

Dipartimento di Informatica e Sistemistica, Università di Napoli
Via Claudio 21, 80125 Napoli, Italy

INVERSE KINEMATICS AND RELATED ISSUES:
A UNIFIED APPROACH TO THEIR SOLUTION

ABSTRACT

The goal of this paper is to give a short tutorial presentation of a recently proposed approach to solving the inverse kinematic problem for general robotic manipulators (redundant and nonredundant). It is also shown that the resulting algorithm serves as a powerful tool to solving many of the issues related to the mapping between the joint space and the task space of a manipulator, such as occurrence of kinematic singularities and determination of the manipulator workspace.

1. INTRODUCTION

The solution to the inverse kinematic problem is of crucial importance for robotic manipulator control. A computational convergent algorithm has been recently proposed in the literature [1-3] which provides a systematic solution tool for general robot kinematic structures, without requiring the inversion of any kinematic function. Indeed the solution is based on the transpose of the Jacobian of the manipulator. One requirement is to express the orientation of the robot's end effector in terms of unit vectors, rather than orientation angles (Euler, RPY) [4]. Nevertheless, this issue turns out to be useful for geometrically task constrained motions such as sliding on a surface, inserting a peg in a hole, etc.

In this paper both nonredundant and redundant structures are considered. For the former ones it is shown how the algorithm can be conveniently partitioned in two stages, allowing a reduction in the computational burden [2,5,6]. For the latter ones it is shown how a set of constraints, in number of the degrees of redundancy, can be incorporated in the solution [3,7,8]. The sole requirement is that such constraints can

be functionally expressed in terms of the unknown joint variables with respect to the same base frame in which the robot's kinematics is expressed. Obstacle avoidance, existence of joint limits and manipulator dexterity all provide constraints that can be modelled as described above.

The occurrence of kinematic singularities is finally analyzed in details; the null space of the transpose of the Jacobian allows the identification of both boundary singularities (due to the boundaries of the workspace) and internal singularities (due to the alignment of certain axes) [9]. It is shown then how the same inverse kinematic solution algorithm can be used to draw the reachable workspace of a general manipulator, as well as how it can provide a solution for the joint variables even in the case of a trajectory that does not entirely lie in the workspace itself.

2. THE GENERAL SOLUTION ALGORITHM

The configuration $\underline{x} \in R^m$ of the manipulator's end effector in the task space can be described as a function of the joint variables $\underline{q} \in R^n$ (n DOF's) by the following nonlinear equation [10]:

$$\underline{x} = \underline{f}(\underline{q}) \quad (1)$$

The relationship between the joint velocities $\dot{\underline{q}}$ and the end effector velocities $\underline{v}$ is described by the following linear equation with the Jacobian matrix $J(\underline{q}) := \partial \underline{f} / \partial \underline{q}$ as the joint-dependent coefficient matrix [4]:

$$\underline{v} = J(\underline{q})\dot{\underline{q}} \quad (2)$$

There are two classical approaches in the literature to solving the inverse kinematics. One tries to solve equation (1) [4,9], the other attempts to solve equation (2) [11]. The first approach is based on the possibility of obtaining a closed-form analytical solution to (1); this is true for manipulators having a spherical wrist, or more generally

satisfying Pieper's sufficient condition [12]. The second approach relies upon the possibility of inverting (2) (the pseudoinverse of J is usually proposed for redundant manipulators); if $J(q)$ is not a full rank matrix, a kinematical singularity occurs and the solution is not feasible.

