*Journal of Intelligent and Robotic Systems* **14**: 43–68, 1995.



# Overview of Damped Least-Squares Methods for Inverse Kinematics of Robot Manipulators

ARATI S. DEO and IAN D. WALKER
*Department of Electrical and Computer Engineering, Rice University, Houston TX 77251, U.S.A.*

(Received: 28 December 1993)

**Abstract.** In this paper, we present a tutorial report of the literature on the damped-least squares method which has been used for computing velocity inverse kinematics of robotic manipulators. This is a local optimization method that can prevent infeasible joint velocities near singular configurations by using a damping factor to control the norm of the joint velocity vector. However, the exactness of the inverse kinematic solution has to be sacrificed in order to achieve feasibility.

The damping factor is an important parameter in this technique since it determines the trade-off between the accuracy and feasibility of the inverse kinematic solution. Various methods that have been proposed to compute an appropriate damping factor are described.

Redundant manipulators, possessing extra degrees of freedom, afford more choice of inverse kinematic solutions than do non-redundant ones. The damped least-squares method has been used in conjunction with redundancy resolution schemes to compute feasible joint velocities for redundant arms while performing an additional subtask. We outline the different techniques that have been proposed to achieve this objective. In addition, we introduce an iterative method to compute the optimal damping factor for one of the redundancy resolution techniques.

**Key words.** Robotics, kinematics, redundancy, damped least squares singularity robustness.

## 1. Introduction

Inverse kinematics of robotic manipulators is an important problem that needs to be solved for trajectory planning and dynamic computations. For most manipulators, a closed-form inverse kinematic function does not exist at the position level. As a result, inverse kinematics is usually carried out at the velocity or acceleration level. In particular, inverse kinematics of redundant manipulators presents a challenging problem due to the lack of uniqueness of solution. Extensive work has been carried out in this field and is continuing to date [30, 33, 39].

In this work, we will be dealing with the computation of joint velocities of a manipulator in the neighborhood of singular configurations, i.e. in regions of the workspace where the joint velocities tend to become infeasibly high, even for very small end-effector motions. The damped least-squares method can be used in this situation to obtain feasible joint motions, at each iteration. This method uses an instantaneous trade-off between the accuracy and feasibility of the inverse kinematic solution to prevent the joint velocities from becoming excessively high. The trade-off is quantified by a factor known in the robotics literature as the damping factor.

In this paper, we review the different methods that have been proposed to determine a suitable damping factor, including iterative methods that can be used to compute an optimal damping factor. In addition, we investigate the techniques through which damped least-squares has been incorporated into redundancy resolution methods to perform additional subtasks with feasible joint velocities using redundant arms. We also introduce a method to compute an optimal damping factor for the case when gradient projection is used with damped least-squares.

The methods described in this paper have been simulated on revolute-jointed planar arms with one to four links [3, 7, 21, 22, 25, 28, 36, 37] as well as spatial arms like the 6 degree of freedom PUMA manipulator [7, 21, 22] and reconfigurable modular manipulator systems [13]. Simulation of redundancy resolution with damped least-squares has been carried out on 3-link and 4-link planar manipulators [8, 24, 32] as well as on the PUMA which was made redundant by adding, in one study, an extra roll joint in the shoulder [9] and in another, by imparting translational motion to the base [8]. Weighted damped least-squares has been applied in experiments to the 5 d.o.f. ABB Trallfa TR 400 robot [4, 5].

This paper is organized as follows. The next section outlines the velocity inverse kinematics problem for robotic manipulators and describes how the damped least-squares technique is useful near singular configurations (also see Appendix A). Various methods to compute a damping factor are enumerated in Section 3. Section 4 describes the redundancy resolution techniques that have been used with damped least-squares. Some simulations are presented in Section 5 and conclusions follow in Section 6.

## 2. Velocity Inverse Kinematics

The resolved-motion rate-control scheme utilizes the following instantaneous relationship between the end-effector joint velocities [38].

$$\dot{\mathbf{x}} = J\dot{\boldsymbol{\theta}}. \tag{1}$$

Here $\dot{\mathbf{x}} \in \mathbb{R}^m$ is the end-effector velocity vector while $\dot{\boldsymbol{\theta}} \in \mathbb{R}^n$ is the vector representing the joint velocities. $J(\boldsymbol{\theta})$ is the manipulator Jacobian matrix with dimensions $m \times n$. For nonredundant manipulators, $m = n$ while for redundant manipulators $m < n$. One way of computing the inverse kinematics of a manipulator at the velocity level from the above relation is [38]

$$\text{Nonredundant manipulator}: \ \dot{\boldsymbol{\theta}} = J^{-1}\dot{\mathbf{x}},$$

$$\text{Redundant manipulator}: \ \dot{\boldsymbol{\theta}} = J^{+}\dot{\mathbf{x}} = J^T(JJ^T)^{-1}\dot{\mathbf{x}},$$

where it is assumed that rank($J$) $= m$. Here $J^{-1}$ is the inverse while $J^{+}$ is the pseudo-inverse of the manipulator Jacobian. For redundant arms, the solution $J^{+}\dot{\mathbf{x}}$ has minimum norm among the infinite number of joint velocity vectors that satisfy Equation (1) (e.g. see [27]). In the following sections, all discussion related to the pseudo-inverse also applies to the inverse in case of nonredundant arms since the pseudo-inverse and inverse are equivalent for a square Jacobian (with full rank). The pseudoinverse solution ($J^{+}\dot{\mathbf{x}}$) solves the following least-squares minimum-norm problem (LMP):

> **(LMP)**, *Least-squares Minimum-norm Problem*
> min $\|\dot{\boldsymbol{\theta}}\|$
> subject to min $\|\dot{\mathbf{x}} - J\dot{\boldsymbol{\theta}}\|$

All manipulators have certain configurations in their workspace, known as singular configurations, at which their freedom of movement is restricted due to peculiar kinematic alignment of the links [2]. Near such configurations, extremely high joint velocities could be required even for small end-effector motions (in the restricted or singular directions). Mathematically, the Jacobian tends to get ill-conditioned as the manipulator approaches a singular configuration.

In the neighborhood of singular configurations, it is likely (depending on the specified end-effector velocity) that all solutions satisfying the minimum-error norm constraint in LMP will be infeasible (note that if $J$ has full row rank, there will exist a $\dot{\boldsymbol{\theta}}$ such that $\dot{\mathbf{x}} = J\dot{\boldsymbol{\theta}}$). In that case, the pseudo-inverse will yield infeasible or physically unrealizable joint velocities as it chooses the minimum-norm vector from among the accurate ones. Thus. the pseudo-inverse gives priority to accuracy of solution over feasibility.

Hence, in order to ensure feasible solutions in the neighborhood of singular confidurations, it is necessary to relax the requirement of accuracy over feasibility.

### 2.1. THE DAMPED LEAST-SQUARES TECHNIQUE

The damped least-squares method, proposed independently in [28] and [36], compromises the requirement of accuracy in tracking the end-effector trajectory in order to obtain feasible joint velocities.* One way of posing the damped least-squares problem is to minimize the sum of the norm of the residual error in the task space, $\|\dot{\mathbf{x}} - J\dot{\boldsymbol{\theta}}\|$, and the norm of the joint velocity vector, $\|\dot{\boldsymbol{\theta}}\|$. The most general form in which the problem can be posed is [28]

> **(DLP-1)**, *Damped Least-squares Problem 1*
> $\min\{\|\dot{\mathbf{x}} - J\dot{\boldsymbol{\theta}}\|^2_{W_1} + \|\dot{\boldsymbol{\theta}}\|^2_{W_2}\}$

* Although we are only addressing the velocity inverse kinematic problem in this paper, damped least-squares has been applied to nonlinear least-squares at the position level [10] as well as for resolved-rate acceleration control [37] and local torque optimization [20].

where $||y||_W^2 = y^T W y$ is a weighted vector norm. The solution to the damped least-squares problem (DLP-1) is given by [28]

$$\begin{aligned} \dot{\boldsymbol{\theta}} &= J^* \dot{\mathbf{x}} = (J^T W_1 J + W_2)^{-1} J^T W_1 \dot{\mathbf{x}}, \\ J^* &= (J^T W_1 J + W_2)^{-1} J^T W_1. \end{aligned} \tag{2}$$

The matrix $J^*(\dot{\boldsymbol{\theta}})$ is termed the *Singularity Robust Inverse (SRI)*. The weighting metrices $W_1$ and $W_2$ are taken to be positive definite thus making the matrix $J^T W_1 J + W_2$ positive definite and hence nonsingular.

