Sensitivity Analysis of Parallel Manipulators Using an Interval Linearization Method

Mikhael Tannous, Stéphane Caro, Alexandre Goldsztejn

Sensitivity Analysis of Parallel Manipulators using an Interval Linearization Method

Mikhael Tannous

Ecole Centrale de Nantes, 1 rue de la Noë, 44321 Nantes, France

Stéphane Caro[1,*]

CNRS, IRCCyN, UMR CNRS n° 6597, 1 rue de la Noë, 44321 Nantes, France

Alexandre Goldsztejn

Université de Nantes, LINA, UMR CNRS n° 6241, Nantes, France

Abstract

The subject of this paper is about an interval linearization method for the sensitivity analysis of manipulators to variations in their geometric parameters. First, the proposed method is presented. Then, three manipulators are used as illustrative examples: The five-bar mechanism, the 3-RRR planar parallel manipulator and the Orthoglide. The benefits and restrictions of the proposed method are also discussed and appropriate indices are derived to show the efficiency of the method. The obtained results are also compared with the results obtained with frequently used methods. The proposed method is simple to implement and provides verified results in low computational time and thus can be applied to complex robots such as the Orthoglide. In particular, the standard linearization method computes unreliable results near singularities, whereas the proposed interval linearization method automatically detects such situations.

Keywords:
parallel manipulators, sensitivity analysis, interval linearization method

*Corresponding author, Tel: +33 2 40 37 69 68; Fax: +33 2 40 37 69 30

Email addresses: mikhael.tannous@ec-nantes.fr (Mikhael Tannous), stephane.caro@irccyn.ec-nantes.fr (Stéphane Caro), alexandre.goldsztejn@univ-nantes.fr (Alexandre Goldsztejn)

1. Introduction

Many robotic applications require high precision. This precision can not be achieved unless all sources of errors have been identified and integrated in dedicated computational methods. Amongst the well known sources of errors we find manufacturing errors, joint clearances, and backlashes in the actuators. The sensitivity analysis of a manipulator aims at knowing the influence of variations in its geometric parameters and/or actuators on its performance. This information is useful for the evaluation of the pose error of the end-effector and for the tolerance synthesis. The three main methods used for the sensitivity analysis to variations in geometric parameters and joint clearances of serial and parallel robots are: (*i*) the probabilistic methods [1]; (*ii*) the linearization methods [2] and (*iii*) the interval analysis methods [3].

As indicated in [4], science was based in the last centuries on deterministic ideas, which consist in believing that since each phenomenon is due to a determined cause, the behavior can be predicted. However, since it is usually impossible to gather enough information for the behavior to be predictable, probabilistic theories were born. The probabilistic approaches are well known in several engineering fields such as nonlinear dynamics and robotics. Recently, in [5], a probabilistic approach was used in the field of Human-Robot communication. In [1], the sensitivity analysis for a two-link planar manipulator and the Stanford arm was conducted according to a probabilistic method. The authors pointed out that the probabilistic model of the kinematics and dynamics of these robots will be reliable if a sufficient number of experiments is conducted, which means that the results will not be verified. The probabilistic approaches present the advantage of having a competitive computational time while being simple and general. However, they do not provide any verified result.

Other methods for sensitivity analysis do exist such as the *linearization method*, which consists in analyzing the sensitivity of the linearized system. The linearization method is a numerical method that does not provide verified results [6, 7, 8]. In [2, 9, 10, 11], the authors worked on the sensitivity analysis of planar parallel manipulators using the linearization method. This method is suitable for small uncertainties and as long as the robot is not close to singular configurations. The computed sensitivity may be lower than the real one. For instance, the linearization method was used to analyze the

sensitivity of the Orthoglide to variations in its geometric parameters [12].

The sensitivity of parallel manipulators to joint clearances is also an important issue. The great impact that joint clearances can have on the end-effector pose of the manipulator was highlighted in [13]. Furthermore, as mentioned in [14], errors due to joint clearances can not be compensated by calibration contrary to manufacturing and assembling errors. To analyze the effect of joint clearances, a complete mathematical model of the joint should be defined. For instance, a joint can be modeled as a journal bearing [15]. The same model was used in [16] in order to compute the effect of joint clearances. A general approach was proposed in [17] but turns to be time consuming. A general error-prediction model has been recently developed in [18]. Accordingly, two optimization problems were formulated in order to find the maximum positioning and orientation errors of the end-effector of the manipulator due to joint clearances. Again, the previous research works without minimizing their importance are based on algebraic methods that do not provide verified results.

On the contrary, interval analysis based approaches provide verified results while being simpler than their counterparts. In [19] the authors used a Newton-Raphson based interval analysis method combined with a bisection algorithm in order to analyze the sensitivity of simple mechanisms to joint clearances. However, their approach is complicated and time consuming because it requires a bisection algorithm. Moreover, the method has not been applied to complex mechanisms, and the authors did not give enough information about the behavior of their algorithm close to singular configurations.

The standard linearization method turns to be simple to use and provides good results as long as the variations in the parameters are small enough. Therefore, the standard linearization method may provide bad results.

In this paper, we propose an interval linearization method that combines the simplicity of the standard linearization method and the verification of the results. Moreover, the proposed method is simpler than the approach described in [19] and allows us to analyze the sensitivity of complex robots to geometric errors and joint clearances.

The paper is organized as follows. Section 2 introduces the proposed interval linearization method for the sensitivity analysis of parallel manipulators. An efficiency index of the method is also developed with regard to the amount of uncertainties. Section 3 deals with the sensitivity analysis of the five-bar mechanism and makes a comparison between the results obtained with the proposed method and the ones obtained with a standard lineariza-

tion method. Section 4 deals with the sensitivity analysis of the 3-RRR planar parallel manipulator with the proposed interval linearization method. Section 5 deals with the sensitivity analysis of the Orthoglide, a three degrees of freedom translational parallel manipulator, using the proposed interval linearization method and two geometric models of the manipulator.

2. The Interval Linearization for Sensitivity Analysis

2.1. Interval analysis

Intervals are denoted by brackets: Intervals of reals are $[x] = [\underline{x}, \overline{x}]$ with $\underline{x}, \overline{x} \in \mathbb{R}$ and $\underline{x} \leq \overline{x}$. Intervals of vectors (also called boxes) are $[\mathbf{x}] = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]$ with $\underline{\mathbf{x}}, \overline{\mathbf{x}} \in \mathbb{R}^n$ and $\underline{\mathbf{x}} \leq \overline{\mathbf{x}}$ (the inequality being defined component wise); in this case the interval $[x_i] = [\underline{x}_i, \overline{x}_i]$ is the i^{th} component of the interval vector. Interval of matrices are defined similarly by $[\mathbf{A}] = [\underline{\mathbf{A}}, \overline{\mathbf{A}}]$ with $\underline{\mathbf{A}}, \overline{\mathbf{A}} \in \mathbb{R}^{n \times m}$ and $\underline{\mathbf{A}} \leq \overline{\mathbf{A}}$.

Interval analysis extends usual real operations (like scalar addition, multiplication, matrix/vector additions and multiplications) to interval objects. These interval operations are defined and implemented so as to rigorously verify the following containment property: The result of the interval operation contains all possible results of the corresponding real operation for real arguments inside the interval arguments. For example, $[x][y]+[x] \supseteq \{xy+x : x \in [x], y \in [y]\}$. Matrix and vector operations are extended in the same way, e.g. $[\mathbf{A}][\mathbf{x}] \supseteq \{\mathbf{A}\mathbf{x} : \mathbf{A} \in [\mathbf{A}], \mathbf{x} \in [\mathbf{x}]\}$, or $[\mathbf{A}]+[\mathbf{B}] \supseteq \{\mathbf{A}+\mathbf{B} : \mathbf{A} \in [\mathbf{A}], \mathbf{B} \in [\mathbf{B}]\}$. See [20] and references therein for more details. These interval operations are implemented in many environments, we used the Matlab library `Intlab` [21].

The interval evaluation of an expression generally provides a pessimistic enclosure of the range, which is a central issue in interval analysis. The so called mean-value extension can provide a tighter enclosure of the range, in particular when interval arguments have small widths. It uses interval evaluations of the function derivatives in order to enclose the function's graph within an interval linearization, whose range contains the range of the original function. More formally,

$$\{f(\mathbf{x}) : \mathbf{x} \in [\mathbf{x}]\} \subseteq f(\tilde{\mathbf{x}}) + \sum_i [f_{x_i}]([\mathbf{x}])\,([x_i] - \tilde{x}_i), \tag{1}$$

where $\tilde{\mathbf{x}} \in [\mathbf{x}]$, usually chosen as its midpoint, and $[f_{x_i}]$ is an interval extension of the partial derivative of f with respect to x_i, which can be computed either as the interval evaluation of its derivatives or by using interval automatic differentiation.

2.2. Problem to be solved

We are given a function $\mathbf{f} : \mathbb{R}^p \times \mathbb{R}^n \to \mathbb{R}^n$, an interval vector $[\mathbf{a}]$ for parameters and an approximate nominal solution $(\tilde{\mathbf{a}}, \tilde{\mathbf{x}}) \in \mathbb{R}^p \times \mathbb{R}^n$ such that $\mathbf{f}(\tilde{\mathbf{a}}, \tilde{\mathbf{x}}) \approx \mathbf{0}$[1]. Provided that the parameter domain $[\mathbf{a}]$ is small enough (which is verified in practice and in the experimentations presented in this paper), the solution set

$$\Sigma(\mathbf{f}, [\mathbf{a}]) := \{\mathbf{x} \in \mathbb{R}^n : \exists \mathbf{a} \in [\mathbf{a}], \mathbf{f}(\mathbf{a}, \mathbf{x}) = \mathbf{0}\} \tag{2}$$

can be split in connected components that correspond to the different assembly modes of the robot. We are interested in enclosing the connected component of $\Sigma(\mathbf{f}, [\mathbf{a}])$, which contains the nominal solution. This connected component actually corresponds to the variations in the nominal solution when parameters vary in the domain $[\mathbf{a}]$ and plays a key role in the analysis of the robot sensitivity to geometric parameters. Let $\Sigma(\mathbf{f}, [\mathbf{a}], \tilde{\mathbf{x}})$ denote this connected component.

We aim at computing a box that contains $\Sigma(\mathbf{f}, [\mathbf{a}], \tilde{\mathbf{x}})$. Such a box is of great interest for sensitivity analysis, since it quantifies rigorously the sensitivity of the pose of the manipulator with respect to variations in the parameters for a given nominal pose. The smallest box is called its *interval hull*, denoted by $[\mathbf{x}^{Hull}]$, and is depicted in Figure 1. However, $[\mathbf{x}^{Hull}]$ cannot be computed in general so an outer approximation $[\mathbf{x}^{Out}]$ of it will actually be computed.

The smaller the distance between $[\mathbf{x}^{Hull}]$ and $[\mathbf{x}^{Out}]$, the better the computed enclosure. To formalize this overestimation and analyze it in the experiments, we define the error for the variable x_i of the pose vector by:

$$\epsilon_{x_i} = 1 - \frac{\text{wid}[\mathbf{x}_i^{Hull}]}{\text{wid}[\mathbf{x}_i^{Out}]}, \tag{3}$$

wid denoting the width of the corresponding box. Since $[\mathbf{x}^{Out}] \supseteq [\mathbf{x}^{Hull}]$ we have $\epsilon_{x_i} \geq 0$. Moreover, ϵ_{x_i} is null if and only if $[\mathbf{x}_i^{Out}] = [\mathbf{x}_i^{Hull}]$. Moreover,

[1] A manipulator is usually modeled by $\mathbf{f}(\mathbf{a}, \mathbf{u}, \mathbf{x}) = 0$ where $\mathbf{a}$ are geometric parameters, $\mathbf{u}$ are commands and $\mathbf{x}$ are positions. Since $\mathbf{u}$ is fixed in the scope of our study, we include it into the geometric parameters in order to also deal with uncertainties in the command. Note that $\mathbf{x}$ and $\mathbf{u}$ play symmetric roles, therefore the dual problem consisting in enclosing the set of commands necessary to keep a pose constant in spite of variations in geometric parameters could be solved similarly.

