MODELING, IDENTIFICATION AND CONTROL, 1993, VOL. 14, NO. 2, 107–119
doi:10.4173/mic.1993.2.5

Resolving redundancy through a weighted damped least-squares solution

JAN RICHARD SAGLI†, INGE SPANGELO† and OLAV EGELAND†

Keywords: *Robotics, redundant degrees of freedom, singularities, weighted damped least squares.*

Singularity robust redundancy resolution with task priority can be implemented using the extended Jacobian technique with weighted damped least-squares. The resulting scheme is simple to implement and involves less computation than the task priority scheme. The minimum singular value of the Jacobian can be estimated reliably and accurately with little computation, and this estimate was used to calculate an appropriate damping factor. A constant damping factor was also used with good results. The scheme was successfully implemented in a simulation study with a seven-joint manipulator with a kinematic design derived from the PUMA geometry.

1. Introduction

Many applications of manipulators with redundant degrees of freedom are conveniently described in terms of a primary end-effector task and a secondary task specifying the position coordinates of the internal motion. This formulation has the problem that artificial or algorithmic singularities are introduced.

Baker and Wampler (1988) solves the problem by using inverse kinematic functions defined on a singularity-free workspace. However it is not clear how to use this method for manipulators with seven joints of more. Baillieul (1985) included the secondary constraints in the task coordinate vector in the extended Jacobian scheme. The method is very simple, but it fails in artificial singularities where the extended Jacobian becomes singular even though the end-effector Jacobian has full rank. Nakamura, Hanafusa and Yoshikawa (1987) proposed the task-priority strategy which was further developed by Maciejewski and Klein (1985). This method is more computationally expensive than the extended Jacobian method, but it will always give a correct primary end-effector solution as long as the end-effector Jacobian has full rank. The artificial singularities are apparently eliminated with the task-priority scheme. However, the solution is ill-conditioned close to the artificial singularities due to the use of a pseudoinverse of a matrix which becomes rank deficient in the artificial singularities. This is solved in (Macejiewski and Klein, 1985) by calculating the efficient rank of the matrix, and treating it as singular wherever it would yield unacceptably large answers. This will give correct end-effector motion, while the internal motion is damped close to artificial singularities.

Received 20 January 1993.

† Center for Robotics Research, Division of Engineering Cybernetics, The Norwegian Institute of Technology, 7034 Trondheim, Norway.

This paper was presented at the IFAC Symposium on Robot Control, Vienna, 1991, and is reprinted with permission from Pergamon Press.

It would be desirable to have a solution which combines the computational simplicity of the extended Jacobian technique with the priority handling of the task-priority scheme and at the same time gives acceptable trajectories in singularities. We propose to use weighted damped least-squares to achieve this. The damped least-squares solution has been proposed for singularity-robust inverse kinematics (Nakamura and Hanafusa, 1986, Wampler, 1986). Damped least-squares could solve the problem of artificial singularities in the extended Jacobian technique, but the end effector position is not given priority in this scheme, so the error due to damping would then just as well be in end-effector coordinates as in the internal motion. However, the desired task priority property can be achieved in the extended Jacobian scheme using weighted damped least-squares where the low priority internal motion is given lower weight than the high priority end-effector motion. This will result in a computationally inexpensive task-priority scheme with singularity robustness. The method is analysed using the singular value decomposition, and the smallest singular value is used as a measure of how close the manipulator is to a singularity. A problem with the damped least-squares solution in inverse kinematic is to select the damping factor. Here there is an added problem due to the scaling. We have investigated the use of a constant damping factor (Wampler, 1986) and a damping factor calculated from an estimate of the smallest singular value (Maciejewski and Klein, 1988).

Seraji and Colbaugh (1990) simultaneously developed a scheme similar to the proposed method, focusing on the manipulability index (Yoshikawa, 1985) as a means for observing if the manipulator is close to being singular. However, the manipulability index does not necessarily give a good measure for the distance to a singularity (Golub and van Loan, 1989).

The scheme was tested out in simulations on a simple planar manipulator and a seven-joint manipulator derived from the PUMA geometry, and the results are reported in this paper.

2. Kinematics

The n-dimensional vector of joint coordinates is denoted by $\boldsymbol{q}$. Differential task-space motion is described by $\delta \boldsymbol{x}_t = \dot{\boldsymbol{x}}_t \delta t$ where δt is a small time increment and $\dot{\boldsymbol{x}}_t$ is an m-dimensional task-space velocity. The $m \times n$ task Jacobian $\boldsymbol{J}_t(\boldsymbol{q})$ is defined by

$$\dot{\boldsymbol{x}}_t = \boldsymbol{J}_t(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{1}$$

and incremental motion is given by

$$\delta \boldsymbol{x}_t = \boldsymbol{J}_t(\boldsymbol{q})\delta \boldsymbol{q} \tag{2}$$

where $\delta \boldsymbol{q} = \dot{\boldsymbol{q}} \delta t$.

