Symbolic Elimination for Parallel Manipulators

Luc Tancredi, Monique Teillaud, Olivier Devillers

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Symbolic Elimination for parallel manipulators

Luc Tancredi, Monique Teillaud, Olivier Devillers

N° 2809

Février 1996

PROGRAMME 4

Rapport de recherche

ISSN 0249-6399

Symbolic Elimination for parallel manipulators

Luc Tancredi[*], Monique Teillaud[**], Olivier Devillers[***]

Programme 4 — Robotique, image et vision
Projet Prisme

Rapport de recherche n°2809 — Février 1996 — 31 pages

Abstract: The forward kinematics problem of parallel robot consists in computing the position of a solid moving in the three-dimensional space with six points on it constrained to lie respectively on six given spheres. This problem can also be represented by a system of algebraic equations, and it is known to admit at most 40 complex solutions.

Our approach consists in adding extra sensors, that give us more information, in order to reduce this bound. We propose a symbolic elimination method, based on the scheme of dialytic elimination. We prove that our method actually gives upper bounds on the number of solutions of the problem. Our implementation supplies us with the first general bounds on this problem with extra sensors.

Key-words: parallel manipulator, forward kinematics problem, algebraic geometry

(Résumé : tsvp)

[*] This work was done while this author was supported by INRIA
[**] INRIA, B.P.93, 06902 Sophia-Antipolis cedex (France), Phone: +33 93 65 77 62, E-mail: Monique.Teillaud@sophia.inria.fr.
[***] INRIA, B.P.93, 06902 Sophia-Antipolis cedex (France), Phone: +33 93 65 77 63, E-mail: Olivier.Devillers@sophia.inria.fr.

Elimination symbolique et robots parallèles

Résumé : Le modèle géométrique direct des robots parallèles consiste à calculer la position d'un solide se déplaçant dans l'espace a trois dimensions, lorsque six de ses points sont contraints à rester sur six sphères données. Ce problème peut aussi être représenté par un système d'équations algébriques, et il est connu qu'il admet au plus 40 solutions complexes.

Notre approche consiste à ajouter des capteurs supplémentaires, qui nous fournissent des informations additionnelles, de façon à réduire cette borne. Nous proposons une méthode d'élimination symbolique, basée sur le schéma de l'élimination dialytique. Nous démontrons que notre méthode donne effectivement des bornes sur le nombre de solutions du problème. Notre implantation nous permet d'obtenir les premières bornes générales pour le modèle géométrique direct avec des capteurs additionnels.

Mots-clé : robot parallèle, modèle géométrique direct, géométrie algébrique

1 Introduction

Parallel manipulators

A general six degrees of freedom parallel manipulator is made of two rigid bodies connected to each other by six links. One of the bodies is fixed, the *base*; the other body is a movable *platform*. Each link is connected to the base by a universal joint and to the platform by a ball-and-socket joint. Linear actuators enable the links to change their length. For each actuator a sensor measures the length of the leg (see Figure 1). We denote by A_i the center of the base joint of the i^{th} leg, B_i the center

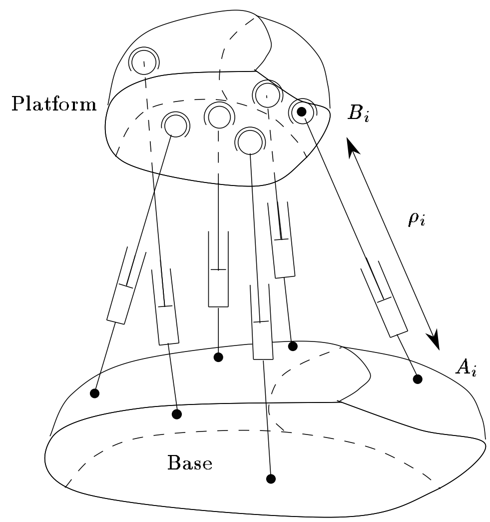

Figure 1: General six degrees of freedom parallel manipulator

of the platform joint. ρ_i will be the length of the i^{th} leg, *i.e.* $\rho_i = ||A_iB_i||$, where $||A_iB_i||$ is the norm of vector A_iB_i.

Forward kinematics problem

We consider the forward (or direct) kinematics problem (FKP for short): for a given set of link lengths ρ_i, $i \in \{1, \ldots, 6\}$, as measured by the actuators' sensors, determine the posture of the platform in the reference frame. In this paper *posture* means position *and* orientation of the platform. An algebraic formulation of the problem is the following :

$$\rho_i^2 = ||A_i B_i||^2, \qquad i \in \{1, \ldots, 6\}$$

where A_i and ρ_i are known and B_i is unknown, for $i \in \{1, \ldots, 6\}$. The FKP does not have a unique solution. An upper bound on the number of complex solutions is 40. This result has been obtained by continuation methods [Rag91, Wam94]. Algebraic proofs are also available [Laz92, Mou93], and recently an algorithm has been presented which gives the forty solutions of the FKP [Hus94]. Examples with 24 real solutions have been found [IP93, Tan95a], but no examples have been found yet with more than 24 real solutions. Many special architectures have been studied but the number of solutions still remains quite high even for very particular architectures (see for example [CR89, IP90, IP91, ZS91, LM94]). In the general case, a systematic study of the bound for different combinatorial classes of manipulators was made by Faugère and Lazard [FL95].

Our approach consists in adding more data, by using extra sensors. As Stoughton and Arai [SA91] and Merlet [Mer93], we choose the use of rotary sensors located on the base joints. Such a sensor placed on the i^{th} leg measures an angle, or equivalently a plane containing A_i and B_i, which results in an equation of the form

$$A_i B_i \cdot N_i = 0$$

where N_i is a vector normal to the plane.

Thus we are given an *over-constrained* set of algebraic equations: six unknowns (six degrees of freedom), and at least seven equations. Intuitively, such a problem usually admits no solution, but if we assume that the robot is well built, the problem must have one solution by hypothesis. We are interested here in obtaining a bound on the maximum number of solutions. Some bounds for the same problem have already been proposed in [TTM95a, TTM95b] for special cases. In this paper we propose a symbolic elimination method, and we prove that this method enables us to tackle the general case: general architecture, and general placements of the extra sensors.

Elimination and resultants

Elimination theory is a fundamental aspect of algebraic geometry. Given s polynomial equations $f_1 = f_2 = \ldots = f_s = 0$ in $\mathbb{K}[x_1, \ldots, x_k]$, for $\mathbb{K}$ a given field, an elimination method consists in two steps: an elimination step and an extension step. The elimination step computes a consequence $f_{s+1} = 0$ in only one variable. The extension step considers $f_{s+1} = 0$ as solved and extends its solutions into solutions of the whole initial system.

Elimination methods are often used in theory of mechanisms. They avoid the drawbacks of iterative methods, such as dealing with solutions at infinity, some typical numeric problems, and the lack of information on the behavior of the solutions [Rot93]. Moreover, methods based on resultants are quicker than Gröbner bases and homotopy methods [Emi94].

During the last century, SYLVESTER introduced the resultant to perform an elimination for two polynomials in one variable. The resultant is a polynomial whose degree gives a bound on the number of common roots of the two polynomials, the Bézout bound. The multivariate resultant [vdW48, Can88] generalizes this notion to the case of k homogeneous polynomials in k variables. EMIRIS [Emi94] shows that this resultant is a particular class of sparse resultant when the coefficient of the polynomials are generic, that is when they do not satisfy any particular relation among each other. Sparse elimination takes place in $(\mathbb{C} \setminus \{0\})^k$, while multivariate resultant is computed in the projective space $\mathbb{P}^k(\mathbb{C})$. EMIRIS' program[1] can compute this sparse resultant. The degree of this polynomial is the mixed volume of the Newton polytopes of the initial polynomials, known as the BKK bound [Ber75]. Other bounds have been found recently for the number of solutions of a system in $\mathbb{C}^k$, the best one is given in [HS96].