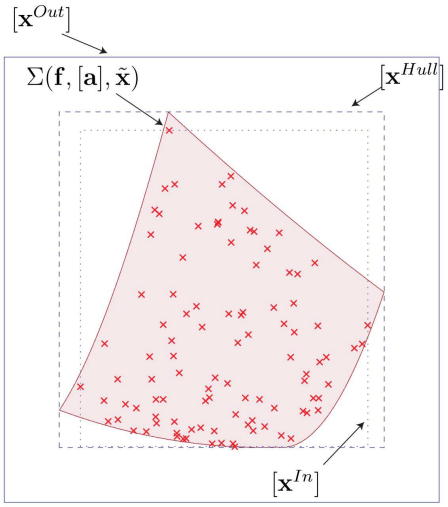


Figure 1: The set $\Sigma(\mathbf{f}, [\mathbf{a}], \tilde{\mathbf{x}})$ is depicted in light; the box in dashed line is its interval hull $[\mathbf{x}^{Hull}]$; the box in full line $[\mathbf{x}^{Out}]$ is an outer approximation of the latter two boxes. The dotted box $[\mathbf{x}^{In}]$ is computed by solving the system for 50 randomly chosen parameter values inside $[\mathbf{a}]$ (solutions are illustrated with crosses) and considering the minimum and maximum values of each variable, it is included inside $[\mathbf{x}^{Hull}]$.

the closer ϵ_{x_i} to zero, the better the computed outer approximation $[\mathbf{x}^{Out}]$. Note that since $[\mathbf{x}^{Hull}]$ cannot be computed, ϵ_{x_i} cannot be computed exactly neither. Instead of using $[\mathbf{x}^{Hull}]$ we use an inner approximation of it $[\mathbf{x}^{In}] \subseteq [\mathbf{x}^{Hull}]$ depicted in Figure 1, which is computed by solving several systems for different values of parameters inside $[\mathbf{a}]$. Such an inner approximation leads to an upper bound of ϵ_{x_i}.

2.3. The Krawczyk method

Our aim is to enclose the variations in the nominal solution $\mathbf{x} = \tilde{\mathbf{x}} + \delta\mathbf{x}$ when the nominal parameters change to $\mathbf{a} = \tilde{\mathbf{a}} + \delta\mathbf{a}$. To this end, the linearization method linearizes the function in the neighborhood of the nominal solution, which leads to the following relationship between $\mathbf{x}$ and $\mathbf{a}$:

$$\mathbf{f}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}}) + \mathbf{f}_{\mathbf{x}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}})(\mathbf{x} - \tilde{\mathbf{x}}) + \mathbf{f}_{\mathbf{a}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}})(\mathbf{a} - \tilde{\mathbf{a}}) = 0, \tag{4}$$

where $\mathbf{f}_{\mathbf{x}}$ and $\mathbf{f}_{\mathbf{a}}$ are the Jacobian of $\mathbf{f}$ with respect to $\mathbf{x}$ and $\mathbf{a}$ respectively. This is a linear system with respect to $\mathbf{x} - \tilde{\mathbf{x}}$, and the linearization method usually explicitly computes its solution as $\mathbf{x} = \tilde{\mathbf{x}} - \mathbf{f}_{\mathbf{x}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}})^{-1}\mathbf{f}_{\mathbf{a}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}})(\mathbf{a} - \tilde{\mathbf{a}})$ (with the assumption that $\mathbf{f}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}}) = 0$, i.e., the evaluation of the function

at the approximate nominal solution is usually neglected). This leads to accurate approximations of the variations in $\mathbf{x}$ provided that uncertainties in parameters are small enough and the matrix $\mathbf{f}_{\mathbf{x}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}})$ is well enough conditioned. These necessary conditions are difficult to quantify and verify and therefore the result of the linearization method is not reliable in general.

In the context of interval computations, the inverse of an interval matrix is usually avoided and linear systems are usually solved using iterative methods. We build such an iterative scheme following the idea of the interval Krawczyk[2] method:

$$\mathbf{x} = \tilde{\mathbf{x}} - \mathbf{C}\,\mathbf{f}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}}) - (\mathbf{C}\,\mathbf{f}_{\mathbf{x}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}}) - \mathbf{I})(\mathbf{x} - \tilde{\mathbf{x}}) - \mathbf{C}\,\mathbf{f}_{\mathbf{a}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}})(\mathbf{a} - \tilde{\mathbf{a}}), \quad (5)$$

where the preconditioning matrix $\mathbf{C}$ is any non-singular matrix. This fixed point equation is obtained by first left-multiplying Eq. (4) by $\mathbf{C}$ and adding and subtracting $(\mathbf{x}-\tilde{\mathbf{x}})$ (the subtracted $(\mathbf{x}-\tilde{\mathbf{x}})$ being collected with its original occurrence, giving rise to the common factor $(\mathbf{C}\,\mathbf{f}_{\mathbf{x}}(\tilde{\mathbf{x}}, \tilde{\mathbf{a}}) - \mathbf{I})$). Thus Eq. (4) and Eq. (5) have the same solutions, which can be computed by applying iteratively the fixed point equation (5) to an initial guessed solution.

Now, we propose an interval linearization procedure by using interval evaluations of functions and derivatives. Equation 5 is reformulated as:

$$[\mathbf{x}] = \tilde{\mathbf{x}} - \mathbf{C}\,[\mathbf{f}](\tilde{\mathbf{x}}, \tilde{\mathbf{a}}) - (\mathbf{C}\,[\mathbf{f}_{\mathbf{x}}]([\mathbf{x}], [\mathbf{a}]) - \mathbf{I})([\mathbf{x}] - \tilde{\mathbf{x}}) - \big(\mathbf{C}\,[\mathbf{f}_{\mathbf{a}}](\tilde{\mathbf{x}}, [\mathbf{a}])\big)([\mathbf{a}] - \tilde{\mathbf{a}}), \quad (6)$$

where $[\mathbf{f}]$, $[\mathbf{f}_{\mathbf{x}}]$ and $[\mathbf{f}_{\mathbf{a}}]$ are interval extensions of $\mathbf{f}$, $\mathbf{f}_{\mathbf{x}}$ and $\mathbf{f}_{\mathbf{a}}$ respectively (these expressions can be evaluated for interval arguments using interval arithmetic). We denote the right hand side of Eq. (6) by $[\mathbf{K}]([\mathbf{x}])$. Using the theory of [23], it can be proved that:

- (**A**) If $[\mathbf{x}]$ satisfies $[\mathbf{x}] = [\mathbf{K}]([\mathbf{x}])$ then $\forall \mathbf{a} \in [\mathbf{a}], \exists \mathbf{x} \in [\mathbf{x}], \mathbf{f}(\mathbf{x}, \mathbf{a}) = 0$.
- (**B**) If $[\mathbf{x}]$ satisfies $[\mathbf{x}] = [\mathbf{K}]([\mathbf{x}])$ then the square interval matrix $[\mathbf{f}_{\mathbf{x}}]([\mathbf{x}], [\mathbf{a}])$ is strongly regular, which implies according to the implicit function theorem that $\mathbf{x}$ changes continuously with $\mathbf{a}$.

[2]Other operators such as the Hansen-Sengupta operator exist and would lead to similar results (see e.g. [22] for a comparison between the Krawczyk and Hansen-Sengupta operators in the context of sensitivity analysis). We have chosen the Krawczyk operator because it is straightforward to implement using e.g. the Matlab library `Intlab` [21].

Algorithm 1: Parametric Krawczyk iteration interleaved with inflation.

Input: $\mathbf{f} : \mathbb{R}^p \times \mathbb{R}^n \to \mathbb{R}^n$, $[\mathbf{a}] \in \mathbb{IR}^p$, $\mathbf{x}^0 \in \mathbb{R}^n$
Output: $[\mathbf{x}^{Out}] \in \mathbb{IR}^n$

1 $k_{\max} \leftarrow 10$; `/* Maximum number of iterations */`
2 $\delta \leftarrow 1.01$; `/* Inter-step inflation ratio */`
3 success $\leftarrow$ **false**;
4 $[\mathbf{x}] \leftarrow \mathbf{x}^0$;
5 **repeat**
6 $\quad [\mathbf{x}'] \leftarrow \text{mid}[\mathbf{x}] + \delta\ ([\mathbf{x}] - \text{mid}[\mathbf{x}])$;
7 $\quad [\mathbf{x}] \leftarrow [\mathbf{K}]([\mathbf{x}'])$;
8 $\quad$ **if** ($[\mathbf{x}] \subseteq \text{int}[\mathbf{x}']\ \wedge \tilde{\mathbf{x}} \in [\mathbf{x}]$) **then** success $\leftarrow$ **true**;
9 $\quad k \leftarrow k + 1$;
10 **until** ($k > k_{\max}$);
11 **if** (success) **then** $[\mathbf{x}^{Out}] \leftarrow [\mathbf{x}]$ **else** $[\mathbf{x}^{Out}] \leftarrow [-\infty, \infty]^n$;
12 **return** ($[\mathbf{x}^{Out}]$);

Therefore, provided that $\tilde{\mathbf{x}} \in [\mathbf{x}]$ we have $[\mathbf{x}] \supseteq \Sigma(\mathbf{f}, [\mathbf{a}], \tilde{\mathbf{x}})$. Note that Property (B) also implies that $[\mathbf{x}]$ does not contain any parallel singularity. However, the exact solution of $[\mathbf{x}] = [\mathbf{K}]([\mathbf{x}])$ cannot be computed in practice, because this iteration does not converge in general in a finite number of steps. In order to overcome this issue, Algorithm 1 interleaves the fixed point iteration $[\mathbf{x}] \leftarrow [\mathbf{K}]([\mathbf{x}])$ with an inflation process at Line 6, which allows obtaining the inclusion $[\mathbf{K}]([\mathbf{x}']) \subseteq \text{int}[\mathbf{x}']$ after a finite (usually very small) number of steps. This latter strict inclusion also entails properties (A) and (B) (see Appendix Appendix A) and the returned box $[\mathbf{x}^{Out}]$ contains $\Sigma(\mathbf{f}, [\mathbf{a}], \tilde{\mathbf{x}})$. If the inclusion $[\mathbf{K}]([\mathbf{x}']) \subseteq \text{int}[\mathbf{x}']$ is not satisfied after $k_{\max}$ iterations[3], then the output $[\mathbf{x}^{Out}] = [-\infty, \infty]^n$ will be interpreted as a failure to enclose $\Sigma(\mathbf{f}, [\mathbf{a}], \tilde{\mathbf{x}})$. In addition to its finite termination, Algorithm 1 presents the advantages to be easy to implement (e.g. it is straightforwardly implemented in Matlab using `Intlab` [21]) and to have a low computational complexity. The method we propose in Algorithm 1 can be summarized as the application of a parametric version of the Krawczyk interval operator

[3]The fixed point iteration may converge slowly or diverge because of large uncertainties or because $[\mathbf{f_x}]([\mathbf{x}], [\mathbf{a}])$ contains some singularities.

interleaved with an inflation process, and we call it for short the *Krawczyk method* in the rest of the paper (and $[\mathbf{x}^{Out}]$ the *Krawczyk box*).

Although the theory is correct for any non-singular preconditioning matrix $\mathbf{C}$, a bad choice can prevent the fixed point iteration from converging. One typical preconditioning matrix used in the context of interval analysis is the midpoint inverse preconditioning, i.e. $\mathbf{C} \approx \left(\text{mid}[\mathbf{f_x}]([\mathbf{x}],[\mathbf{a}])\right)^{-1}$ where an approximate inverse can be used. This preconditioning matrix allows obtaining good convergence properties and is easy to compute for low dimensional systems. It is used for the illustrative examples presented in the next sections.

Related Work.. `Intlab` [21] provides the function `verifynlss`, which is similar to Algorithm 1. The main difference between them consists in the evaluation of $[\mathbf{f}](\tilde{\mathbf{x}},[\mathbf{a}])$: The proposed method uses a mean-value interval extension, while `verifynlss` uses a simple natural interval extension. For small uncertainties, the former is known to provide sharper results, although the former can occasionally be better. Consider for example the following simple system of two equations, two unknowns and three parameters:

$$(x_1 + a_1)^2 + (x_2 - a_2)^2 = a_3^2 \quad (7)$$
$$(x_1 - a_1)^2 + (x_2 + a_2)^2 = a_3^2, \quad (8)$$

where parameters domains are set to $[a_1] = 0.5 \pm 0.025$, $[a_2] = \pm 0.025$ and $[a_3] = 1.0 \pm 0.025$. Starting from the approximate solution $\mathbf{x} = [x_1, x_2]^T = [0.01, 0.85]^T$, the proposed method computes an enclosure of the solution set while `verifynlss` diverges. Nevertheless, since the interval natural extension can happen to compute sharper enclosures than the interval mean-value extension, it is natural to use the following hybrid iteration:

$$[\mathbf{y}] = \left(\mathbf{C}\,[\mathbf{f}](\tilde{\mathbf{x}},\tilde{\mathbf{a}}) + (\mathbf{C}\,[\mathbf{f_a}](\tilde{\mathbf{x}},[\mathbf{a}]))([\mathbf{a}] - \tilde{\mathbf{a}}) \right) \cap \left(\mathbf{C}\,[\mathbf{f}](\tilde{\mathbf{x}},[\mathbf{a}]) \right) \quad (9)$$
$$[\mathbf{x}] = \tilde{\mathbf{x}} - [\mathbf{y}] - (\mathbf{C}\,[\mathbf{f_x}]([\mathbf{x}],[\mathbf{a}]) - \mathbf{I})([\mathbf{x}] - \tilde{\mathbf{x}}). \quad (10)$$

This hybrid iteration is named *the Krawczyk method* and is used in the rest of the paper.

