
Méthodes ensemblistes pour l'étalonnage géométrique

Xavier Baguenard* — **Massa Dao*** — **Luc Jaulin*** — **Wisama Khalil****

*Laboratoire LISA (CNRS - FRE-2656)

Université d'Angers

62 avenue notre Dame du lac

F-49000 Angers

{baguenar, dao}@istia.univ-angers.fr

jaulin@univ-angers.fr

**Laboratoire IRCCyN (CNRS - UMR-6597)

1 rue de la Noe – BP 92101

F-44321 Nantes cedex 03

wisama.khalil@ircyn.ec-nantes.fr

RÉSUMÉ. Dans cet article, une méthode ensembliste utilisant la propagation de contraintes sur les intervalles est proposée pour l'étalonnage d'un robot à 6 degrés de libertés. Cette méthode permet d'enfermer chaque paramètre à estimer à l'intérieur d'un intervalle dont la largeur dépend de l'incertitude sur les mesures collectées, des erreurs de modélisation et du pessimisme inhérent aux méthodes ensemblistes. A travers notre application, nous montrerons que les méthodes intervalle peuvent être utilisées pour traiter des problèmes de très grande dimension (≥ 1000) et fortement non linéaires.

ABSTRACT. This paper presents first results on robot manipulators calibration performed by interval constraints propagation. This method gives an interval domain which confine each parameter. This interval width depends on measure precision, modelling errors and set estimation pessimism. This application shows that it is possible to compute large dimension non-linear problems with interval methods.

MOTS-CLÉS : analyse par intervalles, calcul ensembliste, étalonnage des robots, estimation ensembliste, méthode garantie, propagation de contraintes, robotique.

KEYWORDS: Interval Analysis, Set Computation, Geometric Calibration of Robot, Set Estimation, Reliable Methods, Constraint Propagation.

1. Introduction

Afin de commander correctement les robots, il est fondamental de connaître avec précision leurs paramètres géométriques. Ces paramètres interviennent dans le calcul des modèles géométriques direct et inverse. Le problème de l'étalonnage consiste à estimer ces paramètres géométriques, à partir d'un ensemble de mesures prises sur le robot pour des configurations différentes.

Pour les robots de type série, il est toujours possible d'obtenir un modèle mathématique, dit géométrique direct, qui permet de calculer les coordonnées de l'organe terminal \mathbf{x} en fonction des paramètres géométriques \mathbf{p} et des variables articulaires \mathbf{q} .

Le problème de l'étalonnage des robots de type série est de très grande dimension et est très fortement non linéaire, principalement en raison des fonctions trigonométriques apparaissant dans les contraintes. Le but de cet article est de montrer que sur ce type de problème, les méthodes de propagation de contraintes sur les intervalles sont particulièrement prometteuses.

Dans la section 2, nous présenterons le robot que nous allons utiliser ainsi que deux méthodes pour générer les contraintes associées au problème de l'étalonnage [EVE 89], [HOL 89], [KHA 00] et [KHA 99b]. Une technique de propagation de contraintes sur les intervalles (voir par exemple [VAN 99] et pour une présentation [BLI 01] et [JAU 01]) sera alors introduite en section 3 puis sera appliquée au problème d'étalonnage en section 4.

2. Problème de l'étalonnage géométrique

Le problème de l'étalonnage des paramètres géométriques des robots intervient en robotique afin d'augmenter la précision de positionnement de l'outil. Le modèle géométrique direct [KHA 99a], modèle mathématique associé au robot, permet de connaître les coordonnées de l'organe terminal. Il est calculé à partir de paramètres géométriques. Le problème d'étalonnage revient donc à chercher les meilleures valeurs pour ces paramètres tels que le modèle donne les coordonnées réelles du robot, malgré les erreurs de modélisation et les incertitudes sur les variables.

2.1. Présentation du robot

Le robot Staubli RX90 de la figure 1 est un robot de type série possédant six degrés de liberté d'axes rotoïdes. Une configuration du robot est donnée par le vecteur des coordonnées articulaires :

$$\mathbf{q} = (q_1, \dots, q_6)^t \in \mathbb{R}^6, \quad [1]$$

où les q_i sont les angles des articulations du robot et sont exprimés en radians.

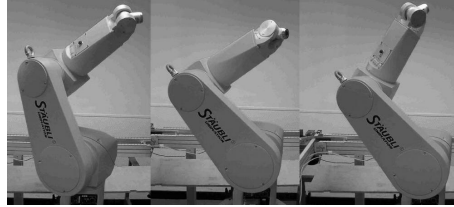


Figure 1. Robot Staubli RX90 dans différentes configurations

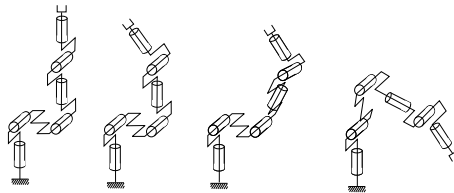


Figure 2. Schéma représentant quatre configurations pour les articulations du robot Staubli

Le schéma de la figure 2 représente le robot dans quatre configurations. Elles correspondent aux vecteurs de configurations suivants :

$$\begin{aligned} \mathbf{q}_a &= (0, 0, 0, 0, 0, 0)^t, & \mathbf{q}_b &= (0, 0, 0, 0, -\frac{\pi}{6}, 0)^t, \\ \mathbf{q}_c &= (0, 0, \frac{\pi}{6}, 0, -\frac{\pi}{6}, 0)^t, & \mathbf{q}_d &= (0, -1.4, 3.7, 0, 0.3, 0)^t. \end{aligned} \quad [2]$$

La configuration \mathbf{q}_a est la position de référence où l'ensemble des variables articulaires sont nulles. La rotation de l'avant dernière articulation rotoïde permet d'obtenir la configuration \mathbf{q}_b . De la même façon, les deux autres configurations sont obtenues par rotation selon chaque axe des articulations rotoïdes.