A more complete overview of related results on systems of algebraic equations can be found for example in [EC96].

Outline of the paper

As far as we know, there is no tool that can deal with a system having more equations than unknowns, as our problem has. That is why we propose here a new method, based on dialytic elimination.

In Section 2, we briefly recall the scheme of dialytic elimination. Then we present in Section 3 our symbolic method. We show that the symbolic case poses complex problems. Our method constructs a polynomial in a single variable, and it is quite involved to prove that the degree of this polynomial actually gives an upper bound on the number of (complex) solutions of the FKP for parallel robots with extra sensors. The proof is given in Section 4. Finally, practical results for parallel robots are presented in Section 5.

[1] `ftp://robotics.eecs.berkeley.edu/pub/emiris/res_solver`

2 Dialytic elimination

SYLVESTER's dialytic elimination computes Sylvester's resultant. This method can only deal with problems in one or two unknowns [Sal85] and a small number of equations. ROTH [Rot93] presented a generalization of this method, by adapting it to several unknowns and a large number of equations.

The general idea consists in transforming a non linear system into a linear system, which is easy to solve. This is performed by considering each monomial as a new unknown, without considering the relations between these monomials. The method can be divided into seven steps:

1. Consider one variable as a parameter: *hide* a variable.

2. In most cases, the number of monomials is bigger than the number of equations. The second step consists in constructing as many equations as necessary to obtain a square system, linear, independent and homogeneous with respect to the monomials. The non-linear relations among the monomials are not considered. New equations are usually constructed by multiplying some equations by monomials. Of course, during that process, new monomials appear. This step must go on until a square system of independent equations is obtained.

3. Rename the monomials (including the constant monomial 1) as new variables. The system becomes linear and homogeneous in the new variables.

4. Compute the determinant of this linear system, which is a polynomial in the hidden variable. Each solution of the initial system necessarily corresponds to a root of this determinant.

5. Compute all the roots of this polynomial.

6. For each root, substitute the hidden variable by it and solve the system obtained in this way, which is linear in the monomials.

7. For each root of the determinant, deduce the values of the variables of the initial system from the values of the monomials and using now the non-linear relations between monomials. Any valid solution must satisfy all the relations between the monomials.

3 Symbolic method

The method we implemented in *Maple*[2] is based on dialytic elimination. However, we encounter different problems, since we are interested in the symbolic case, and we try to give bounds on the number of solutions of the system. In this section, we focus on these problems. We show here how to adapt steps 1 to 5 of the algorithm to the symbolic case.

In the sequel, we are working in the ring of polynomials $\mathbb{C}[x_1, \ldots, x_t]$.

3.1 Choice of the hidden variable

This first step consists in considering one variable as a parameter. Then the rest of the process will go on as if its value was known. It is easy to be convinced that the choice of this hidden variable will influence the execution of the algorithm. In fact, it will change the size of the square linear system constructed afterwards, so it will change its determinant, which is a polynomial in the hidden variable, in particular it will change the degree of this polynomial.

The only thing that can be done, given the present knowledge on this topic, is to propose heuristics of choice, such as:

- the variable of minimal degree among all the equations
- the variable whose sum of degrees in all the equations is minimal
- the variable that, when hidden, will lead to the smallest number of present monomials

On the systems we were given, we could notice that the most efficient heuristic was the second one.

3.2 Construction of a square system

There are two reasons to need a matrix with small size. The first one is practical: when the dimension of the system is too large, it is impossible to compute its determinant symbolically. As we will see, for the FKP problem, this will almost always be the case. The second reason is theoretical: if the size is not minimal, the determinant of the system will be a multiple of the resultant of the initial polynomial, and so spurious solutions will be counted. In fact there is no algorithm ensuring that the

[2] © Waterloo University (Canada) and *Waterloo Maple Software*

system it constructs will have minimal size. Roth proposes techniques to minimize the size of the system, but these techniques highly depend on the structure of the initial system, they only work for a restricted number of cases, and do not give any hint on the symbolic case.

There is no general rule to construct new equations, the only constraint being that each new equation must be independent of the already present equations. As previously said, the usual way of finding a new equation consists in multiplying one of the present equation by a monomial in the variables. Equations and monomials should be chosen in order to obtain a square system with reasonable size. Once more, there is no way to ensure that the obtained system will be of minimal size. For each choice of a pair equation/monomial, we can look at the number of new monomials that will be created and choose the smallest one. We can also consider the maximum degree of created monomials. Notice that when a pair does not create any new monomial, both strategies automatically choose it. It must in fact be chosen because it allows the process to converge to a square system.

Another way could also be inspired by the construction of the *bézoutian* [Dix08]: it is possible to make combinations between two equations in order to make their term of highest degree disappear.

3.3 Degree of the determinant of the system

As previously stated, the problem consisting in working on a particular system of equations can be solved numerically. However we want to give general bounds on the number of solutions of a system, so we keep everything symbolic.

Let us define now some notations that will be used in the sequel of this paper. y denotes the hidden variable and X_i, $(i \in \{1,\ldots,n\})$ represent the monomials that appeared in the equations. Our problem reduces to the linear system:

$$[\Delta_{i,j}(y)]_{1\leqslant i,j\leqslant n} \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = 0 \qquad (1)$$

$\Delta(y) = [\Delta_{i,j}]_{i,j\in\{1,\ldots,n\}}$ is a square matrix of size $n \times n$ whose entries are polynomials in y. If, for a given value y_0 of y, the determinant $|\Delta(y_0)|$ is not equal to 0, then the only solution of the system is $(X_1,\ldots,X_n) = (0,\ldots,0)$. But we want to consider only solutions of this system in the monomials that will give solutions of the initial system in the variables. The solution $(X_1,\ldots,X_n) = (0,\ldots,0)$ cannot give any solution in the variables since one of the monomials, say X_1, is the constant monomial 1. Thus, to yield solutions in the variables, the matrix $\Delta(y_0)$ must have rank at most $n-1$, that is y_0 must be a root of equation $|\Delta(y)| = 0$.

In practice, the determinants we obtain are symbolic and have large size ($\geqslant 16 \times 16$). No present system of symbolic computation (even BAREISS' method [GCL92]) is able to compute explicitly such determinants. Since we are looking for a bound on the number of solutions, we propose an algorithm [Tan95a, Tan95b] allowing us to find an upper bound on the degree of the determinant $|\Delta(y)|$. We will prove in Section 4 that this bound on the number of roots of $|\Delta(y)| = 0$ gives a bound on the number of solutions of the initial system.

An entry $\Delta_{i,j}(y)$ is the coefficient of monomial X_j in equation i of the square system, it is a polynomial in y. A first trivial bound on the degree of $|\Delta|$ is

$$\deg(|\Delta|) \leqslant \sum_{j=1}^{n} \max_{1 \leq i \leq n} \deg \Delta_{i,j}$$

which is far from being tight.

To obtain a better bound, we will examine one by one all the coefficients of the polynomial $|\Delta(y)|$, starting from the leading term, until we find a coefficient different from 0. Each of these coefficients is a determinant constructed from coefficients of Δ.

