WA8 11:15

Efficient Dynamic Resolution of Robot Redundancy

A. De Luca, G. Oriolo

Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Eudossiana 18, 00184 Roma, Italy

Abstract

A new method for solving redundancy in robot arms by locally optimizing a dynamic criterion is presented. Optimization is performed by means of a joint decomposition technique, resulting in a particularly efficient computational scheme which is feasible for real-time control. The technique is illustrated obtaining an alternate and simpler expression in the case of torque minimization. It is known that such a local scheme may exhibit unstable behavior, due to the build-up of high joint velocities that require in turn extremely large torques. Physical intuition is exploited to propose other dynamic criteria, whose minimization generates satisfactory torque profiles. Comparative simulations on a three-link planar arm show the effectiveness of the proposed method.

1. Introduction

Kinematic redundancy in robot manipulators allows to tackle interesting motion planning issues that include arm singularities prevention [1], limitation of joint ranges [2], avoidance of obstacles in the cartesian space [3,4], or task-dependent dexterity enhancement [5,6]. As a counterpart, programming joint-space trajectories $\mathbf{q}(t)$ that realize a given end-effector path $\mathbf{p}(t)$ becomes a more complex problem, due to the non-uniqueness of inverse kinematic solutions to

$$\dot{\mathbf{p}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}, \tag{1}$$

in which the manipulator Jacobian $\mathbf{J}$ is an $m \times n$ matrix, with $m < n$.

For optimization purposes, arm configurations are usually generated by specifying additional velocities $(\mathbf{I}-\mathbf{J}^{\dagger}\mathbf{J})\mathbf{v}$ in the null-space of the Jacobian and choosing $\mathbf{v}$ in the negative-gradient direction of a configuration dependent criterion $L(\mathbf{q})$ to be minimized

$$\dot{\mathbf{q}} = \mathbf{J}^{\dagger}\dot{\mathbf{p}} - \alpha(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J})\nabla_{\mathbf{q}}L(\mathbf{q}). \tag{2}$$

$\mathbf{J}^{\dagger}$ is the (unique) pseudoinverse of $\mathbf{J}$ [7], which in the full row-rank case takes on the expression $\mathbf{J}^{\dagger} = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}$, while α is a scalar stepsize.

Purely kinematic schemes may result in poor dynamic performance, since unrealistic torques may be needed to track the given end-effector trajectory [8,9]. This is quite a serious drawback: unpredictability of the torque or energy demand even over simple and smooth cartesian trajectories may limit the benefits of a redundant arm mechanical design by requiring oversized actuators. A number of methods have been devised in order to include at this stage robot dynamics

$$\mathbf{u} = \mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}). \tag{3}$$

Here, $\mathbf{u}$ are the joint generalized forces, $\mathbf{B}(\mathbf{q})$ is the manipulator inertia matrix and $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})$ collects Coriolis, centrifugal, and gravity terms. For redundancy resolution, the kinematic constraint should be considered at the acceleration level

$$\ddot{\mathbf{p}} = \mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}. \tag{4}$$

All the solutions of (4) can be set in the form

$$\ddot{\mathbf{q}} = \mathbf{J}^{\dagger}(\ddot{\mathbf{p}} - \dot{\mathbf{J}}\dot{\mathbf{q}}) + (\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J})\mathbf{a}, \tag{5}$$

with free choice for the null-space acceleration $\mathbf{a}$.

Khatib [10] proposed to set $\mathbf{a} = \mathbf{0}$, together with the use of an inertia-weighted pseudoinversion

$$\ddot{\mathbf{q}} = \mathbf{J}_B{}^{\dagger}(\ddot{\mathbf{p}} - \dot{\mathbf{J}}\dot{\mathbf{q}}), \tag{6}$$

thus providing local minimization of the weighted norm of joint accelerations. Though appealing for its simplicity, the effects on torque loading are not completely clear since nonlinear velocity and gravity couplings are neglected.

Explicit minimization of joint torque norm was considered by Hollerbach and Suh [11]. According to the torque-acceleration relationship expressed by the dynamic model, this can be achieved by choosing

$$\mathbf{a} = -\left[\mathbf{B}(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J})\right]^{\dagger}\tilde{\mathbf{u}}, \tag{7}$$

where

$$\tilde{\mathbf{u}} = \mathbf{B}\mathbf{J}^{\dagger}(\ddot{\mathbf{p}} - \dot{\mathbf{J}}\dot{\mathbf{q}}) + \mathbf{n}. \tag{8}$$

The minimum-norm torque is then obtained as

$$\mathbf{u} = \tilde{\mathbf{u}} - \mathbf{B}\left[\mathbf{B}(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J})\right]^{\dagger}\tilde{\mathbf{u}}. \tag{9}$$

It has been shown that this local optimization approach may still become unfeasible, due to the build-up of high joint velocities which generate in turn extremely large torques. This instability phenomenon appears to be dramatic for long trajectories and occurs independently from the crossing of kinematic singularities. Moreover, computation of (9) is quite intensive because of the pseudoinversion of a rank-deficient matrix.

