On the Trilaterable Six-Degree-of-Freedom Parallel and Serial Manipulators

Josep M. Porta, Lluís Ros and Federico Thomas
Institut de Robòtica i Informàtica Industrial (UPC-CSIC)
Llorens Artigas 4-6, 08028 Barcelona, Catalonia, Spain
{*porta, llros, fthomas*}*@iri.upc.edu*

***Abstract*— The inverse/direct kinematics of trilaterable serial/parallel manipulators can be stated as a system of distance constraints whose set of solutions can be determined using a sequence of trilaterations, possibly involving points at infinity. It is possible to decide whether a mechanism is trilaterable by relying only on its topology. Based on this fact, we here enumerate all trilaterable serial and in-parallel robots with six degrees of freedom. The relevance of the obtained family of manipulators is established when it is shown to contain the best-known commercial serial robots. As a result of this analysis, we come up with a general method to solve the inverse/direct kinematics of a wide family of manipulators.**

***Index Terms*— Trilateration, Cayley-Menger determinants, position analysis of robots.**

I. Introduction

Locating a point in space from its distances to three other known points is a common geometric operation known as *trilateration*. It is well-known that the direct kinematics of the 3-2-1 and 3/2 in-parallel manipulators can be solved by a sequence of trilaterations. These manipulators were first identified by Hunt and Primrose [7] and their kinematics was further elaborated by Bruyninckx [1]. Song and Kwon have also identified some parallel mechanisms with multi-connected joints whose direct kinematics can also be solved by a sequence of trilaterations [9]. Surprisingly enough, no trilaterable serial manipulators have been previously identified and, most importantly, no systematic classification of all trilaterable manipulators, either serial or parallel, has been carried out. One of the main goals of this paper is to provide such a classification.

Recently, the Theory of Distance Geometry [2] is attracting considerable attention from robot kinematicians as it allows coordinate-free formulations for most position analysis problems. Using such formulations, the inverse/direct kinematics of a mechanism can be performed by using a *distance matrix*, one whose entries are squared distances between pairs of points selected on the axes or legs of the mechanism. While some of these distances are known (such as the distances between points on the same axis or between points on consecutive axes, when dealing with serial manipulators, or the distances between leg-attachment points, when dealing with in-parallel ones), many others are unknown. Then, finding all solutions to an inverse or direct kinematics problem boils down to finding values for these unknown distances that permit completing the matrix into "proper" Euclidean distance matrix [10]. If, by any means, the unknown distances are obtained, one can then easily assign coordinates to the selected points and trivially derive the possible configurations of the mechanism at hand.

The determination of all values for the unknown distances is usually done via a *bound smoothing* process: a large range is initially assigned to the unknowns and their bounds are progressively reduced in an iterative manner, by applying triangular inequalities and other necessary conditions [8]. Finding all possible solutions for a given incomplete distance matrix can be extremely complex in general, as this problem is known to be NP-complete. In this paper, we focus on a subclass of distance constraint solving problems, the *trilaterable* ones, where the values of all unknown distances can be derived following a constructive process in which the distance matrix is progressively completed by deducing the value of one unknown at a time, using trilaterations.

In order to identify all robots whose position analysis can be translated into a trilaterable problem, we first characterize the family of distance matrices that encode all serial and in-parallel robots with six degrees of freedom (DoF). Then, we exhaustively search within this family for those matrices that correspond to trilaterable problems. To this end, we employ a constructive algorithm that also determines a trilateration sequence if at least one exists. The result is the identification of the family of all trilaterable serial and in-parallel robots with six DoF. We show that this family includes the best-known serial robots.

This paper is organized as follows. In the next section, we describe how to translate a position analysis problem into a distance constraint satisfaction problem. Section III, defines the concept of trilaterable distance matrices and shows how, under certain circumstances, an incomplete distance matrix can be completed by using a sequence of trilaterations. Section IV presents a comprehensive study of all serial and in-parallel 6-DoF trilaterable robots, that is, of those robots whose position analysis, when translated into a set of distance constraints, leads to a trilaterable distance matrix. All these concepts are then applied in Section V to the resolution of the inverse kinematics of a Puma 560 manipulator. Finally, Section VI summarizes some points deserving further attention.

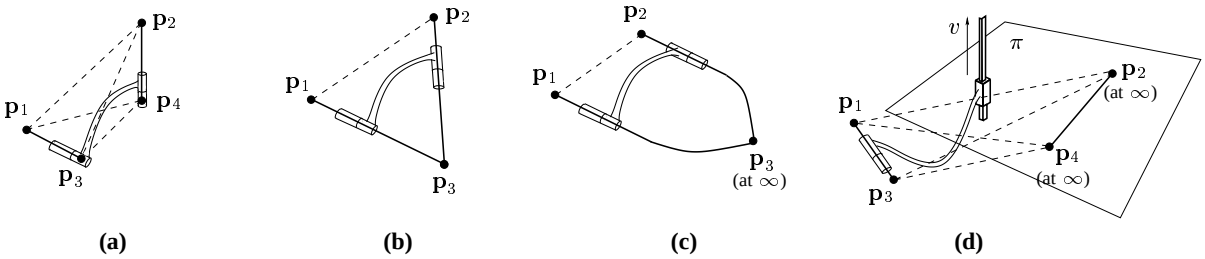

Fig. 1. Modelling a link as a set of points with some fixed distances between them.