A rather different approach has been recently introduced by these authors [1-3] which reformulates the problem as a dynamical one. This leads to a general solution algorithm, whose nice feature is that it requires only the computation of direct kinematic functions, avoiding thus those numerical instabilities concerned with matrix inversion [11]. Consider the dynamic system of fig. 1; the error $\underline{e}$ in the task space can be made arbitrarily small by suitable increasing the feedback gains in the positive definite diagonal matrix K . The resulting update law is

$$\dot{\underline{q}} = J^T K \underline{e} \quad (3)$$

which provides also a solution for the joint velocities, usually required by the control servos in the joint space. Convergence can be shown by defining the Lyapunov function candidate $v = .5 \underline{e}^T \underline{e}$ and verifying that its time derivative $\dot{v}$ is negative definite outside a region in the error space containing the origin $\underline{e} = \underline{0}$ [1-3].

The task space vector $\underline{x}$ in (1) can be partitioned as $\underline{x}^T = (\underline{x}_p^T \ \underline{x}_h^T)$, where $\underline{x}_p$ determines the position of the end effector and $\underline{x}_h$ its orientation. Three position coordinates locate $\underline{x}_p$, whereas three orientation angles (Euler, RPY) are usually specified in $\underline{x}_h$ [4]. Correspondingly it is also $\underline{v}^T = (\underline{v}_p^T \ \underline{v}_h^T)$. Here it is important to remind that, while $\underline{v}_p$ is the actual linear velocity of the end effector $\dot{\underline{x}}_p$, $\underline{v}_h$ is not the true time derivative of $\underline{x}_h$. Instead, $\underline{v}_h$ is the angular velocity $\underline{\omega}$ of the end effector [9,11]. In terms of the above algorithm, this poses the requirement that the orientation of the end effector $\underline{x}_h$ must be described by the unit vectors $\underline{s}$, $\underline{a}$ ($\underline{n}$ is redundant, since $\underline{n} = \underline{s} \times \underline{a}$). In practice, geometrically task constrained orientations of the end effector are naturally specified in terms of $\underline{s}$ and $\underline{a}$ (sliding on a surface, inserting a peg in a hole, etc.), whereas arbitrary motions are more conveniently imposed to the orientation angles..

3. NONREDUNDANT STRUCTURES

With reference to (1), a manipulator is termed nonredundant if $n = m$. For instance, six DOF's are needed to uniquely locate the position and orientation of the end effector.

The general solution algorithm of fig. 1 can be partitioned into two stages, depending upon the configuration of the last axes of motion, generally revolute axes [2,3]. The end effector position, indeed, can be assumed at any point along the structure which is known as a function of the assigned task variables. Obviously, in case that position is located through more than three joint variables, geometrical constraints have to be introduced in order to obtain a unique solution.

In case of a spherical wrist (fig. 2a), for instance, the position is naturally chosen at the intersection of the last three revolute axes and $\underline{x}_p$ will depend only on the first three joint variables (1st stage); the remaining joint variables will be determined through $\underline{x}_h$ (2nd stage) [1].

If the axes intersect two-by-two (fig. 2b), four DOF's will determine the end effector position. Therefore, a constraint on the orientation of the fourth link must be introduced in the 1st stage in order to uniquely locate the point. The last two joint variables will be determined in the 2nd stage [5,6].

Finally, if the axes do not converge at all (fig. 2c), two constraints have to be introduced to uniquely position the point; in this case $\underline{a}$ has to be completely oriented. The outer joint variable will just orient $\underline{s}$ [2,3].

4. REDUNDANT STRUCTURES

If the number of DOF's n exceeds the number of task space variables, the manipulator is termed kinematically redundant relatively to that task. The space of redundant solutions can be conveniently exploited to obtain a more versatile manipulator in terms of its kinematical configuration and its interaction with the environment. In particular, redundancy can be used to meet constraints on joint range availability and/or

to obtain trajectories in the joint space which are collision-free in presence of obstacles along the motion. Another important use of redundancy is in keeping the manipulator in a configuration which makes it as dexterous as possible, that is also avoiding kinematic singularities.

