Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

Interval Methods for Certification of the Kinematic Calibration of Parallel Robots

David Daney and Yves Papegay
COPRIN research group, INRIA Sophia Antipolis,
2004 Route des Lucioles, 06902 Sophia Antipolis, France
Email: {David.Daney,Yves.Papegay}@sophia.inria.fr

Arnold Neumaier
Institut für Mathematik, Universität Wien
Strudlhofgasse 4, A-1090 Wien, Austria
Email: Arnold.Neumaier@univie.ac.at

***Abstract*— In this paper, we demonstrate how methods based on interval arithmetic and interval analysis can be used to achieve numerical certification of the kinematic calibration of a parallel robots. We introduce our work by describing the usual calibration methods and the motivations for a numerical certification. Then, we briefly present the interval methods we used and the kinematic calibration problem. In the main part, we develop our certified approach of this problem in the case of a Gough platform, and we show with numerical examples how this approach avoids wrong solutions produced by classical approach. Details on implementation and performance are also given.**

I. Introduction

High accuracy of position and orientation is a characteristic feature of parallel manipulators that makes them appealing in a lot of applications. However, such an accuracy relies on a robust and accurate calibration of the physical configuration of the robot. This is a difficult task from both theoretical and practical point of view, even if efficiency is not critical as the calibration may be performed off-line.

A robot's configuration is related to kinematic parameters of a robot through the equations of the kinematic model. Calibration is achieved by measuring several robot configurations and identifying the corresponding kinematic parameters. For mathematical reasons, the number of equations given by the measurements has to be at least as large as the number of unknown parameters. Since the measurement data are usually given by a captor, it is necessary to take into account the noise associated with this device. So in practice, the number of equations is larger in order to reduce the sensitivity of the calibration to the uncertainty attached to the data. In this case, the system of equations to solve is *over-constrained*.

The classical method to solve such an over-constrained problem is a least-squares method. But the mere convergence of this iterative method cannot guarantee that, after calibration, the accuracy of the robot is improved in the whole workspace. In practice, post-processing is therefore necessary to validate the results of such a calibration. Unfortunately, in the case of Gough platforms, this step is very costly [1].

Some improvements of the least-squares method, providing a quality index for each solution, have been proposed when a noise model can be associated with the data uncertainties [2]. That may be done if the distribution of the measurement error is known (e.g., from the documentation of the captor). But this noise model may be difficult to obtain – for example when using mechanical constraints for calibration, or for certain measurement devices.

Even in the best cases, only probabilistic results are produced. In this paper we propose a method that gives a *certified* approximation in the sense that, for a set of measurements given with attached uncertainties, we return a list of intervals for the kinematics parameters such that any solution corresponding to an instance of configuration satisfying the measurements has to belong to those intervals. This method is an new version based on interval arithmetic, using interval analysis of the so-called implicit or inverse calibration method, the most studied method for the identification of the kinematic parameters of a parallel robot [3]–[5]).

Extended to a representation of the parameters in terms of intervals – and to the associated arithmetic (Section II), the basic system of equations for the kinematic calibration of Gough platform is developed (Section III). Our algorithm for obtaining the certified solution of this system is described in detail in Section IV. A simulation (Section V) producing certified results reveals that a least-squares method may provide a result which is not compatible with the corresponding measurement data.

II. Interval Arithmetic

Interval arithmetic, introduced by Moore [6], is based on the representation of an uncertain variable x as an interval $\mathbf{x} = [\underline{x}, \overline{x}]$ representing a (possibly conservative) worst case estimate of the range of x.

The *interval evaluation* of a real-valued function $f(x_1, \ldots, x_n)$ is an interval $\mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ such that

$$f(x_1, \ldots, x_n) \in \mathbf{f}(\mathbf{x}) \quad \text{for all } x_1 \in \mathbf{x}_1, \ldots, x_n \in \mathbf{x}_n. \quad (1)$$

The tightest interval evaluation is the range, but any interval containing the range of a function is an interval evaluation of this function. There are numerous ways to calculate an interval evaluation function [7] which produce more or less overestimation of the range; controlling the latter is the key to a successful use of intervals.