The weighting matrix $W_1$ can be chosen to introduce priority into the task vector, e.g. to control the deviation in one task direction more strictly than in other directions (e.g. see [4, 32]). As described in Section 4, this is especially useful in redundancy resolution methods. $W_1$ could also be chosen to make the cost function have units of energy and, hence, produce consistent solutions irrespective of units or scale.

Much of the literature on damped least-squares has studied the case when $W_1 = I$ (i.e. no task priority) and $W_2 = \lambda I$ with $\lambda \geqslant 0$. Accordingly, the SRI becomes

$$J^* = (J^T J + \lambda I)^{-1} J^T. \tag{3}$$

This formulation has been found to be effective in tackling the problem of an ill-conditioned Jacobian and hence in preventing high joint velocities near singularities. (In the numerical literature, it is also known as Levenberg–Marquardt stabilization [18].) It combines minimizing the error in following the specified trajectory with the damping of joint velocities. Thus there is a trade-off between the accuracy with which the desired end-effector trajectory is followed and the feasibility of the joint velocities. This is determined by the value of the nonnegative parameter $\lambda$ which is known as the damping factor.

Note that in (3), $\lambda$ must be non-zero for $J^T J + \lambda I$ to be invertible in the case of redundant arms. This is because for redundant manipulators, the matrix $J^T J$ never possesses full rank. However, using the SVD (singular value decomposition) [17, 22] of $J$, it can be show that $J^*$ can also be expressed as

$$J^* = J^T (J J^T + \lambda I)^{-1}. \tag{4}$$

The matrix $(J J^T + \lambda I)$ is invertible for $\lambda = 0$ as well, provided $J$ has full row rank. In fact, Equation (4) indicates that when rank of $J$ is $m$,

$$\lambda = 0 \Rightarrow J^* = J^T (J J^T)^{-1} = J^+$$

i.e. when there is no damping the SRI reduces to the pseudo-inverse. Thus, the $\lambda = 0$ case corresponds to the minimum-norm exact solution. A larger value of $\lambda$ implies smaller joint velocity norm but some deviation from the desired end-effector trajectory. (See Appendix A for a detailed analysis of the SRI using the SVD.)

Since a non-zero damping factor introduces some error into the tracking of the end-effector trajectory, the commanded end-effector velocity can be modified to include a positional error term (e.g. see [21]) i.e.

$$\dot{\mathbf{x}} = \dot{\mathbf{x}}_d + K_p(\mathbf{x}_d - \mathbf{x}_a)$$

Here, $\mathbf{x}_d$ and $\mathbf{x}_a$ are respectively the desired and actual end-effector positions and $K_p$ is the position-error feedback gain. This will help in reducing the deviation caused by introducing joint velocity damping [21, 32].

It is clear that the damping factor $\lambda$ plays a key role in the Damped Least-Squares formulation since it has to balance the accuracy of the inverse kinematic solution against its feasibility. It is hence important to choose the value of this parameter appropriately so that not only is feasibility of joint velocities ensured at all configurations but at the same time the end-effector deviation from the specified trajectory remains within tolerable limits.

## 3. The Damping Factor

A number of methods to determine the damping factor have been proposed in the literature. Following is a brief description of some of these methods:

1. One of the first methods proposed [36] was to choose a constant value for the damping factor based on a bound on the change in the residual error and estimates of the condition number of the Jacobian. However, a problem with the constant value is that it is difficult to ensure low end-effector deviation away from singularities and effective damping near singularities simultaneously.

   It would be more appropriate to use a variable damping factor, one which has a low (or zero) value away from singularities where the joint velocities do not need damping and a high value in the neighborhood of singularities when the joint velocities are expected to become infeasible.

2. Nakamura *et al.* [28] suggested adjusting the damping factor according to the value of the manipulability measure $w = \sqrt{\det(JJ^T)}$ using a threshold value $w_t$. $w$ is a nonnegative measure which becomes zero at a singular configuration. Thus, $\lambda$ is computed as [28]

$$\begin{aligned} \lambda &= \lambda_0\left(1 - \frac{w}{w_t}\right)^2, \quad \text{if } w < w_t, \\ &= 0, \quad \text{if } w \geqslant w_t. \end{aligned}$$

With this method, no damping is applied when the value of $w$ is greater than $w_t$. Damping is progressively increased as $w$ drops below $w_t$ until it reaches its maximum value $\lambda_0$ at $w = 0$.

3. On similar lines, Kelmar *et al.* [13] proposed changing the damping factor according to the rate of change of the manipulability measure. This makes the determination of the damping factor less dependent on the scaling of the manipulator. Hence, this method is particularly useful when applying damped least-squares to reconfigurable modular manipulator systems, which is the application considered in [13]. If $w_{k-1}$ and $w_k$ are the values of the manipulability measure at the previous and current iterations respectively, then $\lambda$ is computed as

$$\begin{aligned}\lambda &= \lambda_0\left(1 - \frac{w_k}{w_{k-1}}\right), \quad \text{if } \frac{w_k}{w_{k-1}} < w_t,\\ &= 0, \quad \text{otherwise.}\end{aligned}$$

As before, $w_t$ is a threshold value and $\lambda_0$ is a scaling constant.

4. In [3], Chan *et al.* have proposed the error-damped pseudoinverse (EDP) which is the same as the singularity robust inverse. Here, the damping factor is varied according to the residual error in the task-space

$$\lambda = \lambda_0 \|\delta x\|^2, \tag{5}$$

where $\delta x$ is the worksapce error i.e., the difference between the actual end-effector position and the desired end-effector position.

5. Maciejewski *et al.* [21] use the minimum singular value $\sigma_{\min}$ of the Jacobian to set a damping factor which is guaranteed to satisfy a given constraint on either the joint angle velocity, end-effector tracking error or conditioning of the equations. If $\dot{\theta}_{\max}$ is the maximum allowable joint angle velocity over all unit norm end-effector velocities, then the damping factor $\lambda$ has to be at least

$$\lambda_{\dot{\theta}_{\max}} = \frac{\sigma_{\min}}{\dot{\theta}_{\max}} - \sigma_{\min}^2 \quad \text{if} \quad \sigma_{\min} \leqslant \frac{1}{\dot{\theta}_{\max}} \tag{6}$$

If $\sigma_{\min} > 1/\dot{\theta}_{\max}$ then $\lambda = 0$. Let $\Delta R_{\max}$ denote the maximum allowable end-effector tracking error relative to the end-effector velocity $\dot{\mathbf{x}}$. Then the damping factor $\lambda$ should be at most

$$\lambda_{\Delta R_{\max}} = \frac{\sigma_r^2 \Delta R_{\max}}{1 - \Delta R_{\max}}, \tag{7}$$

where $\sigma_r$ is the minimum non-zero singular value of $J$. If the condition number of $(J^T J + \lambda I)$ is bounded to be below $k_{\max}$, then the damping factor can be computed as

$$\lambda_{k_{\max}} = \frac{\sigma_1^2 - k_{\max}\sigma_n^2}{k_{\max} - 1}. \tag{8}$$

$\sigma_1$ and $\sigma_n$ are the maximum and minimum singular values of $J$ respectively ($\sigma_n = 0$ if $m < n$).

Equations (6), (7) and (8) require an estimate of the minimum singular value $\sigma_{\min}$ of the Jacobian. An elegant way of obtaining an estimate of the minimum singular value through the concept of a unit vector $\hat{u}_m$ which rotates along with the singular vectors(s) associated with the minimum singular value(s) has been described in [21].

6. In [5], Chiaverini *et al.* have applied a weighted damped least-squares solution to the 5 d.o.f. ABB Trallfa TR 400 manipulator. Weighting of the task vector and hence the Jacobian enables high accuracy tracking of assigned task-space components while allowing lower tracking performance in other directions. Thus,

$$\widetilde{\dot{\mathbf{x}}} = W\dot{\mathbf{x}} = WJ\dot{\boldsymbol{\theta}} = \widetilde{J}\dot{\boldsymbol{\theta}} \quad (\widetilde{J} = WJ),$$

$$\Rightarrow \dot{\boldsymbol{\theta}} = \widetilde{J}^*\widetilde{\dot{\mathbf{x}}} = (\widetilde{J}^T\widetilde{J} + \lambda I)^{-1}\widetilde{J}^T\widetilde{\dot{\mathbf{x}}} = (J^T W^T W J + \lambda I)^{-1} J^T W^T W\dot{\mathbf{x}}$$

Notice that this is the same solution as Equation (2) with $W_1 = W^T W$. Suppose $d$ is a vector representing the direction in end-effector space in which lower tracking accuracy is a acceptable. Then the matrix $W$ is determined as [5]

$$W = I + (w - 1)dd^T, \quad 0 < w < 1,$$

where $w \in \mathbb{R}$ is a weighting factor. The weighting factor and the damping factor in [5] are computed as

$$\left.\begin{aligned} \lambda &= \left[1 - \left(\frac{\widehat{\sigma}_5}{\varepsilon}\right)^2\right]\lambda_{\max} \\ (w-1)^2 &= \left[1 - \left(\frac{\widehat{\sigma}_5}{\varepsilon}\right)^2\right](w_{\min} - 1)^2 \end{aligned}\right\} \text{ when } \widehat{\sigma}_5 < \varepsilon.$$

If $\widehat{\sigma}_5 \geqslant \epsilon$, then $\lambda = 0$ and $w = 1$. Here, $\widehat{\sigma}_5$ is an estimate of the minimum singular value while $\epsilon$ is a threshold value chosen to define the size of the singular region.

