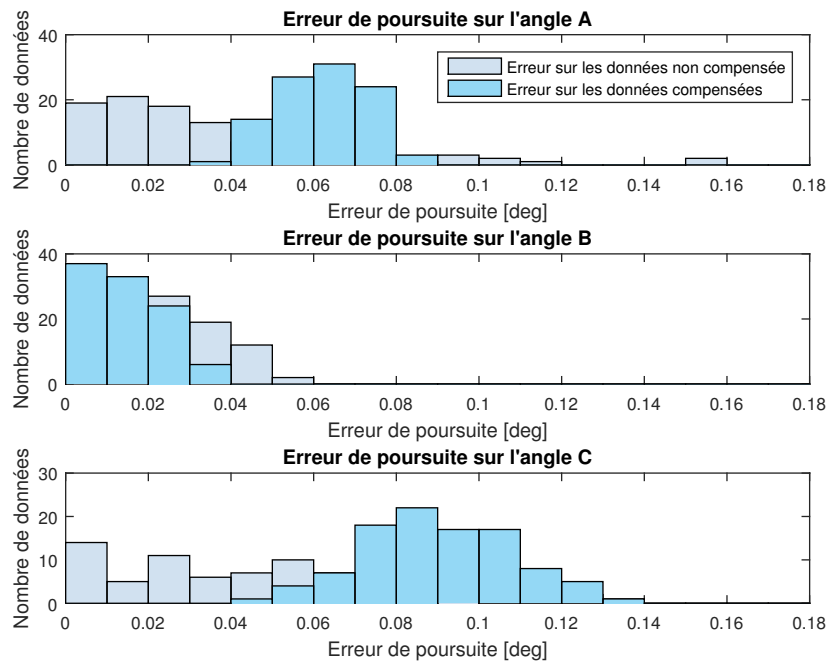


Figure 5.3 Erreur en orientation ABC

possible lors de l'étape de simulation avec les données issues du projet Bombardier.

### 5.2.1 Répartition des pourcentages et normalisation des paramètres d'entrées

Avant de débiter l'entraînement du réseau de neurones, 100 données parmi les 1000 données disponibles sont mises à l'écart pour évaluer les performances du réseau à la fin du processus.

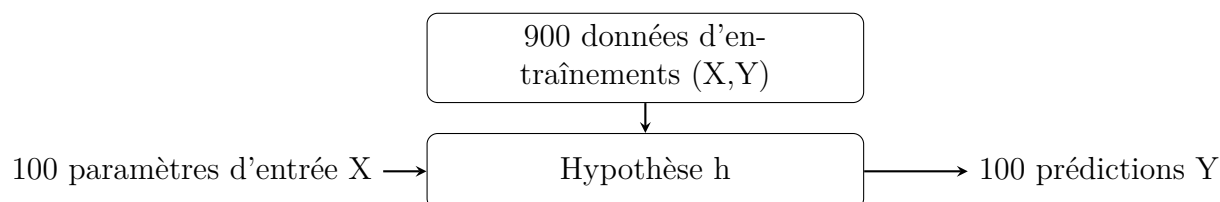


Figure 5.4 Répartition des données d'entraînement et de validation

En effet, 70% de ces 900 données seront utilisées pour l'entraînement lors de l'optimisation. Ensuite, 15% seront utilisées pour la validation de la convergence et 15% pour l'évaluation de la performance. La valeur des pourcentages est une séparation standard. Une fois les hyperparamètres ajustés, il sera possible de déterminer à partir de quel nombre d'itérations l'entraînement du réseau de neurones cesse de converger. Il sera alors possible d'effectuer un entraînement sur la totalité des données d'entraînements puisqu'il sera déjà établi que les résultats seront convergent.

Une fois la convergence ainsi que la performance du réseau assurée, il sera alors possible de procéder à un entraînement sur la totalité des 900 données.

Également, tout comme pour les données du projet Bombardier, chaque paramètre d'entrée  $X_i$  sont nivelé par rapport à la valeur maximale  $X_{max}$ , la valeur minimale  $X_{min}$  ainsi que par la valeur moyenne  $\mu_i$ .

Cette méthode permet de normaliser les résultats et de s'assurer que tous les paramètres d'entrée aient le même poids sur l'entraînement des données. Dans le cas présent, il est nécessaire d'effectuer une normalisation car les positions XYZ [mm] et les orientations [deg] ont des ordres de grandeurs très différents. Si les paramètres d'entrées avaient été les positions des articulations des robots, il n'aurait pas été nécessaire d'effectuer de normalisation car les données auraient toutes été du même ordre.

### 5.2.2 Choix du théorème de régression

Le nombre de données disponibles lors de l'étude sur le projet Bombardier étant limité, le choix du théorème fut limité au théorème de Bayes, théorème favorable à une quantité de données restreintes. Comme il ne s'agissait pas d'une limite pour le présent groupe de données, le théorème de descente de gradient avec momentum et taux d'apprentissage variable (*trainidx*) a été sélectionné. L'incrément de la régression  $\partial\theta$  est calculé à partir de l'incrément précédente  $\partial\theta_{prev}$ . La vitesse de la descente de gradient est régulée par un momentum de l'équation  $m_c$  et un taux d'apprentissage variable  $l_r$  :

$$\partial\theta = m_c \cdot \partial\theta_{prev} + l_r \cdot m_c \cdot \frac{\partial\theta_{prev}}{\partial\theta} \quad (5.1)$$

À chaque itération, si la performance diminue, le taux d'apprentissage est augmenté d'un facteur ( $l_r$ ). Cependant, si la performance augmente d'un certain facteur (*max\_perfinc*), le taux d'apprentissage est ajusté d'un facteur ( $l_{rdec}$ ) et l'incrément qui augmente la performance n'est pas appliqué à la régression.

Il s'agit d'un choix commun pour l'entraînement de réseaux de neurones et ce théorème se porte particulièrement bien au type de données car la variabilité du taux d'apprentissage permet d'augmenter considérablement la vitesse de l'entraînement, contrairement au théorème de Bayes où l'optimisation était limitée par la puissance du matériel fourni.

Il est aussi à noter que ce théorème converge vers un optimum potentiellement local, par conséquent, il n'est pas garanti que l'entraînement obtenu soit optimal pour l'ensemble de données fournies. Il s'agit d'une problématique récurrente lors de l'utilisation de théorème de régression. Pour pallier cette issue, lors de l'optimisation des hyperparamètres, plusieurs itérations de chaque configuration du réseau de neurones seront calculées afin d'établir une erreur moyenne et une déviation standard et ainsi écarter les réseaux dont les performances seraient trop différentes des données moyennes.

### 5.2.3 Détermination des paramètres d'entrées et de l'architecture du réseau

Les paramètres d'entrées utilisés dans la littérature sont variables. Certains articles utilisent les positions des articulations tandis que d'autres utilisent des positions cartésiennes. Prenant en compte le contexte de l'étalonnage, il serait sensé d'utiliser les mêmes paramètres d'entrées

que les paramètres de sortie, soit des positions cartésiennes dans le repère de la base du robot maître. Cependant, comme la régression des réseaux de neurones s'effectuent sur plusieurs plans, on ne peut pas l'apparenter à une régression de modèle physique conventionnelle, c'est-à-dire qu'il n'est pas nécessairement possible d'expliquer comment les données impactent le modèle. Par conséquent, plusieurs combinaisons possibles de paramètres d'entrées ont été comparées en mesurant l'erreur moyenne carrée *Mean Square Error* (MSE) à partir de la fonction de perte.

### Combinaisons de paramètres d'entrée étudiés :

- **XYZABC** : Coordonnées cartésienne et orientation de la bride maître dans le repère de la base du maître  $(X_M, Y_M, Z_M)$  [mm]  $(A_M, B_M, C_M)$  [deg] et coordonnées cartésienne et orientation de la bride esclave dans le repère de la base du maître  $(X_S, Y_S, Z_S)$  [mm]  $(A_S, B_S, C_S)$  [deg]
- **XYZ** : Coordonnées cartésienne de la bride maître dans le repère de la base du maître  $(X_M, Y_M, Z_M)$  [mm] et coordonnées cartésienne de la bride esclave dans le repère de la base du maître  $(X_S, Y_S, Z_S)$  [mm]
- **ABC** : Orientation de la bride maître dans le repère de la base du maître  $(A_M, B_M, C_M)$  [deg] et orientation de la bride esclave dans le repère de la base du maître  $(A_S, B_S, C_S)$  [deg]
- **XYZ J1J2J3** : Coordonnées cartésienne de la bride maître dans le repère de la base du maître et position des trois premières articulations du robot maître  $(X_M, Y_M, Z_M)$  [mm]  $(\theta_{1M}, \theta_{2M}, \theta_{3M})$ [deg] et coordonnées cartésienne de la bride esclave dans le repère de la base du maître et position des trois premières articulations du robot esclave  $(X_S, Y_S, Z_S)$  [mm]  $(\theta_{1S}, \theta_{2S}, \theta_{3S})$ [deg]
- **ABC J1J2J3** : Orientation de la bride maître dans le repère de la base du maître et position des trois premières articulations du robot maître  $(A_M, B_M, C_M, \theta_{1M}, \theta_{2M}, \theta_{3M})$  [deg] et orientation de la bride esclave dans le repère de la base du maître et position des trois premières articulations du robot esclave  $(A_S, B_S, C_S, \theta_{1S}, \theta_{2S}, \theta_{3S})$  [deg]
- **XYZ J4J5J6** : Coordonnées cartésienne de la bride maître dans le repère de la base du maître et position des trois dernières articulations du robot maître  $(X_M, Y_M, Z_M)$  [mm]  $(\theta_{4M}, \theta_{5M}, \theta_{6M})$ [deg] et coordonnées cartésienne de la bride esclave dans le repère de la base du maître et position des trois dernières articulations du robot esclave  $(X_S, Y_S, Z_S)$  [mm]  $(\theta_{4S}, \theta_{5S}, \theta_{6S})$ [deg]
- **ABC J4J5J6** : Orientation de la bride maître dans le repère de la base du maître et position des trois dernières articulations du robot maître  $(A_M, B_M, C_M, \theta_{4M}, \theta_{5M}, \theta_{6M})$

