

Geometric Methods in Robotics

Lluís Ros¹, Federico Thomas¹, Josep Maria Porta¹, Carme Torras¹, Vicente Ruiz¹,
Tom Creemers¹, Josep Cantó², Francesc Corcho² and Assumpta Sabater³

¹ Inst. of Robotics and Industrial Informatics (CSIC-UPC), Llorens Artigas 4-6, 08028 Barcelona
E-mails: {llros, fthomas, jporta, vruiz, ctorras, tcreemers}@iri.upc.es

² Dep. of Chemical Engineering, Technical Univ. of Catalonia, Av. Diagonal 647, 08028 Barcelona
E-mails: {francesc.corcho, josep.canto}@upc.es

³ Dep. of Applied Mathematics III, Technical Univ. of Catalonia, C. Colom 1, 08222 Terrassa
E-mail: assumpta.sabater-pruna@upc.es

Abstract—This article presents a principal research line within the group of Geometric Methods in Robotics: the development of efficient algorithms for kinematic constraint solving, with applications to Robotics, Computer-Aided Geometric Design and Structural Biology. The multidisciplinary nature of the topic justifies the participation of researchers from different areas. Currently, the group involves members of the Institute of Robotics and Industrial Informatics and the departments of Chemical Engineering and Mathematics of the Technical University of Catalonia (UPC).

Index Terms—Kinematic Constraint Solving, Robotics, Structural Biology, Computer-Aided Design.

I. INTRODUCTION

The purpose of this research is to develop efficient algorithms for solving the following strongly NP-hard problem: given an arbitrary collection of kinematic constraints among a set of solids, generate all spatial configurations of these solids that satisfy them. When the number of feasible solutions is infinite, the algorithm must be able to find a discretization of the whole solution space or, when needed, it must find the configurations that minimize a given objective function in the involved variables.

This abstract formulation has concrete instantiations in several problems of Robotics, CAD or Structural Biology, where the considered solids are the rigid links of a robot mechanism, the geometric elements of a CAD design, or the atoms of a molecule, respectively. In those settings, the kinematic constraints are induced by the kinematic pairs of the robot (such as prismatic, revolute, or spherical joints), the geometric constraints of the design (such as incidence, parallelism, orthogonality, scale or size) or the atomic interactions of the molecule (such as covalent bonds, hydrogen bonds or hydrophobic/hydrophilic interactions). We next examine some applications in detail, in each of these fields.

A. Inverse kinematics of robots and molecules

Although the inverse kinematic problem has been solved for specific architectures, the inexistence of a general procedure for solving inverse kinematic problems usually prevents a specific robot design to be used. Having a general method able to solve any architecture (without limitations on the number of kinematic loops, the type of the joints, or the degree of redundancy) would remove this limitation. This is a relevant problem nowadays, given the widespread use of parallel robots (that lead to kinematic loops between the base and the platform), modular reconfigurable robots (that are able to adopt very different architectures whose inverse kinematics is unknown) and humanoid robots (that frequently originate kinematic loops when they grasp an object with both hands, or when they perform motions in contact with the environment). A solution to this problem would also have important applications in Structural Biology, where a common problem is to compute all spatial configurations of large biomolecules, taking into account the constraints of their kinematic loops, in order to determine their biological function in the cell.

B. Collision free path planning and docking

Although there exist efficient path planners for robots with a tree-like architecture, there does not exist an efficient method for this purpose in robots that contain kinematic loops. These loops arise in the coordinated manipulation of an object by several manipulators, in the path planning of parallel robots, or in tasks requiring some elements of the robot to be in contact with the environment. This problem also arises in Drug Design, where one has to elucidate whether a small molecule (such as that of a certain drug) fits in a specific cavity of a larger molecule (such as a protein in the cellular membrane).

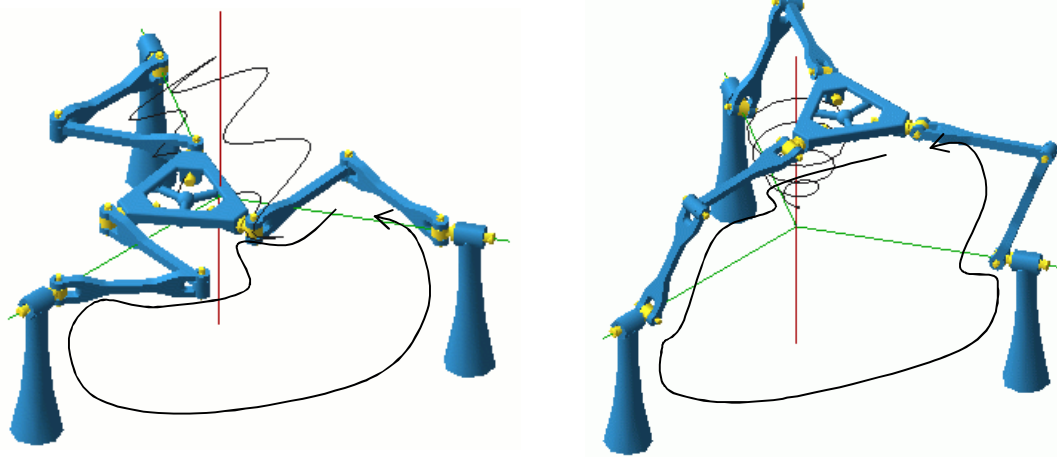


Fig. 1: Kinematic loops appearing in the direct kinematics of parallel manipulators. The figure shows a simulation of the parallel robot 3-URU DYMO, by Dimitri Zlatanov. A kinematic loop is created between any two legs, and all closure constraints of such loops must be simultaneously solved, in order to study the possible motions of the platform. Currently, no system exists that is able to solve the direct or inverse kinematics of a robot of arbitrary architecture.