When the end-effector trajectory to be followed is known in advance, a natural way for overcoming these difficulties is to adopt a global point of view, i.e. to formulate an optimization problem with an integral performance index which quantifies the overall torque demand along the path [12]. As expected, this method outperforms the corresponding local scheme most of the times, indicating the intrinsic limitations of local optimization. However, the global approach is computationally intensive requiring the numerical solution of a two-point boundary value problem. Besides, there are classes of robotic tasks for which an on-line solution is not only desirable, but also strictly required — e.g. when a sensor-driven strategy of motion planning is pursued.

This situation motivates further research for a local optimization scheme that should possess two main characteristics:

(i) a stable behavior when executing both short and long trajectories, with the least possible torque consumption;

(ii) a reduced amount of computation, so to lend itself to efficient real-time implementation.

Along these lines, a strategy has been proposed by Hirose and Ma [13] with the Redundancy Decomposition Control (RDC). The joint variables are decomposed into two groups $(\mathbf{q}_a, \mathbf{q}_b)$ such that, in the associated partitioned Jacobian $(\mathbf{J}_a, \mathbf{J}_b)$, the $m \times m$ block $\mathbf{J}_a$ is nonsingular. Then, joint accelerations are evaluated as

$$\ddot{\mathbf{q}} = \begin{bmatrix} \mathbf{J}_a^{-1}(\ddot{\mathbf{p}} - \dot{\mathbf{J}}\dot{\mathbf{q}}) \\ \mathbf{0} \end{bmatrix}, \tag{10}$$

and used in (3) to generate the driving torques. At each instant this procedure is repeated $\binom{n}{m}$ times, exploring all acceleration partitions and selecting the one that gives the smallest norm for the torque. Remarkably, RDC does not show instabilities for long trajectories in comparison with local torque minimization; yet, at each point the true local minimum torque is not provided.

In this paper, an efficient and stable method for redundancy resolution is presented. Optimization of a general dynamic criterion is performed by means of the joint decomposition technique recently introduced in [14]. The technique is reviewed and then used to provide an alternate expression for the minimum-norm torques generated by the null-space method. This expression is equivalent to (9), but easier to be computed. Physical intuition is then exploited to propose a new class of dynamic criteria which are weighted combinations of torque and acceleration terms. Local minimization of a proper criterion in this class generates satisfactory torque profiles. As special cases, both the methods of Khatib [10] and of Hollerbach and Suh [11] are recovered. Optimization is again carried out through the decomposition approach, resulting in particularly efficient computational schemes feasible for real-time control. Finally, comparative simulations with a three-link planar arm show the good performance of the method.

2. Optimization Via Joint Decomposition

Part of the inefficiency in optimal redundancy resolution schemes stems from the fact that pseudoinversion and projection operators are used. Based on the observation that the actual number of free variables for a redundant robot is equal to its degree of redundancy $n-m$, the optimization process can be performed in a reduced space. De Luca and Oriolo [14] have used this concept at the velocity level to devise a *reduced* gradient method for optimization of a configuration-dependent criterion. The approach presented here is the transposition at acceleration level of the same technique. A preliminary assumption will be needed in the following. Let $f(\mathbf{q})$ denote the direct kinematics of the arm.

Nondegeneracy Assumption. At every point $\mathbf{p}$ on the path, there exists a $\mathbf{q} \in \mathbb{R}^n$ such that $f(\mathbf{q}) = \mathbf{p}$, and for which a partition can be found

$$\mathbf{q} = (\mathbf{q}_a, \mathbf{q}_b), \qquad \mathbf{q}_a \in \mathbb{R}^m,\ \mathbf{q}_b \in \mathbb{R}^{n-m},$$

yielding a nonsingular matrix $\mathbf{J}_a(\mathbf{q}) \triangleq \partial f/\partial \mathbf{q}_a$. ∎

This assumption restricts the admissible end-effector locations, discarding those for which the Jacobian necessarily looses rank. According to the partition of $\mathbf{q}$, the kinematic constraint (4) can be solved in terms of the acceleration $\ddot{\mathbf{q}}_a$ as

$$\ddot{\mathbf{q}}_a = \mathbf{J}_a^{-1}\ddot{\mathbf{x}} - \mathbf{J}_a^{-1}\mathbf{J}_b\ddot{\mathbf{q}}_b, \tag{11}$$

where $\mathbf{J}_b(\mathbf{q}) \triangleq \partial f/\partial \mathbf{q}_b$ is a $m \times (n-m)$ matrix and $\ddot{\mathbf{x}} \triangleq \ddot{\mathbf{p}} - \dot{\mathbf{J}}\dot{\mathbf{q}}$ is the modified cartesian acceleration. It should be stressed that no pseudoinversion is used in this derivation.