In configurations where the task Jacobian $\boldsymbol{J}_t$ has full rank, the end effector has m degrees of freedom. When the Jacobian is rank deficient so that rank $(\boldsymbol{J}) = r$, $r < m$, the end effector has only r degrees of freedom and the manipulator is said to be in a singular configuration.

3. Internal motion

To specify the motion of the manipulator in all n degrees of freedom it is necessary to specify $n-m$ additional constraints. Here we use positional constraints by specifying $\boldsymbol{x}_c$ which is a vector of dimension $n-m$. The constraint Jacobian $\boldsymbol{J}_c$ is defined by

$$\delta \boldsymbol{x}_c = \boldsymbol{J}_c(\boldsymbol{q})\delta \boldsymbol{q} \tag{3}$$

A n-dimensional task increment δx can now be defined:

$$\delta x=\begin{bmatrix}\delta x_t\\ \delta x_c\end{bmatrix} \tag{4}$$

and the $n\times n$ extended Jacobian J defined by (Baillieul, 1985, Egeland, 1987)

$$\delta x=J(q)\delta q=\begin{bmatrix}J_t(q)\\ J_c(q)\end{bmatrix}\delta q \tag{5}$$

The extended Jacobian solution (Baillieul, 1985) is to calculate the joint motion δq from

$$\delta q_{EJ}=J^{-1}(q)\delta x \tag{6}$$

using Gaussian elimination. If any of the rows in the constraint Jacobian J_c becomes linearly dependent of the rows in the task Jacobian J_t, $J(q)$ becomes rank deficient, and no solution δq will exist unless δx is in the range space of J. If this happens in a configuration where the task Jacobian J_t has full rank which means that the end-effector can move in m degrees of freedom, the manipulator is said to be in an artificial singularity. The main drawback with the extended Jacobian technique has been its poor performance with respect to artificial singularities.

In the task priority scheme (Nakamura *et al.*, 1987), the solution is

$$\delta q_{TP}=J_t^{\dagger}\delta x_t+(I-J_t^{\dagger}J_{tc}(I-J_t^{\dagger}J_t)]^{\dagger}(\delta x_c-J_cJ_t^{\dagger}\delta x_t) \tag{7}$$

which can be simplified to (Maciejewski and Klein, 1985)

$$\delta q_{TP}=J_t^{\dagger}\delta x_t+[J_c(I-J_t^{\dagger}J_t)]^{\dagger}(\delta x_c-J_cJ_t^{\dagger}\delta x_t) \tag{8}$$

From Eqn (7) it is easily seen that

$$J_t\delta q_{TP}=\delta x_t \tag{9}$$

which means that if J_t is full rank the end-effector motion is always correct independently of J_c. This apparently means that the problem of artificial singularities has been solved, but this is only partly true. The scheme involves taking the pseudoinverse of the matrix $J_c(I-J_t^{\dagger}J)$ which is rank deficient in artificial singularities. In fact, if the task Jacobian J_t is full rank, this matrix is full rank if and only if the extended Jacobian J is full rank. This can be shown as follows:

Proof

When $J=\begin{bmatrix}J_t\\ J_c\end{bmatrix}$ we have by the definition of the null space

$$\mathcal{N}(J)=\mathcal{N}(J_t)\cap\mathcal{N}(J_c) \tag{10}$$

which means

$$J \text{ is full rank} \Leftrightarrow \mathcal{N}(J_t)\cap\mathcal{N}(J_c)=\{0\} \tag{11}$$

The matrix $J_c(I-J_t^{\dagger}J_t)$ is of dimension $(n-m)\times n$, which means that it has full rank, if and only if,

$$\dim\mathcal{N}(J_c(I-J_t^{\dagger}J_t))=m \tag{12}$$

Generally we have

$$\mathcal{N}(AB)=\mathcal{N}(B)+\mathcal{R}(B)\cap\mathcal{N}(A) \tag{13}$$

Since

$$\mathcal{R}(I-J_t^{\dagger}J)=\mathcal{N}(J_t) \tag{14}$$

we now have

$$\mathcal{N}(J_c(I-J_t^{\dagger}J_t))=\mathcal{N}(I-J_t^{\dagger}J_t)+\mathcal{N}(J_t)\cap\mathcal{N}(J_c) \tag{15}$$

When J_t is full rank

$$\dim\mathcal{N}(I-J_t^{\dagger}J_t)=\dim\mathcal{R}(J_t)=m \tag{16}$$

Eqs. (15) and (16) gives

$$\dim\mathcal{N}(J_c(I-J_t^{\dagger}J_t))=m+\dim(\mathcal{N}(J_t)\cap\mathcal{N}(J_c)) \tag{17}$$

Combining Eqns (11), (12) and (17) completes the proof. □

This means that the task-priority scheme as given by Eqn. (7) or Eqn. (8) gives the same solution as the extended Jacobian scheme in Eqn. (5) except exactly in the singularities. Close to the artificial singularities also the task priority scheme is ill conditioned, and special routines have to be used (Maciejewski and Klein, 1985). However, the scheme has the advantage that if the matrix $J_c(I-J_t^{\dagger}J_t)$ is properly damped, the errors due to damping close to artificial singularities will be purely in the secondary constraint directions.

