Redundancy Resolution Based on Manipulability and Dexterity Measures with Application to REDIESTRO

Xi Ji Shao

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montreal, Quebec, Canada

January 1995

National Library of Canada

Bibliothèque nationale du Canada

Acquisitions and Bibliographic Services Branch

Direction des acquisitions et des services bibliographiques

395 Wellington Street
Ottawa, Ontario
K1A 0N4

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

ISBN 0-612-01350-2

Canada

ABSTRACT

Redundancy Resolution Based on Manipulability and Dexterity Measures
with Application to REDIESTRO

Xi Ji Shao

Many dexterity measures, based on either kinematics or dynamics, have been defined for characterizing manipulability and dexterity of manipulators. But so far, most of the work which deals with the application of these measures focuses on the design of manipulators. In this thesis, two applications of dexterity measures are developed for redundancy resolution of manipulators. One approach deals with an optimization problem based on kinematics while another is concerned with an optimization problem based on dynamics. Several dexterity measures based on both kinematics and dynamics are discussed and compared. The first approach based on kinematics concentrates on constrained optimization with a kinematic conditioning measure. In this approach, a conditioning measure of the Jacobian matrix is optimized while the end-effector of a manipulator tracks a desired trajectory. This ensures that the system sensitivity in both velocity and torque is reduced. The second optimization approach based on dynamics is concerned with optimal control involving the dynamic conditioning index (DCI). Optimal control theory is used to design an optimal control for a manipulator based on minimizing the DCI while the end-effector tracks a desired trajectory. An integral type performance index which results in a global optimization scheme is used. The optimal control problem is formulated as a two-point boundary-value problem. The effectiveness of the approaches is demonstrated by means of simulation for a three degrees of freedom redundant planar manipulator. Finally, simulation results for a seven degrees of freedom redundant isotropic manipulator, REDIESTRO, are presented.

To my Mother and Father

ACKNOWLEDGEMENTS

The author is greatly indebted to his supervisor, Professor R. V. Patel for the encouragement and guidance throughout the course of this investigation and for the financial support.

Thanks are also due to all the member of the computer support group for their technical help.

Finally, the author is grateful to all colleagues and friends who have contributed to their effort in the form of advice, help and suggestions.

Table of Contents

CHAPTER 1 INTRODUCTION

1.1 Background

After the first robot began working in a factory in the early 1960's, applications of and research on robot manipulators have developed rapidly. Today manipulators are used in the factory, office, field and many other aspects of our daily lives. Manipulators are becoming more and more important, especially in space, deep ocean and hazardous environments. There is no doubt that manipulators play an important role in the modern world. Manipulators have been developed for different uses. For example, "industrial" manipulators are employed in a wide range of manufacturing processes such as parts assembling, materials handing and material welding. Hazardous material handing manipulators have been used to remove bombs and hazardous material.

In general, manipulators can be classified as kinematically nonredundant and kinematically redundant based on the number of degrees of freedom. A manipulator is said to be kinematically redundant if it possesses more degrees of freedom than necessary for performing a specified task. Otherwise, it is said to be nonredundant. For example, in two-dimensional space, a planar manipulator with three joints is redundant for achieving end-effector positions in the two-dimensional space without considering end-effector orientation, but it is nonredundant for a task involving both position and orientation of the end-effector. Similarly, a manipulator with seven or more joints in the three-dimensional space is redundant because six degrees of freedom are sufficient to achieve any arbitrary position and orientation of the end-effector in the manipulator's workspace.

Nonredundant manipulators are restricted in some applications such as singularity avoidance and obstacle avoidance. By contrast, redundant manipulators possess dexterity and versatility because of the infinite number of joint motions which result in no end-effector motion. This characteristic can be considered in various applications such as singularity avoidance [1] - [4], obstacle avoidance [5] - [8], joint limit avoidance [9][10] and energy minimization [11] - [13]. On the other hand, redundant manipulators have several disadvantages. For example, redundant manipulators have more joints and actuators, more complex structure, and more complicated control algorithms.

In any case, since redundant manipulators provide dexterity and versatility in applications, research on redundant manipulators is an active area. In recent years a number of different approaches have been proposed for development of redundant manipulators. Developing techniques to achieve end-effector trajectory control while satisfying additional task performance is known as the redundancy resolution problem. The approaches for redundancy resolution for manipulators can be classified based on several viewpoints. For instance, one may consider particular applications and classify these approaches based on the use of redundancy, such as singularity avoidance [1] - [4], obstacle avoidance [5] - [8], and joint torque minimization [11] - [13]. In addition, based on the control level, one may classify these approaches into three categories: redundancy resolution at the position level, redundancy resolution at the velocity level and redundancy resolution at the acceleration level [14]. Moreover, one may classify these approaches based on mathematical formulations: local optimization and global optimization [14]. In all these approaches for redundancy resolution to date, it seems that very few approaches consider redundancy resolution based on well-conditioned control for redundant manipulators. For example, when the Jacobian matrix is used as the transformation matrix which maps the end-effector velocity to the joint velocity, not much attention has been paid to the error propagation caused by the conditioning of the Jacobian matrix while the end-effector tracks a desired trajectory. It is well-known that there are a number of advantages, such as good servo accuracy, noise rejection and singularity avoidance, when a manipulator works in well-conditioned configurations. Therefore, in this thesis, we take into account the kinematic and dynamic conditioning of redundant manipulators by developing redundancy resolution schemes that incorporate optimization of kinematic and dynamic measures.

1.2 Motivation and Objectives of the Thesis

For a redundant manipulator the kinematic transformation from joint space to task space is given by

$$x = f(q), \tag{1.1}$$

where x describes the end-effector position and orientation and q denotes the joint displacement. Differentiating equation (1.1) with respect to time, we obtain

$$\dot{x} = J(q)\dot{q}, \tag{1.2}$$

where $J(q) = \frac{\partial f}{\partial q}$ is the Jacobian matrix, $\dot{x}$ denotes the end-effector velocity and $\dot{q}$ represents the joint velocity. For given end-effector velocity $\dot{x}$, there are several approaches which solve equation (1.2) to obtain the joint velocity $\dot{q}$ and most of them involve an inverse of the Jacobian matrix. However, very few approaches take into account the relationship between the accuracy of the solution and the condition number of the Jacobian matrix. In addition, the relation between the joint torque τ and the end-effector force f_e is expressed as [15]

$$\tau = J^T(q)f_e \tag{1.3}$$

Salisbury and Craig [16] recognized that the condition number of the Jacobian matrix affects the error propagation, but they only considered the condition number of the Jacobian matrix in the kinematic design. Essentially, the condition number of the Jacobian matrix acts as a magnification factor for Cartesian space measurement errors.

Recently, Angeles, Ranjbaran and Patel [17] designed an isotropic seven-axes manipulator which achieves the minimum condition number for its Jacobian matrix at certain configurations. The main propose of such design is accuracy. Therefore, one of the objective of the thesis is to develop algorithms for redundancy resolution which ensure that the condition number of the Jacobian matrix is as small as possible while the end-effector tracks a desired trajectory. Hence the error propagation, which is affected by the condition number of the Jacobian, is reduced. We call this scheme redundancy resolution based on optimization of kinematic measures (RROKM).

In addition, the motion of a manipulator is described by the dynamic equation

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau \quad , \tag{1.4}$$

where $M(q)$, $C(q,\dot{q})\dot{q}$, $g(q)$ and τ denote the mass matrix, centrifugal and Coriolis torque, gravity torque and joint torque, respectively. Many dynamic control approaches which require expressing the dynamic equation in state-space form, and simulations which require obtaining the joint acceleration $\ddot{q}$, involve the inverse of the mass matrix. Ma and Angeles [18] defined the dynamic conditioning index (DCI) which takes into account the condition number of the mass matrix. Several design examples are given in [19]. Taken one step further, another objective of this thesis is to develop an approach which minimizes the DCI along a desired Cartesian space trajectory. Thus the system sensitivity is reduced when the mass matrix is used as a transformation matrix. We call this approach redundancy resolution based on optimization of dynamic measures (RRODM).

In this thesis, the local optimization technique will be used for the RROKM scheme and the optimal control theory will be used for the RRODM scheme. The local optimization is based on instantaneous information while the global optimization considers whole information. But, in general, global optimization requires extensive computation. To date, since software for solving local constraint optimization is available, we can solve RROKM to see how kinematic conditioning measure affects system sensitivity. But this software is not sufficient for redundancy resolution based on optimization of dynamic measures. Therefore, global optimization using optimal control theory is considered. solving RRODM we can also see how dynamic conditioning measures affect system sensitivity.

1.3 Thesis Outline

The remainder of this thesis consists of five chapters.

Chapter 2 gives an overview of some common approaches for redundancy resolution of manipulators. These approaches will be classified into two categories based on the mathematical formulation: local optimization and global optimization. Local optimization considers instantaneous information while global optimization takes into account information over the entire trajectory. Advantages and disadvantages of both local and global optimization are discussed. This overview of local and global optimization approaches is helpful for the development of the RROKM and RRODM schemes which are given in Chapter 3 and 4, respectively.

Chapter 3 deals with a local optimization problem for redundant manipulators: redundancy resolution based on optimizing kinematic measures (RROKM). Several kinematic dexterity measures of manipulators are discussed and compared. The conditioning measure of the Jacobian matrix is exploited in the RROKM scheme. This scheme optimizes the conditioning measure while the end-effector tracks a desired trajectory. This ensures that the system sensitivity is reduced when the Jacobian matrix is used as a transformation matrix. The approach is illustrated by an example consisting of a 3-D planar manipulator.

Chapter 4 concentrates on a global optimization problem for redundant manipulator: redundancy resolution based on optimizing dynamic measures (RRODM). While kinematic dexterity measures are based on the Jacobian matrix, dynamic dexterity measures consider the dynamics of a manipulator. Consideration of the dynamics is important for high speed and high accuracy motion. The dynamic conditioning index [18] is used in the RRODM scheme. Therefore, the redundancy is used to ensure that the mass matrix is as well conditioned as possible along a desired Cartesian space trajectory. The approach is illustrated by an example of a 3-D planer manipulator.

Chapter 5 is concerned with applications of both the RROKM and RRODM schemes. Both schemes are applied to an isotropic redundant manipulator, REDIESTRO that was designed and constructed at McGill University [17].

Finally, Chapter 6 concludes the thesis and suggests ideas for further research in this area.

References

[1] R. V. Mayorga and A. K. C. Wong, "A Singularities Avoidance Approach for the Optimal Local path Generation of Redundant Manipulator," *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1988, pp. 49-54.

[2] S. Chiaverine, B. Siciliano and O. Egeland, "Kinematic Analysis and Singularity Avoidance for a Seven-Joint Manipulator," *Proc. of American Control Conference*, Vol. 3, 1990, pp. 2300-2305.

[3] M. V. Kircanski and T. M. Petrovic, "Inverse Kinematic Solution for a 7 DOF robot with Minimal Computational Complexity and Singularity Avoidance," *Proc. of IEEE Int. Conf. on Robotics and Automation*, Vol. 3, 1991, pp. 2664-2669.

[4] G. L. Long and R. P. Paul, "Singularity Avoidance and the Control of an Eight-Rev-

olute-Joint Manipulator," *Int. Journal of Robotics Research*, Vol. 11, No. 6, 1992, pp. 503-515.

[5] R. Colbaugh, H. Seraji and K. L. Glass, "Obstacle Avoidance for Redundant Robots Using Configuration Control," *Journal of Robotic Systems*, 6(6), 1989, pp. 721-743.

[6] Z. Y. Guo and T. C. Hsia, "Joint Trajectory Generation for Redundant Robots in an Environment with Obstacles," *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 157-162.

[7] S. E. Khadem and R. V. Dubey, "A Global Cartesian Space Obstacle Avoidance Scheme for Redundant Manipulators," *Optimal Control Applications & Methods*, Vol. 12, No. 4, 1991, pp. 279-286.

[8] Z. Shiller and S. Dubowsky, "On Computing the Global Time-Optimal Motions of Robotic Manipulator in the Presence of Obstacles," *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 6, 1991, pp. 785-797.

[9] H. Seraji, "Improved Configuration Control for Redundant Robots," *Journal of Robotic Systems*, 7(6), 1990, pp. 897-928.

[10] T. F. Chan and R. V. Dubey, "A Weighted Least-Norm Solution Based Scheme for Avoiding Joint Limits for Redundant Manipulators," *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1993, pp. 395-402.

[11] M. A. Peshkin and A. C. Sanderson, "Minimization of Energy in Quasi-Static Manipulator," *IEEE Transactions on Robotics and Automation*, Vol. 5, No. 1, 1989, pp. 53-60.

[12] A. Nedungadi and K. Kazerouinian, "A Local Solution with Global Characteristics for the Joint Torque Optimization of a Redundant Manipulator," *Journal of Robotic Systems*, 6(5), 1989, pp. 631-654.

[13] R. V. Mayorga, A. K. C. Wong and K. S. Ma, "An Efficient Local Approach for the Path Generation of Robot Manipulators," *Journal of Robotic Systems*, 7(1), 1990, pp. 23-55.

[14] D. N. Nenchev, "Redundancy Resolution through Local Optimization: A Review," *Journal of Robotic Systems*, 6(6), 1989, pp.769-798.

[15] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, Addison-Wesley, Reading, MA, 1989.

[16] J. K. Salisbury and J. J. Craig, "Articulated Hands: Force Control and Kinematic Issues," *Int. J. of Robotics Research*, Vol. 1, No. 1, Spring, 1982, pp. 4-16.

[17] J. Angeles, F. Ranjaran and R. V. Patel, "On the Design of the Kinematic Structure of Seven-Axes Redundant Manipulator for Maximum Conditioning," *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1992, pp. 494-499.

[18] O. Ma and J. Angeles, "The Concept of Dynamic Isotropy and Its Applications to Inverse Kinematics and Trajectory Planning," *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 481-486.

[19] O. Ma and J. Angeles, "Optimum Design of Manipulators under Dynamic Isotropy Conditions," *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1993.

CHAPTER 2 REDUNDANCY RESOLUTION BY LOCAL AND GLOBAL OPTIMIZATION

2.1 Introduction

Although the availability of redundant degrees of freedom can provide dextrous motion of the robot arm, proper exploitations of this redundancy present a challenging and complicated problem. Redundant manipulators have a characteristic of "self motion", i.e., an infinite number of joint trajectories cause the same end-effector motion. In general, redundant manipulator control schemes must generate a set of proper joint angle trajectories which cause the end-effector to track desired trajectories and at the same time satisfy additional constraints. There are many methods in solving this redundancy resolution problem. These methods can be classified into two fundamental approaches based on the mathematical formulations.

The first approach, which is based on local path information, is referred to as a local optimization approach. The performance criterion in local optimization is instantaneous. The advantage of local optimization is that its computation is relatively simple and hence it is suitable for real-time implementation. However, local optimization cannot in general provide a solution that is optimal over the whole trajectory. In addition, local optimization may cause some undesirable effects such as non-cyclic motion [1]. Some drawbacks of local optimization can be overcome by a global optimization approach.

Global optimization is based on information about the entire trajectory. The performance criterion for global optimization is an integral type. A globally optimal solution can therefore be obtained. However, a global optimization approach is generally vary expensive computationally.

A large number of papers have been published for solving the redundancy resolution problem. In this chapter, we will briefly review some of the common approaches of local and global optimization. These will be helpful for putting into proper perspective the redundancy resolution schemes developed in Chapter 3 and 4.

2.2 Redundancy Resolution by Local Optimization

The forward kinematics relationship of a manipulator is given by

$$x = f(q), \tag{2.1}$$

where x denotes the $m \times 1$ vector of positions and orientations of the end-effector in the task space with respect to a fixed reference frame, q is the $n \times 1$ joint displacement vector, and $f(q)$ represents the $m \times 1$ forward kinematics vector function. Typically, equation (2.1) is very nonlinear. Manipulator position control schemes often require the solution of the forward kinematics problem as well as its inverse - the inverse kinematics problem. While the forward kinematics can be computed very easily for most manipulators, the inverse kinematics problem is much more difficult. Analytical approaches are available for a class of nonredundant manipulators (e.g., wrist-partitioned type). But for the general case one has to resort to numerical techniques.

Differentiating equation (2.1) with respect to time, one can obtain

$$\dot{x} = J(q)\dot{q}, \tag{2.2}$$

where $J = \dfrac{\partial f}{\partial q}$ is the $m \times n$ Jacobian matrix of the end-effector. For non-redundant manipulators $m = n$, and redundant manipulators $m < n$. Equation (2.2) describes the relationship between joint velocities and the end-effector velocity. Hence a manipulator's kinematic control problem involves the solution of this equation, i.e., for a given end-effector velocity $\dot{x}$, it is required to find the solution of joint velocity $\dot{q}$.

Similarly, at the acceleration level, it is required to solve the following equation (obtained by differentiating (2.2)):

$$J(q)\ddot{q} = \ddot{x} - \dot{J}(q)\dot{q} \tag{2.3}$$

for $\ddot{q}$.

Although equations (2.1) - (2.3) describe both redundant and non-redundant manipulators, the solutions for redundant manipulators are much more difficult, and redundancy resolution is a very active area of research. The following section will briefly review some of the important techniques for redundancy resolution.

2.2.1 The Pseudoinverse Approach

The pseudoinverse approach is a method to solve systems of linear equations of the form:

$$Ab = z, \tag{2.4}$$

where A is an $m \times n$ matrix of full rank, b is an $n \times 1$ vector and z is an $m \times 1$ vector. When $n < m$, we have an overdetermined set of equations, while for $m < n$, we have an underdetermined set of equations.

For redundant manipulators, the system of equation (2.2), is underdetermined. One solution is given by

$$\dot{q} = J^{\dagger}\dot{x}, \tag{2.5}$$

where

$$J^{\dagger} = J^T (JJ^T)^{-1}. \tag{2.6}$$

Although the solution (2.5) has been used in many manipulator velocity kinematic control schemes, it has many drawbacks as pointed out by Klein and Huang [1]. The main drawback of this pseudoinverse approach is that it cannot guarantee generation of cyclic motion and does not avoid singularities. Further discussion on cyclic motion can be found in [2]. One treatment of the problem is given in [3].

Solution (2.5) has also been developed by Whitney [4] who resolves redundancy by solving the following constrained optimization problem:

$$\begin{aligned} &\text{Minimize} \qquad p = \frac{1}{2}\dot{q}^T W \dot{q} \\ &\text{Subject to} \quad \dot{x} - J\dot{q} = 0, \end{aligned} \tag{2.7}$$

where W is an $n \times n$ positive definite weighting matrix. The solution of (2.7), which is called the weighted pseudoinverse technique, is obtained as

$$\dot{q}^T = \dot{x}^T [J(q) W^{-1} J(q)^T]^{-1} J(q) W^{-1}. \tag{2.8}$$

The main feature of the weighted pseudoinverse is that one can choose W to emphasize the role of some components of $\dot{x}$ while de-emphasizing other components. When W is chosen as an identity matrix, equation (2.8) reduces to

$$\dot{q} = J^T [J(q) J(q)^T]^{-1} \dot{x}, \tag{2.9}$$

which is the same solution as that given by equations (2.5) and (2.6). As in the pseudoinverse approach, the weighted pseudoinverse approach does not avoid singularities.