The choice of acceleration $\ddot{\mathbf{q}}_b$ in (11) is free and may thus be used for optimization purposes. Note that RDC method sets $\ddot{\mathbf{q}}_b = \mathbf{0}$ in (11), thus giving up the benefits of optimizing torque or any other dynamic criterion, except from a purely combinatorial point of view.

Let a general dynamic criterion $L(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ be given, whose minimization should guide the redundancy resolution scheme. If local optimization is performed, the state $(\mathbf{q}, \dot{\mathbf{q}})$ of the robot arm is assigned at each instant, and L depends only on the acceleration $\ddot{\mathbf{q}}$. Then, a necessary condition for local optimality is obtained by setting to zero the projected gradient of the cost function, i.e. $(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J})\nabla_{\ddot{\mathbf{q}}} L(\ddot{\mathbf{q}})$. Alternatively, a more compact condition is derived as follows. Using the linear relationship (11), the criterion can be rewritten as a function of $\ddot{\mathbf{q}}_b$ alone, denoted by $L'(\ddot{\mathbf{q}}_b)$.

Applying the chain rule of derivation, the *reduced* gradient is introduced as [15]

$$\nabla_{\ddot{\mathbf{q}}_b} L'(\ddot{\mathbf{q}}_b) = \begin{bmatrix} -\mathbf{J}_R^T & \mathbf{I} \end{bmatrix} \nabla_{\ddot{\mathbf{q}}} L, \qquad (12)$$

where $\mathbf{J}_R \triangleq \mathbf{J}_a^{-1}\mathbf{J}_b$. Therefore, optimality requires satisfaction of the following set of $n-m$ equations

$$\begin{bmatrix} -\mathbf{J}_R^T & \mathbf{I} \end{bmatrix} \nabla_{\ddot{\mathbf{q}}} L = \mathbf{0}. \qquad (13)$$

When solving redundancy at a first-order level, a similar set of equations can be derived [16]. In that case, due to the non-existence of a general closed-form solution to inverse kinematics, a numerical technique has to be used for obtaining joint displacements as in [14]. However, if the criterion L is a positive-definite quadratic function of joint accelerations, equations (13) and (11) provide necessary and sufficient conditions for optimality which can be explicitly solved in closed-form.

The problem of local minimization of torque norm considered by Hollerbach and Suh will be next used to illustrate in detail the joint decomposition technique. In this case, the quadratic criterion to be minimized is $L_0 = \frac{1}{2}\mathbf{u}^T\mathbf{u}$. Using (3) and (12), condition (13) yields

$$\begin{bmatrix} -\mathbf{J}_R^T & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{B}^2\ddot{\mathbf{q}} + \mathbf{B}\mathbf{n} \end{bmatrix} = \mathbf{0}. \qquad (14)$$

This equation can be solved for $\ddot{\mathbf{q}}_b$, replacing for $\ddot{\mathbf{q}}_a$ its expression (11); to this aim, the term $\mathbf{B}^2\ddot{\mathbf{q}}$ has to be written in a partitioned form. Let

$$\mathbf{W}_0(\mathbf{q}) \triangleq \mathbf{B}^2(\mathbf{q}) \qquad (15)$$

and decompose $\mathbf{W}_0$ as $(\mathbf{W}_{0a}, \mathbf{W}_{0b})$, according to the partition of the acceleration vector. Thus,

$$\begin{bmatrix} -\mathbf{J}_R^T & \mathbf{I} \end{bmatrix} \begin{bmatrix} (\mathbf{W}_{0b} - \mathbf{W}_{0a}\mathbf{J}_R)\ddot{\mathbf{q}}_b + \mathbf{W}_{0a}\mathbf{J}_a^{-1}\ddot{\mathbf{x}} + \mathbf{B}\mathbf{n} \end{bmatrix} = \mathbf{0}. \qquad (16)$$