More precisely, if $a_{i,j}^{(d_j)}$ is the coefficient of y^{d_j} in the coefficient $\Delta_{i,j}$ of monomial X_j in equation i, that is

$$\Delta_{i,j}(y) = \sum_{d_j} a_{i,j}^{(d_j)} \, y^{(d_j)},$$

then the coefficient of y^d in $|\Delta(y)|$ is the sum of the following determinants

$$\Delta_{(d_1,\ldots,d_n)} = \begin{vmatrix} a_{1,1}^{(d_1)} & \cdots & a_{1,j}^{(d_j)} & \cdots & a_{1,n}^{(d_n)} \\ \vdots & & \vdots & & \vdots \\ a_{n,1}^{(d_1)} & \cdots & a_{n,j}^{(d_j)} & \cdots & a_{n,n}^{(d_n)} \end{vmatrix} \qquad \text{where } \sum_{j=1}^{n} d_j \;=\; d$$

If all the terms $\Delta_{(d_1,\ldots,d_n)}$ for all $(d_1,\ldots,d_n)$ such that $d_1 + \ldots + d_n > d$ equal 0, and if there is one $\Delta_{i,j}^{(d_1,\ldots,d_n)}$ such that $d_1 + \ldots + d_n = d$ that is different from zero, we know that $\deg(|\Delta|) \leq d$. As there is no way to check whether such a sum of determinants equals 0, we have no way to find the exact degree.

We only need know a method to test whether a symbolic determinant equals 0. To this aim, we developed an algorithm derived from Gaussian elimination.

An entry of $\Delta_{(d_1,\ldots,d_n)}$ is represented by

0 if $a_{i,j}^{(d_j)} = 0$

$\star$ otherwise

By a permutation on the lines of $\Delta_{(d_1,\ldots,d_n)}$, we choose a row with non zero pivot. We "subtract" this row from all the rows below it in which the element in the pivot's column is $\star$, in order to obtain 0 values for all the elements below the pivot in its column. The difference with classical Gaussian elimination lies in the way the subtraction is defined, since we are working on a symbolic matrix:

$\star - \star = \star$ on all the columns except the pivot's column

$\star - \star = 0$ on the pivot's column ($\boxed{\star}$)

$\star - 0 = \star = 0 - \star$

$0 - 0 = 0$

An illustration can be seen on the following example:

$$\begin{vmatrix} \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \star & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \boxed{\star} & 0 & \star & \star & \cdots \\ 0 & 0 & \star & 0 & 0 & \star & \cdots \\ 0 & 0 & 0 & 0 & \star & \star & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{vmatrix} \longrightarrow \begin{vmatrix} \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \star & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \boxed{\star} & 0 & \star & \star & \cdots \\ 0 & 0 & 0 & 0 & \star & \star & \cdots \\ 0 & 0 & 0 & 0 & \star & \star & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{vmatrix}$$

The algorithm stops when the matrix is triangular or when there is no more non 0 pivot. The rule $\star - \star = \star$ does not take notice of the possible relations between the coefficients that could lead to simplifications. These relations are in fact too complicated to be checked. So, when our algorithm answers that a determinant $\Delta_{(d_1,\ldots,d_n)}$ equals 0 its answer is correct, but when its answer is that a determinant is different from 0, it is not necessarily true. That is another reason why the bound we obtain on the degree of $|\Delta(y)|$ is only an upper bound.

In practice, as previously said, it is impossible to compute exactly such a symbolic determinant, so if we want to test whether the bound we find is tight, we can compute the determinants $\Delta_{(d_1,\ldots,d_n)}$ after replacing the parameters of the equations by random numeric values. Indeed, such a determinant is a multivariate polynomial in the parameters measured, so it can take value 0 only for a set of parameters with null measure. If we make several random tests, and if $\Delta_{(d_1,\ldots,d_n)}$ is not the zero polynomial, one of these test will return a non zero value, with probability 1.

We claim the following:

Theorem 1 *The degree of the polynomial $|\Delta(y)|$ obtained by the symbolic elimination method gives a bound on the number of solutions of the FKP.*

The proof is given in the next section.

4 Proof of correctness

This section is devoted to the proof of Theorem 1. In our case, since it is hopeless to solve such a general symbolic system (Equation 1), we only want to show bounds on the number of solutions. We will see that this step requires a particular attention in the symbolic case, and that a careful proof is necessary to show that Theorem 1 holds.

The X_i's are monomials in the parameters $x_j, y_j, z_j, (j \in \mathsf{B} \subseteq \{1, \dots, 4\})$ of the robot (the parameterization will be introduced into details in Section 4.2).

In this section, we first explain which problem might arise, we then explicit the equations we choose to represent the FKP, and we finally we prove the claim.

4.1 Problem

We know that there are at most $\deg(|\Delta(y)|)$ values of the hidden variable y that are roots of this determinant. However, this does not give us *a priori* a bound on the number of solutions of the initial system in the variables, because one value of y might lead to several solutions of the initial system. This problem can be illustrated on the following example:

Example 1 *Consider the system:*

$$\begin{aligned} x^2y - 6y + 1 &= 0 \\ x^2 - y &= 0 \end{aligned}$$

If y is hidden, we obtain a square system in the monomials 1 and x^2, there is no need for adding equations. Its determinant is $|\Delta(y)| = y^2 - 6y + 1$, with degree 2, and it admits two distinct positive roots y_1 and y_2. For each of these values, we then solve $x^2 = y$, which gives us two opposite solutions. Thus the system admits four distinct solutions (x_1, y_1), $(-x_1, y_1)$, (x_2, y_2), $(-x_2, y_2)$, though $|\Delta(y)|$ has degree 2.

The relation between the number of solutions of a polynomial system and the degree of an associated univariate polynomial was studied in the case of spare resultant [Emi94]. It was shown that, if the lattice generated by the Newton polytopes of the system was $\mathbb{Z}^n$, then there was no problem: the degree of the sparse resultant is equal to the mixed volume of the polytopes associated with the polynomials of the system; the degree then gives a bound on the number of solutions of the system. This can be illustrated on the preceding example.

Example 2 *On the same example, we can compute the resultant with respect to x:*

$$\begin{vmatrix} y & 0 & -6\,y+1 & 0 \\ 0 & y & 0 & -6\,y+1 \\ 1 & 0 & -y & 0 \\ 0 & 1 & 0 & -y \end{vmatrix} = (y^2 - 6y + 1)^2$$

Its roots are the same as the roots of our polynomial $|\Delta(y)|$, but here they have multiplicity 2. The number of solutions coincide with the degree of the resultants, which is also the sum of multiplicities of its roots.

The difference between $|\Delta(y)|$ and the resultant lies in the fact that all the monomials $1, x, x^2$ are considered to compute the resultant, whereas only the present monomials are considered to compute $|\Delta(y)|$.

As far as we know, this problem has never been solved in its generality for dialytic elimination (or for other *ad hoc* methods of elimination). In the literature, even for special systems, the question is often evaded, and the conclusion comes quickly without precise justification, that the degree of the univariate polynomial that is obtained by any elimination process gives a bound on the number of solutions. This might be false, as we saw on Example 1. Even if some properties of the equations ensure that this may be true in those particular cases, the result is however not immediate.

We do not intend to develop here a study that would yield as many general results as there are for the case of sparse resultants for example [Emi94]. However we will prove that, in the case of the equations of the FKP for parallel robots, with extra sensors, the degree of $|\Delta(y)|$ gives a bound on the number of possible postures.

The remark we made in Example 2 on the multiplicities of the roots of the resultant suggests further reflections. Nothing implies at once that our determinant is a multiple of the resultant, this is even false on the example. We know that, for generic systems, the existence of a root on the system implies the existence of a root of $|\Delta(y)|$. This allows us to say that, under genericity assumption, each root of the resultant is a root of $|\Delta(y)|$. But, as in Example 1-2, the multiplicity of a root in

$|\Delta(y)|$ can be smaller than its multiplicity in the resultant, so that the degree of $|\Delta(y)|$ is smaller, for the same number of solutions to the initial system.

However, by noticing that some precise monomials, such as the initial variables, appear in the system, because they appear in the initial system (that will be described in the following Section 4.2), we can prove Theorem 1. The proof is rather complex, and uses some known results on the FKP of parallel robots, and thus it cannot be extended trivially to other problems. Indeed we cannot use directly the results stated in [Laz83] since they require a lot of monomials to appear in the system, and we cannot ensure that these monomials are actually present.