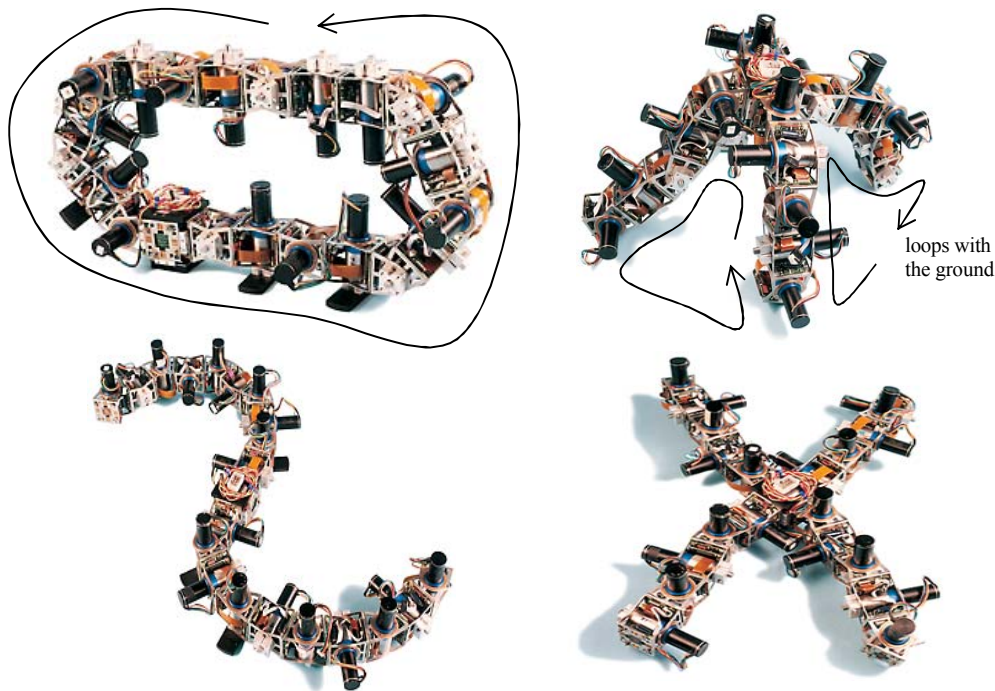


Fig. 2: Kinematic loops in modular reconfigurable robots. The figure shows the robot POLYBOT, designed in the Palo Alto Research Institute of California. Given its modular structure, the robot can adopt a variety of configurations, many of them creating kinematic loops that must be taken into account when analyzing the mobility of the robot or when planning its motions. Since the inverse kinematics of a general robot is not a solved problem, such robots frequently exhibit simpler motions than those they could actually perform, given its rich architecture.

C. Solving constrained designs in CAD

A fundamental problem of computer-aided mechanical design is how to move a set of objects to its final position in an assembly. Usually, this is done by manually specifying a number of coordinate transformations between the reference frames of the distinct objects - a hard task that quite often introduces position errors in the assembly. As an alternative, we propose to specify the final positions of the solids (as well as their allowed relative motions) in an interactive way, using constraints such as that the face of one object must be coplanar with the face of another object, or that a given edge coincides with another one, and so on. Our constraint-based approach would allow an easy specification and simplification of this task. We must point out that, although this is a long-standing challenge of Robotics and Artificial Intelligence, none of the systems developed so far, such as LAMA [1], RAPT [2], TLA [3], or AUTOPASS [4], show a significant performance in cases of practical interest.

The rest of the paper is structured as follows. Section II outlines the historical development of the group. Section III summarizes previous related work in the literature. Section IV presents a recently developed methodology within the group and, finally, Section V shows some specific problem instances that have been solved using it.

II. GROUP GENESIS

Several members of the group have been working in inverse kinematics problems with applications to Robotics and CAD-CAM for the past fifteen years. The first publications date back to the Ph.D. thesis of Federico Thomas [5,6,7], where a method based on Group-Theory was proposed to solve a system of kinematic constraints. Soon after, during the period 1988-1991 and within the project "Constraint-based Spatial Reasoning" (Spanish CICYT reference TIC88-0197), led by Carme Torras, this work was extended to pursue the solutions within specified ranges for the degrees of freedom of the mechanism. The results for planar chains and spherical mechanisms appeared in the Ph.D. thesis of Enric Celaya [8] and these techniques were later applied to the analysis and synthesis of multi-loop mechanisms [9] and redundant manipulators [10]. An algorithm to specify the positions of several solids subject to a number of geometric constraints between them was given in [11], extending other previous developments [12]. Later on, in the period 1996-1999, within the project "Analysis of Spatial Constraints and their Applica-

tion to Mechanical Design and Robot Task Simulation" (Spanish CICYT reference TIC96-0721-C02-01), led by Federico Thomas, this group demonstrated that the general interval-based methods used to solve systems of nonlinear equations, when applied to inverse kinematics problems, are very efficient and easy to implement when compared to other techniques. Several versions of this basic idea were developed in the Ph.D. thesis of Albert Castellet and published in [13,14,15]. Next, the group focused on interval methods based on the properties of Bernstein polynomials, devising a fast algorithm [16,17] that has been later simplified using the properties of multilinear functions [18,19,20].