Finally, denoting the $(n-m)\times(n-m)$ coefficient matrix of $\ddot{\mathbf{q}}_b$ by

$$\mathbf{Y}_0(\mathbf{q}) \triangleq \begin{bmatrix} -\mathbf{J}_R^T & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{W}_{0b} - \mathbf{W}_{0a}\mathbf{J}_R \end{bmatrix} \qquad (17)$$

the reduced acceleration providing the optimal solution is

$$\ddot{\mathbf{q}}_b = -\mathbf{Y}_0^{-1} \begin{bmatrix} -\mathbf{J}_R^T & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{B}\,\mathbf{n} + \mathbf{W}_{0a}\mathbf{J}_a^{-1}\ddot{\mathbf{x}} \end{bmatrix}. \qquad (18)$$

The remaining acceleration components $\ddot{\mathbf{q}}_a$ are obtained substituting (18) into (11), while the resulting torques are evaluated by backsubstitution of $\ddot{\mathbf{q}}$ into (3).

The computational burden for deriving the optimal torques via joint decomposition can be summarized in the *(i)* inversion of the $m\times m$ matrix $\mathbf{J}_a$, and *(ii)* inversion of the $(n-m)\times(n-m)$ matrix $\mathbf{Y}_0$. Note that in the special case of $n-m=1$, i.e. a single degree of redundancy, $\mathbf{Y}_0$ reduces to a scalar. This should be compared with the requirements of the original torque expression (9), namely *(i)* pseudoinversion of the $m\times n$ matrix $\mathbf{J}$, and *(ii)* pseudoinversion of the $n\times n$ matrix $\mathbf{B}(\mathbf{I}-\mathbf{J}^\dagger\mathbf{J})$. In the assumed case of full rank Jacobian, while $\mathbf{J}^\dagger$ simply accounts in the $m\times m$ inversion of $\mathbf{J}\mathbf{J}^T$, the second pseudoinversion must be performed through a Singular Value Decomposition, being $(\mathbf{I}-\mathbf{J}^\dagger\mathbf{J})$ of rank $n-m$ only. This is a rather delicate and computationally demanding operation. Moreover, no major simplifications are obtained in case of single degree of redundancy. Indeed, the two methods produce the same torque values; however, as a result of the previous considerations, the decomposition approach appears to be superior.

Two remarks are in order. The partition of $\mathbf{q}$ needs not to be held constant over the cartesian path; in fact, the basic scheme can be improved from the numerical point of view by choosing at each step the partition which gives the best conditioned $\mathbf{J}_a$ matrix. Moreover, for accurate tracking of the cartesian path, a more robust version of the scheme is obtained by adding to $\ddot{\mathbf{x}}$ the usual error correcting term $\mathbf{K}_p(\mathbf{p}-f(\mathbf{q}))+\mathbf{K}_d(\dot{\mathbf{p}}-\mathbf{J}\dot{\mathbf{q}})$ [13,17].

3. A Stable Optimization Method

The introduction of a redundancy resolution scheme which achieves local minimization of joint torques is motivated by the desire of reducing actuator demands. In its pure form this strategy proves to be uneffective and even disastrous when a relatively long trajectory is to be followed [11]. Explosion of torque requirements at a given point results from the growth of large joint-velocities, coupled with the need of keeping the end-effector on the assigned path. On the other hand, for short cartesian trajectories this local solution is not only satisfactory, but also very close to the global optimal profile [12]. Hence, a local method should be flexible enough to cope with potential instabilities in the long run while taking advantage of the minimum torque solution for short movements: a unified solution for both situations is highly desirable. To this aim, it is worth recalling that local minimization of the acceleration norm [10], both unweighted and inertia-weighted, never produces unstable behavior.

These considerations lead to the definition of a class of dynamic criteria which should be able to deal with instabilities, possibly limiting the actuator torque demand. One simple candidate class is given by

$$L_1 = \frac{1}{2}\begin{bmatrix} k_1\ddot{\mathbf{q}}^T\ddot{\mathbf{q}} + k_2\mathbf{u}^T\mathbf{u} \end{bmatrix}, \qquad (19)$$

i.e. a combination of acceleration and torque norms, with non-negative k_1 and k_2. Torque minimization is recovered for $k_1 = 0$, while for $k_2 = 0$ the unweighted acceleration scheme is obtained. Being the dynamic criterion (19) always a quadratic positive-definite function of the joint accelerations, exact local minimization is possible in closed-form. The joint decomposition approach provides an efficient scheme for computing the optimal torque. Defining matrix $\mathbf{W}_1$ as

$$\mathbf{W}_1(\mathbf{q}) \triangleq k_1\mathbf{I} + k_2\mathbf{B}^2(\mathbf{q}) \qquad (20)$$

and proceeding like in the previous section, the solution is found as

$$\ddot{\mathbf{q}}_b = -\mathbf{Y}_1^{-1}\left[-\mathbf{J}_R^T \quad \mathbf{I}\right]\left[k_2\mathbf{B}\,\mathbf{n} + \mathbf{W}_{1a}\mathbf{J}_c^{-1}\ddot{\mathbf{x}}\right], \qquad (21)$$

in which matrix $\mathbf{Y}_1$ takes the form (17), with $\mathbf{W}_1$ replacing $\mathbf{W}_0$. As before, $\ddot{\mathbf{q}}_a$ is given by backsubstitution of (21) in (11).