- [deg] et orientation de la bride esclave dans le repère de la base du maître et position des trois dernières articulations du robot esclave ( $A_S, B_S, C_S, \theta_{4S}, \theta_{5S}, \theta_{6S}$ ) [deg]
- **J1J2J3J4J5J6** : Position des articulations du robot maître ( $\theta_{1M}, \theta_{2M}, \theta_{3M}, \theta_{4M}, \theta_{5M}, \theta_{6M}$ ) [deg] et position des articulations du robot esclave ( $\theta_{1S}, \theta_{2S}, \theta_{3S}, \theta_{4S}, \theta_{5S}, \theta_{6S}$ ) [deg]

Les résultats moyens sur trois essais de l'erreur moyenne *Mean Square Error* (MSE) de la fonction de perte en fonction des paramètres d'entrée sont illustrés à la figure 5.5.

Tableau 5.1 Paramètres d'entraînement - Paramètres d'entrée

Paramètres d'entraînement	
Paramètres d'entrées	À déterminer
Paramètres de sortie	$\delta_X, \delta_Y, \delta_Z, \delta_A, \delta_B, \delta_C$
Pourcentage d'entraînement	70%
Pourcentage de validation	15%
Pourcentage de test	15%
Architecture	À déterminer
Nombre d'itérations	5000
Objectif de performance	0.001
Nombre de validation	6
Théorème de régression	<i>trainingdx</i>

La figure 5.5 montre que le choix d'une combinaison des coordonnées cartésiennes et de l'orientation des deux brides dans le repère de la base maître (XYZABC) comme paramètres d'entrée offrent de meilleures performances. L'utilisation des positions des articulations en entrée offrent également de bonnes performances, quoi que légèrement inférieures à la combinaison choisie. L'utilisation de données cartésiennes, en plus d'avoir l'erreur la plus faible est également le format de données qui s'intègre le plus facilement à la structure de contrôle actuellement utilisée.

Les données de la figure 5.5 sont les meilleurs résultats obtenus pour la meilleure architecture pour chaque paramètre d'entrée. L'architecture est composée du nombre de couche du réseau de neurones et du nombre de neurones dans chacune d'entre elle. Une fois de plus, le choix

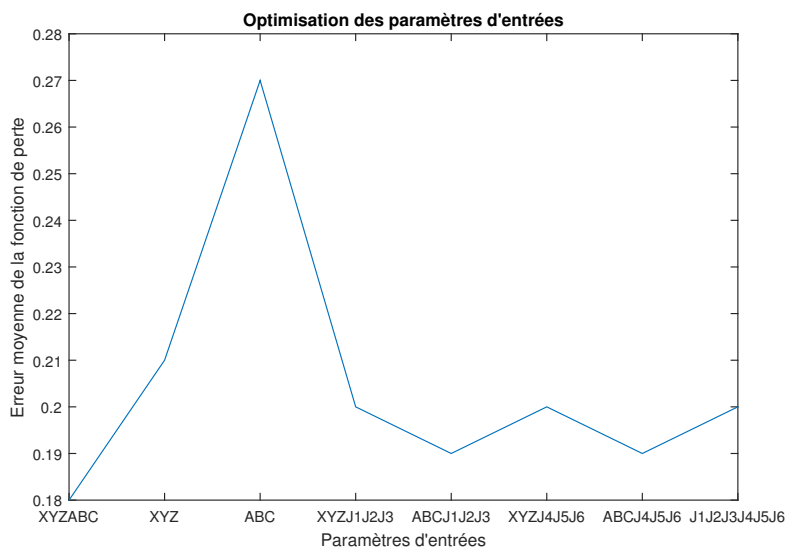


Figure 5.5 Optimisation des paramètres d'entrées du réseau de neurone

est effectué par itération du nombre de couches (1, 2 ou 3) et de neurones dans chacune des couches (2, 4, 8, 16, 32, 64 et 128). La performance de chaque combinaison est ensuite évaluée à l'aide de l'erreur moyenne *Mean Square Error* (MSE) de la fonction de coût. Une étude de l'architecture a été effectuée par itérations sur chaque paramètre d'entrée suggéré. Le détail des résultats obtenus est présent à l'annexe 1, seuls les meilleurs résultats sont représentés sur la figure 5.5.

Pour ces entraînements, le taux d'apprentissage n'est pas fixé comme le théorème choisi modifie la valeur au courant de la régression. L'objectif de performance est fixé à 0,001 ; le nombre de validation à 6 et le nombre d'itérations à 5 000. Il s'agit de valeurs standards.

Les résultats moyens sur trois essais de l'erreur moyenne *Mean Square Error* (MSE) de la fonction de perte en fonction de l'architecture sont illustrés à la figure 5.6.

Les performances obtenues sont très similaires pour les couches comportant 8 et 16 neurones, le choix s'est porté sur une architecture de trois couches comportant 16 neurones car la déviation standard était la plus faible pour une erreur moyenne similaire aux autres combinaisons.

Il aurait également été possible de d'étudier l'effet d'ajouter les polynomiales des paramètres d'entrée ( $X$ ,  $X_2$ ,  $X_3$ , ...). Cependant, des tests préliminaires ont montrés que l'utilisation de polynomiales nuisaient à la convergence du réseau et offraient de moins bons résultats.

Tableau 5.2 Paramètres d'entraînement - Architecture

Paramètres d'entraînement	
Paramètres d'entrées	$X_M, Y_M, Z_M \quad A_M, B_M, C_M \quad X_S, Y_S, Z_S \quad A_S, B_S, C_S$
Paramètres de sortie	$\delta_X, \delta_Y, \delta_Z, \delta_A, \delta_B, \delta_C$
Pourcentage d'entraînement	70%
Pourcentage de validation	15%
Pourcentage de test	15%
Architecture	À déterminer
Nombre d'itérations	5000
Objectif de performance	0.001
Nombre de validation	6
Théorème de régression	<i>trainidx</i>

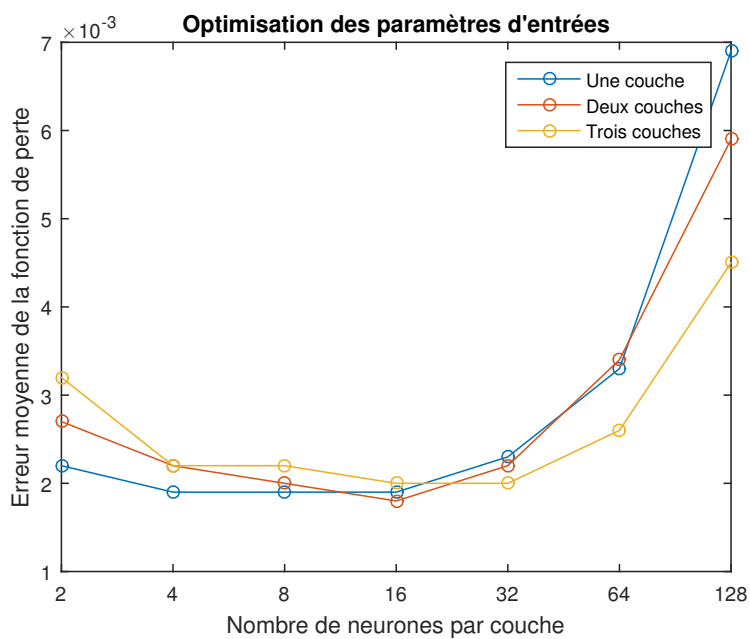


Figure 5.6 Optimisation de l'architecture du réseau de neurones

### 5.2.4 Convergence du réseau de neurone

Une fois l'architecture du réseau définie, il est possible de définir à quel moment de l'entraînement les résultats cessent de converger. L'entraînement est convergent tant et aussi longtemps que le résultat de la fonction de coût diminue. Dès que le résultat commence à augmenter, l'entraînement est considéré comme divergent. Un certain pourcentage des données d'entraînements sont consacrées à cette validation. Une fois que le point de divergence des itérations est identifié, il est possible de passer à un entraînement final avec la totalité des données disponibles.

L'erreur moyenne *Mean Square Error* (MSE) de la fonction de perte en fonction du nombre d'itérations est présenté à la figure 5.7.