7. A variable damping factor scheme is also used in [25]. The damping factor is decided based on the value of a parameter $l$. If $l \leqslant 1, \lambda = 0$, otherwise $\lambda$ is computed as

$$\lambda = \lambda_0(1 - l)^2$$

Two schemes have been proposed to compute $l$ in [25]. The first one is based on maintaining the condition number of the matrix $JJ^T + \lambda I$ below a certain threshold value. $l$ is computed as the ratio of the upper bound on the condition number to the threshold value. The second scheme uses a condition for rank-preservation of the Jacobian from one iteration to the next.

## Remarks

The methods described above (with the exception of No. 4) compute the damping factor $\lambda$ based on some Jacobian-dependent measure such as the manipulability measure, condition number or the minimum singular value which indicates the closeness of the manipulator to a singular configuration. However, it is important to note that proximity to a singularity does not necessarily imply high joint velocities. Near a singular configuration, joint velocities will tend to become infeasible only if the end-effector velocity has a component in the singular direction (i.e. the singular vector corresponding to the minimum singular value(s)).

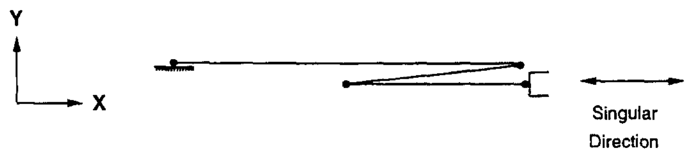


Fig. 1. A singular configuration.

For example, for the configuration shown in Figure 1, a small end-effector valocity in the X-direction (which is the singular direction) will result in high joint velocities while one which is purely in the Y-direction will not. But since the Jacobian is the same in either case, damping will be applied in both situations if the damping factor is computed using a Jacobian-based measure even though it is unnecessary in the latter case.

In [21], this problem has been dealt with by computing a solution in which the components associated with the small singular values of the Jacobian are damped more than the other components. This is achieved by using a filter gain to provide further damping of the singular components in addition to the overall damping factor. The joint velocity is thus computed as

$$\dot{\boldsymbol{\theta}} = J^T(JJ^T + \alpha^2\widehat{u}_m\widehat{u}_m^T + \lambda I)^{-1}\dot{\mathbf{x}}$$

Here, $\widehat{u}_m$ is a unit vector with a significant component in the subspace spanned by the singular vectors associated with the small singular values. Thus, the filter gain $\alpha$ provides further damping of the singular components in addition to the overall damping provided by $\lambda$. The value of $\alpha$ is set based on the minimum singular value as described earlier (method 5) while that of $\lambda$ is decided based on an effective singular value outside of the singular subspace. This scheme allows the manipulator to differentiate between the attainable and unattainable components of the end-effector velocity thus avoiding unnecessary damping when the commanded end-effector velocity has no components in the singular direction(s).

Another common difficulty in the methods mentioned above is the determination of threshold values and value of the constant multiplier $\lambda_0$. The values of these parameters are robot-dependent and will be different for different manipulators. Also, it is not clear how these values can be determined such that proper damping of joint velocities is ensured without incurring very large deviation from the planned end-effector trajectory.

### 3.1. COMPUTING AN OPTIMAL DAMPING FACTOR

An optimal damping factor can be defined as one which results in minimum possible end-effector deviation while maintaining the norm of the joint velocity vector below a desired value, say $\Delta$, at each configuration. This motivates the following optimization problem [22].

> **(DLP-2)**, *Damped Least-squares Problem 2*
> min $\|\mathbf{u}\|$
> subject to $\|\dot{\boldsymbol{\theta}}\| \leqslant \Delta$
> $\dot{\mathbf{x}} = J\dot{\boldsymbol{\theta}} + \mathbf{u}$

Here **u** denotes the error incurred in tracking the desired trajectory. This problem defines a feasible joint velocity vector to be one with norm less than or equal to $\Delta$. Hence, instead of choosing $\lambda$ which was necessary to solve DLP-1, we can now choose $\Delta$ to solve DLP-2. Having control over $\Delta$ ensures that each component of the joint velocity vector will certainly have a value less than $\Delta$. $\Delta$ can be determined from the physical characteristics of the manipulator actuators [6]. By computing the joint velocity vector as a solution to DLP-2, this upper bound on the norm will never be exceeded.

The solution to the DLP-2 problem is the same as (3) but now the damping factor $\lambda$ arises as the Lagrange multiplier of DLP-2. Using the first-order necessary conditions of Lagrange's method, we obtain,

$$\dot{\boldsymbol{\theta}}^*_\lambda = J^T(JJ^T + \lambda I)^{-1}\dot{\mathbf{x}} \tag{9}$$

with $\lambda \geqslant 0$ and $\lambda\big(\|\dot{\boldsymbol{\theta}}^*_\lambda\|^2 - \Delta^2\big) = 0$

We have obtained Equation (4) as the solution rather than Equation (3) by modifying the optimization problem in [22] slightly. Thus, Equation (9) is valid for the case $\lambda = 0$ for both nonredundant and redundant manipulators since the matrix $JJ^T$ is invertible whenever $J$ has full row rank.

The following statements* summarize the solution $\dot{\boldsymbol{\theta}}^*$ to DLP-2:

1. If $\lambda = 0$ and $\|\dot{\boldsymbol{\theta}}_0^*\| \leqslant \Delta$, then $\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{\theta}}_0^* = J^+\dot{\mathbf{x}}$
2. If $\lambda > 0$ and $\|\dot{\boldsymbol{\theta}}_\lambda^*\| = \Delta$, then $\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{\theta}}_\lambda^*$ as in (9).

Thus, whenever the pseudo-inverse solution yields a joint velocity norm less than or equal to the allowable norm $\Delta$, the SRI is equal to the pseudo-inverse and the optimal damping factor $\lambda^* = 0$. Intuitively, this makes sense since the pseudo-inverse gives the minimum-norm solution among all the exact solutions. Therefore, if the pseudo-inverse solution norm is within the allowable limit $\Delta$, then this indicates that there is a feasible solution among all the accurate solutions and hence no damping is required. But, if the norm of the pseudo-inverse solution is greater than $\Delta$, then this means that none of the accurate solutions are feasible and statements (2) of the DLP-2 solution says that the SRI will provide a feasible solution with minimum deviation from the specified trajectory if the damping factor $\lambda$ is such that $\|\dot{\boldsymbol{\theta}}_\lambda^*\| = \Delta$. Hence, let

$$\phi(\lambda) = \|\dot{\boldsymbol{\theta}}_\lambda^*\| = \|J^T(JJ^T + \lambda I)^{-1}\dot{\mathbf{x}}\| = \Delta \tag{10}$$

Then, whenever $\phi(0) > \Delta$ i.e. the pseudo-inverse solution is infeasible, the following nonlinear equation has to be solved for the optimal damping factor.

$$\phi(\lambda) - \Delta = \|\dot{\boldsymbol{\theta}}_\lambda^*\| - \Delta = \|J^T(JJ^T + \lambda I)^{-1}\dot{\mathbf{x}}\| - \Delta = 0 \tag{11}$$

This equation can be expanded using the singular value decomposition (SVD) of the Jacobian matrix $J$. The SVD of $J$ can be written as

$$J = UDV^T \tag{12}$$

where $U \in \mathbb{R}^{m\times n}$ and $V \in \mathbb{R}^{n\times n}$ are orthogonal matrices. $D \in \mathbb{R}^{m\times n}$ is a matrix with all non-diagonal entries equal to zero and with the diagonal entries $D_{i,i} = \sigma_i$ with $\sigma_1 \geqslant \sigma_2 \cdots \geqslant \sigma_m \geqslant 0$. The $\sigma_i$'s are known as the singular values of the Jacobian $J$. The SVD is usually computed numerically [17, 22]. However, for certain manipulators, analytical expressions for the singular values and singular vectors (and hence for the pseudo-inverse and the damped least-squares soultion) have been obtained [15, 16].