4.2 Algebraic parameterization of the FKP

We use a formulation of the FKP proposed by LAZARD [Laz93], because it introduces a small number of monomials and some linear equations. Some joints on the platform are used to form a basis of this platform. A set B of 4 joints is needed in the case of a three dimensional platform (another formulation using only 3 points is also proposed in [Laz93], but it introduces equations of larger degree), while a set B with only 3 joints is enough in the case of a planar platform.

The unknowns will be the coordinates of the vectors A_iB_i for $i \in \mathsf{B}$:

$$\forall i \in \mathsf{B} \;\; A_iB_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}$$

For a planar platform $\mathsf{B} = \{1, 2, 3\}$ and there are 9 unknowns, for a general platform $\mathsf{B} = \{1, 2, 3, 4\}$ and there are 12 unknowns. The equations are the following (the reader will find details on the way they are obtained in [Laz93])

I.

$$\|A_iB_i\|^2 = \rho_i^2 \qquad (\forall i \in \mathsf{B})$$

II.

$$\|B_iB_j\|^2 = K_{ij} \quad (\forall (i, j) \in \mathsf{B}^2, \; i \neq j)$$

III.

$$\|A_iB_i\|^2 = \rho_i^2 \qquad (\forall i \notin \mathsf{B})$$

This can be expressed using the unknowns A_iB_i, for $i \in \mathsf{B}$. In the general case this becomes

$$\rho_i^2 = (-A_1A_i + A_1B_1 + e_i\, B_1B_2 + f_i\, B_1B_3 + g_i\, B_1B_4)^2$$

and for a planar platform

$$\rho_i^2 = (-A_1A_i + A_1B_1 + e_i\, B_1B_2 + f_i\, B_1B_3)^2$$

e_i, f_i, g_i (resp. e_i, f_i for a planar platform) for $i \notin \mathsf{B}$ are the coordinates of B_i in frame $(B_1,\, B_1B_2,\, B_1B_3,\, B_1B_4)$ (resp. $(B_1,\, B_1B_2,\, B_1B_3)$), they only depend on the geometry of the robot.

The non linear terms that appear in this equation are linear combinations of those appearing in equations of types I and II. The equations of type III thus become linear by subtractions of equations of the preceding types.

IV. Informations from the extra sensors, for some $i \in \{1, \ldots, 6\}$:

$$A_iB_i \cdot N_i = 0$$

In the sequel, particularly in Lemma 8 and in Section 5 we will often make the assumption that none of the extra sensors is redundant, which means that they all actually give us some information. This allows us to say that the equations of type IV are independent of the equations of the other types, and that they are mutually independent too. With this property, we can solve, as done in Section 5, the linear system formed by the linear equations of types IV and III to reduce the number of unknowns.

4.3 Degree of $|\Delta(y)|$ and number of postures

The proof of Theorem 1 is done in two steps: a first step consists in proving a relation between the multiplicity of a root y_0 of $|\Delta(y)|$ and the rank of matrix $\Delta(y_0)$. Then, in a second step, by using the shape of the monomials that are present in our system, we show that, for each root y_0 of the polynomial, the number of postures solutions of the FKP, for which the hidden variable y has value y_0 is bounded by the multiplicity of y_0 in $|\Delta(y)|$.

Example 3 *Let us have a look at the following system:*

$$\begin{aligned} a\,x^2y + b\,y + c &= 0 \\ d\,x\,y + e\,y &= 0 \end{aligned}$$

Dialytic elimination leads to a matrix $\Delta(y)$ whose determinant is:

$$\begin{vmatrix} by+c & 0 & ay \\ e\,y & d\,y & 0 \\ 0 & e\,y & d\,y \end{vmatrix} = y^2\left(y\,bd^2 + ye^2a + cd^2\right)$$

$y = 0$ is a root of multiplicity 2, and the rank of $\Delta(0)$ is $1 = 3 - 2$.

$$\begin{vmatrix} c & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{vmatrix}$$

When the rank becomes 0, then $c = 0$, and the multiplicity of the root $y = 0$ becomes 3.

Let us show the following lemma that states the general result corresponding to this example.

Lemma 2 *If the rank of $\Delta(y_0)$ is less or equal to $n - \nu$, then the multiplicity of y_0 in the polynomial $|\Delta(y)|$ is at least equal to ν.*

Proof: When $\nu = 1$ we already noticed that, if the rank of the matrix is less of equal to $n-1$, then trivially y_0 is a root of $|\Delta(y)|$, thus its multiplicity is at least 1.

Let us first show the result for the case of a matrix $\Delta(y_0)$ with rank less of equal to $n-2$ for a given value y_0.

$\Delta(y)$ is the matrix of an endomorphism in a fixed basis $(e_1, \ldots, e_n)$. $\Delta(y_0)$ is not invertible, thus there exists one vector $v_0 \neq 0$ in its kernel. Form the theorem of incomplete basis of elementary linear algebra, we can find $n-1$ vectors in the basis $(e_1, \ldots, e_n)$ forming a new basis with v_0. By renumbering the elements of the basis, let us assume that $(e_1, \ldots, e_{n-1}, v_0)$ is the basis we obtain. In this basis, $\Delta(y)$ becomes

$$\Delta_0(y) = P_0 \Delta(y) P_0^{-1}$$

where P_0 is the matrix connecting the two bases $(e_1, \ldots, e_n)$ and $(e_1, \ldots, e_{n-1}, v_0)$. Of course we have:

$$|\Delta_0(y)| = |\Delta(y)|$$

$\Delta_0(y)$ looks like

$$\begin{pmatrix} \cdot & \cdots & \cdot & p_1(y) \\ \vdots & \vdots & \vdots & \vdots \\ \cdot & \cdots & \cdot & p_n(y) \end{pmatrix}$$

Since v_0 lies in the kernel of $\Delta_0(y_0)$, all the polynomials $p_1(y), \ldots, p_n(y)$ have y_0 as a root, so they can be divided by $y - y_0$.

Let us now expand the determinant of $\Delta_0(y)$ according to the last row. Let $M_i(y)$ be the determinant obtained by deleting the last column and the i^{th} row in $\Delta_0(y)$. We obtain:

$$\begin{aligned} |\Delta_0(y)| &= \sum_{i=1}^{n}(-1)^{n+i-1} p_i(y) M_i(y) \\ &= (y - y_0) \sum_{i=1}^{n}(-1)^{n+i-1} \frac{p_i(y)}{y - y_0} M_i(y) \end{aligned}$$

By hypothesis, the rank of $\Delta_0(y_0)$ is strictly less than $n-1$, so all the sub-determinants of size $(n-1) \times (n-1)$ extracted from $\Delta_0(y_0)$ are equal to 0. In particular all the $M_i(y)$'s are equal to 0 when $y = y_0$, thus each polynomial $M_i(y)$ is divisible by $y - y_0$, which implies:

$$|\Delta_0(y)| = (y - y_0)^2 \sum_{i=1}^{n}(-1)^{n+i-1} \frac{p_i(y)}{y - y_0} \frac{M_i(y)}{y - y_0}$$

So the multiplicity of y_0 is at least 2 in $|\Delta(y)|$.