II. Modelling robots as distance constraints

The position analysis of a mechanism can usually be translated into a set of distance constraints fixing the relative positions between n points $\mathbf{p}_1, \ldots, \mathbf{p}_n$, selected on its links. Such constraints are usually represented by means of a distance matrix D, whose (i, j) entry is $D_{i,j} = \|\mathbf{p}_i - \mathbf{p}_j\|^2$, i.e., the square of the distance between $\mathbf{p}_i$ and $\mathbf{p}_j$. Some matrix entries are a priori known, and the goal is to solve for the remaining unknowns.

We next show how one can perform this translation for any serial or in-parallel manipulator, and how one can compute the coordinates of the selected points, with reference to an absolute frame, once *all* distances in D have been solved for.

On the one hand, a serial robot is simply an open chain of seven rigid bodies (the links), pairwise articulated through prismatic or revolute joints. The inverse kinematics problem is to find all valid configurations of this chain that are compatible with a specified pose for the last link, relative to the first. Since the first and last links ar mutually fixed, the problem is equivalent to that of finding the valid configurations of a closed loop of six pairwise articulated links. The translation of each link into distance constraints depends on the type of joints it connects, either prismatic or revolute, and on whether the axes of these joints are skew or concurrent.

A link connecting two skew revolute axes can be modelled by taking two points on each of these axes, and by connecting them all with rigid bars to form a tetrahedron (Fig. 1-a). In this way, for example, a 6R linkage can be modelled as a ring of six pairwise-articulated tetrahedra, as indicated in Fig. 5.

If the two axes of the link are not skew but intersecting, we can economize points and simply model the link as a triangle of fixed distances (Fig. 1-b), thus reducing the number of unknown entries in D. Note that the case of parallel revolute axes can be seen as a specialization of the previous one, where the point of intersection is an improper point at infinity in the direction of the axes, instead of a common proper point (Fig. 1-c). This will cause no trouble in our analyis below, as a point at infinity can always be approximated by a proper point, sufficiently far away in some direction.

Similar transformations are applied to a link with one or two prismatic joints: since a translation along direction v can always be seen as a rotation about the line at infinity of any plane π orthogonal to v, we can model a prismatic joint as a revolute joint infinitely far away on this plane. Computationally, we will represent such joint by designating two points on π, placed sufficiently far away along different directions (points $\mathbf{p}_2$ and $\mathbf{p}_4$ in Fig. 1-d).

On the other hand, an in-parallel robot is formed by two rigid bodies, the base and the platform, joined by six legs. Each leg is a linear actuator linked to the base and the platform by spherical joints. The direct kinematics problem is to compute all platform poses that are compatible with some specified leg-lengths. The translation of this goal into a set of distance constraints is trivial. We simply put a point on each leg attachment point and specify the fact that all leg lengths are known, and that the distances between any pair of points lying both on the base, or both on the platform, is also known.

The procedures detailed above can be used to define an incomplete distance matrix from an arbitrary mechanism. If, by any means (e.g. via trilateration, as explained in the next section) we can complete D, then we will need to assign coordinates to the selected points in order to obtain the actual configurations of the robot. The standard way to do so is by first computing the Gram matrix G associated with D, whose entries are defined by $G_{ij} = \frac{1}{2}(D^2_{i,n} + D^2_{j,n} - D^2_{i,j})$. The Cholesky factorization of G into $G = XX^t$ yields the matrix X, whose rows yield unique coordinates for $\mathbf{p}_1, \ldots, \mathbf{p}_n$, up to congruences and mirror transformations. For further details see [10]. For an in-parallel manipulator, the coordinates of $\mathbf{p}_1, \ldots, \mathbf{p}_n$ are sufficient to solve its direct kinematics. For the inverse kinematics of a serial one, we further need to obtain all angles between neighboring links from these coordinates.

III. Trilaterable Distance Matrices

As said, trilateration permits locating a point in space from its distances to three other known points. If we denote by $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$ the coordinate vectors of the known points, the coordinate vector of the unknown point, say $\mathbf{p}_4$, can be expressed in terms of known distances as:

$$\mathbf{p}_4 = \mathbf{p}_1 + \frac{1}{D(1,2,3)} \cdot \Big(- D(1,2,3;1,3,4) \cdot (\mathbf{p}_2 - \mathbf{p}_1) + + D(1,2,3;1,2,4) \cdot (\mathbf{p}_3 - \mathbf{p}_1) + \pm \sqrt{D(1,2,3,4)} \cdot ((\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1)) \Big). \quad (1)$$

where

$$D(i_1, \ldots, i_n; j_1, \ldots, j_n) = 2 \left(\frac{-1}{2} \right)^n \begin{vmatrix} 0 & 1 & \ldots & 1 \\ 1 & D_{i_1,j_1} & \ldots & D_{i_1,j_n} \\ 1 & \ldots & \ldots & \ldots \\ 1 & D_{i_n,j_1} & \ldots & D_{i_n,j_n} \end{vmatrix}, \quad (2)$$

with $D_{i,j} = \|\mathbf{p}_i - \mathbf{p}_j\|^2$, is known as the *Cayley-Menger bi-determinant* of the point sequences $\mathbf{p}_{i_1}, \ldots, \mathbf{p}_{i_n}$, and $\mathbf{p}_{j_1}, \ldots, \mathbf{p}_{j_n}$. Since in many cases of interest the two point sequences will be the same, it will be convenient to abbreviate $D(i_1, \ldots, i_n; i_1, \ldots, i_n)$ by $D(i_1, \ldots, i_n)$, which is simply called the *Cayley-Menger determinant* of the involved points. Observe that the squared distance between $\mathbf{p}_i$ and $\mathbf{p}_j$ can be denoted either as $D_{i,j}$ or $D(i,j)$. See [11] for the details on the derivation of Eq. (2) and its advantages with respect to other formulations.

