Closed-Loop Inverse Kinematics Schemes for Constrained Redundant Manipulators with Task Space Augmentation and Task Priority Strategy

Pasquale Chiacchio, Stefano Chiaverini, Lorenzo Sciavicco and Bruno Siciliano

The International Journal of Robotics Research 1991; 10; 410
DOI: 10.1177/027836499101000409

Pasquale Chiacchio
Stefano Chiaverini
Lorenzo Sciavicco
Bruno Siciliano
Dipartimento di Informatica e Sistemistica
Università degli Studî di Napoli "Federico II"
Via Claudio 21, 80125 Napoli, Italy

Closed-Loop Inverse Kinematics Schemes for Constrained Redundant Manipulators with Task Space Augmentation and Task Priority Strategy

Abstract

This article presents new closed-loop schemes for solving the inverse kinematics of constrained redundant manipulators. In order to exploit the space of redundancy, the end-effector task is suitably augmented by adding a constraint task. The success of the technique is guaranteed either by specifying the constraint task ad hoc *or by resorting to a task priority strategy. Instead of previous inverse kinematics schemes that use the Jacobian pseudoinverse, the schemes in this work are shown to converge using the Jacobian transpose. A number of case studies illustrate different ways of solving redundancy in the context of the proposed schemes.*

1. Introduction

Realization of more versatile robot manipulators requires the creative adoption of redundancies in kinematic structure, sensing, control and information handling. Such features are ingeniously integrated in the human arm, which apparently constitutes the ultimate model of a dexterous robot. Despite the potential advantages attainable with redundancy, there appears to be a great reluctance in industry to produce redundant robot systems. The reason perhaps is that redundancy involves mechanical and control complexity and thus increased costs.

This work focuses its attention on kinematic redundancy, which occurs when a manipulator possesses more than the required degrees of freedom to execute a given task. In this case, the inverse kinematic problem admits infinite solutions, and a criterion to select one of them is needed. The most widely adopted method for solving kinematic redundancy is based on the local solution to the differential kinematic equation using the *pseudoinverse* of the Jacobian (Whitney 1969). One shortcoming of this method is that closed task space trajectories will not generally produce closed joint space trajectories (Klein and Huang 1983). Also, despite local minimization of joint velocities, singular configurations are not avoided in any practical sense (Baillieul et al. 1984).

Because the space of redundancy allows for multiple kinematic solutions, greater flexibility in manipulator motion is expected. Indeed, the redundant degrees of freedom can be conveniently exploited to meet a number of additional constraints on the solution of the inverse kinematic problem. Most approaches in the literature solve redundancy in terms of optimizing quadratic type criteria. The pseudoinverse solution is modified by the addition of a term in the space of redundancy—the null space of the Jacobian matrix—which is used for local optimization purposes (Liégeois 1977). Several kinds of constraints have been imposed according to this technique; obstacle avoidance (Kirćanski and Vukobratović 1984), minimization of joint torques (Hollerbach and Suh 1985), maximization of manipulability measures (Yoshikawa 1985a,b), dexterity measures (Klein and Blaho 1987), and task compatibility indices (Chiu 1987) are some possible constraints.

A conceptually different approach to solving redundancy, proposed by Baillieul (1985) and later by Chang (1987), imposes a constraint task to be fulfilled along with the original end-effector task, namely a *task space augmentation*. This leads to algorithms based on the computation of an *extended* Jacobian that have a major advantage over pseu-

The International Journal of Robotics Research,
Vol. 10, No. 4, August 1991,

doinverse techniques in that they are *locally cyclic* (Baker and Wampler 1988) (i.e., closed paths in a simply connected task space give rise to closed paths in the joint space). The disadvantage of this approach is that while the end-effector Jacobian may be singularity free, there will be no guarantee that the extended Jacobian matrix is full rank. Baillieul (1986) pointed out that *algorithmic singularities* may arise from the way in which the constraint task conflicts with the end-effector task. A tool for finding such singularities has lately been proposed (Baillieul 1987).

A crucial point for task space augmentation then remains the specification of suitable constraint tasks. The simplest way is to construct an ad hoc constraint function for each assigned end-effector task such that the occurrence of algorithmic singularities is excluded in advance. This approach is seen to be conceptually equivalent to seeking for inverse kinematic functions on the augmented task space (Wampler 1987).

A more general approach is offered by the *task priority strategy* (Maciejewski and Klein 1985; Nakamura et al. 1987), which solves the conflicting task situations by suitably assigning an order of priority to the given tasks. This is accomplished by satisfying the lower priority task only in the null space of the higher priority task. Therefore the end-effector task will usually be advocated as the primary task and the constraint task as the secondary task, although the order of priority could be inverted if required by the particular application.

All the above techniques are based on the pseudoinverse solution that yields the joint velocities satisfying given end-effector and constraint tasks. Therefore in order to obtain operative inverse kinematic schemes, the above joint velocities must be integrated over time to generate the joint displacements in an *open-loop* fashion.

To overcome this drawback, computationally efficient solution algorithms can be derived from *closed-loop* inverse kinematic (CLIK) schemes based on the computation of either the Jacobian transpose (Balestrino et al. 1984; Wolovich and Elliott 1984; Slotine and Yoerger 1987) or the Jacobian pseudoinverse (Balestrino et al. 1984; Sciavicco and Siciliano 1987a; Tasia and Orin 1987). These schemes solve the inverse kinematic problem by reformulating it in terms of the convergence for an equivalent feedback control system.[1]

1. This approach must not be confused with all those kinematic control techniques that embed the actual manipulator in the loop (e.g., Takegaki and Arimoto 1981).

The CLIK schemes have been applied using the task space augmentation to the case of obstacle and/or joint limit avoidance constraints (Sciavicco and Siciliano 1987a) and to the case of dexterity constraints (Sciavicco and Siciliano 1987b). Later, the problem of occurrence of algorithmic singularities for such schemes was addressed by Sciavicco et al. (1988) and by Sciavicco and Siciliano (1988a).

This article is aimed at presenting new effective CLIK schemes for constrained redundant manipulators that overcome the problem of augmented Jacobian rank deficiencies. If a suitable specification of the constraint task is not allowed, a task priority strategy is adopted. Differently from previous schemes in the literature (e.g., Nakamura et al. 1987), however, the schemes here proposed avoid the use of the pseudoinverse of the constraint Jacobian. The convergence is proved by means of Lyapunov stability theory. Furthermore, the conflicting task situations are characterized in terms of structural properties of the end-effector and constraint Jacobians.

Four different case studies for planar manipulators are worked out to illustrate the implications of task space augmentation and task priority strategy on the CLIK schemes with various types of constraint tasks: manipulating in a cluttered environment, tracking with given orientation, tracking with prescribed dexterity, and avoiding an obstacle.

2. Background

The direct kinematic equation describes the mapping of the $(n \times 1)$ joint vector $\mathbf{q}$ into the $(m \times 1)$ end-effector task vector $\mathbf{x}_E$ as

$$\mathbf{x}_E = \mathbf{f}_E(\mathbf{q}), \tag{1}$$

where $\mathbf{f}_E$ is a continuous nonlinear vector function whose structure and parameters are known for each given manipulator.

The kinematic equation (1) can be differentiated with respect to time, yielding the mapping between the joint velocity vector $\dot{\mathbf{q}}$ and the end-effector task velocity vector $\dot{\mathbf{x}}_E$, through the $(m \times n)$ Jacobian matrix $\mathbf{J}_E(\mathbf{q}) = \partial \mathbf{f}_E / \partial \mathbf{q}$; i.e.,

$$\dot{\mathbf{x}}_E = \mathbf{J}_E(\mathbf{q})\dot{\mathbf{q}}, \tag{2}$$

where the time dependence has been suppressed for notation compactness.

If the manipulator is kinematically redundant with respect to a given task, it is $m < n$. Assuming that the Jacobian matrix $\mathbf{J}_E(\mathbf{q})$ has full rank for almost all $\mathbf{q}$s, $(n - m)$ additional degrees of freedom are available for solving redundancy. If for some $\mathbf{q}^*$, $\mathbf{J}_E$ has

rank less than m, the manipulator is said to be at a singular configuration.

Almost all the approaches for solving redundancy in the literature are based on the mapping (2), in that a solution in terms of the joint velocities is sought. In his pioneering work on resolved rate control, Whitney (1969) proposed to use the Moore-Penrose pseudoinverse of the Jacobian matrix to find the minimum norm solution to (2), i.e.,

$$\dot{\mathbf{q}} = \mathbf{J}_E^{\dagger}(\mathbf{q})\dot{\mathbf{x}}_E, \tag{3}$$

where $\mathbf{J}_E^{\dagger}$ is the $(n \times m)$ matrix defined as $\mathbf{J}_E^{\dagger} = \mathbf{J}_E^T(\mathbf{J}_E\mathbf{J}_E^T)^{-1}$. However, Baillieul, Hollerbach and Brockett (1984) proved that, even though the joint velocities are instantaneously minimized, there is no guarantee at all that kinematic singularities are avoided. By the way, the use of a singularity-robust Jacobian was independently proposed by Nakamura and Hanafusa (1986) and by Wampler (1986) based on damped least-squares methods. Another pitfall of solution (3) is that repeatability of joint trajectories for repeated end-effector trajectories is not preserved (Klein and Huang 1983); this is not desirable in most practical robot applications. In terms of the formulation established by Baker and Wampler (1988), it can be said that solution (3) does not in general possess the so-called *cyclic property*.

Redundancy can be conveniently exploited to meet additional constraints on the inverse kinematic problem in order to obtain greater manipulability in terms of manipulator configurations and interaction with the environment. If the robot is required to move in a cluttered environment, for instance, avoidance of obstacles (Kirćanski and Vukobratović 1984; Maciejewski and Klein 1987) and mechanical joint limits (Liégeois 1977) is usually desired. In other applications it could be of interest to minimize the joint torques along a given path (Hollerbach and Suh 1985).

The other important point in purposely adopting redundancy is the avoidance of kinematic singularities. The manipulability measure introduced by Yoshikawa (1985b) and more generally the dexterity measures surveyed by Klein and Blaho (1987), such as matrix condition number and minimum singular value, all based on the norm of the matrix $\mathbf{J}_E\mathbf{J}_E^T$, represent indices of the ability of a manipulator to arbitrarily position and orient its end effector. The dynamic manipulability measure (Yoshikawa 1985a), instead, takes the arm dynamics into consideration. Related to these measures is also the concept of task compatibility (Chiu 1987), according to which the matrix $\mathbf{J}_E\mathbf{J}_E^T$ is utilized to determine quantitative indices of the ability to perform an exertion/control task along a given direction of the task space.