Currently, the group involves the nine authors of this paper, who collaborate within the R+D project "Design and Implementation of Efficient Parallelized Algorithms of Distance Geometry, with Applications in Computational Kinematics and Proteomics", led by Lluís Ros and funded by the Spanish Ministry of Science (with reference TIC2003-03396).

The group recently received international recognition by being granted the organization of the International Symposium on Advances in Robot Kinematics, in the 2002 edition.

III. RELATED WORK

Although the formulated problem can be approached by using geometric constructive techniques [21], only the algebraic approaches have proved general enough to handle all problem instances. These consist in translating the original kinematic problem into a system of algebraic equations that is then solved using any suitable standard technique. Unfortunately, a good solution to both algebrization and resolution, treated as independent problems, does not necessarily lead to an efficient solution of the kinematic problem. One of the aims of our work is to find a good combination of algebrization and resolution, so that the whole process is easy to understand and implement, and yet computationally efficient in practice.

Finding all solutions to a system of nonlinear polynomial equations within some finite domain is an ubiquitous problem for which a wealth of resolution techniques has been proposed. Reviews of these methods in the context of Robotics, CAD/CAM and Structural Biology can be found for example in [22], [23], and [24], respectively. Broadly speaking, the proposed methods fall into three categories, depending on whether they use algebraic geometry, continuation or interval-based tech-

niques. While methods in the first two categories are in theory *complete* (they are able to find *all* solutions if these exist in a finite number) and *general* (they can tackle *any* system of multivariate polynomial equations), in practice they have a number of limitations. For example, algebraic geometric methods (including those based on elimination and Gröbner bases) usually explode in complexity, may introduce extraneous roots, and can only be applied to relatively simple systems of equations. Beyond this, they may require the solution of a high-degree polynomial, which may be a numerically ill-conditioned step in some cases. It must be noted, however, that despite these inconvenients recent work on sparse resultants is presently uncovering the further potential these methods possess [25]. On the other hand, as noted in [26], continuation techniques must be implemented in exact rational arithmetic to avoid numerical instabilities, leading to important memory requirements because large systems of complex initial value problems have to be solved. For an arbitrary problem, moreover, neither of these approaches is able to obtain the solution variety, if its dimension is greater than zero. Interval-based methods are also complete and general, but, although they can be slow in practice, they present a number of advantages that make them a competitive alternative: (1) contrarily to elimination methods, the equations are tackled in their input form, thus avoiding the need of intuition-guided symbolic reductions (2) they are numerically stable, (3) they also work if the dimension of the solution variety is greater than zero, (4) they deal with variable bounds in a natural way, and (5) they are simple to implement.

Two main classes of interval-based methods have been explored in the Robotics literature: those based on the interval version of the Newton method (also known as the Hansen algorithm) and those based on subdivision. To our knowledge, the first applications of the Hansen algorithm in this field were due to Rao et al. [27] and Didrit et al. [28], who respectively applied the interval Newton method to the inverse kinematics of 6R manipulators and the forward analysis of Stewart-Gough platforms. Rather than plunging into specific mechanisms, Castellet and Thomas then tackled general single-loop inverse kinematics problems [15], showing that the Hansen algorithm can be sped up if it is used in conjunction with other necessary conditions drawn from the problem itself. Later on, successful applications of the interval Newton method were also reported by Merlet in singularity analysis and mechanism design of parallel manipulators [29,30]. Subdivision techniques, in turn, were developed in the early nineties by Sherbrooke

and Patrikalakis in the context of constraint-based Computer Aided Design [26]. These exploit the subdivision property of Bernstein polynomials, which avoids the computation of derivatives while maintaining the quadratic convergence of the Hansen algorithm. Their application to general multi-loop mechanisms was made possible after explicit expressions for the control points of their closure equations were found in [17], allowing their rewriting in Bernstein form. A specific subdivision technique was then developed in [18], which leads to a remarkably simpler algorithm when the problem can be described only by multilinear constraints [19]. Given this simplicity, it seems logical to elucidate whether a formulation of every geometric constraint solving problem is possible in terms of such constraints exclusively. We have showed that the theory of Distance Geometry allows such a formulation, thus permitting a reasonably good symbiosis between algebrization and resolution, as initially sought [20]. This problem formulation and a novel subdivision-based constraint-solving technique for multilinear equations will be summarized next.

IV. METHODOLOGY

A. From kinematic to distance constraints

We propose the use of methods from Distance Geometry to tackle the problem at hand. A preliminary step to that end entails the translation of the original formulation, that specifies a number of kinematic constraints among a set of solids, to the canonical formulation in Distance Geometry, that specifies a number of distance constraints among a set of points. The latter setting asks, for a set of n points in 3-space, with specified distances between some of them, to compute (up to congruences) all spatial realizations of the point set that satisfy the prescribed distances. Such a point set with prescribed distances can be mechanically interpreted as a bar-and-joint framework, a structure of rigid bars (the known distances) joined at their ends with ideal universal joints (the points). Then, the goal can be rephrased as finding all possible configurations of this bar-and-joint framework, up to congruences.