Le robot Staubli est conçu pour manipuler un *outil* ou *organe terminal*, qui se trouve en bout de la chaîne cinématique. L'outil sera repéré par trois points dont les coordonnées dans le repère de l'atelier forment le vecteur :

$$\mathbf{x} = (a_x^1, a_y^1, a_z^1, a_x^2, a_y^2, a_z^2, a_x^3, a_y^3, a_z^3)^t \in \mathbb{R}^9. \quad [3]$$

Rappelons que la structure du robot est définie par un ensemble de paramètres géométriques. Ils représentent les paramètres de construction ou d'installation du robot (voir [KHA 99a]).

1) Le jeu de paramètres $\theta_0, r_0, \alpha_1, d_1, \theta_1^o$ et r_1 , défini la base du robot par rapport au repère atelier grâce à des rotations et des translations sur les repères. Ces paramètres du robot sont mal connus, en effet le modèle géométrique fait reposer sur ces paramètres l'incertitude due à l'installation du robot dans le repère atelier.

2) Les paramètres b_x^i, b_y^i et $b_z^i, i = 1, 2, 3$ fixent trois points sur l'outil et permettent de définir un repère outil par rapport au repère du dernier corps. Trois contraintes relient ces paramètres car les distances entre les trois points sont connues.

3) Les paramètres géométriques $\alpha_j, d_j, \theta_j^o$ et r_j sont liés au robot lui-même, ce sont des données constructeur. Par exemple, d_3 correspond à la longueur du corps 3 du robot et r_4 à la longueur du corps 4. Les θ_j^o correspondent à l'offset sur la variables articulaires $q_j, j = 2, 3, 4, 5$.

Ces paramètres sont désignés par le vecteur :

$$\mathbf{p} = (r_0, \alpha_1, d_1, r_1, \dots, \alpha_5, d_5, r_5, \alpha_6, d_6, \theta_0, \theta_1^o, \dots, \theta_5^o, b_x^1, b_y^1, b_z^1, b_x^2, b_y^2, b_z^2, b_x^3, b_y^3, b_z^3)^t. \quad [4]$$

2.2. Principe de l'étalonnage

Afin d'estimer les paramètres géométriques \mathbf{p} , nous allons procéder comme suit :

- 1) positionner le robot dans r configurations différentes $\mathbf{q}(1), \dots, \mathbf{q}(r)$;
- 2) mesurer la position des trois points fixés sur l'organe terminal $\mathbf{x}(1), \dots, \mathbf{x}(r)$ (les $\mathbf{x}(i)$ appartiennent à \mathbb{R}^9) ;
- 3) générer les contraintes grâce au modèle géométrique direct. Comme nous allons le rappeler dans la section 2.3 suivante, ces contraintes sont de la forme :

$$\mathbf{x}(k) = \mathbf{f}(\mathbf{p}, \mathbf{q}(k)), \quad k = \{1, \dots, r\} ; \quad [5]$$

- 4) résoudre [5] par la méthode de propagation de contraintes présentée en section 3 pour déterminer \mathbf{p} .

Rappelons que pour chaque configuration du robot, le vecteur articulaire $\mathbf{q}(k)$ et le vecteur de position des trois points de l'organe terminal $\mathbf{x}(k)$ sont mesurés. L'incertitude e sur chaque mesure permet d'affecter à chaque variable un intervalle auquel elle appartient. A chaque mesure est ajoutée et retranchée e pour obtenir les bornes supérieure et inférieure de l'intervalle.

2.3. Génération des contraintes

La méthode de propagation de contraintes, utilisée pour résoudre le problème d'étalonnage, repose sur la formalisation du problème sous forme de contrainte.

Le modèle géométrique direct est un formalisme utilisé pour obtenir le minimum de contraintes nécessaire lors du calcul avec des réels. C'est le premier formalisme qui est présenté et qui a permis d'obtenir les premières réductions de nos intervalles de recherche. Notre deuxième approche a été de s'orienter vers un formalisme se rapprochant plus des contraintes, notamment en intégrant la notion de contrainte globale.

Les formalismes sont donc présentés dans les deux rubriques suivantes, les résultats seront comparés section 4.

Formalisme utilisant le modèle géométrique direct. Connaissant le vecteur des variables articulaires \mathbf{q} et le vecteur des paramètres géométriques \mathbf{p} , il est possible de calculer la position d'un point de l'organe terminal dans le repère atelier avec le modèle géométrique direct [KHA 99a]. Ce modèle est appliqué aux trois points de l'outil pour obtenir la relation :

$$\mathbf{x} = \mathbf{f}(\mathbf{p}, \mathbf{q}), \quad [6]$$

où \mathbf{f} s'exprime suivant l'algorithme :