* Although it is not explicit in the problem statement, if there is more than one feasible vector that minimizes $\mathbf{u}$, then the one with minimum norm will be the solution. This will be the case when there exist $\dot{\boldsymbol{\theta}}$ such that $\dot{\mathbf{x}} = J\dot{\boldsymbol{\theta}}$. Hence, in that case, $J^+\dot{\mathbf{x}}$ will be the solution.

Using Equation (12), the function $\phi(\lambda)$ can be evaluated as follows (see Appendix A),

$$\phi(\lambda) = \|J^T(JJ^T + \lambda I)^{-1}\dot{\mathbf{x}}\| = \|D^T(DD^T + \lambda I)^{-1}U^T\dot{\mathbf{x}}\| \tag{13}$$

where we have used the norm-preserving property of the orthogonal matrix $V$ i.e., $\|Vy\| = \|y\|$ for any $y \in \mathbb{R}^n$. Let

$$U^T\dot{\mathbf{x}} = [\gamma_1\gamma_2\cdots\gamma_m]^T \tag{14}$$

Then,

$$\phi(\lambda) = \sqrt{\sum_{i=1}^{m} \frac{\sigma_i^2\gamma_i^2}{(\sigma_i^2 + \lambda)^2}} \tag{15}$$

Differentiating $\phi(\lambda)$ from Equation (15) with respect to $\lambda$ and using (10) and (13),

$$\phi'(\lambda) = -\frac{\sum_{i=1}^{m} \frac{\sigma_i^2\gamma_i^2}{(\sigma_i^2+\lambda)^3}}{\sqrt{\sum_{i=1}^{m} \frac{\sigma_i^2\gamma_i^2}{(\sigma_i^2+\lambda)^2}}} = -\frac{1}{\|\dot{\boldsymbol{\theta}}_\lambda^*\|}\sum_{i=1}^{m} \frac{\sigma_i^2\gamma_i^2}{(\sigma_i^2 + \lambda)^3}.$$

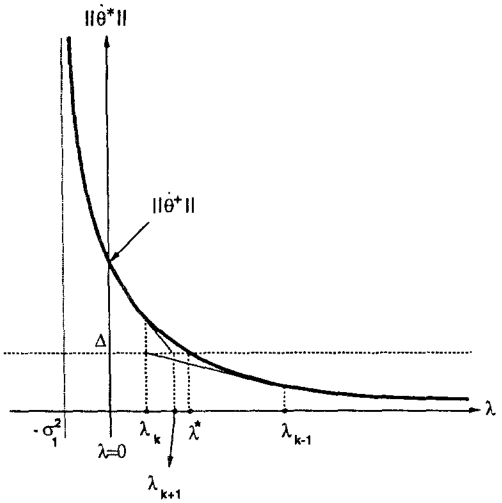


Fig. 2. Plot of damped joint velocity as a function of damping factor.

In [22], Newton's method is used to solve Equation (11). This results in the following iteration for updating a given approximation $\lambda_k$ to $\lambda^*$.

$$\lambda_{k+1} = \lambda_k - \left[\frac{\phi(\lambda_k) - \Delta}{\phi'(\lambda_k)}\right] \tag{17}$$

From Equation (15), the function $\phi(\lambda)$ is approximately of the form $a/(b + \lambda)$, which implies that $1/\phi(\lambda)$ rather than $\phi(\lambda)$, is almost linear for $\lambda > 0$. Since Newton's method uses a linear approximation to the function, an alternate iterative method can be obtained by applying Newton's method to the reciprocal equation [6, 7, 26]

$$\frac{1}{\phi(\lambda)} - \frac{1}{\Delta} = 0.$$

This results in the iteration

$$\lambda_{k+1} = \lambda_k - \left[\frac{\phi(\lambda_k)}{\Delta}\right]\left[\frac{\phi(\lambda_k) - \Delta}{\phi'(\lambda_k)}\right] = \lambda_k - \left[\frac{\|\dot{\theta}^*_\lambda\|}{\Delta}\right]\left[\frac{\phi(\lambda_k) - \Delta}{\phi'(\lambda_k)}\right] \quad (18)$$

Both iterations (17) and (18) are quite efficient in computing the optimal damping factor, usually requiring 2–3 iterations to converge to the solution (planar arm – 2, spatial arm – 3) [7, 22]. However, it is observed that (18) performs better than (17) in instances when a singular value of $J$ is very close to zero, in which case the graph of $\phi(\lambda)$ is quite steep at the origin. In such situations, Newton's method can take more iterations to converge to the solution than iteration (18).

The efficiency of the algorithms can be improved by using the optimal damping factor for the previous configuration as the initial value ($\lambda_1$). It is also important to bound the value of the iterate in order to ensure fast convergence. This is because, if the initial iterate is greater than the optimal value, the next iterate could be negative. Hence, it is necessary to safeguard the iteration schemes with lower and upper bounds for $\lambda$ at each iteration (for more details, see [7, 26]).

In summary, several different methods to compute the damping factor have been studied. Some of these methods estimate the damping factor using Jacobian-dependent measures such as the manipulability measure or the singular values. However, they may not take into account the commanded end-effector velocity, which may result in unneccesary damping when the desired end-effector velocity has no component in the singular direction(s). Moreover, they rely on empirically selected threshold values and multipliers which will be different for different manipulators.

The optimal damping factor is more expensive to compute but has the advantage of resulting in minimum tracking error. The monotonic variation of the joint velocity norm (Figure 2) with the damping factor lends itself to the formulation of efficient iterative algorithms. No *ad hoc* selection of parameters is required in this case. Also, the same algorithm can be used to compute the damping factor for any manipulator.

## 4. Redundancy Resolution with Damped Least-Squares

Manipulators are said to be redundant when they possess more degrees of freedom than the minimum number required to execute a given task. In the case of nonredundant manipulators, the joint configuration for any given end-effector position and orientation is unique except for finite variations. As a result, given an end-effector position, there is usually only one feasible joint configuration. However, in the case of redundant manipulators, there exist infinite joint configurations corresponding to a given end-effector position. This endows redundant manipulators with the property of *self-motion* i.e. the ability to move joints without moving the end-effector. Self-motion makes redundant manipulators capable of optimizing various performance criteria in addition to the main motion task. Some of the criteria that can be satisfied using the redundancy of a manipulator are singularity avoidance, obstacle avoidance, joint-limit avoidance, torque optimization, energy minimization, etc. [12, 14, 23, 29, 34, 35, 39, 40].

Several different approaches to redundancy resolution have been suggested in the literature (for example, see the survey papers [30, 33]). Two popular techniques are task space augmentation and gradient projection. Both these have been used in conjunction with damped least-squares as described below.

In the task space-augmentation method, the $m$-dimensional end-effector position (which constitutes the main task) is augmented by an additional constraint task of dimension $n - m$. This results in a square Jacobian matrix relating the joint space to the task space. Thus if the end-effector position $\mathbf{x}_e \in \mathbb{R}^m$ and the additional constraint task $\mathbf{x}_c \in \mathbb{R}^{n-m}$ are related to the joint position vector by continuous functions

$$\mathbf{x}_e = f_e(\boldsymbol{\theta}); \; \mathbf{x}_c = f_c(\boldsymbol{\theta})$$

then differentiating these relations and concatenating them, we obtain

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{x}}_e \\ \dot{\mathbf{x}}_c \end{bmatrix} = \begin{bmatrix} J_e \\ J_c \end{bmatrix} \dot{\boldsymbol{\theta}} = J\dot{\boldsymbol{\theta}} \qquad \left( J_e = \frac{\partial f_e}{\partial \boldsymbol{\theta}}; \; J_c = \frac{\partial f_c}{\partial \boldsymbol{\theta}} \right)$$

Thus, a square Jacobian is obtained which can be directly inverted (when it has full rank) to obtain $\dot{\boldsymbol{\theta}}$ corresponding to any given $\dot{\mathbf{x}}$. However, the problem with task augmentation is that apart from the manipulator's singularities (when $J_e$ does not have full row rank), additional singularities arise if $J_c$ does not have full row rank ir if $J$ does not have full rank. These are known as algorithmic or artificial singularities [1]. One way of ensuring feasible joint velocities at these singularities is by using damped least squares with the augmented Jacobian (e.g. [9, 32]).