Let us do the same proof again, with the hypothesis now that the rank of $\Delta(y_0)$ is less or equal to $n - \nu$. We now have a basis $v_1, \ldots, v_\nu$ of its kernel, completed into a basis $(e_1, \ldots, e_{n-\nu}, v_1, \ldots, v_\nu)$, after a proper renumbering of the elements of $(e_1, \ldots, e_n)$. Then

$$\Delta_0(y) = P_0 \Delta(y) P_0^{-1}$$

and $\Delta_0(y)$ can be written as

$$\begin{pmatrix} \cdot & \cdots & \cdot & p_{1,1}(y) & p_{1,2}(y) & \cdots & p_{1,\nu}(y) \\ \cdot & \cdots & \cdot & p_{2,1}(y) & p_{2,2}(y) & \cdots & p_{2,\nu}(y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdot & \cdots & \cdot & p_{n,1}(y) & p_{n,2}(y) & \cdots & p_{n,\nu}(y) \end{pmatrix}$$

where the polynomials $p_{i,j}$, $i \in \{1, \ldots, n\}, j \in \{1, \ldots, \nu\}$ all have root y_0 since $v_1, \ldots, v_\nu$ belong to the kernel of $\Delta_0(y_0)$.

Let us now compute $|\Delta_0(y)|$ by the classical definition using all the permutations of the set $\mathcal{S}_n$ of permutations in $\{1, \ldots, n\}$. $\varepsilon(\sigma)$ denotes the signature of permutation

σ.

$$
\begin{aligned}
|\Delta_0(y)| &= \sum_{\sigma\in\mathcal{S}_n} \varepsilon(\sigma)\Delta_{\sigma(1),1}\Delta_{\sigma(2),2}\cdots\Delta_{\sigma(n),n} \\
&= \sum_{\sigma\in\mathcal{S}_n} \big[\varepsilon(\sigma)\Delta_{\sigma(1),1}\Delta_{\sigma(2),2}\cdots\Delta_{\sigma(n-\nu),n-\nu} \\
&\qquad\qquad \cdot\, \Delta_{\sigma(n-\nu+1),n-\nu+1}\cdots\Delta_{\sigma(n),n}\big]
\end{aligned}
$$

The last ν elements

$$\Delta_{\sigma(n-\nu+1),n-\nu+1}, \Delta_{\sigma(n-\nu+2),n-\nu+2}, \ldots, \Delta_{\sigma(n),n}$$

of the matrix are polynomials for which y_0 is necessarily a root since these polynomials lie among $p_{1,1}(y), \ldots, p_{1,\nu}(y), \ldots, p_{n,1}(y), \ldots, p_{1,\nu}(y)$. $y - y_0$ is thus a factor of all these ν elements, which allows us to write $(y-y_0)^\nu$ as a factor in $|\Delta(y)|$. y_0 is thus a root of $|\Delta(y)|$ with multiplicity greater or equal to ν (possibly strictly greater than ν). $\diamondsuit$

Now we must relate the rank of $\Delta(y_0)$ with the number of global solutions of the FKP, in the initial unknowns $x_i, y_i, z_i, i \in \mathsf{B}$, for which the hidden variable y has value y_0. To make everything clearer, let us assume that y is one of the coordinates of vector A_1B_1. It is interesting to distinguish two cases in which we have at least two global solutions for the same value y_0 of y.

In the first case, the global solutions we obtain in this way correspond to at least 2 distinct positions of the point B_1, which means that the coordinate axes have been chosen so that two positions for B_1 are lying in a same plane parallel to a coordinate plane. It is then possible to perturb the axes slightly so that this does not happen. Notice that our study remains valid, since we did not make any assumption so far on the choice of the axes.

In the second case, the case in which we will be from now on, if several solutions are possible for the posture of the robot for a given value y_0 of y, then for all these postures, the point B_1 will have the same position.

Thus the idea of fixing a value y_0 for y and perturbing the axes allows us to assume now that the position of one point of the platform is known.

We would like to show that the degree of $|\Delta(y)|$ gives a bound on the number of solutions of the FKP. Let us state the following conjecture:

> *Conjecture : if the rank of $\Delta(y_0)$ is $n-\nu$, then the system admits at most ν solutions for the value y_0 of the hidden variable.*

Then, each root would lead to a number of solutions that would be bounded by its multiplicity in the polynomial, so by summing the multiplicity, the result would be obtained.

We could not prove this conjecture in the general case, but only for the case of the FKP, which allows us to prove Theorem 1 as will be seen in the rest of this section. The idea of the proof is the following:

For a given root y_0 of $|\Delta(y)|$, we can assume as previously said that B_1 is known. Then we can use some known results [Tan95a] on the number of postures, depending on the robot's architecture. There is ν_0 depending on the architecture of the robot, such that, the number of postures is bounded by ν_0. To our aim, it is enough to show the conjecture for $\nu < \nu_0$.

More precisely,

Lemma 3 *[Tan95a, TTM95a] When the platform of the robot is planar, and when one point B_1 is known on the platform, the number of postures is bounded by 2 in general.*

The bound may become 4 or 8 for some special architectures, for example when there is $i \in \{2, \ldots, 6\}$ such that $B_i = B_1$, or when $B_3, \ldots, B_6$ and B_1 are collinear, or when $A_2, \ldots, A_6$ and B_1 are coplanar.

When B_1 is known, and when one additional sensor is placed on the robot, there is in general a unique solution to the FKP.

The bound for this case is 4, and it may be reached for the special architectures described just above.

So, for the general case, when the platform is planar, it is enough to prove the conjecture for rank $n-1$.

We do not have an exhaustive list of architectures and configurations in which the bound is 4 or 8, such architectures are very particular, but we can only give sufficient conditions under which the bound is greater than the general bound, equal to 2 ([Tan95a, TTM95a]).

Lemma 4 *[Tan95a, TTM95a] For a general platform, if one point B_1 is known on the platform, and if an additional sensor is placed on another segment, then the bound on the number of solutions of the FKP is 4.*

For this case, we need to prove the conjecture until rank $n-3$. We will see that it is necessary to assume that the hidden variable is not among the variables relative to the segment on which the additional sensor is placed.

In both cases, we prove a theorem stating the result.

The following lemma gives the first intermediate result in the case when the rank of the matrix is $n-1$. It is independent of the FKP.

Lemma 5 *For a system in which all monomials of degree 0 and 1 are present, if the rank of $\Delta(y_0)$ equals $n-1$, then there is at most one solution to the system for which the hidden variable has value y_0.*

Proof: In fact if $rank(\Delta(y_0)) = n-1$ for $y = y_0$ then the kernel of $\Delta(y_0)$ is a vector space of dimension 1, generated by $(u_1, u_2, \ldots, u_n)$, where u_i depends of y_0.

Let us go back to the variables contained in the monomials. In our case (see Section 4.2) all the monomials reduced to variables are present. Let us suppose that X_1 be the constant 1, X_2 the variable x_2, X_3 the variable y_2, ... by a renumbering of monomials. The fact that $(X_1, \ldots, X_n)$ is an element of the kernel can be expressed by the existence of a number λ such that

$$\begin{aligned} 1 &= \lambda u_1 \\ x_2 &= \lambda u_2 \\ y_2 &= \lambda u_3 \\ \vdots & \quad \vdots \quad \vdots \end{aligned}$$

u_1 cannot be equal to 0, so a unique value can be deduced for λ. By replacing λ by its value in the following equations, we obtain at most a unique solution for each variable. Some equations can be inconsistent with other ones,so it is also possible to obtain no solution to the system in the variables. $\diamondsuit$

The preceding lemma is quite general, since the only hypothesis that is made concerns the presence of the variables and of the constant monomial. The following lemmas will use the architecture of parallel robots.

The rest of the proof is given first for the case of a planar platform (Section 4.3.1). Then in Section 4.3.2 we will see that the proof is more complicated for a general robot.

4.3.1 Case of a planar platform

As we saw in Lemma 3 that for a planar platform, there are only two solutions in general to the FKP when the position of B_1 is known.