Algorithme f	
entrées : $\mathbf{q} = (q_1, \dots, q_6)^t$, $\mathbf{p} = (\alpha_j, d_j, r_j, \theta_0, \theta_j^o, b_x^i, b_y^i, b_z^i, \dots)^t$.	
sortie : $\mathbf{x} = (a_x^1, a_y^1, a_z^1, a_x^2, a_y^2, a_z^2, a_x^3, a_y^3, a_z^3)^t$.	
1	$\mathbf{M} := \begin{pmatrix} 1 & 0 & 0 & d_6 \\ 0 & \cos \alpha_6 & -\sin \alpha_6 & 0 \\ 0 & \sin \alpha_6 & \cos \alpha_6 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos q_6 & -\sin q_6 & 0 & 0 \\ \sin q_6 & \cos q_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix};$
2	for $j := 5$ to 1,
3	$\theta := \theta_j^o + q_j;$
4	$\mathbf{M} := \begin{pmatrix} 1 & 0 & 0 & d_j \\ 0 & \cos \alpha_j & -\sin \alpha_j & 0 \\ 0 & \sin \alpha_j & \cos \alpha_j & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & r_j \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{M};$
5	end for
6	$\mathbf{M} := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \cos \theta_0 & -\sin \theta_0 & 0 & 0 \\ \sin \theta_0 & \cos \theta_0 & 0 & 0 \\ 0 & 0 & 1 & r_0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{M};$
7	for $i := 1$ to 3, $\mathbf{b}^i = (b_x^i, b_y^i, b_z^i, 1)^t$;
8	$\mathbf{x} := \begin{pmatrix} \mathbf{M} & 0 & 0 \\ 0 & \mathbf{M} & 0 \\ 0 & 0 & \mathbf{M} \end{pmatrix} \begin{pmatrix} \mathbf{b}^1 \\ \mathbf{b}^2 \\ \mathbf{b}^3 \end{pmatrix}.$

Après développement scalaire de relations matricielles de cet algorithme, nous pouvons exprimer \mathbf{f} comme une composition de produits, de sommes, et de fonctions trigonométriques. Le modèle est donc fortement non linéaire.

En effet nous obtenons des contraintes scalaires de la forme :

$$t = \sin \alpha_2 \sin \theta_1 \sin \alpha_3 \sin \theta_3 + (\cos \theta_1 \cos \theta_2 - \cos \alpha_2 \sin \theta_2 \sin \theta_1) \cos \theta_3 - (\cos \alpha_2 \cos \theta_2 \sin \theta_1 + \cos \theta_1 \sin \theta_2) \cos \alpha_3 \sin \theta_3. \quad [7]$$

REMARQUE. — Certaines expressions telles que $\cos \theta_1$ ou $\cos \theta_2 \sin \theta_1$ apparaissent plusieurs fois, ces redondances pénalisent la propagation de contrainte en temps de calcul. La compilation des contraintes permet de minimiser ce problème.

REMARQUE. — Les multi-occurrences de chaque variable engendrent un pessimisme lors du calcul par intervalle. L'obtention des plus petits intervalles encadrant l'ensemble solution n'est alors plus garantie. La deuxième formulation avec la contrainte angle permet une réduction du nombre d'occurrence, le sinus et le cosinus d'un angle étant traités avec une seule contrainte.

Formulation utilisant la contrainte angle. Comme pour le formalisme précédent, nous établissons une relation entre le vecteur des variables articulaires \mathbf{q} , le vecteur des paramètres géométriques \mathbf{p} et la position \mathbf{x} de trois points de l'organe terminal dans le repère atelier.

La relation [6] peut se réécrire sous la forme :

$$\mathbf{x} = \mathbf{g}(\mathbf{p}, \mathbf{q}), \quad [8]$$

où \mathbf{g} est donné dans l'algorithme suivant :

Algorithme g	
entrées :	$\mathbf{q} = (q_1, \dots, q_6)^t$, $\mathbf{p} = (r_0, \alpha_1, d_1, r_1, \dots, \alpha_5, d_5, r_5, \alpha_6, d_6, \theta_0, \theta_1^o, \dots, \theta_5^o)^t$.
sortie :	$\mathbf{x} = (a_x^1, a_y^1, a_z^1, a_x^2, a_y^2, a_z^2, a_x^3, a_y^3, a_z^3)^t$.
1	for $i := 1$ to 3,
2	$(c_1 \ c_2 \ c_3)^t := (b_x^i \ b_y^i \ b_z^i)^t$;
3	$\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} := \text{angle}(q_6, \begin{pmatrix} c_1 \\ c_2 \end{pmatrix})$;
4	$c_1 := c_1 + d_6$; $\begin{pmatrix} c_2 \\ c_3 \end{pmatrix} := \text{angle}(\alpha_6, \begin{pmatrix} c_2 \\ c_3 \end{pmatrix})$;
5	for $k := 5$ to 1,
6	$c_3 := c_3 + r_k$; $\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} := \text{angle}(\theta_k^o + q_k, \begin{pmatrix} c_1 \\ c_2 \end{pmatrix})$;
7	$c_1 := c_1 + d_k$; $\begin{pmatrix} c_2 \\ c_3 \end{pmatrix} := \text{angle}(\alpha_k, \begin{pmatrix} c_2 \\ c_3 \end{pmatrix})$;
8	end for
9	$c_3 := c_3 + r_0$; $\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} := \text{angle}(\theta_0, \begin{pmatrix} c_1 \\ c_2 \end{pmatrix})$;
10	$\mathbf{x}_i := (c_1 \ c_2 \ c_3)^t$;
11	end for
12	$\mathbf{x} := \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{pmatrix}$.

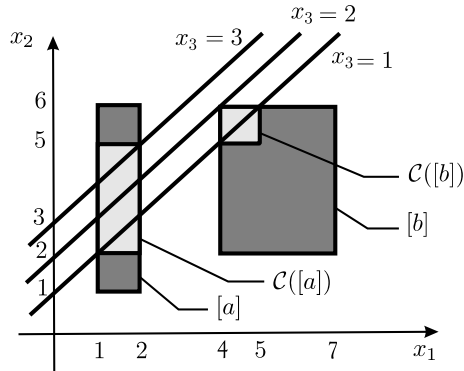


Figure 3. Contraction de la contrainte $x_3 = x_2 - x_1$ sur les pavés $[a]$ et $[b]$

La fonction $angle(\beta, \mathbf{c})$ génère un vecteur de sortie $\mathbf{c}' \in \mathbb{R}^2$ résultat d'une rotation de \mathbf{c} , dans le sens direct, d'un angle β .