The simplest interval evaluation is the *natural evaluation*, in which all mathematical operators in an expression for f are simply substituted by their interval equivalents; the result is

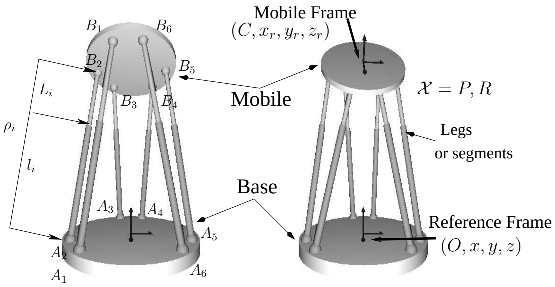

Fig. 1. Gough platform.

highly dependent on the symbolic expression used. Another interesting interval evaluation is the *centered form* (or linear Taylor form) defined as follow :

$$\mathbf{f}_T(\mathbf{x}) = f(x) + \mathbf{A}(\mathbf{x} - x) \tag{2}$$

where $\mathbf{A} = \mathbf{f}[x, \mathbf{x}]$ is a suitable $n \times n$ interval matrix, called a slope matrix.

In the following, we use the following notation related to an interval $\mathbf{x} = [\underline{x}, \overline{x}]$: We write $inf(\mathbf{x})$ for $\underline{x}$, $sup(\mathbf{x})$ for $\overline{x}$, $mid(\mathbf{x})$ for $\frac{1}{2}(\underline{x} + \overline{x})$ and $rad(\mathbf{x})$ for $\overline{x} - \underline{x}$.

III. Kinematics and Calibration

We are studying a Gough platform as depicted in Figure 1. This manipulator consists in two rigid bodies, the *base* and the mobile *platform*, connected by 6 *legs*.

The robot configuration (P, R) is given by a position P and a rotation matrix R. It is associated to the length variation L_i of each leg measured by an "internal" sensor. The matrix R is given is terms of Rodrigues parameters (q_1, q_2, q_3), where $(1 + q_1^2 + q_2^2 + q_3^2)R$ is

$$\begin{pmatrix} 1 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 - 2q_3 & 2q_1q_3 + 2q_2 \\ 2q_1q_2 + 2q_3 & 1 - q_1^2 + q_2^2 - q_3^2 & 2q_2q_3 - 2q_1 \\ 2q_1q_3 - 2q_2 & 2q_2q_3 + 2q_1 & 1 - q_1^2 - q_2^2 + q_3^2 \end{pmatrix}.$$

Physically, each leg is attached to the base by a U-joint and to the platform by a ball joint, and 23 parameters are required to fully model each leg [8]. But, as shown in [9], the principal source of errors in positioning is due to the limited knowledge of the centers of the joints and of the part of the legs' length which is not given by the sensors.

We thus use a simpler model with attachment points a_i in the base frame, b_i in the mobile frame, and offset lengths l_i for the ith leg. This gives 42 parameters, 7 for each leg.

The *inverse kinematics* model expresses the length of the ith leg as follows:

$$\|P + Rb_i - a_i\|^2 = (L_i + l_i)^2 \tag{3}$$

In the case of the Gough platform, the exact *forward kinematics* model is much harder to compute and unpracticable for calibration.

For p selected configurations, a measurement device (coordinate measurement machinery, theodolites, ...) provides the position P_k and the orientation R_k. Additionally, the internal sensor provides the leg lengths $L_{i,k}$ for each configuration. As the legs are independent with respect to the calibration problem, we will divide it in 6 subproblems, one for each leg. We may therefore simplify the notation in the following and omit the i index.