In [32], the following function is minimized.

$$L = \dot{E}_e^T W_e \dot{E}_e + \dot{E}_c^T W_c \dot{E}_c + \dot{\boldsymbol{\theta}}^T W_v \dot{\boldsymbol{\theta}} \tag{19}$$

where $\dot{E}_e = \dot{\mathbf{x}}_e - J_e\dot{\boldsymbol{\theta}}$ and $\dot{E}_c = \dot{\mathbf{x}}_c - J_c\dot{\boldsymbol{\theta}}$. The inverse kinematic solution is then obtained as

$$\dot{\boldsymbol{\theta}} = (J^T W J + W_v)^{-1} J^T W \dot{X}_d.$$

Here, $W = \text{diag}(W_e, W_c)$. Notice that this is the same as the weighted damped least-squares formulation (2), except that different weightings are used for the basic and additional tasks. Thus, the weighting matrices can be used to prioritize the two tasks. Also, in the case of algorithmic (artificial) singularities, incompatibility of the two tasks does not cause numerical difficulties since the matrix $[J^T W J + W_v]$ is nonsingular even when $J$ is rank-deficient. The weighting matrices can be varied (as described in [32]) to ensure that the main task performance does not degrade due to unattainability of the constraint task.

In [9], a similar technique has been used. In this case, the matrix $W_e$ in (19) is taken to be the matrix, $W_c = wI$ with $w \ll 1$ and $W_v = \lambda I$ with $\lambda < w$. This strategy ensures that near an artificial singularity when the main task Jacobian has full row rank, the damping will not seriously affect the main task and will mostly affect the constraint task only, thus achieving task priority. Both $w$ and $\lambda$ are varied with respect to the 'distance from singularities' in order to avoid unnecessary weighting or damping (for more details, see [5, 4, 9]).

Another redundancy resolution method is that of projecting a vector onto the null space of the Jacobian [19, 39, 40], in order to obtain the desired self-motion. In this case, the joint velocity vector is computed as the sum of a particular and homogeneous solution as

$$\dot{\boldsymbol{\theta}} = J^{+}\dot{\mathbf{x}} + (I - J^{+}J)\mathbf{v} \tag{20}$$

The first term $J^{+}\dot{\mathbf{x}}$ is used to perform the main task of end-effector trajectory tracking, while the second term which does not have any effect on the motion of the end-effector is used to generate the self-motion. Note that (20) is a solution to the problem [36],

| |
|---|
| **NPP**, *Nullspace Projection Problem* |
| $\min \|\dot{\boldsymbol{\theta}} - \mathbf{v}\|$ |
| subject to $\min \|\dot{\mathbf{x}} - J\dot{\boldsymbol{\theta}}\|$ |

If an additional criterion $h(\boldsymbol{\theta})$ is to be minimized, the vector $\mathbf{v}$ can be selected as the negative of the gradient of $h(\boldsymbol{\theta})$ [39], i.e.

$$\mathbf{v} = -k\nabla h(\boldsymbol{\theta}) = -k\,\frac{\partial h(\boldsymbol{\theta})}{\partial \boldsymbol{\theta})}$$

In [36], the problem DLP-1 has been combined with NPP as follows:

$$\min \|\dot{\mathbf{x}} - J\dot{\boldsymbol{\theta}}\|^2 + \alpha_1^2\|\dot{\boldsymbol{\theta}} - \mathbf{v}\|^2 + \alpha_2^2\dot{\boldsymbol{\theta}}^T A\dot{\boldsymbol{\theta}} \tag{21}$$

where $A$ is a positive definite, symmetric matrix. This gives the joint velocity vector as

$$\dot{\boldsymbol{\theta}} = (J^T J + \alpha_1^2 I + \alpha_2^2 A)^{-1}(J^T\dot{\mathbf{x}} + \alpha_1^2\mathbf{v}) \tag{22}$$

If we use $A = I$ and substitute $\alpha_1^2 + \alpha_2^2 = \lambda$, then for a nonzero $\lambda$, (22) can be shown to be equivalent to

$$\dot{\boldsymbol{\theta}} = J^*\dot{\mathbf{x}} + \frac{\alpha_1^2}{\lambda}(I - J^*J)\mathbf{v} \tag{23}$$

Notice that this looks similar to Equation (20) with $J^+$ replaced by $J^*$ except for multiplication by the factor $\alpha_1^2/\lambda$.

A similar formulation has been used in [24] but in this case, there is no division of the second term by the damping factor, i.e.

$$\dot{\boldsymbol{\theta}} = J^*\dot{\mathbf{x}} + (I - J^*J)\mathbf{v} \tag{24}$$

In [24], the vector $\mathbf{v}$ has been chosen to keep the manipulator links away from any obstacle present in the workspace.

It is important to note that in (23) and (24), the $(I - J^*J)$ term does have a component orthogonal to the null-space of the Jacobian and hence it will have some effect on the motion of the end-effector [8].

One issue that needs to be addressed is the computation of the optimal damping factor for the redundancy resolution methods. In the case of the task augmentation method, the same techniques as described in Section 3 can be used with the manipulator Jacobian replaced by the augmented Jacobian. However, in this case artificial singularities are introduced and hence, the joint velocities may have to be damped even if the main task is attainable without damping. Thus, unnecessary errors are introduced into the main task and this may not be desirable in some cases.

On the other hand, in case of the gradient projection method, this problem is avoided because if the main task Jacobian has full rank and the augmented Jacobian does not, then the range of the constraint Jacobian (in this case, the row vector $v^T$) is orthogonal to the null space of the main task Jacobian [1]. This means that close to an artificial singularity, the second term on the right-hand side of (20) will be very small. Hence the norm of $\dot{\boldsymbol{\theta}}$ obtained by (20) will be feasible and damping will not be required. The following subsection describes a way of computing the optimal damping factor for the gradient projection solution (23) when the norm of the solution given by (20) is infeasible.

### 4.1. BOUNDING THE TOTAL JOINT VELOCITY

Formulation (23) facilities the computation of an optimal damping factor, i.e. whenever the norm of the joint velocity vector given by (20) is greater than $\Delta$, a damping factor $\lambda$ can be computed such that norm of $\dot{\boldsymbol{\theta}}$ as given by (23) (i.e. the total joint velocity) is equal to $\Delta$.

The technique for computing such a damping factor is similar to that used in Section 3.1. In order to find a damping factor $\lambda$ which yields a joint velocity vector of norm equal to $\Delta$, Equation (25) needs to be solved for $\lambda$.

$$\|\dot{\theta}^*\| = \left\| J^*\dot{\mathbf{x}} - (I - J^*J)\left(\frac{\nabla h}{\lambda}\right)\right\| = \Delta. \tag{25}$$

Using the SVD of the Jacobian $J$ (Equation (12)),

$$\begin{aligned}\dot{\theta} &= J^*\dot{\mathbf{x}} + (I - J^*J)(\nabla h/\lambda)\\ &= VD^*U^T\dot{\mathbf{x}} + V(I - D^*D)(V^T\nabla h/\lambda),\end{aligned} \tag{26}$$

where $D^* = D^T(DD^T + \lambda I)^{-1}$. Thus, using (14) and defining $V^T\nabla h = [\beta_1\beta_2\cdots\beta_n]$, we obtain

$$\|\dot{\theta}^*\| = \sum_{i=1}^{m}\frac{(\sigma_i\gamma_i + \beta_i)^2}{(\sigma_i^2 + \lambda)^2} + \sum_{i=m+1}^{n}\frac{\beta_i^2}{\lambda^2}. \tag{27}$$

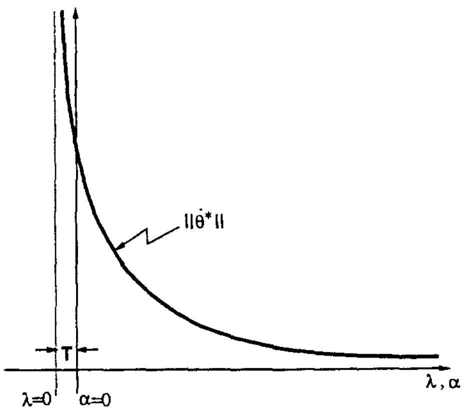


Fig. 3. Shift of axis.

Notice that unlike the function in (13), this function is not defined at $\lambda = 0$. Hence, in order to compute the optimal damping factor, we shift the function as follows:

Choose a small number $T$, say equal to 0.0001 and let

$$\alpha = \lambda - T. \tag{28}$$

Then substituting for $\lambda$ from Equation (28) into Equation (27) we get

$$\|\dot{\theta}^*\|^2 = \sum_{i=1}^{m}\frac{(\sigma_i\gamma_i + \beta_i)^2}{(\sigma_i^2 + \alpha + T)^2} + \sum_{i=m+1}^{n}\frac{\beta_i^2}{(\alpha + T)^2} \tag{29}$$

Equation (27) indicates that the norm of the joint velocity vector goes to infinity when $\lambda$ is zero. By using the substitution (28) we have shifted the axis by $T$ in order to get a finite norm at $\alpha = 0$ (Figure 3). Now the nonlinear equation $\|\dot{\boldsymbol{\theta}}^*\| = \Delta$ can be solved for $\alpha^*$ with the expression in Equation (29) using the iterative methods described in Section 2.1. The corresponding $\lambda^*$ is then computed from (28).