4. Scaling of the Jacobian matrix

Whether the extended Jacobian $J(q)$ is close to being singular or not must be considered with respect to the order of magnitudes we are working with.

The conditioning number κ of a matrix A is defined as

$$\kappa=\|A\|\,\|A^{-1}\|=\frac{\sigma_1}{\sigma_n} \tag{18}$$

where σ_1 and σ_n are the largest and smallest singular values of A, and a large condition number indicates that the matrix is nearly singular.

By scaling the Jacobian matrix such that σ_1 is within the magnitude of one, also σ_n will be a measure for how close $J(q)$ is to being singular. Wampler (1986) suggested to scale the Jacobian matrix with the length of the arm, which will constrain σ_1 such that

$$2^{1/2}\leqslant\sigma_1<(2n)^{1/2} \tag{19}$$

for a six-degree-of-freedom manipulator ($n=6$). We will show this as follows:

Proof

The Jacobian matrix for a manipulator with $n=6$ rotational degrees of freedom is (Whitney, 1972)

$$J_t=\begin{bmatrix} b_1 & b_2\ldots b_n \\ a_0 & a_1\ldots a_{n-1}\end{bmatrix} \tag{20}$$

where a_i is the approach vector of the transformation matrix

$${}^0_iR=[n_i \quad s_i \quad a_i] \tag{21}$$

from frame i to frame 0, where $\|a_i\|=1$. Further

$$b_i=a_{i-1}\times l_i \tag{22}$$

where

$$l_i \sum_{k=i}^{n} {}^0\boldsymbol{p}_i \tag{23}$$

and ${}^0\boldsymbol{p}_i$ is the offset from frame $i-1$ to frame i in 0-coordinates. If the Jacobian is scaled by dividing all lengths with the maximum reach of the manipulator, $l_{1,n}$, where the largest element is less than or equal to one.

The largest singular value of $\bar{\boldsymbol{J}}_t(\boldsymbol{q})$ we get a new Jacobian $\bar{\boldsymbol{J}}_t(\boldsymbol{q})$ is obviously never less than $2^{1/2}$ since

$$\sigma_1 = \|\bar{\boldsymbol{J}}_t\| \geqslant \|2\boldsymbol{a}_i\| 2\boldsymbol{a}_i\| = 2^{1/2} \tag{24}$$

Generally we have for a matrix $\boldsymbol{A}$ that (Golub and van Loan, 1989)

$$\|\boldsymbol{A}\| \leqslant \|\boldsymbol{A}\|_F \tag{25}$$

where $\|\boldsymbol{A}_F = (\sum_{\forall i,j} A_{ij}^2)^{1/2}$ is the Frobenius norm of $\boldsymbol{A}$. This yields

$$\begin{aligned} \|\boldsymbol{J}_t\|_F &= \left(n + \sum_{i=1}^{n} \|\boldsymbol{b}\|^2 / l_{i,n}^2\right)^{1/2} \\ &\leqslant \left(n + 1 + \left(\sum_{i=2}^{n} l_{i,n}/l_{1,n}\right)^2\right)^{1-2} < (2n)^{1/2} \end{aligned} \tag{26}$$

where $l_{i,n}$ is the maximum length from joint $i-1$ to the end-effector.

This finally gives

$$2^{1/2} \leqslant \sigma_1 < (2n)^{1/2} \tag{27}$$

and we can conclude that the smallest singular value, σ_m, is a sufficient measure for the vicinity of a singularity, as σ_1 is constrained to be within the magnitude of one. □

Condition (19) was shown for a manipulator with six degrees of freedom. By using similar arguments it can be shown that translational joints will further constrain the attainable interval of σ_1. The arguments can also be used on the extended Jacobian $\boldsymbol{J}(\boldsymbol{q})$ for manipulators with redundant degrees of freedom.

5. Redundancy resolution with task priority

The solution (6) is not possible in singular configurations, and the solution becomes ill-conditioned close to singularities.

Nakamura and Hanafusa (1986) and Wampler (1986) independently proposed to use the damped least-squares solution of Eqn. (6) in the inverse kinematics algorithm. This method was further developed by Maciejewski and Klein (1988). The solution minimizes

$$\mathscr{L} = \|\delta \boldsymbol{x} - \boldsymbol{J}\delta q\|^2 + \lambda^2 \|\delta \boldsymbol{q}\|^2. \tag{28}$$

Task priority can be achieved with the augmented Jacobian technique if damped least squares is used with weighting. The constraints can be given lower priority by using small weights for constraints and high weights for the primary task motion. Weighted damped least-squares was proposed for inverse kinematics of nonredundant manipulators in (Nakamura and Hanafusa, 1986), but it has not been used in redundancy resolution.