Revisiting the pseudoinverse minimum norm solution (3), it can be shown that the general solution to eq. (2) is given by

$$\dot{\mathbf{q}} = \mathbf{J}_E^{\dagger}(\mathbf{q})\dot{\mathbf{x}}_{Ed} + [\mathbf{I} - \mathbf{J}_E^{\dagger}(\mathbf{q})\mathbf{J}_E(\mathbf{q})]\dot{\mathbf{q}}_0, \tag{4}$$

where $\mathbf{I}$ is the $(n \times n)$ identity matrix and $\dot{\mathbf{q}}_0$ is an $(n \times 1)$ arbitrary joint velocity vector. The solution (3) has then been modified by the addition of the homogeneous term created by the projection operator $(\mathbf{I} - \mathbf{J}_E^{\dagger}\mathbf{J}_E)$, which selects the components of $\dot{\mathbf{q}}_0$ in the null space of the mapping $\mathbf{J}_E$; thus $\dot{\mathbf{q}}_0$ produces no motion at the end effector.

The most widely adopted approach in the literature is to solve redundancy by choosing the vector $\dot{\mathbf{q}}_0$ as the gradient of scalar quadratic functions that suitably describe the above-mentioned constraints. In this way local minima of the selected function are obtained, which may lead to difficulties in some cases.

A different approach to solving the inverse kinematic problem via a suitable *dynamic* reformulation of the problem was independently introduced by Balestrino et al. (1984) and by Wolovich and Elliott (1984) for the nonredundant case ($m = n$). The method is summarized in the following for *unconstrained* redundant manipulators ($m < n$).

Let $\mathbf{x}_{Ed}(t)$ be a desired end-effector trajectory. Aim of a computational inverse kinematic algorithm is to find a joint trajectory solution $\mathbf{q}(t)$ such that $\mathbf{f}_E(\mathbf{q}(t)) = \mathbf{x}_E(t)$ is as close as possible to $\mathbf{x}_{Ed}(t)$. Defining the tracking error

$$\mathbf{e}_E = \mathbf{x}_{Ed} - \mathbf{x}_E \tag{5}$$

allows construction of a suitable feedback control system, whose purpose is to drive $\mathbf{e}_E$ to zero. This reformulation leads to establishing a dynamic solution to the inverse kinematics along a trajectory. In this respect, assuming that $\text{rank}(\mathbf{J}_E) = m$, it can be proved via Lyapunov stability theory that the "control law"

$$\dot{\mathbf{q}} = \mathbf{J}_E^T(\mathbf{q})\mathbf{K}_E\mathbf{e}_E \tag{6}$$

ensures that the end-effector tracking error $\mathbf{e}_E$ is ultimately bounded into an attractive ball containing the origin; the radius of the ball can be made arbitrarily small by suitably increasing the minimum eigenvalue of the positive definite matrix $\mathbf{K}_E$.

It can be recognized that applying the CLIK scheme in Figure 1 along a trajectory assigned at the end effector is equivalent to regard the vector $\mathbf{K}_E\mathbf{e}_E$ as the elastic force that has to be applied at the end

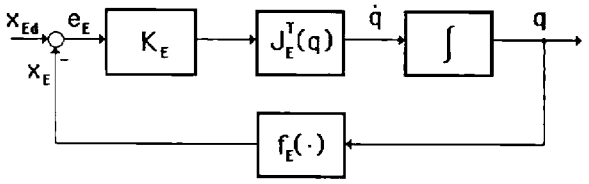

Fig. 1. The CLIK Jacobian transpose scheme for end-effector task.

effector of a virtual manipulator, with null mass and unit viscous damping, in order to track the desired trajectory.

The following remarks on solution (6) are in order.

Remark 1. If $\mathbf{e}_E$ at time $t = 0$ is null (i.e., the initial configuration of the manipulator is known), the tracking error remains into the above attractive ball for all trajectories $\dot{\mathbf{x}}_{Ed} \in \mathcal{R}(\mathbf{J}_E)$, where $\mathcal{R}(\mathbf{J}_E)$ denotes the range space of matrix $\mathbf{J}_E$.

Remark 2. The steady-state error (i.e., when $\dot{\mathbf{x}}_{Ed} = \mathbf{0}$) is null.

Remark 3. At a kinematic singularity, $\text{rank}(\mathbf{J}_E) < m$. Obviously in this case, it has no meaning to consider the above attractive ball, and a different analysis is needed. Specifically, when $\mathbf{K}_E\mathbf{e}_E \in \mathcal{N}(\mathbf{J}_E^T)$ with $\mathbf{e}_E \neq \mathbf{0}$, where $\mathcal{N}(\mathbf{J}_E^T)$ indicates the null space of $\mathbf{J}_E^T$, it is $\dot{\mathbf{q}} = \mathbf{0}$, and the algorithm may in principle get "stuck." It can be easily shown, however, that such an equilibrium point is unstable as long as the time evolution of $\dot{\mathbf{x}}_{Ed}$ takes $\mathbf{K}_E\mathbf{e}_E$ outside $\mathcal{N}(\mathbf{J}_E^T)$. In the most general case, then, $\mathbf{K}_E\mathbf{e}_E$ may have only some components in the null space of $\mathbf{J}_E^T$. In such an occurrence, the algorithm will guarantee the convergence of the sole components of $\mathbf{K}_E\mathbf{e}_E$ outside the null space of $\mathbf{J}_E^T$.[2]

In view of the preceding, the CLIK scheme of Figure 1 can be used either *on-line*, for continuously solving reference end-effector trajectories into joint trajectories while guaranteeing an ultimately bounded tracking error and a null steady state error, or *off-line* to find a joint solution to a given constant end-effector location with guaranteed null positional error.

A more computationally demanding solution that guarantees better tracking performance [a null tracking error is obtained if $\mathbf{e}_E(0) = 0$] can be devised as

$$\dot{\mathbf{q}} = \mathbf{J}_E^{\dagger}(\mathbf{q})[\dot{\mathbf{x}}_{Ed} + \mathbf{K}_E\mathbf{e}_E], \tag{7}$$

which resembles the pseudoinverse solution (3) but it is inherently *closed loop* and thus avoids long-term drifts associated with the inversion of the mapping $\mathbf{J}_E$.

3. CLIK Schemes for Constrained Redundant Manipulators

The above CLIK schemes yield a joint solution for unconstrained redundant manipulators. It is shown in the following sections how those schemes can be modified in the case of *constrained* redundant manipulators.

3.1. Task Space Augmentation

One method of solving redundancy is to impose an additional constraint task to be executed along with the original end-effector task.[3] In details, a functional constraint task on the joint space variables can be considered in the form

$$\mathbf{x}_C = \mathbf{f}_C(\mathbf{q}), \tag{8}$$

where $\mathbf{f}_C$ is an $(r \times 1)$ vector with continuous derivative with respect to $\mathbf{q}$; also it is $r \leq (n - m)$, so as to span at most the whole redundant space. In analogy with eq. (2), one obtains

$$\dot{\mathbf{x}}_C = \mathbf{J}_C(\mathbf{q})\dot{\mathbf{q}}, \tag{9}$$

where $\mathbf{J}_C(\mathbf{q}) = \partial\mathbf{f}_C(\mathbf{q})/\partial\mathbf{q}$.

As a consequence, the *augmented* kinematic equation becomes

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_E \\ \mathbf{x}_C \end{bmatrix} = \begin{bmatrix} \mathbf{f}_E(\mathbf{q}) \\ \mathbf{f}_C(\mathbf{q}) \end{bmatrix}, \tag{10}$$

whose solution $\mathbf{q}$ *should* place the end effector at the location $\mathbf{x}_E$ and meet the constraint $\mathbf{x}_C$. Differentiating (10) with respect to time gives

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{J}_E(\mathbf{q}) \\ \mathbf{J}_C(\mathbf{q}) \end{bmatrix} \dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}, \tag{11}$$

where $\mathbf{J}(\mathbf{q})$ is the *augmented Jacobian* matrix. This technique can be ascribed to Baillieul (1985), who called it the *extended Jacobian technique*, although he considered constraint tasks of particular form, as discussed later. Incidentally, Egeland (1987) adopted

2. For further details about the occurrence of kinematic singularities, the reader is referred to Sciavicco and Siciliano (1988b).

3. For instance, the constraint task could be conveniently specified to keep the manipulator off the neighborhood of singular configurations, thus overcoming the problems raised in Remark 3.

the same framework to design a task space dynamic controller for a redundant manipulator with a small, fast manipulator mounted on a positioning part.

One advantage of the augmented task space approach when the space of redundancy is entirely exploited [i.e., $r = (n - m)$] is that any kinematic inversion method can be used while preserving the cyclic property (Baker and Wampler 1988). Therefore once the dimension of the task space vector and related Jacobian matrix has been suitably augmented so as to include the constraint task, the previous CLIK schemes can be adopted. The Jacobian transpose solution will then result in

$$\dot{\mathbf{q}} = \mathbf{J}^T(\mathbf{q})\mathbf{K}\mathbf{e}, \tag{12}$$

where $\mathbf{K}$ is a positive definite matrix and $\mathbf{e} = \mathbf{x}_d - \mathbf{x}$. As a matter of fact, Sciavicco and Siciliano (1987a,[4]b) obtained satisfactory results by utilizing the solution (12) for the case of a planar four-degree-of-freedom arm with an obstacle and/or a joint limit avoidance constraint and for the case of a three-degree-of-freedom arm with different dexterity constraints, respectively.

According to Remark 3, properly extended to the augmented task space case, problems may arise when $\mathbf{K}\mathbf{e} \in \mathcal{N}(\mathbf{J}^T)$ with $\mathbf{e} \neq \mathbf{0}$. The question then remains whether it is possible to ensure that the joint variables satisfy arbitrarily defined constraint tasks while executing the end-effector task. Even though the matrices $\mathbf{J}_E$ and $\mathbf{J}_C$ are full low rank, in fact, the augmented Jacobian matrix $\mathbf{J}$ in (11) is not guaranteed at all to have full rank for all $\mathbf{q}$s, and *algorithmic singularities* may occur (Baillieul 1986). In particular, the following result on the structural properties of the Jacobian matrices can be established:

PROPOSITION 1: If $\text{rank}(\mathbf{J}_E) = m$ and $\text{rank}(\mathbf{J}_C) = r$, then $\text{rank}(\mathbf{J}) = (r + m)$ if and only if $\mathcal{R}(\mathbf{J}_E^T) \cap \mathcal{R}(\mathbf{J}_C^T) = \{0\}$.

Proof: As $\text{rank}(\mathbf{J}) = \text{rank}(\mathbf{J}_E) + \text{rank}(\mathbf{J}_C) - \dim(\mathcal{R}(\mathbf{J}_E^T) \cap \mathcal{R}(\mathbf{J}_C^T))$, the proposition follows straightforwardly. □

In the preceding, it has been assumed that the constraint task function $\mathbf{f}_C$ could be set independently of the end-effector task, provided that the condition stated in Proposition 1 is satisfied. The most effective way is undoubtedly to construct an ad hoc constraint function for each specific task such that the augmented Jacobian matrix maintains full rank along the whole task execution. The example worked out in Sciavicco et al. (1988) and revisited in the case study section is one of such kind.

A conceptually equivalent approach to solving redundancy, which is dependent from the particular task, is to find an inverse kinematic function by giving the functional form of $(n - m)$ of the joint variables and then solve for the remaining joints as a nonredundant system (Wampler 1987).

3.2. *Task Priority Strategy*

As discussed in the previous section, a crucial point for the success of a task space augmentation approach is the proper choice of the constraint task function. On the other hand, the technique developed by Baillieul (1985) suggests that the null space of the end-effector Jacobian matrix constitutes an effective tool to handle constraints.