The reduction of a set of kinematically constrained solids to an equivalent bar-and-joint framework is always possible. We give some examples of such reduction, and point the reader to [19,20] for further details:

- The direct kinematics of the Stewart-Gough platform in Figure 3a, that entails finding all valid postures of the platform that are compatible with the actuator lengths, is trivially reduced to that of find-

ing all valid configurations that the bar-and joint framework of Figure 3b can adopt.

- The inverse kinematics of the 6R manipulator of Figure 4a, that consists in finding which actuator angles take the end effector to the specified pose, can be reduced to computing all possible configurations of the bar-and-joint structure in Figure 4b.
- In Structural Biology, the computation of all conformations of a given molecule, like the cyclohexane in Figure 5a, can be reduced to finding the configurations of an equivalent robot (Figure 5b) or of its equivalent bar-and-joint framework (Figure 5c).

B. Cayley-Menger equations

Once the reduction to a bar-and-joint framework has been performed, equations can be drawn from the Theory of Distance Geometry to compute all valid configurations of such a framework. Distance Geometry refers to the analytical study of Euclidean Geometry in terms of some polynomials that are invariant under changes of the reference frame. These polynomials are expressed as Cayley-Menger determinants, which solely depend on the unknown distances of the problem at hand. The Cayley-Menger determinant of m points p_1, \dots, p_m is defined as the determinant of a square matrix G , with $m+1$ rows and columns, where the element $G(i,j)$ is the square of the distance between p_i and p_j , and the elements in the last row and column are all one, except $G(m+1, m+1)$, which is set to zero.

A fundamental theorem of Distance Geometry, then, establishes that a set of points with prescribed distances is realizable in 3-space if and only if all Cayley-Menger determinants on groups of five points vanish. In other words, the unknown distances may be computed by solving a system of polynomial equations made up with all Cayley-Menger determinants of five points, made equal to zero.

C. A branch-and-prune algorithm

Cayley-Menger equations happen to be multilinear. For this reason, we have devised an algorithm to solve them that takes advantage of this property. We just summarize it here and refer the reader to [19,20] for further details. The algorithm takes a rectangular box B of the search space as input, and delivers a set of small boxes as output, containing all solutions lying inside B .

Generally speaking, the algorithm isolates the solutions in B by iterating two operations, *box reduction* and *box bisection*, using the following branch-and-prune scheme. Using box reduction, portions of B containing no solution are cut off by narrowing some of its defining

intervals. This process is iterated until either the box is reduced to an empty set, in which case it contains no solution, or the box is "sufficiently" small, in which case it is considered a solution box, or the box cannot be "significantly" reduced, in which case it is split into two sub-boxes via box bisection. If the latter occurs, the whole process is repeated for the newly created sub-boxes, and for the sub-boxes recursively created thereafter, until one ends up with a collection of boxes whose size is under a specified size threshold.

If there are only a finite number of solution points in B , this algorithm returns a collection of small boxes containing them all. If, contrarily, the solution space is an algebraic variety of dimension one or higher, the algorithm returns a collection of small boxes discretizing the portions of this variety contained in B . In all cases, the algorithm is *complete*, in the sense that every solution point will be contained in one of the returned boxes.

V. EXPERIMENTS

The previous methodology has been implemented and successfully applied to the following problems, some of them unsolved up to this date. The first three arise in Robotics, the fourth appears in constraint-based CAD and the last one in Structural Biology. All execution times are given for a Pentium IV at 1.8 GHz, under LINUX.

A. Direct kinematics of a Stewart-Gough platform.

The problem is to find all poses that the platform of Figure 6a can adopt, given the lengths of its six actuators. Here, two Cayley-Menger equations can be set, involving the variables $r_{3,6}$ y $r_{1,4}$, where r_{ij} represents the square distance between the i -th and j -th points on such platform. The curves of these equations intersect at six points (Figure 6b). The algorithm computes the six solutions in 10 milliseconds. In [20], it is shown that the generic root finding methods of MAPLE, for example, based on the Newton-Raphson method, are unable to find two of these solutions, because they lie in a region where the jacobian of the equations almost vanishes.

B. Inverse kinematics of an overconstrained 6R robot

For a 6R kinematic loop (Figure 6c), when the rotation axes of each link are copunctual and form an equal angle in all links, the loop has two assembly modes: one is flexible (a one dimensional continuum of solutions) while the other is rigid (an isolated point in its configuration space).

This problem has been formulated as a system of six Cayley-Menger equations in six unknowns, and has

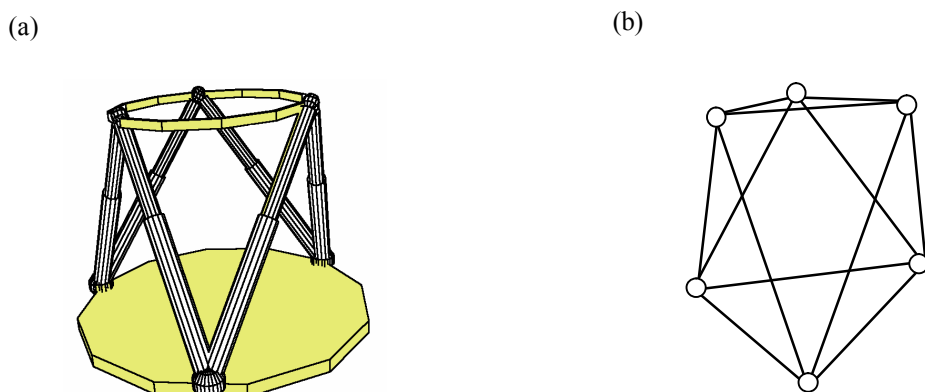


Fig. 3. An octahedral Stewart-Gough platform (a) and its associated bar-and-joint framework (b).