Tableau 5.3 Paramètres d'entraînement - Itérations

Paramètres d'entraînement	
Paramètres d'entrées	$X_M, Y_M, Z_M \quad A_M, B_M, C_M \quad X_S, Y_S, Z_S \quad A_S, B_S, C_S$
Paramètres de sortie	$\delta_X, \delta_Y, \delta_Z, \delta_A, \delta_B, \delta_C$
Pourcentage d'entraînement	70%
Pourcentage de validation	15%
Pourcentage de test	15%
Architecture	[16 16]
Nombre d'itérations	À déterminer
Objectif de performance	0.001
Nombre de validation	6
Théorème de régression	<i>traingdx</i>

Plus on augmente le nombre d'itérations, moindre est l'erreur moyenne de la fonction de perte. Cependant, à partir de 200 itérations, comme montré à la figure 5.7, la poursuite de la régression provoque une divergence des résultats et l'entraînement est mis à l'arrêt. Il est donc possible d'entraîner le réseau jusqu'à 200 itérations afin d'augmenter la performance tout en garantissant la convergence des résultats.

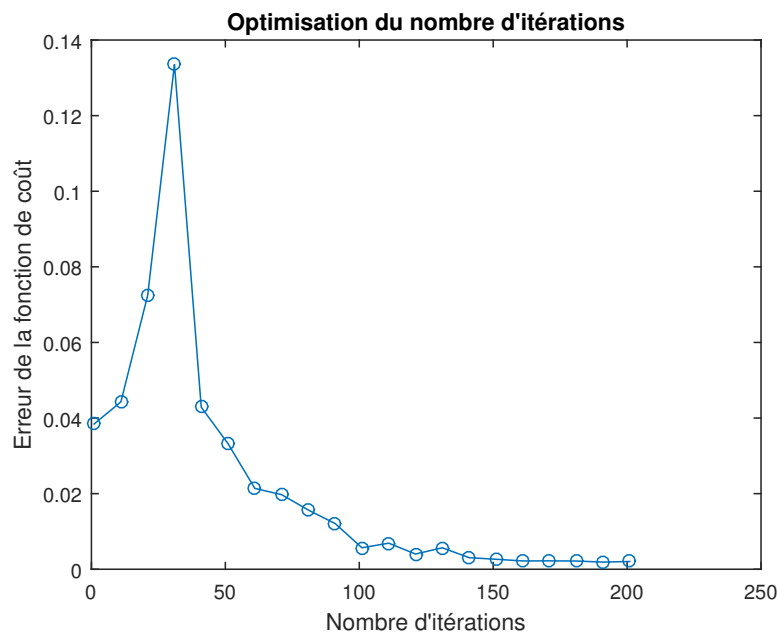


Figure 5.7 Optimisation du nombre d'itérations du réseau de neurones

### 5.3 Résultats finaux

L'entraînement final s'effectue sur la totalité des données d'entraînements. Les comparaisons dans les figures qui suivent sont faites sur un échantillon de 100 données déterminées aléatoirement. Il s'agit de données qui n'ont pas été utilisées lors de l'entraînement, donc le réseau de neurones n'a jamais vu ces informations.

La figure 5.8 illustre la norme euclidienne de l'erreur XYZ dans le repère de la base du robot maître.

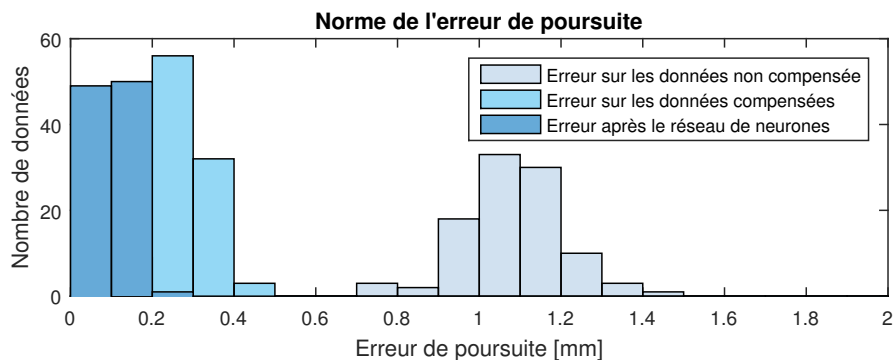


Figure 5.8 Norme euclidienne de l'erreur en XYZ dans le repère de la base du robot maître

Tableau 5.4 Paramètres d'entraînement - Résultats Finaux

Paramètres d'entraînement	
Paramètres d'entrées	$X_M, Y_M, Z_M \quad A_M, B_M, C_M \quad X_S, Y_S, Z_S \quad A_S, B_S, C_S$
Paramètres de sortie	$\delta_X, \delta_Y, \delta_Z, \delta_A, \delta_B, \delta_C$
Pourcentage d'entraînement	100%
Pourcentage de validation	0%
Pourcentage de test	0 %
Architecture	[16 16]
Nombre d'itérations	200
Objectif de performance	0.001
Nombre de validation	6
Théorème de régression	<i>trainidx</i>

Cette première figure illustre l'amélioration apparente de l'erreur à la suite de la calibration, en premier lieu et l'ajustement effectué par l'approximation du réseau de neurones. Les résultats obtenus pour chaque axe et orientation sont illustrés sur les figures 5.9 et 5.10.

Sur ces figures, il est possible de remarquer une nette amélioration de l'erreur à la suite de l'approximation du réseau de neurones. Il est cependant possible d'observer que cette amélioration est plus marquée sur certaines composantes. Le tableau 5.5, comportant l'erreur maximum et l'erreur moyenne de chaque ensemble de donnée permet de commenter davantage sur cette disparité.

Les données présentées au tableau 5.5 montrent une nette amélioration de l'erreur à l'exception de deux paramètres soit l'erreur en Z et l'erreur en B. Pour les deux paramètres, l'erreur maximale augmente de peu, soit 0,004 mm pour l'erreur en Z et 0,002 deg pour l'erreur en B, comparé à une diminution de l'erreur maximale de 0,3 mm pour le l'erreur en X et 0,09 deg pour l'erreur en C. On peut cependant voir sur les figures 5.9 et 5.10 que la répartition de l'erreur s'améliore sur tous les paramètres. Une fois de plus, l'amélioration est plus faible pour l'axe Z, soit une amélioration de 0,01 mm à comparé d'une amélioration moyenne de 0,2 mm pour l'axe X. Il en est de même pour l'orientation B qui dénote une amélioration moyenne de 0.002 deg comparé à 0,07 deg pour l'orientation en C.