The null space of the Jacobian, indeed, is at the heart of the *task priority strategy*, independently proposed by Maciejewski and Klein (1985) and by Nakamura et al. (1987). The end-effector task and the constraint task are assigned different priorities in that the task of lower priority is satisfied only if it does not conflict with the task of higher priority. In the following, new CLIK schemes using the above strategy will be derived. Preliminary work in this direction has been reported by Das et al. (1988) and by Sciavicco and Siciliano (1988a).

Similarly to the end-effector error defined in (5), an error vector $\mathbf{e}_C$ between the reference and the actual constraint task vectors, $\mathbf{x}_{Cd}$ and $\mathbf{x}_C$ respectively, can be defined as

$$\mathbf{e}_C = \mathbf{x}_{Cd} - \mathbf{x}_C. \tag{13}$$

Obviously, it is also

$$\dot{\mathbf{e}}_C = \dot{\mathbf{x}}_{Cd} - \dot{\mathbf{x}}_C. \tag{14}$$

Assuming that higher priority is given to the end-effector task, the following holds:

THEOREM: Assume that $\mathbf{K}_E$ is a positive definite matrix and $\text{rank}(\mathbf{J}_E) = m$. The CLIK scheme based on the "control law"

$$\dot{\mathbf{q}} = \mathbf{J}_E^T(\mathbf{q})\mathbf{K}_E\mathbf{e}_E + [\mathbf{I} - \mathbf{J}_E^\dagger(\mathbf{q})\mathbf{J}_E(\mathbf{q})]\mathbf{J}_C^T(\mathbf{q})\mathbf{K}_C\mathbf{e}_C \tag{15}$$

ensures that the end-effector tracking error is ultimately bounded into an attractive ball centered at $\mathbf{e}_E = \mathbf{0}$, independently of the constraint task. Furthermore, if $\mathbf{K}_C$ is a positive definite matrix and $\text{rank}(\mathbf{J}_C) = r$, and if $\mathcal{R}(\mathbf{J}_E^T) \cap \mathcal{R}(\mathbf{J}_C^T) = \{0\}$, the constraint tracking error $\mathbf{e}_C$ is ultimately bounded into an attractive ball containing the origin; the radius of

4. This paper has been extended in Sciavicco and Siciliano (1988b).

the ball can be made arbitrarily small by suitably increasing the minimum eigenvalue of $\mathbf{K}_C$.

Proof: For the first part, the result simply follows by observing that the second term on the right side of (15) is in the null space of $\mathbf{J}_E$, and then it does not affect the end-effector task.

For the second part, define the positive definite Lyapunov function

$$V_C = \frac{1}{2}\mathbf{e}_C^T\mathbf{K}_C\mathbf{e}_C. \tag{16}$$

Its time derivative along the trajectories of the system (14), (9) under the control (15) turns out

$$\begin{aligned}\dot{V}_C = {} & \mathbf{e}_C^T\mathbf{K}_C^T\dot{\mathbf{x}}_{Cd} - \mathbf{e}_C^T\mathbf{K}_C^T\mathbf{J}_C(\mathbf{q})\mathbf{J}_E^\dagger(\mathbf{q})\mathbf{K}_E\mathbf{e}_E \\ & - \mathbf{e}_C^T\mathbf{K}_C^T\mathbf{J}_C(\mathbf{q})[\mathbf{I} - \mathbf{J}_E^\dagger(\mathbf{q})\mathbf{J}_E(\mathbf{q})]\mathbf{J}_C^T(\mathbf{q})\mathbf{K}_C\mathbf{e}_C.\end{aligned} \tag{17}$$

Because $\mathbf{J}_E$ and $\mathbf{J}_C$ are assumed full rank [i.e., $\dim(\mathcal{R}(\mathbf{J}_E^T)) = m$ and $\dim(\mathcal{R}(\mathbf{J}_C^T)) = r$ with $(m + r) \leq n$], the assumption $\mathcal{R}(\mathbf{J}_E^T) \cap \mathcal{R}(\mathbf{J}_C^T) = \{0\}$ implies that $\mathcal{R}(\mathbf{J}_C^T) \subseteq \mathcal{N}(\mathbf{J}_E)$. As $\mathbf{J}_C^T\mathbf{K}_C\mathbf{e}_C \in \mathcal{R}(\mathbf{J}_C^T)$, then $\mathbf{J}_C^T\mathbf{K}_C\mathbf{e}_C \in \mathcal{N}(\mathbf{J}_E)$. Being $(\mathbf{I} - \mathbf{J}_E^\dagger\mathbf{J}_E)$ a projector onto $\mathcal{N}(\mathbf{J}_E)$, the third term on the right side of (17) is negative definite, and $\dot{V}_C$ is upper bounded as

$$\begin{aligned}\dot{V}_C \leq {} & \| \mathbf{e}_C \| \lambda_M(\mathbf{K}_C) \| \dot{\mathbf{x}}_{Cd} \|_M \\ & + \| \mathbf{e}_C \| \lambda_M(\mathbf{K}_C)(\lambda_M(\mathbf{J}_C\mathbf{J}_C^T))^{1/2} \\ & \times (\lambda_M(\mathbf{J}_E\mathbf{J}_E^T))^{1/2}\lambda_M(\mathbf{K}_E) \| \mathbf{e}_E \| \\ & - \| \mathbf{e}_C \|^2(\lambda_m(\mathbf{K}_C))^2\lambda_m(\mathbf{J}_C(\mathbf{I} - \mathbf{J}_E^\dagger\mathbf{J}_E)\mathbf{J}_C^T)\end{aligned} \tag{18}$$

where $\lambda_{m(M)}(\mathbf{A})$ denotes the minimum (maximum) eigenvalue of matrix $\mathbf{A}$, and $\| \mathbf{v} \|_M$ denotes the maximum of the usual euclidean norm of vector $\mathbf{v}$.

Accounting for the bounds on V_C

$$\frac{1}{2}\lambda_m(\mathbf{K}_C) \| \mathbf{e}_C \|^2 \leq V_C \leq \frac{1}{2}\lambda_M(\mathbf{K}_C) \| \mathbf{e}_C \|^2, \tag{19}$$

eq. (18) yields

$$\begin{aligned}\dot{V}_C \leq {} & \sqrt{2}\frac{\lambda_M(\mathbf{K}_C)}{(\lambda_m(\mathbf{K}_C))^{1/2}}[\| \dot{\mathbf{x}}_{Cd} \|_M \\ & + (\lambda_M(\mathbf{J}_C\mathbf{J}_C^T))^{1/2}(\lambda_M(\mathbf{J}_E\mathbf{J}_E^T))^{1/2}\lambda_M(\mathbf{K}_E) \| \mathbf{e}_E \|]V_C^{1/2} \\ & - 2\frac{(\lambda_m(\mathbf{K}_C))^2}{\lambda_m(\mathbf{K}_C)}\lambda_m(\mathbf{J}_C(\mathbf{I} - \mathbf{J}_E^\dagger\mathbf{J}_E)\mathbf{J}_C^T)V_C.\end{aligned} \tag{20}$$

From this expression, it follows that $\dot{V}_C < 0$ for $V_C > \alpha_C^2$, and $\dot{V}_C \leq 0$ for $V_C = \alpha_C^2$, where

$$\alpha_C = \frac{(\lambda_m(\mathbf{K}_C))^2[\| \dot{\mathbf{x}}_{Cd} \|_M + (\lambda_m(\mathbf{J}_C\mathbf{J}_C^T))^{1/2}(\lambda_M(\mathbf{J}_E\mathbf{J}_E^T))^{1/2}\lambda_M(\mathbf{K}_E) \| \mathbf{e}_E \|]}{\sqrt{2}(\lambda_m(\mathbf{K}_C))^{5/2}\lambda_m(\mathbf{J}_C(\mathbf{I} - \mathbf{J}_E^\dagger\mathbf{J}_E)\mathbf{J}_C^T)}. \tag{21}$$

The constraint tracking error $\mathbf{e}_C$ is then uniformly ultimately bounded into the ball centered at $\mathbf{e}_C = \mathbf{0}$ of radius (Corless and Leitmann 1981)

$$\rho_C = \frac{\alpha_C}{(\lambda_m(\mathbf{K}_C))^{1/2}}, \tag{22}$$

which concludes the proof. □

The resulting CLIK scheme is illustrated in Figure 2. Remarks 1 to 3 still hold. In addition:

Remark 4: If $\text{rank}(\mathbf{J}_C) = r$ but $\mathcal{R}(\mathbf{J}_E^T) \cap \mathcal{R}(\mathbf{J}_C^T) \neq \{0\}$, when $\mathbf{J}_C^T\mathbf{K}_C\mathbf{e}_C \in \mathcal{R}(\mathbf{J}_E^T)$, the third term on the right side of (17) vanishes with $\mathbf{e}_C \neq \mathbf{0}$ (*algorithmic singularity*). In particular, at steady state (i.e., when $\dot{\mathbf{x}}_{Cd} = \mathbf{0}$ and $\dot{\mathbf{x}}_{Ed} = \mathbf{0}$), one obtains $\dot{V}_C = 0$, and some components of $\mathbf{e}_C$ may get stuck to a nonzero value (analogously to what was discussed in Remark 3 for the end-effector task). But $\mathbf{e}_E$ is null in any case, in force of the task priority strategy accomplished.

Remark 5: The order between $\mathbf{e}_E$ and $\mathbf{e}_C$ established by (15) can be inverted whenever desired, i.e.,

$$\dot{\mathbf{q}} = \mathbf{J}_C^T(\mathbf{q})\mathbf{K}_C\mathbf{e}_C + [\mathbf{I} - \mathbf{J}_C^\dagger(\mathbf{q})\mathbf{J}_C(\mathbf{q})]\mathbf{J}_E^T(\mathbf{q})\mathbf{K}_E\mathbf{e}_E \tag{23}$$

with obvious transposition of the above results.

