

A General Method for the Numerical Computation of Manipulator Singularity Sets

Oriol Bohigas, Dimitar Zlatanov, Lluís Ros, Montserrat Manubens, and Josep M. Porta

Abstract—The analysis of singularities is central to the development and control of a manipulator. However, existing methods for singularity set computation still concentrate on specific classes of manipulators. The absence of general methods able to perform such computation on a large class of manipulators is problematic because it hinders the analysis of unconventional manipulators and the development of new robot topologies. The purpose of this paper is to provide such a method for nonredundant mechanisms with algebraic lower pairs and designated input and output speeds. We formulate systems of equations that describe the whole singularity set and each one of the singularity types independently, and show how to compute the configurations in each type using a numerical technique based on linear relaxations. The method can be used to analyze manipulators with arbitrary geometry, and it isolates the singularities with the desired accuracy. We illustrate the formulation of the conditions and their numerical solution with examples, and use 3-D projections to visualize the complex partitions of the configuration space induced by the singularities.

Index Terms—Branch-and-prune method, linear relaxation, nonredundant manipulator, singularity set computation.

I. INTRODUCTION

IN robot singularities, either the forward or the inverse instantaneous kinematic problem becomes indeterminate, and the properties of the mechanism change dramatically, often detrimentally. Despite the importance of such critical configurations, the rich literature on singularity analysis does not provide a method to explicitly compute the singularity set, and to identify the various singularity types in it, on manipulators of a general architecture. Most works on the topic focus on particular classes of singularities and restrict their attention to specific robot designs [1]–[13].

The efforts on characterizing all possible singularity types date back to the nineties [14]–[19]. Based on an input–output velocity equation, a general singularity classification was attempted in [14], but it was soon seen that this classification

overlooks cases where the motion of the mechanism cannot be described solely with the input and output speeds [15]. This led Zlatanov to define a general manipulator model in terms of differentiable mappings between manifolds, giving rise to a rigorous mathematical definition of kinematic singularity [16], [18]. Using the model, six different singularity types were identified, which correspond to the distinct kinematic phenomena that may occur in a singularity.

Although the conditions for the presence of singularities of all types were given in [17] and [18], the formulation of these conditions into a form amenable for computation had yet to be achieved. The goal of the present work is to address this task by defining systems of equations that describe all singularity types and proposing a numerical procedure able to solve them. The methodology is general and applicable to virtually any relevant mechanism geometry. It allows the complete singularity set to be obtained with the desired accuracy and each of its singularity types to be computed independently.

The approach was preliminarily introduced in [20] and is now presented and illustrated in thorough detail. The guiding principle is the importance of a complete characterization of the manipulator motion in order to identify *all* possible singular phenomena. For each such phenomenon, we present, simply and rigorously, the definition, the mechanical significance, the algebraic conditions, and the computation of the corresponding singularity subset. Special emphasis is placed on illustrating concepts and procedures with clear and comprehensible examples. In addition, since a full knowledge of a mechanism's special configurations is key to understanding its motion capabilities, the paper exemplifies the use of 3-D projections to reveal and visualize the complex singularity-induced partition and interconnectedness of the configuration space.

The rest of the paper is organized as follows. Section II briefly recalls the definition of singular configuration and provides systems of equations that characterize the whole singularity set of a manipulator. These systems can already be used to isolate the set, as done in [21] for the planar case; however, additional systems are provided in Section III to independently compute the configurations that belong to each one of the six singularity types that are identified in [16] and [18]. The derivation and application of these systems is next illustrated in Section IV on a simple example admitting an analytical approach. In general, a numerical method is needed to solve the equations, and Section V provides one based on a branch-and-prune strategy and linear relaxations. Section VI demonstrates the performance of the method with the analysis of a planar and a spatial manipulator. Finally, Section VII summarizes the main conclusions of the paper and suggests points for future work.

Manuscript received May 6, 2013; accepted September 20, 2013. Date of publication October 18, 2013; date of current version April 1, 2014. This paper was recommended for publication by Associate Editor V. Krovi and Editor B. J. Nelson upon evaluation of the reviewers' comments. This work was supported in part by the Spanish Ministry of Economy and Competitiveness under Contract DPI2010-18449. The work of M. Manubens was supported by a Juan de la Cierva contract.

O. Bohigas, L. Ros, M. Manubens, and J. M. Porta are with the Kinematics and Robot Design Group, Institut de Robòtica i Informàtica Industrial, CSIC-UPC, 08028 Barcelona, Spain (e-mail: obohigas@iri.upc.edu; lros@iri.upc.edu; mmanuben@iri.upc.edu; porta@iri.upc.edu).

D. Zlatanov is with the PMAR Laboratory (DIMEC), Università di Genova, 16145 Genoa, Italy (e-mail: zlatanov@dimec.unige.it).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TRO.2013.2283416

II. CHARACTERIZATION OF THE SINGULARITY SET

Every configuration of a manipulator can be described by a tuple \mathbf{q} of scalar generalized-coordinate variables. For manipulators with closed kinematic chains, or when a joint does not admit a global parameterization, the configuration space is given by the solution set of a system of nonlinear equations

$$\Phi(\mathbf{q}) = \mathbf{0} \quad (1)$$

that expresses the assembly constraints imposed by the joints [22]. In addition, the feasible instantaneous motions of the manipulator can be characterized by a linear system of equations

$$\mathbf{L} \mathbf{m} = \mathbf{0} \quad (2)$$

where \mathbf{L} is a matrix that depends on the configuration \mathbf{q} , and \mathbf{m} is the so-called velocity vector of the manipulator [18]. The vector \mathbf{m} takes the form $\mathbf{m} = [\Omega^o{}^\top, \Omega^a{}^\top, \Omega^p{}^\top]^\top$, where Ω^o , Ω^a , and Ω^p provide the output, input, and passive velocity vectors, respectively. Typically, Ω^o encodes the velocity of a point and/or the angular velocity of an end-effector body, and Ω^a and Ω^p encompass the actuated and unactuated joint speeds. Such a system, called the velocity equation in [18], can be obtained for any manipulator [23], and therefore, it can be used for the practical identification of singularities.

In this paper, we assume that the manipulator is nonredundant. This implies that the dimensions of Ω^o and Ω^a are equal to the global mobility n of the mechanism, defined as the dimension of the configuration space, i.e., as the maximum dimension of its tangent space, wherever such a space exists [24].

In general, the instantaneous kinematic analysis of a manipulator addresses the following two main problems:

- 1) the *forward* instantaneous kinematics problem (FIKP): Find \mathbf{m} given the input velocity Ω^a ; and
- 2) the *inverse* instantaneous kinematics problem (IIKP): Find \mathbf{m} given the output velocity Ω^o .

Note that, contrary to what is assumed elsewhere [14], in both cases, it is required to find *all* velocity components of \mathbf{m} , not just those referring to the output or input velocities, respectively. Following [18], a configuration is said to be *nonsingular* when both the FIKP and the IIKP have unique solutions for any input or output velocity and *singular* otherwise.

Let \mathbf{L}_I , \mathbf{L}_O , and \mathbf{L}_P be the submatrices of \mathbf{L} obtained by removing the columns corresponding to the input, output, and both the input and output velocities, respectively. It is easy to see that the singular configurations are those in which either \mathbf{L}_I or \mathbf{L}_O is rank deficient. If a matrix is rank deficient, its kernel has to be nonnull and, in particular, it must include a vector of unit norm. Thus, all singularities can be determined by solving the following two systems of equations:

$$\left. \begin{array}{l} \Phi(\mathbf{q}) = \mathbf{0} \\ \mathbf{L}_I^\top \boldsymbol{\xi} = \mathbf{0} \\ \|\boldsymbol{\xi}\|^2 = 1 \end{array} \right\}, \quad \left. \begin{array}{l} \Phi(\mathbf{q}) = \mathbf{0} \\ \mathbf{L}_O^\top \boldsymbol{\xi} = \mathbf{0} \\ \|\boldsymbol{\xi}\|^2 = 1 \end{array} \right\}. \quad (3)$$

The first equation of each system constrains \mathbf{q} to be a feasible configuration of the mechanism, and the second and third equations enforce the existence of a nonzero vector in the kernel of the corresponding matrix. Note that $\|\boldsymbol{\xi}\|^2$ can be any consistent

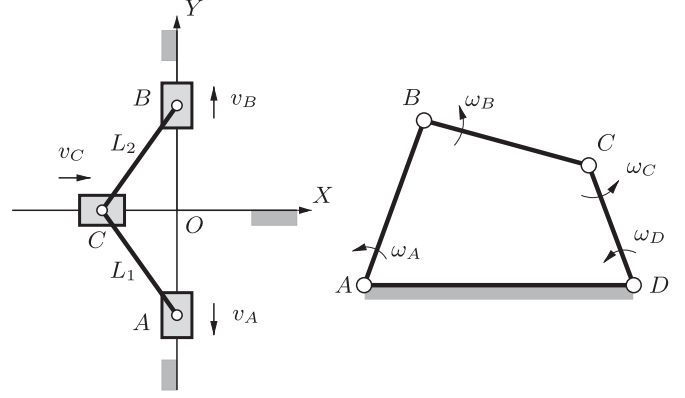


Fig. 1. (Left) 1-DOF mechanism with three sliders. The prismatic joints at A and B are on a line perpendicular to the axis of the prismatic joint at C . (Right) Four-bar mechanism. The angular velocities indicated refer to relative motions, e.g., ω_B is the angular velocity of link BC relative to link AB .

norm, for instance, $\boldsymbol{\xi}^\top \mathbf{D} \boldsymbol{\xi}$, with \mathbf{D} a diagonal matrix with the proper physical units. There is no need for the norm to be invariant with respect to change of frame or units. In short, the condition $\|\boldsymbol{\xi}\|^2 = 1$ only serves to guarantee that $\boldsymbol{\xi}$ is not $\mathbf{0}$. The solutions of the system on the left in (3) include all singularities where the FIKP is indeterminate (forward singularities), while the solutions of the system on the right include all singularities where the IIKP is indeterminate (inverse singularities).