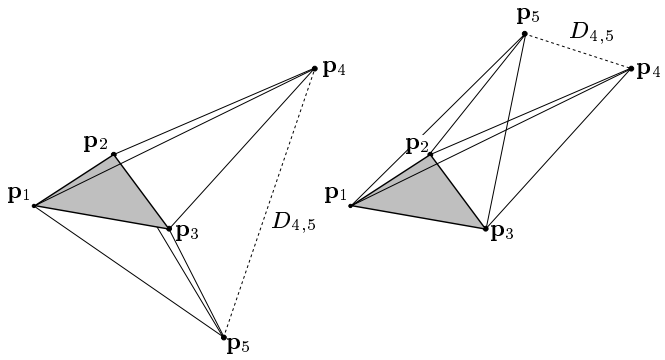

Fig. 2. Computing the location of $\mathbf{p}_4$ and $\mathbf{p}_5$ from their distances to $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$ by applying two trilaterations permits finding $\|\mathbf{p}_5 - \mathbf{p}_4\|$. Two solutions are possible that depend on the relative location of $\mathbf{p}_4$ and $\mathbf{p}_5$ with respect to the plane defined by points $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$.

Now, let us suppose that we compute the location of two points, say $\mathbf{p}_4$ and $\mathbf{p}_5$, from their distances to three points whose locations are given by $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$ by applying two trilaterations (Fig. 2). Then, the expression for the squared distance $D_{4,5} = \|\mathbf{p}_5 - \mathbf{p}_4\|^2$ can be derived by substituting the expressions for $\mathbf{p}_5$ and $\mathbf{p}_4$, obtained from the application of (2), which leads to:

$$D_{4,5} = \frac{a^2 D(1,2) + b^2 D(1,3) + 2ab D(1,2;1,3)}{D^2(1,2,3)} + \frac{c}{D(1,2,3)}, \quad (3)$$

where

$$\begin{aligned} a &= D(1,2,3;1,3,5) - D(1,2,3;1,3,4), \\ b &= D(1,2,3;1,2,4) - D(1,2,3;1,2,5), \\ c &= D(1,2,3,4) + D(1,2,3,5) \\ &\quad \pm 2\sqrt{D(1,2,3,4) D(1,2,3,5)}. \end{aligned}$$

Observe that no point coordinates appear in the result, only inter-point distances, and that two solutions are possible (Fig. 2), corresponding to the two possible signs for the square root.

The Cayley-Menger determinant $D(i_1, \ldots, i_n)$ is $((n-1)!)^2$ times the squared hypervolume of the simplex spanned by the points $\mathbf{p}_{i_1}, \ldots, \mathbf{p}_{i_n}$ in $\mathbb{R}^{n-1}$ [6]. This permits to realize that only when $D(1,2,3) = 0$ —i.e. when $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$ are aligned— the distance $D_{4,5}$ is undefined, and only when $D(1,2,3,4)D(1,2,3,5) = 0$ — i.e when $\mathbf{p}_4$ or $\mathbf{p}_5$ lie on the same plane as $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$— the solution for $D_{4,5}$ is unique (i.e., the mechanism is in a singularity). If $D(1,2,3,4) < 0$ or $D(1,2,3,5) < 0$ one of the tetrahedra can not be assembled with the assigned distances, meaning that the problem is inconsistent.

In what follows, we will say that a distance matrix is trilaterable if, and only if, its unknown distances can be determined only by using Formula (3) repeatedly. In other words, we can determine whether a distance matrix is trilaterable by applying an iterative process where, in each iteration, we try to determine one of the unknown distances via Formula (3), possibly using distances determined in previous iterations. Fig. 3 shows an algorithm that implements this idea. It takes as input an adjacency matrix A that provides information about the known distances, with $A_{i,j} = 1$ if $D_{i,j}$ is known, and 0 otherwise, and returns a trilateration sequence or an empty sequence, depending on whether A is trilaterable or not.

IV. Trilaterable Robots

A trilaterable robot is one whose corresponding distance matrix is trilaterable. Obviously, trilaterable robots are a restricted class of all possible robots, as the set of trilaterable distance matrices is a sub-class of all possible distance matrices. However, although focusing on trilaterable robots could be regarded as too restrictive, we will later see that this class includes most industrial robots.