3. Case study 1: Five-bar mechanism

3.1. Manipulator architecture

The five-bar mechanism consists of a closed kinematic chain composed of five revolute joints and five links of lengths l_0, l_1, l_2, l_3 and l_4, as shown in

Figure 2. The two revolute joints attached to link l_0 are actuated. Point P denotes the end of the manipulator, x_p and y_p are its Cartesian coordinates.

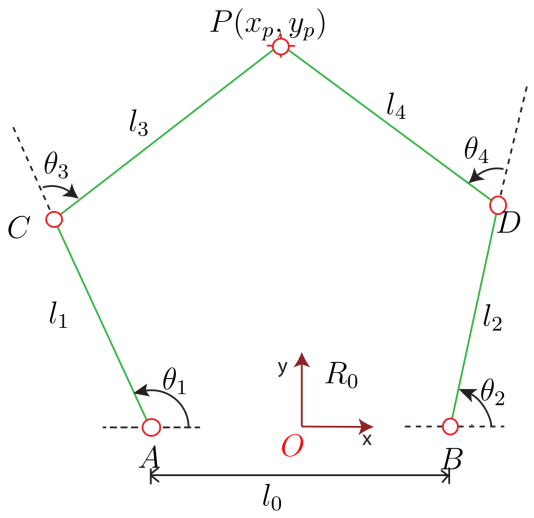


Figure 2: Five-bar mechanism

3.2. Sensitivity analysis of the five-bar mechanism

Since the main sources of errors in the five-bar mechanism are the uncertainties in the leg lengths and the joint clearances, the Cartesian coordinates of point P are the components of the variable vector $\mathbf{x}$ of the problem at hand, while its leg lengths are its parameters, namely, the components of vector $\mathbf{a}$. Here, there is no variation in l_0 as the base of the manipulator is supposed to be rigid. As a consequence, only parameters l_1, l_2, l_3 and l_4 are considered.

Thus, the five-bar mechanism is described with the following system of equations [24]:

$$\begin{aligned} \left(x_p + \frac{l_0}{2} - l_1 \cos\theta_1\right)^2 + (y_p - l_1 \sin\theta_1)^2 - l_3^2 &= 0 \\ \left(x_p - \frac{l_0}{2} - l_2 \cos\theta_2\right)^2 + (y_p - l_2 \sin\theta_2)^2 - l_4^2 &= 0 \end{aligned} \tag{11}$$

that takes the form $\mathbf{f}(\mathbf{a}, \mathbf{x}) = 0$.

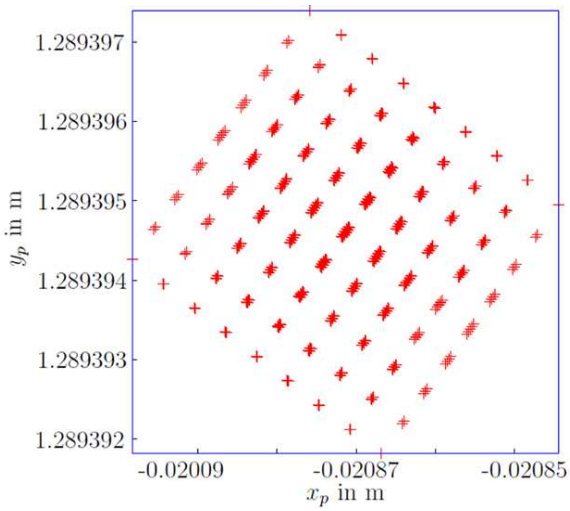
We consider the following nominal values for the geometric parameters of the five-bar mechanism under study: $l_0 = 3$ m, $l_1 = 1$ m, $l_2 = 1$ m, $l_3 = 1$ m and $l_4 = 1$ m. Let the uncertainties in the links l_i be $\dfrac{\Delta l_i}{l_i} = 10^{-6}$ for $i = 1, \ldots, 4$.

θ_1 and θ_2 are the actuated joint angles, whereas θ_3 and θ_4 are the passive joint angles. Let point P be above line CD and θ_1 and θ_2 be equal to $\pi/6$ rad and $3\pi/4$ rad, respectively. Note that such a configuration is not singular and the mechanism is in a given assembly mode.

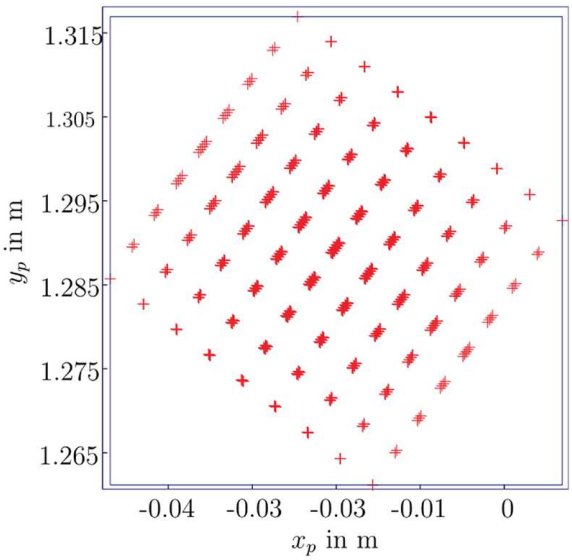
Figure 3 illustrates the possible positioning errors of point P in this configuration. The crosses represent the possible positions of point P of the end-effector due to geometric errors in the legs. The crosses are obtained by evaluating the position of the end-effector by means of the direct kinematic model for a finite set of parameter values inside $[\mathbf{a}] = [\underline{a}, \overline{a}]$. The values to select in each interval $[\mathbf{a}]$, is the lowest value $\underline{a}$ and the highest value $\overline{a}$. This will result in 2^n different possible positions, where n represents the number of parameters. It is obvious that this method cannot provide a verified enclosure. The set of values chosen in each interval cannot guarantee that the calculated pose of the end-effector is a minimum or a maximum. Otherwise, there is no need of proposing an interval linearization method. It is also time consuming, especially for a high number of parameters.

Another drawback of such a technique is the need of the direct kinematic model of the manipulator under study, which may be complicated for some manipulators. The box containing all of the crosses is an inner approximation $[\mathbf{x}^{In}]$ (see Sec 2.2) of the optimal box $[\mathbf{x}^{Out}]$ containing all possible positions of the end-effector due to all combinations of parameter errors. This inner approximation $[\mathbf{x}^{In}]$ and the Krawczyk box are represented with two boxes, the largest one being the Krawczyk box shown in Fig. 3. The two boxes are similar in Fig. 3.2. As a matter of fact, the overestimation of the method is $\epsilon_x = 0.00029\%$, and $\epsilon_y = 0.00029\%$, ϵ_x and ϵ_y being defined with Eq. (3).

However, the higher the uncertainties in the parameters, the less accurate the method. Table 1 represents the evolution of the overestimation of the method with respect to the uncertainties in the leg lengths. Figure 3.2 shows that the overestimation of the Krawczyk method becomes too high for relative errors larger than 10^{-2}. This is the price to pay for a rigorous interval linearization. However, manufacturing errors are usually quite smaller, and the Krawczyk method overestimation is then accurate and acceptable.



(a) $\Delta l_i / l_i = 10^{-6}$



(b) $\Delta l_i / l_i = 10^{-2}$

Figure 3: Accuracy of the Krawczyk method

Table 1: Efficiency of the Krawczyk operator

	Relative Uncertainties $\Delta l_i/l_i$, $i = 1, \ldots, 4$				
	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}
ϵ_x	0.00029 %	0.0029 %	0.0296 %	0.296 %	2.939 %
ϵ_y	0.00029 %	0.0029 %	0.0296 %	0.295 %	2.898 %

To analyze more carefully the error due to the Krawczyk method (given in Table 1), it can be plotted in a logarithmic scale with respect to the uncertainties in the leg lengths as illustrated in Fig. 4. It turns out that the error due to the Krawczyk operator is, in the case of the five-bar mechanism, a linear function of the uncertainties in the leg lengths. The lower the uncertainties, the more accurate the Krawczyk operator.

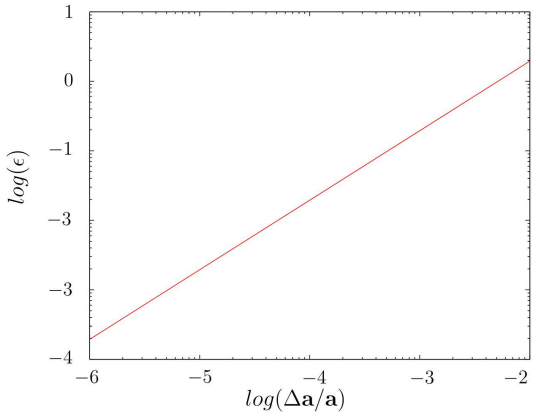


Figure 4: Efficiency of the Krawczyk method

Let $\Delta\mathbf{a}$ denote the uncertainties in the parameters and ϵ be the error due to the method. The linear relation shown in Fig. 4 leads to the following error model:

$$\epsilon \approx \left(\frac{\Delta\mathbf{a}}{\mathbf{a}}\right)^{0.991}. \tag{12}$$

It is obvious that the method is very accurate for low uncertainties.

3.3. Isocontours of the position error of the end-effector

The standard linearization methods are so far the most common methods used for sensitivity analysis of mechanisms. It consists in finding the relationship between the pose error of the end-effector $\delta\mathbf{p}$ and the uncertainties $\delta\mathbf{a}$ in the geometric parameters:

$$\delta\mathbf{p} = \mathbf{J}_s \delta\mathbf{a}. \tag{13}$$

Matrix $\mathbf{J}_s$ characterizes the mapping between the end-effector pose error $\delta\mathbf{p}$ and the variations in the geometric parameters.

The maximum position error Δp_{max} is defined as follows:

$$\Delta p_{max} = \sqrt{\Delta x_p^2 + \Delta y_p^2}, \tag{14}$$

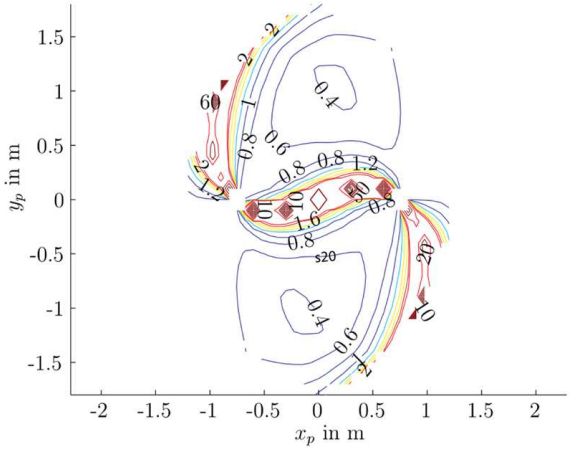
$\Delta x_p = \overline{x_p} - \underline{x_p}$, $\underline{x_p}$ and $\overline{x_p}$ being the lower and upper bounds of $[x_p]$, respectively. $[x_p]$ is the box containing the position of the end-effector along the x-axis. Likewise, $\Delta y_p = \overline{y_p} - \underline{y_p}$, $\underline{y_p}$ and $\overline{y_p}$ are the lower and upper bounds of $[y_p]$, respectively. $[y_p]$ is the box containing the position of the end-effector along the y-axis.

Figure 5 shows the isocontours of the maximum position error Δp_{max} of the end-effector of the five-bar mechanism under study and for uncertainties in the parameters equal to $\Delta l_i/li = 10^{-4}$ for $i = 1$, 2, 3 and 4. Figure 3.3 represents the isocontours obtained with the Krawczyk operator, while Fig. 3.3 represents the isocontours plotted by means of the standard linearization method (all values are multiplied by 1000 for a better clarity).