A weighted task increment is defined by

$$\delta\tilde{x} = W_w \delta x = \tilde{J}\delta q \tag{29}$$

where

$$\tilde{J} = W_w J = \begin{bmatrix} J_t \\ W_c J_c \end{bmatrix} \tag{30}$$

where $W_c = I$ is the weighting matrix of the constraints.

The weighted damped least-squares solution minimizes

$$\mathscr{L} = \|\delta x_t - J_t \delta q\|^2 + w^2 \|\delta x_c - J_c \delta q\|^2 + \|\delta q\|^2 w_\lambda \tag{31}$$

$$= (\delta x_t - J_t \delta q)^{\mathrm{T}} (\delta x_t - J_t \delta q) \tag{32}$$

$$+ w^2 (\delta x_c - J_c \delta q)^{\mathrm{T}} (\delta x_c - J_c \delta q) + \delta q^{\mathrm{T}} W_\lambda \delta q$$

and the solution is again found from

$$(\tilde{J}^{\mathrm{T}} \tilde{J} + W_\lambda)\delta q = \tilde{J}^{\mathrm{T}} \delta\tilde{x} \tag{33}$$

which in matrix form can be written out as

$$\delta q = (J_t^{\mathrm{T}} J_t + J_c^{\mathrm{T}} W_c J_c + W_\lambda)^{-1} (J_t^{\mathrm{T}} \delta x_t + J_c^{\mathrm{T}} W_c \delta x_c) \tag{34}$$

The singular value decomposition of $\tilde{J}(q)$ is written

$$\tilde{J} = \sum_{i=1}^{n} \tilde{\sigma}_\kappa u_i v_i^{\mathrm{T}} \tag{35}$$

If $W_\lambda = \lambda^2 I$, the solution in terms of the singular value decomposition is

$$\delta q = \sum_{i=1}^{n} \frac{\tilde{\sigma}_i}{\tilde{\sigma}_i^2 + \lambda^2} \tilde{v}_i \tilde{u}_i^{\mathrm{T}} \delta x \tag{36}$$

which in practice can be implemented as

$$\delta q = \sum_{i=1}^{n} \frac{\tilde{\sigma}_i}{\tilde{\sigma}_i^2 + \lambda^2} \tilde{v}_i \tilde{u}_i^{\mathrm{T}} (\delta x + K e_x) \tag{37}$$

to ensure zero task space error after a period with deviation due to weighting, i.e. a proportional term, Ke_x is included in addition to the derivative term δx for convergence purposes.

The singular values $\tilde{\sigma}_i$ and the singular vectors $\tilde{v}_i$ and $\tilde{u}_i$ will depend on the weighting matrix W_w. This has no impact on the solution δq as long as λ is zero, but close to singularities where $\lambda > \tilde{\sigma}_{r+1}$ for some $r < n$ the solution can be shaped by selecting proper weights.

We propose to do this as follows:

1. The problem is first formulated so that the maximum singular value is within the magnitude of one as proposed in §4. The attainable region for σ_1 can easily be calculated exploiting Eqn. (26) or a similar one derived for the actual manipulator.

2. Then the weighting of Eqn. (29) is introduced. The weighting matrix W_c is selected as

$$W_c = wI \tag{38}$$

with $w \ll 1$. For full rank J_t this will shape the output singular vectors $\tilde{u}_i$ so that span $\{u_1, \dots u_m\}$ will tend to be in

$$\mathcal{R}\begin{bmatrix} J_t \\ 0 \end{bmatrix},$$

or in other words: The m first output singular vectors will mainly contain components in the task directions. The motivation for varying w may be to obtain smoother motion in the transition area when the manipulator approaches a singularity and also the damping matrix W_λ is changed from zero. However, this is handled by the continuously changing λ.
3. The damped least-squares solution is used with $\lambda < w$.

In the artificial singularities this will result in accurate motion in the task directions spanned by $\tilde{u}_1, \dots, u_m$ as $\tilde{\sigma}_i^2 \gg \lambda^2$ for $i = 1, \dots, m$ if $W_\lambda = \lambda^2 I$. The damping will only give inexact motion in the directions spanned by $\tilde{u}_{m+1}, \dots, \tilde{u}_n$ which are orthogonal to the task directions. This is the desired task-priority property.

Another advantage with the scheme is that it will also work properly in true singularities where $J_t(q)$ is rank deficient and the end-effector loses degrees of freedom. This can also be achieved with the task priority scheme, but then a damped pseudoinverse must be computed both for $J_t(q)$ and $J_c(I - J_t^\dagger J_t)$ which requires a lot of computation.

6. The effect of weighting on the damping error

In the case of no damping, that is with $\lambda = 0$, the solution with and without weighting is obviously the same. However, with damping it is evident from Eqn. (31) that the solution can be shaped by weighting so that the errors due to damping are mainly in the constraint directions. We have analysed this using the singular value decomposition.