With reference to the general inverse kinematic solution algorithm of fig. 1, a set of constraints can be systematically introduced in the definition of the task variables, up to eventually cover the number of degrees of redundancy ($n - m$). Such constraints, indeed, have to be functionally expressed in terms of the unknown joint variables with respect to the same base frame in which the manipulator kinematics is given. The result is an augmented direct kinematic equation of the type [3]

$$\underline{y} = \underline{f}'(\underline{q}) \quad (4)$$

where $\underline{y} \in \mathbb{R}^{m+v}$, with $0 \leq v \leq n-m$. In terms of the solution algorithm, an augmented Jacobian matrix J_e will be obtained, whose first m rows will be the same as in J and whose last v rows will be obtained by properly differentiating with respect to $\underline{q}$ the extra task variables introduced in $\underline{f}'$.

In case of a constraint on a joint limit ($q_{i\min} \leq q_i \leq q_{i\max}$), the proper variable to introduce in the task space will be simply the distance of the actual q_i from either limit, $d_q = q_i - q_{i\min}$ or $d_q = q_{i\max} - q_i$, depending on which limit is involved. The resulting row in the Jacobian J_e will have a ± 1 in correspondence of the i -th column (all the other elements will be zero), so as to prevent the joint variable to approach the limit [7,13].

In case of an obstacle in the manipulator workspace, a point at minimum distance along the kinematic structure from the obstacle can be determined (obstacle avoidance point) [7,14] (fig. 3). The task space variable to introduce will be the distance of the obstacle avoidance point from the obstacle; in fact, such distance can be shown to be a function of those joint variables which determine the position of the obstacle avoidance point. The Jacobian of the obstacle avoidance point will obviously contribute to form the additional row of the augmented Jacobian J_e .

If the manipulator is required to be dexterous, one of the

manipulability measures proposed in the literature can be directly adopted as the additional task space variable [8]. The determinant of the square root of JJ^T [15] is perhaps the simplest measure for detecting a kinematic singularity, whenever it approaches zero. Other measures can be thought of, depending on the particular kinematic structure and on the necessity of extracting one particular singularity out of different ones (the shoulder singularity, the wrist singularity, etc.).

A very important point in applying the general solution algorithm of fig. 1 to constrained redundant manipulators is the logical activation + inactivation of constraints. It is possible, indeed, to wind up in a "deadlocked" situation where further movement is not possible (overconstrained system) and the algorithm has to be aborted [7].

5. SOME REMARKS CONCERNING SINGULARITIES: DETERMINATION OF THE WORKSPACE

A solution of the type (3) recalls the fundamental relationship between the joint torques $\underline{\tau} \in \mathbb{R}^n$ and the task (Cartesian) forces $\underline{f} \in \mathbb{R}^m$ applied at the end effector [9], i.e.

$$\underline{\tau} = J^T(\underline{q})\underline{f} \quad (5)$$

It can be recognized that applying the solution (3) is equivalent to regard the manipulator as a set of weightless links with unity damping at each joint; K_e play the role of elastic forces.

It has been anticipated that when $\text{rank}(J) < m$, a singularity occurs. As far as the update law (3) is concerned, it results $J^T \underline{e} = \underline{0}$ when $\underline{e}$ belongs to the null space of J^T . On the basis of the above equivalence, however, the null space of J^T spans those directions along which a Cartesian force is completely neutralized by the mechanical structure of the manipulator. From the implementation viewpoint, such configuration is detected by having $\|\dot{\underline{q}}\| = 0$ and $\|\underline{e}\| \neq 0$.

The direction of the null space of the transpose of J^T identifies not only the so-called "internal" singularities (inside the reachable workspace) due to the alignment of certain axes, but also the so-called "boundary" singularities (at the boundaries of the workspace) at which

the manipulator is fully extended [9].

On the basis of equation (5), it is seen that, under the action of a force imagined applied to the end effector, the manipulator reaches a (stable) static equilibrium configuration in which it assumes maximum extension in the direction of the force [16]. If a point $\hat{x}$ out of the workspace is assigned, the algorithm of fig. 1 will take the actual x to a boundary singularity. Therefore, if a trajectory is planned which contains points outside the workspace, the algorithm will not fail, but it will automatically generate a solution which proceeds along the boundaries of the workspace.