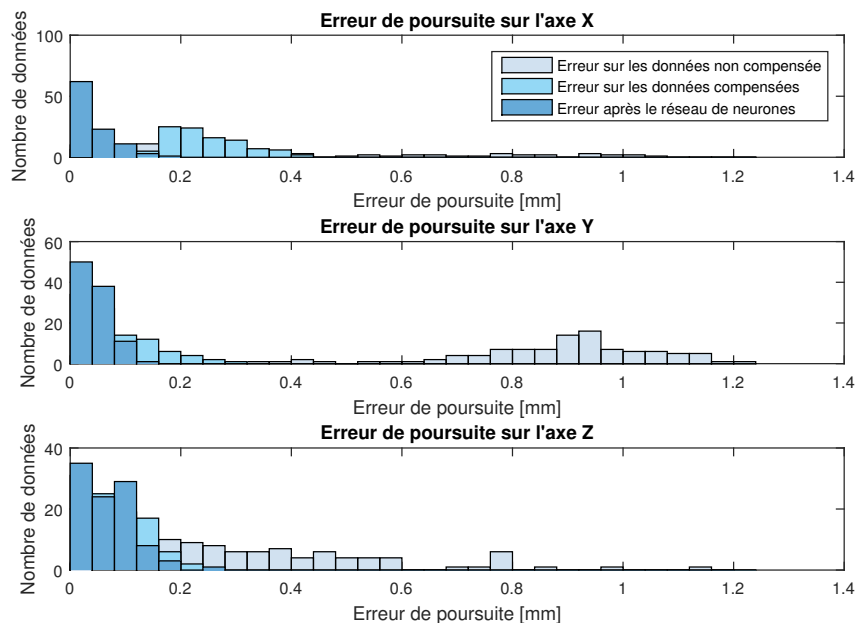


Figure 5.9 Histogramme de l'erreur en position

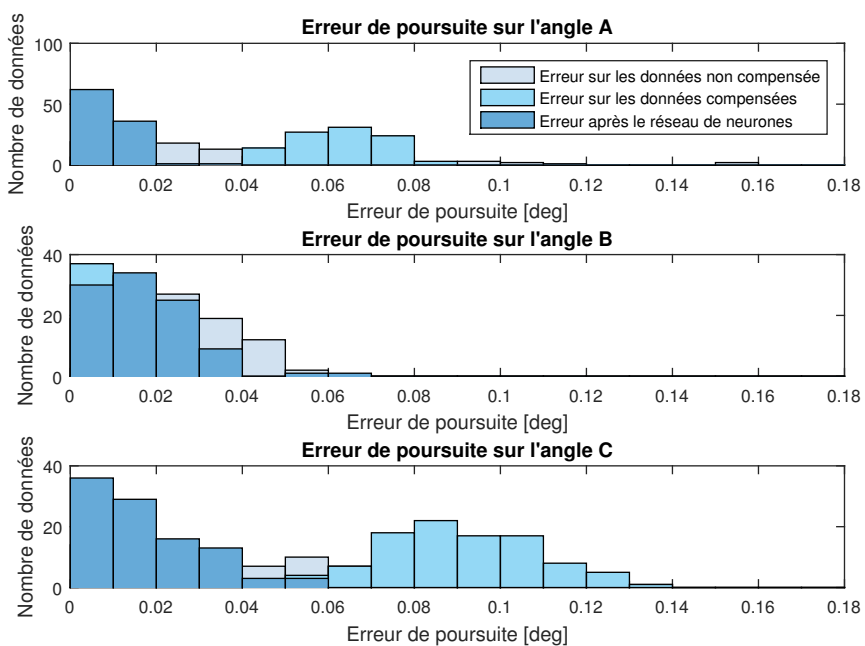


Figure 5.10 Histogramme de l'erreur en orientation ABC

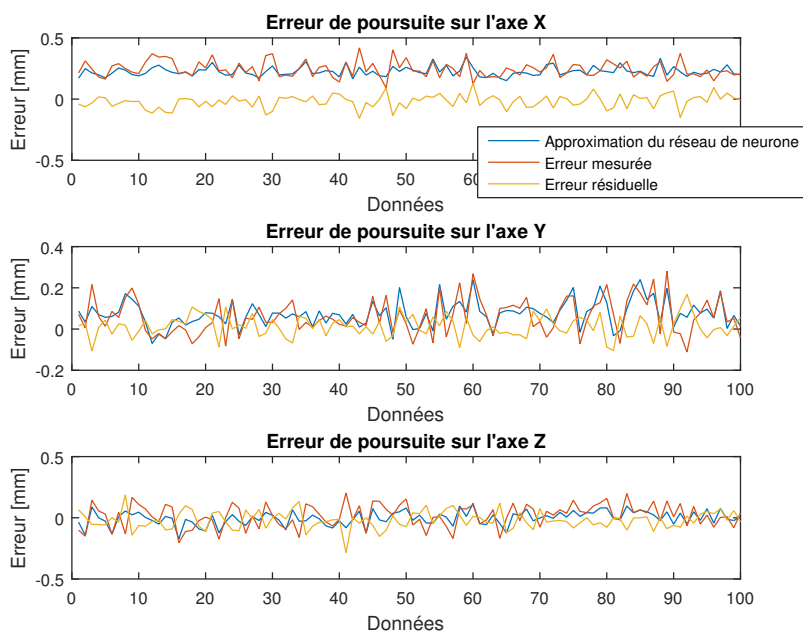


Figure 5.11 Prédiction de l'erreur en position XYZ

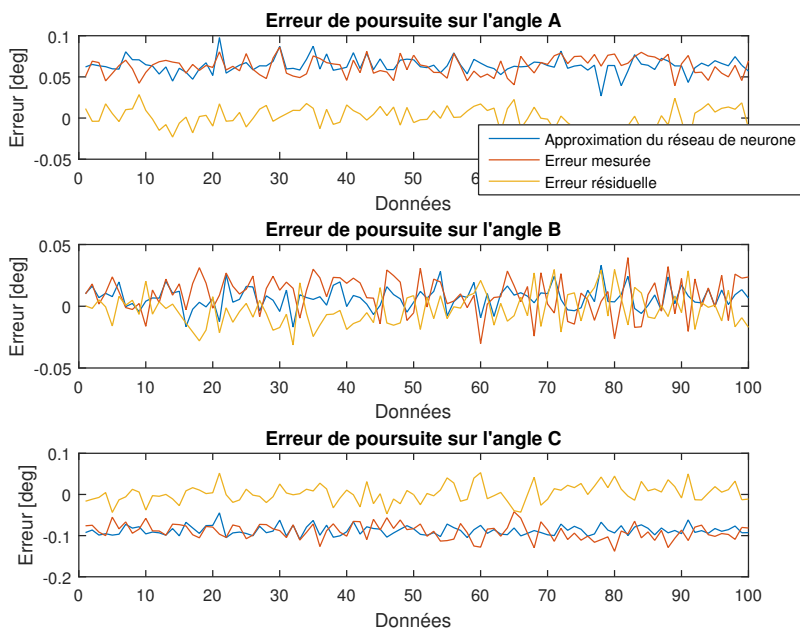


Figure 5.12 Prédiction de l'erreur en orientation ABC

Tableau 5.5 Erreur de poursuite maximale et moyenne pour chaque ensemble de données

Erreur de poursuite maximale pour chaque ensemble de données						
	Position [mm]			Orientation [deg]		
	$\delta_X$	$\delta_Y$	$\delta_Z$	$\delta_A$	$\delta_B$	$\delta_C$
Données non-compensées	1.0485	1.2141	1.1561	0.1575	0.0524	0.1375
Données compensées	0.4188	0.2811	0.2038	0.0862	0.0394	0.1381
Approximation du réseau	0.16667	0.1348	0.2124	0.0272	0.0414	0.045
Erreur de poursuite moyenne pour chaque ensemble de données						
	Position [mm]			Orientation [deg]		
	$\delta_X$	$\delta_Y$	$\delta_Z$	$\delta_A$	$\delta_B$	$\delta_C$
Données non-compensées	0.3322	0.8869	0.3432	0.0340	0.0239	0.0563
Données compensées	0.2440	0.0821	0.0776	0.0627	0.0147	0.0902
Approximation du réseau	0.0438	0.0366	0.0661	0.0114	0.0126	0.0172

L'erreur sur l'axe Z est possiblement causée par le bruit induit par la gravité. Les résultats sur cet axe pourraient être améliorés en utilisant un module de compensation de la gravité et en faisant une moyenne de quelques mesures pour chaque position. L'erreur sur l'orientation B, quant à elle est possiblement causée par l'ordre des données de cette dernière. En effet, les données sont normalisées lors de leur entrée dans le réseau de neurone, mais pas à leur sortie. Il serait possible d'équilibrer cette différence d'ordre en ajoutant des poids sur les valeurs de sortie, néanmoins, l'erreur sur ce paramètre est négligeable même sans compensation, il n'est donc pas nécessaire d'effectuer cette manipulation.

Pour conclure, les résultats obtenus montrent que les prédictions du réseau de neurone permettent de réduire l'erreur de poursuite entre les deux robots et d'améliorer la précision de tâches coopératives.

## CHAPITRE 6 CONCLUSION

### 6.1 Synthèse des travaux

L'objectif de cette démarche scientifique est d'améliorer la précision de robots travaillant en coopération. À la suite de la complétion de l'état de l'art, les travaux se sont portés sur le processus d'étalonnage des deux robots, plus précisément de l'erreur de poursuite entre les deux brides.

Afin d'estimer les erreurs de poursuite et ainsi les compensations à appliquer, il a été décidé de faire usage de réseaux de neurones. Afin de valider cette méthode, des tests préliminaires ont été effectués sur des données disponibles provenant d'un projet d'étalonnage antérieur comprenant deux robots. Les limitations des données disponibles n'a pas permis de statuer sur la performance de l'utilisation de réseau de neurones, mais il a été possible de déterminer que la méthode était convergente. Néanmoins, il fut déterminé nécessaire d'obtenir davantage de données expérimentales pour pouvoir statuer des performances du réseau de neurones.

Des outils ont été conçus afin de supporter des cibles optiques compatibles avec le Polaris Vega XT et le Leica LTD800. Le processus expérimental prévoyait initialement de développer une méthode faisant seulement usage du Polaris Vega XT afin d'obtenir une procédure économique et entièrement automatique qui serait ensuite validée par le Leica LTD800. Il n'a malheureusement pas été possible d'utiliser le Polaris Vega XT et l'entièreté des mesures ont donc été effectuées avec le Leica LTD800. Un total de 1000 combinaisons de poses du robot maître et du robot esclave ont été générés sur le logiciel Robot DK. La position et l'orientation des deux robots ont été mesurées pour l'ensemble de 1000 données une première fois sans étalonner les robots et une seconde fois à la suite d'un étalonnage géométrique.

Les données du second ensemble, soit celui où était appliqué une compensation géométrique, ont été utilisés pour entraîner un réseau de neurones dont l'objectif est d'approximer l'erreur résiduelle de l'étalonnage en prenant pour entrée les valeurs nominales des poses des deux robots.

Les résultats obtenus montrent que les prédictions du réseau de neurone permettent de réduire l'erreur de poursuite entre les deux robots autant en position qu'en orientation.

## 6.2 Limitations de la solution proposée et améliorations futures

Pour conclure, bien que les résultats obtenus soient satisfaisants, la procédure pour leur obtention n'est absolument pas viable. La méthodologie expérimentale faisant usage du capteur Leica nécessite la prise de mesure manuelle de six cibles pour un millier de poses ce qui est très exigeant et peu envisageable dans un contexte autre qu'académique. Dans le futur, il serait primordial d'automatiser le processus expérimental sur l'appareil de mesure Polaris Vega XT, tel que prévu initialement.

Les résultats obtenus par approximation neurones sont adéquats, mais, les valeurs sur l'erreur d'approximation présentées dans ce rapport ont été obtenues par simulation c'est-à-dire qu'il est tenu pour acquis que les compensations dictées par le réseau seraient effectuées parfaitement par les deux robots. Afin de pleinement évaluer la méthode, il faudrait appliquer les corrections aux positions nominales à la fois de la calibration géométrique et de l'approximation faite par le réseau, puis, mesurer à nouveau les poses des deux robots.

Il aurait aussi été pertinent d'évaluer la performance de l'utilisation d'un réseau de neurones par rapport à d'autres méthodes de modélisation afin de déterminer si les bonnes performances sont bien dues à l'utilisation de l'intelligence artificielle et non simplement de part la quantité et la qualité des données disponibles.

Finalement, il serait aussi intéressant de voir si une modélisation par réseau de neurone serait possible dans le cadre d'une compensation appliquée sur des mesures dynamiques.