The error in joint coordinates due to damping is

$$\tilde{e}_q = \sum_{i=1}^{n} \frac{\lambda^2}{\tilde{\sigma}_i(\tilde{\sigma}_i^2 + \lambda^2)} v_i \tilde{u}_i^{\mathrm{T}} \delta x \tag{39}$$

The error in $\tilde{x}$ is then $\tilde{e}_r = \tilde{J} e_q$, by inserting Eqn. (35) in Eqn. (39) this can be written

$$\tilde{e}_r = \sum_{i=1}^{n} \frac{\lambda^2}{\tilde{\sigma}_i^2 + \lambda^2} \tilde{u}_i \tilde{u}_i^{\mathrm{T}} \delta \tilde{x} \tag{40}$$

which can be approximated by

$$\tilde{e}_r = \sum_{i=m+1}^{n} \frac{\lambda^2}{\tilde{\sigma}_i^2 + \lambda^2} \tilde{u}_i \tilde{u}_i^{\mathrm{T}} \delta \tilde{x} \tag{41}$$

if it is assumed that $\tilde{\sigma}_i \gg \lambda$ for $i \in \{1, \dots, m\}$. It is clearly seen that the error will mainly be in x_c as the vectors $\tilde{u}_i$, $i \in \{m+1, \dots, n\}$ are orthogonal to the task directions in

$$\mathcal{R}\begin{bmatrix} J_t \\ 0 \end{bmatrix}.$$

7. Implementation aspects

In well-conditioned configurations the singular values associated with the task directions will be close to unity while the singular values associated with the constraint directions will close to w. This means that the damping factor λ must be less than the weighting factor w to give an accurate solution in well conditioned configurations. If λ becomes too small, there may be problems with the conditioning of the weighted damped least-squares problem close to singularities. A good solution seems to be to use a variable damping factor. We have therefore used the method proposed by Maciejewski and Klein (1988) to estimate the smallest singular value.

7.1. *Estimation of minimum singular value*

The method is based on the fact that after repeated calculations of

$$\boldsymbol{y}^k = \boldsymbol{A}\boldsymbol{y}^{k-1} \tag{42}$$

where $\boldsymbol{A}$ is an $n \times n$ matrix and $\boldsymbol{y}^k$ is the value of the vector $\boldsymbol{y}$ after kth iteration, the solution will be dominated by the eigenvector corresponding to the largest eigenvalue of A. This can be seen by first assuming that the initial vector $\boldsymbol{y}^0$ is spanned by the eigenvectors $\boldsymbol{m}_i$ of $\boldsymbol{A}$, i.e.

$$\boldsymbol{y}^0 = \sum_{i=1}^{n} c_i m_i \tag{43}$$

The eigenvalues of $\boldsymbol{A}$ are denoted α_i, and are ordered such that $\alpha_1 \geqslant \alpha_2 \geqslant \ldots \geqslant \alpha_n$. Thus

$$\begin{aligned} \boldsymbol{y}^k &= \boldsymbol{A}^k \boldsymbol{y}^0 \\ &= \sum_{i=1}^{n} c_i \alpha_i^k m_i \\ &\approx c_1 \alpha_1^k \boldsymbol{m}_l \end{aligned} \tag{44}$$

and it is clear that unless $c_1 = 0$, y^k will be dominated by the element associated with the maximum eigenvalue. An estimate $\hat{\alpha}_1$ of the largest eigenvalue can therefore be found as

$$\hat{\alpha}_1 = \left(\frac{\|y_k\|}{c_1 \|\boldsymbol{m}_1\|} \right)^{1/k} \tag{45}$$

If the initial value $\boldsymbol{y}^0 \approx m_1$, a good estimate $\hat{\alpha}_1$ can already be obtained after one iteration remembering the eigenvector/eigenvalue definition

$$\boldsymbol{A}\boldsymbol{m}_1 = \alpha_1 \boldsymbol{m}_1 \tag{46}$$

Similarly, an estimate of the smallest eigenvalue α_n of $\boldsymbol{A}$ can be found through inverting the above relation such that

$$\boldsymbol{y}^k = \boldsymbol{A}^{-1} \boldsymbol{y}^{k-1} \tag{47}$$

because the maximum eigenvalue of $\boldsymbol{A}^{-1}$ will be the reciprocal of the minimum eigenvalue of $\boldsymbol{A}$.

The estimation procedure can be extended to the SVD case for robot control since the singular values of $\boldsymbol{J}$ are the square roots of the eigenvalues of $\boldsymbol{J}^T\boldsymbol{J}$, or equivalently of $\boldsymbol{J}\boldsymbol{J}^T$. If now the Cholesky decomposition has been performed for the calculation of $\delta \boldsymbol{q}$ from Eqn. (33), this can also be utilized for estimation of the minimum singular value σ_n and the input singular vector $\boldsymbol{v}_n$ from

$$(\boldsymbol{J}^T\boldsymbol{J} + \boldsymbol{W}_\lambda)\boldsymbol{v}'_n = \left(\sum_{i=1}^{n} (\sigma_i^2 + \lambda^2) \boldsymbol{v}_i \boldsymbol{v}_i^T \right) \hat{\boldsymbol{v}}'_n = \hat{\boldsymbol{v}}_n^p \tag{48}$$

where $\hat{v}_n^p$ is the normalized estimate

$$\hat{v}_n = \frac{\hat{v}'_n}{\|\hat{v}'_n\|} \tag{49}$$

from the previous sample, and $W_\lambda = \lambda^2 I$. The prime is used to emphasize that $\hat{v}'_n$ is not yet normalized estimate of $\hat{v}_n$.