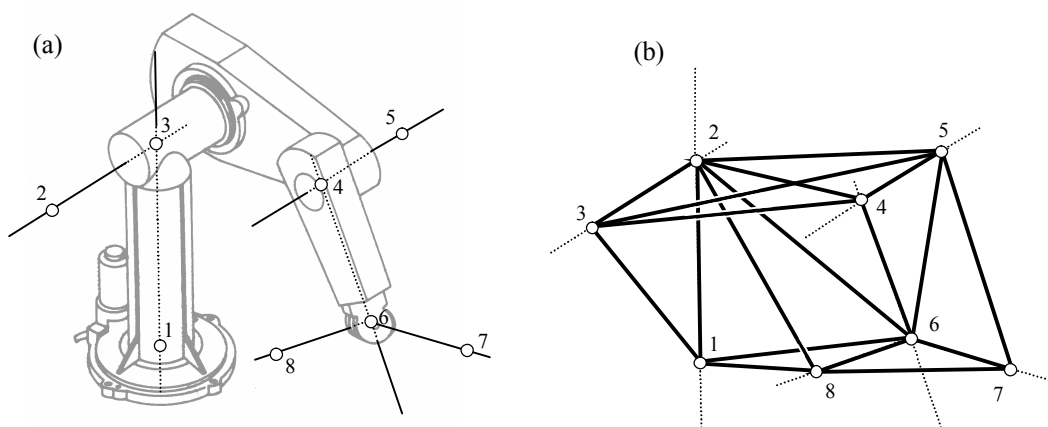


Fig. 4. A PUMA robot with six degrees of freedom (a) and its associated bar-and-joint framework (b).

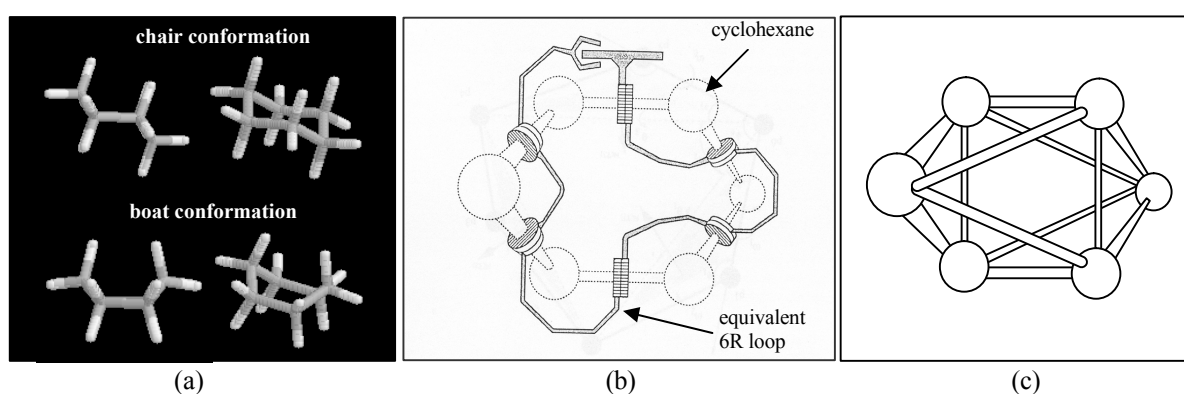


Fig. 5. The natural conformations of cyclohexane (a) can be obtained by solving the inverse kinematics of a 6R loop (b), mechanically equivalent to this molecule. The configurations of this loop correspond to the configurations of an associated bar-and-joint framework (c).