## RÉFÉRENCES

- [1] Polaris vega xt. [En ligne]. Disponible : <https://www.ndigital.com/optical-measurement-technology/polaris-vega/polaris-vega-xt/>
- [2] *Passive Polaris Vega Firmware Revision History.*
- [3] *Polaris Tool Design Guide.*
- [4] Metrology - flat retroreflector target. [En ligne]. Disponible : <https://www.metrologystore.com/photogrammetry-reflective-tape-retro-dot-spot-size-shank-diameter/reflective-target-reflector-offset-shank-diameter-spot-size-photogrammetry-flat-dot>
- [5] Hubbs machine - laser tracker sphere mount. [En ligne]. Disponible : [https://hubbsmachine.com/laser-tracker-5-sphere-mount-/127-5-sm.html#/material-stainless\\_steel/h\\_to\\_centerline\\_of\\_sphere-\\_5000/a\\_shank\\_diameter-0\\_2500\\_1\\_0000\\_standard](https://hubbsmachine.com/laser-tracker-5-sphere-mount-/127-5-sm.html#/material-stainless_steel/h_to_centerline_of_sphere-_5000/a_shank_diameter-0_2500_1_0000_standard)
- [6] McMaster - steel metric ball-nose spring plunger. [En ligne]. Disponible : <https://www.mcmaster.com/3391A65-3391A651/>
- [7] Kr 300 r2500 ultra f. [En ligne]. Disponible : [https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/0000189044\\_en.pdf?rev=0c33cae61ab74db48270ae3d085f5e52&hash=6CE49AB7467F4C0A13AA9C120BD564B9](https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/0000189044_en.pdf?rev=0c33cae61ab74db48270ae3d085f5e52&hash=6CE49AB7467F4C0A13AA9C120BD564B9)
- [8] A. Quilter, “Composites in aerospace applications.” [En ligne]. Disponible : [https://ihsmarkit.com/pdf/Composites-Aerospace-Applications-whitepaper\\_264558110913046532.pdf](https://ihsmarkit.com/pdf/Composites-Aerospace-Applications-whitepaper_264558110913046532.pdf)
- [9] T. Johnson, “Composite in aerospace,” *Thoughtco*, feb 2019. [En ligne]. Disponible : [https://ihsmarkit.com/pdf/Composites-Aerospace-Applications-whitepaper\\_264558110913046532.pdf](https://ihsmarkit.com/pdf/Composites-Aerospace-Applications-whitepaper_264558110913046532.pdf)
- [10] “Thermoplastic composites, lightweight structures at lower manufacturing costs.” [En ligne]. Disponible : <http://www.collinsaerospace.com/what-we-do/industries/commercial-aviation/aerostructures/advanced-structural-materials/thermoplastic-composites>
- [11] I. Schiel *et al.*, “An investigation of in-situ AFP process parameters using CF/LM-PAEK,” *Advanced Manufacturing : Polymer and Composites Science*, vol. 6, 10 2020.

- [12] L. Zhao *et al.*, “A thermo-viscoelastic analysis of process-induced residual stress in fibre-reinforced polymer–matrix composites,” *Materials Science and Engineering A-structural Materials Properties Microstructure and Processing - MATER SCI ENG A-STRUCT MATER*, vol. 452, p. 483–498, 04 2007.
- [13] M. P et Quinet, *L'étalonnage des robots manipulateurs industriels*, Lavoisier, édit. Hermes Science, 2013.
- [14] C. Smith *et al.*, “Dual arm manipulation—a survey,” *Robotics and Autonomous Systems*, vol. 60, p. 1340–1353, 10 2012.
- [15] A. Rodrguez-Ángeles et V. Parra-Vega, “Adaptive control with impedance of cooperative multi-robot system,” vol. 2, 06 1998, p. 1522 – 1527 vol.2.
- [16] R. Assistant *et al.*, “Sensor-based estimation and control of forces and moments in multiple cooperative robots,” *Journal of Dynamic Systems Measurement and Control- transactions of The Asme - J DYN SYST MEAS CONTR*, vol. 126, 01 2004.
- [17] H. Osumi *et al.*, “Cooperative system for multiple position-controlled robots with free joint mechanisms,” 05 1997, p. 1484 – 1489 vol.2.
- [18] M. Azadi *et al.*, “Inverse dynamics control of two 5 dof cooperating robot manipulators,” vol. 4, 01 2005.
- [19] W. Peng *et al.*, “Double-sided incremental forming : A review,” *Journal of Manufacturing Science and Engineering*, vol. 141, p. 1, 03 2019.
- [20] X. Sheng et X. Zhang, “Fuzzy adaptive hybrid impedance control for mirror milling system,” *Mechatronics*, vol. 53, p. 20–27, 08 2018.
- [21] Z. Shaokun *et al.*, “Real-time thickness compensation in mirror milling based on modified smith predictor and disturbance observer,” *International Journal of Machine Tools and Manufacture*, vol. 144, p. 103427, 06 2019.
- [22] J. Xiao *et al.*, “Research on the collaborative machining method for dual-robot mirror milling,” *The International Journal of Advanced Manufacturing Technology*, vol. 105, 12 2019.
- [23] P. Prat et E. Gueydon, “Cooperative robots for full automation,” 10 2011.
- [24] H. Huan *et al.*, “Dynamic modeling and sensitivity analysis of dual-robot pneumatic riveting system for fuselage panel assembly,” *Industrial Robot : An International Journal*, vol. 43, p. 221–230, 03 2016.
- [25] H.-k. Huang et G. Lin, “Rapid and flexible prototyping through a dual-robot workcell,” *Robotics and Computer-Integrated Manufacturing*, vol. 19, p. 263–272, 06 2003.

- [26] G. Canzhi *et al.*, “Trajectory planning method for improving alignment accuracy of probes for dual-robot air-coupled ultrasonic testing system,” *International Journal of Advanced Robotic Systems*, vol. 16, p. 172988141984271, 03 2019.
- [27] M. Kong *et al.*, “Analysis of kinematic relationship of superposition cooperative dual-industrial robot,” 10 2020, p. 140–143.
- [28] T. Zhang et F. Ouyang, “Offline motion planning and simulation of two-robot welding coordination,” *Frontiers of Mechanical Engineering*, vol. 7, 03 2012.
- [29] Y. Gan et D. Dong, “Robot calibration for cooperative process under typical installation,” *Journal of Applied Mathematics*, vol. 2014, p. 1–12, 04 2014.
- [30] Y. Gan, “Base frame calibration for coordinated industrial robots,” *Robotics and Autonomous Systems*, vol. 59, p. 563–570, 07 2011.
- [31] H. Deng *et al.*, “Base frame calibration for multi-robot coordinated systems,” 12 2015, p. 1489–1494.
- [32] C. Ruan *et al.*, “Base frame calibration for multi-robot cooperative grinding station by binocular vision,” 12 2017, p. 115–120.
- [33] J. Riise *et al.*, “Deployment of ultrasonic through-transmission inspection using twin cooperative industrial robots,” 09 2016.
- [34] J. Wang *et al.*, “Towards simultaneous coordinate calibrations for cooperative multiple robots,” 09 2014, p. –.
- [35] Y. Qin *et al.*, “Simultaneous calibration of the hand-eye, flange-tool and robot-robot relationship in dual-robot collaboration systems,” *Sensors*, vol. 22, p. 1861, 02 2022.
- [36] G. Wang *et al.*, “Simultaneous calibration of multicoordinates for a dual-robot system by solving the  $axb = ycz$  problem,” *IEEE Transactions on Robotics*, vol. 37, 01 2021.
- [37] W. Xu *et al.*, “Complete calibration of industrial robot with limited parameters and neural network,” 12 2016, p. 103–108.
- [38] P.-H. Kuo *et al.*, “Pso and neural network based intelligent posture calibration method for robot arm,” 10 2016, p. 003 095–003 100.
- [39] S. Gadringer *et al.*, “Robot calibration combining kinematic model and neural network for enhanced positioning and orientation accuracy,” *IFAC-PapersOnLine*, vol. 53, p. 8432–8437, 01 2020.
- [40] Z. Wang *et al.*, “A robot calibration method based on joint angle division and an artificial neural network,” *Mathematical Problems in Engineering*, vol. 2019, p. 1–12, 03 2019.

- [41] D. Wang *et al.*, “Robot manipulator calibration using neural network and a camera-based measurement system,” *Transactions of The Institute of Measurement and Control - TRANS INST MEASURE CONTROL*, vol. 32, 01 2010.
- [42] H.-N. Nguyen *et al.*, “A new calibration method for enhancing robot position accuracy by combining a robot model-based identification approach and an artificial neural network-based error compensation technique,” *Advances in Mechanical Engineering*, vol. 11, p. 168781401882293, 01 2019.
- [43] P. N. Le et H.-J. Kang, “Robot manipulator calibration using a model based identification technique and a neural network with the teaching learning-based optimization,” *IEEE Access*, vol. PP, p. 1–1, 06 2020.
- [44] *Polaris Vega User Guide for ST, VT, and XT Models*.
- [45] E. Marinetto *et al.*, “Multicamera optical tracker assessment for computer aided surgery applications,” *IEEE Access*, vol. PP, p. 1–1, 10 2018.
- [46] C. Jun *et al.*, “Using optical tracking for kinematic testing of medical robots,” *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 14, p. e1890, 01 2018.
- [47] A. Wiles *et al.*, “Accuracy assessment and interpretation for optical tracking systems,” *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 5367, 05 2004.
- [48] G. Wang *et al.*, *Instructions for Use of Da Vinci Surgical Robots*, 05 2021, p. 7–89.