3. Propagation de contraintes sur les intervalles

Sur un problème de grande dimension les techniques de bisection demandent un temps de résolution exponentiel. Pour cette raison, nous avons choisi dans un premier temps de n'utiliser que la propagation de contraintes sur les intervalles. Cette section présente donc le principe de cette méthode qui sera utilisée pour l'étalonnage du robot, les problèmes de ce type de résolution et les solutions envisagées.

Projection. Pour un ensemble S fermé, notons $C_S^*([\mathbf{x}])$ le plus petit pavé (c'est-à-dire le produit cartésien d'intervalles réels) contenant $S \cap [\mathbf{x}]$. C_S^* est appelé *contracteur* (car il enferme $S \cap [\mathbf{x}]$) *optimal* (car il n'en existe pas de plus petit, repéré par l'étoile en exposant) associé à l'ensemble S . Le calcul de C_S^* est communément appelé *projection*. Un ensemble pour lequel on dispose d'un contracteur optimal est appelé *ensemble primitif*. Les deux exemples suivants illustrent cette notion de projection pour des ensembles S primitifs.

Exemple 1. Soit l'ensemble :

$$S = \{(x_1, x_2, x_3) \in \mathbb{R}^3 \mid x_3 = x_2 - x_1\}. \tag{9}$$

et les domaines :

$$[a] = [1, 2] \times [1, 6] \times [1, 3], \tag{10}$$

$$[b] = [4, 7] \times [2, 6] \times [1, 3]. \tag{11}$$

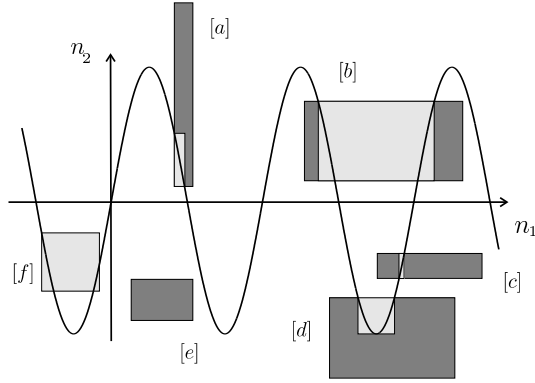


Figure 4. Opérateur de contraction de la contrainte sinus

Le pavé obtenu après contraction de [a] et de [b] par C_S^* est le plus petit pavé contenant toutes les solutions appartenant à S . La figure 3 illustre les contractions :

$$C_S^*([a]) = [1, 2] \times [2, 5] \times [1, 3], \tag{12}$$

$$C_S^*([b]) = [4, 5] \times [5, 6] \times [1, 2]. \tag{13}$$

Exemple 2. Considérons maintenant l'ensemble :

$$S = \{(x, y) \in \mathbb{R}^2 \mid y = \sin(x)\}. \tag{14}$$

Les pavés [a], ... , [f] et [e] sont représentés en gris foncé sur la figure 4, les pavés contractés $C_S^*([x] \times [y])$ sont en gris clair. Appliquer le contracteur C_S^* sur un pavé $[x] \times [y]$ génère le plus petit pavé contenant l'ensemble des points appartenant à $S \cap [x] \times [y]$. Sur la figure 4, différentes configurations sont représentées : (i) le pavé [e] n'intersecte aucun point de la contrainte sinus et donc $C_S^*([e]) = \emptyset$, (ii) le pavé [f] ne peut être contracté $C_S^*([f]) = [f]$, (iii) les pavés [b] et [c] ne peuvent être contractés que suivant un axe et (iv) les pavés [a] et [d] peuvent être contractés suivant les deux directions.

Propagation. Afin d'illustrer la propagation de contrainte, considérons maintenant l'ensemble S défini par la conjonction de deux contraintes c_1 et c_2 :

$$S \triangleq \{(x_1, x_2) \in \mathbb{R}^2 \mid c_1 : x_1 = \sin(x_2) \text{ et } c_2 : x_1 = x_2^3\}. \tag{15}$$

Cherchons des intervalles, si possible petits, qui contiennent toutes les valeurs possibles pour x_1 et x_2 . Ceci permettra de réaliser un contracteur C_S pour S qui ne sera plus nécessairement optimal. L'ensemble S peut être défini comme l'intersection des deux ensembles primitifs :

$$S_1 = \{(x_1, x_2) \mid x_1 = \sin(x_2)\}, \tag{16}$$

$$S_2 = \{(x_1, x_2) \mid x_1 = (x_2)^3\}, \tag{17}$$

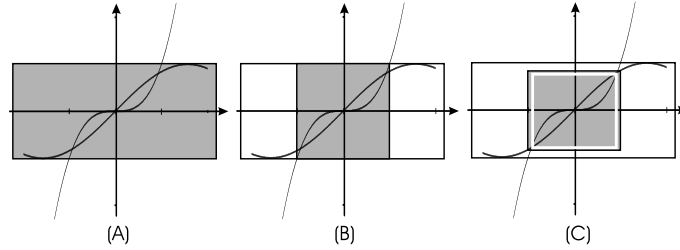


Figure 5. Représentation des trois premières étapes de propagation de contraintes