Note that the solutions in (2.5) and (2.8) are special forms of the general solution of equation (2.2). In fact, the general solution of equation (2.2) can be written as [17]

$$\dot{q} = J^{\dagger} \dot{x} + (I - J^{\dagger} J) \beta, \tag{2.10}$$

where I is an $n \times n$ identity matrix, β is an $n \times 1$ arbitrary vector, and $(I - J^{\dagger} J)$ is an $n \times n$ null space projection matrix. The first term on the right-hand side of equation (2.10), which has the form of (2.5), is called the minimum norm solution, and the second term on the right-side of equation (2.10) is referred to as the homogeneous solution. The projection matrix $(I - J^{\dagger} J)$ projects the arbitrary vector β onto the null space of J. Hence $(I - J^{\dagger} J) \beta$ generates joint motion only, but does not contribute to any end-effector motion. Therefore, this motion is referred to as the self-motion of the manipulator. This general solution was used by Liegeois [5]. The arbitrary vector β may be chosen in order to generate trajectories which exploit the redundancy in a manipulator in some useful manner. This provides many applications such as singularity avoidance [6] - [9], obstacle avoidance [10][41], joint limit avoidance [11][12] and energy minimization [13][14]. Various techniques have been developed to determine the vector β, including the gradient projection approach.

2.2.2 The Gradient Projection Approach

This technique tends to move a manipulator gradually toward the optimum configuration which is associated with a gradient performance criterion while the end-effector tracks a desired trajectory.

If we choose a smooth function h, which represents a desired performance criterion as:

$$h = h(q), \tag{2.11}$$

then the arbitrary vector β in equation (2.10) can be defined as

$$\beta = \nabla h(q), \tag{2.12}$$

where $\nabla h(q) = \frac{\partial h}{\partial q}$ is the gradient of $h(q)$. Hence the homogeneous solution is obtained by projecting ∇h onto the null space of the Jacobian matrix. This scheme has been widely used in solving various redundancy resolution problems.

In the case of singularity avoidance, Yoshikawa [8] proposed to select the arbitrary vector β as

$$\beta = \nabla h(q) k, \tag{2.13}$$

where h is the manipulability measure defined by [9]

$$h(q) = \sqrt{det(JJ^T)} \tag{2.14}$$

and the scalar constant k is taken to be positive if $h(q)$ is to be maximized and negative if $h(q)$ is to be minimized. A larger value of k optimizes $h(q)$ at a faster rate. However the maximum allowable value of k is limited by bounds on the joint velocity. A computer simulation in [8] shows that the manipulator keeps as far as possible from a singular configuration by choosing the arbitrary vector as (2.13).

To avoid obstacles, Yoshikawa [8] defined the performance criterion as

$$h(q) = \frac{1}{2}(q-q_r)^T W (q-q_r), \tag{2.15}$$

where q_r is an $n \times 1$ constant arm posture vector which is necessary for avoiding an obstacle, and W is an $n \times n$ diagonal constant matrix. Hence the arbitrary vector β can be written as

$$\beta = W(q-q_r)k. \tag{2.16}$$

In [8], Yoshikawa demonstrated that the manipulator successfully avoids the obstacle.

One major problem of the gradient projection method is its complexity when applied to real-time kinematic control. As shown in the above section, if we apply the equation (2.10) directly, we have to determine $J^\dagger$ and its null space projection matrix in order to obtain the minimum norm solution and the homogeneous solution. Although the Gaussian elimination method proposed by Klein [15] reduces the computation time, the method still requires the computation of JJ^T which has numerous disadvantages. Dubey, Euler and Babcock [11] developed an efficient gradient projection method in which the computation of the generalized inverse of the Jacobian is eliminated. They also used this algorithm in an application involuting joint limit avoidance.

In the algorithm in [11], the joint velocity $\dot{q}$, which satisfies equation (2.2) can be written as

$$\dot{q} = \dot{q}_R + k\dot{q}_N \tag{2.17}$$

where $\dot{q}_R$ is any particular solution in the range space of J and satisfies $\dot{x} = J(q)\,\dot{q}$, and $\dot{q}_N$ is a homogeneous solution which satisfies $J\dot{q}_N = 0$. If $n = m+1$, we can rewrite $\dot{x} = J(q)\,\dot{q}$ as

$$\dot{x} = \begin{bmatrix} \upsilon & J_p \end{bmatrix} \dot{q}, \tag{2.18}$$

where υ is the first column of the Jacobian matrix, and J_p is an $m \times m$ matrix. If we assume that the first element of $\dot{q}_R$ is zero, then from equation (2.18), we can obtain the particular solution as

$$\dot{q}_R = \begin{bmatrix} 0 \\ J_p^{-1}\dot{x} \end{bmatrix}. \tag{2.19}$$

Similarly, assuming that the first element of $\dot{q}_N$ is equal to one, by solving the homogeneous solution

$$\begin{bmatrix} \upsilon & J_p \end{bmatrix} \dot{q}_N = 0, \tag{2.20}$$

we obtain

$$\dot{q}_N = \begin{bmatrix} 1 \\ -J_p^{-1}\upsilon \end{bmatrix}. \tag{2.21}$$

Also, we recall that the minimum norm solution of (2.2) is given by

$$\dot{q} = J^{\dagger}\dot{x}. \tag{2.22}$$

For equation (2.17), the square of the 2-norm of $\dot{q}$ can be expressed as

$$\|\dot{q}\|^2 = \|\dot{q}_R + k\dot{q}_N\|^2 = (\dot{q}_R + k\dot{q}_N)^T(\dot{q}_R + k\dot{q}_N). \tag{2.23}$$

We differentiate $\|\dot{q}\|^2$ with respect to k to obtain

$$\frac{\partial\|\dot{q}\|^2}{\partial k} = (\dot{q}_R + k\dot{q}_N)^T\dot{q}_N \tag{2.24}$$

By setting $\frac{\partial\|\dot{q}\|^2}{\partial k} = 0$, we obtain

$$k = -\frac{\dot{q}_R^T\dot{q}_N}{\dot{q}_N^T\dot{q}_N}. \tag{2.25}$$

Substituting equation (2.25) into (2.17), we get

$$\dot{q} = \dot{q}_R - \left[\frac{\dot{q}_R^T\dot{q}_N}{\dot{q}_N^T\dot{q}_N}\right]\dot{q}_N. \tag{2.26}$$

Equation (2.10) with (2.13) can be rearranged as

$$
\begin{aligned}
\dot{q} &= J^{\dagger}\dot{x} + (I - J^{\dagger}J)\,\beta \\
&= J^{\dagger}\dot{x} + k\,(I - J^{\dagger}J)\,\nabla h \\
&= J^{\dagger}\,(\dot{x} - kJ\nabla h) + k\nabla h.
\end{aligned} \tag{2.27}
$$

According to equations (2.19), (2.21) and (2.26), and replacing $\dot{x}$ by $(\dot{x} - kJ\nabla h)$, the solution of equation (2.27) is obtained as

$$
\dot{q} = \dot{q}_R - \left[\frac{\dot{q}_R^T \dot{q}_N}{\dot{q}_N^T \dot{q}_N}\right] \dot{q}_N + k\nabla h, \tag{2.28}
$$

where

$$
\dot{q}_R = \begin{bmatrix} 0 \\ J_p^{-1}\,(\dot{x} - kJ\nabla h) \end{bmatrix} \tag{2.29}
$$

and

$$
\dot{q}_N = \begin{bmatrix} 1 \\ -J_p^{-1}\upsilon \end{bmatrix}. \tag{2.30}
$$

It should be noted that the pseudoinverse of the Jacobian matrix is not required in the solution (2.28), (2.29) and (2.30).

2.2.3 The Damped Least-Squares Approach

It is well know that the solution of the pseudoinverse control for redundant manipulators does not exist at singularities [16]. In the neighborhood of a singular point, even if the pseudoinverse approach can find the solution, the operation of the manipulator may become unstable because the condition number of the Jacobian matrix becomes very large, so that a small error, such as measurement error or computer round-off, occurring in the input may produce a very large error in the solution [17] [18]. Therefore, it is important for manipulator control to plan joint trajectories that avoid singular points.

To address the singularity avoidance problem, the damped least-squares method was independently proposed by Wampler [19], and Nakamura and Hanafusa [20]. The damped least-squares method involves a trade off between a large solution for joint velocity $\dot{q}$, and a large residual end-effector velocity error $\dot{e} = J\dot{q} - \dot{x}$. Then, instead of solving the equation $\dot{x} = J(q)\dot{q}$ exactly to obtain $\dot{e} = 0$, we minimize the scalar positive quadratic function:

$$\psi = \dot{e}^T W_1 \dot{e} + \dot{q}^T W_2 \dot{q}, \tag{2.31}$$

where W_1 and W_2 are $m \times m$ and $n \times n$ positive definite weighting matrices respectively. To obtain the solution $\dot{q}$ which minimizes ψ, we differentiate ψ with respect to $\dot{q}$ to yield:

$$\frac{\partial \psi}{\partial \dot{q}} = 2J^T W_1 (J\dot{q} - \dot{x}) + 2W_2 \dot{q} = 2(J^T W_1 J + W_2)\dot{q} - 2J^T W_1 \dot{x}. \tag{2.32}$$

By setting $\frac{\partial \psi}{\partial \dot{q}} = 0$, we get the optimal solution:

$$\dot{q} = (J^T W_1 J + W_2)^{-1} J^T W_1 \dot{x}. \tag{2.33}$$

Note that $(J^T W_1 J + W_2)$ is positive definite. This implies that singularities do not occur even if the Jacobian matrix J loses rank. In fact, when $W_2 = 0$, equation (2.33) reduces to the exact solution of (2.2). Equation (2.33) gives an "approximate" solution. Near singularities, W_2 causes the damped least-squares solution given by (2.33) to be feasible instead of having a large magnitude. However, W_2 has the undesirable characteristic of causing tracking errors even when a manipulator is far from a singular configuration. In fact, at distances from singular points, a weighting matrix W_2 is not necessary. Therefore, instead of using a fixed weighting matrix W_2 as in the Wampler's formulation [19], Nakamura and Hanafusa [20] suggested that W_2 should be adjusted automatically to a large value in the neighborhood of singularities and a small value away from singularities in order to obtain feasible solutions in the neighborhood of singularities and reduce tracking errors away from singularities. Nakamura and Hanafusa propose that the weighting W_2 can be a function of the manipulability measure as follows:

$$W_2 = diag(\omega)\,, \tag{2.34}$$

where

$$\omega = \begin{cases} \omega_0 (1 - w/w_0)^2 & for \quad \omega < \omega_0 \\ 0 & for \quad \omega \geq \omega_0 \end{cases} \tag{2.35}$$

and

$$w = \sqrt{det(JJ^T)} \tag{2.36}$$

is the manipulability measure introduced by Yoshikawa [9]; ω_0 is a constant value of the damping factor at singular points, and w_0 is a threshold that represents the boundary of the neighborhood of singular points.

In order to apply the damped least-squares technique to their Reconfigurable Modular Manipulator System (RMMS), Kelmar and Khosla [21] modified the damping factor ω. Instead of defining an absolute threshold w/w_0, Kelmar and Khosla proposed the ratio of two subsequent values of the manipulability measure, w_{k+1}/w_k. Then, the improved damping factor is defined as

$$\omega = \begin{cases} \omega_0 (1 - w_{k+1}/w_k)^2 & for \quad w_{k+1}/w_k < \zeta \\ 0 & otherwise \end{cases} \tag{2.37}$$

where ζ is an appropriate constant obtained from experimental results [21]. Comparing equations (2.35) and (2.37), one finds that it is not necessary to determine the threshold of the singular boundary in (2.37). Hence, this method can be used in a situation where no information about singularities is available.

Another method for defining the damping factor was proposed by Chen and Lawrence [22] and is given by

$$\omega = \omega_0 \dot{q}^T \dot{q}. \tag{2.38}$$

The motivation for this formulation is the fact that near singular points, small changes in the end-effector velocity $\dot{x}$ require large joint velocities. This formulation does not require much computation. In [23], Wampler and Leifer extended the damped least-squares method to the acceleration level.

Although local optimization methods provide simplicity in the formulation and computation of solutions, they have several drawbacks as explained in the previous section. Some disadvantages of local optimization methods can be overcome by global optimization methods which are briefly discussed below.

2.3 Redundancy Resolution by Global Optimization

Although the heavy computation involved in global optimization precludes its use in real-time control, it may be a preferred solution in some off-line schemes. To date, global optimization techniques have been used in several applications for redundant manipulators such as minimum-time trajectory planning [32] - [37] and minimum-energy control [38] - [41]. To use global optimization for redundant manipulators, in general, there are two main mathematical methods: calculus of variations and optimal control.

2.3.1 Calculus of Variations

A fundamental problem in the calculus of variations is that of minimizing a performance index Γ in an integral form [24]

$$\Gamma = \int_{\rho_0}^{\rho_f} p\,(q, \dot{q}, \rho)\, d\rho. \tag{2.39}$$

If we set

$$\dot{q} = u = f(q, u, \rho)\,, \tag{2.40}$$

the integral (2.39) can be expressed as

$$\Gamma = \int_{\rho_0}^{\rho_f} p\,(q, u, \rho)\, d\rho. \tag{2.41}$$

To minimize Γ, we define the Lagrangian l as

$$l = p(q, u, \rho) + \lambda^T u, \tag{2.42}$$

where λ is denotes the Lagrange multiplier. The necessary conditions for a minimum are given by the Euler-Lagrange equations

$$\frac{\partial l}{\partial q} - \frac{d}{d\rho}\frac{\partial l}{\partial \dot{q}} = 0 \tag{2.43}$$

and

$$\frac{\partial l}{\partial \lambda} = 0. \tag{2.44}$$

Clearly, if we replace the independent variable ρ by time t, consider q the state variable and u the control variable, the calculus of variations can be directly applied to redundancy resolution of manipulators, i.e., for a redundant manipulator, we consider the problem of minimizing the performance index

$$\Gamma = \int_{t_0}^{t_f} p(q, \dot{q}, t)\, dt, \tag{2.45}$$

subject to the $m \times 1$ kinematic constraints

$$x - f(q) = 0. \tag{2.46}$$

The Lagrangian equation becomes

$$l = p(q, \dot{q}, t) + \lambda^T (x - f(q)) \,, \tag{2.47}$$

The necessary conditions for a minimum are given by the equations

$$\frac{\partial l}{\partial q} - \frac{d}{dt}\frac{\partial l}{\partial \dot{q}} = 0 \tag{2.48}$$

and

$$\frac{\partial l}{\partial \lambda} = x - f(q) = 0. \tag{2.49}$$

Using the calculus of variations to solve the global optimization problems for redundant manipulators, one may choose the performance index as

$$\Gamma = \int_{t_0}^{t_f} \frac{1}{2}\dot{q}^T W \dot{q}\, dt, \tag{2.50}$$

where W is an $n \times n$ constant positive definite weighting matrix. Then, the Lagrangian equation can be written as

$$l = \frac{1}{2}\dot{q}^T W \dot{q} + \lambda^T (x - f(q)) \,. \tag{2.51}$$

Substituting (2.51) into (2.48), we get

$$W\ddot{q} - J^T \lambda^T = 0. \tag{2.52}$$

Equation (2.52) gives

$$\ddot{q} = W^{-1} J^T \lambda^T. \tag{2.53}$$

Differentiating (2.49) twice with respect to time yields

$$\ddot{x} - J\ddot{q} - \dot{J}\dot{q} = 0, \tag{2.54}$$

and substituting (2.53) into (2.54), we obtain

$$\lambda^T = (JW^{-1}J^T)^{-1} (\ddot{x} - \dot{J}\dot{q}). \tag{2.55}$$

Eliminating λ^T in (2.53) using (2.55), we have

$$\ddot{q} = W^{-1} J^T (JW^{-1}J^T)^{-1} (\ddot{x} - \dot{J}\dot{q}). \tag{2.56}$$

Equation (2.56) has been obtained independently by Kazerounian and Wang [25], and Martin et al [26]. Note that (2.56), which satisfies the necessary conditions (2.48) and (2.49), does not consider boundary conditions. For specifying uniquely the solution $\ddot{q}$, boundary conditions have to be taken into account. In general, we may consider the initial and final values of x which satisfy (2.1) and (2.2) at the initial time t_0 and final time t_f, i.e.,

$$x(t) = f(q(t)) \qquad \text{at } t = t_0 \text{ and } t = t_f, \tag{2.57}$$

$$\dot{x}(t) = J(q(t))\,\dot{q}(t) \qquad \text{at } t = t_0 \text{ and } t = t_f. \tag{2.58}$$

Equation (2.56) with the boundary condition (2.57) and (2.58) defines a second-order differential equation system. Solving these equations we obtain a minimum for the objective function (2.50) for a given trajectory.

Another application using the calculus of variations is proposed by Suh and Hollerbach [27]. They solved the global torque optimization problem and compared the global optimization approach with the local optimization approach. In addition, instead of deriving directly the necessary conditions for solving the optimization problem, Won et al. [31] obtained an approximate optimal solution by using Fourier series which are determined by exploiting Powell's method.

2.3.2 Optimal Control

The most general optimal control problem consists of four parts: the plant or process, the performance index, the final state constraint, and the class of admissible controllers [28] [29].

The plant or process relates the state of response $y(t)$ to the input or control $u(t)$ by the time-varying differential equation

$$\dot{y}(t) = F(y(t), u(t), t). \tag{2.59}$$

The process is linear or nonlinear, and of nth order, i.e., described by a set of a first order differential equations.

With the process defined by (2.59), the general performance index P can be written as

$$P = \varphi(y(t_f), t_f) + \int_{t_0}^{t_f} L(y(t), u(t), t)\, dt, \tag{2.60}$$

where $y(t)$ denotes the state vector at the instant t. The performance index P is taken to depend partly on the final state $y(t_f)$ and partly on the behavior of the system in reaching this final state. If it is desired to give no special weight to the final state, then φ can be set to zero.

If the final state is either completely specified, or is subject to a constraint, then one can define a final-state constraint vector as

$$\phi(y(t_f), t_f) = 0. \tag{2.61}$$

Note that the roles of the final weighting function φ and the fixed final function ϕ are different - φ is a function of the final state which we want to make small, while ϕ is a function of the final state which is required to be exactly zero.