## ANNEXE A DÉTAIL DES RÉSULTATS OBTENUS PAR ITÉRATIONS DES PARAMÈTRES D'ENTRÉES ET DE L'ARCHITECTURE

Les tableaux A.1 et A.2 comporte le détail des résultats obtenus par itérations des paramètres d'entrées et de l'architecture. Le tableau A.1 présente la moyenne de l'erreur moyenne carrée de la fonction de perte sur 5 essais pour chaque architecture de chaque paramètre d'entrée et le tableau A.2 présente la déviation standard de l'erreur.

Tableau A.1 Erreur moyenne obtenues par itérations des paramètres d'entrées et de l'architecture sur 5 essais

Erreur moyenne		Nombre de neurones dans chaque couche							
		2	4	8	16	32	64	128	256
XYZ	1	0.0022	0.0019	0.0019	0.0019	0.0023	0.0033	0.0069	0.0143
	2	0.0027	0.0022	0.0020	0.0018	0.0022	0.0034	0.0059	0.0133
ABC	3	0.0032	0.0022	0.0022	0.0020	0.0020	0.0026	0.0045	0.0662
	1	0.0028	0.0021	0.0021	0.0022	0.0027	0.0039	0.0067	0.0163
XYZ	2	0.0043	0.0023	0.0023	0.0022	0.0024	0.0031	0.0049	0.0097
	3	0.0029	0.0029	0.0022	0.0022	0.0023	0.0336	0.0044	0.0087
ABC	1	0.0033	0.0028	0.0027	0.0028	0.0033	0.0049	0.0112	0.0266
	2	0.0037	0.0030	0.0028	0.0027	0.0032	0.0039	0.0078	0.0176
XYZ	3	0.0030	0.0031	0.0030	0.0029	0.0029	0.0730	0.0059	0.0948
	1	0.0023	0.0020	0.0021	0.0021	0.0023	0.0031	0.0050	0.0080
J1J2J3	2	0.0026	0.0022	0.0023	0.0022	0.0023	0.0029	0.0051	0.1561
	3	0.0028	0.0027	0.0022	0.0022	0.0024	0.0029	0.0034	0.0865
ABC	1	0.0021	0.0019	0.0020	0.0019	0.0022	0.0037	0.0068	0.0164
	2	0.0031	0.0023	0.0020	0.0020	0.0022	0.0031	0.0070	0.0148
J1J2J3	3	0.0034	0.0025	0.0021	0.0021	0.0022	0.0288	0.1061	0.0109
	1	0.0021	0.0020	0.0020	0.0020	0.0022	0.0033	0.0052	0.0115
XYZ	2	0.0023	0.0021	0.0021	0.0021	0.0023	0.0032	0.0477	0.0095
	3	0.0033	0.0023	0.0022	0.0021	0.0022	0.0029	0.0043	0.0322
ABC	1	0.0027	0.0023	0.0019	0.0020	0.0024	0.0033	0.0060	0.0135
	2	0.0033	0.0027	0.0025	0.0021	0.0024	0.0032	0.0053	0.0130
J4J5J6	3	0.0035	0.0030	0.0061	0.0024	0.0024	0.0164	0.0687	0.0114
	1	0.0036	0.0021	0.0020	0.0021	0.0024	0.0033	0.0057	0.0106
J1J2J3	2	0.0025	0.0021	0.0022	0.0021	0.0021	0.0032	0.0721	0.0091
	3	0.0031	0.0024	0.0022	0.0021	0.0021	0.0291	0.0040	0.0507

Tableau A.2 Déviation standard de l'erreur obtenue par itérations des paramètres d'entrées et de l'architecture sur 5 essais

		Déviation standard de l'erreur							
		Nombre de neurones dans chaque couche							
		2	4	8	16	32	64	128	256
XYZ	1	0.0001	0.0001	0.0001	0.0001	0.0002	0.0002	0.0002	0.0006
	2	0.0003	0.0001	0.0001	0.0001	0.0001	0.0002	0.0001	0.0025
ABC	3	0.0006	0.0003	0.0001	0.0001	0.0001	0.0003	0.0001	0.0974
	1	0.0010	0.0001	0.0001	0.0001	0.0001	0.0003	0.0008	0.0016
XYZ	2	0.0006	0.0002	0.0002	0.0001	0.0001	0.0001	0.0003	0.0005
	3	0.0006	0.0009	0.0001	0.0001	0.0002	0.0534	0.0004	0.0008
ABC	1	0.0005	0.0001	0.0001	0.0001	0.0001	0.0002	0.0010	0.0016
	2	0.0006	0.0001	0.0003	0.0001	0.0002	0.0002	0.0013	0.0014
XYZ	3	0.0001	0.0002	0.0001	0.0001	0.0002	0.0680	0.0001	0.1428
	1	0.0002	0.0001	0.0001	0.0001	0.0001	0.0003	0.0002	0.0019
J1J2J3	2	0.0001	0.0001	0.0001	0.0001	0.0001	0.0004	0.0003	0.1268
	3	0.0003	0.0005	0.0002	0.0001	0.0001	0.0002	0.0001	0.0689
ABC	1	0.0001	0.0002	0.0001	0.0001	0.0002	0.0004	0.0004	0.0011
	2	0.0007	0.0002	0.0001	0.0001	0.0001	0.0003	0.0008	0.0007
J4J5J6	3	0.0001	0.0005	0.0001	0.0001	0.0001	0.0452	0.0973	0.0018
	1	0.0001	0.0001	0.0001	0.0001	0.0001	0.0002	0.0003	0.0016
XYZ	2	0.0002	0.0001	0.0003	0.0001	0.0002	0.0001	0.0729	0.0005
	3	0.0006	0.0002	0.0001	0.0001	0.0002	0.0003	0.0007	0.0425
ABC	1	0.0002	0.0001	0.0001	0.0001	0.0001	0.0002	0.0006	0.0007
	2	0.0004	0.0003	0.0004	0.0001	0.0002	0.0002	0.0005	0.0009
J4J5J6	3	0.0003	0.0003	0.0061	0.0001	0.0002	0.0244	0.1110	0.0015
	1	0.0022	0.0003	0.0001	0.0001	0.0002	0.0003	0.0002	0.0008
J1J2J3	2	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.1157	0.0003
	3	0.0006	0.0003	0.0003	0.0001	0.0001	0.0458	0.0002	0.0729

## ANNEXE B DÉTAIL DE LA CONCEPTION DE L'OUTIL

Deux effecteurs sont conçus dans le but de supporter les capteurs de métrologie pour le Polaris de NDI. Chaque outil comporte 8 capteurs disposés sur deux faces à  $15^\circ$  de déviation l'une de l'autre.

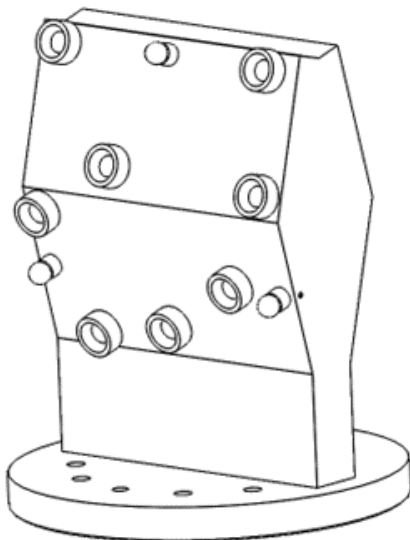


Figure B.1 Outil du robot maître

### Limitation de la disponibilité

- Machines disponibles : CNC 5 axes (Autre équipe)
- Fini disponible : fini à bombardement de sable (*sand blasting*) et peinture
- Matériau disponible : Aluminium (Acier Lachine)

Contacts : Yves Fortin, Daniel O'Keefe, Patrick Turnblom,

### Choix du matériau

- Opaque et non-réfléchissant
- Matériau rigide qui permet un usinage précis
- Un fini de surface permet de réduire la réflexion de la lumière

L'aluminium est commandé sous forme de block. Les dimensions de la pièce doivent être augmentée de 1 pouce supplémentaire. Les deux effecteurs ont une taille de 240×240 mm ou 9.5×9.5 po donc le bloc doit être au minimum de 10.5×10.5 po.

### Capteurs NDI



Figure B.2 Capteurs NDI, tiré de [4]

Pour l'application donnée, NDI recommande l'utilisation de capteurs en forme de disque ayant un diamètre supérieur ou égal à 11 mm. Ces capteurs sont généralement utilisés sous forme d'adhésifs et sont jetés après utilisation. Des disques avec des arbres ont été choisis car il est plus facile de les situer dans l'espace et ces derniers sont réutilisables. Le fournisseur produit un capteur sur mesure pour répondre aux requis.