From the analysis of the singularities performed above, it becomes quite straightforward to adopt the same algorithm to draw the boundaries of the workspace as well as those lines inside the workspace which limit the regions with different aspects [17]. This will be explained in the following.

The reachable workspace of a manipulator is the volume in the Cartesian space defined as the geometrical locus of the end effector point p , when the joint variables range between the two extreme values [17]:

$$p = p(q) \quad q_{imin} \leq q_i \leq q_{imax} \quad (6)$$

The volume is finite, closed, connected ($p(q)$ is continuous) and, therefore, entirely defined by its boundary surface. This surface is made up of planar, spherical, toroidal and cylindrical surface elements.

A rank deficiency in the Jacobian matrix J of the manipulator is caused by any of the following situations, whenever a loss of mobility occurs:

- a) a null column vector (the corresponding joint is blocked at either of the two limits, q_{imin} or q_{imax} : $\partial p / \partial q_i = 0$),
- b) linear dependence of the column vectors (the manipulator is in a singular configuration, usually two or more axes are aligned: $\partial p / \partial q_i \times \partial p / \partial q_j = 0$).

All the possible combinations of two or more of the above situations are to be considered to derive the surfaces characterizing the workspace and its significant zones. In particular, the colinearity of the column vectors in the Jacobian is of primary importance to determine the aspects of the manipulator [17]. For a two DOF planar mechanism, for instance, a single point can be accessed by $\underline{p}$ for two different aspects, "elbow-down" and "elbow-up". For a redundant mechanism (three DOF's in a plane), instead, there will be two families of aspects rather than two single aspects.

Based upon the above premises, the boundaries of the workspace and the curves limiting those regions with different aspects can be directly determined by applying the inverse kinematic algorithm of fig. 1; the sole requirement is to choose the reference trajectory $\hat{\underline{p}}$ according to the possible combinations of the above two situations a) and b). A simple example with a two-DOF planar mechanism will illustrate the procedure.

The trajectory $\hat{\underline{p}}(t)$ is obviously chosen as a circle of radius $l > l_1 + l_2$, being l_1 and l_2 the lengths of the two links. In order to start the algorithm, assume to locate the arm at the configuration $q_1 = q_{1min}$, $q_2 = q_{2min}$ (point A of fig. 4). The corresponding $\hat{\underline{p}}(t=0)$ is chosen at an angle $\alpha(t=0) = \text{atan2}(p_x(t=0), p_y(t=0))$ with the X_0 axis.

The logical sequence to account for the joint limits (null column vectors in J) is as follows. The stop on q_2 is released ($\partial \underline{p} / \partial q_1 = \underline{0}$); the tip of the arm $\underline{p}(t)$ draws the arc AB, as $\alpha(t)$ increases. At point B, in fact, the limit q_{2max} is encountered. Then the stop on q_1 is released ($\partial \underline{p} / \partial q_2 = \underline{0}$); $\underline{p}(t)$ draws the arc BC, since at C the limit q_{1max} is reached. Then the stop on q_2 is released again ($\partial \underline{p} / \partial q_1 = \underline{0}$); the corresponding arc drawn is CD. The stop on q_1 is finally released again ($\partial \underline{p} / \partial q_2 = \underline{0}$), drawing the arc DA.

It is necessary now to account for the colinearity between the two columns of J . Restarting from point A, both joint stops are released; the tip of the arm will initially redraw AB, up to that point E, corresponding to $q_1 = q_{1min}$, $q_2 = 0$ ($\partial \underline{p} / \partial q_1 \times \partial \underline{p} / \partial q_2 = \underline{0}$). From there on it will proceed along a new arc, up to encounter the point F, corresponding to $q_1 = q_{1max}$, $q_2 = 0$.