Now, depending on the cause of the degeneracy, six substantially different types of singularities can be recognized. These are *redundant input* (RI), *redundant output* (RO), *impossible input* (II), *impossible output* (IO), *increased instantaneous mobility* (IIM), and *redundant passive motion* (RPM) singularities. Each of the six types corresponds to a different change in the kinematic properties of the manipulator, and it is, therefore, desirable to know whether a configuration belongs to a given type and to compute all possible configurations of that type.

III. CHARACTERIZATION OF THE SINGULARITY TYPES

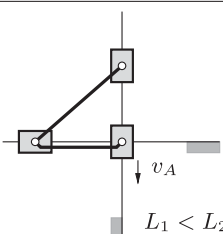
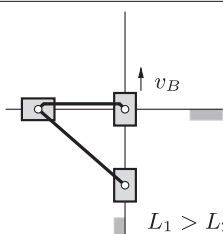
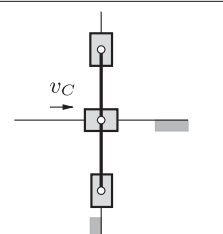
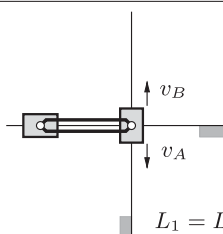
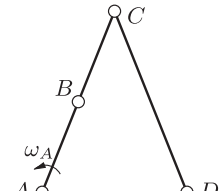
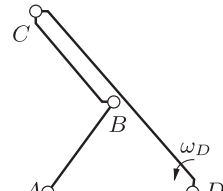
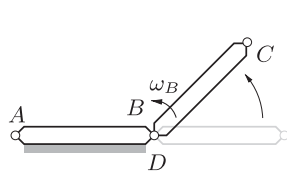
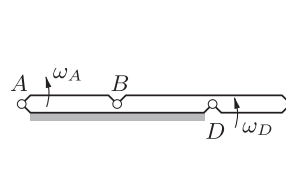
The definitions of each one of the six singularity types are recalled next. Following each definition, a system of equations that characterizes the configurations of the type is derived. The three-slider and four-bar mechanisms of Fig. 1 are used to illustrate the different singularity types on mechanisms with prismatic and revolute joints. Each mechanism has one degree of freedom and, unless otherwise stated, the input and output velocities are those of points A and B , i.e., v_A and v_B , for the three-slider mechanism, and the angular velocities of links AB and DC , i.e., ω_A and ω_D , for the four-bar mechanism.

A. Redundant Input

A configuration is a singularity of RI type if there exists an input velocity vector $\Omega^a \neq \mathbf{0}$, and a vector Ω^p , that satisfy the velocity equation (2) for $\Omega^o = \mathbf{0}$, i.e., such that

$$\mathbf{L}_O \begin{bmatrix} \Omega^a \\ \Omega^p \end{bmatrix} = \mathbf{0}$$

TABLE I
SIX SINGULARITY TYPES EXEMPLIFIED WITH THREE-SLIDER AND FOUR-BAR MECHANISM CONFIGURATIONS

RI, IO	RO, II	RPM	IIM
			
			

with $\Omega^a \neq 0$. Since such a vector exists whenever there exists a unit vector with $\Omega^a \neq 0$, q is a singularity of RI type if, and only if, the system of equations

$$\begin{cases} \Phi(q) = 0 \\ L_O \xi = 0 \\ \|\xi\|^2 = 1 \end{cases} \quad (4)$$

is satisfied for some value of $\xi = [\Omega^a, \Omega^p]^T$ with $\Omega^a \neq 0$.

Two examples of these singularities are provided in Table I, first column. In the top configuration, v_A can have any value, while v_C must be zero, and thus, point B cannot move. In the bottom configuration, the output link DC cannot move, since the velocity of point C must be zero, while ω_A can have any value.

B. Redundant Output

A configuration is a singularity of RO type if there exists an output velocity vector $\Omega^o \neq 0$, and a vector Ω^p , that satisfy the velocity equation for $\Omega^a = 0$, i.e., such that

$$L_I \begin{bmatrix} \Omega^o \\ \Omega^p \end{bmatrix} = 0$$

with $\Omega^o \neq 0$. Following a similar reasoning as stated previously, q is of RO type if, and only if, it satisfies the equations

$$\begin{cases} \Phi(q) = 0 \\ L_I \xi = 0 \\ \|\xi\|^2 = 1 \end{cases} \quad (5)$$

for some value of $\xi = [\Omega^o, \Omega^p]^T$ with $\Omega^o \neq 0$.

The three-slider and the four-bar mechanisms in the second column of Table I are shown in a singularity of RO type. On the former, the instantaneous output v_B can have any value, while point A must have zero velocity. The same happens on the latter, where the input link AB is locked, while the instantaneous output ω_D can have any value.

C. Impossible Output

A configuration is a singularity of IO type if there exists a vector $\Omega^o \neq 0$ in the output-velocity space for which the velocity equation cannot be satisfied for any combination of Ω^a and Ω^p . This means that there is a nonzero vector $[\Omega^o, 0, 0]^T$ that cannot be obtained by projection of any vector $[\Omega^o, \Omega^a, \Omega^p]^T$ belonging to the kernel of L .

In order to derive the system of equations for this type, let $V = [v_1, \dots, v_r]$ be a matrix whose columns form a basis of the kernel of L . Then, all vectors $[\Omega^o, 0, 0]^T$ that can be obtained by projection of some vector of the kernel of L are those in the image space of the linear map given by

$$A = [I_{n \times n} \quad 0] V$$

where n is the dimension of Ω^o . Thus, a singular configuration is of IO type if the map is not surjective, i.e., if A is rank deficient. In this situation, it can be seen that there exists a unit vector Ω^{o*} in the kernel of A^T and, hence, a vector $[\Omega^{o*}, 0, 0]^T$ in the kernel of V^T . Such a vector is orthogonal to all vectors v_1, \dots, v_r , and therefore, it must belong to the image of L^T . In conclusion, there must exist a nonzero vector Ω^{o*} satisfying

$$L^T u = \begin{bmatrix} \Omega^{o*} \\ 0 \\ 0 \end{bmatrix}$$

for some vector u , which can be chosen of unit norm. Therefore, a configuration q is an IO type singularity if, and only if, it satisfies

$$\begin{cases} \Phi(q) = 0 \\ L^T u = [\Omega^{o*} \quad 0^T \quad 0^T]^T \\ \|u\|^2 = 1 \end{cases} \quad (6)$$

with $\Omega^{o*} \neq 0$. For all solutions of this system, the obtained value of Ω^{o*} corresponds to a nonfeasible output at the corresponding configuration.

The configurations in the first column of Table I are also singularities of IO type because any nonzero output is impossible in them.

D. Impossible Input

A configuration is a singularity of II type if there exists an input velocity vector $\Omega^a \neq 0$ for which the velocity equation cannot be satisfied for any combination of Ω^o and Ω^p . Following a similar reasoning as for the IO type, a configuration q is a singularity of II type if, and only if, there exists a nonzero vector Ω^{a*} such that

$$L^T u = \begin{bmatrix} 0 \\ \Omega^{a*} \\ 0 \end{bmatrix}$$

for some vector u , which can also be chosen of unit norm. Thus, a configuration q will be a singularity of II type if, and only if, it satisfies

$$\left. \begin{aligned} \Phi(q) &= 0 \\ L^T u &= [0^T \quad \Omega^{a*T} \quad 0^T]^T \\ \|u\|^2 &= 1 \end{aligned} \right\} \quad (7)$$

with $\Omega^{a*} \neq 0$.

The three-slider and the four-bar mechanisms in the second column of Table I are also in singularities of II type since any nonzero input is impossible in these configurations.

E. Redundant Passive Motion

A configuration is a singularity of RPM type if there exists a vector Ω^p in the input-velocity space that satisfies the velocity equation for $\Omega^a = 0$ and $\Omega^o = 0$, i.e., such that

$$L_P \Omega^p = 0$$

with $\Omega^p \neq 0$. This will happen when the kernel of L_P is nonzero, and thus, the following system of equations

$$\left. \begin{aligned} \Phi(q) &= 0 \\ L_P \Omega^p &= 0 \\ \|\Omega^p\|^2 &= 1 \end{aligned} \right\} \quad (8)$$

encodes all RPM type singularities q .