For each subproblem, we define a vector of parameters $x = (a, b, l)$, a list of measurements $(M_1, \ldots, M_p)$ with $M_k = (P_k, R_k, L_k)$, and a function f such that:

$$f(x, M_k) = \|P_k + R_k b - a\|^2 - (L_k + l)^2$$

From a theoretical point of view the calibration equations should be:

$$f(x, M_k) = 0, \qquad \text{for } k = 1, \ldots, p. \tag{4}$$

The solution of this system in the 7 kinematic parameters a, b, l is possible if $N = 7$. Due to the noise in the measurements associated with the captors, those equations are approximately valid only for the actual kinematic parameters, and the computed solution of (4) may be significantly different. To reduce this problem we use more equations than the minimum required, $N > 7$.

To solve the over-constrained system, one typically uses optimization (the analytic Jacobian is given in [4]), or linearization [3], which allows to find a least-squares solution. As we shall see, interval analysis and constraint programming techniques offer a useful alternative to those methods.

IV. Proposed Method

We propose to solve the over-constrained system (4) by using interval programming methods.

We assume that the uncertain coefficients M_k of the equation (4) may take all possible values inside an interval of variation denoted by $\mathbf{M}_k$, and combine these intervals into the interval vector $\mathbf{M}$. Our goal is to determine the continuum $\mathcal{S}(\mathbf{M})$ of kinematic parameters x satisfying (4),

$$\mathcal{S}(\mathbf{M}) = \{x | f(x, M_k) = 0 \text{ with } M_k \in \mathbf{M}_k,\ k = 1, \ldots, p\}. \tag{5}$$

To determine the set $\mathcal{S}(\mathbf{M})$, which generally has a complicated shape, is a difficult problem. But it is possible to simplify the problem by computing an enclosure of this set by a box $\mathbf{x}$. If the overestimation is small, $\mathbf{x}$ contains all relevant information about $\mathcal{S}(\mathbf{M})$. A visualization of those sets in the two-dimensional case is given in Figure 2.

In this paper we use a Taylor expension to obtain a linear appoximation of $\mathcal{S}(\mathbf{M})$. (Alternatively, it may be obtained through the semantics of the equations – see [10].) Then we use linear programming to compute the extreme values of this linear approximation. This gives a box $\mathbf{x}$ containing $\mathcal{S}(\mathbf{M})$. Using the quadratic approximation results from [11] it is not difficult to see that if the uncertainties in the M_k is of the order $O(\epsilon)$ then the size of the resulting box is at most

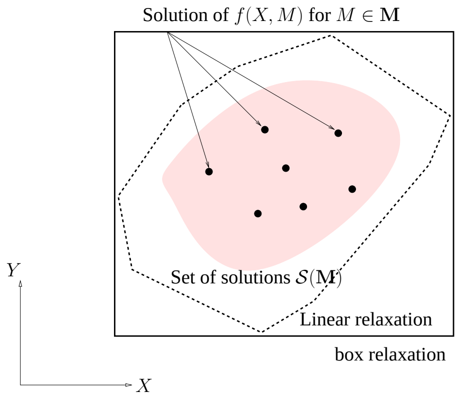

Fig. 2. 2D example of the solution set $\mathcal{S}(\mathbf{M})$

$O(\epsilon^2)$larger than the tightest possible box enclosing $\mathcal{S}(\mathbf{M})$. Thus, in practice, the overestimation has little effect on the quality of the results.

As the linear approximation depends on the initial estimate used for $\mathbf{x}$, it is necessary to use a fixed point algorithm to iteratively sharpen the solution set. The iteration terminates naturally when the bounds of $\mathbf{x}$ no longer improved much, i.e., when the maximal box width does not decrease significantly in some iteration step. If desired, we can get a closer approximation of the solution set $\mathcal{S}(\mathbf{M})$ by bisecting the computed box $\mathbf{x}$ and restart the iterative process with the two resulting boxes as initial estimates.

While we tested several interval methods, we present here only the interval evaluation which provided the sharpest approximation of $\mathcal{S}(\mathbf{M})$. It is particularly adapted to over-constrained systems of equation. However, since there are many more possibilities to explore we think that an improved analysis of the system is possible.