It can be observed that the solution (15) requires the computation of $\mathbf{J}_E^\dagger$ in any case. This suggests using a solution of the kind shown in (7) for the end-effector task but still preserving the use of the Jacobian transpose for the constraint task. To the purpose, the following holds:

COROLLARY: Under the same assumptions of the

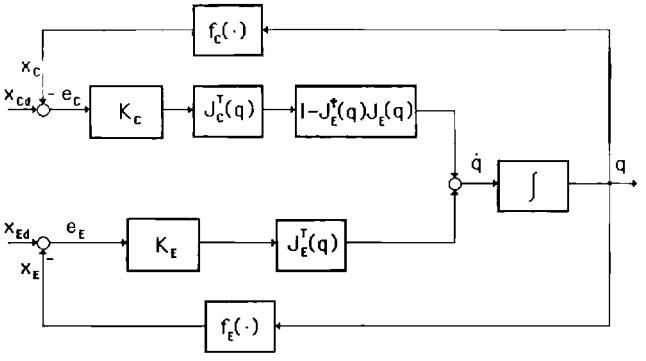

Fig. 2. The CLIK Jacobian transpose scheme for end-effector and constraint tasks with priority to the end-effector task.

above theorem, the "control law"

$$\dot{\mathbf{q}} = \mathbf{J}_E^{\dagger}(\mathbf{q})[\dot{\mathbf{x}}_{Ed} + \mathbf{K}_E\mathbf{e}_E] + [\mathbf{I} - \mathbf{J}_E^{\dagger}(\mathbf{q})\mathbf{J}_E(\mathbf{q})]\mathbf{J}_C^T(\mathbf{q})\mathbf{K}_C\mathbf{e}_C \tag{24}$$

ensures that the end-effector tracking error is null, provided that $\mathbf{e}_E$ at time $t = 0$ is null, independently of the constraint task. Furthermore, the constraint tracking error $\mathbf{e}_C$ is ultimately bounded as above, but the radius of the attractive ball is smaller than (22) if

$$\frac{(\lambda_M(\mathbf{K}_E))^2(\lambda_M(\mathbf{J}_E\mathbf{J}_E^T))^{1/2}}{(\lambda_m(\mathbf{K}_E))^2(\lambda_m(\mathbf{J}_E\mathbf{J}_E^T))^{1/2}} > 1. \tag{25}$$

Proof: Regarding the end-effector task, the performance of the solution (24) is the same as that of the solution (7), in force of the task priority.

Proceeding as in the proof of the theorem, the new α_C in (21) under the control (24) becomes

$$\alpha_C' = \frac{(\lambda_M(\mathbf{K}_C))^2[\,\|\,\dot{\mathbf{x}}_{Cd}\,\|_M + (\lambda_M(\mathbf{J}_C\mathbf{J}_C^T))^{1/2} \times (\lambda_M(\mathbf{J}_E^{\dagger T}\mathbf{J}_E^{\dagger}))^{1/2}(\,\|\,\dot{\mathbf{x}}_{Ed}\,\|_M + \lambda_M(\mathbf{K}_E)\,\|\,\mathbf{e}_E\,\|\,)]}{\sqrt{2}(\lambda_m(\mathbf{K}_C))^{5/2}\lambda_m(\mathbf{J}_C(\mathbf{I} - \mathbf{J}_E^{\dagger}\mathbf{J}_E)\mathbf{J}_C^T)} \tag{26}$$

As $\|\,\mathbf{e}_E\,\| = 0$ from above, while an upper bound for $\|\,\mathbf{e}_E\,\|$ under the control (15) can be easily obtained with the same technique as in Balestrino et al. (1984) and in Wolovich and Elliott (1984), the result follows from comparison of (26) with (21) and observing that $\lambda_M(\mathbf{J}_E^{\dagger T}\mathbf{J}_E^{\dagger}) = 1/\lambda_m(\mathbf{J}_E\mathbf{J}_E^T)$.

□

The resulting CLIK scheme is shown in Figure 3. Clearly, both Remarks 4 and 5 apply also to this case.

It should be pointed out that, differently from previous solutions with task priority based on pure pseudoinverses (Maciejewski and Klein 1985; Nakamura et al. 1987), both schemes of Figures 2 and 3 require the computation of the transpose of the constraint Jacobian.

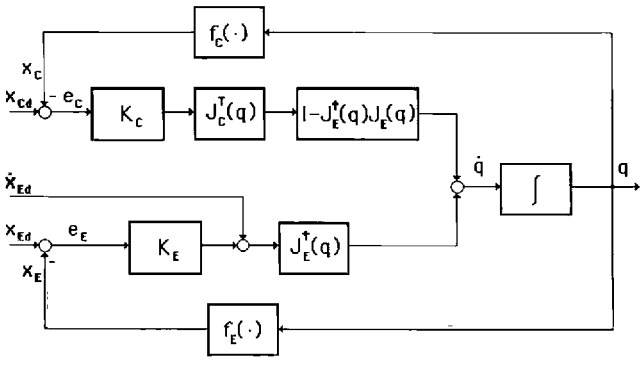

Fig. 3. The CLIK scheme for end-effector and constraint tasks with end-effector Jacobian pseudoinverse and constraint Jacobian transpose.

Notice that the second term on the right side of (17) describes the interference of the end-effector task onto the constraint task. In order to cancel out this term, the solution (24) can be modified into:

$$\dot{\mathbf{q}} = \mathbf{J}_E^{\dagger}(\mathbf{q})[\dot{\mathbf{x}}_{Ed} + \mathbf{K}_E\mathbf{e}_E] + \tilde{\mathbf{J}}_C^{\dagger}(\mathbf{q})\{\dot{\mathbf{x}}_{Cd} - \mathbf{J}_C(\mathbf{q})\mathbf{J}_E^{\dagger}(\mathbf{q})[\dot{\mathbf{x}}_{Ed} + \mathbf{K}_E\mathbf{e}_E]\mathbf{K}_C\mathbf{e}_C\}, \tag{27}$$

where $\tilde{\mathbf{J}}_C = \mathbf{J}_C(\mathbf{I} - \mathbf{J}_E^{\dagger}\mathbf{J}_E)$, which is logically equivalent to the task priority scheme proposed by Maciejewski and Klein (1985) and by Nakamura et al. (1987), as it requires the computation of both $\mathbf{J}_E^{\dagger}$ and $\tilde{\mathbf{J}}_C^{\dagger}$.

Regarding the possibility that $\tilde{\mathbf{J}}_C$ admits a pseudoinverse, the following can be stated:

PROPOSITION 2: If $\text{rank}(\mathbf{J}_E) = m$ and $\text{rank}(\mathbf{J}_C) = r$, then $\text{rank}(\tilde{\mathbf{J}}_C) = r$ if and only if $\mathcal{R}(\mathbf{J}_C^T) \cap \mathcal{R}(\mathbf{J}_E^T) = \{0\}$.

Proof: First observe that $\text{rank}(\mathbf{J}_E) = m$ implies that $\text{rank}(\mathbf{I} - \mathbf{J}_E^{\dagger}\mathbf{J}_E) = (n - m)$. Then, as $\text{rank}((\mathbf{I} - \mathbf{J}_E^{\dagger}\mathbf{J}_E)\mathbf{J}_C^T) = \text{rank}((\mathbf{J}_C^T) - \dim(\mathcal{R}(\mathbf{J}_C^T) \cap \mathcal{N}(\mathbf{I} - \mathbf{J}_E^{\dagger}\mathbf{J}_E))$, the proposition follows by noticing that $\mathcal{N}(\mathbf{I} - \mathbf{J}_E^{\dagger}\mathbf{J}_E) = \mathcal{R}(\mathbf{J}_E^T)$.

□

Remark 6: If a pseudoinverse to $\tilde{\mathbf{J}}_C$ exists, then $\text{rank}(\mathbf{J}) = (r + m)$ and the solution (27) becomes just a computationally simpler expression of a pure augmented task space solution of the kind $\dot{\mathbf{q}} = \mathbf{J}^{+}(\dot{\mathbf{x}} + \mathbf{K}\mathbf{e})$, with $\mathbf{J}$ as in (11).

On the basis of the above remarks, it can be recognized that the solutions (15) and (24) are to be preferred to the solution (27) when the augmented Jacobian matrix $\mathbf{J}$ is not guaranteed to have full rank (see Proposition 1) (i.e., whenever there could be incompatibility between the end effector and the constraint tasks). Nevertheless, the solution (24) is computationally simpler than solution (27), because it replaces some pseudoinverse calculations with the Jacobian transpose. The savings in computation comes at the cost of larger tracking errors for the constraint task, but because the constraint is lower priority—and often constant over time—some error is not crucial, and the computational savings is worthwhile. Three of the case studies developed in the following section will provide a clear under-

standing of the potential offered by the CLIK schemes based on the above solutions.

4. Case Studies

Four different case studies are developed to illustrate the performance of the proposed CLIK schemes.

The above Lyapunov analysis was aimed at proving the convergence of the schemes with the Jacobian transpose in the continuous-time domain, whereas the schemes are to be implemented with finite sampling rates. This is anticipated to limit the maximum values of the feedback gains. Preliminary analyses in the discrete-time domain were attempted by De Maria and Marino (1985) and by Das et al. (1988), which revealed that the feedback gains are to be chosen of the order of the sampling rate, and they should be adjusted as a function of the manipulator's configurations. In all examples, a fine tuning of the feedback gains has been accomplished according to these indications. A simple forward Euler integration rule has been used with a sampling time of 1 ms, in view of an on-line implementation of the CLIK schemes; this is seen to be sufficient to perform all the computations required by the schemes at each sample with currently available microprocessor architectures. It is understood, however, that if there is enough time (on-line or off-line), more than one iteration can be used to reduce the tracking error to essentially zero, as it would normally be done in any numerical method for finding the solution of a set of nonlinear equations.

4.1. Case Study 1

For special applications in nuclear fusion plants, maintainance robots are usually required to operate in a constrained plasma vessel toroidal environment. To illustrate a typical case study, a planar snakelike seven-degree-of-freedom robot is considered (Sciavicco et al. 1988).

The manipulator task is to reach a desired end-effector position located inside an annular region. The manipulator is assumed to start from a joint configuration that places the end effector outside the torus.

The insertion strategy requires that once the joints (their Cartesian positions) enter the torus, they are constrained to belong to a geometric locus so as to avoid collisions between manipulator links and torus inner walls. The chosen locus is a circle of radius $(r + R)/2$ concentric with the two circles of radii r and R that mark the annular region boundaries.

As the degree of redundancy is equal to five, no more than five constraints can be added to define the augmented task space. In view of the described task, forcing to zero the distance between Cartesian joint positions and the geometric locus can be assumed as an attractivity constraint, i.e.,

$$f_{Ci}(\mathbf{q}^*) = \| \mathbf{p}_T - \mathbf{f}_i(\mathbf{q}^*) \|,$$

$$\mathbf{q}^{*T} = (q_1 \ldots q_{i-1}), \qquad i = 4, 8$$

where $\mathbf{p}_T$ are the annular region center coordinates, and $\mathbf{f}_i(\mathbf{q}^*)$ is the direct kinematic function of the ith joint, both with respect to the same base frame. The desired value of the constraint is obviously chosen as $x_{Cdi} = (r + R)/2$. In the planar case it turns out that

$$\mathbf{f}_i(\mathbf{q}^*) = \begin{bmatrix} f_{ix}(\mathbf{q}^*) \\ f_{iy}(\mathbf{q}^*) \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{i-1} l_j c_{1j} \\ \sum_{j=1}^{i-1} l_j s_{1j} \end{bmatrix}, \qquad i = 4, 8$$

where $c_{1j} = \cos(q_1 + \cdots + q_j)$ and $s_{1j} = \sin(q_1 + \cdots + q_j)$; notice that $\mathbf{f}_8 = \mathbf{f}_E$.

It must be emphasized that an insertion strategy that *pulls* the end effector along a trajectory inside the torus while sequentially activating the constraints on the joints 7, 6, . . . would be problematic. It may easily happen, indeed, that the position constraint on the end effector conflicts with the attractivity constraints on the intermediate joints, leading to the occurrence of *algorithmic singularities*.