One major difference between computing the optimal damping factor for the SRI alone and for Equation (23) is the fact that the transition from a zero to non-zero value of $\lambda$ is not as smooth in the latter case as it is in the former. This is because $\lambda = 0$ in the latter case does not correspond to substituting $J^*$ with $J^+$. However, using the algorithm outlined above will result in Equation (20) being used when no damping is required and otherwise the damping factor is computed as described.

## 5. Simulations

The simulations in this section use some of the techniques described in the previous sections to demonstrate the usefulness of the damped least-squares method in obtaining feasible joint velocity solutions near singularities.

First, a 3-link planar manipulator with revolute joints and link lengths equal to 1, 0.5 and 0.5 units respectively, is used to show the comparison between the inverse kinematic solutions obtained by using the pseudo-inverse and the SRI (see Figure 4 and 5). The specified trajectory is a rectangle ($0.9 \times 0.7$ units). The starting configuration is $\theta_0 = [5^\circ \ -175^\circ \ 175^\circ]^T$ which is nearly singular. The end-effector is initially at the lower corner of the square and is made to move in a counter-clockwise direction along the sides of the square. Here $\Delta = 5$.

Note that the use of the pseudo-inverse results in high peaks in the joint velocity vector norm near singular configurations of the manipulator. On the other hand, with the SRI, the joint velocity vector norm is contained below $\Delta = 5$ throughout the manipulator's motion by the use of the optimal damping factor. Here, iteration (18) has been used to compute the damping factor. On an average, 2 iterations were required to compute the optimal value at each configuration.

Similar results are obtained in the case of the 6 d.o.f. spatial PUMA arm (Figure 6). The Jacobian matrix used for the simulations has been taken from [31]. The arm is started close to a shoulder singularity which occurs when the wrist is located along the axis of the shoulder (and in this case, above the shoulder). The singular directions for this configuration lie along the $y$-axis of the world frame, orthogonal to the plane formed by the two elbow links.

When the arm is made to move in this singular direction with $\dot{\mathbf{x}}$ specified as [0 0.5 0 0 0 0], the pseudo-inverse solution tracks the desired trajectory accurately, but has a very high norm initially (Figure 6). On the other hand, when the SRI with the optimal damping factor is used to compute the joint velocities, the norm stays feasible throughout the motion. In this case, there is some deviation

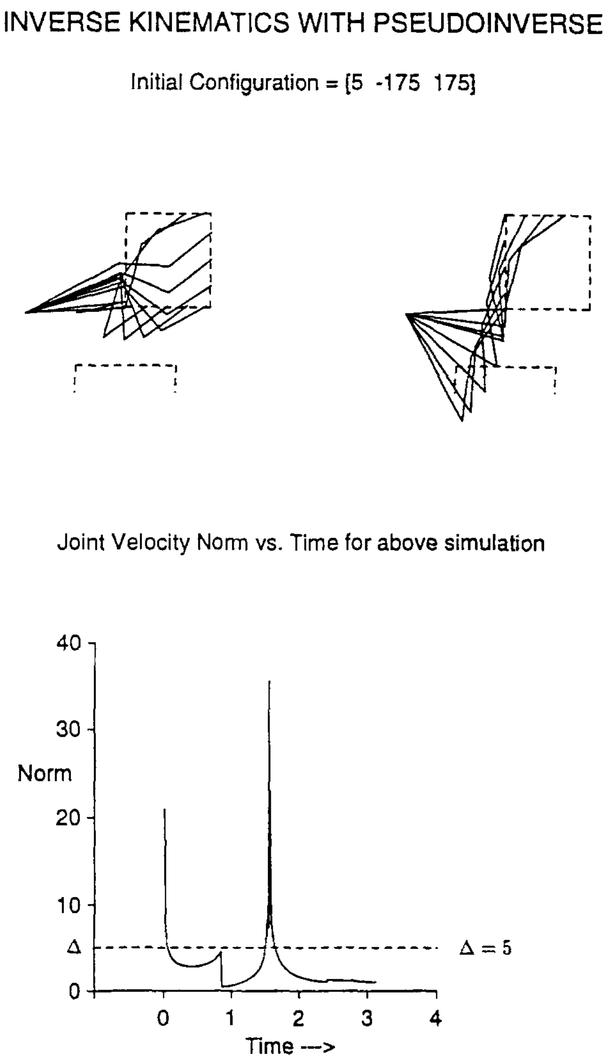


Fig. 4. Planar arm (pseudo-inverse).

from the specified trajectory as seen in Figure 6. In this simulation, the algorithm required 3 iterations to compute the optimal damping factor at each configuration for which the pseudo-inverse solution was not feasible.

Comparison between the pseudo-inverse and the SRI can also be found in [13, 28, 36, 37]. Simulations comparing the various methods of computing the damping factor have been presented in [3, 21, 25].

The next simulation (Figure 7) demonstrates the use of damped least-squares with the performance of an additional subtask for the 3-link planar redundant manipulator. We use the same trajectory as in the earlier simulations (Figures 4, 5), but this time an obstacle is placed below the lower left corner of the square

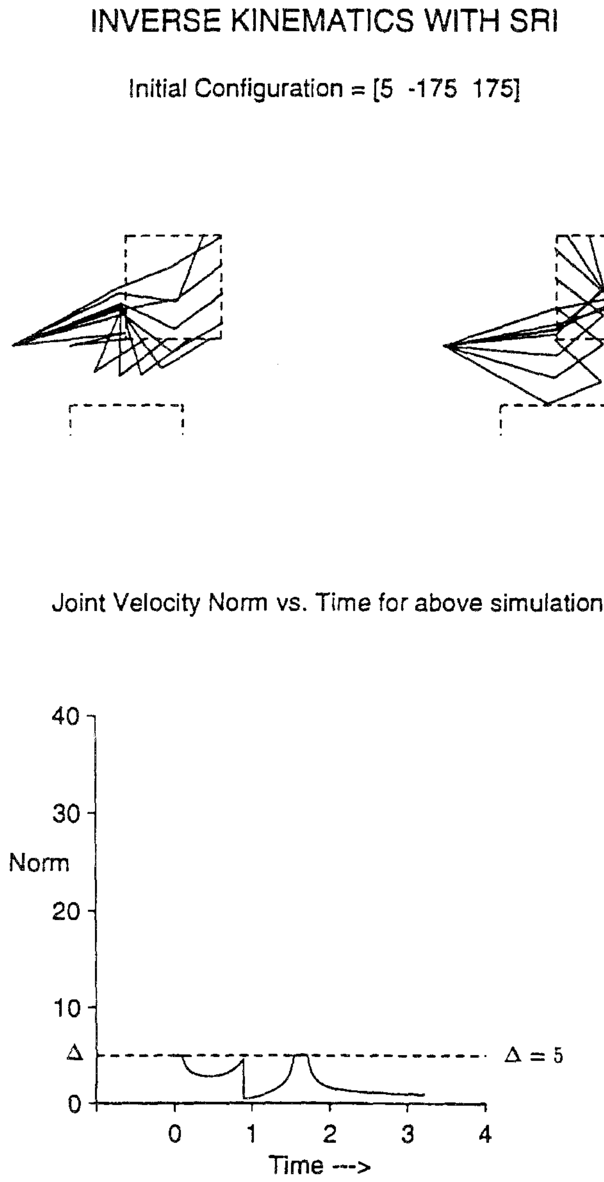


Fig. 5. Planar arm (SRI).

trajectory and the technique described in Section 4.1 is used to compute the inverse kinematics. Hence, as seen in Figure 7, the joint velocity norm is less than or equal to Δ throughout the motion. In addition, the manipulator has now managed to avoid the obstacle in its workspace by coming down the last side of the square with an *elbow up* configuration as opposed to the *elbow down* configuration in the previous simulations. The computation of the optimal damping factor took 3 iterations on the average.

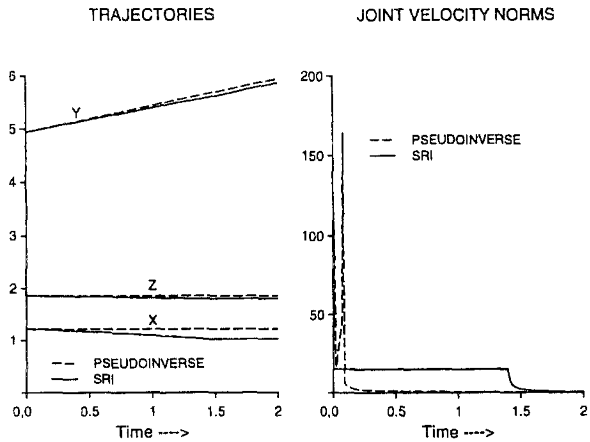


Fig. 6. Damped least-squares with the PUMA.