Two examples of these singularities are provided in Table I, third column. In the three-slider mechanism, both the input A and the output B must have zero velocity, while the velocity of point C can be nonzero. A four-bar mechanism with a kite geometry, as shown in the table, can collapse so that all joints lie on a single line, and B and D coincide. If the input and output are the velocities at joints A and C , ω_A and ω_C , the mechanism can move from the configuration shown in gray, maintaining zero velocity at both the input and output joints. Nonzero velocity is present only at the passive joints B and D . Hence, both mechanisms are shown in a singularity of RPM type.

F. Increased Instantaneous Mobility

A configuration is a singularity of IIM type if L is rank deficient. In fact, these are configurations where the instantaneous

mobility is greater than the number of degrees of freedom. The definition directly allows writing the system of equations

$$\left. \begin{aligned} \Phi(q) &= 0 \\ L^T \xi &= 0 \\ \|\xi\|^2 &= 1 \end{aligned} \right\} \quad (9)$$

that will be satisfied for some ξ by a configuration q if, and only if, it is a singularity of IIM type. These are also called *configuration-space* singularities, because they correspond to points where the tangent space is ill defined, and thus, both the FIKP and IIKP become indeterminate for any definition of input or output on the given velocity variables.

The mobility of the three-slider and the four-bar mechanisms in the fourth column of Table I increases from 1 to 2 at the shown configurations, and thus, they exhibit a singularity of IIM type.

IV. ILLUSTRATIVE EXAMPLE

To exemplify how the previous systems can be used to obtain the configurations of each singularity type, consider the three-slider mechanism in Fig. 1. Let (x_P, y_P) denote the coordinates of points $P \in \{A, B, C\}$ relative to the reference frame OXY in the figure, and let L_1 and L_2 be the lengths of the connector links. Clearly, a configuration of the mechanism can be described by the tuple $q = (y_A, y_B, x_C)$ because $x_A = x_B = y_C = 0$ in any configuration. Since the distances from A to B and from B to C must be kept equal to L_1 and L_2 , (1) is

$$\left. \begin{aligned} y_A^2 + x_C^2 &= L_1^2 \\ y_B^2 + x_C^2 &= L_2^2 \end{aligned} \right\} \quad (10)$$

from which we realize that the C-space corresponds to the intersection of two cylinders in the space of y_A , y_B , and x_C .

The velocity equation in (2) could now be obtained using the revolute- and prismatic-joint screws [18], but a more compact expression can in this case be derived by differentiating (10). Taking v_A and v_B as the input and output velocities, the differentiation yields

$$Lm = \begin{bmatrix} 0 & 2y_A & 2x_C \\ 2y_B & 0 & 2x_C \end{bmatrix} \begin{bmatrix} v_B \\ v_A \\ v_C \end{bmatrix} = 0$$

so that L_I , L_O , and L_P are, respectively

$$\begin{bmatrix} 0 & 2x_C \\ 2y_B & 2x_C \end{bmatrix}, \quad \begin{bmatrix} 2y_A & 2x_C \\ 0 & 2x_C \end{bmatrix}, \quad \begin{bmatrix} 2x_C \\ 2x_C \end{bmatrix}.$$

Any of the systems in (3)–(9) can now be written, and note that they can be solved analytically in this case. For example, if $L_1 = L_2 = 1$, the C-space has a single connected component composed of two ellipses intersecting on the x_C axis [see Fig. 2(a)], and the solutions of the systems in (3) reveal that the singularity set has six isolated configurations, marked in red in Fig. 2(a) bottom, with the following values of q :

$$\begin{aligned} (0, 0, 1), & \quad (0, 0, -1), & \quad (-1, -1, 0) \\ (1, 1, 0), & \quad (1, -1, 0), & \quad (-1, 1, 0). \end{aligned}$$

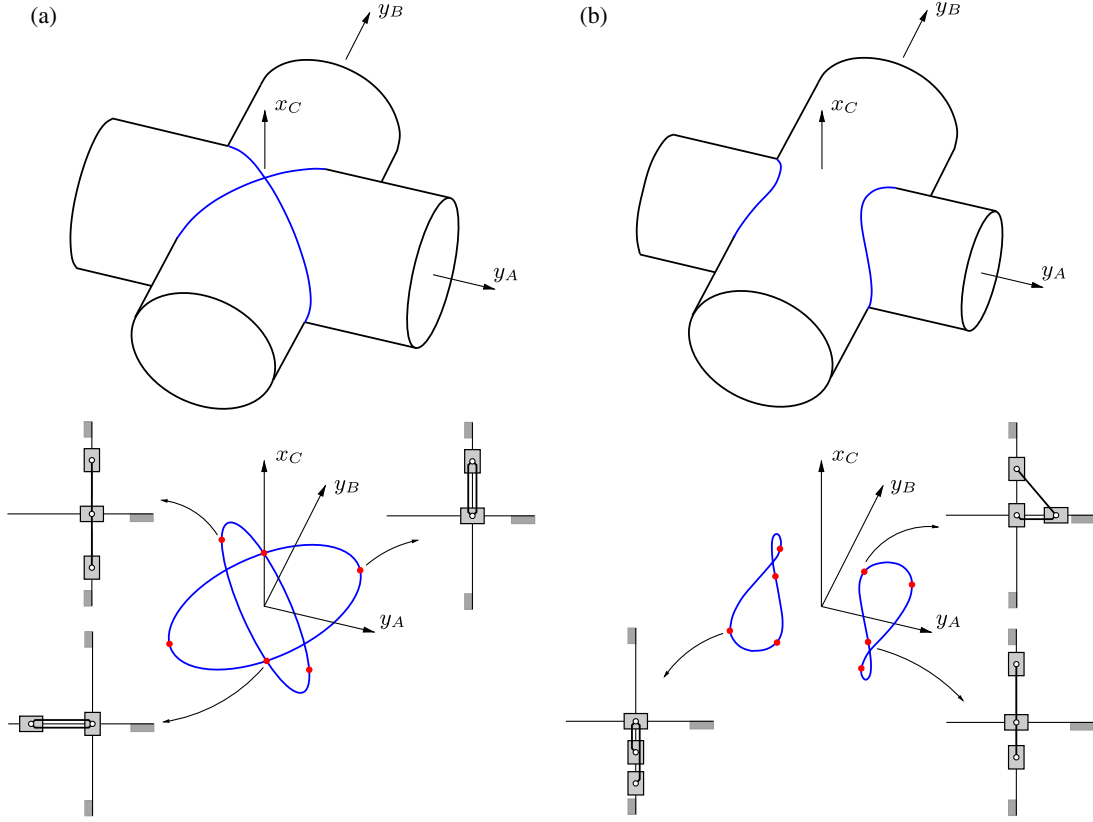


Fig. 2. Configuration space (in blue) and singularities (red dots) of the three-slider mechanism for (a) $L_1 = L_2$ and (b) $L_1 > L_2$ with some examples of singular configurations depicted. In this mechanism, the configuration space corresponds to the intersection of two cylinders at right angles.

All of these configurations satisfy both systems in (3) so that both the FIKP and the IIKP are indeterminate in them. It turns out, moreover, that the four configurations with $x_C = 0$ satisfy the systems in (6)–(8), meaning that they are both singularities belonging to each of the IO, II, and RPM types. Each of the other two configurations, which lie on the x_C axis, is a singularity of RI, RO, and RPM types, because it satisfies the systems (4), (5), and (9). These two configurations are in fact C-space singularities, i.e., points where the tangent space is ill defined. The C-space self-intersects at these points and presents a bifurcation that allows a change in the mode of operation from both sliders moving on the same side of the horizontal axis, $y_A y_B \geq 0$, to one slider moving on each side, $y_A y_B \leq 0$.

The topology of the C-space changes when $L_1 \neq L_2$. It no longer presents any bifurcation and is instead formed by two connected components [see Fig. 2(b)]. By solving (3) for $L_1 = 1$ and $L_2 = 0.8$, for example, eight singularities are obtained:

$$\begin{aligned} &(1, 0.8, 0), (-1, -0.8, 0), (1, -0.8, 0), (-0.6, 0, 0.8) \\ &(-1, 0.8, 0), (0.6, 0, -0.8), (0.6, 0, 0.8), (-0.6, 0, -0.8). \end{aligned}$$

As before, each configuration with $x_C = 0$ is a singularity of IO, II, and RPM types, while the other four configurations all belong to each of the RO and II types. There are no IIM-type singularities. In this case, to change the operation mode from $y_A \geq 0$ to $y_A \leq 0$, the mechanism has to be disassembled.

It must be noted that if a singularity identification were attempted by means of an input–output velocity equation, for instance, $y_A v_A = y_B v_B$, which holds for all configurations, then the singularities with $x_C = 0$ would not be detected.

V. ISOLATING THE SINGULARITY SETS

In the previous example, it was possible to solve all systems in (3)–(9) analytically, because they are simple, but this is not the case in general. The need to resort to a numerical method is often imperative in complex manipulators, where such systems are typically big and define positive-dimensional singularity sets. This section provides such a method by adapting a branch-and-prune strategy that was introduced earlier for position and workspace analysis [25], [26]. The method is based on formulating the systems in a quadratic form, then defining an initial box bounding all points of the solution sets, and, finally, exploiting the special form of the equations to iteratively remove portions of the box that contain no solution. This approach is advantageous because our solution sets can be of dimensions 0, 1, 2, or higher, and they are defined in the real field. Alternative approaches like homotopy methods are mainly designed to isolate zero- or one-dimensional solutions, and they must compute the roots in the complex field, which may increase the solution dimension unnecessarily [27]. Methods that are based on elimination exhibit similar drawbacks and easily explode in complexity with the problem size [28].