Conversely, a more effective strategy is chosen that *pushes* the manipulator inside the torus. The joints are driven inside the torus one at a time, and then the corresponding attractivity constraint is activated while the position constraint is relaxed. Moreover, it is assumed that the ith joint constraint is satisfied by varying only the $(i - 1)$th joint variable. This is crucial in order to avoid the above algorithmic singularities, as the augmented Jacobian matrix in (11) will turn out to be block diagonal.

The pushing strategy is described in the following by listing each sequential subtask that brings a joint external to the torus to the point P (Fig. 4) inside the torus and activates the attractivity constraints for the joints already inside:

Subtask	*Position Constraint*	*Attractivity Constraints*
1	$\mathbf{f}_8(q_6, q_7)$	None
2	$\mathbf{f}_7(q_5, q_6)$	$f_{C8}(q_7)$
3	$\mathbf{f}_6(q_4, q_5)$	$f_{C8}(q_7)$, $f_{C7}(q_6)$
⋮	⋮	⋮
6	$\mathbf{f}_3(q_1, q_2)$	$f_{C8}(q_7)$, $f_{C7}(q_6)$, $f_{C6}(q_5)$, $f_{C5}(q_4)$, $f_{C4}(q_3)$

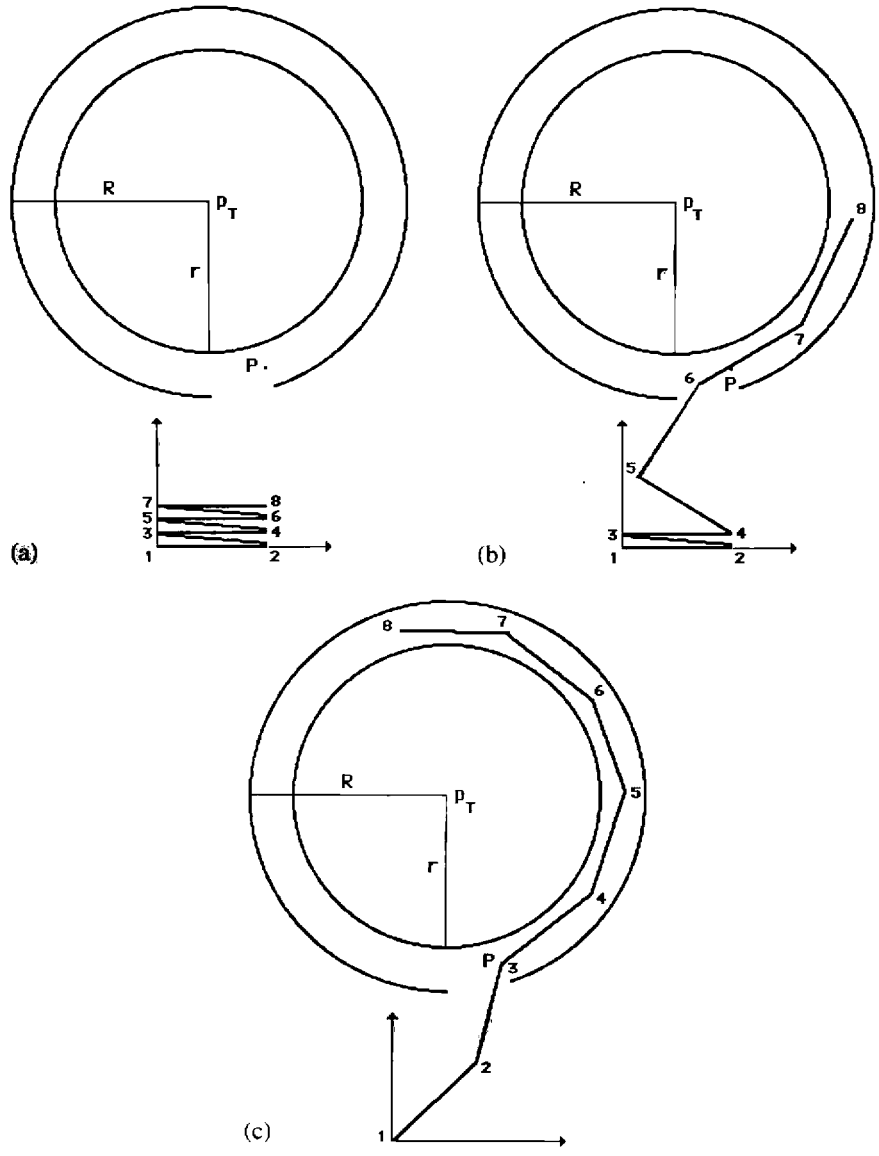

Fig. 4. Case study 1: A snakelike planar robot operating in a constrained annular region. A, *Initial configuration.* B, *Intermediate configuration.* C, *Final configuration.*

The notation $\mathbf{f}_j(q_{j-1}, q_{j-2})$ means that the jth joint trajectory tracking is obtained by moving only the $(j - 1)$th and the $(j - 2)$th joint variables; $f_{Ck}(q_{k-1})$ denotes the fact that the kth joint is forced to belong to the geometric locus by moving only the $(k - 1)$th joint variable.

In view of the preceding, the augmented kinematic function during the execution of the generic subtask **is**

$$\begin{bmatrix} \mathbf{p}_i \\ \mathbf{x}_C \end{bmatrix} = \begin{bmatrix} \mathbf{f}_i(q_1, \ldots, q_{i-1}) \\ f_{C,i+1}(q_1, \ldots, q_i) \\ \vdots \\ f_{C8}(q_1, \ldots, q_7) \end{bmatrix},$$

where $\mathbf{p}_i$ denotes the position of the ith joint.

The associated augmented Jacobian matrix is obtained by differentiating the above kinematic equation with respect only to the joint variables chosen to satisfy each constraint, i.e.,

$$\mathbf{J}(\mathbf{q}) = \text{block diag}\left(\frac{\partial \mathbf{f}_i}{\partial q_{i-1}\partial q_{i-2}}, \frac{\partial f_{C,i+1}}{\partial q_i}, \ldots, \frac{\partial f_{C8}}{\partial q_7}\right).$$

It can be recognized that the dimension of the augmented task space progressively increases from 2 to 7 in passing from subtask 1 to 6. In each subtask algorithmic singularities cannot occur at all because of the block diagonal structure of $\mathbf{J}$, provided that kinematic singularities [i.e., $\text{rank}(\partial \mathbf{f}_i/\partial q_{i-1}\partial q_{i-2}) < 2$] do not occur.

The manipulator consists of seven equal links of length 1.236 m. The torus has an inner radius r = 1.75 m and an outer radius R = 2.25 m with center at $\mathbf{p}_T^T$ = (0.618 4) m with respect to the base frame of the manipulator.

Figure 4*A* shows the starting configuration when the arm is wound up at $\mathbf{q}^T(0)$ = (0 175 −175 175 −175 175 −175)°.

The CLIK scheme based on (12) is then applied according to the above keen task space augmentation with $\mathbf{K}$ = 30 $\mathbf{I}$. In Figure 4*B* an intermediate configuration is illustrated corresponding to subtask 3. The final configuration is shown in Figure 4*C* where the end effector has reached a point diametrically opposed to P. The resulting joint trajectories are plotted in Figure 5, where the sequence of motions in the joint variables $q_7, q_6, \ldots, q_1$ can be recognized according to the logic of the above subtasks.

It is understood that the extraction of the manipulator from the torus can be performed by following a logically inverse strategy of the insertion strategy proposed.

4.2. *Case Study 2*

The end effector of a three-degree-of-freedom planar arm is commanded to follow a given path while keeping a desired orientation. The end-effector task vector is

$$\mathbf{x}_E = \begin{bmatrix} \sum_{i=1}^{3} l_i c_{1i} \\ \sum_{i=1}^{3} l_i s_{1i} \end{bmatrix},$$

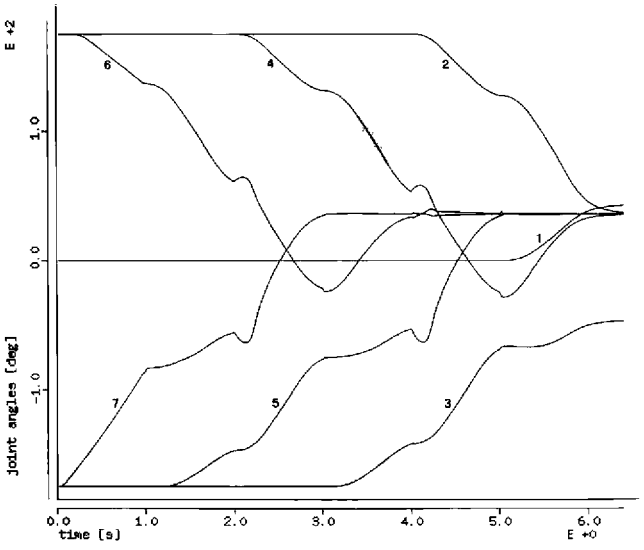

Fig. 5. Joint trajectories for case study 1 with the CLIK scheme of Fig. 3.

whereas the constraint task is specified as

$$x_C = q_1 + q_2 + q_3.$$

The link lengths are 0.5, 0.3, and 0.2 m. The desired path is

$$\mathbf{x}_{Ed} = \begin{cases} \begin{bmatrix} 0.7 - 0.2\cos(\pi t) \\ 0.2\sin(\pi t) \end{bmatrix} \text{m}, & 0 \le t < 6 \text{ s}; \\ \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} \text{m}, & t \ge 6 \text{ s}. \end{cases}$$

The desired orientation constraint

$$x_{Cd} = -90°$$

imposes the third link to point downward. The initial joint configuration is chosen such that $\mathbf{x}_E(0) = \mathbf{x}_{Ed}(0)$ and $x_C(0) = x_{Cd}(0)$. Figure 6 illustrates the task layout. Notice that the constraint task cannot be fulfilled along the whole end-effector path.

The CLIK scheme based on the solution (12) is applied first with $\mathbf{K}$ = diag (1500, 1500, 500). It can be recognized that neither the end-effector task nor the constraint task are tracked satisfactorily (Fig. 7*A*). This is imputable to the conflict arising between the two tasks when the first two links become aligned (q_2 = 0) [i.e., rank($\mathbf{J}$) = 2 (see Proposition 1)]. From then on, both tracking errors increase as a result of persistent rank deficiency until the given path permits the arm to recover the desired orientation. Notice that both errors vanish at steady state. As anticipated in theory, the joint trajectories—not reported here for brevity—are repetitive.[5]

The CLIK scheme based on solution (15) is applied next, with $\mathbf{K}_E$ = 1500 $\mathbf{I}$ and k_C = 500 so as to compare with the previous scheme. Differently from above, Figure 7*B* shows that the end-effector tracking error is decreased along the whole path. The constraint task is satisfied when it does not disturb the primary task; otherwise the constraint tracking error is increased, having given priority to the end-effector task (see Remark 4).

Finally the CLIK scheme based on the solution (24) is applied with the same $\mathbf{K}_E$ and k_C as above. Because the end-effector Jacobian pseudoinverse is used, the end-effector tracking error is remarkably one order of magnitude smaller, while the constraint tracking error essentially remains the same (Fig. 7*C*).