pour lesquels on dispose de contracteurs $\mathcal{C}_{\mathbb{S}_1}^*$ et $\mathcal{C}_{\mathbb{S}_2}^*$ optimaux. On obtient la contraction suivante :

$$\mathcal{C}_{\mathbb{S}_1}^*([\mathbf{x}]) = [-1, 1] \times] - \infty, +\infty[\quad [18]$$

puis :

$$\mathcal{C}_{\mathbb{S}_2}^*(\mathcal{C}_{\mathbb{S}_1}^*([\mathbf{x}])) = [-1, 1] \times [-1, 1] \quad [19]$$

et encore :

$$\mathcal{C}_{\mathbb{S}_2}^*(\mathcal{C}_{\mathbb{S}_2}^*(\mathcal{C}_{\mathbb{S}_1}^*([\mathbf{x}]))) = [-0.84, 0.85] \times [-1, 1]. \quad [20]$$

On assiste donc pas à pas à la contraction du pavé de $[\mathbf{x}]$. Les pavés obtenus à la suite de trois contractions sont visualisés en gris sur le schéma 5. Le principe de propagation de contraintes est donc de composer itérativement les contracteurs $\mathcal{C}_{\mathbb{S}_2}^*$ et $\mathcal{C}_{\mathbb{S}_1}^*$ afin d'obtenir un contracteur $\mathcal{C}_{\mathbb{S}}$ pour $\mathbb{S} = \mathbb{S}_1 \cap \mathbb{S}_2$. La composition des contracteurs optimaux $\mathcal{C}_{\mathbb{S}_1}^*$ et $\mathcal{C}_{\mathbb{S}_2}^*$ s'arrête à l'obtention d'un point fixe $[\bar{\mathbf{x}}]$ qui satisfait :

$$\mathcal{C}_{\mathbb{S}_1}^*([\bar{\mathbf{x}}]) = \mathcal{C}_{\mathbb{S}_2}^*([\bar{\mathbf{x}}]) = [\bar{\mathbf{x}}]. \quad [21]$$

Attention, même si $\mathcal{C}_{\mathbb{S}_1}^*$ et $\mathcal{C}_{\mathbb{S}_2}^*$ sont tous les deux optimaux, le contracteur $\mathcal{C}_{\mathbb{S}}$ n'est pas nécessairement optimal. En effet, les plus petits intervalles contenant toutes les solutions sont représentés par le rectangle blanc de la figure 5c et la propagation de contrainte s'arrête au point fixe $[\bar{\mathbf{x}}]$ parfois avant l'obtention de cette boîte minimale.

Décomposition des contraintes. Les contraintes de notre problème d'étalonnage sont généralement complexes et il faut effectuer une décomposition. Afin d'illustrer cette décomposition, considérons les contraintes :

$$y_1 = \cos(i_1 + i_2) + \sin(i_1 + i_2), \quad [22]$$

$$y_2 = i_3 \cdot \sin^2(i_1 + i_2). \quad [23]$$

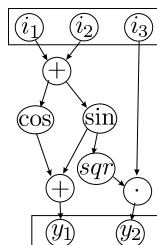


Figure 6. Représentation du DAG des contraintes

Ces dernières peuvent être décomposées en contraintes primitives :

$$\begin{aligned}
 a_1 &= i_1 + i_2, & a_5 &= i_1 + i_2, \\
 a_2 &= \cos(a_1), & a_6 &= \sin(a_5), \\
 a_3 &= i_1 + i_2, & a_7 &= a_6^2, \\
 a_4 &= \sin(a_3), & y_1 &= a_2 + a_4, \\
 y_1 &= a_2 + a_4. & y_2 &= i_3 \cdot a_7.
 \end{aligned}
 \tag{24}$$

Aux variables intermédiaires introduites sont associées à un domaine égal à $]-\infty, +\infty[$ car *a priori*, aucune information sur ces domaines n'est disponible.

Compilation des contraintes. La liste des contraintes primitives générées par décomposition peut contenir des contraintes inutiles qui peuvent pénaliser considérablement le processus de propagation. Dans notre exemple, le sinus des variables $i_1 + i_2$ apparaît deux fois avec les variables temporaires a_4 et a_6 .

L'élimination automatique des contraintes inutiles peut se faire grâce à des techniques utilisées pour l'élaboration des compilateurs. Ces techniques utilisent les notions de DAG (Directed Acyclic Graph) et de table de hachage (voir [AHO 95]). Pour notre exemple, les contraintes obtenues après compilation sont de la forme :

$$\begin{aligned}
 a_1 &= i_1 + i_2, \\
 a_2 &= \cos(a_1), \\
 a_4 &= \sin(a_1), & a_7 &= a_4^2, \\
 y_1 &= a_2 + a_4. & y_2 &= i_3 \cdot a_7.
 \end{aligned}
 \tag{25}$$

En résumé. Pour effectuer une propagation de contraintes, on décompose les contraintes en contraintes primitives, on les compile afin d'éviter les redondances inutiles dans les contraintes. Enfin, on projette chacune des contraintes primitives jusqu'au point fixe.