A. Equation Formulation

In order to formulate the equations, note that the structure of all systems in (3)–(9) is very similar. The first line is always (1), because all solution points must correspond to feasible configurations of the manipulator. The second line always involves \mathbf{L} or one of its submatrices, and the third line constrains the norm of some vector. For a manipulator that involves nonhelical lower pairs, the formulation proposed in [25] makes (1) directly adopt the form of a polynomial system of quadratic equations and allows writing the components of \mathbf{L} using linear terms only [23]. Thus, the second equation of all systems will be quadratic as well, and the third equation is directly a quadratic expression. The helical pair could also be treated using the developments in [25], but its treatment is here omitted for ease of explanation.

Written in the previous way, any one of the systems only involves monomials of the form x_i , x_i^2 , or $x_i x_j$, where x_i and x_j refer to any two of their variables. Thus, by introducing changes of variables of the form $x_k = x_i^2$ and $x_l = x_i x_j$, it is possible to expand the systems into the form

$$\begin{cases} \Lambda(\mathbf{x}) = \mathbf{0} \\ \Gamma(\mathbf{x}) = \mathbf{0} \end{cases} \quad (11)$$

where \mathbf{x} is a vector encompassing the variables of the original system and the newly introduced x_k and x_l ones, $\Lambda(\mathbf{x}) = \mathbf{0}$ is a collection of linear equations in \mathbf{x} , and $\Gamma(\mathbf{x}) = \mathbf{0}$ is a collection of scalar quadratic equations. In the systems of (4)–(7), there is a vector that must be different from zero, but since the technique can also handle nonstrict inequalities as explained later, this latter condition can be enforced by setting

$$\|\Omega^a\|^2 \geq \epsilon \quad (12)$$

for systems (4) and (7), and

$$\|\Omega^o\|^2 \geq \epsilon \quad (13)$$

for systems (5) and (6), where ϵ is a sufficiently small value. By using these inequalities, whose terms are also quadratic, some singularities might be overlooked, but ϵ can be made arbitrarily small, reducing the set of missed solutions to a negligible size.

B. Initial Bounding Box

It can be shown that all variables in the systems can only take feasible values within bounded intervals. For example, from the results in [25], one can readily define such intervals for the variables in \mathbf{q} , and the vector in the last line of each system has all of its components in the range $[-1, 1]$. In the case of (6), the feasibility intervals for the entries of Ω^{o*} can be readily obtained by mapping the known intervals using $\mathbf{A}_o^T \mathbf{u} = \Omega^{o*}$, where \mathbf{A}_o is formed by the columns of \mathbf{L} corresponding to the output velocity vector. A similar mapping, but using the columns of the input velocity, allows the determination of feasibility intervals for Ω^{a*} in (7). Finally, by propagating the intervals of the previous variables through the expressions $x_k = x_i^2$ and $x_l = x_i x_j$, it is straightforward to define bounded intervals for the x_k and x_l variables.

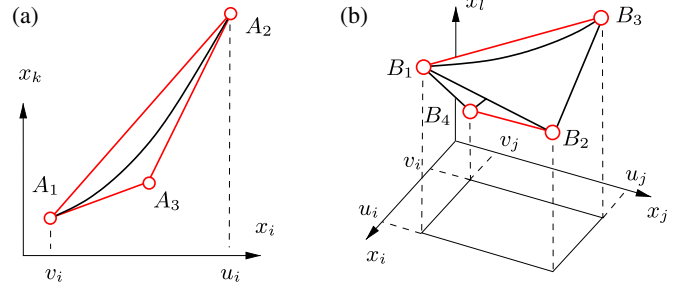


Fig. 3. Polytope bounds within box \mathcal{B}_c for (a) parabola and for (b) hyperbolic paraboloid.

In conclusion, from the Cartesian product of all such intervals, it is possible to define an initial box \mathcal{B} bounding the location of all points \mathbf{x} satisfying (11).

C. Numerical Solution

The algorithm for solving (11), together with (12) or (13) in the case of (4)–(7), applies two operations on \mathcal{B} : box *shrinking* and box *splitting*. Using box shrinking, portions of \mathcal{B} containing no solution are eliminated by narrowing some of its defining intervals. This process is repeated until either 1) the box is found to contain no solution and is marked as *empty*, 2) the box is “sufficiently” small and can be considered a *solution* box, or 3) the box cannot be “significantly” reduced. In the latter case, the box is bisected via box splitting, and the whole process is recursively applied to the resulting subboxes until all box sides are below a given threshold σ .

The crucial operation in this scheme is box shrinking, which is implemented as follows. The solutions falling in some subbox $\mathcal{B}_c \subseteq \mathcal{B}$ must lie in the linear variety defined by $\Lambda(\mathbf{x}) = \mathbf{0}$. Thus, we may shrink \mathcal{B}_c to the smallest possible box bounding this variety inside \mathcal{B}_c . The limits of the shrunk box along dimension x_i can be found by solving the linear programs

$$\text{LP1: Minimize } x_i, \text{ subject to: } \Lambda(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in \mathcal{B}_c$$

$$\text{LP2: Maximize } x_i, \text{ subject to: } \Lambda(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in \mathcal{B}_c.$$

However, observe that \mathcal{B}_c can be further reduced because the solutions must also satisfy all equations $x_k = x_i^2$ and $x_l = x_i x_j$ in $\Gamma(\mathbf{x}) = \mathbf{0}$. These equations can be taken into account by using their linear relaxations [25]. Note that, if $[v_i, u_i]$ denotes the interval of \mathcal{B}_c along dimension x_i , then we have the following.

- 1) The portion of the parabola $x_k = x_i^2$ lying inside \mathcal{B}_c is bound by the triangle $A_1A_2A_3$, where A_1 and A_2 are the points where the parabola intercepts the lines $x_i = v_i$ and $x_i = u_i$, and A_3 is the point where the tangent lines at A_1 and A_2 meet [see Fig. 3(a)].
- 2) The portion of the hyperbolic paraboloid $x_l = x_i x_j$ lying inside \mathcal{B}_c is bound by the tetrahedron $B_1B_2B_3B_4$, where the points B_1, \dots, B_4 are obtained by lifting the corners of the rectangle $[v_i, u_i] \times [v_j, u_j]$ vertically to the paraboloid [see Fig. 3(b)].

Thus, linear inequalities corresponding to these bounds can be added to LP1 and LP2. This usually produces a much larger reduction of \mathcal{B}_c , or even its complete elimination if one of the

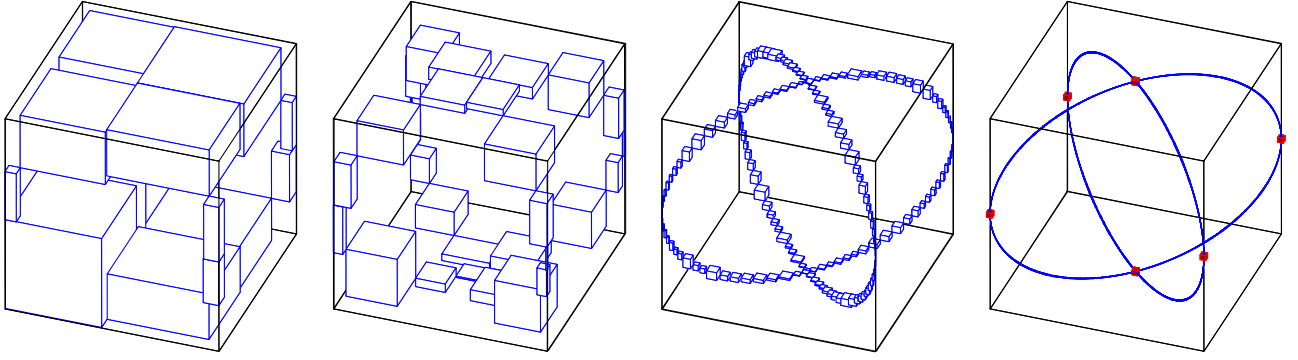


Fig. 4. Progression of the numerical algorithm on computing the configuration space of the three-slider mechanism for $L_1 = L_2$. From left to right, the sequence shows four stages of the computation, with the computed singularities of the mechanism shown overlaid in the right plot (in red). The method provided in this paper allows computing such boxes directly, without needing to isolate the whole configuration space. The boxes were magnified for clarity, because the box shrinking process yields too small boxes to be discerned.

linear programs is found unfeasible. In this step, the inequalities needed to model the conditions in (12) or (13) can also be taken into account by adding them to the linear programs.

As it turns out, the previous algorithm explores a binary tree of boxes whose internal nodes correspond to boxes that have been split at some time and whose leaves are either solution or empty boxes. The collection B of all solution boxes is returned as output, and it is said to form a *box approximation* of the singularity set, because it forms a discrete envelope of the set whose accuracy can be adjusted through the σ parameter. Notice that the algorithm is complete, in the sense that it will succeed in isolating all solution points accurately, provided that a small-enough value for σ is used.

The application of the method to the three-slider mechanism can be seen in Fig. 4, which shows box approximations of the C-space in blue color, which are obtained by applying the method to (10) only. The red boxes correspond to singular configurations obtained by solving the systems in (4)–(9).