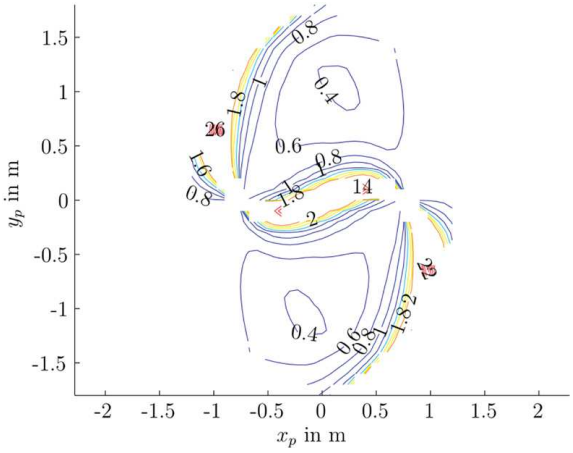
However, since the chosen working mode of the mechanism contains parallel singularities, and since the sensitivity of a mechanism is highly affected by the singular configurations, it is important to detect the location of the singular configurations for a given working mode and to compare the sensitivity of the mechanism in singular and non-singular configurations using the two different methods.

Figure 6 represents the singularities of the five-bar mechanism for the chosen working mode in its Cartesian workspace.

Figure 7 illustrates a deeper comparison between the values of the sensitivity obtained with the standard linearization and the Krawczyk method by checking the sensitivity at different points along segment AF (see Figure 6) with both methods.



(a) The linearization method



(b) The Krawczyk method

Figure 5: Isocontours of the maximum position error of the end-effector throughout the manipulator Cartesian workspace

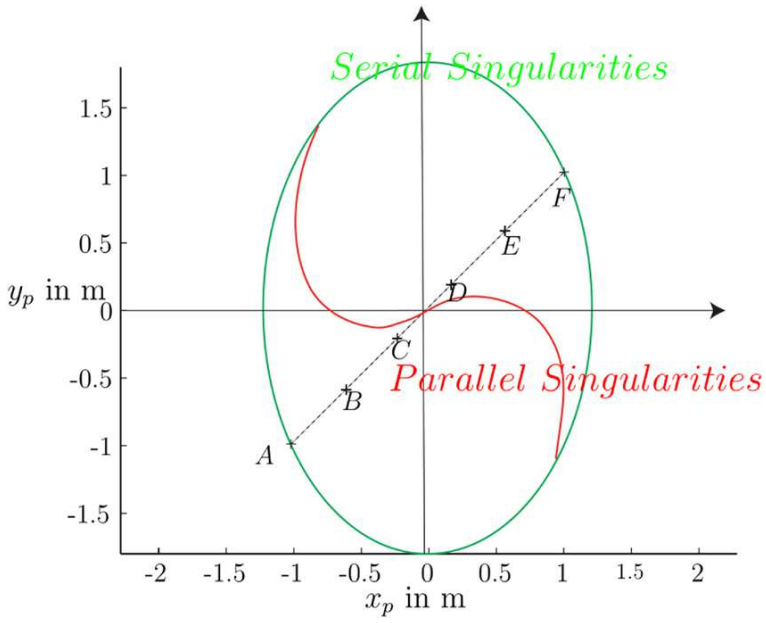


Figure 6: Singularities for the selected working mode

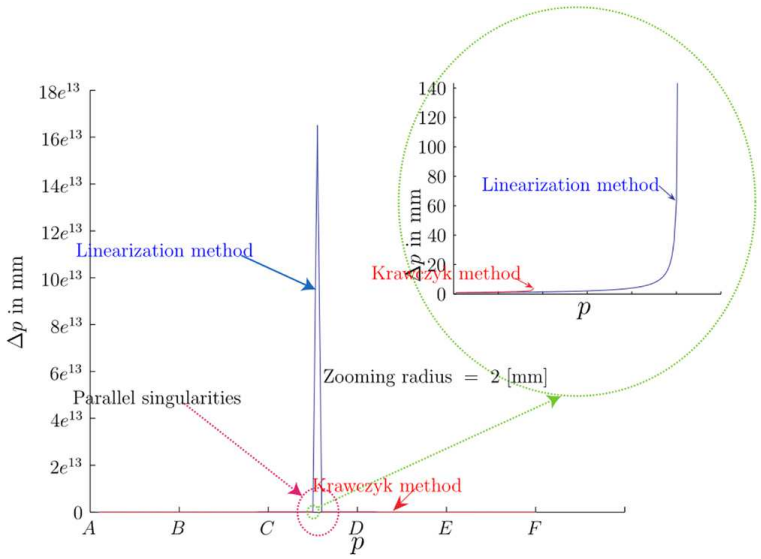


Figure 7: Comparison between the linearization and Krawczyk method along segment AF

For non-singular configurations, the values obtained with the Krawczyk operator are slightly higher than those obtained with the standard linearization method: The latter is based on a linearization of the mathematical model, while the Krawczyk method considers non linearities inside the interval linearization process and provides verified results.

However, the Krawczyk method stops in the vicinity of parallel singularities. The Krawczyk method and the standard linearization method do not diverge in the neighbourhood of serial singularities. Both the methods provide good results close to serial singularities as explained in Sec 2.3.

Nevertheless, the Krawczyk method stops working while the standard linearization method still works but provides numbers that do not make any sense in the vicinity of parallel singularities as shown in Figure 7. As the standard linearization model is badly conditioned when approaching the parallel singularities, the values provided by the standard linearization method tend toward infinity, while the Krawczyk method stops working before giving values that do not make sense. It is noteworthy that the Krawczyk method always provides verified results. Contrary to the Krawczyk method, there is no criterion indicating when the results obtained with the standard linearization method are not reliable.

4. Case study 2: 3-*RRR* planar parallel manipulator

4.1. Manipulator architecture

The 3-RRR planar parallel manipulator under study is shown in Figure 8. It is composed of three identical limbs attached to the moving platform at one end and to the base at the other end. Each limb contains two links and three revolute joints of axes normal to the moving platform. R stands for revolute joint. As a result, the moving platform can translate along the x-axis and the y-axis and rotate about an axis normal to the moving platform. The equations of motion of the manipulator can be found in [11, 25].

4.2. Efficiency of the Krawczyk algorithm

The Krawczyk method is used to evaluate the position error of the moving-platform for a given configuration of the manipulator. The corresponding configuration of the manipulator and the geometric parameters of the manipulator are defined thereafter:

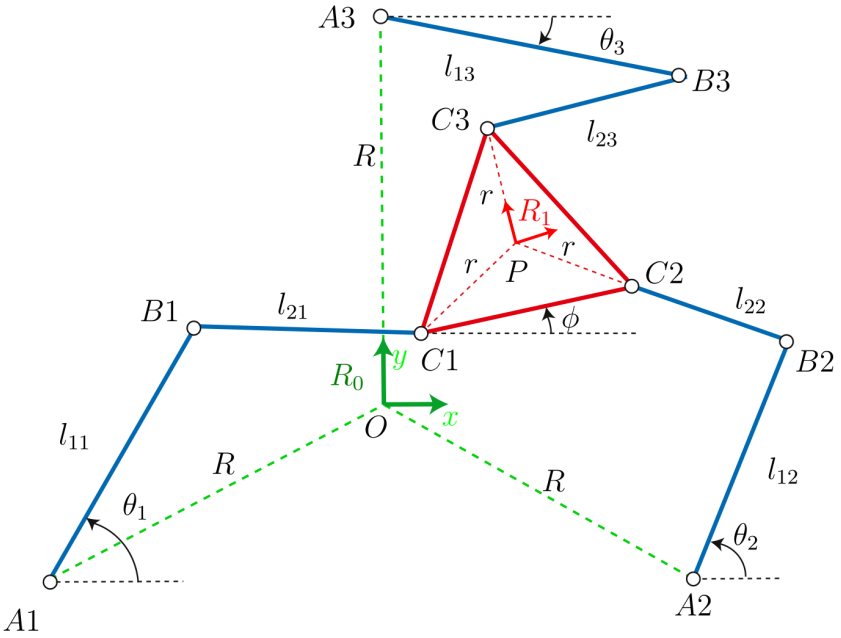
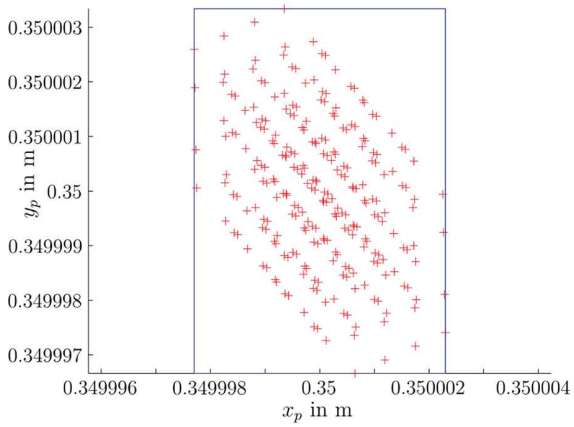


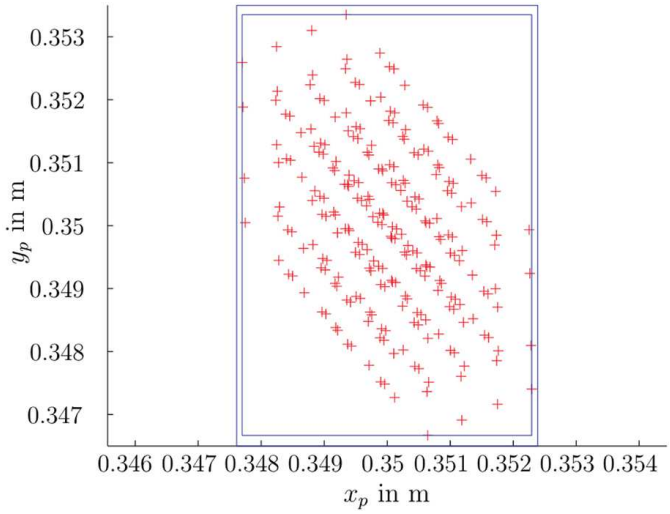
Figure 8: 3-RRR planar parallel manipulator

1. The command variables in the case of the 3-RRR manipulator are θ_1, θ_2, and θ_3. For illustrating the efficiency of the Krawczyk method we choose the following position of the geometric center of the moving-platform expressed in the base frame R_0: $x_p = 0.35$ m and $y_p = 0.35$ m. The orientation of the moving-platform is set to: $\phi = \dfrac{\pi}{4}$ rad.
2. The geometric parameters depicted in Fig. 8 take the following values: $l_{11} = l_{12} = l_{21} = l_{22} = l_{31} = l_{32} = r = 1$ m and $R = 2$ m. They are also the components of vector $\mathbf{a}$ defined in Sec. 2.2 for this problem, i.e., $\mathbf{a} = [l_{11}\, l_{12}\, l_{21}\, l_{22}\, l_{31}\, l_{32}\, r\, R]^T$.

The crosses in Fig. 4.2 represent the possible positions of the geometric center P of the moving-platform for relative errors in the geometric parameters equal to 10^{-6}, i.e., $\Delta a_i/a_i = 10^{-6}$. The Krawczyk box contains all crosses without significant overestimation. Table 2 shows the overestimations ϵ_x, ϵ_y and ϵ_ϕ along the x- and y-axes and about the z-axis, respectively. Those overestimations are evaluated with Eq. (3) and characterize the efficiency of the Krawczyk algorithm for the 3-RRR planar parallel manipulator in a given configuration. It is noteworthy that the higher the relative errors $\Delta a_i/a_i$, the



(a) $\Delta a_i/a_i = 10^{-6}, i = 1, \ldots, 8$



(b) $\Delta a_i/a_i = 10^{-6}, i = 1, \ldots, 8$

Figure 9: Possible positions of point P

Table 2: Overestimations ϵ_x, ϵ_y and ϵ_ϕ along the x- and y-axes and about the z-axis

	$\Delta a_i/a_i = 10^{-6}$	$\Delta a_i/a_i = 10^{-5}$	$\Delta a_i/a_i = 10^{-4}$	$\Delta a_i/a_i = 10^{-3}$
ϵ_x	0.00373 %	0.0373 %	0.3739 %	3.8204 %
ϵ_y	0.00453 %	0.0453 %	0.4543 %	4.6043 %
ϵ_ϕ	0.00366 %	0.0366 %	0.3670 %	3.7482 %

higher the overestimations, the less efficient the Krawczyk algorithm.

The crosses in Fig. 4.2 illustrate the possible positions of point P for relative errors in geometric parameters equal to 10^{-3}, i.e., $\Delta a_i/a_i = 10^{-3}$. Figure 4.2 also highlights a large overestimation case.