The critical issue in the use of criterion L_1 is the proper selection of weighting coefficients k_1, k_2. This is a common problem to be faced for all combined optimization criteria [15]. For long trajectories, a larger k_1 is expected to prevent high accelerations that would eventually lead to unacceptable torques. In case of short movements, the reverse choice seems more convenient for keeping low torque profiles. Just a rough qualitative information is needed at this point for the on-line synthesis of joint trajectories, while the specific characteristics of the future commanded cartesian motion do not come into play. The flexibility gained with (19) justifies its introduction.

However, a closer examination of the above criterion reveals a possible shortcoming in the way joint accelerations are treated. Using (19), the goal of avoiding velocity build-up is achieved in a rather conservative way. In fact, a large joint acceleration is not always dangerous for stability: in particular, it is acceptable as long as it opposes in sign the corresponding velocity. In the case of self-motions, Maciejewski [9] recognized this to be similar to a stabilizing negative feedback action. Motivated by this, the following modified dynamic criterion is proposed:

$$L_2 = k_1\dot{\mathbf{q}}^T\ddot{\mathbf{q}} + \frac{1}{2}k_2\mathbf{u}^T\mathbf{u}. \qquad (22)$$

For each joint, the first term possibly allows a negative acceleration when its velocity is positive, or viceversa. Moreover, the higher the velocity, the larger is the effective weight attributed to this stabilizing term. Being L_2 still quadratic in $\ddot{\mathbf{q}}$, the decomposition approach to minimization provides the optimal value for the redundant accelerations as

$$\ddot{\mathbf{q}}_b = -\mathbf{Y}_2^{-1}\left[-\mathbf{J}_R^T \quad \mathbf{I}\right]\left[k_1\dot{\mathbf{q}} + k_2\mathbf{B}\,\mathbf{n} + \mathbf{W}_{2a}\mathbf{J}_a^{-1}\ddot{\mathbf{x}}\right], \qquad (23)$$

where

$$\mathbf{W}_2(\mathbf{q}) \triangleq k_2\mathbf{B}^2(\mathbf{q}), \qquad (24)$$

and $\mathbf{Y}_2$ is defined as in (17), having replaced $\mathbf{W}_0$ with $\mathbf{W}_2$.

4. Simulation Results

The dynamic schemes of local redundancy resolution have been applied to a 3R-planar robot arm without gravity. Links have unit length, mass of 10.0 kg, and are modeled as uniform thin rods. Absolute (i.e. referred to the x axis) joint variables are adopted and the torques are defined accordingly. Actuators are assumed to be located at the base. If the motors were positioned at the joints, a constant weighting matrix would be needed to perform minimization of the actual torques. The kinematic task is specified as a straight-line cartesian path for the end-effector, with zero initial and final velocity. Joints are supposed to start at rest from a given initial arm configuration. End-effector acceleration is of the bang-bang type, with equal acceleration and deceleration in the first and the last half, respectively. A second-order Runge-Kutta integration method is used to obtain joint positions and velocities from the accelerations computed by the dynamic resolution scheme. An integration step of 1 msec is accurate enough to avoid the use of closed-loop correction.

Results for one short and two long movements are reported for comparing the performance of the following local dynamic resolution schemes:

- Torque (L_0) optimization using (18), denoted by T;
- Redundancy Decomposition Control, or RDC;
- Combined torque/acceleration (L_1) minimization using (21), denoted by C1;
- Modified torque/acceleration (L_2) minimization using (23), denoted by C2.

In the first simulation the arm starts from $\mathbf{q} = (-45°, 90°, -45°)$ and the end-effector moves by 0.2 m in both x and y directions, with unitary magnitude acceleration. Figure 1 shows the torques generated by RDC and by the combined torque/acceleration minimization C1 (scale is in Nm). Being the movement relatively short, the weights in L_1 were chosen as $k_1 = k_2 = 1$. Torque profiles of C1 are practically the same found by pure torque minimization T, which are not reported. As such, they are also very close to global optimality. The better performance of C1 method over RDC is evident, particularly on joint 1 torque. The reason for this is understood from Fig. 2, where arm configurations are reported at equal temporal spacings. While C1 smoothly distributes motion over all joints, RDC keeps the first link at rest requiring for this a large amount of torque.