To summarize what has been said, if the rank of $\Delta(y_0)$ is $n-1$, then the root y_0 is simple in the polynomial $|\Delta(y)|$, and there is at most one posture that is solution of the FKP. If the rank of $\Delta(y_0)$ is less or equal to $n-2$, from this lemma 3 the number of postures is at most 2, and from Lemma 2 the root has multiplicity at least 2.

We deduce the following theorem:

Theorem 6 *The number of solutions of the system, and thus the number of possible postures for the robot, for a robot with planar platform, is bounded by the degree in y of $|\Delta(y)|$, except for some very special architectures indicated in Lemma 3.*

4.3.2 Case of a general platform

We saw in Lemma 4 that when a joint is known, and when an additional information is given by another sensor on another leg, then there are at most 4 solutions to the FKP.

We must still show that for $\nu = 2, 3$, if the rank of $\Delta(y_0)$ is $n - \nu$, then the number of solutions of the FKP for $y = y_0$ is bounded by ν. Theorem 6 will then be generalized to the case of a general platform.

Lemma 7 *On the equations of the FKP, if the rank of $\Delta(y_0)$ is $n-2$, then there are at most two solutions of the system for which y has the same value y_0.*

Proof: If $rank(\Delta(y_0)) = n-2$, then the kernel of $\Delta(y_0)$ has dimension 2, and it is generated by two vectors $U(y_0) = (u_1, u_2, \ldots, u_n)$ and $V(y_0) = (v_1, v_2, \ldots, v_n)$ whose coordinates depend on y_0. We can write, as in the preceding lemma, the following equalities in which we are interested in priority in the variables representing the coordinates of vector A_2B_2

$$
\begin{aligned}
1 &= \lambda u_1 + \mu v_1 && (2)\\
x_2 &= \lambda u_2 + \mu v_2 && (3)\\
y_2 &= \lambda u_3 + \mu v_3 && (4)\\
z_2 &= \lambda u_4 + \mu v_4 && (5)\\
x_3 &= \lambda u_5 + \mu v_5 && (6)\\
\vdots &\ \ \vdots\ \ \vdots
\end{aligned}
$$

Of course this assumes that we possibly renumbered the monomials.

In the first equality, u_1 and v_1 cannot be simultaneously equal to 0. Let us assume that $v_1 \neq 0$ for example. Then we can deduce that

$$\mu = \frac{1 - \lambda u_1}{v_1} \tag{7}$$

which allows us to write:

$$x_2 = \lambda \left(u_2 - \frac{u_1}{v_1} v_2 \right) + \frac{v_2}{v_1} \tag{8}$$

$$y_2 = \lambda \left(u_3 - \frac{u_1}{v_1} v_3 \right) + \frac{v_3}{v_1} \tag{9}$$

$$z_2 = \lambda \left(u_4 - \frac{u_1}{v_1} v_4 \right) + \frac{v_4}{v_1} \tag{10}$$

Then we can have two cases:

∘ First case: Equations 8, 9, 10 define the parametric equations of a line. This line gives the direction of vector A_2B_2.

Notice that we defined $U(y_0)$ and $V(y_0)$ without using the shape of the equations given by the FKP. We also looked at the rank of the matrix, but we did not used its shape, which is directly related to the equations of the problem.

If we now use the equation representing the length of the second leg, $x_2^2 + y_2^2 + z_2^2 = \rho_2^2$, and the preceding expressions we obtained for x_2, y_2, z_2, we obtain an equation of degree 2 in λ. This equation cannot be the trivial equation $0 = 0$, so it can only have two solutions in λ, which means that A_2B_2 can only take two positions, if we know its direction and its length. The two values of λ we obtain in that way allow us to deduce μ from equality 7, then by replacing into equations 6 and the following, we determine at most two postures for the robot.

∘ Second case: Equations 8, 9, 10 define a unique point (when the coefficients of λ are all equal to 0). In this case, the positions of both B_1 and B_2 are known, so there are at most two solutions, except when the robot lies in a singular posture (see [Tan95a, TTM95a]).

In both cases, the number of postures is bounded by 2. ◊

We did not use so far the fact that there was an additional sensor. This will be required for the proof of the next lemma.

Lemma 8 *On the equations of the FKP for parallel robots, for a robot with general platform, when an additional sensor is placed on one segment, then if the rank of $\Delta(y_0)$ is $n-3$, there are at most 3 solutions for which y has the same value y_0.*

In fact this result holds as soon as the sensor is placed on a segment, say 2, that does not correspond to the hidden variable, and for a posture where B_1, B_2, A_3, A_4, A_5, A_6 *are not coplanar.*

Proof: The general scheme of the proof is the same as the previous one.

If $rank(\Delta(y_0)) = n-3$, we write the following equations in which $U(y_0) = (u_1, u_2, \ldots, u_n)$, $V(y_0) = (v_1, v_2, \ldots, v_n)$ and $W(y_0) = (w_1, w_2, \ldots, w_n)$ form a basis of the kernel of $\Delta(y_0)$:

$$\begin{aligned}
1 &= \lambda u_1 + \mu v_1 + \nu w_1 &(11)\\
x_2 &= \lambda u_2 + \mu v_2 + \nu w_2 &(12)\\
y_2 &= \lambda u_3 + \mu v_3 + \nu w_3 &(13)\\
z_2 &= \lambda u_4 + \mu v_4 + \nu w_4 &(14)\\
x_3 &= \lambda u_5 + \mu v_5 + \nu w_5 &(15)\\
y_3 &= \lambda u_6 + \mu v_6 + \nu w_6 &(16)\\
z_3 &= \lambda u_7 + \mu v_7 + \nu w_7 &(17)\\
\vdots & \quad \vdots \quad \vdots &
\end{aligned}$$

By a possible renumbering of the indices, we assume that the second segment is equipped with the additional sensor, which is possible if there is an additional sensor on a segment for which none of the coordinates is the hidden variable y (remember that the hidden variable is one coordinate of $A_1 B_1$).

For example $w_1 \neq 0$, hence

$$\nu = \frac{1 - \lambda u_1 - \mu v_1}{w_1} \tag{18}$$

which allows us to write:

$$\begin{aligned}
x_2 &= \lambda\left(u_2 - \frac{u_1}{w_1}w_2\right) + \mu\left(v_2 - \frac{v_1}{w_1}w_2\right) + \frac{w_2}{w_1} &(19)\\
y_2 &= \lambda\left(u_3 - \frac{u_1}{w_1}w_3\right) + \mu\left(v_3 - \frac{v_1}{w_1}w_3\right) + \frac{w_3}{w_1} &(20)\\
z_2 &= \lambda\left(u_4 - \frac{u_1}{w_1}w_4\right) + \mu\left(v_4 - \frac{v_1}{w_1}w_4\right) + \frac{w_4}{w_1} &(21)
\end{aligned}$$

These equations define a plane, or a line, or a point.

∘ First case : the equations 19, 20, 21 are the parametric representation of a plane P containing vector A_2B_2. The equation of length for segment 2 (type I), together with the equation expressing that $\|B_1B_2\|$ is constant (type II) allow us to deduce that B_2 lies on a circle $\mathcal{C}_2$. This circle might be contained in P. But then, if the sensor gives some non redundant information, P cannot be the same plane as the plane measured by the sensor (otherwise the sensor would be useless). So in all cases, the intersection of the circle with one plane gives at most two points for B_2.

This can be expressed equivalently on λ and μ by saying that theses parameters satisfy both an equation of degree 2 (equation of a sphere for B_2) and a linear equation (equation of the plane given by the sensor). After eliminating μ from the linear equation, this yields an equation of degree 2 for λ, so at most two solutions for the pair (λ, μ). The equations 15 and the following, together with equality 18 then give at most two postures for the robot.