**Les capteurs sont très fragiles et devrait être inséré par slide-fit et non press-fit.**

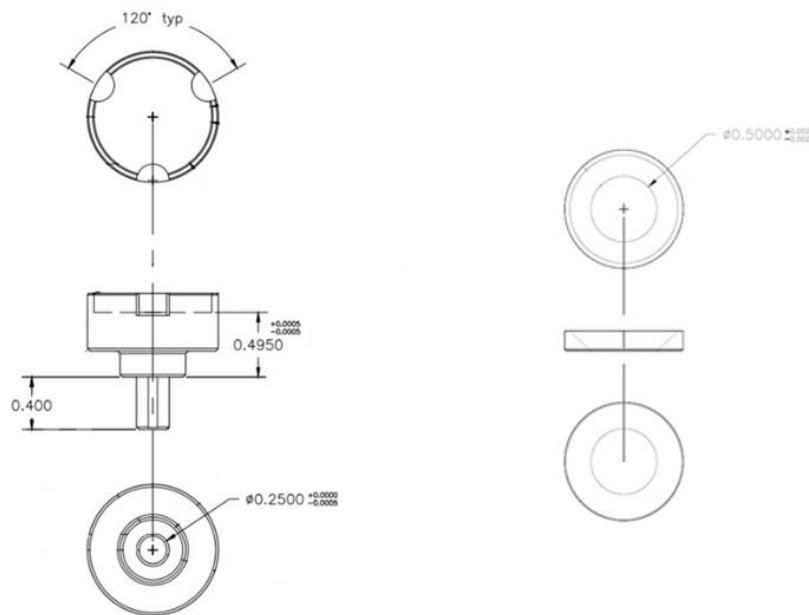


Figure B.3 Capteurs NDI - Dessins Techniques

Fourni par le fournisseur (Il est à noter que toutes les mesures sont en pouces) (Diamètre de 1 po)

Chaque face comporte un total de 4 capteurs. La technologie nécessitent 3 à 6 capteurs pour être fonctionnelle et NDI recommande d'utiliser au moins 4 capteurs. Pour maintenir une solution économique, il est établi que quatre capteurs seront utilisés. Les capteurs doivent être répartis le plus loin possible les un des autres. La distance minimale entre les capteurs est de 40 mm dans la pyramide Vega et 50 mm dans la pyramide Vega étendue. La distance minimale entre les marqueurs est donc d'au moins 50 mm + la tolérance d'usinage.

La géométrie du placement des capteurs doit être assez unique pour que le capteur de position puissent distinguer les faces une de l'autre [3].

- Différence minimale entre les segments : 3.5 mm
- Différence minimale entre les paires similaires : 2.0°
- Les outils qui sont des images miroir les uns des autres ne peuvent pas être utilisés ensemble car leurs géométries sont trop similaires.
- La précision de placement des capteurs ne peut pas excéder 0.05 mm.

## Lexique

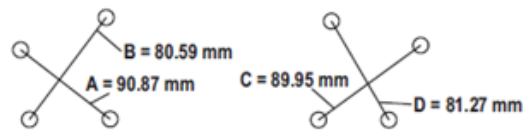
Longueur du segment : La distance entre deux marqueurs sur un outil.

Paire de segments : Deux segments sur le même outil.

Angle des segments : Angle entre deux segments.

Segments similaires : Longueur de segments ayant moins de 3.5 mm de différence.

Paires similaires : Deux paires de segments, chacun sur un outil ou une face différente de manière à ce que chaque segment soit similaire à un segment de l'autre paire. Sur la figure suivante, les segments A et B forment une paire et les segments C et D forment une autre paire. A et C sont des segments similaires ainsi que C et D, donc les paires AB et CD sont des paires similaires.



En se basant sur les requis établis par NDI, les motifs suivants ont été déterminés pour les deux outils :

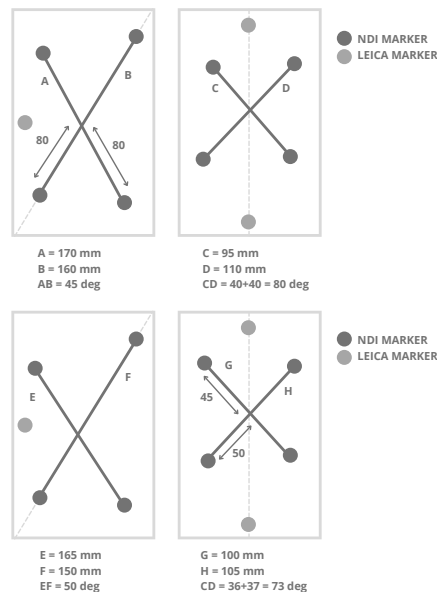


Figure B.4 Positions des cibles du Polaris sur les outils

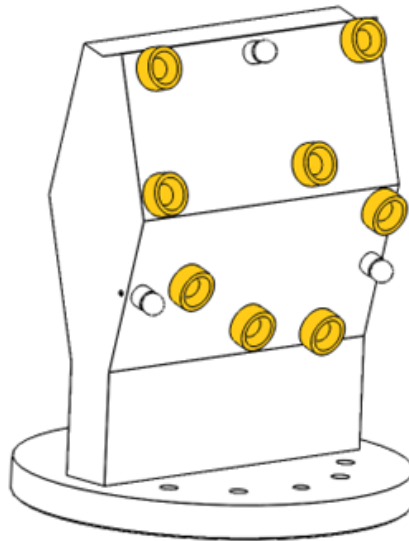


Figure B.5 Capteurs NDI sur l'outil maître

### Capteurs Leica

La validation de la calibration nécessite l'utilisation d'un appareil optique Leica. Trois capteurs additionnel sont ajoutés à chaque outil afin de créer un plan et de localiser les brides dans l'espace en position et orientation.

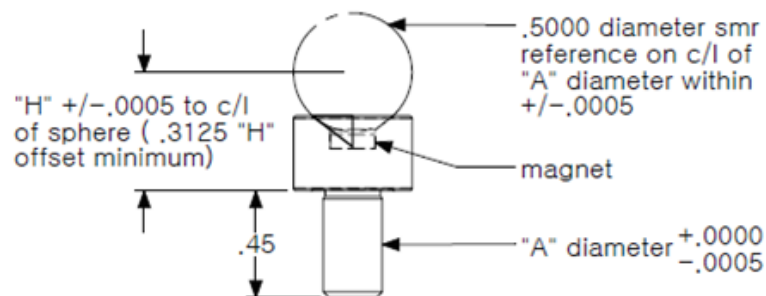


Figure B.6 Dessins techniques des capteurs Leica, tiré de [5]

$$H = 0.5'' \quad A = 0.25''$$

Pour insérer les capteurs, des trous de 0.25 pouce et des vis pressions sont ajoutés aux outils. Les outils sont trempés dans du *locktight* et fixés avec les vis pressions. Les vis recommandées sont les 3408A94 de McMaster mais les vis 3391A65 ont été sélectionnées comme elles sont plus courtes et plus adaptées aux géométries des effecteurs.

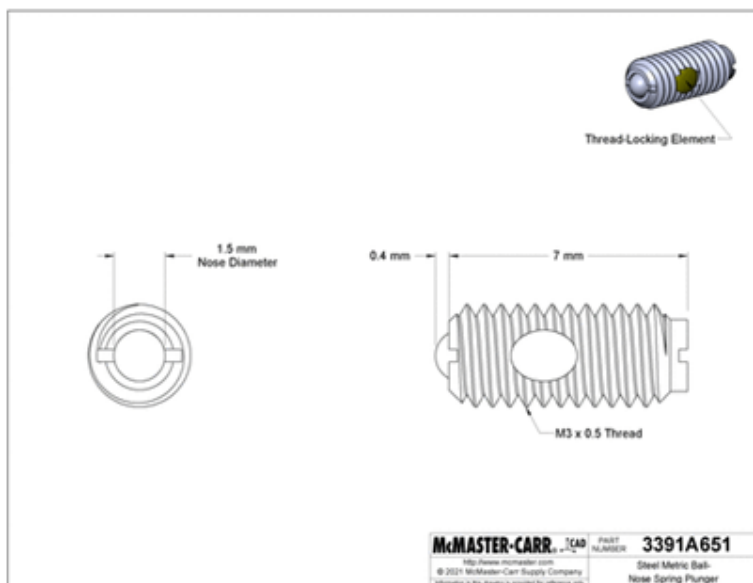


Figure B.7 Vis Pression des capteurs Leica, tiré de [6]

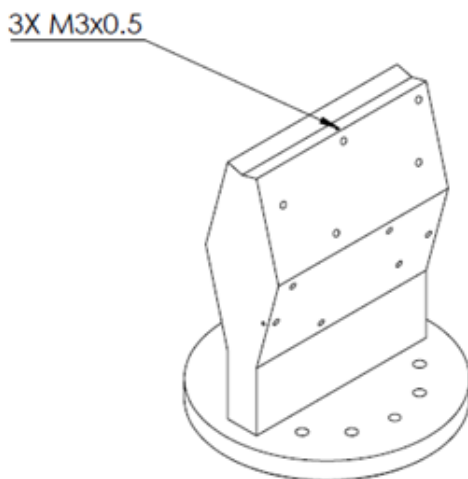


Figure B.8 Fixation des capteurs Leica sur l'effecteur

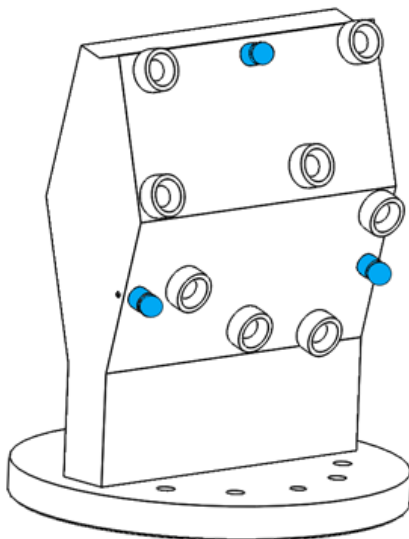


Figure B.9 Capteurs Leica sur l'outil maître

### Connexion avec la bride du robot

Les brides des deux robots sont identiques et comporte 8 alésages M10 et 2 alésages pour le maintien de la plaque *Dowel Pin*.