A. Interval Newton Formulation of Implicit Equations

We shall write $F(x, M)$ for the vector valued function with components $F_k(x, M) = f(x, M_k)$. A centered form interval extension of $F(x, M)$ performed in two step gives:

$$\begin{aligned} \mathbf{f}(\mathbf{x}, \mathbf{M}) &= \mathbf{f}(x, \mathbf{M})) + \mathbf{A}(\mathbf{x}, \mathbf{M})(\mathbf{x} - x) \\ &= f(x, M)) + \mathbf{B}(x, \mathbf{M})(\mathbf{M} - M) + \mathbf{A}(\mathbf{x}, \mathbf{M})(\mathbf{x} - x) \end{aligned} \tag{6}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the natural interval extension of the identification Jacobian matrix $\partial f(x, M)/\partial x$ resp. $\partial f(x, M)/\partial M$, computed from explicit expressions, and where x and M are selected in $\mathbf{x}$ and in $\mathbf{M}$ as $x = mid(\mathbf{x})$ and $M = mid(\mathbf{M})$.

We want to determine an enclosure $\mathbf{x}$ for the vectors x such that $F(x, M) = 0$ for some $M \in \mathbf{M}$. Given a trial enclosure $\mathbf{x}_j$ (which is guessed for $j = 0$, we want to use the information in the centered form to reduce the radius of $\mathbf{x}_j$, thus producing a better enclosure x_{j+1}. Newton's method may be extended to the interval case [7], [11], giving a recipe called the *Newton operator* to construct a box $N_j(x_j, \mathbf{x}_j)$, defined as an enclosure of all vectors $x \in \mathbf{x}_j$ satisfying the linear inclusion

$$A(x - \mathbf{x}_j) \in -f(x_j, M) - \mathbf{B}(x_j, \mathbf{M})(\mathbf{M} - M) \quad \text{with } A \in \mathbf{A}. \tag{7}$$

Then the interval Newton method is defined by

$$\mathbf{x}_{j+1} := \mathbf{x}_j \cap N_j(\mathbf{x}_j, x_j). \tag{8}$$

The interval Newton method is terminated if the size of the box is no longer substantially decreased by the interval Newton method, which is tested by a criterion of the form $\|rad(\mathbf{x}_j)\|_1 - \|rad(\mathbf{x}_{j+1})\|_1 < \Delta$.

There are several ways to solve the linear inclusion (7), one of which will be presented in next subsection. For details on properties (convergence, unicity ...) of the interval Newton method, the reader may consult [11]. There it is shown that, in particular, no solution of $F(x, M) = 0$ contained in the initial trial box $\mathbf{x}_0$ can be lost (i.e., lie outside some $\mathbf{x}_j$). As a consequence, if the intersection of $\mathbf{x}_j$ and $N_j(\mathbf{x}_j, x_j)$ is empty for some j then, since $x_{j+1} = 0$ by (8), there was no solution in the initial trial box $\mathbf{x}_0$. Moreover, if some $\mathbf{x}_{j+1}$ is in the interior of $\mathbf{x}_j$ then it is certain that $\mathbf{x}_0$ (and hence all $\mathbf{x}_j$) contains for every $M \in \mathbf{M}$ a solution of $F(x, M) = 0$. This makes the interval Newton method an excellent tool for certified computations.

B. Reformulation as a linear programming problem

We have seen in the previous subsection that the heart of the proposed method is to solve Eq. 7. A correct presentation of that problem is to find the set of solutions

$$\Sigma(\mathbf{A}, \mathbf{b}) = \{x | Ax = b, A \in \mathbf{A}, b \in \mathbf{b}\}, \tag{9}$$

where $\mathbf{A}$ is an interval matrix and $\mathbf{b}$ is an interval vector. To determine $\Sigma(\mathbf{A}, \mathbf{b})$ or only the tightest enclosing box is an NP-hard problem and hence expensive in higher dimensions – the shape of the set can be quite complicated. But it is possible to find an enclosure of $\Sigma(\mathbf{A}, \mathbf{b})$ by an interval vector $\mathbf{x}$ with limited overestimation, provided that the intervals are narrow enough.