D. Computational Cost

The computational cost of the algorithm can be evaluated by analyzing the cost of one iteration, and the number of iterations to be performed, both in terms of the number of bodies (n_b) and joints (n_j) of the manipulator. On the one hand, we can consider that an iteration includes the box shrinking process for a given box. This involves solving $2n_x$ linear programs, where n_x is the number of variables in (11). Since n_x depends linearly on n_b and n_j , and Karmarkar's bound for the complexity of linear programming is $O(n_x^{3.5})$ [29], we can conclude that the cost of one iteration is worst-case polynomial in n_b and n_j . On the other hand, it is difficult to predict how many iterations will be required to isolate all solutions. The number of iterations largely depends on the chosen σ , as well as on the dimension d of the singularity subset considered. For $d = 0$, the algorithm is quadratically convergent in the vicinity of the roots. For $d \geq 1$, the cost is inversely proportional to σ in the best case. For a fixed σ , however, the amount of solution boxes grows exponentially with d so that an initial guess on the execution time is usually made on the basis of d only. The value of d can be estimated

TABLE II
PERFORMANCE DATA ON THE REPORTED TEST CASES

	Sing. Set	d	$N_{eq}-N_{var}$	N_{boxes}	σ	ϵ	t (s)
Planar	RI	1	19-20	14903	0.01	10^{-5}	12
	RO	1	19-20	12773	0.01	10^{-5}	12
	IO	1	19-20	14906	0.01	10^{-5}	14
	II	1	19-20	13062	0.01	10^{-5}	13
	RPM	0	19-18	8	0.01	-	4
	IIM	-	21-20	0	0.01	-	2
Spatial	fixed ori.	2	25-27	146420	0.02	-	79
	fixed pos.	2	37-39	195982	0.25	-	2554

by noting that the singularity set is typically of codimension one relative to the C-space and using the Grübler–Kutzbach formula on n_b and n_j to determine the C-space dimension. Detailed properties of the algorithm, including an analysis of its completeness, correctness, and convergence order, are given in [25].

VI. TEST CASES

The performance of the approach is next illustrated in two test cases. The results were obtained using a parallelized version of the method implemented in C [30]. Table II summarizes the main performance data on the various singularity sets analyzed. For each set, we indicate its dimension (d), the number of equations (N_{eq}) and variables (N_{var}) in its defining system, the number of solution boxes returned by the method (N_{boxes}), the accuracy threshold assumed (σ), the ϵ parameter where applicable, and the time required to compute the set (t), in seconds, on a Xeon processor grid able to run 160 threads in parallel.

A. Planar Manipulator

The 2-DOF mechanism shown in Fig. 5 is used to illustrate the computation of each one of the singularity sets in detail. The inputs of the manipulator are the joint velocities of A and E , and the output is the velocity of point G . By gathering the loop-closure equations of the mechanism, and introducing two further equations to include the position of G , (1) can be formulated as

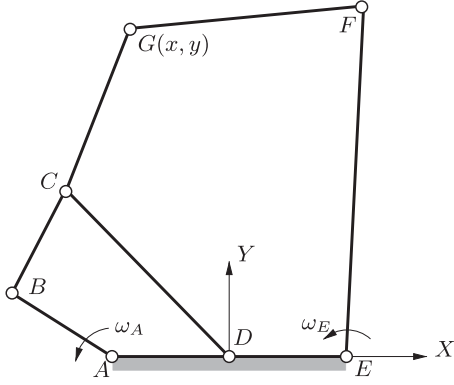


Fig. 5. Two-degree-of-freedom planar manipulator. The link dimensions are $AB = AD = BC = DE = 1$, $CD = FG = 2$, $CG = 1.5$, and $EF = 3$.

follows:

$$\left. \begin{aligned} \cos \theta_A + \cos \theta_B - 2 \cos \theta_D - 1 &= 0 \\ \sin \theta_A + \sin \theta_B - 2 \sin \theta_D &= 0 \\ 2 \cos \theta_D + \frac{3}{2} \cos \theta_C + 2 \cos \theta_G - 3 \cos \theta_E - 1 &= 0 \\ 2 \sin \theta_D + \frac{3}{2} \sin \theta_C + 2 \sin \theta_G - 3 \sin \theta_E &= 0 \\ -x + 2 \cos \theta_D + \frac{3}{2} \cos \theta_C &= 0 \\ -y + 2 \sin \theta_D + \frac{3}{2} \sin \theta_C &= 0 \end{aligned} \right\} \quad (14)$$

where $\theta_A, \theta_B, \theta_C, \theta_D, \theta_E$, and θ_G are the counterclockwise angles of links AB, BC, CG, DC, EF , and GF , respectively, relative to the ground, and x and y are the coordinates of point G relative to a fixed frame centered in D . The velocity equation of the manipulator may now be obtained by differentiating (14) with respect to all variables, but it could also be obtained using the twist loop equations, or by any other means. In order to achieve the desired quadratic formulation, the changes of variables $c_\tau = \cos \theta_\tau$ and $s_\tau = \sin \theta_\tau$ can now be applied for all $\tau \in \{A, B, C, D, E, G\}$. Since the variables c_τ and s_τ represent the cosine and sine of a variable, the circle equations $c_\tau^2 + s_\tau^2 = 1$ also need to be introduced into the systems for every angle θ_τ .

Given that the manipulator has two degrees of freedom, its configuration space is a surface, which is shown projected onto the x, y , and θ_A variables in Fig. 6. This surface was obtained from the computation of all solutions of (1) using the same numerical technique presented in the previous section. Note that by fixing x, y , and θ_A , there are still two possible positions of point F so that most of the points in this projection correspond, in fact, to two different configurations of the manipulator. Only the points where E, F , and G are aligned represent a single configuration, and these are exactly the boundaries of the two “holes” that the surface presents.

The singularity set is generally of lower dimension than the configuration space so that only curves or points are to be expected in the solution set of all systems of equations. The result of the computation of each singularity type is shown in Figs. 7 and 8, projected onto the output and one input (x, y, θ_A), and onto the output only, respectively. In Fig. 7, the configuration space is shown in blue, separated in two parts so that a cross

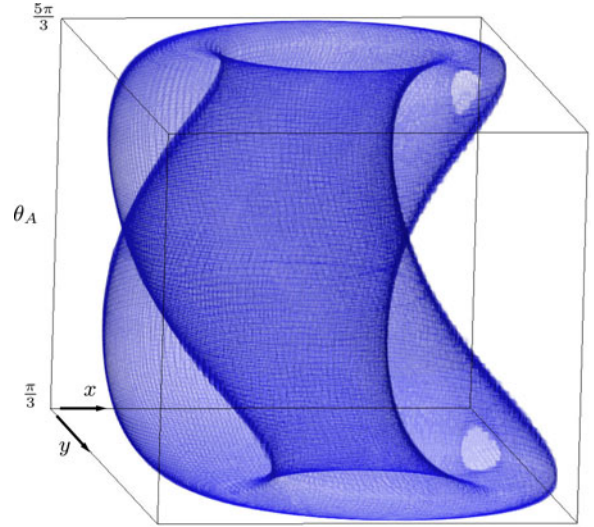


Fig. 6. 2-D configuration space of the manipulator in Fig. 5 computed at $\sigma = 0.1$. Two holes can be seen, whose boundary corresponds to configurations where E, F , and G are aligned.

section can be seen, but both parts are actually connected through π and $-\pi$, as shown in Fig. 6. The gray area in Fig. 8 represents all attainable positions of point G , i.e., the workspace of the manipulator.

As it turns out, this manipulator contains no IIM configurations, and the computation of this type of singularity gives no box as output. On the contrary, there are eight distinct RPM singularities, which in these projections appear coincident in pairs as four orange boxes, corresponding to the two possible locations of F . Using a different projection, for instance, onto $(\theta_A, \theta_E, \theta_D)$, the eight boxes appear separated.

The green curves correspond to singularities that are of both the RI and IO types. These configurations can be seen to contour the two “holes” of the configuration space in this projection. The red curves correspond to configurations simultaneously belonging to the RO and II types. Even if the curves for RI and IO seem to coincide everywhere, there are some IO configurations that are not of RI type, and the same happens for II and RO singularities, respectively. This is illustrated in Fig. 7 with a closeup on the left that shows only the output of computing RI singularities. These gaps on the curves of RI and RO, which can be found by properly adjusting the ϵ parameter, coincide with the location of the RPM singularities, and hence, the RPM singularities are also of II and IO types (but not of RI or RO types). Fig. 8(a) shows an example of an (RPM, II, IO) singularity, while Fig. 8(b) and (c) shows examples of (RI, IO) and (RO, II) singularities, respectively.