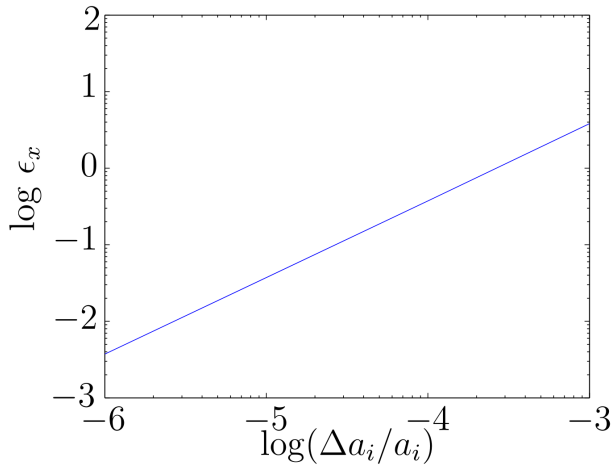


Figure 10: Efficiency of the Krawczyk algorithm for the 3-RRR planar parallel manipulator

Figure 10 illustrates the linear relation between the logarithm of the positioning error of the moving-platform along the x-axis and the logarithm of relative errors $\Delta a_i/a_i$. From Fig 10, it turns out that:

$$\epsilon_x \approx \left(\frac{\Delta a_i}{a_i}\right)^{1.0034}. \tag{15}$$

4.3. Isocontours of the positioning and orientation errors

The maximal positioning error of the geometric center P of the moving platform is defined as in Eq. (14). The maximal orientation error of the

moving platform is defined as:

$$\Delta\phi_{max} \;=\; \overline{\phi} - \underline{\phi} \tag{16}$$

$[\phi]$ denotes the box containing the orientation of the moving platform.

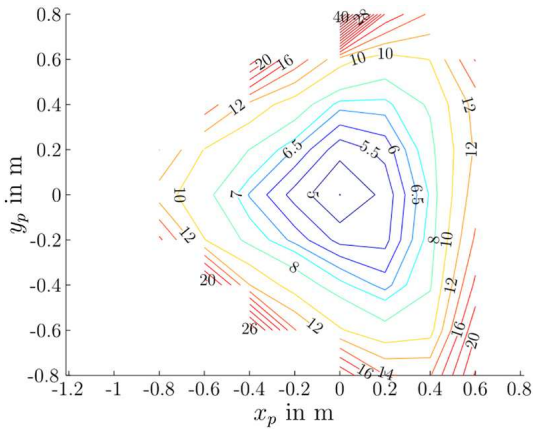
Figures 11 and 12 show the isocontours of Δp_{max} and $\Delta\phi_{max}$ throughout the Cartesian workspace of the manipulator evaluated with the Krawczyk method and the standard linearization method, respectively. The nominal geometric parameters of the manipulator are the following $l_{11} = l_{12} = l_{21} = l_{22} = l_{31} = l_{32} = r = 1$ m and $R = 2$ m. The nominal orientation of the moving platform is equal to $\frac{\pi}{6}$ and the relative errors in the geometric parameters are equal to $\Delta a_i/a_i = 10^{-5}$, $i = 1, \ldots, 8$.

It is apparent that the results obtained with the Krawczyk method and the standard linearization method are similar as long as the manipulator is far from singularities. In the neighborhood of singularities, which appear on the edge of the Cartesian workspace, the Krawczyk method still works while the standard linearization method provides unreliable results. Table 3 shows the comparison between the maximum and minimum values for the sensitivity obtained with the Krawczyk method and the standard linearization method. The maximum positioning and orientation errors computed by the standard linearization method do not make any sense. On the contrary, the Krawczyk method is always reliable.

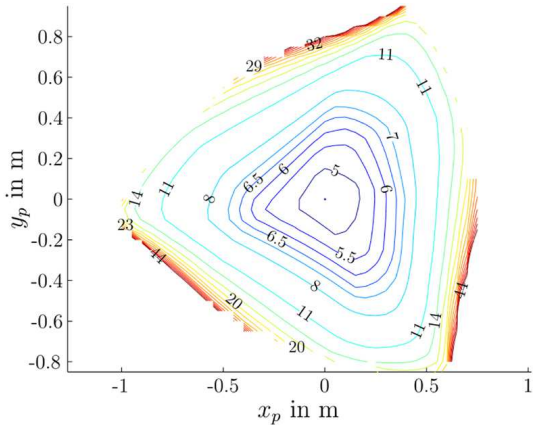
Table 3: Comparison between the minimum and maximum values of Δp_{max} and $\Delta\phi_{max}$ obtained with the Krawczyk and the standard linearization methods

	Krawczyk Method	Standard linearization method
Min Δp_{max} [m]	$4.4899\,10^{-5}$	$4.4878\,10^{-5}$
Max Δp_{max} [m]	$43.4090\,10^{-5}$	$2.4889\,10^{-2}$
Min $\Delta\phi_{max}$ [rad]	$3.6326\,10^{-5}$	$3.6314\,10^{-5}$
Max $\Delta\phi_{max}$ [rad]	$26.4128\,10^{-5}$	$2.1387\,10^{-2}$

Therefore, we can rely on the Krawczyk method for the sensitivity analysis of 3-RRR planar parallel manipulators even in the vicinity of singular configurations. The standard linearization method cannot provide verified results. This makes the Krawczyk method more robust since it can be applied without a prior investigation of the singular configurations of the robot. The Krawczyk method is reliable near serial and parallel singularities.

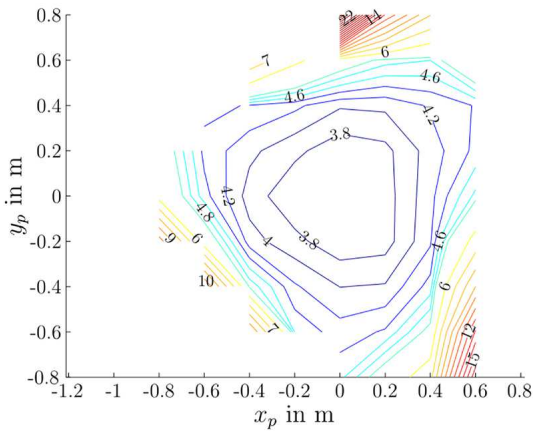


(a) Krawczyk method

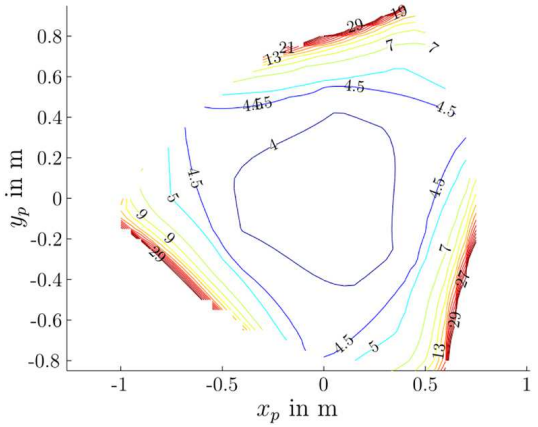


(b) standard linearization method

Figure 11: Isocontours of Δp_{max} [10^{-5} m]



(a) Krawczyk method



(b) standard linearization method

Figure 12: The isocontours of $\Delta\phi_{max}$ [10^{-5} rad]

The Krawczyk method is used in the next section to analyze the sensitivity of the Orthoglide, which is a three degrees-of-freedom translational parallel manipulator.

5. Case study 3: Sensitivity analysis of the Orthoglide

5.1. Introduction

The Orthoglide is a three degrees-of-freedom translational parallel manipulator and is depicted in Fig. 13. It is composed of three identical limbs connected to a moving platform at one end and to the base at the other end. Each limb is a serial kinematic chain containing a prismatic joint P, a revolute joint R, a parallelogram joint P_a and another revolute joint mounted in series. The frame $\mathcal{F}_b$ is attached to the base and the frame $\mathcal{F}_p$ is attached to the moving-platform.

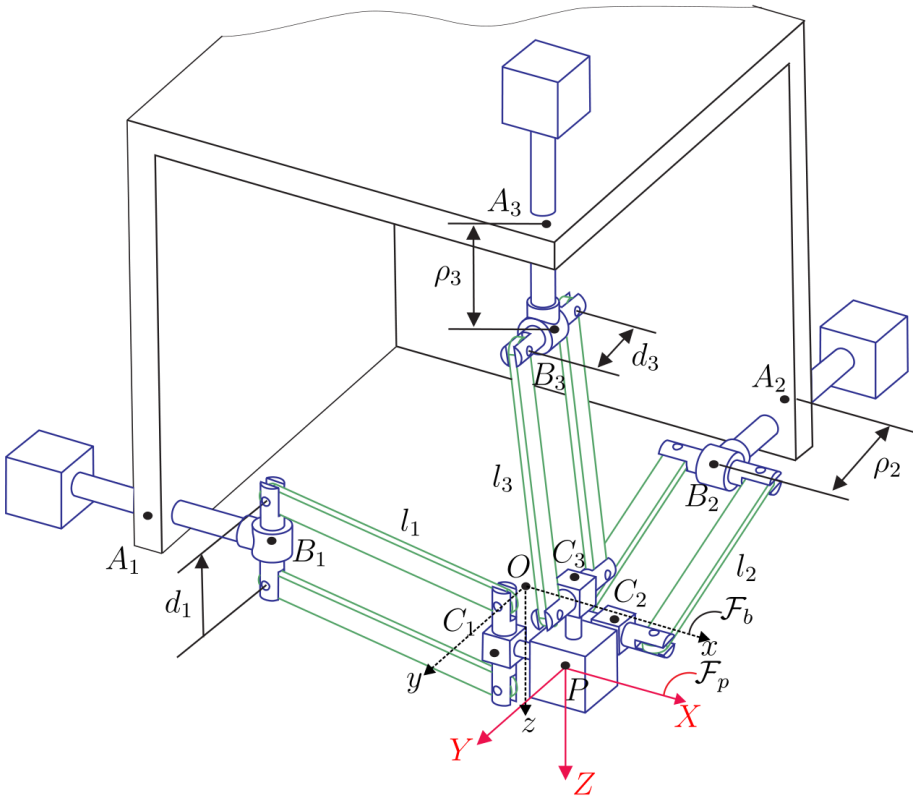


Figure 13: CAD modeling of the Orthoglide

The dimensions of the Orthoglide are given in [12, 26] and are the following:

- $l_i = 310.58$ mm for $i = 1,\ 2,\ 3$.
- $d_i = 80$ mm for $i = 1,\ 2,\ 3$.
- $r_i = 31$ mm for $i = 1,\ 2,\ 3$.

The manipulator has a regular cubic dexterous workspace of size 200 mm $\times$ 200 mm $\times$ 200 mm. ρ_1, ρ_2, and ρ_3 are the actuated joint variables of the robot, namely, its command variables .

The sensitivity analysis of the Orthoglide is conducted using two parallelogram joint modelings:

1. The parallelogram joints are modeled with their global length, i.e., the variations in l_i are considered only, $i = 1,\ 2,\ 3$.
2. The variations in the small and large links of the the parallelogram joints are considered.

5.2. First modeling of the parallelogram joints

Based on [12], we can create a mathematical model of the Orthoglide. The parallelograms are modeled by their global length as shown in Fig. 14.

This model does not allow us to take into account the orientation errors of the end-effector and the variations in the links of the parallelogram joints, namely, variations in parameters d_{i1}, d_{i2}, l_{i1} and l_{i2} as they are all replaced by the global length of the parallelogram.

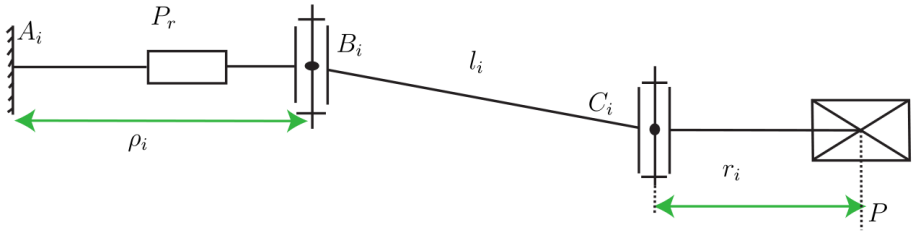


Figure 14: Parameterization of the i^{th} parallelogram joint

Let $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$ denote the Cartesian coordinates of points A_1, A_2 and A_3 expressed in frame $\mathcal{F}_b$:

$$\mathbf{a}_1 = \begin{bmatrix} -a_1 & 0 & 0 \end{bmatrix}^T \tag{17a}$$

$$\mathbf{a}_2 = \begin{bmatrix} 0 & -a_2 & 0 \end{bmatrix}^T \tag{17b}$$

$$\mathbf{a}_3 = \begin{bmatrix} 0 & 0 & -a_3 \end{bmatrix}^T. \tag{17c}$$

Let $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$ denote the Cartesian coordinates of points B_1, B_2 and B_3 expressed in $\mathcal{F}_b$:

$$\mathbf{b}_1 = \begin{bmatrix} -a_1 + \rho_1 & b_{1y} & b_{1z} \end{bmatrix}^T \tag{18a}$$

$$\mathbf{b}_2 = \begin{bmatrix} b_{2x} & -a_2 + \rho_2 & b_{2z} \end{bmatrix}^T \tag{18b}$$

$$\mathbf{b}_3 = \begin{bmatrix} b_{3x} & b_{3y} & -a_3 + \rho_3 \end{bmatrix}^T, \tag{18c}$$

b_{1y}, b_{1z}, b_{2x}, b_{2z}, b_{3x} and b_{3y} represent the uncertainties in the positions of points B_i for $i = 1, 2, 3$.