∘ Second case: the equations 19, 20, 21 define a line. This happens when the two vectors generating the above plane are not independent, that is *wlog* when there is a constant k such that

$$\begin{aligned}
\left(u_2 - \frac{u_1}{w_1}w_2\right) &= k\left(v_2 - \frac{v_1}{w_1}w_2\right)\\
\left(u_3 - \frac{u_1}{w_1}w_3\right) &= k\left(v_3 - \frac{v_1}{w_1}w_3\right)\\
\left(u_4 - \frac{u_1}{w_1}w_4\right) &= k\left(v_4 - \frac{v_1}{w_1}w_4\right)
\end{aligned}$$

So the line containing B_2 has equation

$$\begin{aligned}
x_2 &= (k\lambda + \mu)\left(v_2 - \frac{v_1}{w_1}w_2\right) + \frac{w_2}{w_1}\\
y_2 &= (k\lambda + \mu)\left(v_3 - \frac{v_1}{w_1}w_3\right) + \frac{w_3}{w_1}\\
z_2 &= (k\lambda + \mu)\left(v_4 - \frac{v_1}{w_1}w_4\right) + \frac{w_4}{w_1}
\end{aligned}$$

The intersection of this line with $\mathcal{C}_2$ (defined as in the first case) gives at most 2 points B_2^1 and B_2^2, corresponding to 2 values α^1 and α^2 of the parameter $k\lambda + \mu$. So

$$\begin{aligned}
\mu^1 &= \alpha^1 - k\lambda\\
\mu^2 &= \alpha^2 - k\lambda
\end{aligned}$$

This allows us to eliminate μ in the equations 15, 16 and 17corresponding with point B_3. We know that when the positions of 2 joints of the platform are known, there are at most 2 possible postures [Tan95a, TTM95a]. So there will be 2 positions $B_3^{i\prime}$ and $B_3^{i\prime\prime}$ when the position of B_1 and the position B_2^i of B_2 are known, for $i = 1, 2$, which gives at most 4 postures for the robot. Let us study in which cases these 4 postures can really be solutions of the problem.

It is easy to check that Equations 15, 16 and 17, for each value μ^1 and μ^2, become respectively the equations of lines D_3^1 (containing points $B_3^{1\prime}$ and $B_3^{1\prime\prime}$) and D_3^2 (containing $B_3^{2\prime}$ and $B_3^{2\prime\prime}$), and that these 2 lines are parallel.

We also know that B_3 must lie on a circle $\mathcal{C}_3$, intersection of the sphere of center A_3 and radius ρ_3, and of the sphere of center B_1 and radius $\|B_1B_3\|$. $B_3^{i\prime}$ and $B_3^{i\prime\prime}$ also lie on the sphere of center B_2^i and radius $\|B_2B_3\|$, for $i = 1, 2$. So the points B_1, B_2^1 and A_3 belong to the bisecting plane of $B_3^{1\prime}$ and $B_3^{1\prime\prime}$. In the same manner, B_1, B_2^2 and A_3 belong to the bisecting plane of $B_3^{2\prime}$ and $B_3^{2\prime\prime}$. Moreover, as D_3^1 and D_3^2 are parallel, these two planes are identical. So B_1, B_2^1, B_2^2 and A_3 are coplanar.

If the same reasoning is repeated for points B_4, B_5, B_6, we conclude that, in order to have 4 possible postures, B_1, B_2^1, B_2^2, A_3, A_4, A_5, A_6 must be coplanar.

In all the other cases, there will not be more than 3 solutions.

∘ Third case: Equations 19, 20, 21 define a unique point. The conclusion then comes as in the preceding lemma.

In all cases (except the indicated special case) the number of postures is bounded by 3. ◇

So we can now deduce the final theorem:

Theorem 9 *The number of solutions of the system, and thus the number of possible postures for the FKP with extra sensors, for a robot with general platform, is bounded by the degree in y of the determinant $|\Delta(y)|$, except for the special configurations indicated in Lemma 8.*

4.3.3 Conclusion

Theorem 1 is now proved (except for some very particular cases).

As previously noticed, this bound is only an upper bound, and it is probably not tight, for several reasons : first we only obtain an upper bound for the degree of $|\Delta(y)|$; second, the number of real roots of the polynomial may be inferior to the number of its complex roots given by its degree; finally $|\Delta(y)|$ is a multiple of the resultant, it may have spurious roots that will not lead to any solution of the system, and even if a root leads to a solution, its multiplicity may be smaller.

5 Results

In this section, the symbolic method proposed previously will be applied and we will give bound on the number of solutions of the FKP for a robot equipped with 1 to 6 additional sensors.

We tried several parameterization of the problem, and we chose Lazard's parameterization (Section 4.2), because it gives the best results. Elimination methods have many advantages, but they are very sensitive to the number of unknowns and the degree of the equations. This parameterization does not increase the degrees of equations, it even supplies us with linear equations, and it gives a small number of monomials, compared with other possible parameterizations.

It is interesting to have as least unknowns as possible in the starting system. We noticed that it was very helpful to eliminate some of the variables by solving the linear equations (type III and IV). This allows us to remove a lot of monomials, as shown in Table 1 in the case of a planar platform. The size of the square system we obtain by the symbolic elimination is smaller when the number of monomials is smaller. And fortunately this does not spoil the proof of correctness, since all the eliminated variables are linear combinations of the non eliminated ones, and then all the equations written in the proofs of the lemmas still hold.

Sensors	3	2	1
Before linear elimination		28	
After linear elimination	10	15	21

Table 1: Number of monomials present in the equations before and after the resolution of the linear equations

5.1 Planar platform

Table 2 gives bounds on the number of solutions, depending on the number of extra sensors that are added on the robot. They also give the number of unknowns in the initial non-linear system, the number of equations in the square system obtained by the elimination method, and the trivial bound on the degree of its determinant, from which we derive a tighter bound as explained in Section 3.3. The symbolic Gaussian elimination was used to this aim, except for the case when only one sensor is used, which leads to too large a computation time: it only gave us a bound equal to 27, and we obtained the bound 20 by taking random numerical parameters of the robot for the evaluation of the degree of the determinant of the dialytic matrix. Results obtained with another representation than Lazard's one can be found for comparison in [Tan95a], they are far less good.

Sensors	Unknowns	Dim matrix	Trivial bound	Final bound
6	0*	NS**	NS	1
5	1	NS	NS	2
4	2	NS	NS	2
3	3	4	6	4
2	4	16	14	9
1	5	64	41	20

* After solving the linear equations, there is no more unknown since there are (6+3) linear equations and 9 unknowns.

** Non significant data: it does not pay to build a dialytic matrix

Table 2: Bounds on the number of solutions of the FKP for a robot with planar platform (9 unknowns)

When there are more than 3 sensors (we always assume that the sensors are not redundant, which means that they actually give information), it is not interesting to build the dialytic matrix. Indeed it is better to solve the non-linear system by taking advantage of its structure when the linear equations have been eliminated. We obtain in this way a better bound. If 6 sensors are used and give information, we obtain a unique solution by solving the linear system corresponding to the 6 equations of type IV given by the sensors and the 3 equations of type III.

The CPU times given in Table 3 are the times we needed to obtain the bound with the symbolic method of Section 3.3. They are only indicative. In fact in practice this computation is not done since we only want to compute numerically the result.

Degree	39	38	37	36	35	34	33	32
CPU time (s)	0,26	0,85	3,47	13,7	50	161	488	1488

Table 3: Symbolic test whether the coefficient of the term of a given degree in $|\Delta(y)|$ equals 0, for one additional sensor and a planar platform (SS10 75 MHz)

5.2 General platform

For a general platform, the sizes of the matrices grow very quickly, and the bounds we obtain are much higher than in the previous case. For example, Table 4 gives values for 5 and 6 additional sensors.