A. Trilaterable In-Parallel Robots

According to Section II, the distance constraints model of an in-parallel manipulator with m points in the base and

Trilateration Sequence
Input: A, an $n \times n$ matrix with $A_{i,j} = 1$ if distance $D_{i,j}$ is known and 0 otherwise.
Output: A trilateration sequence, empty if A is not trilaterable.
Process:

```
1   T ← ∅
2   do
3     found ← false
4     ∀i, j;  1 ≤ i < j ≤ n
5       if A_{i,j} = 0
6         ∀a, b, c;  1 ≤ a < b < c ≤ n,  a, b, c ≠ i, j
7             s ← A_{a,b} + A_{a,c} + A_{b,c}
8                 +A_{a,i} + A_{b,i} + A_{c,i}
9                 +A_{a,j} + A_{b,j} + A_{c,j}
10            if s = 9
11              found ← true
12              A_{i,j} ← 1
13              T ← T ∪ {(a, b, c) → (i, j)}
14            endif
15      endif
16  while found
17  if ∀i, j (1 ≤ i < j ≤ n);  A_{i,j} = 1
18    return T
19  else
20    return ∅
```

Fig. 3. An algorithm that returns a trilateration sequence to complete a distance matrix D if at least one exists. The known entries of D are input in the form of an adjacency matrix A. A trilateration to derive $D_{i,j}$ from the distances to points a, b and c is symbolically represented in line 13 as $(a, b, c) \rightarrow (i, j)$.

n points in the platform will lead to an adjacency matrix P of the form

$$P = \begin{pmatrix} \mathbf{1}_m & L \\ L^T & \mathbf{1}_n \end{pmatrix}, \tag{4}$$

where $\mathbf{1}_m$ and $\mathbf{1}_n$ are, respectively, $m \times m$ and $n \times n$ matrices of ones, encoding the fact that all points in the base and the platform are mutually rigid, and L is an $m \times n$ matrix encoding the connections established by the legs. Thus, $L_{i,j} = 1$ if there is a leg connecting point i in the base and point j in the platform and, consequently, matrix L includes at least one 1 for each row and column. For the general case where $m = n = 6$, L is a diagonal matrix. All other in-parallel robots are specializations of this case where some points in the base and/or in the platform are coincident. Note further that the minimum number of points either in the base or in the platform must be three. Taking all this into account, it is easy to derive the adjacency matrices of all in-parallel robots with $m, n \in \{3, \ldots, 6\}$ and $m \leq n$, and check whether they are trilaterable using the algorithm in Fig. 3. The result of this exploration, after removing isomorphic structures, is shown in Fig. 4. Each robot is represented here as a bipartite graph, where the edges correspond to the legs of the robot, and the two vertex sets correspond to their attachment points in the base and the platform.

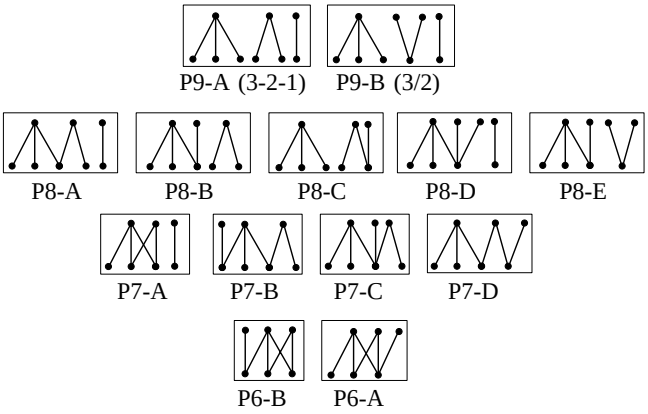

Fig. 4. Schematic representation of all trilaterable in-parallel robots.

Examining the figure, we can see that we have 13 in-parallel trilaterable robots divided into four categories according to the total number of points needed to define the base and the platform. There are no trilaterable in-parallel robots with more than 9 points. Note that the two configurations with 9 points correspond to the well-known 3-2-1 and 3/2 in-parallel robots, already identified as trilaterable by other authors [7].

B. Trilaterable Serial Robots

A 6 Dof serial robot whose end effector is fixed relative to its base, can be seen as a closed chain of six rigid bodies pairwise articulated, that is, as a 6R closed mechanism (Fig. 5-a). Since translations can be seen as rotations centered at infinity, this representation is general enough for our purposes. By taking two points on each axis, this 6R mechanism can be translated into a set of distance constrains between 12 points, as explained in Section II, which correspond to the edge lengths of six pairwise articulated tetrahedra (Fig. 5-b). If two consecutive axes meet at a point (possibly at infinity if they are parallel), the tetrahedron can be substituted by a triangle. However, note that no more than three axes can meet at a point if we want the corresponding serial robot to have 6 DoF.

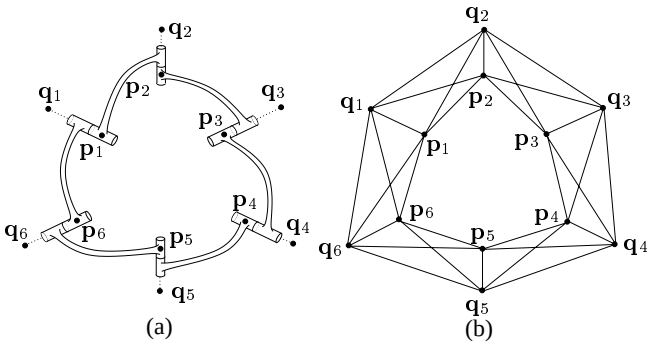

Fig. 5. A general 6R mechanism (a), and its representation in terms of distance constraints (b).