Fig. 7 also shows yellow (arcs of) curves that correspond to configurations where points D, B , and G are aligned. For each yellow-marked triple (x, y, θ_A) , with D, B , and G collinear, there are two possible locations of point C . In contrast, point C is uniquely determined for any other (x, y, θ_A) . Thus, a point on a yellow curve corresponds to four different configurations, because each of points C and F can have two positions. As is visible in the figure, these are the points of self-intersection of

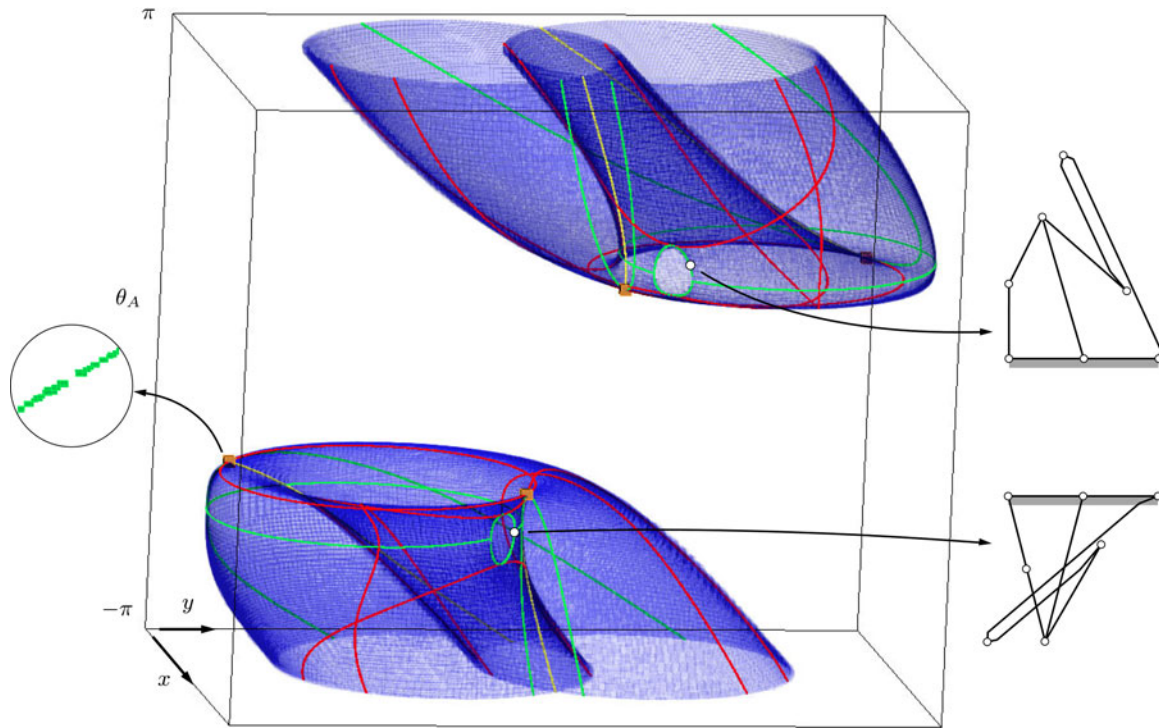


Fig. 7. Singular configurations of the mechanism in Fig. 5 shown overlaid onto a projection of its configuration space. Different colors are used to identify the several singularity types encountered: green for the RI and IO types, red for the RO and II types, and orange for the RPM type.

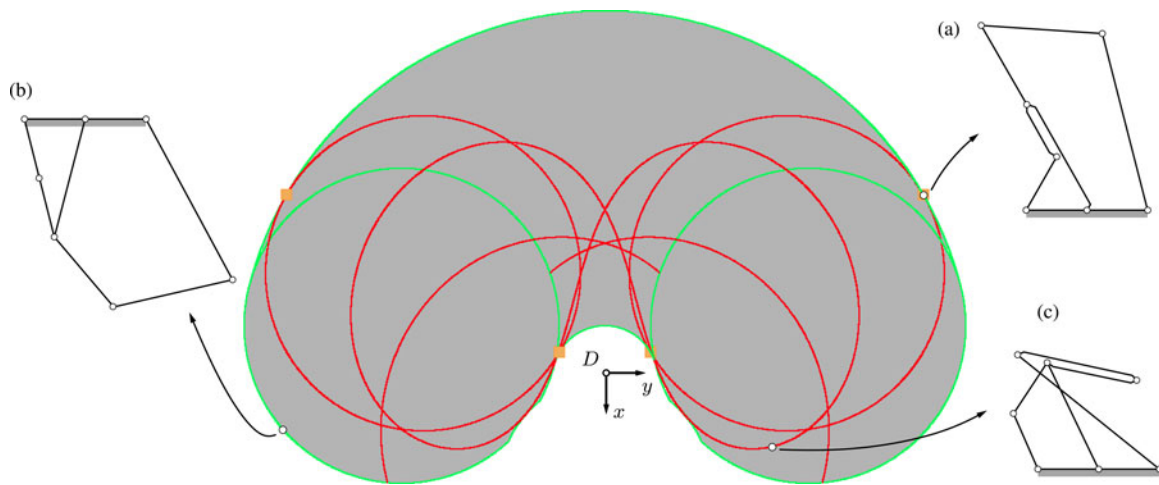


Fig. 8. Projection of the plot in Fig. 7 to the (x, y) plane. (a) Singularity of RPM, IO, and II types. (b) Singularity of RI and IO types. (c) Singularity of RO and II types.

the *projection* of the configuration space on the (x, y, θ_A) space. The four configurations for each point can be identified with the two sides (“in” and “out”) of the two sheets that intersect. The configuration space itself has no self-intersections as there are no configuration space, or IIM-type, singularities. The yellow points are only singularities of the projection map. The four orange vertices of the yellow curve arcs in Fig. 7 correspond to the eight configurations where D , B , G , and C are collinear. These are the mechanism’s RPM-type singularities. They are branching points for the inverse kinematics solution, because point C can move in two different ways out of such a configuration. The

other configurations where the working mode changes are those where E , F , and G are aligned.

Using the same color code, Figs. 9 and 10 show the projection of the results onto the 3-D space of the two input angles and one passive joint angle $(\theta_A, \theta_E, \theta_D)$ and onto the 2-D input space only. The eight RPM singularities appear separated. As before, for fixed values of θ_A , θ_E , and θ_D , there are still two possible locations of point C in general, and almost all points in this projection correspond to two distinct configurations of the manipulator. It can be seen that the configuration space presents four “holes” in these projections. These four contours are made

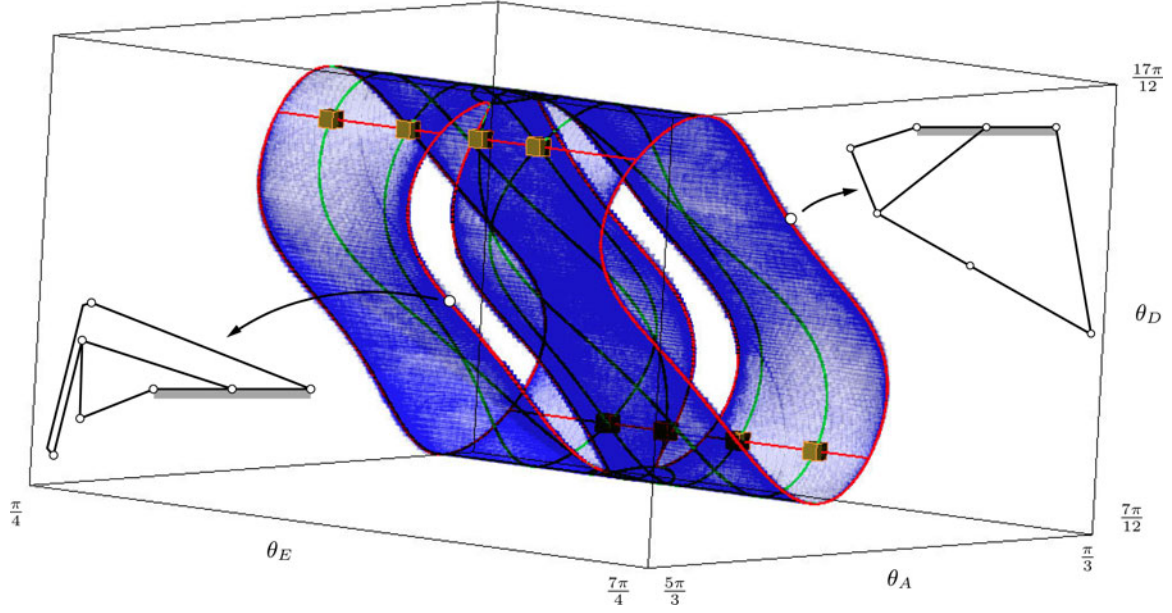


Fig. 9. Projection of the configuration space and the computed singularities to the $(\theta_A, \theta_E, \theta_D)$ space, together with two configurations where C , G , and F are aligned. Green corresponds to the RI and IO types, red to the RO and II types, and orange to the RPM type. There are no singularities of IIM type.

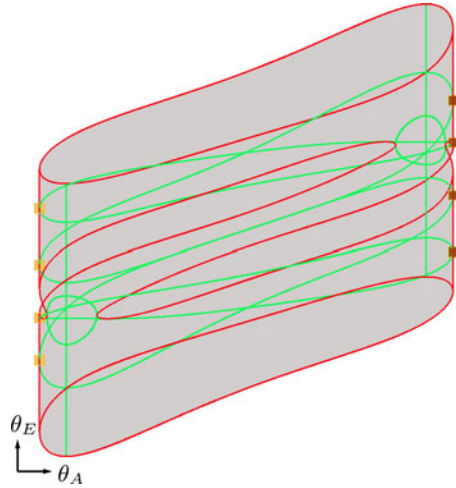


Fig. 10. Projection of the plot in Fig. 9 to the (θ_A, θ_E) space.

of those configurations where G , C , and F are aligned and there is only one possibility for C . Note that none of these “holes” coincides with one in the previous projection; however, once again, crossing each curve allows the transition between two different working modes. One can imagine the two working modes as the two “sides” of the surface of the configuration-space projection. To “get to the opposite side,” i.e., to change working mode, the motion curve must “go through a hole.”