By defining the Hamiltonian equation [28][29]

$$H(y, u, t) = L(y, u, t) + \lambda^T F(y, u, t), \tag{2.62}$$

we require that the optimal controller satisfy the necessary conditions which are given by the state equation

$$\dot{y} = \frac{\partial H}{\partial \lambda} = F(y, u, t) \tag{2.63}$$

and the costate equation

$$-\dot{\lambda} = \frac{\partial H}{\partial y} = \left[\frac{\partial F}{\partial y}\right]^T \lambda + \frac{\partial L}{\partial y}. \tag{2.64}$$

The optimal control input u^* can be obtained by the stationary condition:

$$\frac{\partial H}{\partial u} = \frac{\partial L}{\partial u} + \left[\frac{\partial F}{\partial u}\right]^T \lambda = 0. \tag{2.65}$$

Then the formulation of the optimal controller becomes the two-point boundary-value problem (TPBVP) which satisfies the following boundary conditions [28][29]:

$$y(t_0) = y_0, \tag{2.66}$$

$$,\left[\frac{\partial \varphi}{\partial y} + \left[\frac{\partial \phi}{\partial y}\right]^T \gamma - \lambda\right]^T\Bigg|_{t = t_f} dy(t_f) + \left[\frac{\partial \varphi}{\partial t} + \left[\frac{\partial \phi^T}{\partial t}\right]^T \gamma + H\right]\Bigg|_{t = t_f} dt_f = 0 \tag{2.67}$$

where y_0 is the given initial value and γ is a vector of constant multiplier associated with the final-state constraint.

In the formulation of the optimal control discussed above, equation (2.63) is just the constraint equation, or the system equation. The constraint equation and the costate equation are coupled differential equations and, together, they define a TPBVP. In general, we are not concerned what λ is, but we note that solving for λ is an intermediate step in finding the optimal input u^*.

The optimal control problem discussed above is based on the assumption that the control variable $u(t)$ is unconstrained. The condition for the optimal input u^* satisfies equation (2.65). If the control variable $u(t)$ is constrained to lie in an admissible region R_c, for example,

$$u(t) \leq N \qquad N \in R_c, \tag{2.68}$$

Pontryagin et al. [30] show that the stationary condition (2.65) must be replaced by

$$H^* (y(t), u(t), \lambda(t)) = min\ H(y(t), u(t), \lambda(t)) \qquad for\ all \quad u \in R_c, \qquad (2.69)$$

where * denotes optimal quantities. Equation (2.69) can be written as

$$H^* (y(t), u(t), \lambda(t)) \le H(y(t), u(t), \lambda(t)) \qquad for\ all \quad u \in R_c. \qquad (2.70)$$

The optimal control with condition (2.70) is called Pontryagin's minimum principle. We can see that Pontryagin's minimum principle becomes useful for cases of constrained control variable problems. For example, the control condition is easy to satisfy. Although the control condition $\frac{\partial H}{\partial u} = 0$ looks straightforward, sometimes it is not easy to find $\frac{\partial H}{\partial u} = 0$, even in the case of the unconstrained problem. Also, Pontryagin's minimum principle does not require the differentiation of H with respect to u.

Although the optimal control problem requires a large amount of computation, several researchers have tried to employ optimal control techniques in redundancy resolution of manipulators. Some schemes of redundancy resolution using optimal control strategies consider manipulator dynamics. The dynamic equation of a rigid-link manipulator is given by

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau, \qquad (2.71)$$

where

$M(q)$: $n \times n$ mass matrix of manipulators,

$C(q, \dot{q})\,\dot{q}$: $n \times 1$ vector of centrifugal and Coriolis forces,

$g(q)$: $n \times 1$ vector of gravity effects,

q: $n \times 1$ vector of joint displacements,

τ: $n \times 1$ vector of joint torques.

If we set $y = \begin{bmatrix} y_1 & y_2 \end{bmatrix}^T = \begin{bmatrix} q & \dot{q} \end{bmatrix}^T$, equation (2.71) can be written as

$$\dot{y} = F(y, \tau), \tag{2.72}$$

where

$$F = \begin{bmatrix} y_2 \\ -M^{-1}(y_1)\,[C(y_1, y_2)\,y_2 + g(y_1)] + (M^{-1}(y_1))\,\tau \end{bmatrix}. \tag{2.73}$$

In general, there are two major applications of optimal control techniques in redundancy resolution: minimum-time control and minimum-energy control.

The minimum-time control problem may be considered as a special case of the more general optimal control problem. The minimum-time control problem deals with minimizing the performance index

$$P = \int_{t_0}^{t_f} 1\, dt, \tag{2.74}$$

i.e., $L = 1$ and $\varphi = 0$, with the constraints

$$c(y(t), \tau(t)) \le 0. \tag{2.75}$$

The minimum-time control problem involves the determination of a trajectory which minimizes the travel time from a given initial configuration to a specified final configuration. This implies that a manipulator moves its initial configuration to its final configuration as fast as possible under the constraint (2.75). With the Hamiltonian equation

$$H = 1 + \lambda^T F, \tag{2.76}$$

the minimum-time control problem is obtained by solving the following TPBVP:

$$\dot{y} = F(y, \tau), \tag{2.77}$$

$$-\dot{\lambda} = \left[\frac{\partial F}{\partial y}\right]^T \lambda, \tag{2.78}$$

$$\frac{\partial H}{\partial u} = \left[\frac{\partial F}{\partial u}\right]^T \lambda = 0, \tag{2.79}$$

$$\lambda^T F\big|_{t = t_f} = -1, \tag{2.80}$$

with n initial and final conditions

$$y(t_0) = y_0, \tag{2.81}$$

$$y(t_f) = y_f, \tag{2.82}$$

where y_0 and y_f are given initial and final configurations. In addition, the dynamic equation (2.72) can be written as

$$\dot{y} = A(y) + B(y)\tau \quad , \tag{2.83}$$

where

$$A(y) = \begin{bmatrix} y_2 \\ -M^{-1}(y_1)\,[C(y_1, y_2)\,y_2 + g(y_1)] \end{bmatrix}, \tag{2.84}$$

and

$$B(y) = \begin{bmatrix} 0 \\ M^{-1}(y_1) \end{bmatrix}. \tag{2.85}$$

To solve the constrained optimal control problem, instead of using (2.79), we attempt to obtain

$$H^*(y(t), \tau(t), \lambda(t)) = min\ H(y(t), \tau(t), \lambda(t)) \tag{2.86}$$

without violating the constraint (2.75). We assume that the control constraint is

$$-\tau_0 \le \tau \le \tau_0, \tag{2.87}$$

where τ_0 denotes the bounds on the joint torques. Using (2.83) in the Hamiltonian, we have

$$H = 1 + \lambda^T(A(y) + B(y)\tau) = 1 + \lambda^T A(y) + \lambda^T B(y)\tau\,. \tag{2.88}$$

Clearly, by minimizing H with respect to τ, taking into account the constraint (2.87), we obtain

$$\tau(t) = \begin{cases} \tau_0 & for \quad \lambda^T B(y) < 0 \\ -\tau_0 & for \quad \lambda^T B(y) > 0 \end{cases} \tag{2.89}$$

Thus, this control scheme is of "bang-bang" type whose value depends on the sign of the switching function $\lambda^T B(y)$. This minimum-time solution was obtained by Niv and Auslander [32]. Other attempts to solve this problem by simplifying the manipulator model resulted in non-optimal solutions [33]. More applications of minimum-time control can also be found in [34] - [37].

For the minimum-energy control problem, the performance index P, in general, can be written as

$$P = \varphi(q(t_f), \dot{q}(t_f), t_f) + \int_{t_0}^{t_f} (\frac{1}{2}\dot{q}^T W_1 \dot{q} + \frac{1}{2}\tau^T W_2 \tau)\, dt, \tag{2.90}$$

where W_1 and W_2 are positive definite constant weighting matrices. Minimizing the performance index P implies that the joint velocity $\dot{q}$ and the joint torque τ are to be kept bounded in the entire workspace according to the relative magnitude of the elements of W_1 and W_2. For example, if it is more important to keep joint velocities small, the diagonal elements of W_1 should be selected to have larger magnitudes than these of W_2. If small joint torques are more important in the control process, then the converse would be the case. The Hamiltonian equation can be written as

$$H = \frac{1}{2}\dot{q}^T W_1 \dot{q} + \frac{1}{2}\tau^T W_2 \tau + \lambda^T F = \frac{1}{2}y_2^T W_1 y_2 + \frac{1}{2}\tau^T W_2 \tau + \lambda^T F \tag{2.91}$$

Then the minimum-energy control problem satisfies the following TPBVP:

$$\dot{y} = F(y, \tau), \tag{2.92}$$

$$-\dot{\lambda} = \left[\frac{\partial F}{\partial y}\right]^T \lambda + W_1 y_2, \tag{2.93}$$

$$\frac{\partial H}{\partial u} = W_2 \tau + \left[\frac{\partial F}{\partial u}\right]^T \lambda = 0, \tag{2.94}$$

with the boundary condition

$$y(t_0) = y_0, \tag{2.95}$$

$$\left(\frac{\partial \varphi}{\partial y} - \lambda\right)\bigg|_{t = t_f} = 0. \tag{2.96}$$

One application of the minimum-energy optimal control problem for manipulators was given by Nakamura and Hanafusa [38]. They solved the problem for both kinematics and dynamics. For kinematics, they minimized the performance index

$$P = \int_{t_0}^{t_f} \dot{q}^T \dot{q}\, dt \tag{2.97}$$

subject to $\dot{x} - J\dot{q} = 0$. For dynamics, they minimized the performance index

$$P = \int_{t_0}^{t_f} \tau^T \tau dt \qquad (2.98)$$

subject to $\dot{y} - F(y, \tau) = 0$. In addition, Yen and Nayurka [39] applied a quasi-linearization method to solve the optimal control problem for torque optimization for a manipulator, and Hu and Goldenberg [40] developed an algorithm for torque optimization for multiple redundant manipulators. Khadem and Dubey [41] dealt with the energy immunization problem incorporating obstacle avoidance.

From the brief overview above, we can see that redundancy resolution, both by local and global optimization, did not consider system sensitivity. Some researchers have taken the conditioning measures into account in order to reduce system sensitivity, but only in the context of manipulator design [42] [43] [44] [45]. Therefore, this thesis will develop two approaches for redundancy resolution based on optimization of kinematic and dynamic measures to see how conditioning measures affect system sensitivity from a control perspective. Constraint optimization technique and optimal control theory will be used.

References

[1] C. A. Klein and C. H. Huang, "Review of Pseudoinverse Control for Use with Kinematically Redundant Manipulators," *IEEE Trans. systems, Man, and Cybernetics*, Vol. 13, No. 3, 1983, pp. 245-250.

[2] C. A. Klein and K. B. Kee, "The Nature of Drift in Pseudoinverse Control of Kinematically Redundant Manipulators," *IEEE Trans. on Robotics and Automation*, Vol. 5, No.2, 1989, pp. 231-234.

[3] T. Shamir and Y. Yomdin, "Repeatability of Redundant Manipulators: Mathematical Solution of the Problem," *IEEE Trans. on Robotics and Automation*, Vol. 33, No. 11,

1988, pp. 1004-1009.

[4] D. E. Whitney, "Resolved Motion Rate Control of Manipulator and Human Prostheses," *IEEE Trans. on Man-Machine Systems*, Vol 10, No. 2, 1969, pp. 47-53.

[5] A. Liegeois, "Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms," *IEEE Trans. systems, Man, and Cybernetics*, Vol. 7, No. 12. 1977, pp. 868-871.

[6] K. Kazeruonian and Z. Wang, "On the Singularity Avoidance and Uniqueness of Solutions in Redundant Industrial Robots," *Proc. ASME Int. Conf. on Computers in Engineering*, 1988, pp. 293-296.

[7] S. Chiaverini B. Siciliano and O. Egeland, "Kinematic Analysis and Singularity Avoidance for a Seven-Joint Manipulator," *Proc. of American Control Conf.*, Vol. 3, 1990, pp. 2300-2305.

[8] T. Yoshikawa, "Analysis and Control of Robot Manipulators with Redundancy," *First Int. Symposium of Robotics Research*, MIT Press, Cambridge, 1984. pp. 735-747.

[9] T. Yoshikawa, "Manipulability of Robotic Mechanisms," *Int. J. of Robotics Research*, Vol. 4, No. 2, Summer, 1985. pp. 3-9.

[10] A. K. Chassiakos and M. A. Christodoulu, "Kinematic Control and Obstacle Avoidance for Redundant Manipulators," *Proc. of the 25th Conf. on Decision and Control*, 1986, pp. 96-97.

[11] R. V. Dubey, J. A. Euler and S. M. Babcock, "An Efficient Gradient Projection Scheme for a Seven-Degree-of-Freedom Redundant Robot with Spherical Wrist," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1988, pp. 28-36.

[12] T. F. Chan and R. V. Dubey, "A Weighted Least-Norm Solution Based Scheme for Avoiding Joint Limits for Redundant Manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, Vol. 3, 1993, pp.395-402.

[13] R. V. Mayorga, A. K. C. Wong and K. S. Ma, "An Efficient Local Approach for the

Path Generation of Robot Manipulators," *Journal of Robotic Systems*, Vol. 7, No. 1, Feb. 1990, pp. 23-55.

[14] A. Nedungadi and K. Kazerounian, "A Local Solution with Global Characteristics for the Joint Torque Optimization of a Redundant manipulator," *Journal of Robotic Systems*, Vol. 6, No. 5, Oct. 1989, pp. 631-654.

[15] C. A. Klein, "Use of Redundancy in the Design of Robotic Systems," *Robotics Research: Second Int. Symposium*, MIT Press, Cambridge, 1985, pp. 207-214.

[16] G. H. Golub and Van Loan, *Matrix Computations*, 2nd. ed., Johns Hopkins University Press. Baltimore, 1989.

[17] A. Ben-Israel and T. N. E. Greville, *Generalized Inverses: Theory and Applications*, John Wiley and Sons, New York, 1974.

[18] J. K. Salisbrury and J. J. Craig, "Articulated Hands: Force Control and Kinematic Issues," *Int. Journal of Robotics Research*, Vol. 1, No. 1, Spring 1982, pp. 4-16.

[19] C. W. Wampler, II, "Manipulator Inverse Kinematics Solution Based on Vector Formulations and Damped Least-Squares Methods," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 16, No. 1, 1986, pp. 93-101.

[20] Y. Nakamura and H. Hanafusa, "Inverse Kinematic Solutions With Singularity Robustness for Robot Manipulator Control," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 108, 1986, pp. 163-171.

[21] L. Kelmar and P. K. Khosla, "Automatic Generation of Kinematics for a Reconfigurable Modular Manipulator System," *Proc. IEEE Int. Conf. Robotics and Automation*, 1988, pp. 663-668.

[22] S. K. Chan and P. D. Lawrence, "General Inverse Kinematics with the Error Damped Pseudoinverse," *Proc. IEEE Int. Conf. Robotics and Automation*, 1988, pp. 834-839.

[23] C. W. Wampler, II and L. J. Leifer, "Applications of Damped Least-Squares Methods to Resolved-Rate and Resolved-Acceleration Control of Manipulators," *Journal*

of Dynamic Systems, Measurement, and Control, Vol. 110, 1988, pp. 31-38.

[24] L. A. Pars, *An Introduction to the Calculus of Variations*, Heinemann, London, 1965

[25] K. Kazerounian and Z. Wang, "Global versus Local Optimization in Redundancy Resolution of Robotic Manipulators," *Int. Journal of Robotics Research*, Vol. 7, No. 5, 1988, pp. 3-12.

[26] D. P. Martin, J. Bailleul and J. M. Hollerbach, "Resolution of Kinematic Redundancy Using Optimization Techniques," *IEEE Trans. on Robotics and Automation*, Vol. 5, No.4, August 1989, pp. 529-533.

[27] K. C. Suh and J. M. Hollerbach, "Local versus Global Torque Optimization of Redundant Manipulators," *Proc. IEEE Int. Conf. Robotics and Automation*, 1987, pp. 619-624.

[28] A. E. Bryson, Jr. and Y. C. Ho, *Applied Optimal Control*, Hemishere, New York, 1975.

[29] F. L. Lewis, *Optimal Control*, John Wiley & Sons, New York, 1986.

[30] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze and E. F. Mishchenko, The *Mathematical Theory of Optimal Processes*, Wiley-Interscience, New York, 1962.

[31] J. H. Won, B. W. Cho and M. J. Chung, "A Numerical Method of Perdetermined Optimal Resolution for a Redundant Manipulator," *IEEE Trans. on Robotics and Automation*, Vol. 9, No. 2, April 1993, pp. 215-220.

[32] M. Niv and D. M. Auslander, "Optimal Control of a Robot with Obstacles," *Proc. of American Control Conf.*, June 1984, pp. 280-287.

[33] J. T. Wen and A. A. Desrochers, "Sub-Optimal Control Strategies for Robotic Manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, April 1986, pp. 402-406.

[34] J. E. Bobrow, S. Dubowsky and J. S. Gibson, "Time-Optimal Control of Robotic Manipulators Along Specified Paths," *Int. Journal of Robotics Research*, Vol. 4, No.3, 1985, pp. 3-17

[35] B. K. Kim and K. G. Shin, "Minimum-Time Path Planning for Robot Arms and Their Dynamics," *IEEE Trans on Automatic Control*, AC-30, 1985, pp.213-223.

[36] B. Donald and P. Xavier, "A Provably Good Approximation Algorithm for Optimal-Time Trajectory Planning," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1989, pp. 958-963.

[37] Z. Shiler and H. H. Lu, "Robust Computation of Path Constrained Time Optimal Motions," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 144-149.

[38] Y. Nakamura and H. Hanafusa, "Optimal Redundancy Control of Robot Manipulators," *Int. Journal of Robotics Research*, Vol. 6, No. 1, Spring 1987, pp. 32-42.

[39] V. Yen and M. Nagurka, "Optimal Trajectory Planning of Robotic Manipulators via Quasi-Linearization and State Parameterization," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1989, pp.1116-1121.

[40] Y.-R Hu and A. A. Goldenberg, "Dynamic Control of Multiple Coordinated Redundant Manipulators with Torque Optimization," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1990, pp.1000-1005.

[41] S. E. Khadem and R. V. Dubey, "A Global Cartesian Space Obstacle Avoidance Scheme for Redundant Manipulators," *Optimal Control Applications & Methods*, Vol. 12, 1991, pp. 279-286.

[42] J. K. Salisbury and J. J. Craig, "Articulated Hands: Force Control and Kinematic Issues," *Int. J. of Robotics Research*, Vol. 1, No. 1, pp. 4-16, Spring, 1982, pp. 4-16.

[43] J.-O. Kim and P. K. Khosla, "Dexterity Measures for Design and Control of Manipulators," *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, IROS'91, Nov. 3-5, 1991, pp. 758-763.