Now

$$\left(\sum_{i=1}^{n}(\sigma_i^2+\lambda^2)v_i v_i^{\mathrm{T}}\right)\hat{v}_n \approx (\hat{\sigma}_n^2+\lambda^2)\hat{v}_n = \hat{v}_n^p \tag{50}$$

which gives

$$\hat{\sigma}_n = \left(\frac{\|\hat{v}_n^p\|}{\|\hat{v}_n\|} - \lambda^2\right)^{1/2} = \left(\frac{1}{\|\hat{v}_n\|} - \lambda^2\right)^{1/2} \tag{51}$$

The proposed technique for maintaining an estimate of the smallest singular value requires that $v_n^p \approx \pm v_n^p$ if there is only one small singular value. In the case of several small singular values, $\hat{v}_n^p$ must mainly be spanned by the input singular vectors v_i^p corresponding to these small singular values. This is possible if the initial $\hat{v}_n$ is set to v_n which can be calculated off-line from a full singular value decomposition. Then, when the smallest singular value σ_n changes along the trajectory, the corresponding estimated input singular vector $\hat{v}_n$ will rotate so that a strong component always lies within the subspace spanned by the smallest singular vectors.

7.2. *Determination of weighting and damping factor*

Utilizing the estimate of the smallest singular value, the damping factor then was calculated from

$$\lambda^2 = \begin{cases} 0 & \text{when } \hat{\sigma}_n > \underline{\sigma} \\ \underline{\sigma}^2 - \hat{\sigma}^2 & \text{otherwise} \end{cases} \tag{52}$$

which gives

$$\frac{\tilde{\sigma}_n}{\tilde{\sigma}^2+\lambda^2} = \begin{cases} {}^1/\tilde{\sigma}_n & \text{when } \hat{\sigma}_n > \underline{\sigma} \\ \tilde{\sigma}_n/\underline{\sigma}^2 & \text{otherwise} \end{cases} \tag{53}$$

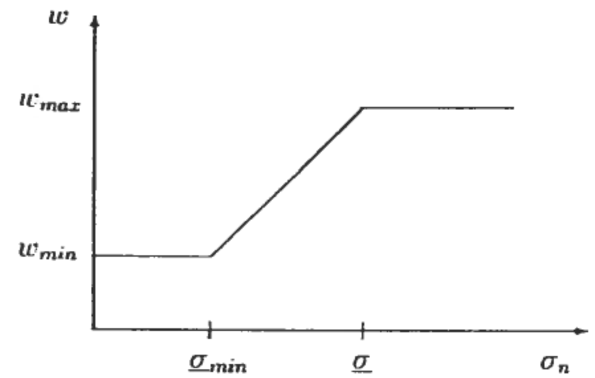

Figure 1. Possible varying choice of weighting w as in Eqn. (54).

It is also possible to use a varying w which is dependent on the size of the smallest singular value. Inspired by the choice of the damping factor from Eqn. (52), w can be calculated as

$$w=\begin{cases} w_{\max} & \text{if } \sigma<\sigma_n \\ \dfrac{w_{\max}-w_{\min}}{\underline{\sigma}-\underline{\sigma}_{\min}}(\sigma_n-\sigma)+w_{\max} & \text{if } \underline{\sigma}_{\min}\leqslant\sigma_n\leqslant\underline{\sigma} \\ w_{\min} & \text{otherwise} \end{cases} \tag{54}$$

The scheme is roughly illustrated in Fig. 2.

The simulations were performed on the four-degree-of-freedom experimental manipulator shown schematically in Fig. 2. The manipulator consists of a translational joint in the base and three rotational joints, with Denavit–Hartenberg parameters given in Table 1. If the primary task is to position the end-effector in x- and y-position, i.e.

$$\mathbf{x}_c=\begin{bmatrix} x \\ y \end{bmatrix} \tag{55}$$

the only true singularity occurs when $q_2=q_3=q_4=0$. This means that the manipulator is stretched out in x-direction, and is therefore an external singulatory.

For this manipulator, and the choice of primary task as in Eqn. (55), there are several possibilities for additional coordinates in the augmented task space approach.