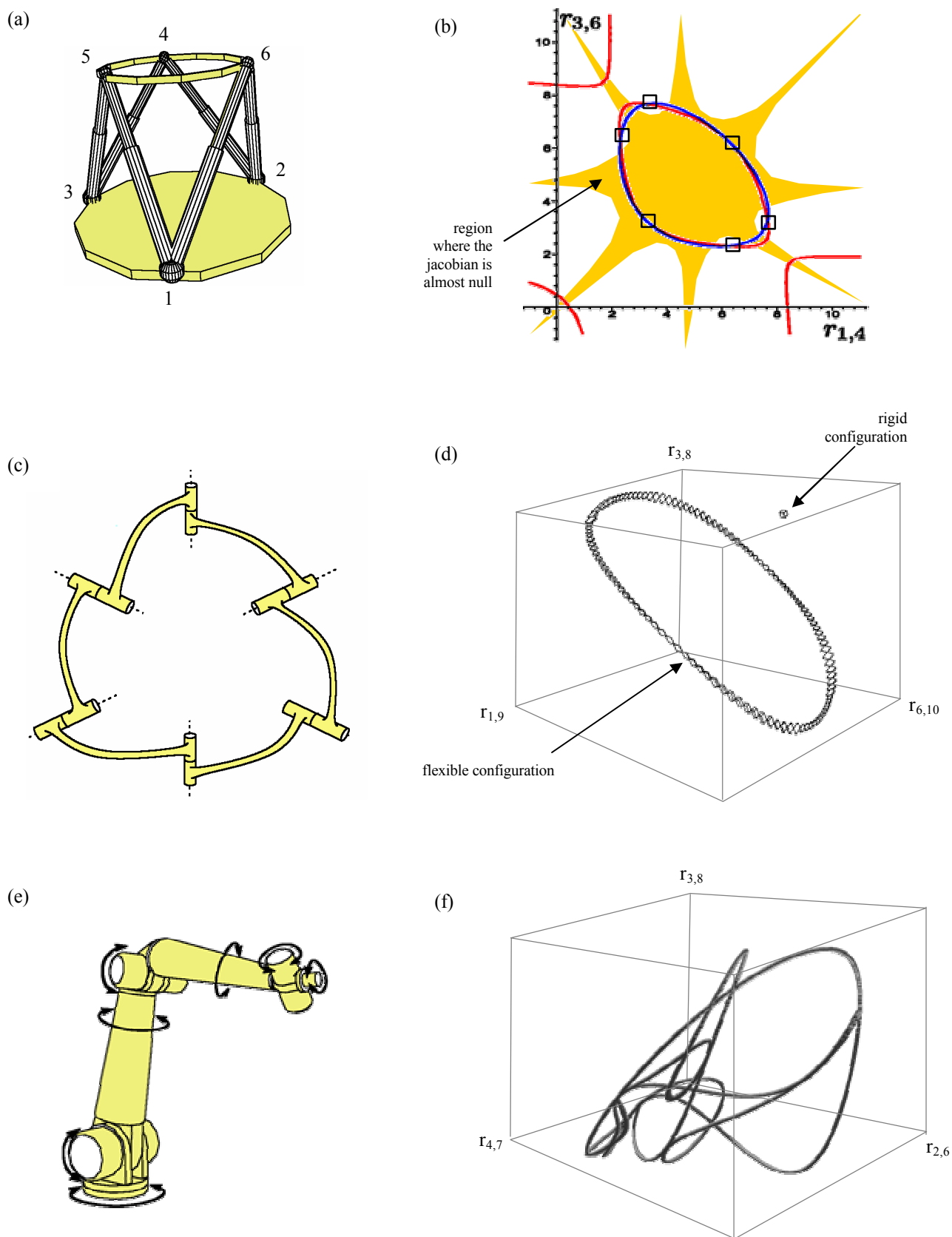


Fig. 6. Kinematic problems solved by the proposed method in the field of Robotics. **(a and b)** Direct kinematics of a Stewart-Gough platform. **(c and d)** Inverse kinematics of an overconstrained 6R robot. **(e and f)** Inverse kinematics of a 7R robot.