[44] C. A. Klein and B. E. Blaho, "Dexterity Measures for the Design and control of Kinematically Redundant Manipulators," *Int. J. of Robotics Research*, Vol. 6, No. 2, Summer, 1987, pp. 72-83.

[45] O. Ma and J. Angeles, "The Concept of Dynamic Isotropy and its Applications to

Inverse Kinematics and Trajectory Planning," *IEEE Int. Conf. Robotics and automation*, 1990, pp.481-486.

CHAPTER 3 REDUNDANCY RESOLUTION BASED ON OPTIMIZATION OF KINEMATIC MEASURES

3.1 Introduction

A variety of techniques for resolving redundancy for kinematically redundant manipulators have been proposed. The most effective aspects of these techniques include generalized inverse techniques [1] - [5] and the extended Jacobian method [6] - [10]. In addition, some algorithms seek to obtain solutions which optimize certain secondary performance criteria. These include singularity avoidance [11][12], obstacle avoidance [13][14][16], energy minimization [3], and torque minimization [15]. Redundancy resolution based on optimizing kinematic measures (RROKM) that is developed in this chapter ensures that a redundant manipulator always works along a well-conditioned trajectory, i.e., the Jacobian matrix is as well-conditioned as possible while the end-effector of the redundant manipulator tracks a desired trajectory in Cartesian space. A well-conditioned Jacobian matrix ensures that measurement and numerical rounding errors do not increase significantly as a result of transformations using the Jacobian matrix.

3.2 Development of the RROKM Scheme

The proposed scheme is optimal in the sense that the condition number of the Jacobian matrix is as well-conditioned as possible while the end-effector tracks a desired Cartesian space trajectory. To solve this problem, a constrained optimization method is used.

There are several such measures for redundant and nonredundant manipulators, e.g., see [19][20][23][24]. These measures are based on the Jacobian matrix and can be expressed in the terms of its singular value decomposition [18]. Although all these kinematic dexterity measures can be used in an objective function in the optimization, we use only the condition number [23] and the isotropy measure defined by Kim and Khosla [25] in our RROKM scheme. The isotropy measure of Kim and Khosla (we shall call it the KK isotropy measure) yields a more efficient computational scheme than the condition number.

3.2.1 Kinematic Dexterity Measures for Manipulators

An important characteristic for manipulators is a high degree of dexterity. A number of different kinematic dexterity measures have resulted in many different criteria for quantification. These measures include the manipulability measure [19], the determinant [20], the condition number of the Jacobian matrix [21][23], the KK isotropy measure [25], smallest singular value [26] and kinematic conditioning index (KCI) [27]. When the condition number is its optimal value of one, the resulting manipulator has been defined as isotropic [23][27]. Isotropic manipulator has one important characteristic, mainly, the transformation accuracy. In addition, when the condition number is in optimal value, sensitivity in both velocity and torque is a minimum. Kinematic dexterity measures for robot manipulators are mainly based on the Jacobian matrix.

Recalling that, for kinematically redundant manipulators, the forward kinematic model is defined by

$$x = f(q), \tag{3.1}$$

where x denotes the $m \times 1$ vector of positions and orientations of the end-effector in the work space with respect to a fixed reference frame, q is the $n \times 1$ joint displacement vector, and $f(q)$ represents the $m \times 1$ forward kinematic vector function. The differential kinematic model is written as

$$\dot{x} = J(q)\,\dot{q}, \tag{3.2}$$

where $J = \dfrac{\partial f}{\partial q}$ is the $m \times n$ Jacobian matrix of the end-effector. It will be assumed in the rest of this thesis that the matrix J has been made dimensionally homogeneous by normalization using a "characteristic length" where appropriate [29]. For redundant manipulators $n > m$ so that the general solution to (3.2) is typically presented as [1][12][17]

$$\dot{q} = J^{\dagger}\dot{x} + (I + J^{\dagger}J)\beta, \tag{3.3}$$

where $J^{\dagger}$ denotes the pseudoinverse $J^{\dagger} = J^T (JJ^T)^{-1}$ if rank(J) $= m$, $(I - J^{\dagger}J)$ is a projection operator onto the null space of J and β is an arbitrary vector in joint space. The second term in (3.3) is called the homogeneous solution to (3.2) because it results in no end-effector motion. Often, the homogeneous solution is used to optimize some secondary criteria under the constraints of the specified end-effector velocity.

For $m \neq n$, let J have the singular value decomposition [18]:

$$J = USV^T, \tag{3.4}$$

where the superscript T denotes the matrix transpose, and U and V are $m \times m$ and $n \times n$ orthogonal matrices, respectively. If J has rank m and $m > 0$, then S is an $m \times n$ matrix of the form:

$$S = \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix}, \tag{3.5}$$

where Σ is an $m \times m$ diagonal matrix denoted by $diag(\sigma_i)$ with σ_i, the non-zero singular values of J, ordered as

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_m > 0. \tag{3.6}$$

Several kinematic dexterity measures can be expressed based on the singular values of the Jacobian matrix, for example, the manipulability measure [19], the condition number [21][23], smallest singular value [26] and the KK isotropy measure [25].

A. The Manipulability Measure

The manipulability measure was defined by Yoshikawa [19]. Based on the singular value decomposition, this measure can be expressed as

$$M_1 = \prod_{i=1}^{m} \sigma_i = \sqrt{det(JJ^T)}. \tag{3.7}$$

In order to graphically describe manipulability in the workspace of a manipulator, Yoshikawa defined the concept of a manipulability ellipsoid. If the n joint velocities are normalized so that

$$\|\dot{q}\|^2 = (\dot{q}_1^2 + \dot{q}_2^2 + \ldots + \dot{q}_n^2) \leq 1, \tag{3.8}$$

then the manipulability ellipsoid has principal axes $\sigma_1 u_1, \sigma_2 u_2, \ldots, \sigma_m u_m$, where u_i is the ith column vector of U. The volume of the ellipsoid is proportional to the manipulability measure, so that the farther the manipulator is from a singularity position, the higher the manipulability measure, i.e., the larger the volume of the manipulability ellipsoid.

When $m = n$, i.e., for non-redundant manipulators, the measure of manipulability reduces to the determinant of the Jacobian matrix J. This has been used to evaluate wrist configurations by Paul and Stevenson [20].

B. The Condition Number

The condition number is expressed as the ratio of the largest singular value to the smallest (non-zero) singular value of J [21]:

$$M_2 = \frac{\sigma_1}{\sigma_m}. \tag{3.9}$$

When all singular values have the same magnitudes, i.e., $\sigma_i = \sigma$, the condition number has a minimum value of one and the sensitivity in both velocity and torque is the smallest. Salisbury and Craig [23] used M_2 in an optimization criterion to obtain ideal dimensions for their articulated hand manipulator. In [23] the condition number of the Jacobian matrix is considered a measure of the kinematic accuracy of manipulators. However, Chiu [24] questioned the use of the Jacobian condition number as a measure of accuracy, but this is based on the fallacy that if f_e denotes the end-effector force vector and δf_e denotes the error in f_e, then f_e and δf_e are linearly dependent vectors. Hence the relative force error is independent of the manipulator kinematics. We will show that the input and output relative errors in the linear system with the form (3.2) is the same only when the input error and input vectors are linearly dependent and also illustrate by simulation for a planar manipulator in section 3.3.

C. KK Isotropy Measure

Kim and Khosla [25] proposed a measure of isotropy as a ratio of the geometric mean and the arithmetic mean of the eigenvalues of the Jacobian matrix J. This definition can be expressed in the form of the singular value decomposition of J as:

$$M_3 = \frac{\left(\prod_{i=1}^{m} \sigma_i^2\right)^{1/m}}{\frac{1}{m}\sum_{i=1}^{m} \sigma_i^2}. \tag{3.10}$$

It should be noted that $0 < M_3 \leq 1$. When all the singular values of J have the same magnitudes, i.e., $\sigma_i = \sigma$, this measure has a maximum value of one. The computation of M_3 is more efficient than that of M_2.

D. Smallest Singular Value

Klein and Blaho [26] suggested that the smallest singular value can be used as a measure of dexterity:

$$M_4 = \sigma_m. \tag{3.11}$$

This measure indicates the distance of J from singularity.

3.2.2 Formulation of the Problem of RROKM

In kinematic control, we wish to solve for the joint velocity vector $\dot{q}$ in (3.2) for a given end-effector velocity vector $\dot{x}$. Therefore, this involves the generalized inverse of the matrix J. In general, if the Jacobian matrix is not singular, the numerical inversion of J and the solution of (3.2) are always possible. However, if any errors appear in measuring or rounding-off $\dot{x}$ in (3.2), they may be magnified in computing $\dot{q}$ depending on the condition number of J. To see this, let us suppose that $\dot{x}$ of (3.2) is perturbed by $\delta\dot{x}$. The relation between the relative error of the perturbation $\frac{\|\delta\dot{x}\|}{\|\dot{x}\|}$ and the relative error of the solution $\frac{\|\delta\dot{q}\|}{\|\dot{q}\|}$ can be expressed as [21][22]

$$\frac{\|\delta\dot{q}\|}{\|\dot{q}\|} \leq cond\,(J)\,\frac{\|\delta\dot{x}\|}{\|\dot{x}\|}, \tag{3.12}$$

where $cond\,(J)$ denotes the condition number of the Jacobian matrix J (see Appendix). Thus, the relative error in $\dot{q}$ can be an amplification by a factor $cond(J)$ of the relative error in $\dot{x}$. In this sense cond(J) quantifies the sensitivity of the problem of solving $J\dot{q} = \dot{x}$. It is obvious that a small change in $\dot{x}$ may cause a large change in $\dot{q}$ when J is ill-conditioned (i.e., $cond\,(J) \gg 1$).

Therefore, the problem of RROKM requires that the Jacobian matrix J be as well-conditioned as possible while the end-effector tracks a desired trajectory. It is, thus, necessary to solve the following optimization problem:

$$\begin{aligned} &\text{Optimize} && M_i\,(q\,(t))\,, \\ &\text{Subject to} && x_d\,(t) - f(q\,(t)) = 0, \end{aligned} \tag{3.13}$$

where $x_d(t)$ is the desired end-effector trajectory and $M_i\,(q\,(t))$, $i = 1, 2, 3$ or 4, is one of the measures of conditioning defined in Section 3.2.1. This is an equality constrained optimization problem. The MATLAB function *constr* [28] can be used to solve the problem. The function uses a Sequential Quadratic Programming (SQP) method [28]. At each iteration, a Quadratic Programming (QP) problem is solved. Also, at each iteration an estimate of the Hessian of the Lagrangian is updated using the Broyden-Fletcher-Goldfarb-Shannon (BFGS) formula.

This is a local optimization problem, and the initial configuration of the redundant manipulator is assumed to be at a local optimum. The optimization problem is solved at each sampling period while the end-effector follows the desired trajectory. Solution of the problem ensures that corresponding to each sampled point on the desired end-effector trajectory, the condition number of the Jacobian matrix is as well-conditioned as possible.

3.3 Simulation Examples

In this section, several simulation results for a three-link revolute-joint planar manipulator are presented to illustrate the RROKM scheme.

Consider the three-link revolute-joint planar manipulator (m=2, n=3) shown in Fig 3.1. If we do not consider the end-effector orientation, this manipulator has one degree of redundancy. The forward kinematics and the Jacobian matrix of the manipulator are given by

$$x = f(q) = \begin{bmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} \end{bmatrix} \tag{3.14}$$

and

$$J(q) = \begin{bmatrix} -(l_1 s_1 + l_2 s_{12} + l_3 s_{123}) & -(l_2 s_{12} + l_3 s_{123}) & -l_3 s_{123} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} \end{bmatrix} \tag{3.15}$$

respectively, where $c_1 = \cos(q_1)$, $c_{12} = \cos(q_1 + q_2)$, $s_1 = \sin(q_1)$, $s_{12} = \sin(q_1 + q_2)$, and so on. Assume that the link lengths are $l_1 = l_2 = l_3 = 1$ m.

The end-effector is required to track the desired trajectory which is chosen as

$$x_d(t) = \begin{bmatrix} x_d^x \\ x_d^y \end{bmatrix} = \begin{bmatrix} p_d^x(t_0) + (p_d^x(t_f) - p_d^x(t_0))\left(\frac{t}{t_f} - \frac{1}{2\pi}\sin\left(2\pi\frac{t}{t_f}\right)\right) \\ p_d^y(t_0) + (p_d^y(t_f) - p_d^y(t_0))\left(\frac{t}{t_f} - \frac{1}{2\pi}\sin\left(2\pi\frac{t}{t_f}\right)\right) \end{bmatrix}, \tag{3.16}$$

where t_o and t_f represent the initial and final values of time, $x_d(t_0) = \left[p_d^x(t_o)\ p_y^y(t_o)\right]^T$ and $x_d(t_f) = \left[p_d^x(t_f)\ p_d^y(t_f)\right]^T$ denote the initial and final values of the end-effector positions in Cartesian space.

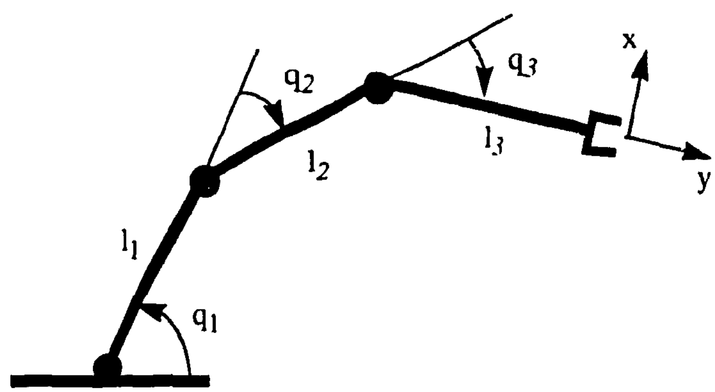


Fig. 3.1 Three-link revolute-joint planar manipulator.

All the kinematic dexterity measures described in Section 3.2.1 can be used as objective functions in the optimization. The RROKM scheme ensures that the condition number of the Jacobian matrix is as well-conditioned as possible while the end-effector tracks the desired trajectory. However, instead of using the condition number of the Jacobian matrix as the objective function directly, we use the KK isotropy measure as the objective function. It will be shown later that both measures yield the same result. The KK isotropy measure can be written as

$$M_3(q) = \frac{\left(\prod_{i=1}^{2} \sigma_i^2\right)^{1/2}}{\frac{1}{2}\sum_{i=1}^{2} \sigma_i^2} = \frac{\sqrt{det(JJ^T)}}{trace(JJ^T)/2}$$
$$= \frac{((l_1 l_2 s_2 + l_1 l_3 s_{23})^2 + (l_1 l_3 s_{23} + l_2 l_3 s_3)^2 + (l_2 l_3 s_3)^2)^{1/2}}{(l_1^2 + 2l_2^2 + 3l_3^2 + 2l_1 l_2 c_2 + 2l_1 l_3 c_{23} + 4l_2 l_3 c_3)/2}, \tag{3.17}$$

The last equality in (3.17) results from substituting (3.15) for J for the 3-link planar manipulator. For $l_1 = l_2 = l_3 = 1$ meter, (3.17) reduces to

$$M_3(q) = \frac{((s_2 + s_{23})^2 + (s_{23} + s_3)^2 + s_3^2)^{1/2}}{(3 + c_2 + c_{23} + 2c_3)}. \tag{3.18}$$

Therefore, the RROKM scheme is obtained by solving the constrained optimization problem:

$$\text{Minimize} \quad f(q) = -M_3(q) = -\frac{((s_2 + s_{23})^2 + (s_{23} + s_3)^2 + s_3^2)^{1/2}}{3 + c_2 + c_{23} + 2c_3},$$

$$\text{Subject to} \quad x_d^x - (l_1 c_1 + l_2 c_{12} + l_3 c_{123}) = 0$$
$$x_d^y - (l_1 s_1 + l_2 s_{12} + l_3 s_{123}) = 0. \tag{3.19}$$

Note that the KK isotropy measure is a concave function which has a maximum value of one. The objective function therefore takes the minus sign since

$$Maximum \quad f(x) = - Minimum \quad (-f(x)) . \tag{3.20}$$

The initial end-effector position is $x_d(t_o) = \left[p_d^x(t_o) \; p_d^y(t_o)\right]^T = \left[0.8 \; 0\right]^T$ and the final end-effector position is $x_d(t_f) = \left[p_d^x(t_f) \; p_d^y(t_f)\right]^T = \left[2.5 \; 0\right]^T$. In this simulation, the initial configuration of the manipulator is selected to be optimal, i.e., $M_3(q)$ is a local maximum. At every sampling instant while the redundant manipulator is moving along the desired trajectory, the optimizer minimizes the given objective function subject to the constraints, and determines the joint angle vector that is associated with the optimal value of the objective function. The solution is therefore locally optimal.

The simulation results are depicted in Fig. 3.2. Fig. 3.2 (a) shows the configurations of the redundant manipulator in which the objective function is the KK isotropy measure, and Fig. 3.2 (b) shows the variation of the condition number M_2 and the KK isotropy measure M_3 along the trajectory. For comparison, Fig. 3.3 (a) shows the configurations of the redundant manipulator when the objective function is the manipulability measure ($M_1(q)$). The variation of the condition number of J and the manipulability measure along the trajectory is depicted in Fig. 3.3 (b).

Although the simulation shows that both the condition number and the KK isotropy measure give the same simulation results, the analytic expression of the KK isotropy measure is easier to compute and more useful in practice. The condition number of the Jacobian matrix of the three-link planar manipulator can be written as

$$cond(J) = \frac{\sigma_1}{\sigma_2} = \frac{\sqrt{\lambda_1}}{\sqrt{\lambda_2}} = \left(\frac{T + \sqrt{T^2 - 4\Delta}}{T - \sqrt{T^2 - 4\Delta}}\right)^{1/2}, \tag{3.21}$$

where

$$T = trace(JJ^T) = l_1^2 + 2l_2^2 + 3l_3^2 + 2l_1l_2c_2 + 2l_1l_3c_{23} + 4l_2l_3c_3 \qquad (3.22)$$

and

$$\Delta = det(JJ^T) = (l_1l_2s_2 + l_1l_3s_{23})^2 + (l_1l_3s_{23} + l_2l_3s_3)^2 + (l_2l_3s_3)^2. \qquad (3.23)$$

For $l_1 = l_2 = l_3 = 1$ meter, the terms T and Δ become

$$T = 6 + 2c_2 + 2c_{23} + 4c_3 \qquad (3.24)$$

and

$$\Delta = (s_2 + s_{23})^2 + (s_{23} + s_3)^2 + s_3^2. \qquad (3.25)$$

Comparing (3.21), (3.24) and (3.25) with (3.18), we see that the computation of the KK isotropy measure is much simpler. More details on the comparison of the condition number and the KK isotropy measure can be found in [25].