Basic interval analysis method suitable for this are preconditioned Gauss elimination and Krawczyk's method (see [7], [11]–[13]). We tested an improved algorithm proposed by Rump [14] based on these methods and implemented in the package INTLAB given under Matlab. The provided tool, while highly useful for square systems of equations, is not adapted to overdetermined problems: though it can solve them, the enclosure is usually inferior to the method proposed in the following, which is based on a reformulation of the problem to a linear programming problem.

The new method consists on two steps: In the first step, we overestimate $\Sigma(\mathbf{A}, \mathbf{b})$ by a convex polyhedron defined by scalar linear inequalities. In the second step, we determine by linear programming (for example the simplex algorithm) the minimal and the maximal value of each component of points in the polyhedron. This provides an enclosure $\mathbf{x}$ of $\Sigma(\mathbf{A}, \mathbf{b})$. Again, results from [11] imply that the overestimation is of higher order, and hence small, if the intervals in the entries of

$\mathbf{A}$ and $\mathbf{b}$ are narrow. To improve the quality of $\mathbf{x}$ the two steps are repeated until no significant improvement is obtained.

For any matrix A (which we choose as the midpoint of $\mathbf{A}$), we can use a Krawczyk-type decomposition

$$\mathbf{A}x - \mathbf{b} = (\mathbf{A} - A)x - \mathbf{b} + Ax$$

to see that any $x \in \Sigma(\mathbf{A}, \mathbf{b})$ satisfies the linear inequalities $Ux \leq u$, where

$$U = \begin{pmatrix} A \\ -A \end{pmatrix}, \quad u = \begin{pmatrix} -inf((\mathbf{A} - A).\mathbf{x} - \mathbf{b}) \\ sup((\mathbf{A} - A).\mathbf{x} - \mathbf{b}) \end{pmatrix}.$$

This observation goes back to [15], and gives for narrow interval coefficients a nearly optimal polyhedral enclosure of $\Sigma(\mathbf{A}, \mathbf{b})$. We therefore call $2n$ ($n = dim(\mathbf{x})$) times a linear programming solver to solve the problems

$$\underline{x}_k = \min\{x_k \mid x_k \mid Ux \leq u\},$$

$$\overline{x}_k = \max\{x_k \mid x_k \mid Ux \leq u\}$$

for $k = 1, \ldots, n$. This algorithm produces an enclosing box for $\Sigma(\mathbf{A}, \mathbf{b})$, and hence can be used to define the Newton operator, and hence the interval Newton iteration discussed in the previous subsection.

V. Application and Simulation

In our simulations all geometric parameters of the Left-Hand robot of INRIA [16] are available. We implemented in Matlab (with the optimization toolkit and the INTLAB package [14]) implicit calibration and its certification using the method just presented. Since our calibration method decouples the problem into 6 independent leg calibrations, we concentrate on the calibration of the first leg. The true values of the kinematic parameters, denoted by x^a, are shown in Table I.

Attachment points [cm]						Leg length offset [cm]
Base platform			Mobile platform			
x	y	z	u	v	w	l
-9.7	9.1	0.0	-3.0	7.3	0.0	52.2496

TABLE I

True kinematic parameters of one leg

The x^a serve to construct a set of 21 configurations by solving the equation (3) for L using 21 randomly generated configurations. In addition to the calculated value of the leg length, the chosen positions and orientations simulate the values obtained by a measurement device without errors. The vector describing the exact measurement is denoted by $M^a = [M_1^a, \ldots, M_{21}^a]$.

The above values x^a are perturbed and denoted by x^r to simulate an initial estimation of the kinematic parameters given, for instance, by the robot constructor. The amplitude of the uniformly distributed perturbation is equal to $+/-0.1$ cm. For the proposed certification method an initial interval vector $\mathbf{x}_0$ is done as $\mathbf{x}_0 = [x^r - 0.1, x^r + 0.1]$.