Likewise, let $\mathbf{c}_1$, $\mathbf{c}_2$ and $\mathbf{c}_3$ denote the Cartesian coordinates of points C_1, C_2 and C_3 expressed in $\mathcal{F}_b$. Therefore,

$$\mathbf{c}_1 = \begin{bmatrix} x_p - r_1 & 0 & 0 \end{bmatrix}^T \tag{19a}$$

$$\mathbf{c}_2 = \begin{bmatrix} 0 & y_p - r_2 & 0 \end{bmatrix}^T \tag{19b}$$

$$\mathbf{c}_3 = \begin{bmatrix} 0 & 0 & z_p - r_3 \end{bmatrix}^T, \tag{19c}$$

x_p, y_p and z_p being the Cartesian coordinates of point P and r_i the distance between point C_i and point P, $i = 1, 2, 3$.

Thus, the equation system that is suitable for the Krawczyk method takes the form:

$$||B_1C_1||^2 - l_1^2 = 0 \tag{20}$$

$$||B_2C_2||^2 - l_2^2 = 0 \tag{21}$$

$$||B_3C_3||^2 - l_3^2 = 0 \tag{22}$$

Let us consider the uncertainties in the following twelve parameters: b_{1y}, b_{1z}, b_{2x}, b_{2z}, b_{3x}, b_{3y}, r_i, and l_i for $i = 1, 2, 3$. The direct method requires 2^{12} pose errors to be computed. It gives us the boundaries of the possible end-effector positions represented with crosses in Figs. 5.2 and 5.2.

Figure 5.2 shows the efficiency of the Krawczyk method in the (x_p, y_p) plane for errors in the parameters equal to 10^{-4} m. The results are similar in planes (x_p, z_p) and (y_p, z_p). Figure 5.2 depicts the efficiency of the method when the relative geometric errors are equal to 10^{-2} m. We can notice that the overestimation is large when the geometric errors become larger than 10^{-2} m.

Table 4 shows the efficiency of the method with regard to the relative errors in the geometric parameters.

Table 4: Efficiency of the Krawczyk method for the sensitivity analysis of the Orthoglide

	Relative Uncertainties		
	10^{-4}	10^{-3}	10^{-2}
ϵ_x	0.071632	0.719432	7.559653
ϵ_y	0.071632	0.719432	7.559653
ϵ_z	0.071628	0.719072	7.525974

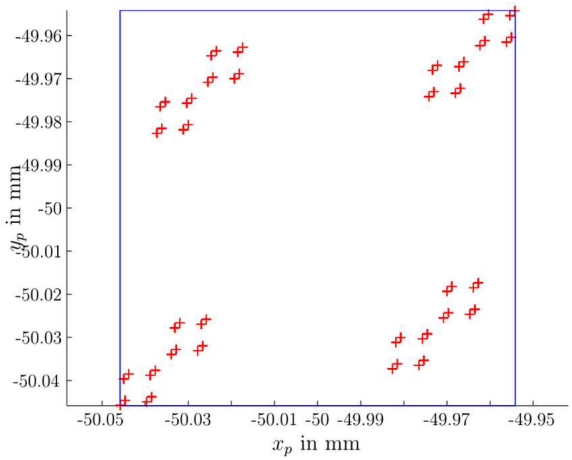
Figure 16 shows the logarithm of the overestimation ϵ defined in Eq. (3) as a function of the logarithm of the relative errors in the geometric parameters, namely $\log(\Delta\mathbf{a}/\mathbf{a})$. It appears that this overestimation is the same along the x-, y- and z-axes. From Fig. 16, the evolution of the efficiency of the Krawczyk method with respect to uncertainties in parameters $\mathbf{a}$ is polynomial. Indeed,

$$\epsilon \approx \left(\frac{\Delta\mathbf{a}}{\mathbf{a}}\right)^{1.011} . \tag{23}$$

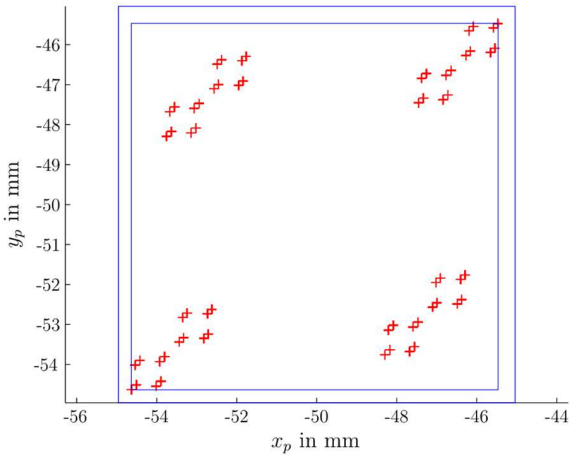
The Orthoglide has a cubic dexterous workspace as shown in Fig. 17. The Cartesian coordinates of points Q_1 and Q_2, two opposite vertices of the cubic workspace, are expressed in frame $\mathcal{F}_b$ as follows:

- $Q_1 = (-73.21, -73.21, -73.21)$ [mm]
- $Q_2 = (126.79, 126.79, 126.79)$ [mm]

Here the sensitivity of the Orthoglide to variations in its geometric parameters is analyzed along segment Q_1Q_2.



(a)



(b)

Figure 15: Orthoglide's end-effector positions in the (x_p, y_p) plane for relative geometric errors equal to (a) 10^{-4} and (b) 10^{-2}

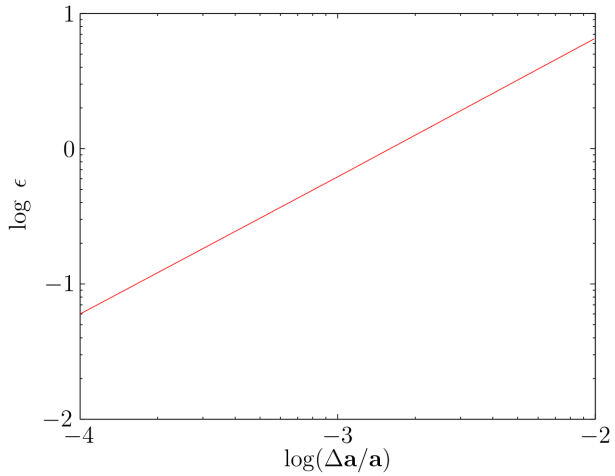


Figure 16: Efficiency of the Krawczyk method for the sensitivity analysis of the Orthoglide

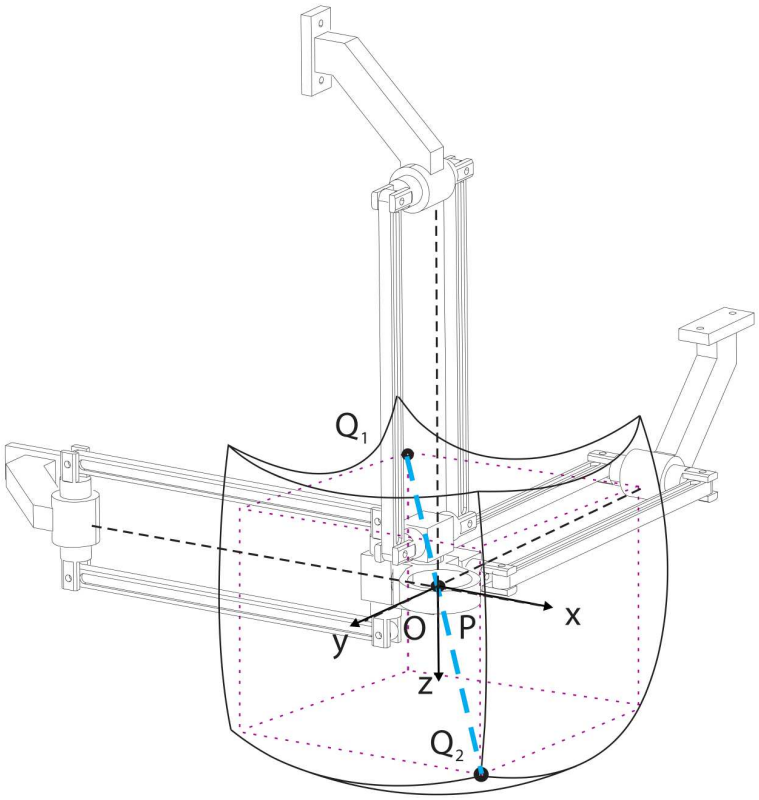
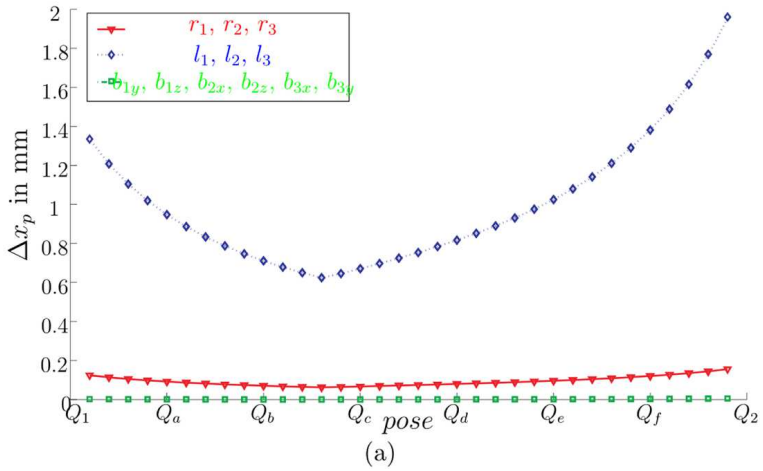
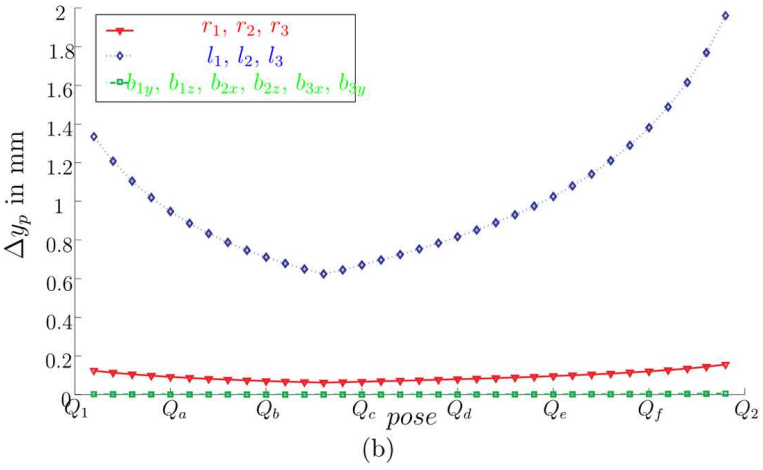


Figure 17: The Orthoglide's workspace


r_1, r_2, r_3
l_1, l_2, l_3
b_{1y}, b_{1z}, b_{2x}, b_{2z}, b_{3x}, b_{3y}
Δx_p in mm
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
Q_1 Q_a Q_b Q_c $pose$ Q_d Q_e Q_f Q_2

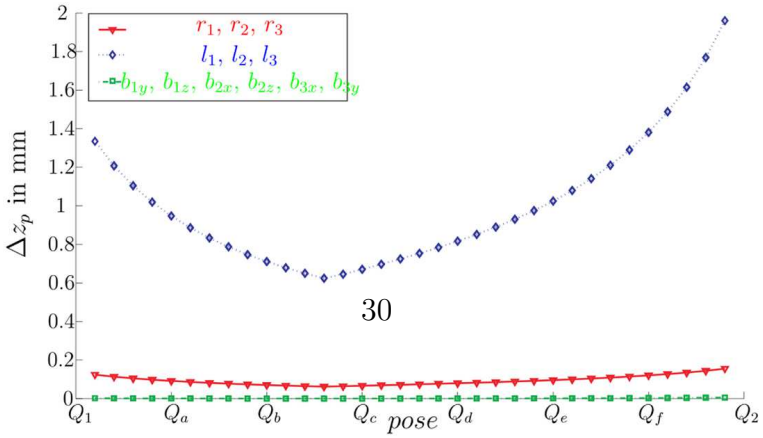
(a)

(a)


r_1, r_2, r_3
l_1, l_2, l_3
b_{1y}, b_{1z}, b_{2x}, b_{2z}, b_{3x}, b_{3y}
Δy_p in mm
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
Q_1 Q_a Q_b Q_c $pose$ Q_d Q_e Q_f Q_2

(b)

(b)


r_1, r_2, r_3
l_1, l_2, l_3
b_{1y}, b_{1z}, b_{2x}, b_{2z}, b_{3x}, b_{3y}
Δz_p in mm
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
Q_1 Q_a Q_b Q_c $pose$ Q_d Q_e Q_f Q_2

(c)

The sensitivity of the Orthoglide along the x-, y- and z-axes is conducted separately. The sensitivity indices are the differences between the upper and lower bounds of the end-effector position along the x-, y- and z-axes and are denoted by Δx_p, Δy_p and Δz_p.