In Fig. 5-b, we have two cycles of points, $\{\mathbf{p}_1, \ldots, \mathbf{p}_6\}$ and $\{\mathbf{q}_1, \ldots, \mathbf{q}_6\}$, where the couples $(\mathbf{p}_i, \mathbf{q}_i)$ define the rotation axes. Since we allow that two or more consecutive axes meet at a point, consecutive points in these cycles can actually coincide and, thus, in general, the two cycles will be $\{\mathbf{p}_1, \ldots, \mathbf{p}_m\}$ and $\{\mathbf{q}_1, \ldots, \mathbf{q}_n\}$ with $m, n \leq 6$. Then, the adjacency matrix S representing all interconnections has the form:

$$S = \begin{pmatrix} C_m & A \\ A^T & C_n \end{pmatrix}, \tag{5}$$

where C_m and C_n are the adjacency matrices for the points in the cycles $\{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$ and $\{\mathbf{q}_1, \ldots, \mathbf{q}_m\}$, respectively (i.e., they are cyclic tridiagonal matrices), and A is the matrix encoding the connections between points lying in different cycles. Now, all candidates for matrix A can be readily enumerated by realizing that the connections they encode should only correspond to a sequence of tetrahedra and triangles. The number of options can be further reduced taking into account that at least one tetrahedron must be included in the sequence because the robot end-effector can be placed arbitrarily with respect to its base. The result of this enumeration leads to 243 candidates. Discarding those that have more than three co-punctual axes and those that do not pass the trilaterability test, we finally obtain 8 non-isomorphic configurations that correspond to the 8 serial robots shown in Fig. 6. Each robot is represented here as a bipartite graph made out of two vertex sets, each corresponding to one of the vertex cycles $\{\mathbf{p}_1, \ldots, \mathbf{p}_m\}$ and $\{\mathbf{q}_1, \ldots, \mathbf{q}_n\}$. Edges between the vertices represent the robot's joint axes.

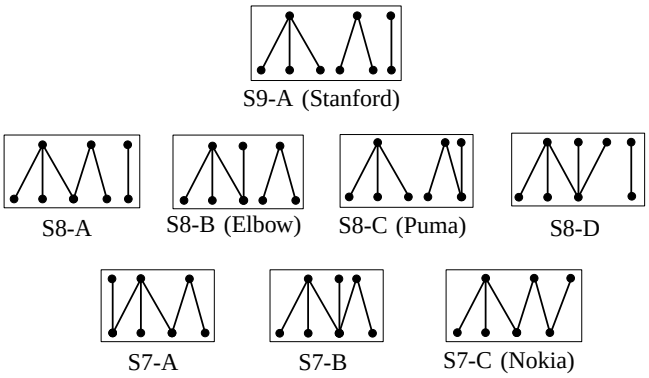

Fig. 6. Schematic representation of all trilaterable serial robots.

A close look to this family of serial robots reveals that all of its members have three co-punctual axes. Thus, all serial trilaterable robots are decoupled, that is, their inverse kinematics can be decomposed into rotational and translational components.

We note that the family of trilaterable serial robots, although small, includes most popular 6 DoF serial manipulators such as the Stanford, Cincinnati Milacron, Puma and Nokia manipulators, as shown in Fig. 7.

Serial	Parallel
S9-A	P9-A, P9-B
S8-A	P8-A
S8-B	P8-B, P8-E
S8-C	P8-C
S8-D	P8-D
S7-A	P7-B
S7-B	P7-C
S7-C	P7-D

TABLE I
CORRESPONDENCE BETWEEN TRILATERABLE IN-PARALLEL AND SERIAL ROBOTS.

C. Relating the serial and the in-parallel families

Comparing Figs. 4 and 6 we observe that the schematic representations for the in-parallel and serial robots are similar. Indeed, there exists a correspondence between these two families.

Every trilaterable serial robot, defined as two point cycles, $\{\mathbf{p}_1, \ldots, \mathbf{p}_m\}$ and $\{\mathbf{q}_1, \ldots, \mathbf{q}_n\}$, plus some distance constraints on them, can be converted to a trilaterable in-parallel one proceeding as follows. First, remove all the fixed distances of the serial robot except those corresponding to point pairs $(\mathbf{p}_i, \mathbf{q}_j)$ lying on the same joint axis. (These will play the role of the legs in the parallel robot.) Then, let the point cycles $\{\mathbf{p}_1, \ldots, \mathbf{p}_m\}$ and $\{\mathbf{q}_1, \ldots, \mathbf{q}_n\}$ represent, respectively, the points on the base and the platform of the in-parallel robot. Finally, let the base and platform be rigid by placing as many distance constraints as point pairs in the base and the platform. Doing this on each serial trilaterable robot, we obtain the correspondences shown in Table I.

Unfortunately, the converse is not true. That is, not every trilaterable in-parallel robot can be converted to a trilaterable serial one. Note that the parallel robots P7-A, P6-A, and P6-B have no counterpart in the serial family.

V. WORKING OUT THE PUMA 560 ROBOT

The presented methodology has been fully implemented in MATLAB. As an example, we here follow in detail how it can be used to solve the inverse kinematics of a PUMA 560 manipulator.

First, each link of the robot must be modelled as a set of distance constraints between a set of points. Fig. 7-b shows this robot in its home position, and the points that should be selected to derive such constraints. Coordinates for all points can be easily obtained in terms of the Denavit-Hartenberg parameters of this robot, shown in Table II.