5. This will be the case of all the following examples with a "full" task space augmentation.

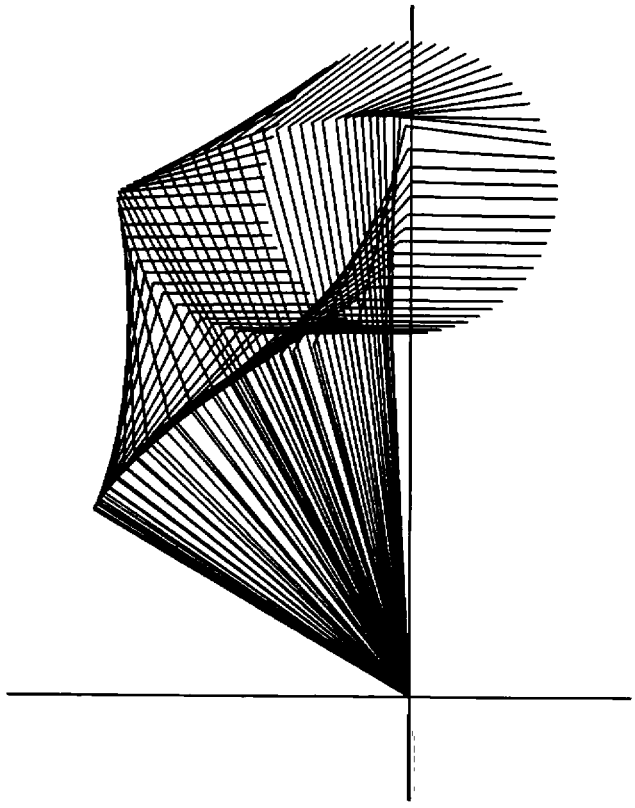

Fig. 6. Case study 2: A three-degree-of-freedom planar arm tracking a circular path while attempting to maintain a desired orientation.

4.3. Case Study 3

The end effector of a three-degree-of-freedom planar arm is commanded to follow a given path while keeping a desired value of dexterity measure. The end-effector task vector is the same as in the previous case study, while the constraint task is

$$x_C = q_3 - q_2.$$

The link lengths are all 0.5 m. The desired path is the same as in the previous case study, while the desired dexterity constraint

$$x_{Cd} = 0°$$

requires the arm to be kept in a "dexterous" configuration. The initial joint configuration is chosen such that $\mathbf{x}_E(0) = \mathbf{x}_{Ed}(0)$ and $x_C(0) = -60°$. Figure 8 illustrates the task layout. Notice that there are no conflicts between the two tasks, but the constraint task is not satisfied at the initial arm configuration.

The CLIK scheme based on the solution (12) is applied first with $\mathbf{K} = \text{diag}\,(500, 500, 5)$. It can be recognized that a short transient occurs, during which the initial constraint error is recovered to the detriment of the end-effector tracking error (Fig. 9*A*). It should be pointed out that a low value of the third element of $\mathbf{K}$ has been chosen with the purpose of achieving a smooth recovery. From then on q_2 is kept very close to q_3, and both tasks are satisfied; the errors vanish at steady state. It could be seen that the joint trajectories become repetitive after the transient is over.

The CLIK scheme based on the solution (15) is

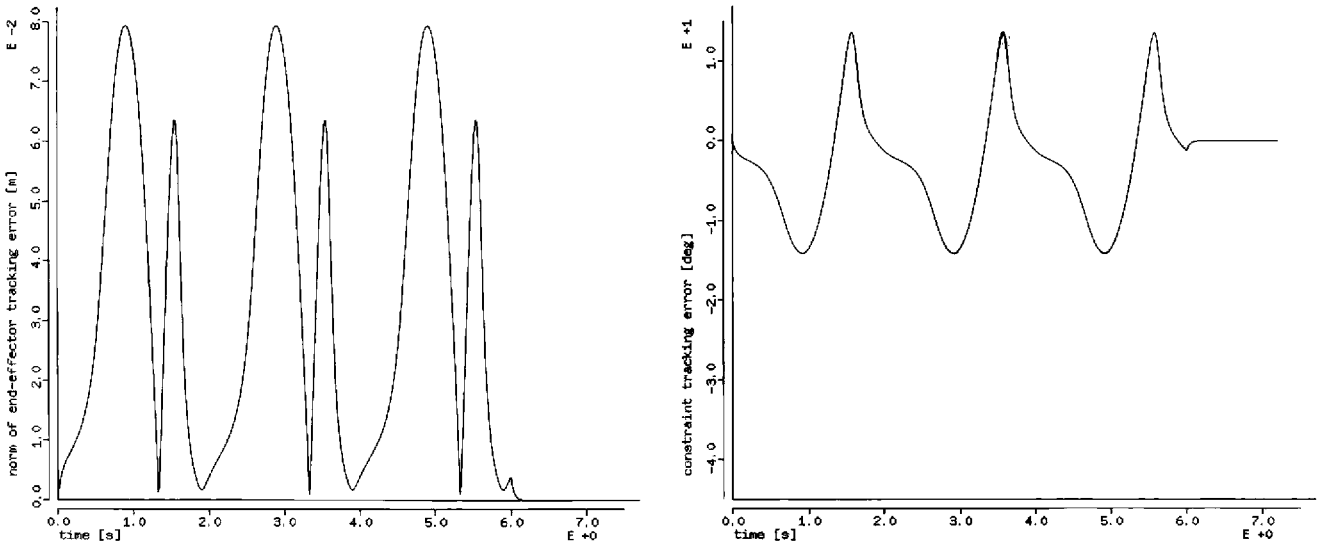

Fig. 7a. Case study 2: Task space tracking errors with solution (12).

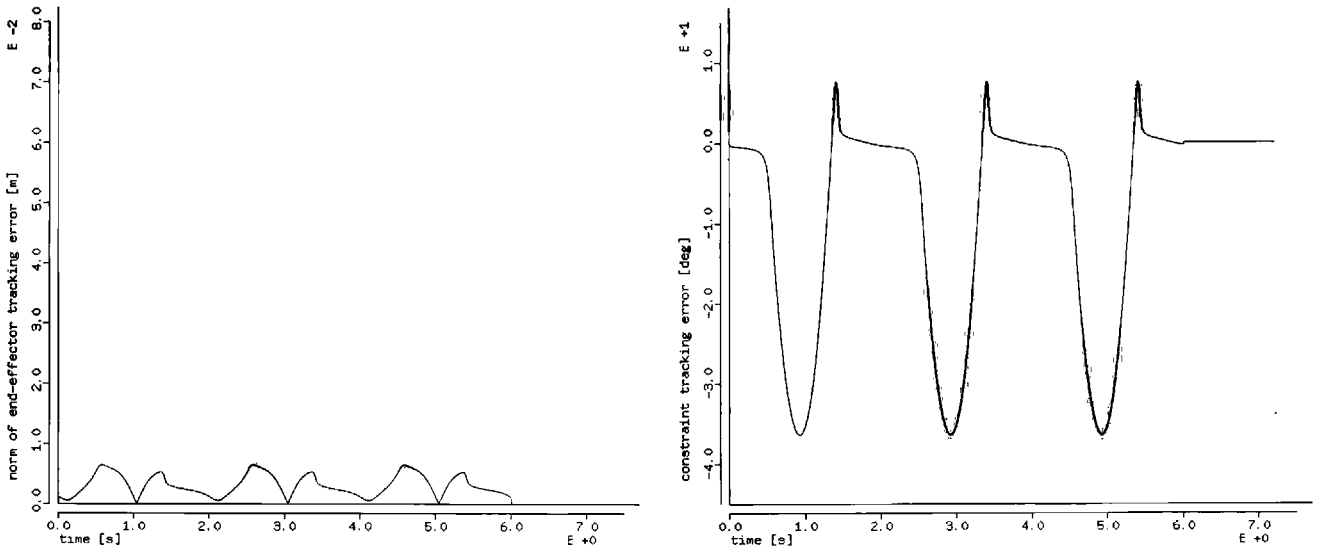

Fig. 7b. Case study 2: Task space tracking errors with solution (15).

applied next, with $\mathbf{K}_E = 500\ \mathbf{I}$ and $k_C = 5$ so as to compare with the previous scheme. The advantage gained with respect to the previous scheme is that the end-effector tracking performance is satisfactory, even during the transient, in force of the task priority strategy, while the constraint tracking error does not appreciably change (Fig. 9*B*).

The CLIK scheme based on solution (24) is finally applied with the same $\mathbf{K}_E$ and k_C as above. Because the end-effector Jacobian pseudoinverse is used, the end-effector tracking error is remarkably one order of magnitude smaller, while the constraint tracking error is essentially the same (Fig. 9*C*).

4.4. Case Study 4

The end effector of a two-degree-of-freedom planar arm is commanded to follow a given path while avoiding a disk-shaped obstacle of radius r_0 and center at $\mathbf{p}_O$ with respect to the base frame. The end-effector task vector is

$$\mathbf{x}_E = \begin{bmatrix} \sum_{i=1}^{2} l_i c_{1i} \\ \sum_{i=1}^{2} l_i s_{1i} \end{bmatrix},$$

while the constraint task is specified as

$$x_C = \| \mathbf{x}_E - \mathbf{p}_O \|.$$

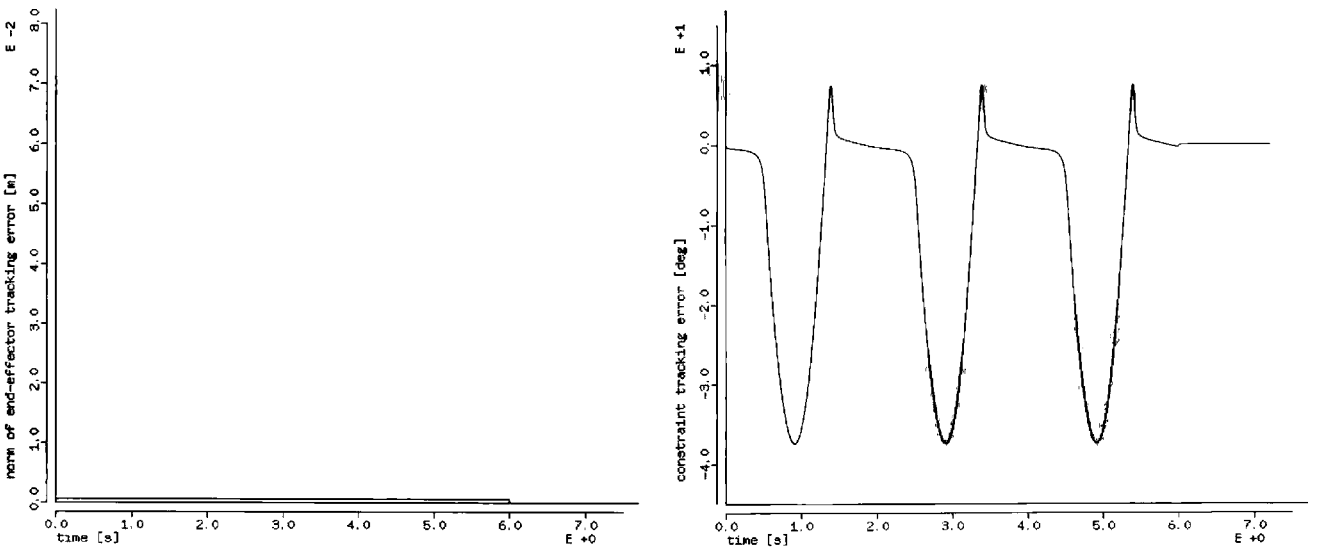

Fig. 7c. Case study 2: Task space tracking errors with solution (24).