The second simulation starts from the same arm posture, but a longer trajectory of 0.83 m in x and y directions is traveled, again with unitary acceleration in magnitude. The torque profiles for the four methods are reported in Fig. 3. In this case, C1 is executed with $k_1 = 100$ and $k_2 = 1$, while C2 uses $k_1 = 10000$ and $k_2 = 1$. This is a typical case where pure torque minimization leads to unstable behavior — plots of T are truncated in Fig. 3. RDC and C1 results look quite similar, although C1 is superior in reducing torque demand of proximal links. The torque profiles of C2 method approach a bang-bang behavior, initially requiring more effort than RDC and C1 but later settling on a regular plateau and avoiding spikes. This interesting behavior is largely dependent on the very large value k_1 chosen for C2. Besides visual appreciation, a compact way to compare the overall performance of the various local algorithms is through the evaluation of the global torque criterion $\int_0^{t_f} \mathbf{u}^T \mathbf{u} dt$. In this case, the values 460.6, 247.5, and 906.3 are obtained for the RDC, C1, and C2 methods, respectively, indicating C1 as the best scheme. On the other hand, the C2 method produces a final arm posture with practically zero residual kinetic energy. The arm configurations shown in Figs. 4–5 illustrate also the dramatic movement generated by the T method.

Finally, in a third simulation a 1.78 m long path in both x and y directions is considered. The arm starts from $\mathbf{q} = (-180°, -90°, -90°)$, with an acceleration of 3 m/sec^2 along the path. Again T is not feasible, while the results for C1 and C2 are quite similar. Figure 6 shows the torque norm for the stable methods RDC and C2. Here, C2 uses $k_1 = 100$ and $k_2 = 1$. The global torque criterion yields the values 3027, 1857, and 1808 for the RDC, C1, and C2 methods, respectively. From Fig. 7 it can be seen that C2 produces a whipping-like motion, using the natural dynamics of the movement to reduce torque demand.

5. Conclusions

A new method for solving robot redundancy by optimization of a general dynamic criterion has been presented. By using a joint decomposition approach, optimization is performed in the reduced space of redundant joint variables, so that no matrix pseudoinversion is needed. The proposed scheme is efficient and feasible for on-line implementation. Moreover, two different weighted combinations of torque and acceleration terms have been introduced as local criterions. In spite of their local nature, the resulting methods exhibit stable behavior even on long trajectories, performing better than other existing nonglobal techniques. Moreover, the possibility of actually minimizing overall torque demand is preserved on shorter paths. Future work will focus on the automatic tuning and updating of the weights in the cost criterion, relying on a deeper insight into the physics of the problem.

Acknowledgements

This work was partially supported by CNR *Progetto Finalizzato Robotica*, Linea 4.3.

References

[1] T. Yoshikawa, "Manipulability of robotics mechanisms", *Int. J. Robotics Res.*, vol. 4, no. 2, pp. 3–9, 1985.

[2] A. Liégeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms", *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, no. 12, pp. 868–871, 1977.

[3] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments", *Int. J. Robotics Res.*, vol. 4, no. 3, pp. 109–117, 1985.

[4] J. Baillieul, "Avoiding obstacles and resolving kinematic redundancy", in *Proc. IEEE Int. Conf. on Robotics and Automation* (San Francisco, CA, Apr. 7–10, 1986), pp. 1698–1704.

[5] S. L. Chiu, "Task compatibility of manipulators postures", *Int. J. Robotics Res.*, vol. 7, no. 5, pp. 13–21, 1988.

[6] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators", *Int. J. Robotics Res.*, vol. 6, no. 2, pp. 3–15, 1987.

[7] T. L. Boullion and P. L. Odell, *Generalized Inverse Matrices*. New York, NY: Wiley Interscience, 1971.

[8] C. A. Klein and A. I. Chirco, "Dynamic simulation of a kinematically redundant manipulator system", *J. Robotic Systems*, vol. 4, no. 1, pp. 5–23, 1987.

[9] A. A. Maciejewski, "Kinetic limitations in the use of redundancy in robotic manipulators", in *Proc. IEEE Int. Conf. on Robotics and Automation* (Scottsdale, AZ, May 14–19, 1989), pp. 113–118.

[10] O. Khatib, "Dynamic control of manipulators in operational space", in *Proc. 6th IFToMM Congress on Theory of Machines and Mechanisms* (New Dehli, India, 1983), pp. 1123–1131.