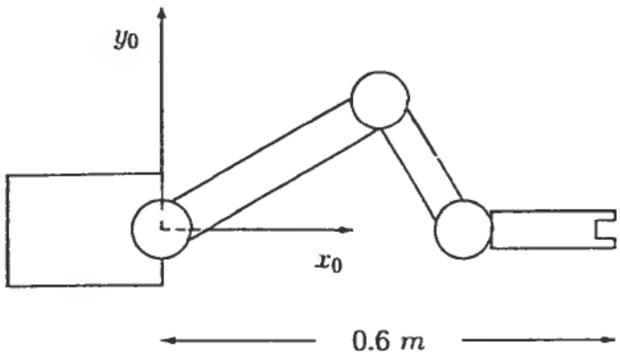

Figure 2. Schematic 4-DOF manipulator in its initial positions for the simulations.

Denavit–Hartenberg parameters					
Joint	θ	α	a	d	mass
1	0	0	q_1	0	31 kg
2	0	0	0·4 m	0	2 kg
3	0	0	0·2 m	0	2 kg
4	0	0	0·2 m	0	1 kg

Table 1. Denavit–Hartenberg parameters and link masses for planar experimental manipulator.

In the proceeding example the choice was

$$x_c = \begin{bmatrix} \theta \\ q_1 \end{bmatrix} \tag{56}$$

where $\theta = q_2 + q_3 + q_4$ is the end-effector orientation. The redundancy resolution was simply chosen as

$$\theta = 0 \tag{57}$$

$$q_1 = x - 0{\cdot}6\,\mathrm{m} \tag{58}$$

while the parameters in the weighted damped least-squares algorithm were $w = 0{\cdot}1$ and $\underline{\sigma} = 0{\cdot}1$. The matrix W_λ was chosen as $W_\lambda = \lambda^2 I$.

Figure 3 qualitatively shows how the manipulator moved during the simulation period, from the initial position

$$q(t=0) = [0 \quad 0{\cdot}5054 \quad -1{\cdot}8235 \quad 1{\cdot}3181]^T \tag{59}$$

The references for the end-effector were

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{cases} \begin{bmatrix} -t + 0{\cdot}6 \\ t \end{bmatrix} (\mathrm{m}) & \text{if } 0\ \mathrm{s} < t < 0{\cdot}65\ \mathrm{s} \\ \begin{bmatrix} -0{\cdot}05 \\ 0{\cdot}65 - t \end{bmatrix} (\mathrm{m}) & \text{if } 0{\cdot}65\ \mathrm{s} \leqslant t < 1{\cdot}3\ \mathrm{s} \end{cases}$$

It can be seen from Figure 3 how the constraints on x_c were satisfied as long as this did not conflict with the pimrary goal, which was to position the end-effector. However, when x_t and x_c could not be achieved simultaneously, the secondary task motion got less priority through the combined action of weighting and damping. This is more distinctly illustrated in Fig. 4, where the reference from the redundancy resolution (58) is shown together with the actual q_1. It is clearly seen that as the artificial singularity was approached, the reference tracking got reduced priority. The error was eliminated when this again became feasible. The same was the case for the end-effector orientation θ shown in Fig. 5. The smallest singularity σ_4 and corresponding calculated λ is shown in Fig. 6.

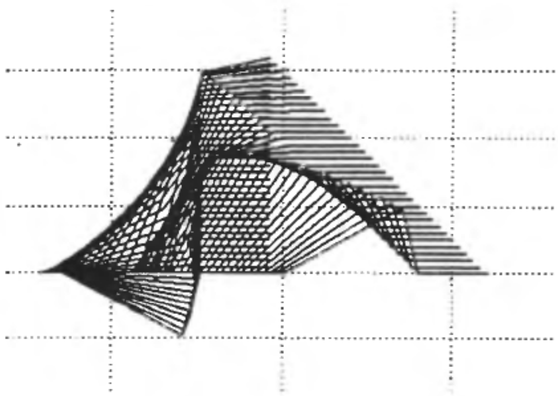

Figure 3. 'Animation' of planar manipulator.

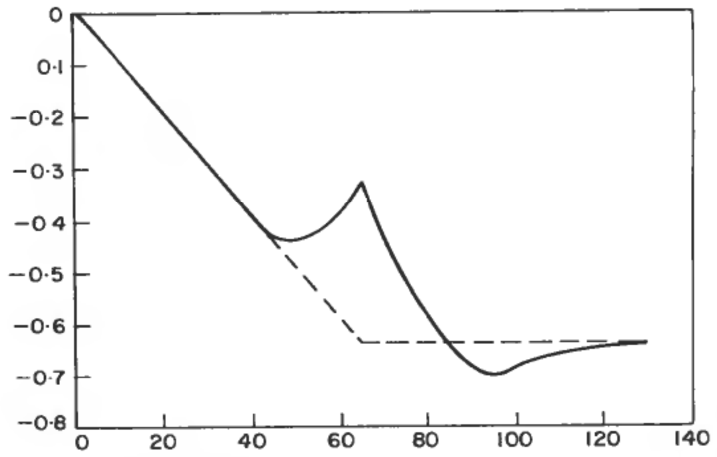

Figure 4. Reference (dashed) and actual position for q_1.