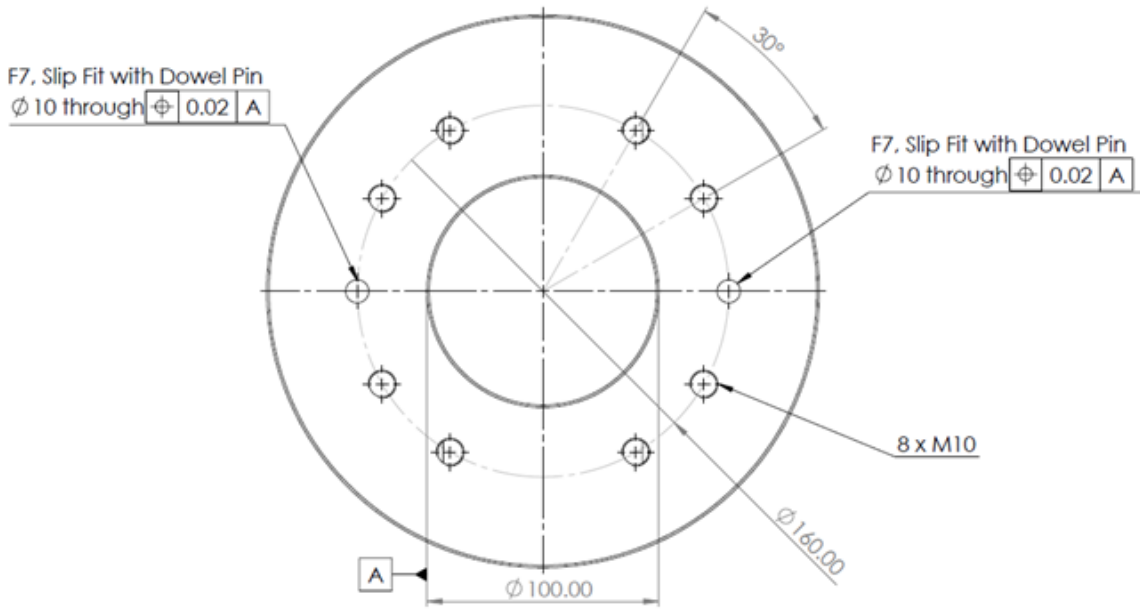


Figure B.10 Dessin technique de la bride 1, fourni par le CNRC

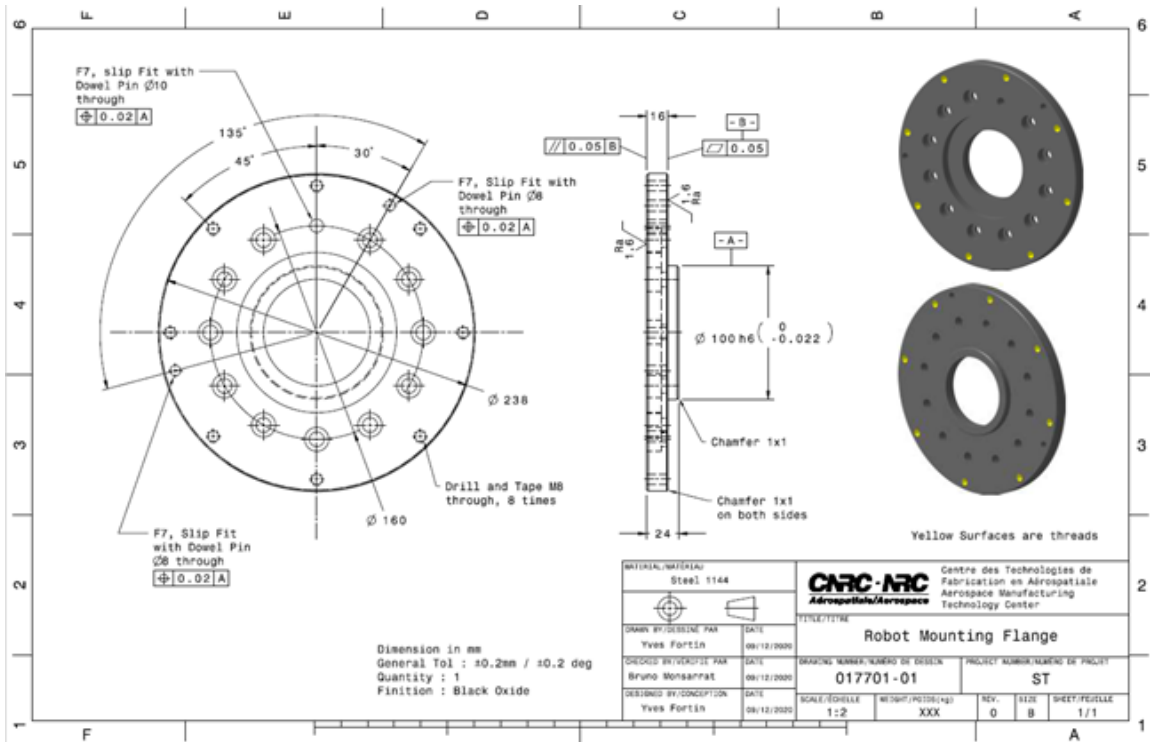


Figure B.11 Dessin technique de la bride 2, fourni par le CNRC

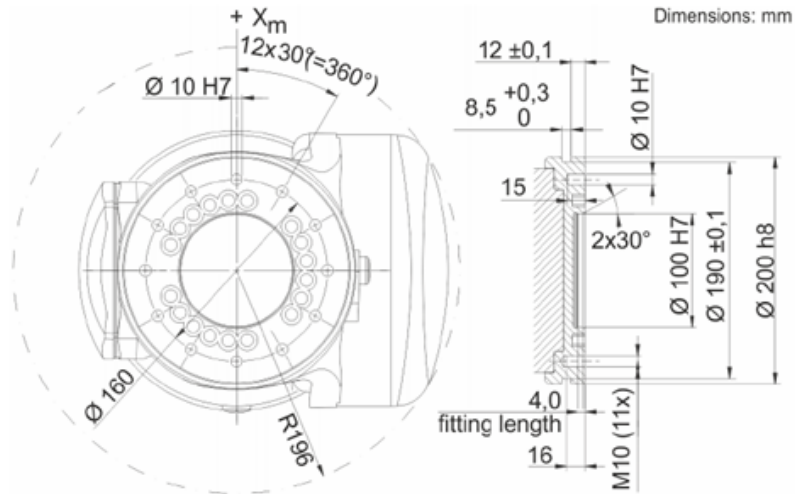


Figure B.12 Dessin technique de la bride 3, [7]

## Tripod

Le Polaris sera fixé sur un tripod Manufroto à haute charge. Une plaque est conçue pour lier l'appareil au tripod.

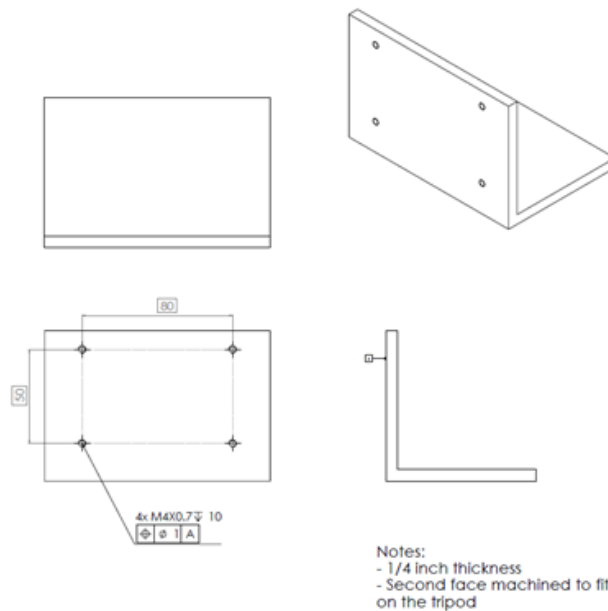


Figure B.13 Dessin technique de la plaque de maintien

## Capteurs

Les alésages n'ont pas besoin d'avoir un fond plat. Les tolérances de placement doivent être en dessous de 0.05 mm.

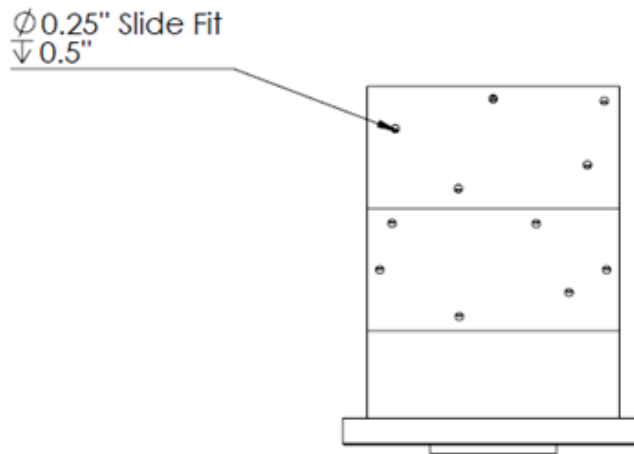


Figure B.14 Placement des capteurs