When all measured quantities are exact:

- the least-square algorithm converges accurately to x^a,
- the certification algorithm converges to an interval vector $\mathbf{x}^a = [x^a - 10^{-8}, x^a + 10^{-8}]$. Note that 10^{-8} is the Δ given in Section IV-A used to terminate the Newton scheme.

We conclude that both methods provide the exact kinematic parameters when no errors are associated with measurement.

Now we simulate uniformly distributed noises associated with measurement devices. The amplitude of the errors are $\epsilon_P = +/-5$ μm for position measurement, $\epsilon_L = +/-5$ μm for leg length measurement. The orientation is modeled by a normalized vector and an angle. The error on the vector direction is equal to $\epsilon_v = +/-5$ μm and, on the angle, it is equal to $\epsilon_a = +/-10^{-3}$ degree. These simulations permit to obtain a realistic measurement vector M^r. Now the interval vector $\mathbf{M}^r = [M^r - \epsilon, M^r + \epsilon]$ contain the true measurement M^a. Note that ϵ is done as a function of $\epsilon_P, \epsilon_L, \epsilon_v, \epsilon_a$. The error ϵ_q may be easily deduced from ϵ_v and ϵ_a to model the error associated with the 3 Rodrigues parameters.

We apply our proposed algorithm to reduce the width of the initial estimation $\mathbf{x}_0$. We obtain $\mathbf{x}_1$ -see Table II, III and IV.

	Base attachment points [cm]					
	x		y		z	
	mid	rad	mid	rad	mid	rad
$\mathbf{x}_0$	-9.6609	0.1000	9.0442	0.1000	0.0251	0.1000
$\mathbf{x}_1$	-9.6825	0.0784	9.0988	0.0455	0.0041	0.0791
$\mathbf{x}_2$	-9.6842	0.0767	9.1011	0.0433	0.0038	0.0788
$\mathbf{x}_s$	-9.6904	0.0735	9.0978	0.0205	0.0210	0.1895

TABLE II

Comparison of the result obtained by interval method vs. least-square method

	Mobile attachment points [cm]					
	u		v		w	
	mid	rad	mid	rad	mid	rad
$\mathbf{x}_0$	-3.0468	0.1000	7.2122	0.1000	-0.0334	0.1000
$\mathbf{x}_1$	-3.0055	0.0587	7.2937	0.0187	-0.0127	0.0793
$\mathbf{x}_2$	-3.0026	0.0558	7.2946	0.0177	-0.0119	0.0786
$\mathbf{x}_s$	-2.9943	0.0562	7.3002	0.0071	-0.0080	0.0726

TABLE III

Comparison of the result obtained by interval method vs. least-square method

We compare this result to the classical least-square method (a Levenberg-Marquardt algorithm provided by Matlab). To do this, we choose randomly 1000 measurement vectors $[P, R, L]$ inside $\mathbf{M}^r \cap [M^a - \epsilon, M^a + \epsilon]$ i.e. that guarantees that the measurement data are inside the range certificated by the interval method and not at a distance greater than ϵ to the exact measurement M^a. We obtain 1000 solutions to the implicit calibration problem. The Figure 3 presents these observations and compares them to the element of the interval vector $\mathbf{x}_1$ which corresponds to the offset of the leg length.