B. Spatial Manipulator

To illustrate the method on a spatial manipulator, we next apply it to the Stewart–Gough platform. For the sake of conciseness, we concentrate on computing the forward singularity locus only, which is the most relevant and representative of the kind of complexity to be confronted in the spatial case. This

amounts to formulating and solving the left system in (3) using the proposed approach.

The platform consists of a moving plate connected to a fixed base by means of six legs, where each leg is a universal-prismatic-spherical chain (see the left side of Fig. 11). The six prismatic joints are actuated, allowing to control the six degrees of freedom of the platform, and the remaining joints are passive [31].

The assembly constraints can be formulated as follows. Let A_i and B_i be the center points of the universal and spherical joints. Let also \mathcal{F}_1 and \mathcal{F}_2 be fixed and mobile reference frames, centered in O and P , respectively. Then, the constraints imposed by each leg on the moving plate can be written as

$$\mathbf{p}^{\mathcal{F}_1} = \mathbf{a}_i^{\mathcal{F}_1} + d_i \mathbf{d}_i^{\mathcal{F}_1} - \mathbf{R} \mathbf{b}_i^{\mathcal{F}_2} \quad (15)$$

$$\|\mathbf{d}_i^{\mathcal{F}_1}\|^2 = 1 \quad (16)$$

where $\mathbf{p}^{\mathcal{F}_1}$, $\mathbf{a}_i^{\mathcal{F}_1}$, and $\mathbf{b}_i^{\mathcal{F}_2}$ are the position vectors of points P , A_i , and B_i in the indicated frames, and $\mathbf{d}_i^{\mathcal{F}_1}$ is a unit vector along the i th leg, expressed in frame \mathcal{F}_1 . In addition, d_i is the length of the leg, which represents the displacement of the prismatic joint, and \mathbf{R} is the rotation matrix that provides the orientation of \mathcal{F}_2 relative to \mathcal{F}_1 . The pose of the platform is given by $(\mathbf{p}^{\mathcal{F}_1}, \mathbf{R})$.

In this case, (1) is the system formed by (15) and (16) for all legs, together with the conditions

$$\begin{aligned} \|\mathbf{s}\|^2 &= 1, & \mathbf{s} \cdot \mathbf{t} &= 0 \\ \|\mathbf{t}\|^2 &= 1, & \mathbf{s} \times \mathbf{t} &= \mathbf{w}_i \end{aligned}$$

that force $\mathbf{R} = [\mathbf{s}, \mathbf{t}, \mathbf{w}]$ to represent a valid rotation.

The velocity equation can be obtained by writing the expression of the output twist $\hat{\mathbf{T}}$ following each leg

$$\hat{\mathbf{T}} = \Omega_i^a \hat{\mathbf{S}}_i^a + \sum_{j=1}^5 \Omega_{i,j}^p \hat{\mathbf{S}}_{i,j}^p \quad (17)$$

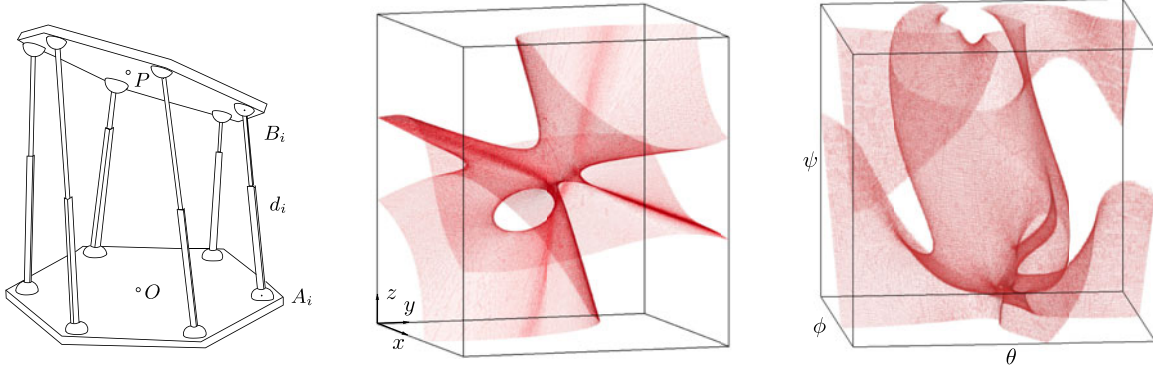


Fig. 11. (Left) Stewart–Gough platform. (Center and right) Slices of its forward singularity set for a constant orientation given by $\phi = -2^\circ$, $\theta = 30^\circ$, and $\psi = -87^\circ$, and for the fixed position $\mathbf{p}^{F1} = [10, 10, 10]^T$. The position and orientation variables of the platform have been limited to the ranges $[-100, 100]$ and $[-90^\circ, 90^\circ]$, respectively.

where $\hat{\mathbf{S}}_i^a$ and $\hat{\mathbf{S}}_{i,j}^p$ are the unit twists of the active and the five passive joints of the i th leg, respectively. By gathering (17) for all legs, we obtain a 36×42 matrix \mathbf{L} , and a velocity vector \mathbf{m} containing the six components of the output twist, the six active velocities of the prismatic joints, and the 30 passive joint velocities of the universal and spherical joints. This results in a relatively large system of equations, but by multiplying each side of (17) by a unit screw reciprocal to all passive joint twists of the leg, we can conclude that the forward singularities are the configurations for which the conventional screw Jacobian \mathbf{J} is singular [18], [32]. This condition is advantageous because \mathbf{J} is only 6×6 , and generally produces a much smaller system.

For some configurations, the space of reciprocal screws of a given leg may be of dimension larger than one, and (17) should be multiplied by a whole basis of reciprocal screws of the leg [33]. In the Stewart–Gough platform, this can only happen when the center of the leg’s spherical joint is in the plane of the two revolute-joint axes of the universal joint, which result in a singularity of RPM type. Since joint limits and other constraints typically exclude such singularities in real platforms, we will not compute them here.

Two slices of the forward singularity locus are shown in Fig. 11, computed at a constant orientation and at a constant position of the platform. Alternative slices could also be obtained if desired, simply by fixing a different set of pose parameters. The geometric dimensions assumed here correspond to the academic manipulator studied in [6]. The Euler angles ϕ , θ , and ψ are those for which $\mathbf{R} = \mathbf{R}_z(\psi)\mathbf{R}_y(\theta)\mathbf{R}_x(\phi)$, which also coincide with the ones assumed in [6]. From the results in Table II, we note that it is computationally much harder to compute the constant position slice. This agrees with the fact that the system to be solved is much larger, and its equations are highly nonlinear, in comparison with those of the constant orientation slice.

VII. CONCLUSION

This paper has proposed a method for the numerical computation and detailed classification of the entire singularity set of a lower-pair manipulator with arbitrary geometry. Systems of equations have been defined to compute the set, and each one

of the singularity subsets has been identified in [18]. To solve any of the systems, a numerical method that is based on linear relaxations has been proposed, which can obtain a box approximation of the solution set with the desired accuracy, even in the presence of self-intersections or dimension changes in the set [23], [34]. The approach is based on a recursive segmentation and reduction of the search space and is particularly practical and useful on low degree-of-freedom manipulators like the one in Section VI-A. This example has been chosen for its high illustrative value, since it allows a clear analysis and presentation of the results in a moderate-dimensional case. It also shows how complex can be the topology of the configuration space and its singularity-induced partitions. As demonstrated in Section VI-B, the analysis of manipulators with higher dimensional singularity sets does not add fundamental difficulties to the method, other than increasing the computation times, as with any other method. The detailed interpretation and visualization of the singularity sets of these and other manipulators will be the subject of future work. Additional work is envisaged to also extend the developments to deal with redundant manipulators [18], [35].

REFERENCES

- [1] C. Gosselin and J. Wang, “Singularity loci of a special class of spherical three-degree-of-freedom parallel mechanisms with revolute actuators,” *Int. J. Robot. Res.*, vol. 21, no. 7, pp. 649–659, 2002.
- [2] I. A. Bonev. (2002). “Geometric analysis of parallel mechanisms,” Ph.D. dissertation, Faculté des Sciences et de Génie, Univ. de Laval. [Online]. Available: <http://goo.gl/IKhw6>
- [3] I. A. Bonev, D. Zlatanov, and C. Gosselin, “Singularity analysis of 3-DOF planar parallel mechanisms via screw theory,” *ASME J. Mech. Design*, vol. 125, pp. 573–581, 2003.
- [4] I. Ebert-Uphoff, J. Lee, and H. Lipkin, “Characteristic tetrahedron of wrench singularities for parallel manipulators with three legs,” *Proc. Inst. Mech. Eng. C, J. Mech. Eng. Sci.*, vol. 216, no. 1, pp. 81–93, 2002.
- [5] J. Wang and C. Gosselin, “Singularity loci of a special class of spherical 3-DOF parallel mechanisms with prismatic actuators,” *ASME J. Mech. Design*, vol. 126, no. 2, pp. 319–326, 2004.
- [6] H. Li, C. Gosselin, M. Richard, and B. St-Onge, “Analytic form of the six-dimensional singularity locus of the general Gough–Stewart platform,” *ASME J. Mech. Design*, vol. 128, pp. 279–288, 2006.
- [7] I. A. Bonev and C. Gosselin, “Analytical determination of the workspace of symmetrical spherical parallel mechanisms,” *IEEE Trans. Robot.*, vol. 22, no. 5, pp. 1011–1017, Oct. 2006.
- [8] P. Ben-Horin and M. Shoham, “Singularity condition of six-degree-of-freedom three-legged parallel robots based on Grassmann–Cayley algebra,” *IEEE Trans. Robot.*, vol. 22, no. 4, pp. 577–590, Aug. 2006.