Comparing Fig. 3.2 (a) with Fig 3.3 (a), we can see that the motions of the three-link planar manipulator using M_1 as the objective function are smoother than those using M_2 and M_3. However, using M_2 and M_3 in the optimization reduces system sensitivity. The simulations below will show the effect of measurement or rounding errors on the RROKM scheme. The simulation results are compared by choosing different objective functions in the optimization. The kinematically redundant manipulator is the same as the one considered previously. The desired trajectory of the end-effector is as specified by (3.16). The end-effector is required to move from the initial position

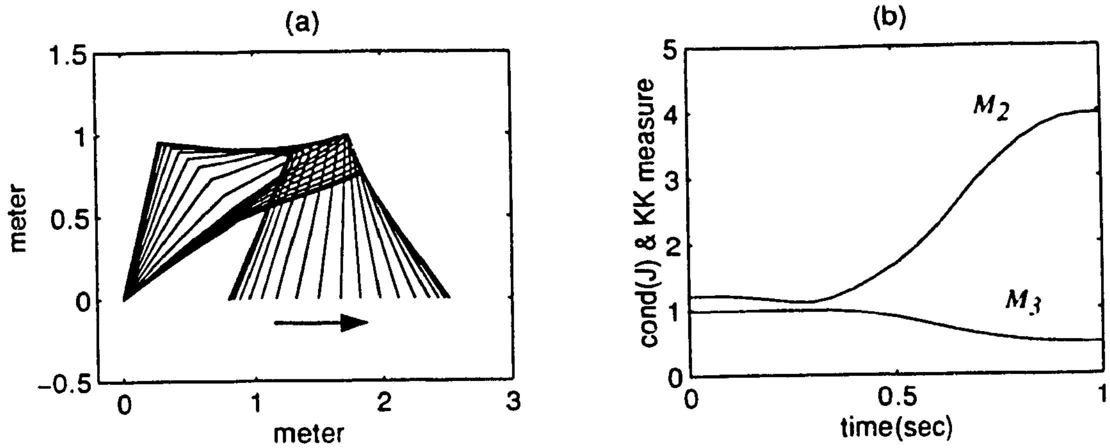


Fig. 3.2 (a) Configurations obtained using the RROKM scheme when the objective function is the KK isotropy measure. (b) The variation of the condition number M_2 and the KK isotropy measure M_3.

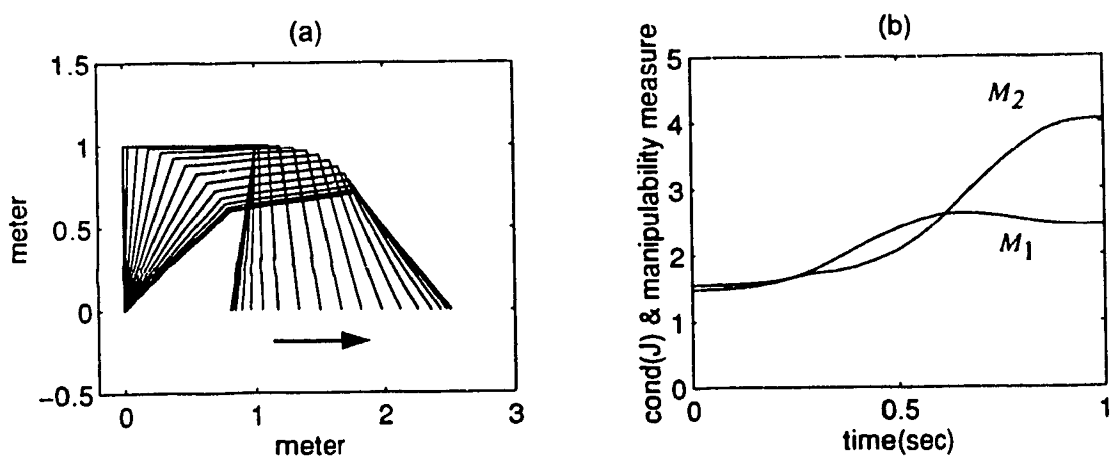


Fig. 3.3 (a) Configurations obtained using the RROKM scheme when the objective function is the manipulability measure. (b) The variation of the condition number M_2 and the manipulability measure M_1.

$$x_d(t_o) = \left[p_d^x(t_o)\ p_d^y(t_o)\right]^T = \left[1\ 0.25\right]^T$$

to the final position

$$x_d(t_f) = \left[p_d^x(t_f)\ p_d^y(t_f)\right]^T = \left[2\ 0.5\right]^T$$

in one second ($t_f = 1$).

Fig. 3.4 (a) shows the motion of the three-link planar manipulator using the RROKM scheme. The objective function is the KK isotropy measure. Fig. 3.4 (b) depicts the motion when the objective function is the manipulability measure. The condition numbers which correspond to the motion in Fig. 3.4 (a) are smaller over the entire trajectory than those corresponding to the motion in Fig. 3.4 (b). This is shown in Fig. 3.5, where the solid line represents the condition number corresponding to Fig. 3.4 (a) and the dashed line represents that corresponding to Fig. 3.4 (b). Next, we suppose that the end-effector velocity $\dot{x}$ is perturbed from $\dot{x}$ to $\dot{x} + \delta\dot{x}$, where

$$\delta\dot{x} = \begin{bmatrix} 0.001\dot{x}_1 \\ 0.0001\dot{x}_2 \end{bmatrix}.$$

Fig. 3.6 (a) depicts the relative error in the end-effector velocity in task space (the dotted line). The relative errors in the joint velocities are shown by the solid line for the motion in Fig. 3.4 (a), and the dashed line for the motion in Fig. 3.4 (b).

According to inequality (3.12), the relative error $\frac{\|\delta\dot{q}\|}{\|\dot{q}\|}$ is bounded by the relative error $\frac{\|\delta\dot{x}\|}{\|\dot{x}\|}$ times *cond(J)*. However, if $\delta\dot{x}$ and $\dot{x}$ are linearly dependent vectors, i.e., the perturbation in task velocity has the form

$$\delta\dot{x} = \begin{bmatrix}\alpha\dot{x}_1 \\ \alpha\dot{x}_2\end{bmatrix} = \alpha\dot{x}, \tag{3.26}$$

where α is a scalar, then both task and joint velocities have the same relative errors. To see this, let $\delta\dot{q}$ be the change in joint velocities corresponding to the change $\delta\dot{x}$. Then, from (3.2) we get

$$\delta\dot{x} = J\delta\dot{q}. \tag{3.27}$$

Multiplying both side of (3.2) by α, we get

$$\delta\dot{x} = \alpha\dot{x} = \alpha J\dot{q} = J\alpha\dot{q}. \tag{3.28}$$

Comparing (3.28) with (3.27), we have

$$\delta\dot{q} = \alpha\dot{q}. \tag{3.29}$$

The relative error in task velocity is given by

$$\frac{\|\delta\dot{x}\|}{\|\dot{x}\|} = \frac{|\alpha|\|\dot{x}\|}{\|\dot{x}\|} = |\alpha| \tag{3.30}$$

and the relative error in joint velocities is

$$\frac{\| \delta \dot{q} \|}{\| \dot{q} \|} = \frac{|\alpha| \| \dot{q} \|}{\| \dot{q} \|} = |\alpha|. \tag{3.31}$$

The above result can be illustrated for the planar manipulator by setting

$$\delta \dot{x} = \begin{bmatrix} 0.001 \dot{x}_1 \\ 0.001 \dot{x}_2 \end{bmatrix}.$$

The simulation results are plotted in Fig. 3.6 (b). The 2-norm has been used to compute the relative error plotted in Figures 3.6 (a) and (b).

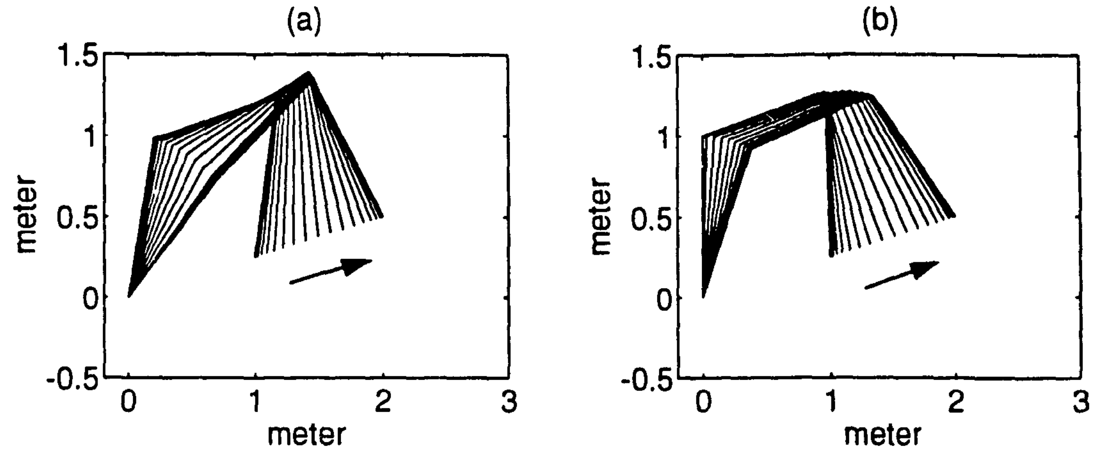


Fig. 3.4 Configurations obtained using the RROKM scheme. (a) With the KK isotropy measure. (b) With the manipulability measure.

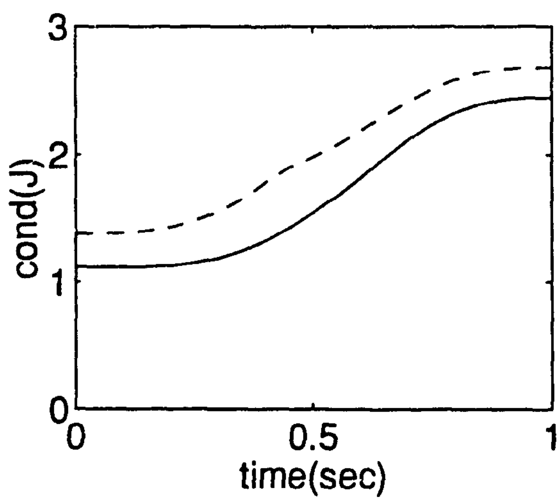
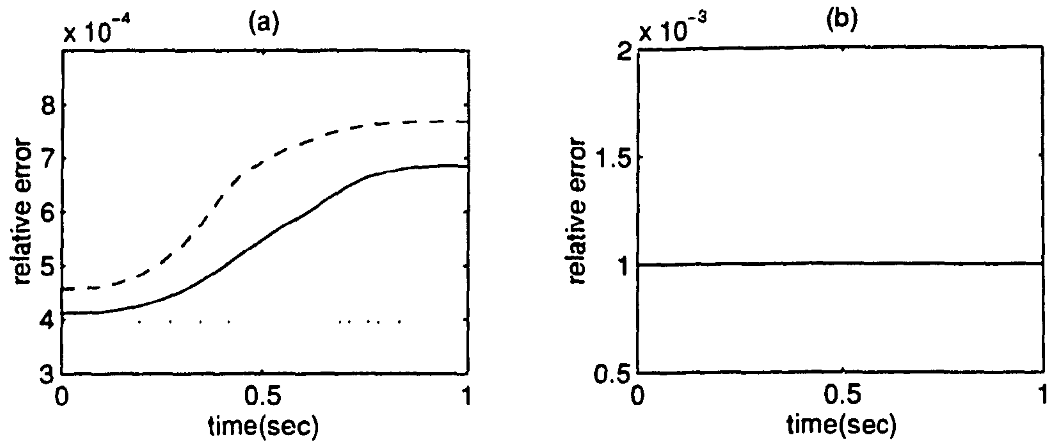


Fig. 3.5 Comparison of the condition numbers corresponding to the motions in Fig. 3.4: —— for 3.4 (a); ---- for 3.4 (b).



(a): $\delta \dot{x} = \begin{bmatrix} 0.001\dot{x}_1 \\ 0.0001\dot{x}_2 \end{bmatrix}$ (b): $\delta x = \begin{bmatrix} 0.001 x_1 \\ 0.001\dot{x}_2 \end{bmatrix}$

Fig. 3.6 Comparison of the effects of two different perturbations.

..... **relative error in task velocities**

— **relative error in joint velocities obtained with the KK isotropy measure**

--- **relative error in joint velocities obtained with the manipulability measure**

References

[1] C. R. Rao and S. K. Mitra, *Generalized Inverse of Matrices and Its Applications*, John Wiley & Sons, New York, 1971.

[2] D. E. Whitney, "Resolved Motion Rate Control of Manipulator and Human Prosthesis", *IEEE Trans. Man-Machine Systems*, 10, 1969, pp. 47-53.

[3] A. Liegeois, "Automatic Supervisory Control of the Configuration and Behavicr of Multi-body Mechanisms," *IEEE Trans, Systems, Man, and Cybernetics*, SMC-7, 1977, pp. 868-871.

[4] C. Chevellereau and W, Khalil, "A New Method for the Solution of the Inverse Kinematics of Redundant Robots," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1988, pp.37-42.

[5] H. Zghal, R. V. Dubey and J. A. Euler, "Efficient Gradient Projection Optimization for Manipulators with Multiple Degrees of Redundancy," *Proc. IEEE Int. Conf. on Robotics and Automation*, Vol. 2, 1990, pp. 1006-1011.

[6] J. Baillieul, J. Hollerbach and R. Brockett, "Programming and Control of Kinematically Redundant Manipulators," *Proc. 23rd IEEE Conf. on Decision and Control*, Dec. 1984, pp. 768-774.

[7] J. Baillieul, "Kinematic Programming Alternatives for Redundant Manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1985, pp. 722-728.

[8] J. Baillieul, "Avoiding Obstacles and Resolving Kinematic Redundancy," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1986, pp. 1698-1704.

[9] J. Baillieul, R. Brockett, J. Hollerbach, D. Martin, R. Percy and R. Thomas, "Kinematically Redundant Robot Manipulators," *Proc. NASA Workshop on Space Telerobotics*, Vol. 2, Jan. 1987, pp. 245-255.

[10] J. Baillieul, "Design of Kinematically Redundant Mechanisms," *Proc. 24th IEEE Conf. on Decision and Control*, Dec. 1985, pp. 18-21.

[11] T. Yoshikawa, "Analysis and Control of Robot Manipulators with Redundancy,"

Robotics Research: *The First Int. Symposium*, MIT Press, Cambridge, 1984. pp.735-747

[12] Y. Nakamura and H. Hanafusa, "Task-Priority Based Redundancy Control of Robot Manipulators," *Int. J. of Robotics Research*, Vol. 6, No. 4, Summer, 1987, pp. 3-15.

[13] P. H. Chang, "A Closed-Form Solution for Inverse Kinematics of Robot Manipulators with Redundancy," *IEEE J. of Robotics and Automation*, Vol. RA-3, No. 5, 1987, pp. 393-403.

[14] C. A. Klein, "Use of Redundancy in the Design of Robotics Systems," *Robotics Research: Second Int. Symposium*, MIT Press, Cambridge, MA, 1985, pp. 207-214.

[15] J. M. Hollerbach and K. C. Suh, "Redundancy Resolution of Manipulators Through Torque Optimization," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1985, 1016-1021.

[16] D. Wang and Y. Hamam, "Optimal Trajectory Planning of Manipulators with Collision Detection and Avoidance," *Int. J. of Robotics Research*, Vol. 11, No. 5, Oct. 1992, pp. 460-468.

[17] A. Ben-Israel and T. N. E. Greville, *Generalized Inverses: Theory and Applications*, John Wiley & Sons, New York, 1974.

[18] V. C. Klema and A. T. Laub, "The Singular Value Decomposition: Its Computation and Some Applications," *IEEE Trans. Automatic Control*, AC-25(2), 1980, pp. 164-176.

[19] T. Yoshikawa, "Manipulability of Robotic Mechanisms," *Int. J. of Robotics Research*, Vol. 4, No. 2, Summer, 1985. pp 3-9.

[20] R. P. Paul and C. N. Stevenson, "Kinematics of Robot Wrists," *Int. J. of Robotics Research*, Vol. 2, No. 1, Spring, 1983, pp. 31-38.

[21] G. H. Golub and C. F. Van Loan, *Matrix computations*, 2nd ed., Johns Hopkins University Press, Baltimore, 1989.

[22] P. E. Gill, W. Murray and M. H. Wright, *Numerical Linear Algebra and Optimiza-*

tion, Vol. 2, Addison-Wesley, Reading, MA, 1991.

[23] J. K. Salisbury and J. J. Craig, "Articulated Hands: Force Control and Kinematic Issues," *Int. J. of Robotics Research*, Vol. 1, No. 1, Spring, 1982, pp. 4-16.

[24] S. L. Chiu, "Kinematic Characterization of Manipulators: An Approach to Defining Optimality," *IEEE Int. Conf. on Robotics and Automation*, 1988, pp. 828-833.

[25] J.-O. Kim and P. K. Khosla, "Dexterity Measures for Design and Control of Manipulators," *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, IROS'91, Nov. 3-5, 1991, pp. 758-763.

[26] C. A. Klein and B. E. Blaho, "Dexterity Measures for the Design and control of Kinematically Redundant Manipulators," *Int. J. of Robotics Research*, Vol. 6, No. 2, Summer, 1987, pp. 72-83.

[27] J. Angeles and C. S. Lopez-Cajun, Kinematic Isotropy and the Conditioning Index of Serial Robotic Manipulators, *Int. J. of Robotics Research*, Vol. 11, No. 6, Dec., 1992, pp. 560-571.

[28] A. Grace, *Optimization toolbox for use with MATLAB*, The MathWorks, Inc., Natick, MA, Nov., 1990.

[29] J. Angeles, F. Ranjbaran and R. V. Patel. "On the Design of the Kinematic Structure of Seven-Axis Redundant Manipulators for Maximum Conditioning," *Proc. IEEE Int. Conf. Robotics and Automation*, 1992, pp. 494-499.

CHAPTER 4 REDUNDANCY RESOLUTION BASED ON OPTIMIZATION OF DYNAMIC MEASURES

4.1 Introduction

In this chapter, we consider the problem of redundancy resolution based on optimizing a dynamic measure (RRODM). For many years, researchers have studied the problem of dynamic control of manipulators to achieve high speed or high precision motion. Several researchers have used optimal control to achieve minimum-time motion [1][2][3] and minimum-energy motion [4][5][6] for manipulators. In this chapter we will focus on the problem of redundancy resolution and control based on optimization of dynamic measure. That is, the redundant manipulators will exploit redundancy to be as close as possible to dynamic isotropy [7] while tracking a desired end-effector trajectory. Dynamic isotropy ensures that the mass matrix of the manipulator is as well conditioned as possible. Hence, system sensitivity is reduced when the mass matrix is used as the transformation matrix in a control strategy.