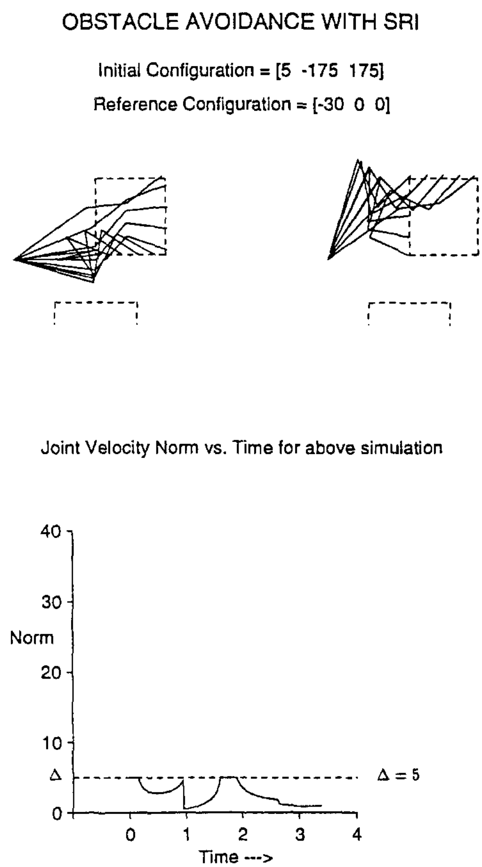


Fig. 7. Planar arm (obstacle avoidance).

Other examples of damped least-squares with redundancy resolution with planar manipulators as well as spatial manipulators have been presented in [8, 9, 24, 32].

## 6. Conclusions

In this paper, we have attempted to present a comprehensive account of the damped least-squares technique which can be used to compute the inverse kinematics of robotic manipulators. The method has been shown to be useful for computing velocity inverse kinematics near singular configurations of the manipulator as it affords a way of obtaining feasible joint velocities in cases where the pseudo-inverse of the Jacobian matrix may yield infeasible solutions.

Feasibility of the joint velocity solution is obtained at the cost of the accuracy of the solution in terms of tracking the desired end-effector trajectory. The parameter which determines the amount of damping applied to the joint velocities is the damping factor $\lambda$. The case $\lambda = 0$ corresponds to the pseudo-inverse solution. Larger the value of $\lambda$, lesser is the norm of the joint velocity vector obtained but greater is the deviation from the end-effector trajectory.

We have outlined the various methods that have been suggested in the literature for the computation of the damping factor. Almost all the methods keep the damping factor zero in regions where a feasible pseudo-inverse solution is possible. Many of the methods use a Jacobian-based measure such as the manipulability measure or the minimum singular value to determine the proximity of the manipulator to singularities. However, this may lead to unnecessary damping if the commanded end-effector velocity is not taken into account.

Alternatively, an optimal damping factor that keeps the joint velocity vector-norm below a certain value $\Delta$ and guarantees minimum possible end-effector deviation, can be computed as described in Section 3.1. Although this method is computationally more expensive than the other methods, it has the advantage of requiring no threshold values or ad hoc constants. Also, it naturally chooses the pseudo-inverse solution whenever it is feasible i.e. when $\|J^{+}\dot{\mathbf{x}}\| \leqslant \Delta$. The iterative methods that can be used to compute the optimal damping factor are easy to implement and usually converge in 2–3 iterations, as shown in the simulations.

The damped least-squares technique has also been used in redundancy resolution methods to combine the requirement of feasible joint velocities with the satisfaction of an additional constraint task. When used with the augmented Jacobian method, this technique prevents infeasible soultions near algorithmic or artificial singularities in addition to the manipulator's own singularities. The SRI has also been used with the gradient projection formulation to optimize a subtask criterion. In this case, two terms similar to the particular ($J^{+}\dot{\mathbf{x}}$) and homogeneous ($(I - J^{+}J)\mathbf{v}$) solutions but with $J^{+}$ replaced by $J^{*}$ are obtained. However, it must be noted that in this case, the $(I - J^{*}J)$ term used is not entirely in the null space of the Jacobian. Hence it will affect the motion of the end-effector depending on the value of $\lambda$. We have also presented a new method for computing an optimal damping factor for this formulation, which will enable the norm of the joint velocity vector to be bounded below a desired value while performing the additional subtask.

It must be noted that the damped least-squares method at the velocity level uses a linear model of the direct kinematic function of the manipulator. This linear model may not be a good representation of the function in certain regions, notably at the workspace boundary. Hence in such situations, the linear model may have to be supplemented with second-order information.

In conclusion, the damped least-squares method is an attractive way of dealing with an ill-conditioned Jacobian matrix in the neighborhood of singular configurations. It affords a way of obtaining feasible joint velocities in such regions. Use of an optimal damping factor will guarantee damping only when necessary and just enough to ensure feasible solutions with minimum possible end-effector deviation. Combining redundancy resolution with damped least-squares enables the performance of an additional subtask with the assurance of feasible joint velocities in the case of redundant manipulators.

## Acnowledgements

This work was supported in part by NSF under grant MSS-9024391. The authors would also like to thank Danny Sorensen for his helpful suggestions and comments.

## Appendix A

Given Equation (12), the pseudo-inverse $J^+$ can be computed as [11]

$$J^+ = VD^+U^T \tag{30}$$

where $D^+$ is an $n \times m$ diagonal matrix with

$$\begin{aligned} D(i,i) &= 1/\sigma_i, && \text{if } \sigma_i \neq 0 \\ &= 0 && \text{if } \sigma_i = 0 \end{aligned}$$

Thus, as a singularity is approached the smaller singular values of $J$ tend to go to zero resulting in very high singular values of $J^+$. Also, when the singularity is reached, there is a discontinuous switch as the large singular values of $J^+$ drop from a very high value to zero. In terms of the SVD, the singularity robust inverse $J^*$ is given by

$$\begin{aligned} J^* &= J^T(JJ^T + \lambda I)^{-1} = VD^TU^T(UDD^TU^T + \lambda UU^T)^{-1} \\ &= VD^TU^T\{U(DD^T + \lambda I)^{-1}U^T\} \\ &= V\{D^T(DD^T + \lambda I)^{-1}\}U^T \end{aligned}$$

$$J^* = V \begin{bmatrix} \sigma_1/(\sigma_1^2+\lambda) & 0 & . & . & 0 \\ 0 & \sigma_2/(\sigma_2^2+\lambda) & . & . & 0 \\ . & & & . & 0 \\ . & & & . & \\ . & & & & \sigma_m/(\sigma_m^2+\lambda) \\ 0 & . & & . & 0 \\ 0 & . & & . & 0 \end{bmatrix} U^T \tag{31}$$

Thus, if the joint velocity vector is computed using the SRI as $\dot{\boldsymbol{\theta}}^* = J^*\dot{\mathbf{x}}$, then its norm can be computed from Equation (31) as

$$||\dot{\boldsymbol{\theta}}^*|| = \sqrt{\sum_{i=1}^{m} \frac{\sigma_i^2\gamma_i^2}{(\sigma_i^2+\lambda)^2}} \tag{32}$$

where $U^T\dot{\mathbf{x}} = [\gamma_1 \ldots \gamma_m]^T$ and $||Vy|| = ||y||$ for any $n$-vector $y$ since $V$ is an orthogonal matrix.

From Equation (30), the norm of the pseudo-inverse solution is given by

$$||\dot{\boldsymbol{\theta}}^+|| = ||J^+\dot{\mathbf{x}}|| = \sqrt{\sum_{i=1}^{m} \frac{\gamma_i^2}{\sigma_i^2}} \tag{33}$$

Comparing Equations (32) and (33), it is clear that the SRI yields an inverse kinematic solution which has norm less than that yielded by the pseudo-inverse. The price paid for this reduction in norm is that the end-effector is deviated away from the trajectory by the SRI. This deviation can be computed using the SRI and its norm is given by

$$||\dot{\mathbf{x}} - J\dot{\boldsymbol{\theta}}^*|| = \sqrt{\sum_{i=1}^{m} \left[\frac{\lambda}{\sigma_i^2+\lambda}\right]^2}$$

It is clear that for the pseudo-inverse ($\lambda = 0$), this error will be zero when the Jacobian has full row rank. Equation (31) indicates that the SRI will not suffer from the problem of discontinuous switching of solutions as the pseudo-inverse does. As a singular value $\sigma_i$ of $J$ goes to zero (when the manipulator is approaching a singular configuration), the term $\sigma_i/(\sigma_i^2+\lambda)$ increases till it reaches a maximum at $\sigma_i = \lambda$ and then decreases to zero. Thus there is no discontinuous jump in the solution and even though the end-effector may deviate away from the intented trajectory it will do so in a continuous fashion, without any sudden jerks.