[11] J. M. Hollerbach and K. C. Suh, "Redundancy resolution of manipulators through torque optimization", *IEEE J. Robotics Automat.*, vol. RA-3, no. 4, pp. 308–316, 1987.

[12] J. M. Hollerbach and K. C. Suh, "Local versus global torque optimization of redundant manipulators", in *Proc. IEEE Int. Conf. on Robotics and Automation* (Raleigh, NC, Mar. 31–Apr. 3, 1987), pp. 619–624.

[13] S. Hirose and S. Ma, "Redundancy decomposition control for multi-joint manipulator", in *Proc. IEEE Int. Conf. on Robotics and Automation* (Scottsdale, AZ, May 14–19, 1989), pp. 119-124.

[14] A. De Luca and G. Oriolo, "The reduced gradient method for solving redundancy in robot arms", *11th IFAC World Congress* (Tallinn, USSR, Aug. 13–17, 1990).

[15] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 1984.

[16] P. H. Chang, "A closed-form solution for inverse kinematics of robot manipulators with redundancy", *IEEE J. Robotics Automat.*, vol. RA-3, no. 5, pp. 393–403, 1987.

[17] L. Sciavicco and B. Siciliano, "A computational technique for solving robot end-effector trajectories into joint trajectories", in *Proc. American Control Conf.* (Atlanta, GA, June 15–17, 1988), pp. 535–536.

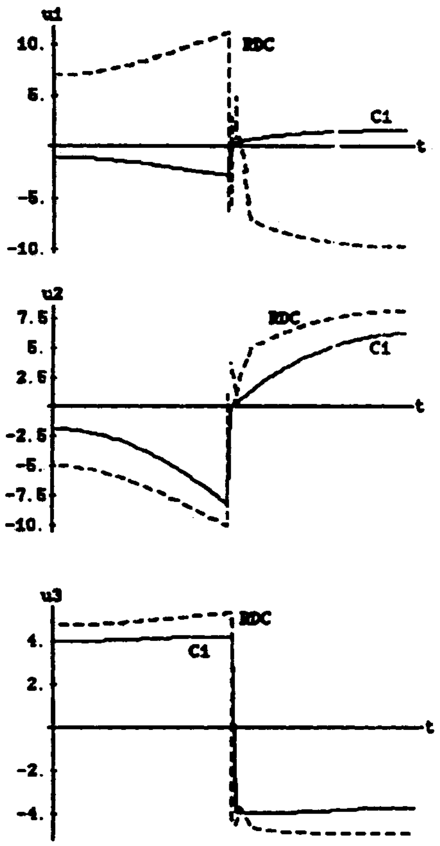

Fig. 1 – Short movement. Comparison of joint torques for C1 and RDC.

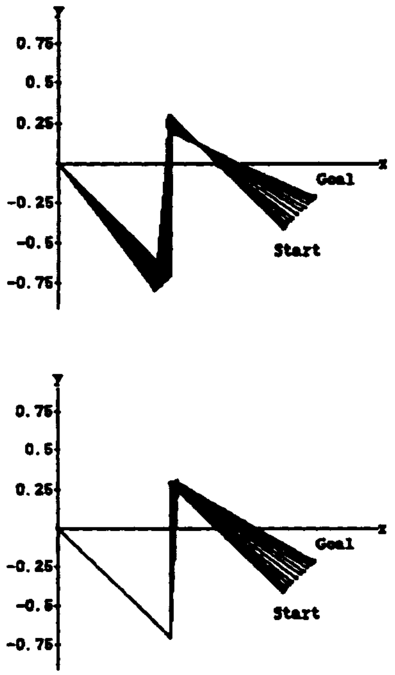

Fig. 2 – Short movement. Arm configurations for C1 (above) and RDC (below).