In this chapter, after presenting several dynamic dexterity measures, we develop a control strategy involving RRODM. Then, in order to show the effectiveness of the RRODM scheme, we consider the system sensitivity by comparing the simulation results with and without minimizing the dynamic conditioning measures.

4.2 Development of the RRODM Scheme

The dynamic equation of motion of a rigid-link manipulator can be written in the form (see equation (2.70)):

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau. \tag{4.1}$$

The RRODM scheme that is discussed in this section will ensure that the problem of tracking a trajectory is carried out in an environment that is dynamically as well conditioned as possible. The controller for this control scheme will be developed such that the manipulator tracks a desired trajectory while minimizing a dynamic dexterity measure. Several dynamic dexterity measures for manipulators will be discussed in this section. Optimal control theory is the main tool in the development of the RRODM scheme. The formulation for the controller developed for the RRODM results in a two-point boundary-value problem (TPBVP).

4.2.1 Dynamic Dexterity Measures for Manipulators

A. The Generalized Inertia Ellipsoid

The generalized inertia ellipsoid (GIE) suggested by Asada [8] is an approach for a geometric representation of manipulator dynamics. The GIE configuration of a manipulator in its workspace is useful for the analysis and design of manipulators. Asada defined the generalized inertia tensor as

$$M_x(q) = [J(q)M^{-1}(q)J^T(q)]^{-1}, \tag{4.2}$$

where $J(q)$ is an $m \times n$ Jacobian matrix and $M(q)$ is the $n \times n$ mass matrix associated with the joint displacement q. By examining the eigenvalues of $M_x(q)$, he suggests that the GIE can show how the manipulator accelerates in various Cartesian directions. Asada defined a graphical ellipsoid associated with the GIE. The axes of the ellipsoid are associated with the eigenvectors of the generalized inertia tensor $M_x(q)$, and the reciprocals of the square root of the corresponding eigenvalues provide the lengths of the axes of the ellipsoid. The conditioned points in the manipulator workspace are indicated by the volume of the ellipsoids. The well-conditioned points in the workspace are indicated by large volume ellipsoids while the ill-conditioned points in the workspace are indicated by small volume ellipsoids.

B. The Dynamic Manipulability Ellipsoid

Yoshikawa extended the kinematic manipulability measure [9] to a dynamic manipulability measure [10]. Based on the relationship between the acceleration of the end-effector and the joint torques:

$$\ddot{x}-\dot{J}\dot{q} = JM^{-1}\left[\tau - C(q,\dot{q})\,\dot{q} - g(q)\right], \tag{4.3}$$

the measure of dynamic manipulability is defined by

$$W_d = \sqrt{det\left[J\,(M^T M)^{-1} J^T\right]}, \tag{4.4}$$

where $\ddot{x}$ is the $m \times 1$ end-effector acceleration vector, and J is the $m \times n$ Jacobian matrix. Like the kinematic measure of manipulability, we can express JM^{-1} by its singular value decomposition:

$$JM^{-1} = U\Sigma V^T, \tag{4.5}$$

where U and V are $m \times m$ and $n \times n$ orthogonal matrices, respectively and

$$\Sigma = \begin{bmatrix} \sigma_i & 0 \\ 0 & 0 \end{bmatrix}, \tag{4.6}$$

where the (non-zero) singular values σ_i can be ordered as

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_m > 0. \tag{4.7}$$

Hence, as was done for the kinematic measure of manipulability, we can define the dynamic measure of manipulability as

$$W_d = \prod_{i=1}^{m} \sigma_i. \tag{4.8}$$

C. The Dynamic Conditioning Index

Recently, Ma and Angeles [7] introduced the dynamic conditioning index (DCI) which is defined as

$$\mu = \frac{1}{2}\eta^T Q \eta, \tag{4.9}$$

where Q is a diagonal weighting matrix, and η is a vector which represents a measure of the distance between the real mass matrix and an ideal mass matrix for the same manipulator. By defining an $n \times n$ error matrix

$$E(q) = M(q) - \varepsilon I = \begin{bmatrix} E_{11} & E_{12} & \dots & E_{1n} \\ E_{21} & E_{22} & \dots & E_{2n} \\ \dots & \dots & \dots & \dots \\ E_{n1} & E_{n2} & \dots & E_{nn} \end{bmatrix}, \tag{4.10}$$

where $M(q)$ is the $n \times n$ mass matrix of a manipulator, I denotes the $n \times n$ identity matrix, and ε is defined as

$$\varepsilon = \frac{1}{n} trace[M(q)], \tag{4.11}$$

we can get the vector η which consists of $n(n+1)/2$ components:

$$\eta = \begin{bmatrix} E_{11} \\ E_{12} \\ \cdot \\ \cdot \\ E_{1n} \\ E_{22} \\ E_{23} \\ \cdot \\ \cdot \\ E_{nn} \end{bmatrix}. \tag{4.12}$$

The DCI is a quadratic form and describes the dynamic conditioning of the mass matrix. A mass matrix of the εI is said to be dynamically isotropic [7], i.e., it has the best conditioning possible (condition number = 1). We will use this index as a criterion in our RRODM scheme.

4.2.2 Formulation of the Problem of RRODM

The forward kinematics of a manipulator are defined by

$$x = f(q)\,, \tag{4.13}$$

where x is an $m \times 1$ vector in the task space and q is an $n \times 1$ vector in the joint space. Differentiation of (4.13) with respect to time yields the differential kinematics:

$$\dot{x} = J(q)\,\dot{q}. \tag{4.14}$$

The nonlinear equation of motion of a rigid-link manipulator can be written in the form

$$M(q)\,\ddot{q} + C(q,\dot{q})\,\dot{q} + g(q) = \tau. \tag{4.15}$$

Equation (4.15) describes the manipulator dynamics without any payload. Now, by defining the state vector

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \tag{4.16}$$

equations (4.13) and (4.14) can be written as

$$x = f(y_1) \tag{4.17}$$

and

$$\dot{x} = J(y_1)\, y_2. \tag{4.18}$$

Equation (4.15) can be expressed in the state space form:

$$\dot{y} = F(y, \tau), \tag{4.19}$$

where

$$F(y, \tau) = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} y_2 \\ N(y_1, y_2) + D(y_1)\,\tau \end{bmatrix}, \tag{4.20}$$

$$N(y_1, y_2) = -D(y_1)\,[C(y_1, y_2)\, y_2 + g(y_1)] \tag{4.21}$$

and

$$D(y_1) = M^{-1}(y_1). \tag{4.22}$$

The RRODM requires that the end-effector of a kinematically redundant manipulator tracks a desired trajectory while minimizing a dynamic conditioning measure. Hence the RRODM can be formulated as the following constrained optimization problem:

$$\text{Minimize } P = \varphi(y((t_f), t_f)) + \int_{t_o}^{t_f} L(y, \tau, t)\, dt, \tag{4.23}$$

$$\text{Subject to } \dot{y} = F(y, \tau, t), \tag{4.24}$$

where the terms in the performance index P are given by

$$\varphi(y(t_f), t_f) = \frac{1}{2}e_f^T W_1 e_f + \frac{1}{2}\dot{e}_f^T W_2 \dot{e}_f, \tag{4.25}$$

$$L(y, \tau, t) = \mu + \frac{1}{2}\tau^T R \tau + \frac{1}{2}e^T S e, \tag{4.26}$$

$$e = x - x_d, \tag{4.27}$$

$$\dot{e} = \dot{x} - \dot{x}_d, \tag{4.28}$$

$$e_f = e(t_f). \tag{4.29}$$

In (4.26), μ is the dynamic conditioning index [7] given by

$$\mu = \frac{1}{2}\eta^T Q \eta. \tag{4.30}$$

In (4.25), (4.26) and (4.30), W_1, W_2 and S are $m \times m$ positive definite weighting matrices, and R and Q are $n \times n$ and $n(n+1)/2 \times n(n+1)/2$ positive definite weighting matrices, respectively.

The optimization problem presented in (4.23) and (4.24) is an optimal control problem. Optimal control theory can be used to solve this problem. The performance index described by (4.23) is minimized over the entire motion of the end-effector of the manipulator. Minimizing the first term in (4.26) controls dynamic conditioning because it causes the mass matrix involved to be close to dynamic isotropy (εI). The second term in (4.26) represents the control energy and the third term in (4.26) ensures that the end-effector tracks the desired trajectory. Equation (4.25) specifies constraints on the final state of the end-effector.

Note that the optimization problem is characterized by compromise and trade-offs of different criteria via appropriate choices of the weighting matrices in the performance index. For example, if conditioning is considered to be very important, a larger value of Q should be chosen relative to the other weighting matrices. If it is more important that tracking error be very small, then the value of S should be much larger than that of Q or R.

An optimal controller is designed based on a model of a plant and involves optimization of a performance index. We assume that the plant is described by the general non-linear equation

$$\dot{y}(t) = F(y(t), u(t), t), \tag{4.31}$$

where $y(t)$ is a state variable and $u(t)$ is a control input. The performance index which is to be optimized over a fixed time interval $t_0 \leq t \leq t_f$ can be written in the general form:

$$P = \varphi(y(t_f), t_f) + \int_{t_0}^{t_f} L(y(t), u(t), t)\, dt, \tag{4.32}$$

where $y(t_f)$ denotes the state vector at the final instant $t = t_f$. Thus, φ is taken to depend on the final state and L depends on the behavior of the system in reaching this final state.

The optimal control problem is then to determine the optimal input $u^*(t)$ over a fixed time interval $t_0 \leq t \leq t_f$ to drive the plant (4.31) along a trajectory $y^*(t)$ such that the performance index P is minimized and the final state satisfies

$$\phi(y(t_f), t_f) = 0, \tag{4.33}$$

where ϕ describes the constraint on the final state of the system. An outline of an approach for solving this problem was given in Chapter 2. The approach is based on introducing Lagrange λ and γ, and defining

$$\hat{P} = \varphi(y(t_f), t_f) + \gamma^T \phi(y(t_f), t_f) + \int_{t_0}^{t_f} (H(y, u, t) - \lambda^T \dot{y})\, dt, \tag{4.34}$$

where the Hamiltonian, H, is defined as

$$H(y, u, t) = L(y, u, t) + \lambda^T F(y, u, t) \tag{4.35}$$

The necessary conditions described in Chapter 2 are obtained by applying a small increment to (4.34) and setting it zero, i.e., $d\hat{P} = 0$ [11][12].

The optimal control problem with the performance index (4.32)

$$P = \varphi(y(t_f), t_f) + \int_{t_o}^{t_f} L(y(t), u(t), t)\, dt \tag{4.36}$$

is called the Bolza problem [13]. In the case that $L = 0$, the performance index reduces to

$$P = \varphi(y(t_f), t_f). \tag{4.37}$$

The optimal control problem with the performance index (4.37) is called the Mayer problem [13]. If $\varphi = 0$ in equation (4.32), then

$$P = \int_{t_o}^{t_f} L(y(t), u(t), t)\, dt. \tag{4.38}$$

This problem is called the Lagrange problem [13].

In addition, the performance index can be chosen based on an application. If it is required to drive the system from an initial state t_0 to a final state t_f in the minimum time, the performance index should be chosen as

$$P = \int_{t_o}^{t_f} dt. \tag{4.39}$$

The optimal control problem with the performance index (4.39) is called the minimum-time problem. The minimum-fuel problem can be obtained by defining the performance index

$$P = \int_{t_o}^{t_f} |u|\, dt. \tag{4.40}$$

In a fixed time interval, if we want to minimize the energy throughout the intermediate state and also the final state, then the performance index to be minimized in the minimum-energy problem is

$$P = \frac{1}{2}y^T(t_f)\,W y(t_f) + \int_{t_o}^{t_f}\left[\frac{1}{2}y^T(t)\,S y(t) + \frac{1}{2}u^T(t)\,R u(t)\right]dt\,, \tag{4.41}$$

where W, R and S are positive definite weighting matrices.

If it is requited to track a desired trajectory $y_d(t)$ while minimizing some criteria, the performance index (4.32) should be employed with

$$\varphi(y(t_f), t_f) = \frac{1}{2}e_f^T W e_f \tag{4.42}$$

and

$$L(y, u, t) = \frac{1}{2}e^T S e + \frac{1}{2}u^T R u, \tag{4.43}$$

where,

$$e = y(t) - y_d(t)\,, \tag{4.44}$$

$$e_f = y(t_f) - y_d(t_f) \tag{4.45}$$

and W, S and R are positive definite weighting matrices. The weighting matrices W and S in (4.42) and (4.43) determine the tracking errors of the system. By selecting appropriate weighting matrices W and S, the tracking errors can be reduced to acceptable values. In practice, it is generally necessary to run several computer simulations to adjust the weighting matrices until a desired performance is achieved.

4.2.3 Derivation of the Optimal Controller

Since the Hamiltonian function for the given problem is

$$H(y,\tau,t) = L(y,\tau,t) + \lambda^T F(y,\tau,t), \tag{4.46}$$

the state equation is given by

$$\dot{y} = \frac{\partial H}{\partial \lambda} = F(y,\tau,t) = \begin{bmatrix} y_2 \\ N(y_1,y_2) + D(y_1)\tau \end{bmatrix}, \tag{4.47}$$

which is exactly the same as (4.20) and the costate equation is obtained as

$$\begin{aligned} \dot{\lambda} &= -\frac{\partial H}{\partial y} = -\frac{\partial L}{\partial y} - \left[\frac{\partial F}{\partial y}\right]^T \lambda \\ &= -\begin{bmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{bmatrix} - \begin{bmatrix} \frac{\partial F_1}{\partial y_1} & \frac{\partial F_1}{\partial y_2} \\ \frac{\partial F_2}{\partial y_1} & \frac{\partial F_2}{\partial y_2} \end{bmatrix}^T \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} \end{aligned} \tag{4.48}$$

Noting that in (4.20) $F_1 = y_2$, $F_2 = F_2(y_1,y_2)$ and $L = L(y_1)$, we have

$$\dot{\lambda} = -\begin{bmatrix} \frac{\partial L}{\partial y_1} \\ 0 \end{bmatrix} - \begin{bmatrix} 0 & \left[\frac{\partial F_2}{\partial y_1}\right]^T \\ I & \left[\frac{\partial F_2}{\partial y_2}\right]^T \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}$$

$$
= -\begin{bmatrix} \dfrac{\partial L}{\partial y_1} + \left[\dfrac{\partial}{\partial y_1}[N(y_1, y_2) + D(y_1)\tau]\right]^T \lambda_2 \\ \lambda_1 + \left[\dfrac{\partial}{\partial y_2}[N(y_1, y_2)]\right]^T \lambda_2 \end{bmatrix}. \tag{4.49}
$$

The stationary condition gives:

$$
\begin{aligned}
0 = \frac{\partial H}{\partial \tau} &= \frac{\partial L}{\partial \tau} + \left[\frac{\partial F}{\partial \tau}\right]^T \lambda \\
&= \frac{\partial L}{\partial \tau} + \begin{bmatrix} \dfrac{\partial F_1}{\partial \tau} \\ \dfrac{\partial F_2}{\partial \tau} \end{bmatrix}^T \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} \\
&= R\tau + \left[\frac{\partial F_2}{\partial \tau}\right]^T \lambda_2 \\
&= R\tau + D(y_1)^T \lambda_2,
\end{aligned} \tag{4.50}
$$

from where we obtain the optimal input τ^* :

$$
\tau^* = -R^{-1} D(y_1)^T \lambda_2 . \tag{4.51}
$$

Substituting equation (4.51) into equation (4.49) yields $4n$ equations with $4n$ unknowns y_1, y_2, λ_1 and λ_2 ($M(y_1)$, $C(y_1, y_2)$ and $g(y_1)$ will be written as M, C and g hereafter):

$$\dot{y}_1 = y_2$$

$$\dot{y}_2 = -M^{-1}(Cy_2 + g) - M^{-1}R^{-1}(M^{-1})^T\lambda_2,$$

$$\dot{\lambda}_1 = -\left[\frac{\partial\eta}{\partial y_1}\right]^T Q\eta - J^T Se + \left[\frac{\partial}{\partial y_1}[M^{-1}(Cy_2 + g) + M^{-1}R^{-1}(M^{-1})^T\lambda_2]\right]^T\lambda_2,$$

$$\dot{\lambda}_2 = -\lambda_1 + \left[\frac{\partial}{\partial y_2}[M^{-1}(Cy_2 + g)]\right]^T\lambda_2. \tag{4.52}$$

The final condition is determined by (2.67). The final time t_f in our application is fixed, so $dt_f = 0$ and the second term of (2.67) is automatically equal to zero. Because $y(t_f)$ is not fixed, $dy(t_f)$ is not zero. Therefore, it is required that [12][11]

$$\left(\frac{\partial\varphi}{\partial y} - \lambda\right)\bigg|_{t = t_f} = 0 \tag{4.53}$$

(Note that there is no function ϕ in our application (see (2.67))). Substituting (4.25) into (4.53) we get

$$\frac{\partial}{\partial y}\left[\frac{1}{2}e_f^T W_1 e_f + \frac{1}{2}\dot{e}_f^T W_2 \dot{e}_f\right] = \lambda|_{t = t_f}. \tag{4.54}$$

The initial condition

$$y(t_0) = y_0, \tag{4.55}$$

where y_0 is given. Equation (4.52) with the boundary conditions (4.54) and (4.55) denotes a two-point boundary-value problem (TPBVP).

4.3 Simulation Examples

The RRODM scheme results in a TPBVP which can be solved numerically in a number of ways. Many computer routines are currently available for solving the nonlinear TPBVP. One routine which uses the "multiple shooting" method is BVPMS in the IMSL library [14]. To describe this method, we consider Fig. 4.1. The differential equations with boundary conditions are integrated by initial value methods from one boundary to the other boundary. Several trial integrations are performed until both boundary conditions are satisfied. The integration of the differential equation resembles the trajectory of a shot from a gun (the initial boundary value) to a target (the final boundary value).