Thus the SRI is capable of overcoming the two main limitations of the pseudo-inverse in the region of singular configurations i.e., infeasibly high joint velocities and discontinuous trajectory.

## References

1. Baillieul, J.: Kinematic programming alternatives for redundant manipulators, in *Proc. 1985 IEEE Int. Conf. Robotics and Automation*, St. Louis, MI, May 1885, pp. 722–728.
2. Baillieul, J., Hollerbach, J. and Brockett, R.: Programming and control of kinematically redundant manipulators, in *Proc. 23rd Conf. Decision and Control*, Las Vegas, NV, 1984, pp. 768–774.
3. Chan, S. K. and Lawrence, P. D.: General inverse kinematics with the error damped pseudo-inverse, in *Proc. 1988 IEEE Conf. Robotics and Automation*, Philadelphia, PA, 1988, pp. 834–839.
4. Chiaverini, S., Egeland, O. and Kanestrom, R. K.: Achieving user-defined accuracy with damped least-squares inverse kinematics, in *Proc. 5th Int. Conf. Advanced Robotics*, 1991, pp. 672–677.
5. Chiaverini, S., Egeland, O. and Kanestrom, R. K.: Weighted damped least-squares in kinematic control of robotic manipulators, To appear in *Advanced Robotics*, Paper AR No. 227.
6. Deo, A. S.: Application of optimal damped least-squares methods to inverse kinematics of robotic manipulators, Master's Thesis, Department of Electrical and Computer Engineering, Rice University, Houston, TX, April 1991.
7. Deo, A. S. and Walker, I. D.: Optimal damped least-squares for inverse kinematics of robot manipulators, in *Proc. 1991 SPIE Conf. Co-operative Intelligent Robotics in Space-II*: Vol. 1612, Boston, MA, 1991, pp. 123–138.
8. Deo, A. S. and Walker, I.D.: Robot subtask performance with singularity robustness using optimal damped least-squares, in *Proc. 1992 IEEE Int. Conf. Robotics and Automation*, Nice, May 1992, pp. 464–470.
9. Egeland, O., Sagli, J. R., Spangelo, I. and Chiaverini, S.: A damped least-squares solution to redundancy resolution, in *Proc. 1991 IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, April 1991, pp. 945–950.
10. Goldenberg, A. A., Benhabib, B. and Fenton, R. G.: A complete generalized solution to the inverse kinematics of robots, *IEEE J. Robotics and Automation* **RA-1**(1) (1985), 14–20.
11. Golub, G. H. and Van Loan, C. F.: *Matrix Computation*, 2nd edn, Johns Hopkins Univ. Press, Baltimore, MD, 1989.
12. Hollerbach, J. M. and Suh, K. C.: Redundancy Resolution of Manipulators through Torque Optimization, *IEEE J. Robotics and Automation* **RA-3**(4) (1987), 308–316.
13. Kelmar, L. and Khosla, P. K.: Automatic generation of kinematics for a reconfigurable modular manipulator system, in *Proc. 1988 IEEE Conf. Robotics and Automation*, Philadelphia, PA, 1988, pp. 663–668.
14. Khatib, O.: A unified approach for motion and force control of robot manipulators: The operational space formulation, *IEEE J. Robotics and Automation* **RA-3**(1) (1987), 43–53.
15. Kirćanski, M. V.: Inverse kinematic problem near singularities for simple manipulators: Symbolical damped least-squares solution, in *Proc. 1993 IEEE Int. Conf. Robotics and Automation* **1** (1993), pp. 974–979.
16. Kirćanski, M. V.: Symbolical singular value decomposition for a 7-dof manipulator and its application to robot control, in *Proc. 1993 IEEE Int. Conf. Robotics and Automation* **3** (1993), pp. 895–900.
17. Klema, V. C. and Laub, A. J.: The singular value decomposition: its computation and some applications, *IEEE Trans. Automatic Control* **AC-25**(2) (1980), 164–176.
18. Lawson, C. L. and Hanson, R. J.: *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, NJ, 1974.

19. Liégeois, A.: Automatic supervisory control of the configuration and behavior of multibody mechanisms, *IEEE Trans. Systems, Man and Cybernetics* **SMC-7** (1977), 868–871.
20. Ma, S., Hirose, S. and Nenchev, D. N.: Improving local torque optimization techniques for redundant robotic mechanisms, *J. Robotic Systems* **8**(1) (1991), 75–91.
21. Maciejewski, A. A. and Klein, C. A.: Numerical filtering for the operation of robotic manipulators through kinematically singular configurations, *J. Robotic Systems* **5**(6) (1988), 527–552.
22. Maciejewski, A. A. and Klein, C. A.: The singular value decomposition: Computation and application to robotics, *Int. J. Robotics Res.* **8**(6) (1989), 63–79.
23. Maciejewski, A. A. and Klein, C. A.: Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments, *Int. J. Robotics Res.* **4**(3) (1985), 109–116.
24. Mayorga, R. V., Ma, K. S. and Wong, A. K. C.: A robust local approach for the obstacle avoidance of redundant robot manipulators, in *Proc. 1992 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 1992, pp. 1727–1734.
25. Mayorga, R. V., Wong, A. K. C. and Milano, N.: A fast damped least-squares solution to manipulator inverse kinematics and singularities prevention, in *Proc. 1992 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 1992, pp. 1177–1184.
26. Moré, J. J.: The Levenberg–Marquardt algorithm: implementation and theory, in G. A. Watson, ed., *Lecture Notes in Mathematics* **630**, Springer, Berlin, 1978, pp. 105–116.
27. Nakamura, Y.: *Advanced Robotics – Redundancy and Optimization*, Addison-Wesley Reading, MA, 1991.
28. Nakamura, Y. and Hanafusa, H.: Inverse kinematic solutions with singularity robustness for robot manipulator control, *ASME J. Dynamic Systems, Measurement and Control* **108** (1986), 163–171.
29. Nakamura, Y., Hanafusa, H. and Yoshikawa, T.: Task priority Based redundancy control of robot manipulators, *Int. J. Robotics Res.* **6**(2) (1987), 3–15.
30. Nenchev, D. N.: Redundancy resolution through local optimization: A review, *J. Robotic Systems* **6**(6) (1989), 769–798.
31. Paul, R. P., Shimano, Bruce and Mayer, G. E.: Differential kinematic control equations for simple manipulators, *IEEE Trans. Systems, Man and Cybernetics*, **SMC-11**(6) (1981), 456–460.
32. Seraji, H. and Colbaugh, R.: Improved configuration control for redundant robots, *J. Robotic Systems* **7**(6) (1990), 897–928.
33. Siciliano, B.: Kinematic control of redundant robot manipulators: A tutorial, *J. Intelligent and Robotic Systems* **3** (1990), 201–212.
34. Walker, I. D.: The use of kinematic redundancy in reducing impact and contact effects in manipulation, in *Proc. 1990 IEEE Conf. Robotics and Automation*, Cincinnati, OH, pp. 434–439.
35. Walker, I. D. and Marcus, S. I.: Subtask performance by redundancy resolution for redundant robot manipulators, *IEEE J. Robotics and Automation* **4**(3) (1988), 350–354.
36. Wampler, C. W.: Manipulator inverse kinematic solutions based on vector formulation and damped least-squares methods, *IEEE Trans. Systems, Man and Cybernetics* **SMC-16**(1) (1986), 93–101.
37. Wampler, C. W. and Leifer, L. J.: Applications of damped least-squares methods to resolved-rate and resolved-acceleration control of manipulators, *ASME J. Dynamic Systems, Measurement and Control* **110**(1) (1988), 31–38.
38. Whitney, D. E.: Resolved motion rate control of manipulators and human prostheses, *IEEE Trans. Man-Machine Systems* **MMS-10**(2) (1969), 47–53.

39. Yoshikawa, T.: Analysis and control of robot manipulators with redundancy, in Brady and Paul (eds), *Robotics Research – 1st Int. Symp.*, MIT Press, Cambridge, MA, 1985, pp. 735–747.
40. Yoshikawa, T.: Manipulability and redundancy control of robotic mechanisms, in *Proc. IEEE Int. Conf. Robotics and Automation*, St. Louis, MI, 1985, pp. 1004–1009.