Figures 5.2-(c) represent the sensitivity of the Orthoglide's end-effector pose to variations in the twelve geometric parameters that are considered along the x-, y- and z-axes. The uncertainties in parameters l_i and r_i, $i = 1, 2, 3$, are chosen arbitrarily to be equal to 1/1000 times the leg length and the uncertainties in parameters b_{1y}, b_{1z}, b_{2x}, b_{2z}, b_{3x}, and b_{3y} are supposed to be equal to 10^{-3} mm.

From Figs. 5.2-(c), it is apparent that the sensitivity of the Orthoglide's end-effector position is highly dependent of the robot configuration. As a matter of fact, this sensitivity is a minimum in the isotropic configuration, namely, when point P and the origin O of the base frame coincide. It also appears that the sensitivity of the Orthoglide's end-effector position is a maximum to variations in lengths l_1, l_2 and l_3, while it is a minimum to uncertainties in the location of points B_i, namely to variations in geometric parameters b_{1y}, b_{1z}, b_{2x}, b_{2z}, b_{3x} and b_{3y}.

5.3. Second modeling of the parallelogram joints

Figure 19 represents the kinematic model of the Orthoglide considering variations in all its geometric parameters. Nevertheless, links A_iB_i and C_iP are supposed to remain orthogonal to $B_{i1}B_{i2}$ and $C_{i1}C_{i2}$, respectively, $i = 1, 2, 3$.

As shown in Fig. 20, $d_{i1} = d_{i2} = d_i$ and $l_{i1} = l_{i2} = l_i$ in the nominal case, i.e., when the variations in the geometric parameters are null.

First, let $\mathbf{n}_i$ be a unit vector orthogonal to the parallelogram plane, i.e, $\mathbf{n}_i$ is orthogonal to the parallelogram $B_{i1}B_{i2}C_{i2}C_{i1}$, namely,

$$\mathbf{n}_i = \frac{\mathbf{B}_{i2}\mathbf{C}_i \times \mathbf{B}_{i2}\mathbf{B}_{11}}{||\mathbf{B}_{i2}\mathbf{C}_i \times \mathbf{B}_{i2}\mathbf{B}_{11}||_2}, i = 1,\ 2,\ 3. \tag{24}$$

Let $\mathbf{u}_i$ and $\mathbf{v}_i$ be the unit vectors directed along vector $\mathbf{C}_i\mathbf{P}$ and vector $\mathbf{C}_{i2}\mathbf{C}_{i1}$, respectively.

$$\mathbf{v}_i = \frac{\mathbf{C}_i\mathbf{P}}{||\mathbf{C}_i\mathbf{P}||_2}, i = 1,\ 2,\ 3. \tag{25}$$

Vector $\mathbf{u}_i$ is defined such that

$$\mathbf{u}_i = \mathbf{n}_i \times \mathbf{v}_i, i = 1,\ 2,\ 3. \tag{26}$$

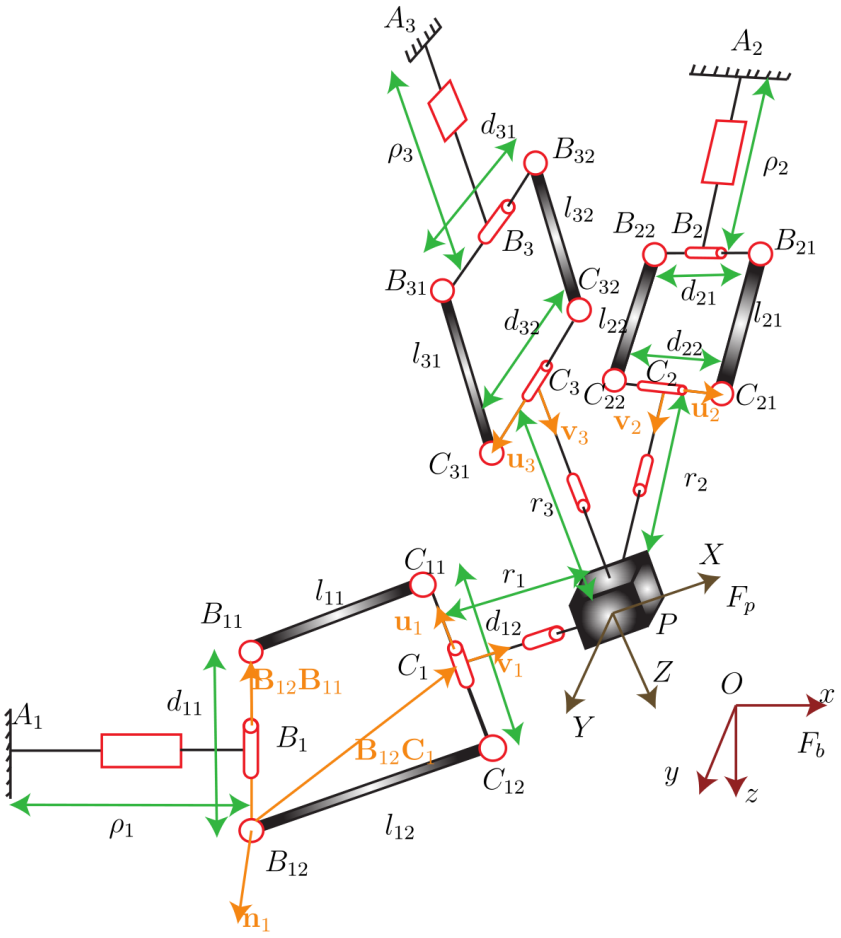


Figure 19: Schematic representation of the Orthoglide with variations in all its geometric parameters

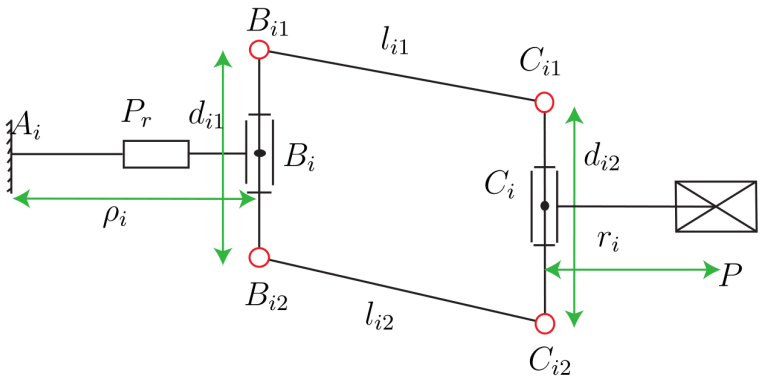


Figure 20: Parametrization of the i^{th} leg of the Orthoglide

The latter amounts to

$$\mathbf{u}_i = \frac{\mathbf{C}_{i2}\mathbf{C}_{i1}}{||\mathbf{C}_{i2}\mathbf{C}_{i1}||_2}, i = 1,\ 2,\ 3 \tag{27}$$

where $||.||_2$ denotes the 2-norm.

Contrary to the modeling used in Sec 5.2 that only considers the positioning errors of the Orthoglide's end-effector, the modeling introduced in this section deals with both its positioning and orientation errors as the geometric variations in the parallelogram joints are considered in the mathematical model.

In order to take into account the positioning and orientation errors of the end-effector, we should have six variables. It means that we need to write six independent expressions that connect the variables to the parameters in the form $\mathbf{f}(\mathbf{a}, \mathbf{x}) = 0$.

Frame $\mathcal{F}_p$ of origin O_p is attached to the end-effector as shown in Fig. 19, whereas frame $\mathcal{F}_b$ of origin O_b is attached to the base.

The transformation matrix from frame $\mathcal{F}_p$ to frame $\mathcal{F}_b$ takes the form[4]:

$$\mathbf{T}_p^b = \begin{bmatrix} \mathbf{R}_p^b & [\mathbf{o}_p]_b \\ \mathbf{0}_3 & 1 \end{bmatrix}, \tag{28}$$

with

$$\mathbf{R}_p^b = \begin{bmatrix} c_\phi c_\theta & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ s_\phi c_\theta & s_\phi s_\theta s_\psi - c_\phi c_\psi & s_\phi s_\theta c_\psi + c_\phi s_\psi \\ -s_\theta & c_\theta s_\psi & c_\theta c_\psi \end{bmatrix}. \tag{29}$$

[4] c_ϕ, c_θ, c_ψ, s_ϕ, s_θ and s_ψ stand for $\cos\phi$, $\cos\theta$, $\cos\psi$, $\sin\phi$, $\sin\theta$ and $\sin\psi$, respectively.

Euler angles ϕ, θ and ψ are used to parameterize the orientation of the end-effector. $[\mathbf{o}_p]_b$ is the position vector of point O_p expressed in frame $\mathcal{F}_b$ and $\mathbf{0}_3$ denotes the three-dimensional zero vector.

In order to formulate the required expressions for the Krawczyk operator, we first express the Cartesian coordinates of points A_i and B_i in frame $\mathcal{F}_b$. Since joints are supposed to be perfect, $B_{i1}B_{i2}$ remains perpendicular to A_iB_i and we can easily get the Cartesian coordinates of points B_{i1} and B_{i2}.

In frame $\mathcal{F}_p$, we can get the coordinates of points C_i as a function of the pose of the end-effector. The Cartesian coordinates of points C_i expressed in frame F_b are obtained by means of matrix $\mathbf{R}_p^b$.

As a result, the six equations required for the Krawczyk operator can be formulated as:

$$\begin{aligned} ||\mathbf{B}_{i1}\mathbf{C}_{i1}||_2^2 - l_{i1}^2 = 0 \\ ||\mathbf{B}_{i2}\mathbf{C}_{i2}||_2^2 - l_{i2}^2 = 0, i = 1,\, 2,\, 3 \end{aligned} \tag{30}$$

The relative geometric errors are supposed to be equal to 10^{-4}. Moreover, the positioning error of the end-effector is expressed as:

$$\Delta p = \sqrt{\Delta x_p^2 + \Delta y_p^2 + \Delta z_p^2} \tag{31}$$

where Δx_p, Δy_p, and Δz_p denote the positioning error of the end-effector along the x-, y- and z-axes, respectively.

The orientation error μ_r of the end-effector is expressed as:

$$\mu_r \;=\; \arccos \frac{Tr(\mathbf{R}_p^b) - 1}{2} \tag{32}$$

where $Tr(\mathbf{R}_p^b)$ is the trace of the rotation matrix $\mathbf{R}_p^b$ expressed in Eq. (29).

The sensitivity analysis of the Orthoglide to the variations in its geometric parameters is conducted along segment Q_1Q_2 like in Sec. 5.2,. Uncertainties are considered separately in each parameter.

Figures 21 and 22 represent the sensitivity of the orientation and position of the end-effector to variations in all geometric parameters along segment Q_1Q_2, respectively.