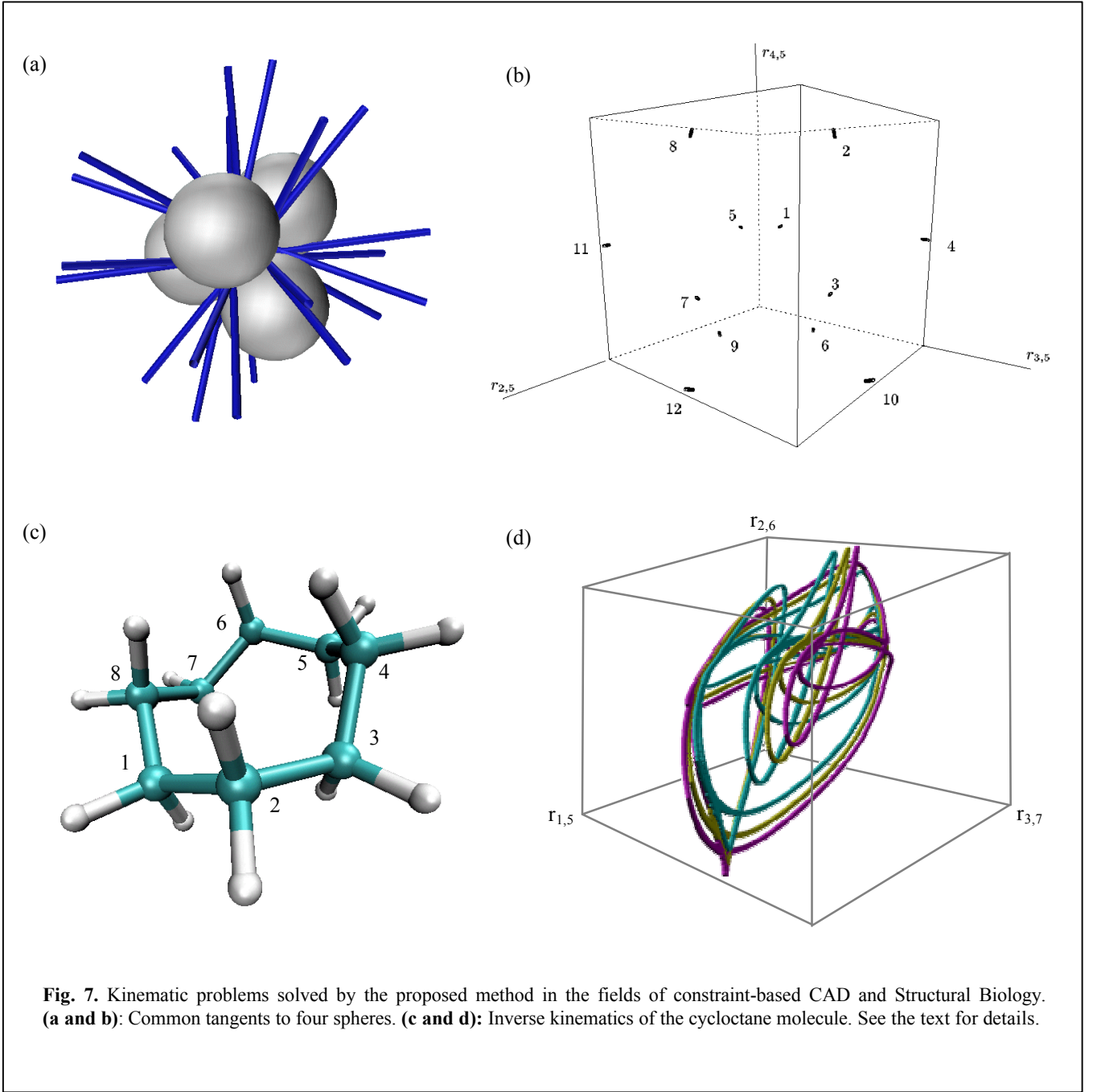


Fig. 7. Kinematic problems solved by the proposed method in the fields of constraint-based CAD and Structural Biology. **(a and b):** Common tangents to four spheres. **(c and d):** Inverse kinematics of the cyclooctane molecule. See the text for details.

been solved in 30 milliseconds. The returned solution boxes are shown in Figure 6d. It is assumed that Manocha's algorithm [31] is able to solve any 6R loop. However, it can be shown that such algorithm is unable to tackle this particular case, because it has infinite solutions.

C. Inverse kinematics of a 7R robot

The inverse kinematics of a robot with seven rotational degrees of freedom, like that of figure 6e, yields the one-dimensional continuum of possible configurations given in Figure 6f. The algorithm spends 14 minutes to solve the associated system of 11 equations in 11 variables. As

far as we know, this is the first time that such a solution space has been fully discretized.

D. All lines tangent to four spheres

Finding all lines simultaneously tangent to four spheres is a hard problem in Computational Geometry and, to our knowledge, an efficient solution to it was not given until 2001 [32]. The problem finds several applications in Computer Graphics and Geometric Modeling, including visibility computations with moving viewpoints, computing smallest enclosing cylinders of point sets and placement problems in CAD. Using the general algorithm we propose, the problem translates to

a system of 44 equations in 18 variables (it is thus redundant) and can be solved in 180 seconds, yielding the 12 solutions depicted in Figure 7a, corresponding to the 12 box clusters of Figure 7b.

E. Inverse kinematics of the cyclooctane molecule

This exemplifies a central problem in Structural Biology: to compute all configurations that a molecule can adopt (like that of the cyclooctane depicted in Figure 7c), knowing its topological structure, and the lengths of its covalent bonds. In Robotics, the problem is identical to that of computing all configurations of an 8R loop. Such a loop has two degrees of redundancy and, as a result, its solution space must be an algebraic variety of dimension 2. In Figure 7d we show several consecutive slices of this variety. A fine discretization of this solution space permits the validation of a long-standing conjecture in Chemistry: that the cyclooctane molecule can be assembled in two different modes, and that no path exists from one mode to the other, without breaking the carbon ring.