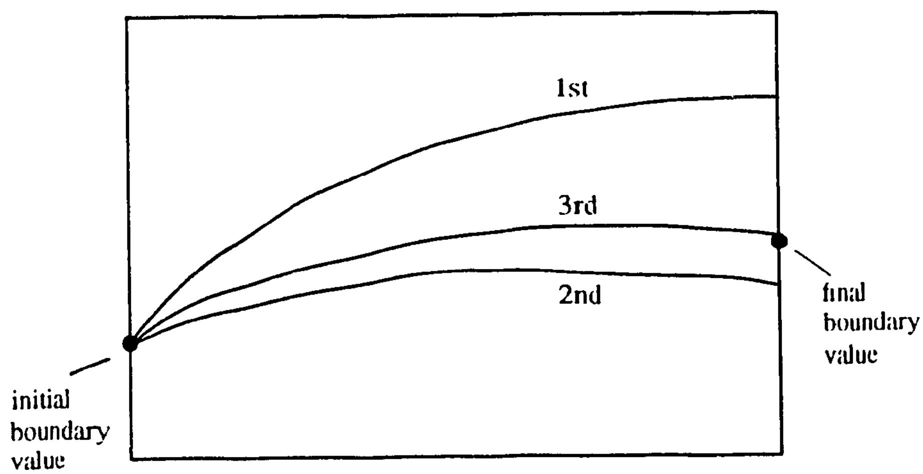


Fig. 4.1 Illustration of the multiple shooting method for solving a TPBVP.

To illustrate the RRODM scheme, we consider the planar three-link manipulator in a horizontal plane as shown in Fig. 3.1. For the dynamic equation (4.15) of the three-link manipulator, the mass matrix M is a 3×3 symmetric positive definite matrix:

$$M = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}, \tag{4.56}$$

where

$$\begin{aligned}
M_{11} &= a_1 + 2ac_2 + 2bc_{23} + 2dc_3 \\
M_{12} &= a_2 + ac_2 + bc_{23} + dc_3 \\
M_{13} &= a_3 + bc_{23} + 2dc_3 \\
M_{21} &= M_{12} \\
M_{22} &= a_2 + 2dc_3 \\
M_{23} &= a_3 + dc_3 \\
M_{31} &= M_{13} \\
M_{32} &= M_{23} \\
M_{33} &= a_3
\end{aligned}$$

and

$$\begin{aligned}
a_1 &= (m_1 + m_2 + m_3)\, l_1^2 + (m_2 + m_3)\, l_2^2 + m_3 l_3^2 \\
a_2 &= (m_2 + m_3)\, l_2^2 + m_3 l_3^2 \\
a_3 &= m_3 l_3^2 \\
a &= (m_2 + m_3)\, l_1 l_2 \\
b &= m_3 l_1 l_3 \\
d &= m_3 l_2 l_3.
\end{aligned}$$

The matrix C is defined as

$$C = \begin{bmatrix} 0 & h_{12}(2\dot{q}_1 + \dot{q}2\ ) & h_{13}(2\dot{q}_1 + 2\dot{q}_2 + \dot{q}_3) \\ h_{21}\dot{q}_1 & 0 & h_{23}(2\dot{q}_1 + 2\dot{q}_2 + \dot{q}_3) \\ h_{31}\dot{q}_1 & h_{32}(2\dot{q}_1 + \dot{q}_2) & 0 \end{bmatrix}, \tag{4.57}$$

where

$$h_{12} = -h_{21} = -as_2 - bs_{23}$$

$$h_{12} = -h_{21} = -as_3 - bs_{23}$$

$$h_{23} = -h_{32} = -ds_3.$$

The robot link lengths are chosen as $l_1 = l_2 = l_3 = 1$ meter, and the link masses are chosen as $m_1 = m_2 = m_3 = 10$ kg. The weighting matrices which appear in equations (4.52) - (4.54) are chosen as follows: $R^{-1} = diag(500)$, $S = diag(10000)$, $Q = diag(0.01)$, $W_1 = diag(1000)$, and $W_2 = diag(1000)$. The end-effector of the manipulator is required to track the trajectory specified by equation (3.16). The initial boundary conditions are given by

$$y_1(t_0) = \begin{bmatrix} -35° & 135° & -135° \end{bmatrix}^T, \tag{4.58}$$

$$y_2(t_0) = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T. \tag{4.59}$$

The final boundary condition are specified by equations (4.54) and (4.55). The forward kinematic equation of the manipulator is:

$$f(y_1) = \begin{bmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} \end{bmatrix}, \tag{4.60}$$

where $y_1 = \left[y_{11}\ y_{12}\ y_{13}\right]^T = \left[q_1\ q_2\ q_3\right]^T$, $c_1 = \cos(y_{11})$, $c_{12} = \cos(y_{11}+y_{12})$, $s_1 = \sin(y_{11})$, $s_{12} = \sin(y_{11}+y_{12})$, and so on. The Jacobian matrix is specified by equation (3.15).

For the three-link planar manipulator, the optimal controller consists of 12 differential equations which are given by equations (4.52) with 12 boundary conditions (6 initial boundary conditions given by (4.58) and (4.59); and 6 final boundary conditions specified by (4.54)). The optimal control problem was solved using the IMSL routine DBVPMS in double precision. The routine DIPRK in the IMSL library was used to solve the 12 equations corresponding to (4.52) at each "shot". If we take m shooting points (including the initial and final boundary points), then a system of $12 \times m$ nonlinear equations are solved by Newton's method.

The simulation results are shown in Figures 4.2 - 4.5. The configurations of the three-link planar manipulator resulting from the RRODM scheme are plotted in Fig. 4.2. The end-effector moves along the straight line in one second from the initial point $\left[1.4647\ -0.1623\right]^T$ to the final point $\left[1.8\ 0.1\right]^T$. The tracking error of the end-effector is shown in Fig. 4.3, where the solid line denotes the tracking error in the x direction, and the dashed line represents the tracking error in the y direction. It should be noted that in RROKM, the end-effector tracking performance is achieved by satisfying the constraint in (3.13) which results in zero tracking error. In RRODM, however, the end-effector tracking performance is obtained by achieving a compromise between the minimization of μ, $\tau^T R \tau$ and $e^T S e$. Therefore, the tracking error may not be exactly zero. Fig. 4.4 shows the variation in the DCI along the trajectory. Note that the magnitude of DCI depends on the weighting matrix Q. The optimal joint torques are depicted in Fig. 4.5, where the solid, dashed and dotted lines denote the torques applied at joints 1, 2 and 3, respectively.

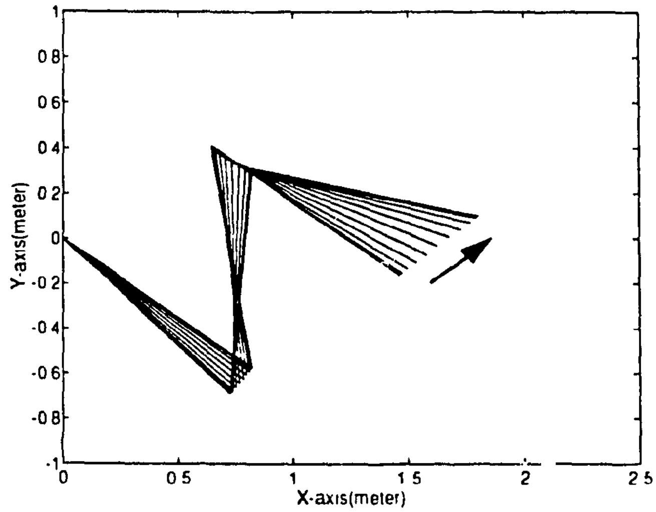


Fig. 4.2 Robot configurations resulting from RRODM.

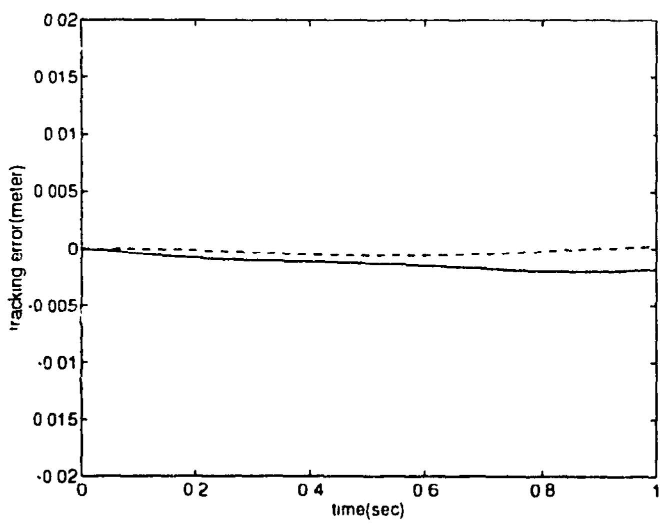


Fig. 4.3 End-effector tracking error resulting from RRODM.

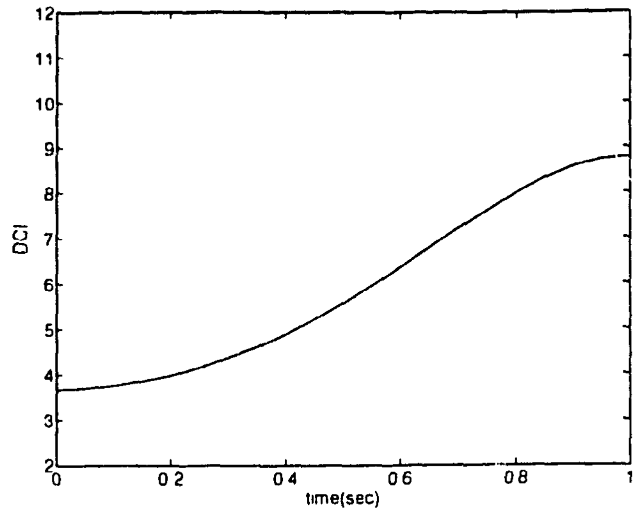


Fig. 4.4 Variation of the DCI resulting from RRODM.

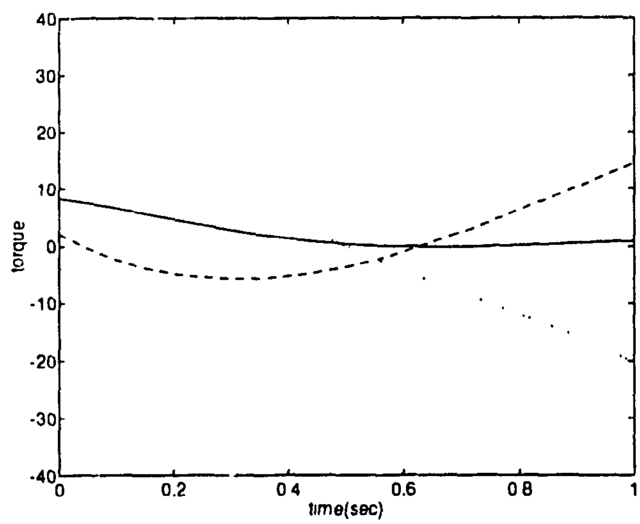


Fig. 4.5 Joint torque resulting from RRODM.

The system sensitivity with regard to dynamics can be shown by rewriting the dynamic equation (4.15) as

$$\ddot{q} = M^{-1}(\tau - C\dot{q} - g). \tag{4.61}$$

Solving (4.61) involves the inversion of the mass matrix M. The inversion of the mass matrix also appears in the well-known linearization techniques [15][16]. More discussion on the issue of the inversion of the mass matrix can be found in [7].

Let $\tilde{\tau} = \tau - C\dot{q} - g$, equation (4.61) can be rewritten as

$$M\ddot{q} = \tilde{\tau}. \tag{4.62}$$

When we compute the joint acceleration $\ddot{q}$ for dynamic control or dynamic simulations, suppose that the right-hand side of (4.62) is perturbed from $\tilde{\tau}$ to $\tilde{\tau} + \delta\tilde{\tau}$, the relative change of the joint acceleration is

$$\frac{\|\delta\ddot{q}\|}{\|\ddot{q}\|} \le cond(M)\frac{\|\delta\tilde{\tau}\|}{\|\tilde{\tau}\|}. \tag{4.63}$$

Equation (4.63) shows that the relative change of the joint acceleration can be a magnification of the relative change in $\tilde{\tau}$ by a factor equal to the condition number of the mass matrix M, i.e., *cond(M)* quantifies the sensitivity of the problem of solving (4.62) for $\ddot{q}$.

In order to show how the system described by equation (4.62) is sensitive to the condition number of the mass matrix M, we use the RRODM with $Q = 0$, i.e., without minimizing the DCI. The simulation results are shown in Fig. 4.6, 4.7 and 4.8. In addition, Fig. 4.9 shows the variation of the condition number of the mass matrix M when the

RRODM with $Q = \text{diag}(0.01)$ is used, which corresponds to Figs. 4.2 - 4.5, while Fig. 4.10 shows the variation in the condition number of M when $Q = 0$. Comparing Figs. 4.9 and 4.10 we see a significant increase in the condition number of M when minimization of DCI is not included in RRODM scheme.

Finally, we assume that the right-hand side of equation (4.62) is perturbed from $\tilde{\tau} = \left[\tilde{\tau}_1 \ \tilde{\tau}_2 \ \tilde{\tau}_3\right]^T$ to $\tilde{\tau} + \delta\tilde{\tau}$, where, $\delta\tilde{\tau} = \left[0.001\tilde{\tau}_1 \ 0.001\tilde{\tau}_2 \ 0.0001\tilde{\tau}_3\right]^T$. Using the 2-norm to calculate the relative error, Fig 4.11 shows the relative error resulting from RRODM that includes DCI, while Fig. 4.12 depicts the relative error when the RRODM does not include DCI.

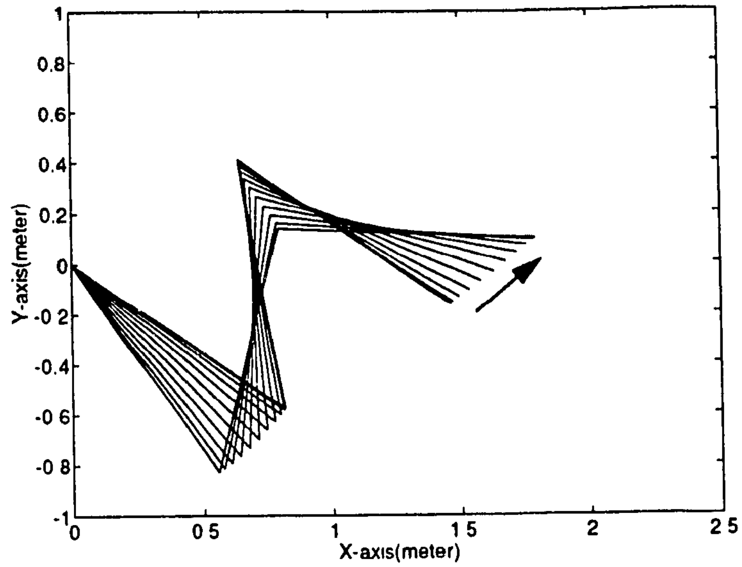


Fig. 4.6 Robot Configurations resulting from RRODM without DCI.

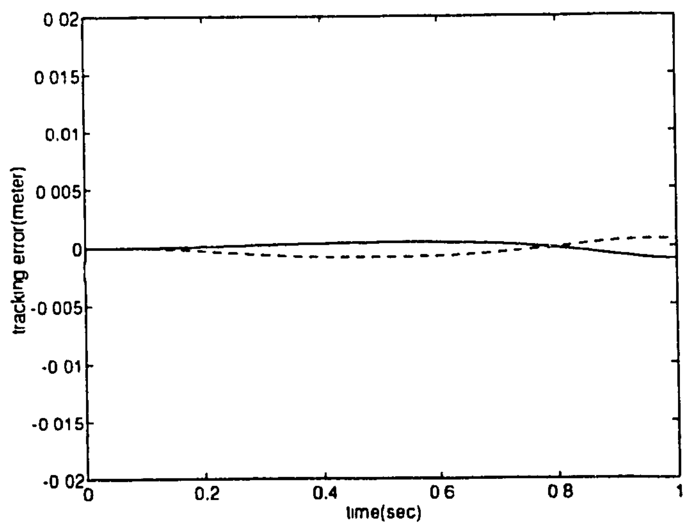


Fig. 4.7 End-effector tracking error resulting from RRODM without DCI.

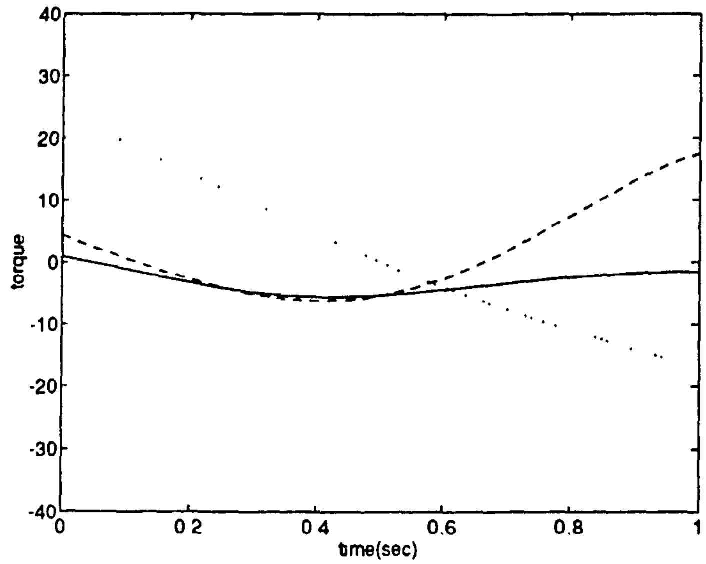


Fig. 4.8 Joint torque resulting from RRODM without DCI.

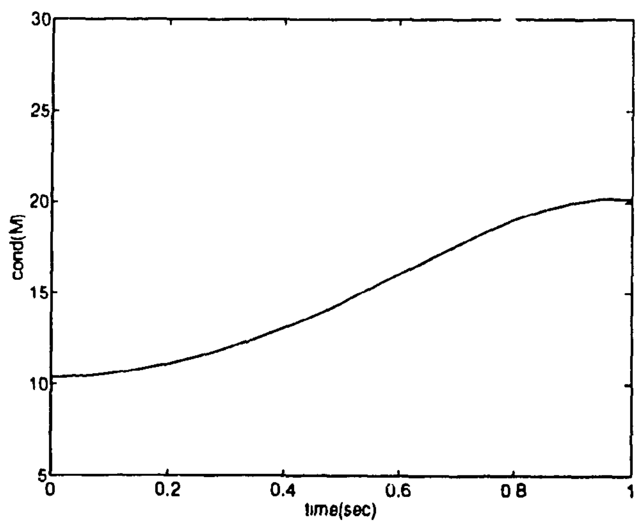


Fig 4.9 Variation of the condition number of M resulting from RRODM with DCI.