According to the process explained in Section II, link 1 can be modelled by the fixed distances between three points, 1, 6, and 2, as it connects two intersecting axes. Similarly, link 2 is modelled by the fixed distances between points 6, 2, and 7, but point 2 must be placed at infinity, as the axes of this link are parallel. Link 3 has two skew

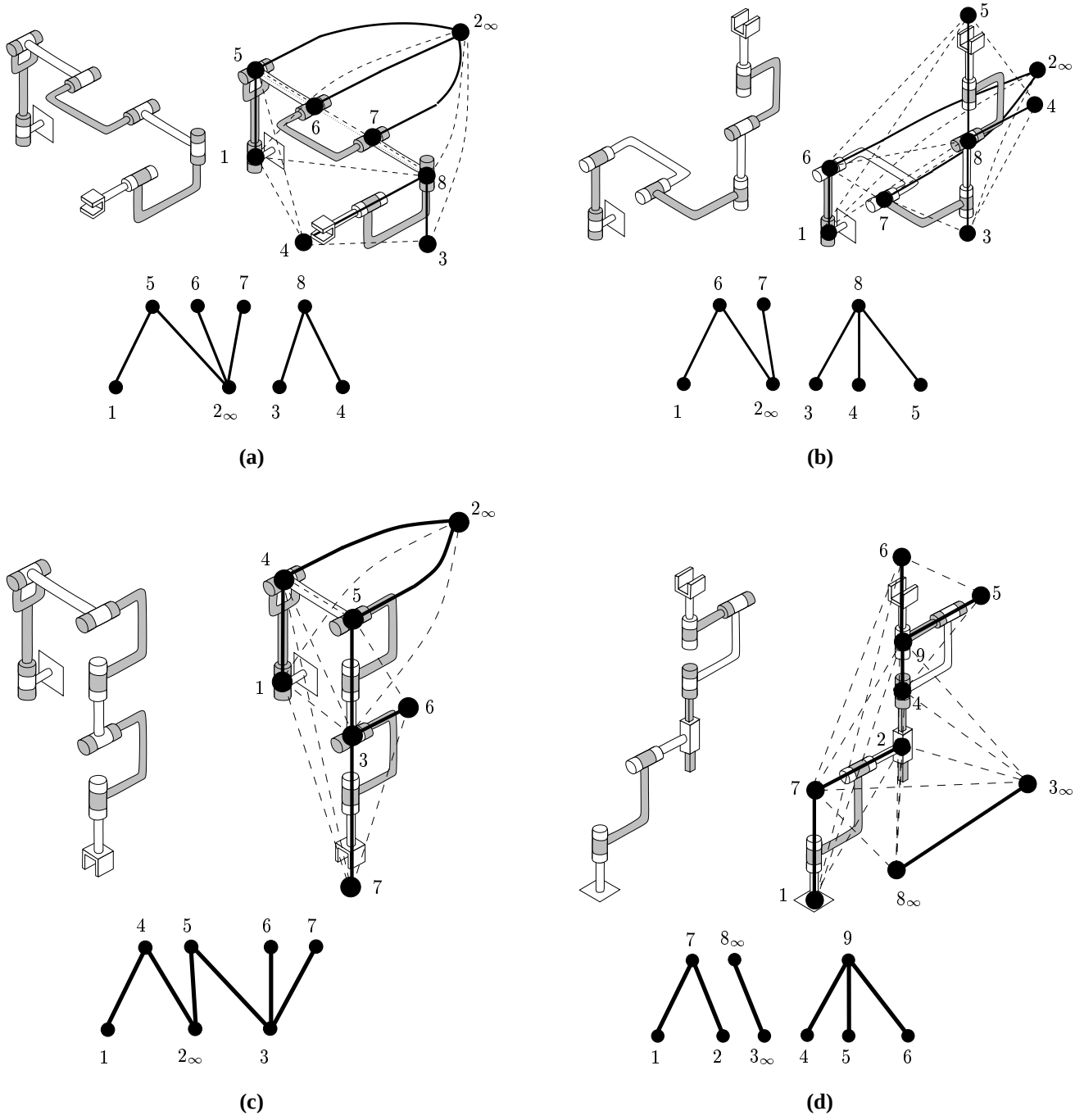

Fig. 7. (a) Elbow manipulator or Cincinnati Milacron (S8-B). (b) Puma manipulator (S8-C). (c) Nokia manipulator (S7-C). (d) Stanford manipulator (S9-A). In (a), (b), and (c), parallelism between axes is exploited to simplify its analysis, but note that, even if this artifact is not introduced, (a) and (b) are still trilaterable. Robot (d) includes a translational degree of freedom that, according to our formalism, is represented by a rotation axis at infinity, becoming then trilaterable. (Figure partially adapted from [5].)

$$D = \begin{pmatrix} 0 & 101.05 & D_{1,3} & D_{1,4} & \mathbf{1.314} & 1 & D_{1,7} & \mathbf{1.877} \\ 101.05 & 0 & 97.077 & D_{2,4} & D_{2,5} & 100.05 & 100.05 & 97.264 \\ D_{1,3} & 97.077 & 0 & 1.186 & D_{3,5} & D_{3,6} & 0.023 & 0.186 \\ D_{1,4} & D_{2,4} & 1.186 & 0 & 2 & D_{4,6} & D_{4,7} & 1 \\ \mathbf{1.314} & D_{2,5} & D_{3,5} & 2 & 0 & \mathbf{0.596} & D_{5,7} & 1 \\ 1 & 100.05 & D_{3,6} & D_{4,6} & \mathbf{0.596} & 0 & 0.186 & \mathbf{0.318} \\ D_{1,7} & 100.05 & 0.023 & D_{4,7} & D_{5,7} & 0.186 & 0 & 0.209 \\ \mathbf{1.877} & 97.264 & 0.186 & 1 & 1 & \mathbf{0.318} & 0.209 & 0 \end{pmatrix}$$