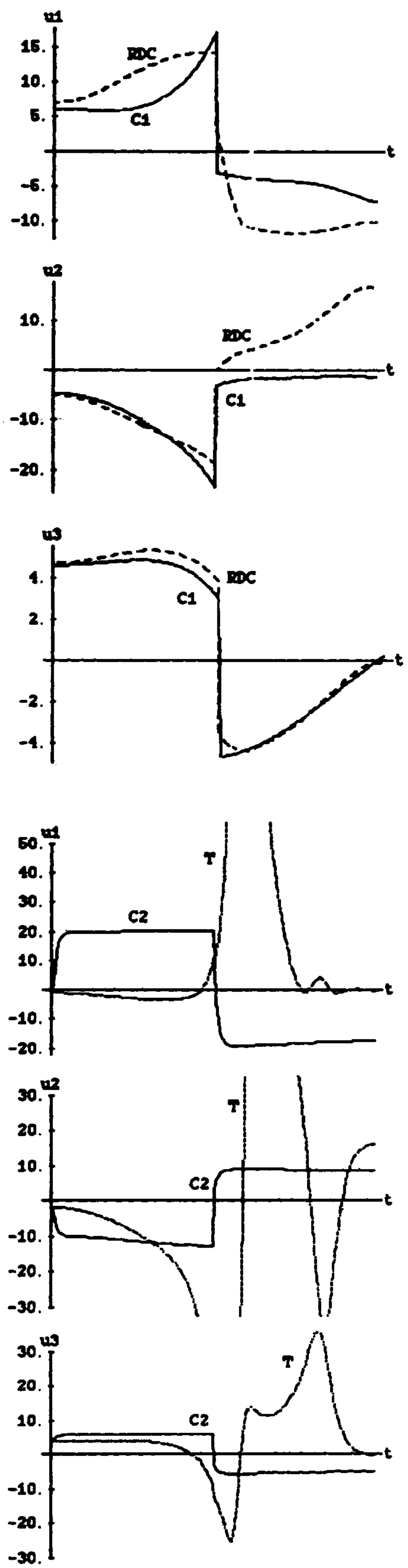

Fig. 3 – Long movement 1. Comparison of joint torques for C1 and RDC (above); T and C2 (below).

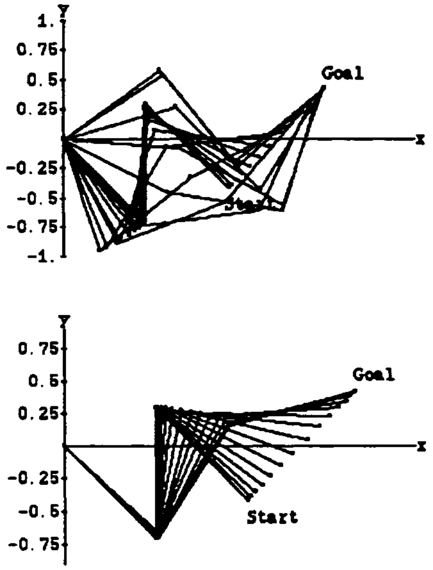

Fig. 4 – Long movement 1. Arm configurations for T (above) and RDC (below).

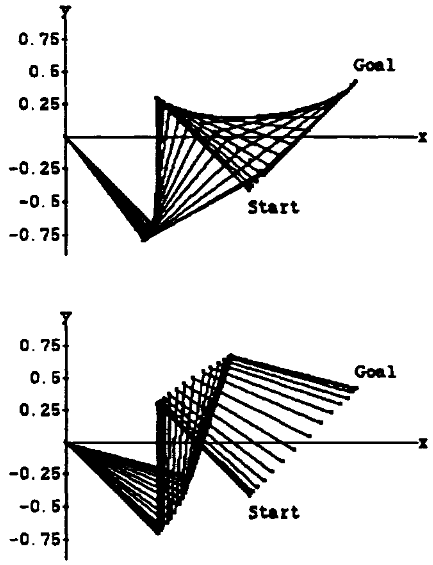

Fig. 5 – Long movement 1. Arm configurations for C1 (above) and C2 (below).

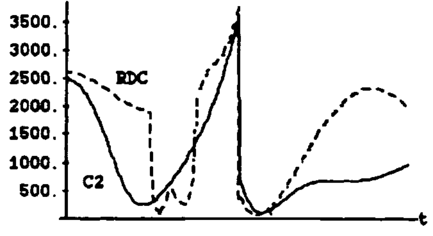

Fig. 6 – Long movement 2. Comparison of torque norm: C2 and RDC.

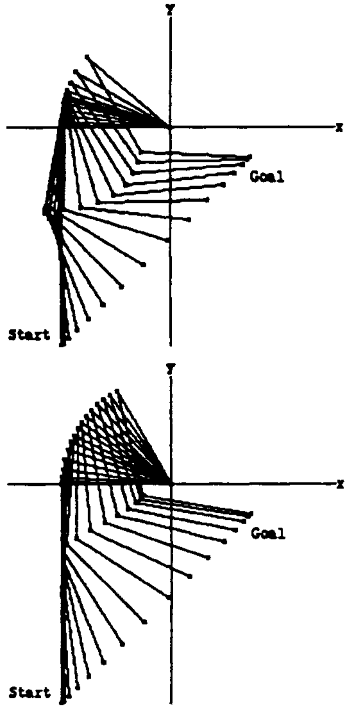

Fig. 7 – Long movement 2. Arm configurations for C2 (above) and RDC (below).