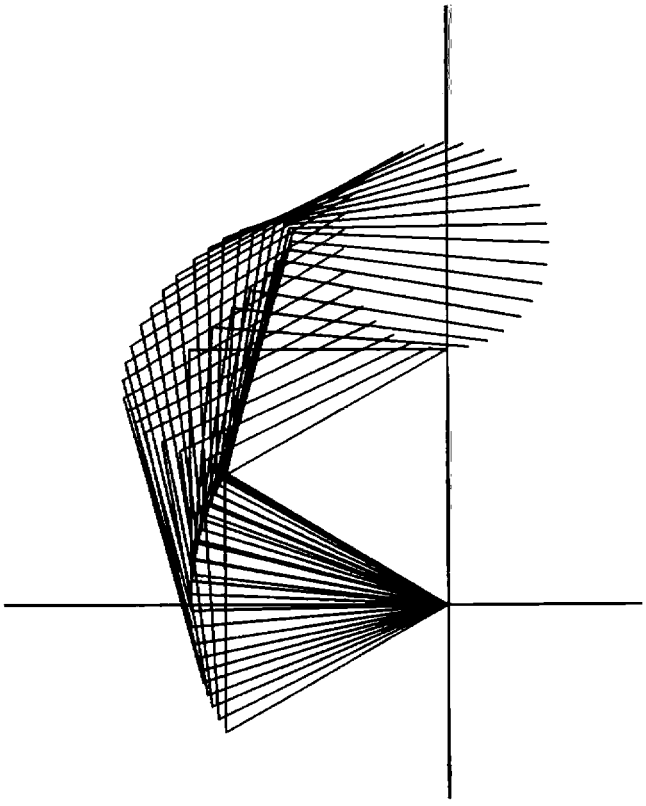

Fig. 8. Case study 3: A three-degree-of-freedom planar arm tracking a circular path while achieving a desired dexterity measure.

The link lengths are both 0.5 m. The obstacle is characterized by $r_0 = 0.175$ m and $\mathbf{p}_O^T = (0.65 \quad 0.25)$ m. The desired path is

$$\mathbf{x}_{Ed} = \begin{bmatrix} p_{dx} \\ p_{dy} \end{bmatrix}$$

$$p_{dx} = 0.5 \text{ m}$$

$$p_{dy} = \begin{cases} 0.5 - 0.253t^2 \text{ m}, & 0 \le t < 0.888 \text{ s}; \\ 0.699 - 0.449t \text{ m}, & 0.888 \le t < 1.112 \text{ s}; \\ 1.012 - 1.012t + 0.253t^2 \text{ m}, & 1.112 \le t < 2 \text{ s}; \\ 0 \text{ m}, & t \ge 2 \text{ s}; \end{cases}$$

while the desired obstacle avoidance constraint is

$$x_{Cd} = 0.2 \text{ m}$$

such that a 0.025 m safety threshold distance is guaranteed. The initial joint configuration is chosen such that $\mathbf{x}_E(0) = \mathbf{x}_{Ed}(0)$ and $x_C(0) > x_{Cd}$. Figure 10 illustrates the task layout.

Although the constraint task is not active, the CLIK scheme based on the solution (6) is applied with $\mathbf{K}_E = 2000\,\mathbf{I}$. When the given path enters the collision area specified by x_{Cd} (i.e., $\| \mathbf{x}_{Ed} - \mathbf{p}_O \| \le 0.2$), the algorithm is switched to the CLIK scheme based on the solution (23) with $\mathbf{K}_E$ as above and $k_C = 2000$. This ensures that priority is given to the constraint task. As expected, the actual end-effector path deviates from the original straight one, thus surrounding the obstacle in the collision area.

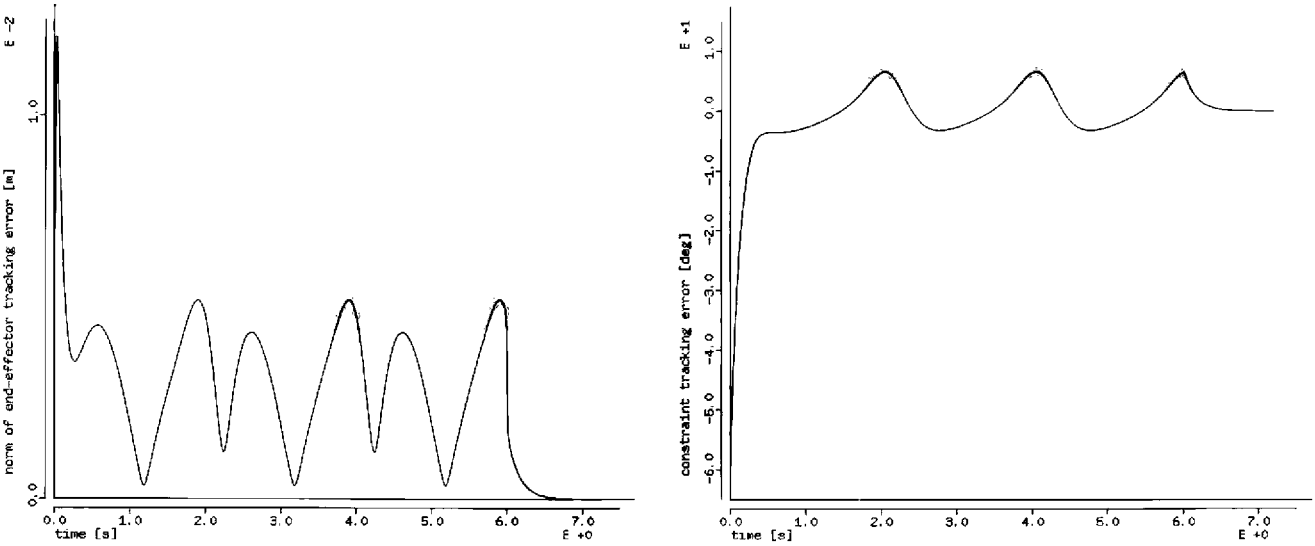

Fig. 9a. Case study 3: Task space tracking errors with solution (12).

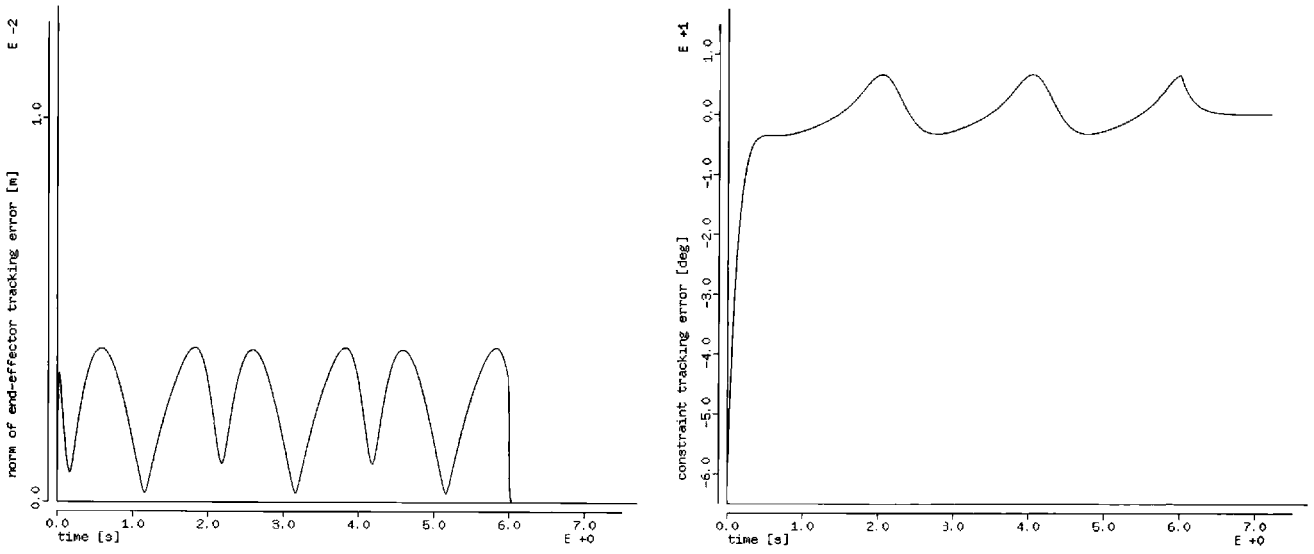

Fig. 9b. Case study 3: Task space tracking errors with solution (15).

5. Conclusions

The problem of solving the inverse kinematics for redundant manipulators when additional constraints are imposed has been studied in this article. The solution has been obtained in the framework of closed-loop inverse kinematics algorithms. It has been shown how the end-effector task can be suitably augmented with the constraint task, and a solution based on the Jacobian transpose can be devised. Using this solution, the fulfullment of both tasks is possible only if algorithmic singularities do not arise from conflicting task situations.

Case study 1 has provided an example of ad hoc specification of the constraint task so as to guarantee full rank of the augmented Jacobian over the entire task. On the other hand, in case study 2, the constraint task could not always be satisfied along with the end-effector task; in such a case, the results of the pure Jacobian transpose scheme have confirmed that the tracking performance is poor with respect to both tasks.

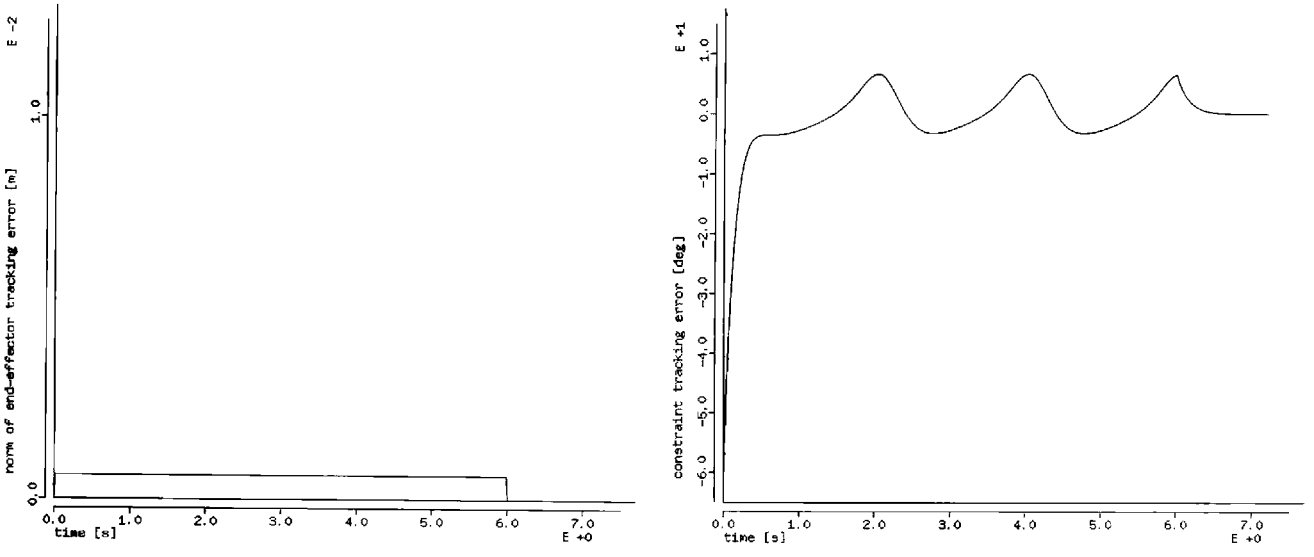

Fig. 9c. Case study 3: Task space tracking errors with solution (24).