(a)

triangular base		distance
$(2,6,8)$	$\rightarrow$	$(1,7)$
$(1,6,8)$	$\rightarrow$	$(2,5)$
$(2,7,8)$	$\rightarrow$	$(3,6)$
$(2,6,7)$	$\rightarrow$	$(1,3)$
$(2,6,8)$	$\rightarrow$	$(3,5)$
$(1,2,6)$	$\rightarrow$	$(5,7)$
$(3,5,8)$	$\rightarrow$	$(1,4)$
$(3,5,8)$	$\rightarrow$	$(2,4)$
$(3,5,8)$	$\rightarrow$	$(4,6)$
$(3,5,8)$	$\rightarrow$	$(4,7)$

(b)

sol. #	$D_{1,3}$	$D_{1,4}$	$D_{1,7}$	$D_{2,4}$	$D_{2,5}$	$D_{3,5}$	$D_{3,6}$	$D_{4,6}$	$D_{4,7}$	$D_{5,7}$
1	0.9001	2.3787	0.8859	112.2223	86.2746	0.7010	0.1865	0.5491	1.0313	0.9265
2	0.9001	3.3748	0.8859	84.3052	86.2746	0.7010	0.1865	2.0875	1.3875	0.9265
3	0.9255	2.2916	0.8859	93.3075	115.0944	0.8136	0.2262	0.5337	1.2383	0.5647
4	0.9255	3.4620	0.8859	103.2200	115.0944	0.8136	0.2262	2.1029	1.1804	0.5647
5	2.1126	0.8838	2.0421	112.7286	86.2746	0.7507	0.2262	0.5143	0.9574	0.9187
6	2.1126	4.8697	2.0421	83.7989	86.2746	0.7507	0.2262	2.1223	1.4613	0.9187
7	1.9842	0.6262	2.0421	88.3026	115.0944	1.1000	0.1865	0.7653	1.4018	0.8902
8	1.9842	5.1273	2.0421	108.2249	115.0944	1.1000	0.1865	1.8712	1.0170	0.8902
9	0.9255	2.1864	0.8859	112.7964	86.2746	0.7587	0.2262	0.5166	0.9566	0.9265
10	0.9255	3.5672	0.8859	83.7311	86.2746	0.7587	0.2262	2.1199	1.4622	0.9265
11	0.9001	2.5523	0.8859	94.4065	115.0944	0.7948	0.1865	0.5828	1.3148	0.5647
12	0.9001	3.2012	0.8859	102.1210	115.0944	0.7948	0.1865	2.0537	1.1040	0.5647
13	1.9842	0.7981	2.0421	112.1319	86.2746	0.6936	0.1865	0.5467	1.0329	0.9187
14	1.9842	4.9555	2.0421	84.3956	86.2746	0.6936	0.1865	2.0899	1.3859	0.9187
15	2.1126	0.6337	2.0421	88.1064	115.0944	1.1479	0.2262	0.7051	1.3226	0.8902
16	2.1126	5.1199	2.0421	108.4211	115.0944	1.1479	0.2262	1.9314	1.0962	0.8902

(c)

sol. #	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	11.75	200.37	171.57	133.73	124.20	28.30
2	11.75	200.37	171.57	313.73	235.80	208.30
3	228.62	339.63	14.67	283.19	115.57	30.58
4	228.62	339.63	14.67	103.19	244.43	210.58
5	11.75	97.66	14.67	221.40	239.70	77.82
6	11.75	97.66	14.67	41.40	120.30	257.83
7	228.62	82.34	171.57	58.262	264.26	106.19
8	228.62	82.34	171.57	238.26	95.74	286.19

(d)

d_∞	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	222.76	337.40	14.88	106.64	250.01	206.41
10	228.62	339.63	14.67	103.19	244.43	210.58
100	229.18	339.84	14.68	102.88	243.87	210.98
1000	229.24	339.86	14.68	102.84	243.82	211.02
10000	229.25	339.86	14.68	102.84	243.81	211.03
∞ (exact)	229.25	339.86	14.68	102.84	243.81	211.03

(e)

Fig. 8. Solving the inverse kinematics of the PUMA 560 by the proposed method. (a) Input distance matrix. (b) Trilateration sequence needed to complete it. (c) The sixteen possible completions. (d) The eight solutions to the problem, selected among the 32 possible coordinatizations derived from the solutions in (c). (e) Evolution of the fourth solution, as we increase d_∞. Squared distances are expressed in m^2 and angles in degrees.

i	α_i (deg.)	a_i (m)	θ_i (deg.)	d_i (m)
1	90	0	θ_1	0
2	0	0.4318	θ_2	0
3	−90	0.0203	θ_3	0.15005
4	90	0	θ_4	0.4318
5	−90	0	θ_5	0
6	0	0	θ_6	0

TABLE II

DENAVIT-HARTENBERG PARAMETERS OF THE PUMA 560 ROBOT.

axes and, thus, it is modelled by the fixed distances of the tetrahedron defined by 7, 2, 3, and 8. Links 4 and 5 connect concurrent axes and are thus represented by the triangles (3, 8, 4), and (4, 8, 5), respectively. These distances yield the non-bold numeric entries of the D matrix, shown in Fig. 8.