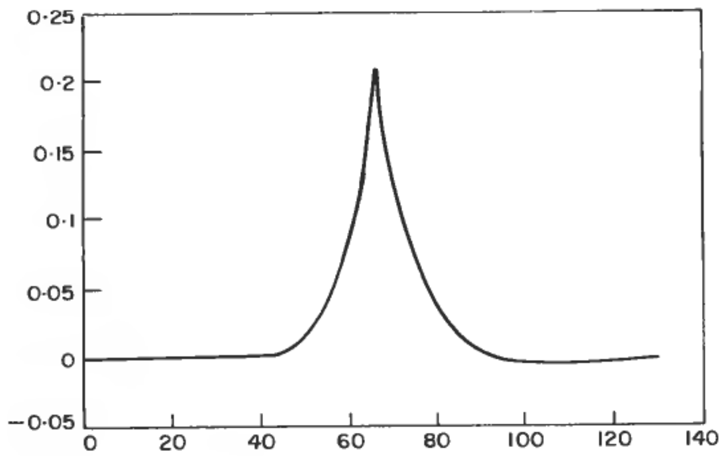

Figure 5. End-effector orientation θ.

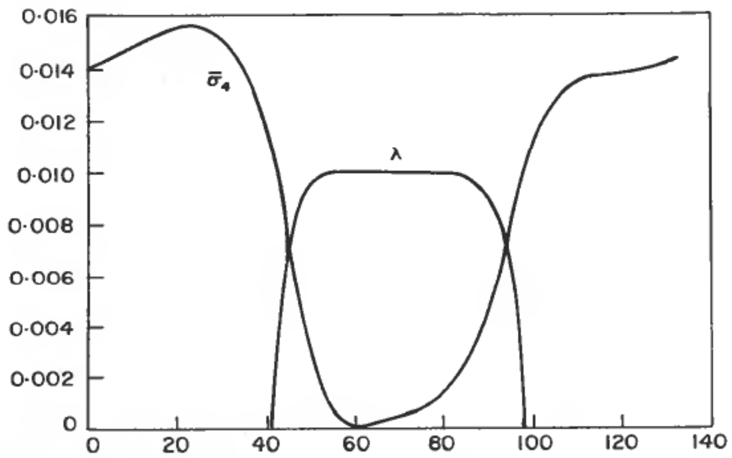

Figure 6. The minimum singular value ρ_1 and corresponding damping factor λ.

8. Conclusions

A method for handling artificial singularities for manipulators with redundant degrees of freedom has been proposed and tested in simulations. The method includes task priority based on a weighted damped least-squares strategy, which is a generalized version of the ordinary damped least-squares method.

Acknowledgments

This work was supported by the Royal Norwegian Council for Scientific and Industrial Research through the Center for Robotics Research at the Norwegian Institute of Technology.

References

Baillieul, J., (1985). Kinematic programming alternatives for redundant manipulators. In *Proc. IEEE Int. Conf. on Robotics and Automation*. St. Louis, Missouri. March 1985. Pp. 722–728.

Baker, D. R., and Wampler, C. W., 1988). On the inverse kinematics of robot manipulators with redundancy. *Int. J. of Robotics Research*, **7**, 3–21.

Egeland, O., (1987). Task-space tracking with redundant manipulators. *IEEE J. Robotics and Automation*, **3**, 471–475.

Golub, G. H., and van Loan, C. F., (1989). *Matrix Computations* (Johns Hopkins, London).

Maciejewski, A. A., and Klein, C. A., (1985). Obstacles avoidance for kinematically redundant manipulators in varying environments. *Int. J. of Robotics Research*, **4**, 99–117.

Maciejewski, A. A., and Klein, C. A., (1988). Numerical filtering for the operation of robotic manipulators through kinematically singular configurations. *J. Robotic Systems*. **5**, 527–552.

Nakamura, Y., and Hanafusa, H., (1986). Inverse kinematic solutions with singularity robustness for robot manipulator control. *ASME J. Dynamic Syst., Meas., Contr.*, **108**, 163–171.

Nakamura, Y., Hanafusa, H., and Yoshikawa, T., Task-priority based redundancy control of robot manipulators. *Int. J. of Robotic Research*, **6**, 3–15.

Seraji, H., and Colbaugh, R., (1990). Improved configuration control for redundant robots. *J. of Robotic Systems*, **7**, 897–928.

Wampler, C. W., (1986). Manipulator inverse kinematic solutions based on vector formulations and damped least-squares method. *IEEE Trans. Syst., Man and Cybern.*, **16**, 93–101.

Whitney, D. E., (1972). The mathematics of coordinated control of prosthetic arms and manipulators. *J. Dynamics, Measurement and Control*, **94**, 303–309.

Yoshikawa, T., (1985). Manipulability and redundancy control of robotic mechanisms. In *Proc. IEEE Int. Conf. on Robotics and Automation*, St. Louis, Missouri. March 1985. Pp. 1004–1009.