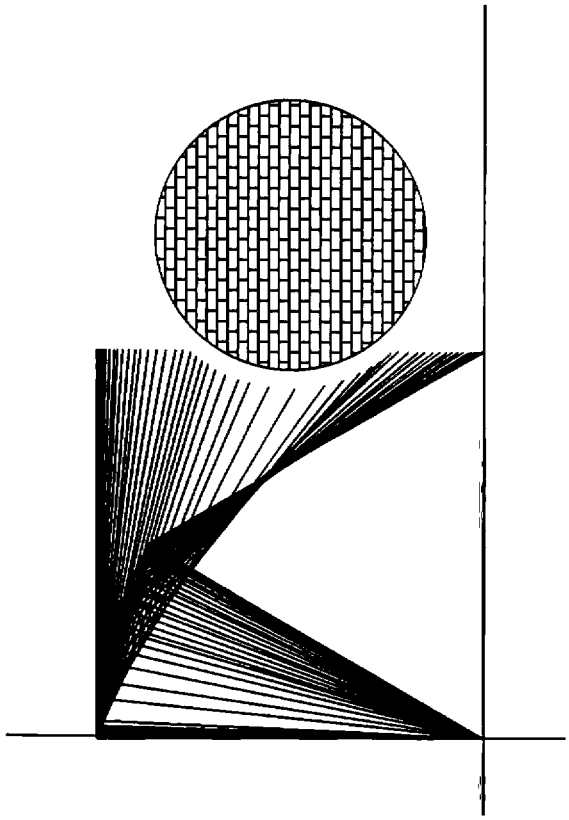

Fig. 10. Case study 4: A two-degree-of-freedom planar arm tracking a preplanned straight line path while avoiding a disk-shaped obstacle.

By adopting a task priority strategy, a new closed-loop inverse kinematics scheme has been developed by projecting the constraint Jacobian onto the null space of the end-effector Jacobian. Priority is given to the end effector (but the order can be inverted, if desired; see case study 4), and the convergence of the constraint task error has been proved via a Lyapunov argument. The results in case study 2 have indeed confirmed the theory in that end-effector tracking performance remains satisfactory when the constraint task conflicts with the primary task. Remarkably, the convergence of the scheme has been further tested in case study 3 where the constraint task is not satisfied at the initial arm configuration.

Finally, by observing that the above projection operation requires the computation of the pseudoinverse of the higher priority task Jacobian, the previous scheme has been modified. The new scheme preserves the nice feature of requiring the computation of the transpose of the constraint Jacobian, but with better tracking performance for the end-effector task. The results in case studies 2 and 3 have confirmed the theoretical expectations. It has been argued also that this solution may prove more effective than other existing solutions that are based on the computation of the pseudoinverses of both the end-effector Jacobian and the matrix obtained by projecting the constraint Jacobian onto the null space of the end-effector Jacobian.

The schemes derived in this article provide an inverse kinematic solution for joint displacements and velocities. Future research efforts will be dedicated to extend the solution to joint acceleration in the same framework, as achieved already by Siciliano (1988) for nonredundant manipulators.

Acknowledgments

This work is based on research supported in part by Ministero dell'Università e della Ricerca Scientifica e Tecnologica under 40% funds and in part by Consiglio Nazionale delle Ricerche under contract 90.00400.PF67.

References

Baillieul, J. 1985 (St. Louis, March 25–28). Kinematic programming alternatives for redundant manipulators. *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*. Silver Spring, Md.: IEEE Computer Society Press, pp. 722–728.

Baillieul, J. 1986 (San Francisco, April 7–10). Avoiding obstacles and resolving kinematic redundancy. *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*. Washington, D.C.: IEEE Computer Society Press, pp. 1698–1704.

Baillieul, J. 1987 (Raleigh, N.C., March 31–April 3). A constraint oriented approach to inverse problems for kinematically redundant manipulators. *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*. Washington, D.C.: IEEE Computer Society Press, pp. 1827–1833.

Baillieul, J., Hollerbach, J. M., and Brockett, R. W. 1984 (Las Vegas, Dec. 12–14). Programming and control of kinematically redundant manipulators. *Proceedings of the 23rd IEEE Conference on Decision and Control*. New York, N.Y.: IEEE, pp. 768–774.

Baker, D. R., and Wampler, C. W. 1988. On the inverse kinematics of redundant manipulators. *Int. J. Robot. Res.* 7(2):3–21.

Balestrino, A., De Maria, G., and Sciavicco, L. 1984 (Budapest, July 2–6). Robust control of robotic manipulators. *Preprints of the 9th IFAC World Congress*, vol. 6. pp. 80–85.

Chang, P. H. 1987. A closed-form solution for inverse kinematics of robot manipulators with redundancy. *IEEE J. Robot. Automat.* RA-3(5):393–403.

Chiu, S. L. 1987 (Raleigh, N.C., March 31–April 3). Con-

trol of redundant manipulators for task compatibility. *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*. Washington, D.C.: IEEE Computer Society Press, pp. 1718–1724.

Corless, M. J., and Leitmann, G. 1981. Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamic systems. *IEEE Trans. Auto. Control* AC-26(5):1139–1144.

Das, H., Slotine, J.-J. E., and Sheridan, T. B. 1988 (Philadelphia, April 24–29). Inverse kinematic algorithms for redundant systems. *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*. Washington, D.C.: IEEE Computer Society Press, pp. 43–48.

De Maria, G., and Marino, R. 1985 (Tokyo, Sept. 9–10). A discrete algorithm for solving the inverse kinematic problem of robotic manipulators. *Proceedings of the 2nd International Conference on Advanced Robotics*. Tokyo: JIRA, pp. 275–282.

Egeland, O. 1987. Task-space tracking with redundant manipulators. *IEEE J. Robot. Automat.* RA-3(5):471–475.

Hollerbach, J. M., and Suh, K. C. 1985 (St. Louis, March 25–28). Redundancy resolution of manipulators through torque optimization. *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*. Silver Spring, Md.: IEEE Computer Society Press, pp. 1016–1022.

Kirćanski, M., and Vukobratović, M. 1984 (Udine, Italy, June 16–29). Trajectory planning for redundant manipulators in the presence of obstacles. *Preprints of the 5th CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators*, pp. 43–50.

Klein, C. A., and Blaho, B. E. 1987. Dexterity measures for the design and control of kinematically redundant manipulators. *Int. J. Robot. Res.* 6(2):72–83.

Klein, C. A., and Huang, C. H. 1983. Review of pseudoinverse control for use with kinematically redundant manipulators. *IEEE Trans. Sys. Man Cybernet.* SMC-13:245–250.

Liégeois, A. 1977. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Trans. Sys. Man Cybernet.* SMC-7:868–871.

Maciejewski, A. A., and Klein, C. A. 1987. Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *Int. J. Robot. Res.* 4(3):109–117.

Nakamura, Y., and Hanafusa, H. 1986. Inverse kinematic solutions with singularity robustness for robot manipulator control. *ASME J. Dynam. Sys. Measurement Control* 108(3):163–171.

Nakamura, Y., Hanafusa, H., and Yoshikawa, T. 1987. Task-priority based redundancy control of robot manipulators. *Int. J. Robot. Res.* 6(2):3–15.

Sciavicco, L., and Siciliano, B. 1987a (Raleigh, N.C., March 31–April 3). A dynamic solution to the inverse kinematic problem for redundant manipulators. *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*. Washington, D.C.: IEEE Computer Society Press, pp. 1081–1087.

Sciavicco, L., and Siciliano, B. 1987b (Versailles, Oct. 13–15). An inverse kinematic solution algorithm for dexterous redundant manipulators. *Proceedings of the 3rd International Conference on Advanced Robotics*. Bedford: IFS Publications Ltd., pp. 247–256.

Sciavicco, L., and Siciliano, B. 1988a (Salò, June 27–July 1). On the solution of inverse kinematics of redundant manipulators. *Preprints of the NATO Advanced Research Workshop: Robots with Redundancy*. New York, N.Y.: Springer-Verlag, in press.

Sciavicco, L., and Siciliano, B. 1988b. A solution algorithm to the inverse kinematic problem for redundant manipulators. *IEEE J. Robot. Automat.* RA-4(4):303–310.

Sciavicco, L., Siciliano, B., and Chiacchio, P. 1988 (Atlanta, June 15–17). On the use of redundancy in robot kinematic control. *Proceedings of the 1988 American Control Conference*. Madison, Wis.: Omnipress, pp. 1370–1375.

Siciliano, B. 1988 (Ljubljana, Sept. 19–21). Closed-loop computational schemes of robot inverse kinematics. *Proceedings of the International Meeting: Advances in Robot Kinematics*, pp. 113–121. Ljubljana Yugoslavia, Jožef Stefan Institute.

Slotine, J. J. E., and Yoerger, D. R. 1987. A rule-based inverse kinematics algorithm for redundant manipulators. *Int. J. Robot. Automat.* 2(2):86–89.

Takegaki, M., and Arimoto, S. 1981. A new feedback method for dynamic control of manipulators. *ASME J. Dynam. Sys. Measurement Control* 102(2):119–125.

Tsai, Y. T., and Orin, D. E. 1987. A strictly convergent real-time solution for inverse kinematics of robot manipulators. *J. Robot. Sys.* 4(4):477–501.

Wampler, C. W. 1986. Manipulator inverse kinematic solutions based on damped least-squares solutions. *IEEE Trans. Sys. Man Cybernet.* SMC-16(1):93–101.

Wampler, C. W. 1987 (Raleigh, N.C., March 31–April 3). Inverse kinematic functions for redundant manipulators. *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*. Washington, D.C.: IEEE Computer Society Press, pp. 610–617.

Whitney, D. E. 1969. Resolved motion rate control of manipulators and human prostheses. *IEEE Trans. Man-Machine Sys.* MMS-10:47–53.

Wolovich, W. A., and Elliott, H. 1984 (Las Vegas, Dec. 12–14). A computational technique for inverse kinematics. *Proceedings of the 23rd IEEE Conference on Decision and Control*. New York: IEEE, pp. 1359–1363.

Yoshikawa, T. 1985a. Dynamic manipulability of robot manipulators. *J. Robot. Sys.* 2(1):113–124.

Yoshikawa, T. 1985b. Manipulability of robotic mechanisms. *Int. J. Robot. Res.* 4(2):3–9.