At this point the required workspace is completely determined by the

area limited by the arcs AEFCGDA, where G is the intersection between arcs BC and CD. Moreover, it can be recognized that the area limited by AEFDA corresponds to the first aspect "elbow-down" attainable by the arm, whereas the area limited by BCFEB characterizes the second aspect "elbow-up" of the arm. Therefore, the area EBGFE is the dexterous region of the reachable workspace [9] where the arm can assume both aspects, up and down (fig. 4).

6. CONCLUSIONS

This paper has presented a short tutorial of a new algorithm to solve the inverse kinematic problem for a general (nonredundant and redundant, constrained or unconstrained) manipulator as well as to determine its workspace. The real breakthrough of the proposed solution is that it is based only on direct kinematic functions computation. A variety of applications to different structure geometries can be found in the references by these authors. Here a novel, simple application to the workspace determination of a two-DOF planar arm has been offered.

REFERENCES

- [1] A. Balestrino, G. De Maria, L. Sciavicco and B. Siciliano, "A Novel Approach to Coordinate Transformation for Robotic Manipulators," Università di Napoli, Int. Rep., 1984.
- [2] G. De Maria, L. Sciavicco and B. Siciliano, "A General Solution Algorithm to Coordinate Transformation for Robotic Manipulators," Proc. '85 Int. Conf. Advanced Robotics, Tokyo, Japan, Sept. 1985.
- [3] L. Sciavicco and B. Siciliano, "Solving the Inverse Kinematic Problem for Robotic Manipulators," Prep. 6th CISM-IFTOMM Ro.Man.Sy, Cracow, Poland, Sept. 1986.
- [4] M. Vukobratovic and M. Kircanski, Kinematics and Trajectory Synthesis of Manipulation Robots: Scientific Fundamentals of Robotics 3, Springer-Verlag, Berlin, 1986.
- [5] L. Sciavicco and B. Siciliano, "Coordinate Transformation: A Solution Algorithm for One Class of Robots," IEEE Trans. Syst., Man, Cybern., vol. SMC-16, no. 4, pp. 550-559, July/Aug. 1986.

- [6] L. Sciavicco and B. Siciliano, "An Inverse Kinematic Solution Algorithm for Robots with Two-by-Two Intersecting Axes at the End Effector," Proc. IEEE Int. Conf. Robotics, Automation, San Francisco, CA, Apr. 1986.
- [7] L. Sciavicco and B. Siciliano, "A Dynamic Solution to the Inverse Kinematic Problem for Redundant Manipulators," Proc. IEEE Int. Conf. Robotics, Automation, Raleigh, NC, Mar./Apr. 1987.
- [8] L. Sciavicco and B. Siciliano, "An Inverse Kinematic Solution Algorithm for Dexterous Redundant Manipulators," '87 Int. Conf. Advanced Robotics, Versailles, France, Oct. 1987, subm. for pres.
- [9] J.J. Craig, Introduction to Robotics: Mechanics and Control, Addison-Wesley, Reading, MA, 1986.
- [10] J. Denavit and R.S. Hartenberg, "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices," ASME J. Appl. Mech., vol. 22, pp. 215-221, June 1955.
- [11] D.E. Whitney, "The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators," Trans. ASME J. Dyn., Syst., Meas., Contr., vol. 94, pp. 303-309, Dec. 1972.
- [12] D. Pieper, The Kinematics of Manipulators Under Computer Control, Ph.D. Thesis, Stanford University, 1968.
- [13] M. Vukobratovic and M. Kircanski, "A Dynamic Approach to Nominal Trajectory Synthesis for Redundant Manipulators," IEEE Trans. Syst., Man, Cybern., vol. SMC-14, no. 4, pp. 580-586, July/Aug. 1984.
- [14] M. Kircanski and M. Vukobratovic, "Trajectory Planning for Redundant Manipulators in the Presence of Obstacles," Proc. 5th CISM-IFTOMM Ro.Man.Sy, Udine, Italy, June 1984.
- [15] T. Yoshikawa, "Analysis and Control of Robot Manipulators with Redundancy," Proc. 1st Int. Symp. Rob. Res., Bretton Woods, NH, Aug./Sept. 1983.
- [16] A. Kumar and K.J. Waldron, "The Workspaces of a Mechanical Manipulator," ASME J. Mech. Des., vol. 103, no. 3, pp. 665-672, July 1981.
- [17] A. Liégeois, Performance and Computer Aided Design: Robot Technology 7, Kogan Page, London, UK, 1985.