From Figs. 21 and 22, links l_{i2}, $i = 1, 2, 3$, have the largest influence on the pose of the end-effector. We can also notice that the position of the end-effector is more sensitive to the variations in the actuated joints than its orientation with regard to the sensitivity to the variations in the

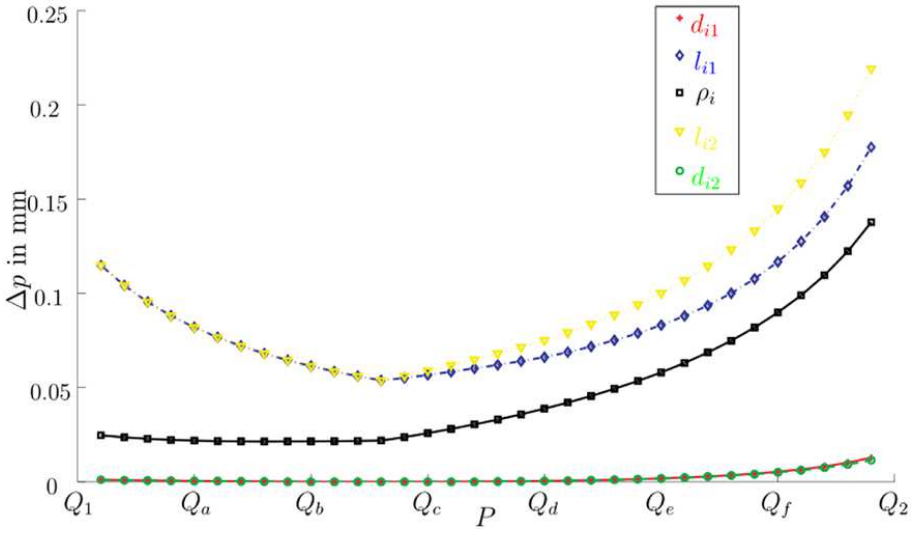


Figure 21: Sensitivity of the orientation of the end-effector to variations in geometric parameters along segment Q_1Q_2.

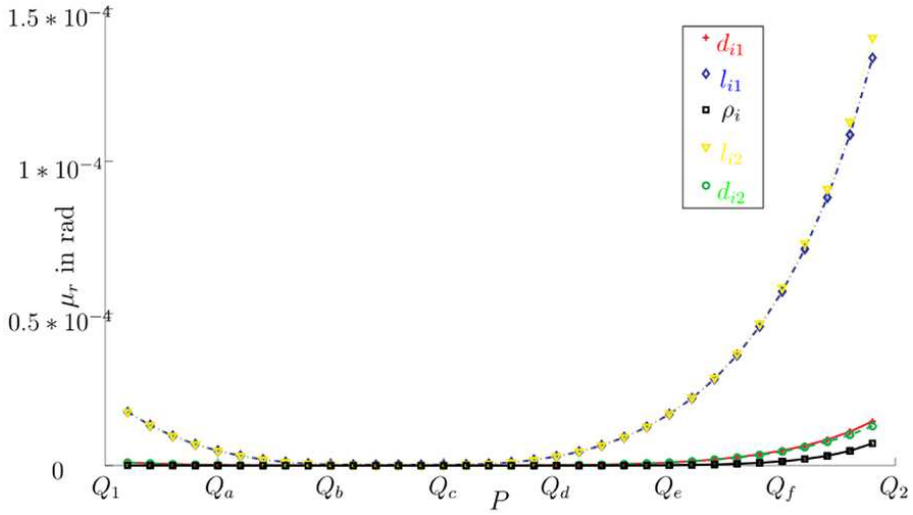


Figure 22: Sensitivity of the orientation of the end-effector to variations in geometric parameters along segment Q_1Q_2.

other geometric parameters. However, it is apparent that the varitions in the actuated joints have larger influence on the end-effector's pose than variations in link lentghs d_{i1} and d_{i2}.

The comparison of these results with those obtained in [12] also shows that the Krawczyk operator is efficient and relevant for the sensitivity analysis of the Orthoglide.

6. Conclusions

The interval linearization method proposed in this paper provides *verified* results for the sensitivity analysis of serial and parallel manipulators since it is based on a rigorous linearity that takes nonlinearities into consideration. This fact represents a major advantage in comparison with probabilistic and standard linearization methods. The method turns to give accurate results even when the manipulator under study is close to a singular configuration. As a matter of fact, as long as the algorithm converges, it will provide verified and relevant results. It means that we provided an auto validation algorithm: If it provides results, they will be verified even close to singular configurations. It is a major advantage in comparison with the standard linearization method where results are provided even too close to singular configurations, but are not reliable. It has also proved to be general since it works for simple mechanisms such as the five-bar mechanism, and complex ones such as the Orthoglide.

The method proposed in this paper was implemented using the Interval Laboratory `Intlab` [21] that works under `Matlab`. Implemented on a 1.86 GHz Intel Pentium Dual-core CPU, and a 3 GHz memory, thirty seconds were required to draw the isocontours for the five-bar mechanism, whereas up to one minute was required to analyze the sensitivity of the Orthoglide. However, when the full mathematical model of the Orthoglide was used, the computational time had been higher, since the corresponding model is very complex and implements hundreds of trigonometric functions in each of the six equations. We can say that our method is faster that the algorithm presented in [19] since no bisection is needed.

As a conclusion, the proposed method for the sensitivity analysis of serial and parallel manipulators provides verified results and the computation time is acceptable. Later on, the method will be used to compute the maximum pose errors of serial and parallel manipulators due to joint clearances.

Appendix A. Proof of the main result

We prove here that $[\mathbf{K}]([\mathbf{x}]) \subseteq \text{int}[\mathbf{x}]$ implies properties (A) and (B) of Section 2.3. First of all, by Corollary 4.4.7 of [23] $[\mathbf{K}]([\mathbf{x}]) \subseteq \text{int}[\mathbf{x}]$ implies that $\mathbf{C}[\mathbf{f_x}]([\mathbf{x}],[\mathbf{a}])$ is an H-matrix, and thus by definition of strong regularity that $[\mathbf{f_x}]([\mathbf{x}],[\mathbf{a}])$ is strongly regular, which proves Property (B).

For an arbitrary $\mathbf{a} \in [\mathbf{a}]$, let us define $\mathbf{g}(\mathbf{x}) := \mathbf{f}(\mathbf{x},\mathbf{a})$. Using the well known properties of interval centered extensions,

$$\big(\mathbf{C}[\mathbf{f_a}](\tilde{\mathbf{x}},[\mathbf{a}])\big)([\mathbf{a}]-\tilde{\mathbf{a}}) + \mathbf{C}[\mathbf{f}](\tilde{\mathbf{x}},\tilde{\mathbf{a}}) \supseteq \mathbf{C}\mathbf{f}(\tilde{\mathbf{x}},\mathbf{a}), \tag{A.1}$$

which is $\mathbf{C}\mathbf{g}(\tilde{\mathbf{x}})$. Furthermore, we have $\mathbf{g_x}(\mathbf{x}) = \mathbf{f_x}(\mathbf{x},\mathbf{a})$ and therefore $[\mathbf{f_x}]([\mathbf{x}],[\mathbf{a}])$ is an interval extension of the derivatives of $\mathbf{g}$. So far we have proved that

$$[\mathbf{K}]([\mathbf{x}]) \supseteq \tilde{\mathbf{x}} - \mathbf{C}\mathbf{g}(\tilde{\mathbf{x}}) - (\mathbf{C}[\mathbf{g_x}]-\mathbf{I})([\mathbf{x}]-\tilde{\mathbf{x}}). \tag{A.2}$$

We can finally apply Theorem 5.1.8 of [23] which proves that $[\mathbf{K}]([\mathbf{x}]) \subseteq \text{int}[\mathbf{x}]$ implies $\mathbf{g}$ has an unique solution inside $[\mathbf{x}]$. Since this holds for an arbitrary $\mathbf{a} \in [\mathbf{a}]$, we have proved Property (A).

References

[1] Rao, S. S., and P.K., B., 2001. "Probabilistic approach to manipulator kinematics and dynamics". *Reliability Engineering and System Safety,* **72**, pp. 45–58.

[2] Caro, S., Binaud, N., and Wenger, P., 2009. "Sensitivity analysis of 3-RPR planar parallel manipulators". *Journal of Mechanical Design,* **131**, pp. 121005 1–121005 13.

[3] Merlet, J.-P., 2009. "Interval analysis for certified numerical solution of problems in robotics". *International journal for applied mathemathics and computer science,* **vol. 19**, pp. 399–412.

[4] Muszynska, A., and Goldman, P., 1995. "Chaotic responses of unbalanced rotor/bearing/stator systems with looseness or rubs". *Chaos, Solitons & Fractals,* **5**, pp. 1683–1704.

[5] Kaupp, T., Makarenko, A., and Durrant-Whyte, H., 2008. "Human-robot communication for collaborative decision making - a probabilistic approach". *Robotics and Autonomous Systems,* **58**, pp. 444–456.

[6] Di Gregorio, R., and Parenti-Castelli, V., 1999. "Influence of the geometric parameters of the 3-UPU parallel manipulator on the singularity loci". In Proceedings of the 1999 International Workshop on Parallel Kinematic Machines PKM'99, Milano, Italia.

[7] Fan, K.-C., Wang, H., and Zhao, J.-W., 2003. "Sensitivity analysis of the 3-PRS parallel kinematic spindle platform of a serial-parallel machine tool". *International Journal of Machine Tools and Manufacture,* **43**, p. 1561–1569.

[8] Han, C., Kim, J., and Kim, J., 2002. "Kinematic sensitivity analysis of the 3-UPU parallel mechanism". *Mechanism and Machine Theory,* **37**, p. 787–798.

[9] Caro, S., Binaud, N., and Wenger, P., 2009. "Sensitivity analysis of 3-RPR planar parallelmanipulators". *ASME Journal of Mechanical Design,* **131**, pp. 121005–1–121005–13.

[10] Caro, S., Binaud, N., and Wenger, P., 2008. "Sensitivity analysis of planar parallel manipulators". In Proceedings of ASME Design Engineering Technical Conferences,New-York City, New-York, USA.

[11] Binaud, N., Caro, S., and Wenger, P., 2010. "Sensitivity comparison of planar parallel manipulators". *Mechanism and Machine Theory,* **45**(11), pp. 1477–1490.

[12] Caro, S., Wenger, P., Bennis, F., and Chablat, D., 2006. "Sensitivity analysis of the Orthoglide : A three-dof translational parallel kinematic machine". *ASME Journal of Mechanical Design,* **128**, pp. 392–402.

[13] Han, C., Kim, J., Kim, J., and Chongwoo Park, F., 2002. "Kinematic sensitivity analysis of the 3-UPU parallel mechanism". *Mechanism and Machine Theory,* **37**, pp. 787–798.

[14] Zhuang, H., 1997. "Self-calibration of parallel mechanisms with a case study on stewart platforms". *IEEE Transactions on Robotics and Automation,* **13(3)**, pp. 387–397.

[15] Wang, H., and Roth, B., 1989. "Position errors due to clearances in journal bearings". *ASME Journal of Mechanisms, Transmissions, and Automation in Design,* **111**, pp. 315–320.

[16] Innocenti, C., 2002. "Kinematic clearance sensitivity analysis of spatial structures with revolute joints". *ASME Journal of Mechanical Design,* **124**, pp. 52–57.

[17] Voglewede, P., and Ebert-Uphoff, I., 2004. "Application of workspace generation techniques to determine the unconstrained motion of parallel manipulators". *ASME Journal of Mechanical Design,* **126(2)**, pp. 283–290.

[18] Binaud, N., Cardou, P., Caro, S., and Wenger, P., 2010. "The kinematic sensitivity analysis of robotic manipulators to joint clearances". In ASME 2010 International Design engineering Technical Conferences and Computers and Information in Engineering Conference.

[19] Wu, W., and Rao, S. S., 2004. "Interval approach for the modeling of tolerances and clearances in mechanism". *Journal of Mechanical Design,* **126**, pp. 581–592.

[20] Moore, R., Kearfott, B., and Cloud, M., 2009. *Introduction to Interval Analysis*. Siam.

[21] Rump, S., 1999. "INTLAB - INTerval LABoratory". In *Developments in Reliable Computing*, T. Csendes, ed. Kluwer Academic Publishers, Dordrecht, pp. 77–104. `http://www.ti3.tu-harburg.de/rump/`.

[22] Goldzstejn, A., 2008. "Sensitivity analysis using a fixed point interval iteration". *Hal,* **00339377**, pp. 1–5.

[23] Neumaier, A., 1990. *Interval Methods for Systems of Equations*. Cambridge Univ. Press.

[24] Wenger, P., Gosselin, C., and Maille, B., 1999. "A comparative study of serial and parallel mechanism topologies for machine tool". In Int. Workshop on Parallel Kinematic Machines, Milan, Italy, p. 23–35.

[25] Bonev, I., Zlatanov, D., and Gosselin, C., 2003. "Singularity analysis of 3-dof planar parallel mechanisms via screw theory". *ASME Journal of Mechanical Design,* **125**, p. 573–581.

[26] Pashkevich, A., Wenger, P., and Chablat, D., 2007. "Kinematics and workspace analysis of a three-axis parallel manipulator: the Orthoglide". In IEEE International Conference on Robotics and Automation.