Sensors	Unknowns	Dim matrix	Trivial bound	Final bound
6	4	10	5	5
5	5	31	18	11

Table 4: Bounds on the number of solutions of the FKP for a robot with general platform (12 unknowns)

6 Conclusion

Some difficulties arise in the symbolic method we presented in this paper.

It is important to minimize the size of the square system we obtain. But there is no way to determine *a priori* which would be the optimal size.

Up to now, the symbolic computation of the degree is very long because it consists in a recursive process involving very large matrices. As previously said, we can compute the degree by random parameters for the robot. Then, numerically, the computation of the determinant can be avoided and replaced by methods of eigenvalues, as used in [ZA94] for parallel robots (see also [RR95] for details).

In spite of these difficulties, we proved that this symbolic method gives actual bounds on the number of solutions of the FKP of parallel robots, with additional sensors. We presented the results obtained with this method.

We leave the conjecture presented in Section 4.3 as an open question for general systems of algebraic equations. Does it hold for example if all the monomials of degree less than some constant appear in the system ?

Another important advantage of this method is that it allows the symbolic matrix to be re-used again: it can be computed off-line symbolically for a symbolic robot, and then, if numeric solvings are needed for numerous cases of different architectures and configurations, each of these computations can be done on-line using the pre-computed matrix.

Acknowledgements

We thank Jean-Pierre Merlet for introducing the field of parallel robots to us. We are grateful to Bernard Mourrain and Ioannis Emiris for helpful discussions.

References

[Ber75] D.N. Bernstein. The number of roots of a system of equations. *Funktsional'nyi Analiz i Ego Prilozhenyia*, 9:1–4, 1975. Translated from russian.

[Can88] J.F. Canny. Elimination theory. In *The complexity of robot motion planning*, chapter 3. MIT Press, Cambridge, MA, USA, 1988.

[CR89] S. Charentus and M. Renaud. Calcul du modèle géométrique direct de la plate-forme de Stewart. Technical Report 89260, LAAS, Toulouse, France, July 1989.

[Dix08] A.L. Dixon. The eliminant of three quantics in two independent variables. In *Proceedings of the London Mathematical Society*, volume 7 of *2*, pages 49–69, London, UK, 1908.

[EC96] I.Z. Emiris and J.F. Canny. Efficient incremental algorithms for the sparse resultant and the mixed volume. *Journal of Symbolic Computation*, 11:1–33, 1996.

[Emi94] I.Z. Emiris. *Sparse Elimination and Applications in Kinematics*. Ph. D. thesis, University of California at Berkeley, Berkeley California, USA, 1994.

[FL95] J-C. Faugère and D. Lazard. The combinatorial classes of parallel manipulators. *Mechanism and Machine Theory*, 30(6):765–776, August 1995.

[GCL92] K.O. Geddes, S.R. Czapor, and G. Labahn. *Algorithms for computer algebra*. Kluwer Academic Publishers, Boston, 1992.

[HS96] B. Huber and B. Sturmfels. Bernstein's theorem in affine space. *Discrete and Computational Geometry*, 1996. to appear.

[Hus94] M.L. Husty. An algorithm for solving the direct kinematics of Stewart-Gough-type platforms. Technical Report TR-CIM-94-7, McGill University, Montréal, Canada, June 1994.

[IP90] C. Innocenti and V. Parenti-Castelli. Direct position analysis of the Stewart platform mechanism. *Mechanism and Machine Theory*, 25(6):611–621, 1990.

[IP91] C. Innocenti and V. Parenti-Castelli. Direct kinematics of the 6-4 fully parallel manipulator with position and orientation uncoupled. In *European Robotics and Intelligent Systems Conference*, Corfu, Greece, June 1991.

[IP93] C. Innocenti and V. Parenti-Castelli. Direct kinematics in analytical form of a general 5-4 fully-parallel manipulator. In J. Angeles, G. Hommel, and P. Kovács, editors, *Computational Kinematics*, pages 141–152. Kluwer Academic Publishers, 1993.

[Laz83] D. Lazard. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *EUROCAL*, number 162 in LNCS, pages 146–156. Springer Verlag, 1983.

[Laz92] D. Lazard. Stewart platform and Gröbner basis. In *Advances in Robot Kinematics*, pages 136–142, Ferrare, Italy, September 1992.

[Laz93] D. Lazard. Generalized Stewart Platform: How to compute with rigid motions? In *IMACS*, pages 85–88, Lille, France, May 1993.

[LM94] D. Lazard and J-P. Merlet. The (true) Stewart platform has 12 configurations. In *International Conference on Robotics and Automation*, pages 2160–2165, San Diego, California, USA, May 1994. IEEE.

[Mer93] J-P. Merlet. Closed-form resolution of the direct kinematics of parallel manipulators using extra sensors data. In *International Conference on Robotics and Automation*, pages 200–204, Atlanta, USA, May 1993. IEEE.

[Mou93] B. Mourrain. The 40 generic position of a parallel robot. In M. Bronstein, editor, *ISSAC'93*, ACM press, pages 173–182, Kiev, Ukrain, July 1993.

[Rag91] M. Raghavan. The Stewart platform of general geometry has 40 configurations. In ASME *Design and Automation Conference*, volume 32-2, pages 397–402, Chicago, IL, USA, September 1991.

[Rot93] B. Roth. Computations in kinematics. In J. Angeles, G. Hommel, and P. Kovács, editors, *Computational Kinematics*, pages 3–14. Kluwer Academic Publishers, 1993.

[RR95] M. Raghavan and B. Roth. Solving polynomial systems for the kinematic analysis and synthesis of mechanisms and robot manipulators. *Transaction of the* ASME *Journal of Mechanical Design*, 117(2):71–78, 1995.

[SA91] R. Stoughton and T. Arai. Optimal sensor placement for forward kinematics evaluation of a 6-dof parallel link manipulator. In IEEE *International Workshop on Intelligent Robots and Systems, (IROS)*, pages 785–790, Osaka, Japan, November 1991.

[Sal85] G. Salmon. *Lessons introductory to the modern higher algebra.* Chelsea, New York, USA, 1885.

[Tan95a] L. Tancredi. *De la simplification et la résolution du modèle géométrique direct des robots parallèles.* Ph. D. thesis, École Nationale Supérieure des Mines de Paris, Sophia Antipolis, France, December 1995.

[Tan95b] L. Tancredi. Elimination method for parallel manipulators. In PoSSo *open workshop on applications of* PoSSo *and real solving*, Iraklio, Greece, June 1995.

[TTM95a] L. Tancredi, M. Teillaud, and J-P. Merlet. Extra sensors for solving the forward kinematics problem of parallel manipulators. In *Ninth World Congress on the Theory of Machines and Mechanisms*, volume 3, pages 2122–2126, Milano, Italy, September 1995. IFToMM.

[TTM95b] L. Tancredi, M. Teillaud, and J-P. Merlet. Forward kinematics of a parallel manipulator with additional rotary sensors measuring the position of platform joints. In Jean-Pierre Merlet and Bahram Ravani, editors, *Computational Kinematics*, pages 261–270. Kluwer Academic Publishers, 1995.

[vdW48] B.L. van der Waerden. *Modern algebra, Vol. II.* Frederick Ungar Publishing Co, 1948.

[Wam94] C.W. Wampler. Forward displacement analysis of general six-in-parallel (Stewart) platform manipulators using soma coordinates. Research Report GMR-8179, General Motors Research Laboratories, May 1994.

[ZA94] K.E. Zanganeh and J Angeles. The direct kinematics of general parallel manipulators with minimum sensor data. In *ISRAM*, pages 333–338, Hawaï, USA, August 1994.

[ZS91] C.D. Zhang and S.M. Song. Forward kinematics of a class of parallel (Stewart) platforms with closed-form solutions. In *International Conference on Robotics and Automation*, pages 2676–2681, Sacramento, CA, USA, April 1991. IEEE.

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399