- [9] J.-P. Merlet, "A formal-numerical approach for robust in-workspace singularity detection," *IEEE Trans. Robot.*, vol. 23, no. 3, pp. 393–402, Jun. 2007.
- [10] D. Kanaan, P. Wenger, S. Caro, and D. Chablat, "Singularity analysis of lower mobility parallel manipulators using Grassmann–Cayley algebra," *IEEE Trans. Robot.*, vol. 25, no. 5, pp. 995–1004, Oct. 2009.
- [11] J. Schadlbauer and M.-L. Husty, "A complete analysis of the 3-RPS manipulator," in *Machines and Mechanism*. New Delhi, India: Narosa, 2011, pp. 410–419.
- [12] J. Borriàs, F. Thomas, and C. Torras, "Singularity-invariant families of line-plane 5-SPU platforms," *IEEE Trans. Robot.*, vol. 27, no. 5, pp. 837–848, Oct. 2011.
- [13] M.-L. Husty, J. Schadlbauer, S. Caro, and P. Wenger, "Self-motions of 3-RPS manipulators," *New Trends Mech. Mach. Sci., Theory Appl. Eng.*, vol. 7, pp. 121–130, 2012.
- [14] C. Gosselin and J. Angeles, "Singularity analysis of closed-loop kinematic chains," *IEEE Trans. Robot. Autom.*, vol. 6, pp. 281–290, Jun. 1990.
- [15] D. Zlatanov, R. Fenton, and B. Benhabib, "Singularity analysis of mechanisms and robots via a motion-space model of the instantaneous kinematics," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1994, pp. 980–985.
- [16] D. Zlatanov, R. Fenton, and B. Benhabib, "A unifying framework for classification and interpretation of mechanism singularities," *ASME J. Mech. Design*, vol. 117, pp. 566–572, 1995.
- [17] D. Zlatanov, B. Benhabib, and R. Fenton, "Identification and classification of the singular configurations of mechanisms," *Mech. Mach. Theory*, vol. 33, pp. 743–760, 1998.
- [18] D. Zlatanov. (1998). "Generalized singularity analysis of mechanisms," Ph.D. dissertation, Univ. Toronto, Toronto, ON, Canada. [Online]. Available: <http://goo.gl/rnAUy>
- [19] F. Park and J. Kim, "Singularity analysis of closed kinematic chains," *ASME J. Mech. Design*, vol. 121, pp. 32–38, 1999.
- [20] O. Bohigas, D. Zlatanov, L. Ros, M. Manubens, and J. M. Porta, "Numerical computation of manipulator singularities," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 1351–1358.
- [21] O. Bohigas, M. Manubens, and L. Ros, "Singularities of non-redundant manipulators: A short account and a method for their computation in the planar case," *Mech. Mach. Theory*, vol. 68, pp. 1–17, 2013.
- [22] J. G. De Jalón and E. Bayo, *Kinematic and Dynamic Simulation of Multi-body Systems*. New York, NY, USA: Springer-Verlag, 1993.
- [23] O. Bohigas. (2013). "Numerical computation and avoidance of manipulator singularities," Ph.D. dissertation, Univ. Politècnica de Catalunya, Barcelona, Spain. [Online]. Available: <http://goo.gl/wlR0i>
- [24] J. Selig, *Geometric Fundamentals of Robotics*. New York, NY, USA: Springer-Verlag, 2005.
- [25] J. M. Porta, L. Ros, and F. Thomas, "A linear relaxation technique for the position analysis of multi-loop linkages," *IEEE Trans. Robot.*, vol. 25, no. 2, pp. 225–239, Apr. 2009.
- [26] O. Bohigas, M. Manubens, and L. Ros, "A complete method for workspace boundary determination on general structure manipulators," *IEEE Trans. Robot.*, vol. 28, no. 5, pp. 993–1006, Oct. 2012.
- [27] C. W. Wampler and A. J. Sommese, "Applying numerical algebraic geometry to kinematics," in *21st Century Kinematics*. New York, NY, USA: Springer-Verlag, 2013, pp. 125–160.
- [28] D. Cox, J. Little, and D. O'Shea, *An Introduction to Computational Algebraic Geometry and Commutative Algebra*, 3rd ed. New York, NY, USA: Springer-Verlag, 2007.
- [29] N. K. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, pp. 373–395, 1984.
- [30] J. M. Porta, L. Ros, O. Bohigas, M. Manubens, C. Rosales, and L. Jaillet, "The CUIK suite: Motion analysis of closed-chain multibody systems," *IEEE Robot. Autom. Mag.*, 2013, in press.
- [31] J.-P. Merlet, *Parallel Robots*. New York, NY, USA: Springer-Verlag, 2006.
- [32] K. J. Waldron and K. H. Hunt, "Series-parallel dualities in actively coordinated mechanisms," *Int. J. Robot. Res.*, vol. 10, pp. 473–480, 1991.
- [33] D. Zlatanov, R. Fenton, and B. Benhabib, "Analysis of the instantaneous kinematics and singular configurations of hybrid-chain manipulators," in *Proc. ASME 23rd Biennial Mech. Conf.*, 1994, vol. 72, pp. 467–476.
- [34] O. Bohigas, D. Zlatanov, M. Manubens, and L. Ros, "On the numerical classification of the singularities of robot manipulators," in *Proc. ASME Int. Design Eng. Tech. Conf.*, 2012, pp. 1287–1296.
- [35] A. Müller, "On the terminology and geometric aspects of redundant parallel manipulators," *Robotica*, vol. 31, no. 1, pp. 137–147, 2013.



Oriol Bohigas received the double M.S. degree in mechanical engineering from the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, and in aeronautical engineering from the Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), Toulouse, France, in 2006. In 2013, he received the Ph.D. degree in robotics from the UPC.

After a research internship with Airbus France, Toulouse, he worked for two years as a Space Engineer with the Centre National d'Etudes Spatiales, Toulouse. He is currently with the Kinematics and Robot Design Group, the Institut de Robòtica i Informàtica Industrial, Barcelona. His research interests include workspace and singularity analysis of robot mechanisms.



Dimitar Zlatanov received the Diploma in mathematics and mechanics from the University of Sofia, Sofia, Bulgaria, in 1989 and the Ph.D. degree in mechanical engineering from the University of Toronto, Toronto, ON, Canada, in 1998.

He has held positions with Laval University, Quebec City, QC, Canada; the University of Innsbruck, Innsbruck, Austria; and Tokyo City University, Setagaya, Japan. He is currently on the faculty with the Department of Mechanical and Energy Engineering, University of Genoa, Genoa, Italy. His research interests include the design, kinematics, dynamics and control of mechanisms and robotic systems.

ests include the design, kinematics, dynamics and control of mechanisms and robotic systems.



Lluís Ros received the mechanical engineering degree in 1992 and the Ph.D. degree (Hons.) in industrial engineering in 2000, both from the Universitat Politècnica de Catalunya, Barcelona, Spain.

From 1993 to 1996, he worked with the Control of Resources Group, Institut de Cibernètica, Barcelona. He was a Visiting Scholar with York University, Toronto, ON, Canada, in 1997; the University of Tokyo, Tokyo, Japan, in 1998; and the Laboratoire d'Analyse et Architecture des Systèmes, Toulouse, France, in 1999. He joined the Institut de

Robòtica i Informàtica Industrial, Barcelona, in 1997, where he has been an Associate Researcher with the Spanish National Research Council since 2005. His research interests include geometry and kinematics, with applications to robotics, computer graphics, and machine vision.



Montserrat Manubens received the degree in mathematics from the Universitat de Barcelona, Barcelona, Spain, in 2001 and the Ph.D. degree (Hons.) in computer algebra from the Universitat Politècnica de Catalunya, Barcelona, in 2008.

From 2009 to 2010, she worked with the Robotics Group, Institut de Recherche en Communications et Cybernétique de Nantes, Nantes, France, in the analysis of cuspidal robots. Since 2011, she has been a Juan de la Cierva contractor with the Institut de Robòtica i Informàtica Industrial, Barcelona. Her current research interests include mathematics and kinematics, with applications to robotics.

research interests include mathematics and kinematics, with applications to robotics.



Josep M. Porta received the engineering degree in computer science in 1994 and the Ph.D. degree (Hons.) in artificial intelligence in 2001, both from the Universitat Politècnica de Catalunya, Barcelona, Spain.

From 2001 to 2003, he was a Postdoctoral Researcher with the University of Amsterdam, Amsterdam, The Netherlands, doing research in autonomous robot localization using vision. Currently, he is an Associate Researcher with the Spanish National Research Council, Institut de Robòtica i Informàtica

Industrial, Barcelona. His current research interests include planning under uncertainty and computational kinematics.