Amélioration par l'ajout de contrainte globale. On connaît un contracteur optimal C_S^* pour les contraintes primitives mais le contracteur obtenu grâce à la propagation rétropropagation n'est pas dans tous les cas optimal. Regrouper certaines

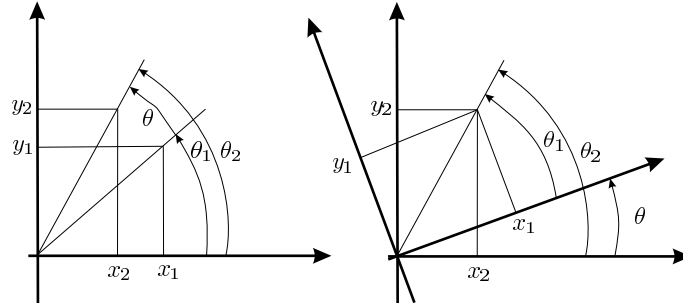


Figure 7. Correspondance changement de repère et rotation

contraintes primitives pour les traiter directement de façon optimale permet de réduire ce problème. C'est ce qui est proposé avec l'utilisation de la contrainte angle :

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \text{angle} \left(\theta, \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \right). \quad [26]$$

Pour cette contrainte, nous n'avons pas encore de contracteur optimal C_S^* .

En l'absence de ce contracteur, l'intérêt d'une telle contrainte peut être mis en évidence par le rajout de contraintes redondantes. Dans une optique de calcul avec les réels une seule contrainte permet d'obtenir les valeurs souhaitées. Avec l'approche propagation de contraintes sur les intervalles, l'ensemble des contraintes liées à la rotation permettent une contraction plus forte sur les domaines initiaux. Les contraintes supplémentaires qui pénalisent souvent les méthodes classiques, permettent ici de relancer la propagation de contrainte.

Pour trouver l'ensemble des contraintes exprimons la rotation de deux manières différentes. Soit par le déplacement d'un angle θ , soit par un changement de repère comme illustré sur la figure 7. Dans le premier formalisme le point (x_2, y_2) est l'image du point (x_1, y_1) par une rotation dans le sens direct d'angle θ . Dans le second formalisme (x_1, y_1) et (x_2, y_2) sont les coordonnées d'un même point dans des repères de même origine mais séparés par une rotation d'un angle θ .

Ce qui correspond aux contraintes :

1) équations de changement de repère entre le repère 1 et 2 :

$$x_2 = x_1 \cos \theta - y_1 \sin \theta, \quad [27]$$

$$y_2 = x_1 \sin \theta + y_1 \cos \theta, \quad [28]$$

2) équations de changement de repère entre le repère 2 et 1 :

$$x_1 = x_2 \cos \theta + y_2 \sin \theta, \quad [29]$$

$$y_1 = -x_2 \sin \theta + y_2 \cos \theta, \quad [30]$$

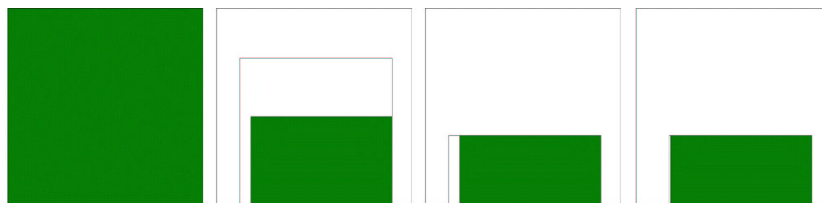


Figure 8. Illustration des contractions suite à l'ajout de contraintes redondantes. Chacune des 4 sous-figures correspond au pavé $[-9, 0.6] \times [5.5, 5.6]$ dans l'espace (θ, x_2)

3) égalité des longueurs liant l'origine au point 1 et 2 sur la figure 7 :

$$\sqrt{x_1 + y_1} = \sqrt{x_2 + y_2}, \quad [31]$$

4) relation angulaire :

$$\tan(\theta) = \tan(\theta_2 - \theta_1), \quad [32]$$

$$= \frac{\sin \theta_2 \cos \theta_1 - \cos \theta_2 \sin \theta_1}{\cos \theta_2 \cos \theta_1 + \sin \theta_2 \sin \theta_1}, \quad [33]$$

$$\tan(\theta) = \frac{y_2 x_1 - x_2 y_1}{x_2 x_1 + y_2 y_1}. \quad [34]$$

Un test a été réalisé pour illustrer les contractions dues à l'ajout des contraintes redondantes. Sur un pavé initial, est appliqué quatre opérateurs de contractions, tels que :

$$[x_1] \times [y_1] \times [\theta] \times [x_2] \times [y_2] = [2.9, 3.5] \times [1.9, 5.5] \times [-9, 0.6] \times [5.5, 5.6] \times [0, 10]. \quad [35]$$

Le premier ne prend en compte que les contraintes [27] et [28], et ne permet suivant les variables θ et y_2 aucune contraction. Ceci est représenté par le premier carré à gauche sur la figure 8.

Les trois carrés suivants correspondent à une contraction due à l'ajout de contrainte redondante : respectivement de gauche à droite, on a ajouté les contraintes [29] et [30], puis [31] et [34]. Les domaines obtenus sont avec l'ajout de contraintes de plus en plus petit. On peut remarquer que ces points fixes en gris sur la figure sont obtenus en appliquant deux fois la propagation et retro-propagation sur l'ensemble des contraintes primitives. La première itération correspond aux cadres apparaissant sur la figure. Le point fixe n'est donc pas obtenu directement.

REMARQUE. — Le domaine obtenu avec l'ensemble des contraintes n'est pas encore le domaine minimal satisfaisant les contraintes. La mise en place d'un algorithme donnant directement la boîte minimale pour la contrainte globale permettrait d'obtenir ce pavé minimal et de façon plus rapide.

4. Simulation

Dans cet article l'étalonnage des paramètres géométriques n'a pas été réalisé sur des mesures réelles. Une simulation a donc été effectuée de façon à se rapprocher de la réalité. Une fois la méthode mise au point, le passage aux mesures réelles pourra être envisagé.