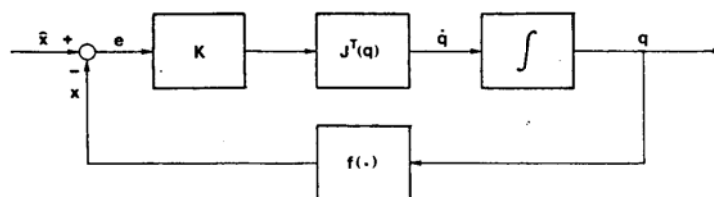


Fig. 1. The General Inverse Kinematic Solution Algorithm.

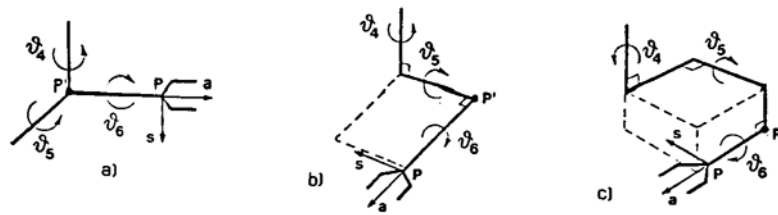


Fig. 2. Three Basic Kinematic Configurations at the End Effector:
a) Spherical Wrist, b) 2-by-2 Intersecting Axes, c) Nonconverging Axes.

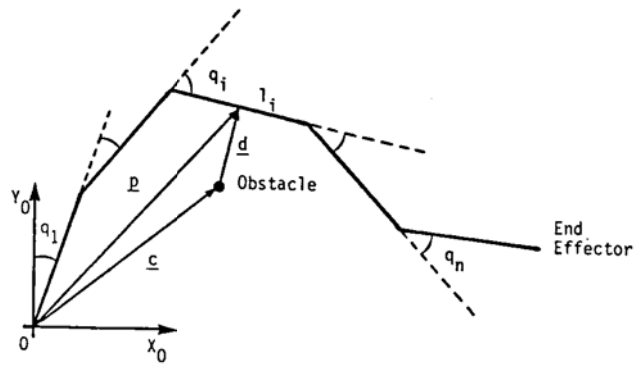


Fig. 3. Geometry of a Planar Manipulator Showing the Point Nearest to the Obstacle.

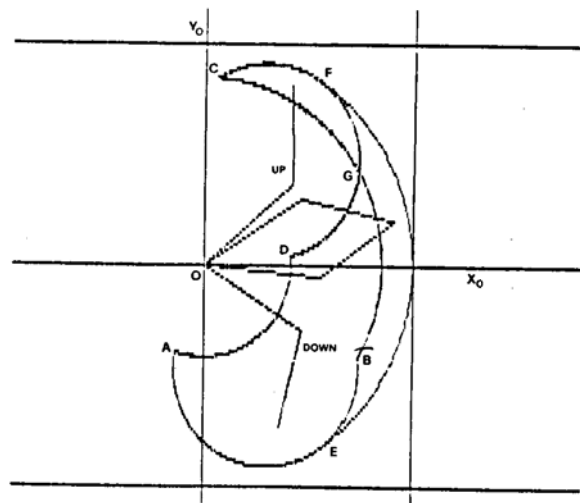


Fig. 4. Workspace of a 2-DOF Planar Arm.