Method	Offset [cm] l	
	mid	rad
$\mathbf{x}_0$	52.3376	0.1000
$\mathbf{x}_1$	52.3164	0.0789
$\mathbf{x}_2$	52.3161	0.0787
$\mathbf{x}_s$	52.2366	0.1597

TABLE IV

COMPARISON OF THE RESULT OBTAIN THOUGHT INTERVAL METHOD VS LEAST-SQUARE METHOD

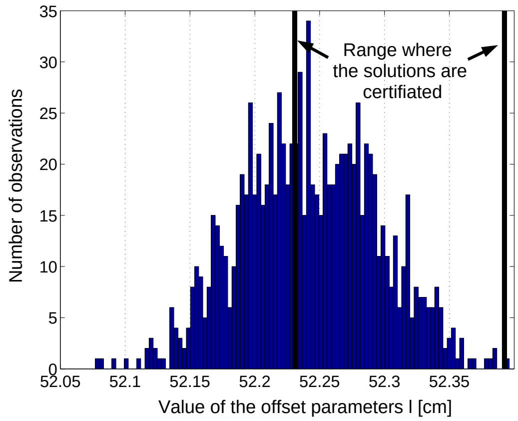

Fig. 3. Observation of 1000 solutions of the offset of the leg length obtain for 1000 set of measurement data chosen inside a possible range of variation

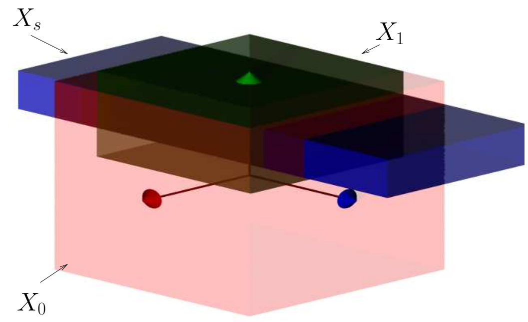

Fig. 4. Visual comparison between $\mathbf{x}_0$ $\mathbf{x}_1$ and $\mathbf{x}_s$ for mobile kinematic parameters x, y, z

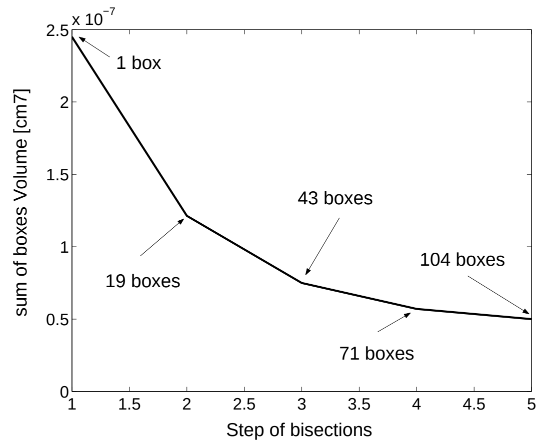

Fig. 5. Volume of the boxes at each step of the bisection process

The maximum and the minimal values of each components of the 1000 observations permit to construct an interval vector, denoted by $\mathbf{x}_s$, where all least-square solutions are localized. The Tables II, III and IV compare $\mathbf{x}_0$, $\mathbf{x}_1$ and $\mathbf{x}_s$. Visually, the Figure 4 presents this comparison for the base attachment point (Note that the frame is center in the middle of $\mathbf{x}_0$).

For at least 2 kinematic parameters (z and l), the radius of their components in $\mathbf{x}_s$ is greater than their equivalent in $\mathbf{x}_1$. Then, some solutions provided by the least square method (their well convergence have been checked) are outside the certified enclosure of the exact set of solution provided the interval method. We may conclude that those special points are not correct with respect to the noise associated with measurement. Their certification is not possible.

To improve our result, a possibility is to bisect each component of $\mathbf{x}_1$ and process the proposed algorithm on each of the boxes obtained. Many rules for bisection have been tested. We choose to present the case where the initial box $\mathbf{x}_1$ is split into two parts, 5% away from its inferior. At each bisection step we test 128 boxes; many of these are eliminated by simple evaluation using Equation 6 or the proposed algorithm. The initial box for the next step of bisection is the largest box obtained in the previous step. This process is repeated for 5% away from the superior limits the largest box. This ensures that the boundary of $\mathbf{x}_1$ is filtered with priority.