Nous avons choisi les paramètres géométriques du robot STAUBLI RX90 avec les valeurs nominales données par la table suivante :

j	α_j	d_j	θ_j^o	r_j
0	-	-	$\frac{\pi}{2}$	0.5
1	0.1	0	0	0
2	$-\frac{\pi}{2}$	0	0	0
3	0	0.5	0	0
4	$\frac{\pi}{2}$	0	0	0.5
5	$-\frac{\pi}{2}$	0	0	0
6	$\frac{\pi}{2}$	0	-	-

Tableau 1. Valeurs nominales des paramètres géométriques recherchés

Les trois points fixés sur l'organe terminal sont repérés par leurs coordonnées b_x^i , b_y^i et b_z^i dans le repère lié au corps terminal. Ces coordonnées sont des paramètres géométriques pour l'étalonnage et sont choisies avec les valeurs données par la table suivante :

i	b_x^i	b_y^i	b_z^i
1	0.1	0.2	0.1
2	0.1	0.1	0.2
3	0.2	0.1	0.1

Tableau 2. Coordonnées nominales des trois points fixés sur l'organe terminal

En prenant une configuration aléatoire du vecteur \mathbf{q} , nous effectuons une simulation du modèle direct qui nous donne les positions $\mathbf{x} \in \mathbb{R}^9$ des trois points de l'organe terminal. Pour cinquante configurations \mathbf{q} différentes, nous avons ainsi obtenu par simulation un ensemble de cinquante vecteurs de position \mathbf{x} . Afin de prendre en compte les incertitudes de mesure présentes dans les mesures réelles, nous avons ajouté à chaque composante des vecteurs \mathbf{q} et \mathbf{x} un bruit aléatoire uniforme compris dans l'intervalle d'incertitude sur la mesure. Nous obtenons ainsi des intervalles pour chaque configuration (\mathbf{x}, \mathbf{q}) contenant les points de simulation sans avoir d'information sur la localisation de ces points dans les intervalles obtenus.

Exemple 3. La variable articulaire q_1 est tirée de façon aléatoire dans son domaine de variation $[-\pi, \pi]$. Avec cette valeur, a_x^1 est calculé par simulation du modèle géométrique.

Sur ces deux valeurs sont ajoutées un bruit aléatoire uniforme :

$$\tilde{a}_x^1 = a_x^1 + e_x, \quad [36]$$

$$\tilde{q}_1 = q_1 + e_q. \quad [37]$$

où e_x appartient à l'intervalle $[-10^{-4}, 10^{-4}]$ correspondant à l'incertitude de mesure sur les capteurs mesurant \mathbf{x} , et e_q à l'intervalle $[-10^{-5}, 10^{-5}]$ correspondant au pas du codeur générant \mathbf{q} .

Les intervalles pour a_x^1 et q_1 :

$$[a_x^1] = [\tilde{a}_x^1 - 10^{-4}, \tilde{a}_x^1 + 10^{-4}], \quad [38]$$

$$[q_1] = [\tilde{q}_1 - 10^{-5}, \tilde{q}_1 + 10^{-5}], \quad [39]$$

permettent de prendre en compte les incertitudes sur les valeurs supposées réelles \tilde{a}_x^1 et \tilde{q}_1 .

De même il faut déterminer un intervalle de recherche pour chaque paramètre que nous cherchons à étalonner. Ces domaines de recherche correspondent à la connaissance que nous avons sur les paramètres avant de réaliser l'étalonnage.

Exemple 4. Les paramètres de la base et la position des trois points sur l'organe terminal sont mal connus, nous prenons donc un domaine de recherche de $\pm 0.1m$ sur les longueurs et de $\pm 0.17rad$ pour les angles. Pour les paramètres du robot lui-même nous disposons d'une meilleure information, la recherche s'effectue à $\pm 0.01m$ pour les longueurs et à $\pm 0.035rad$ pour les angles.

5. Résultats

L'ensemble de la procédure de simulation a été suivi pour tous les tests présentés dans cette section.

Le premier des tests, réalisé pour l'étalonnage de *tous les paramètres* \mathbf{p} , n'a pas permis d'obtenir une contraction des domaines de recherche. La méthode de résolution pour l'étalonnage doit être améliorée.

Pour les deux tests suivants, l'étalonnage est limité aux longueurs mal connues et aux longueurs non nulles des paramètres du robot. Tous les autres paramètres n'étant pas étalonnés sont supposés pour cet essai connus avec une précision de 10^{-5} .

L'arrêt de la propagation s'effectue quand la contraction du vecteur d'intervalles représentant les variables n'est plus que d'un facteur 10^{-5} .

Le tableau 3 nous donne les résultats pour l'étalonnage des 14 paramètres de longueurs du robot Staubli Rx90. Le tableau présente pour chaque variable l'intervalle de recherche, l'intervalle obtenu par simple propagation et celui obtenu en utilisant la contrainte angle.