REFERENCES

- [1] Lozano-Pérez, "The design of a mechanical assembly system," Artificial Intelligence Laboratory, AI TR 397, MIT, 1976.
- [2] Popplestone, R.J., Ampler, A.P., y Bellos, I.M., "An interpreter for a language for describing assemblies", Artificial Intelligence, Vol. 14, No. 1, pp. 79-107, 1980.
- [3] Kramer, G.A., "Solving Geometric Constraint Systems". A Case Study in Kinematics, MIT Press, 1992.
- [4] Lieberman L.I., and Wesley, M.A., "AUTOPASS: an automatic programming system for computer controled mechanical assembly, IBM Journal of Research and Development, July 1997.
- [5] Torras, C., and Thomas, F., "Planning with constraints: applications to sensor-based robot assembly tasks", 1st. IFAC Symposium on Robot Control (SYROCO), Barcelona, November 1985.
- [6] Thomas, F., "Planificación de tareas de ensamblado basada en análisis de restricciones", Ph.D. dissertation. Technical University of Catalonia, July 1988.
- [7] Thomas, F., and Torras, C., "Group theoretic approach to the computation of symbolic part relations", IEEE Journal of Robotics and Automation, Vol. 4, No. 6, December 1988, pp. 622-634.
- [8] Celaya, E., "Geometric reasoning for the determination of the position of objects linked by spatial relationships". Ph.D. dissertation. Technical University of Catalonia, October 1992.
- [9] Celaya, E., and Torras, C., "Solving multi-loop linkages with limited-range joints", Mechanism and Machine Theory, Vol. 29, No. 3, pp. 373-391, April 1994. Pergamon Press.
- [10] Celaya, E., and Torras, C., "On Finding the Set of Inverse Kinematic Solutions for Redundant Manipulators", in Computational Kinematics, J. Angeles, G. Hommel and P. Kovács (editors), pp. 85-94, Kluwer Academic Publishers, 1993.
- [11] Celaya, E., "LMF: A program for positioning objects using geometric relationships", in Applications of Artificial Intelligence in Engineering VII, D.E. Grierson et al. (editors), pp. 873-881, Computational Mechanics Publications, Southampton, and Elsevier Applied Science, London, 1992.
- [12] Thomas, F., and Torras, C., "Inferring feasible assemblies from spatial constraints", IEEE Transactions on Robotics and Automation, Vol. 8, No. 2, April 1992, pp. 228-239.
- [13] Castellet, A., "An interval method for solving inverse kinematic problems". Ph.D. dissertation. Technical University of Catalonia, October 1998.
- [14] Castellet, A., and Thomas, F., "Using interval methods for solving inverse kinematic problems", Proc. of the NATO Advanced Study Institute on Computational Methods in Mechanisms, Vol. II, pp. 135-145, June 16-28, 1997, Varna, Bulgaria.
- [15] Castellet, A., and Thomas, F., "An algorithm for the solution of inverse kinematic problems based on an interval method", in Advances in Robot Kinematics, M. Husty and J. Lenarcic (editors), pp. 393-403, Kluwer Academic Publishers, 1998.
- [16] Bombín, C., Ros, L., and Thomas, F., "On the computation of the direct kinematics of parallel spherical mechanisms using Bernstein polynomials", Proc. of the IEEE International Conference on Robotics and Automation, pp. 3332-3337, May 21-26, 2001, Seoul, South Korea.
- [17] Bombín, C., Ros, L., and Thomas, F., "A concise Bézier clipping technique for solving inverse kinematics problems", in Advances in Robot Kinematics, J. Lenarcic y M.M. Stanisic (editors). Kluwer Academic Publishers, 2000.
- [18] Porta, J.M^a, Ros, L., Thomas, F., and Torras, C., "Solving multi-loop linkages by iterating 2D clippings", in Advances in Robot Kinematics, F. Thomas y J. Lenarcic (editors). Kluwer Academic Publishers, 2002.
- [19] Porta, J. M., Thomas F., Ros, L. and Torras, C., "A Branch-and-prune algorithm for solving systems of distance constraints". IEEE International Conference on Robotics and Automation, Vol. 1, pp. 342-348, September 2003.
- [20] Porta, J.M., Ros, L., Thomas, F. and Torras, C., "A branch-and-prune solver for distance constraints". Submitted.
- [21] Fudos, I., and Hoffmann, C., "A graph-constructive approach to solving systems of geometric constraints," ACM Transactions on Graphics, vol. 16, no. 2, pp. 179-216, 1997.
- [22] Nielsen, J., and Roth, B., "Formulation and solution for the direct and inverse kinematics problems for mechanisms and mechatronic systems," in Proceedings of the NATO Advanced Study Institute on Computational Methods in Mechanisms, J. Angeles and E. Zakhariyev, Eds., vol. I, June 1997, pp. 233-252.
- [23] Hoffmann, C. H., and Yuan, B., "On spatial constraints solving approaches," in Automated Deduction in Geometry: Third International Workshop, Lecture Notes in Computer Science, vol. 2061, 2000.
- [24] Emiris, I., and Mourrain, B., "Computer algebra methods for studying and computing molecular conformations," Algorithmica, no. 25, pp. 372-402, 1999.
- [25] Canny, J., and Emiris, I., "A subdivision-based algorithm for the sparse resultant," Journal of the ACM, vol. 47, no. 3, pp. 417-451, 2000.
- [26] Sherbrooke, E., and Patrikalakis, N., "Computation of the solutions of nonlinear polynomial systems," Computer Aided Geometric Design, vol. 10, no. 5, pp. 379-405, 1993.
- [27] [rao] Rao, R. S., Asaithambi, A., and Agrawal, S. K., "Inverse kinematic solution of robot manipulators using interval analysis," ASME Journal of Mechanical Design, vol. 120, pp. 147-150, 1998.
- [28] Didrit, O., Petitot, M., and Walter, E., "Guaranteed solution of direct kinematic problems for general configurations of parallel manipulators," IEEE Transactions on Robotics and Automation, vol. 14, no. 2, pp. 259-266, 1998.
- [29] Merlet, J.-P., "A formal numerical approach to determine the presence of singularity within the workspace of a parallel robot," in Proceedings of the 2nd Workshop on Computational Kinematics, Seoul, South Korea, May 2001, pp. 167-176.
- [30] Merlet, J.-P., "An improved design algorithm based on interval analysis for parallel manipulator with specified workspace," in Proc. of the IEEE Int. Conf. on Robotics and Automation, vol. 2, Seoul, South Korea, May 2001, pp. 1289-1294.
- [31] Manocha, D., y Canny, J.F., "Efficient kinematics for general 6R manipulators", IEEE Transactions on Robotics and Automation, Vol. 10, No. 5, pp. 648-657, 1994.
- [32] Macdonald, I., Pach, J., and Theobald, T., "Common tangents to four unit balls in R³", Discrete Computational Geometry, Vol. 26, No. 1, pp. 1-17, 2001.