The usual formulation of the inverse kinematics further considers a 0th and a 6th link, respectively modelling the 'world' and the 'hand', and a homogeneous transformation positioning the hand with respect to the world. Note that, in doing so, these two elements become mutually rigid, and it is thus possible to consider them as a *single* link constraining the relative position between the first and last revolute axes. In our setting, this is equivalent to fixing the distances of the tetrahedron (1, 6, 5, 8). Note that these distances, while fixed, depend on the specific pose of the end effector relative to the world, or, in other words, on the specific inverse kinematics problem to be solved. They correspond to the numeric bold entries in D, computed in this example for a pose of the hand given by the joint angles $(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) = (229.25°, 339.86°, 14.68°, 102.84°, 243.81°, 211.03°)$.

Now, by applying the algorithm in Fig. 3, one obtains the trilateration sequence given in Fig. 8-b, which can be used to complete D by iteratively applying Formula 3. Since there are 10 trilateration steps, a total of $2^{10} = 1024$ possible completions exist for D. By examining them all, only 16 yield consistent Euclidean matrices; that is, only 16 have an associated Gram matrix that is positive semidefinite of rank 3. These are enumerated in Fig. 8-c. Following Section II, each such matrix yields two specular coordinatizations of the points $1, \ldots, 8$. If among these coordinatizations we discard those where the orientation of the tetrahedra $(1, 6, 5, 8)$ and $(2, 3, 7, 8)$ is different from that in the actual robot, we finally obtain the expected 8 solutions for the joint angles $\theta_1, \ldots, \theta_6$, shown in Fig. 8-d. The reader can check that the fourth such solution is very close to the pose of the hand specified above, the small error being due to the fact that point 2 is not sufficiently far away from the robot (here it is placed at a distance $d_\infty = 10$m from the line defined by points 6 and 7). The last table in Fig. 8 shows how, by solving the same problem at increasing values of d_∞, this solution tends to the exact one.

VI. CONCLUSIONS

We have presented an explicit enumeration of all six degrees-of-freedom in-parallel and serial manipulators and a general method for their position analysis, that is, for solving their direct/inverse kinematics. This method has been implemented in MATLAB, and the resolution of the inverse kinematics of a PUMA 560 robot has been reported as an example. This method, if implemented in a symbolic algebra package, would yield closed-form formulas for the direct/inverse kinematics of any trilaterable in-parallel or serial manipulator. This is a point that deserves further efforts as it would permit to analyze the computational efficiency of the resulting formulas, and their numerical stability, in front of other well-known formulations obtained for particular architectures.

VII. ACKNOWLEDGMENTS

This work has been partially funded by the Spanish Ministry of Education through the I+D project DPI2004-07358, the Catalan Research Commission through the "Robotics and Control Group", and by two Ramón y Cajal contracts supporting the first and second authors.

REFERENCES

[1] H. Bruyninckx, "Forward kinematics for Hunt-Primrose parallel manipulators," *Mechanism and Machine Theory*, Vol. 34, pp. 657-664, 1999.

[2] G. M. Crippen and T. Havel, *Distance Geometry and Molecular Conformation*, John Wiley and Sons Inc., New York, 1988.

[3] B. Dasgupta and T.S. Mruthyunjaya, "The Stewart platform manipulator: a review," *Mechanism and Machine Theory*, Vol. 35, pp. 15-40, 2000.

[4] J. Denavit and R.S. Hartenberg, *Kinematic Synthesis of Linkages*, McGraw-Hill Book Inc., New York, 1964.

[5] M. Eppinger and E. Kreuzer, "Evaluation of methods for solving the inverse kinematics of manipulators," Prooceedings of the 8th CISM-IFToMM Symposium on Theory and Practice of Robot Manipulators, pp. 32-39, 1990.

[6] T. Havel, "Some Examples of the Use of Distances as Coordinates in Euclidean Geometry," *J. of Symbolic Computation*, No. 11, pp. 579-593, 1991.

[7] K.H. Hunt and E.J.F. Primrose, "Assembly configurations of some in-parallel-actuated manipulators," *Mechanism and Machine Theory*, Vol. 28, No. 1, pp. 31-42, 1993.

[8] J.M. Porta, L. Ros, F. Thomas, and C. Torras, "A Branch-and-Prune Solver for Distance Constraints," *IEEE Trans. on Robotics*, to appear.

[9] S.-K. Song and D.-S. Kwon, "A tetrahedron approach for a unique closed-form solution of the forward kinematics of six-dof parallel mechanisms with multiconnected joints," *Journal of Robotic Systems*, Vol. 19, No. 6, pp. 269-281, 2002.

[10] F. Thomas, J.M. Porta, and L. Ros, "Distance Constraints Solved Geometrically," Advances in Robot Kinematics, Kluwer Academic Publishers, pp. 123-132, 2004.

[11] F. Thomas and L. Ros, "Revisiting Trilateration for Robot Localization," *IEEE Trans. on Robotics*, to appear.