	Domaines de recherche	Domaines obtenus avec la formulation matricielle	Domaines obtenus avec la contrainte angle	$\frac{\text{Taille1}}{\text{Taille2}}$
r_0	[0.4, 0.6]	[0.488772, 0.501644]	[0.494046, 0.50101]	0.54
d_1	[0, 0.1]	[0, 0.000964102]	[0, 0.000558009]	0.58
r_1	[0, 0.1]	[0, 0.00911161]	[0, 0.00693694]	0.76
d_3	[0.49, 0.51]	[0.497796, 0.50185]	[0.498385, 0.501133]	0.68
r_4	[0.49, 0.51]	[0.499031, 0.502242]	[0.499216, 0.50114]	0.60
b_x^1	[0, 0.2]	[0.0993651, 0.10123]	[0.0996052, 0.100629]	0.55
b_y^1	[0.1, 0.3]	[0.199615, 0.200586]	[0.199502, 0.200455]	0.98
b_z^1	[0, 0.2]	[0.0992392, 0.100921]	[0.0997107, 0.100714]	0.60
b_x^2	[0, 0.2]	[0.0993244, 0.101355]	[0.0996747, 0.100712]	0.51
b_y^2	[0, 0.2]	[0.099531, 0.100757]	[0.0994585, 0.10031]	0.69
b_z^2	[0.1, 0.3]	[0.199172, 0.200965]	[0.199535, 0.200642]	0.62
b_x^3	[0.1, 0.3]	[0.199372, 0.201203]	[0.199689, 0.200578]	0.49
b_y^3	[0, 0.2]	[0.0995955, 0.100753]	[0.0997562, 0.100319]	0.49
b_z^3	[0, 0.2]	[0.0991525, 0.100918]	[0.0995661, 0.100557]	0.56

Tableau 3. Comparatif donnant les résultats de l'étalonnage pour les deux formalisations

La contraction du domaine de recherche varie de 80 % à 99 %. Ce chiffre est peu significatif, en effet la taille du domaine de recherche importe peu pour cette méthode de calcul. En revanche la taille des intervalles obtenus, autrement dit l'incertitude calculée autour de nos paramètres, est intéressante. Cette taille varie de 0,01m pour les paramètres les moins contractés à 0,001m.

Le deuxième formalisme intégrant la contrainte angle permet une amélioration de la contraction de plus de 30 %. La dernière colonne du tableau présente ce rapport entre les tailles des intervalles obtenus avec les deux formalismes.

Ce résultat est obtenu en 137, 3s alors que le premier formalisme trouve la solution en 8, 34s. Ces temps de calcul peuvent laisser penser que l'amélioration de la contraction n'est pas avantageuse comparée à la perte de temps engendrée. Ce temps de plus de 2min peut s'expliquer par le rajout de nombreuses contraintes redondantes. Il augmente le nombre contraintes primitives (le fichier passe de 360Ko à 837Ko) et oblige la procédure de propagation de contrainte à balayer l'ensemble des contraintes primitives plusieurs fois. La mise en place de la contrainte globale angle traitée non plus avec des contraintes redondantes mais directement, permettra d'obtenir, si ce n'est un meilleur résultat, du moins le même qu'avec les contraintes redondantes dans un temps beaucoup plus court.

Un étalonnage uniquement pour quelques paramètres angulaires permet une contraction de ces angles. Le nombre d'angles contractés augmente lors de l'utilisation du deuxième formalisme. Cette progression est encourageante pour la mise en place de l'étalonnage sur l'ensemble des paramètres.

6. Conclusion

Dans cet article, nous avons présenté les premiers résultats concernant le problème de l'étalonnage géométrique des robots. Notre contribution principale est d'avoir, pour la première fois, testé cet étalonnage par méthode de propagation de contrainte dont l'un des avantages est de ne perdre aucune solution.

Cette propriété permet d'obtenir une information sur l'intervalle d'incertitude autour de la valeur du paramètre recherché. Mais cette méthode ne donne pas encore la précision nécessaire pour la connaissance de l'ensemble des paramètres du robot.

Les améliorations apportées par le nouveau formalisme laissent supposer que la mise en place de la notion de contrainte globale pour l'angle permettra une meilleure contraction. Avec les contracteurs associés nous espérons obtenir plus rapidement un meilleur encadrement pour nos paramètres à estimer.

D'autres types de contracteurs existent : la box-consistance prenant en compte la dérivée des contraintes, le simplex appliqué uniquement aux parties linéaires des contraintes... Toutes ces méthodes peuvent permettre de s'approcher du plus petit pavé encadrant nos paramètres. La collaboration entre ces outils différents pour qu'ils se débloquent mutuellement semble une perspective intéressante.

7. Bibliographie

- [AHO 95] AHO A., SETHI R., ULLMAN J., *Compilateurs Principes, techniques et outils*, Collection Informatique intelligence artificielle, InterEditions, 1995.
- [BLI 01] BLIEK C., ALL, *Algorithms for Solving Nonlinear Constrained and Optimisation Problems : State of the Art*, Coconut project, 2001.
- [EVE 89] EVERETT L., « Forward calibration of closed loop jointed manipulators », *The Int. J. of Robotics Research*, vol. 8, n° 4, 1989, p. 85-91.
- [HOL 89] HOLLERBACH J., « A survey of kinematic calibration », *The Robotics Review, MIT Press Cambridge*, vol. 1, 1989, p. 207-242.
- [JAU 01] JAULIN L., KIEFFER M., DIDRIT O., WALTER E., *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*, Springer-Verlag, London, 2001.
- [KHA 99a] KHALIL W., DOMBRE E., *Modélisation, identification et commande des robots*, Collection Robotique, Hermès, Paris, 1999.
- [KHA 99b] KHALIL W., LEMOINE P., « Gecaro, a system for the geometric calibration of robots », *Revue APII-JESA*, vol. 33, 1999.
- [KHA 00] KHALIL W., BESNARD S., « Comparison study of the geometric parameters calibration methods », *Int. Journal of robotics and Automation*, vol. 15, n° 2, 2000, p. 56-67.
- [VAN 99] VANEMDEN M., « Algorithmic Power from Declarative Use of Redundant Constraints. », *Constraints*, vol. 4, n° 4, 1999, p. 363-381.