After 4 steps (on superior and inferior bound), the set of calibration solutions is described by the union of 104 boxes. Figure 5 shows that the total volume of these boxes decreases to a limit. The area of the solution is greatly improved. But if we compute the smallest box (denoted by $\mathbf{x}_2$) which contains all the 104 boxes, Tables II, III and IV show that the range of the variables of the improved enclosure $\mathbf{x}_2$ is comparable to $\mathbf{x}_1$. This shows that our enclosure method is indeed close to optimal, and little can be gained by bisection when only the rangesof the solution set, and not its shape, is of interest.

Regarding the results, we may conclude that some possible solutions provided by a least-square method do not satisfy the system of equations 4 for the given range of variation of the measurement data. The properties of interval arithmetic show that ”least-square solution” are not included in the exact set of solutions of the system 4 parameterized by measurements.

VI. CONCLUSION

In this article we presented a method based on interval analysis that provides a numerically certified result to kinematic calibration problem of Gough platform.

Even if some further work may have to be done to improve the interval methods we used, the main contribution of this work is to provide the first certified method for this problem and to show that usual methods may produce unrealistic results.

REFERENCES

[1] D. Daney, "Kinematic calibration of the gough platform," *Robotica*, vol. 21, no. 6, pp. 677–690, 2003.

[2] C. Wampler, J. Hollerbach, and T. Arai, "An implicit loop method for kinematic calibration and its application to closed-chain mechanisms," *IEEE Trans. on Robotics and Automation*, vol. 11, no. 5, pp. 710–724, October 1995.

[3] Z. Geng and L. Haynes, "An effective kinematics calibration method for Stewart platform," in *ISRAM*, Hawaï, 15-17 august 1994, pp. 87–92.

[4] H. Zhuang, J. Yan, and O. Masory, "Calibration of Stewart platforms and other parallel manipulators by minimizing inverse kinematic residuals," *J. of Robotic Systems*, vol. 15, no. 7, pp. 395–405, 1998.

[5] C. Innocenti, "Algorithms for kinematic calibration of fully-parallel manipulators," in *Computational Kinematics*, B. R. J-P. Merlet, Ed. Kluwer, 1995, pp. 241–250.

[6] R. Moore, *Interval Analysis*. Prentice Hall, 1966.

[7] E. Hansen, Ed., *Global optimization using interval analysis*. Marcel Dekker, 1992.

[8] P. Vischer and R. Clavel, "Kinematic calibration of the parallel Delta robot," *Robotica*, vol. 16, no. 2, pp. 207–218, April 1998.

[9] O. Masory, J. Wang, and H. Zhuang, "Kinematic modeling and calibration of a stewart platform." *Advanced Robotics*, vol. 11, pp. 519–539, 1997.

[10] Y. Lebbah, M. Rueher, and C. Michel, "A global filtering algorithm for handling systems of quadratic equations and inequations," *Lecture Notes in Computer Science*, vol. 2470, pp. 109–123, 2002. [Online]. Available: http://link.springer-ny.com/link/service/series/0558/bibs/2470/24700109.htm;http://link.springer-ny.com/link/service/series/0558/papers/2470/24700109.pdf

[11] A. Neumaier, *Interval Methods for Systems of Equations*, ser. Encyclopedia of Mathematics and its Applications. Cambridge, UK: Cambridge University Press, 1990, vol. 37.

[12] R. Krawczyk, "Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerschranken," *Computing*, vol. 4, pp. 187–201, 1969.

[13] A. Neumaier, *Introduction to Numerical Analysis*. Cambridge Univ. Press, Cambridge, 2001.

[14] R. S.M., "Solving algebraic systems with high accuracy," in *A New Approach to Scientific Computation*, U. Kulisch and W. Miranker, Eds. San Diego, Academic Press, 1983, pp. 51–120.

[15] A. Neumaier, "The enclosure of solutions of parameter-dependent systems of equations." in *Reliability in Computing*, R. Moore, Ed. Academic Press, 1988, pp. 269–286.

[16] J.-P. Merlet, *Parallel robots*. Dordrecht, Pays-Bas: Kluwer, 2000.