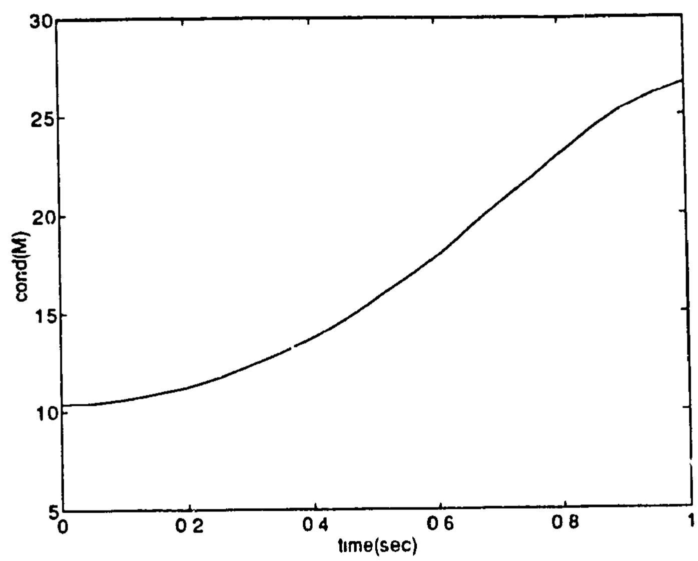


Fig. 4.10 Variation of the condition number of *M* resulting from RRODM without DCI.

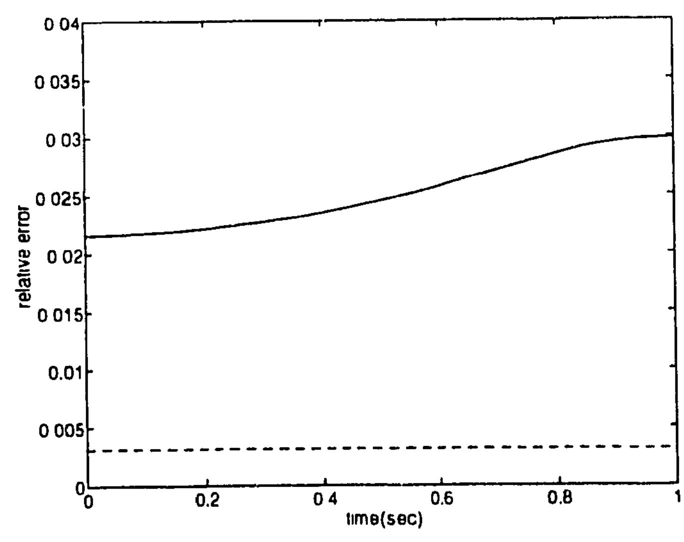


Fig. 4.11 Relative error resulting from RRODM with DCI.

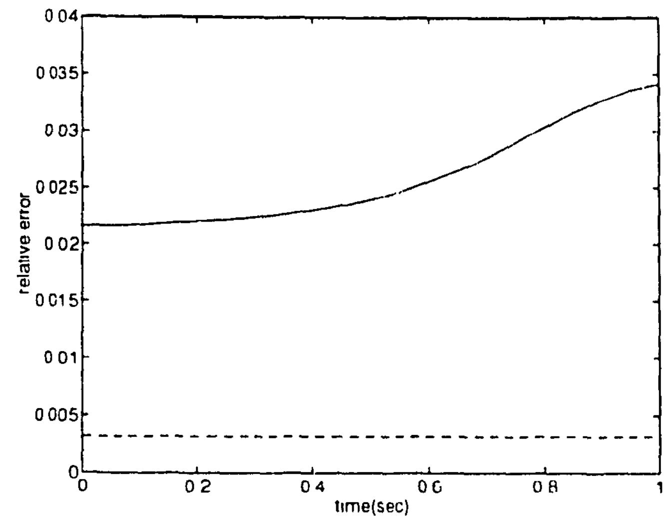


Fig. 4.12 Relative error resulting from RRODM without DCI.

References

[1] J. E. Bobrow, S. Dubowsky and J. S. Gibson, "Time-Optimal Control of Robotic Manipulators Along Specified Paths," *Int. J. of Robotics Research*, Vol. 4, No. 3, 1985, pp. 3-17.

[2] B. K. Kim and K. G. Shin, "Minimum-Time Path Planning for Robot Arms and Their Dynamics." *IEEE Transaction on System, Man and Cybernetic*, SMC-15, 1985, pp. 213-223.

[3] K. G. Shin and N. D. McKay, "Minimum-Time Control of a Robotic Manipulator with Actuator Constraints," *IEEE Transaction on Automatic Control*, AC-30, 1985, pp. 531-542.

[4] V. Yen and M. Nagurka, "Optimal Trajectory Planning of Robotic Manipulators via Quasi-Linearization and State Parameterization," *Proc. IEEE Int. Conf. on Robotics*

and Automation, 1989, pp.1116-1121.

[5] Y.-R. Hu and A. A. Goldenberg, "Dynamic Control of Multiple Coordinated Redundant Manipulators with Torque Optimization," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 1000-1005.

[6] S. Gupta and J. Y. S. Luh, "Closed-Loop Control of Manipulators with Redundant Joints Using the Hamilton-Jacobi-Bellman Equation," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1991, pp. 472-477.

[7] O. Ma and J. Angeles, "The Concept of Dynamic Isotropy and its Applications to Inverse Kinematics and Trajectory Planning," *IEEE Int. Conf. Robotics and automation*, 1990, pp.481-486.

[8] H. Asada, "A Geometrical Representation of Manipulator Dynamics and Its Application to Arm Design," *J. Dyn. Syst. Meas. and Control*, Vol. 105, Sep. 1983, pp. 131-135.

[9] T. Yoshikawa, "Manipulability of Robotic Mechanisms," *Int. J. of Robotics Research*, Vol. 4, No. 2, Summer, 1985, pp. 3-9.

[10] T. Yoshikawa, "Dynamic Manipulability of Robot Manipulators," *Journal of Robotic Systems*, Vol. 2, No. 1, 1985, pp.113-124.

[11] F. L. Lewis, *Optimal Control*, John wiley and Sons, New York, 1986.

[12] A. E. Bryson, Jr. and Y.-C. Ho, *Applied Optimal Control*, Hemishere, New York, 1975.

[13] D. A. Piere, *Optimization Theory with Applications*, John Wiley & Sons, Inc. New York, 1969.

[14] IMSL INC, *MATH/LIBRARY User's Manual*, IMSL INC, Houston, Texas, August, 1989.

[15] V. D. Tourassis and C. P. Neuman, "Properties and Structure of Dynamic Robot Models for Control Engineering Applications," *Mechanism and Machine Theory*, Vol. 20. No. 1, 1985, pp. 27 - 40.

[16] C. A. Balafoutis, P. Misra and R. V. Patel, "Recursive Evaluation of Linearized Dynamic Robot Models," *IEEE J. Robotics and Automation*, Vol. RA-2, 1986, pp. 146 - 155.

Chapter 5 Redundancy Resolution for REDIESTRO based on Optimization of Kinematic and Dynamic Conditioning

5.1 Introduction

The RROKM and RRODM schemes presented in the previous chapters have the same goal - that of improving accuracy. The RROKM scheme considers the conditioning of the Jacobian matrix of a manipulator, while the RRODM scheme is concerned with the conditioning of the mass matrix. In this chapter, both schemes will be applied to an isotropic redundant manipulator with seven revolute joints. First, we describe the isotropic redundant manipulator. Then, the applications of both the RROKM and RRODM schemes will be presented via computer simulations using a kinematic model in the case of RROKM and a dynamic model in the case of RRODM. In the RRODM scheme, the dynamic equations of the manipulator are generated using the Newton-Euler dynamics algorithm.

5.2 The Isotropic Manipulator

The isotropic manipulator called REDIESTRO (REDundant Isotropically Enhanced Seven-Turning-pair RObot) was designed and assembled [1] at the McGill Center for Intelligent Machines (CIM). The main feature of the manipulator is kinematic isotropy, i.e., at certain points in its workspace the condition number of the Jacobian matrix of the manipulator is equal to one. Therefore, for control schemes such as rate control and force feedback control which involve transformations using the Jacobian matrix, REDIESTRO can be made to perform operations with high accuracy in the neighborhood of the isotropic configurations.

REDIESTRO is a seven-axes, revolute-coupled redundant manipulator. The Hartenberg-Denavit parameters of REDIESTRO in the notation used in [2] are given in Table 5.1.

Table 5.1 Hartenberg-Denavit Parameters of REDIESTRO

Link i	a_i(mm)	b_i(mm)	α_i (degree)	q_i(degree)
1	0	0	-58.31	q_1
2	231.13	-22.91	-20.0289	q_2
3	0	36.93	105.26	q_3
4	398.84	0	60.91	q_4
5	0	-471.59	59.88	q_5
6	135.59	578.21	-75.47	q_6
7	234.44	-145.05	0	q_7

Home configuration $q_{1\text{-}7}$(degree):

$$q_{1-7} = \begin{bmatrix} 0 & -11.01 & 91.94 & 113.93 & -2.26 & 150.25 & 63.76 \end{bmatrix}^T$$

Characteristic Length = 220.6505 mm

Maximum Reach = 1488.0 mm

In table 5.1, a_{i-1}, b_i, α_{i-1}, q_i are the link length, joint offset, twist angle and joint angle, respectively.

5.3 Application of the RROKM Scheme

The Jacobian matrix of REDIESTRO is a 6×7 matrix that maps the 7×1 vector of joint angular velocities to the 6×1 vector of Cartesian (end-effector position and orientation) velocities. It can be computed using the expression [4]:

$$J = \begin{bmatrix} z_1 \times p_1 & z_2 \times p_2 & \dots & z_7 \times p_7 \\ z_1 & z_2 & \dots & z_7 \end{bmatrix} \tag{5.1}$$

where z_i is the unit vector along the ith revolute joint and p_i is the position vector from the origin of link frame $\{i\}$ to the origin of the end-effector frame $\{7\}$.

The desired trajectory is chosen such that the end-effector moves along a straight line with a fixed end-effector orientation:

$$x_d(t) = \begin{bmatrix} p_d(t) & \omega_d(t) \end{bmatrix}^T, \tag{5.2}$$

where $\omega_d(t)$ is a 3×1 end-effector orientation vector with a fixed value equal to its initial value:

$$\omega_d(t) = \omega_d(t_0) = \begin{bmatrix} \omega_d^{\gamma}(t_0) \\ \omega_d^{\beta}(t_0) \\ \omega_d^{\alpha}(t_0) \end{bmatrix}, \tag{5.3}$$

and $p_d(t)$ is a 3×1 desired end-effector position vector which is given by

$$p_d(t) = \begin{bmatrix} p_d^x(t) \\ p_d^y(t) \\ p_d^z(t) \end{bmatrix} = \begin{bmatrix} p_d^x(t_0) + [p_d^x(t_f) - p_d^x(t_0)] \left[\frac{t}{t_f} - \frac{1}{2\pi} \sin(2\pi \frac{t}{t_f}) \right] \\ p_d^y(t_0) + [p_d^y(t_f) - p_d^y(t_0)] \left[\frac{t}{t_f} - \frac{1}{2\pi} \sin(2\pi \frac{t}{t_f}) \right] \\ p_d^z(t_0) + [p_d^z(t_f) - p_d^z(t_0)] \left[\frac{t}{t_f} - \frac{1}{2\pi} \sin(2\pi \frac{t}{t_f}) \right] \end{bmatrix}, \qquad (5.4)$$

where p_d^x, p_d^y and p_d^z represent the X, Y and Z components of the position vector respectively and t_o (t_o=0) and t_f denote the initial and final value of time. The actual end-effector trajectory is computed using forward kinematics of REDIESTRO based on its Hartenberg-Denavit parameters. The initial home configuration of the manipulator is an isotropic one given by

$$q = \left[0.0° \; -11.01° \; 91.94° \; 113.93° \; -2.26° \; 150.25° \; 63.76° \right]^T . \qquad (5.5)$$

The end-effector position p_e which corresponds to this configuration is

$$p_e = p_e(t_0) = \left[0.0618 \; 0.2314 \; 0.1747 \right]^T . \qquad (5.6)$$

The end-effector of the manipulator is required to move from the point $p_d(t_0) = p_e(t_0)$ to a point

$$p_d(t_f) = \left[0.5618 \; 0.7314 \; 0.1747 \right]^T \qquad (5.7)$$

along a straight line described by equation (5.4) in one second. The joint angle trajectories are given in Fig. 5.1. The condition number of the Jacobian matrix is plotted in Fig. 5.2, and the KK isotropy measure is shown in Fig. 5.3. Fig. 5.2 shows that the condition number of the Jacobian matrix has the minimum value (=1) in the isotropic (home) configuration. In Fig. 5.3 this corresponds to the maximum value (=1) for the KK isotropy measure.

In addition, while the motions obtained by the pseudoinverse control are usually noncyclic [5][6][7], the RROKM scheme can generate cyclic motions. Fig. 5.4 shows the joint trajectories in a cyclic motion in which the robot end-effector tracks a circular trajectory of radius $r = 0.25$m with the center at

$$\left[c_x\ c_y\ c_z\right]^T = \left[0.3118\ 0.2314\ 0.1747\right]^T. \tag{5.8}$$

The desired circular end-effector position trajectory is given by

$$p_d(t) = \begin{bmatrix} c_x + r\cos(0.5\pi t + \pi) \\ c_y + r\sin(0.5\pi t) \\ c_z \end{bmatrix}. \tag{5.9}$$

The initial configuration of the manipulator is the isotropic home configuration. The variations in the condition number of the Jacobian and the KK isotropy measure are shown in Fig. 5.5 and Fig. 5.6, respectively.

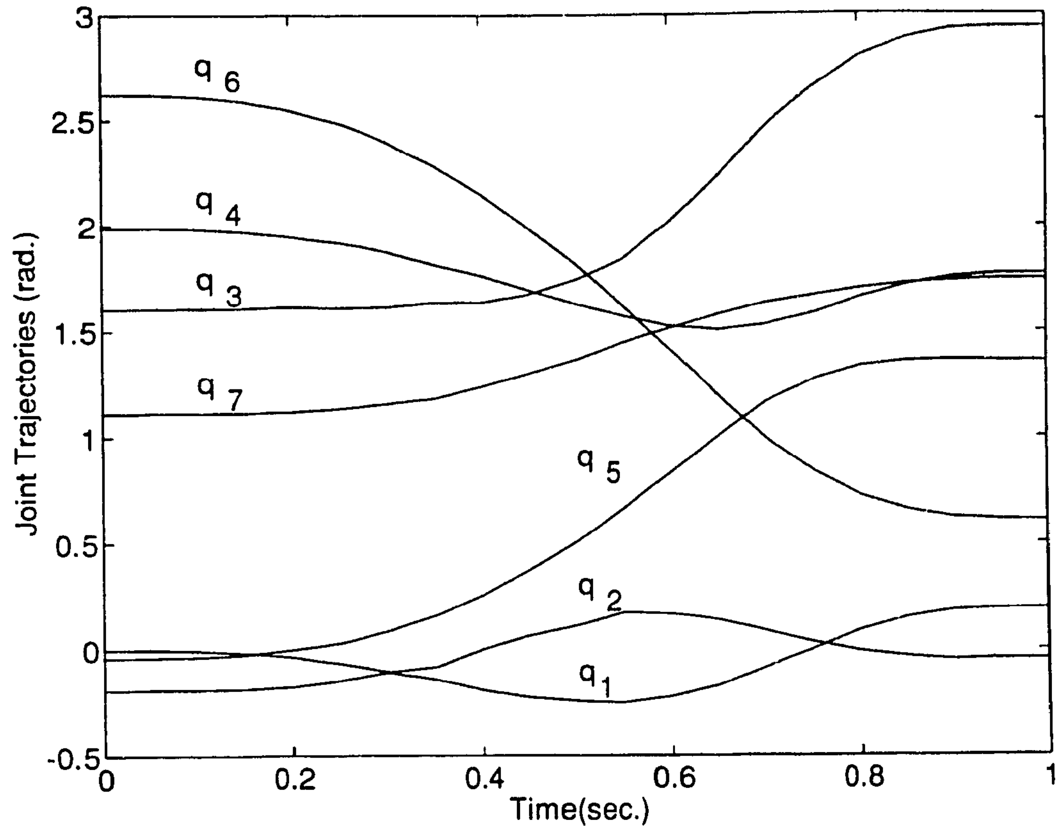


Fig. 5.1 Joint trajectories of REDIESTRO resulting from RROKM.

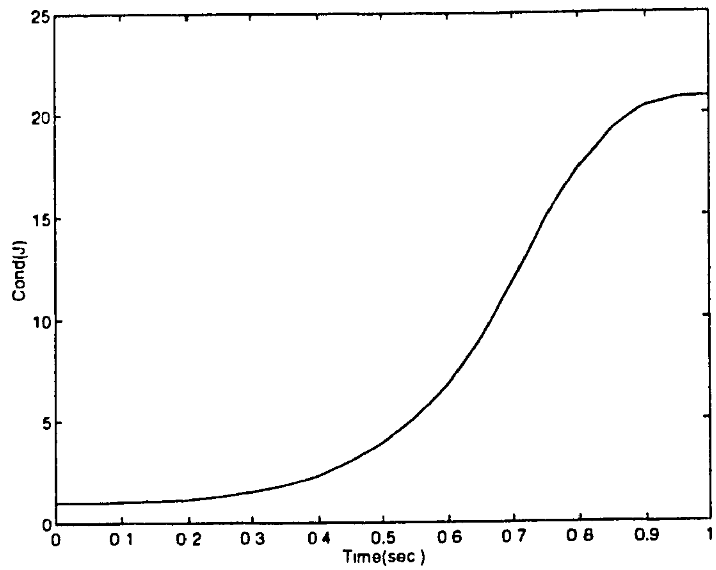


Fig. 5.2 Variation in the condition number of the Jacobian matrix of REDIESTRO resulting from RROKM.

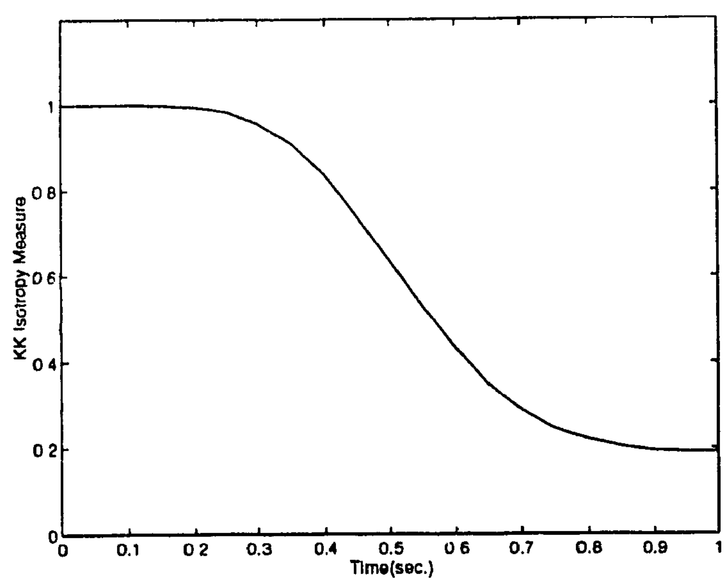


Fig. 5.3 Variation in the KK isotropy measure for REDIESTRO resulting from RROKM.

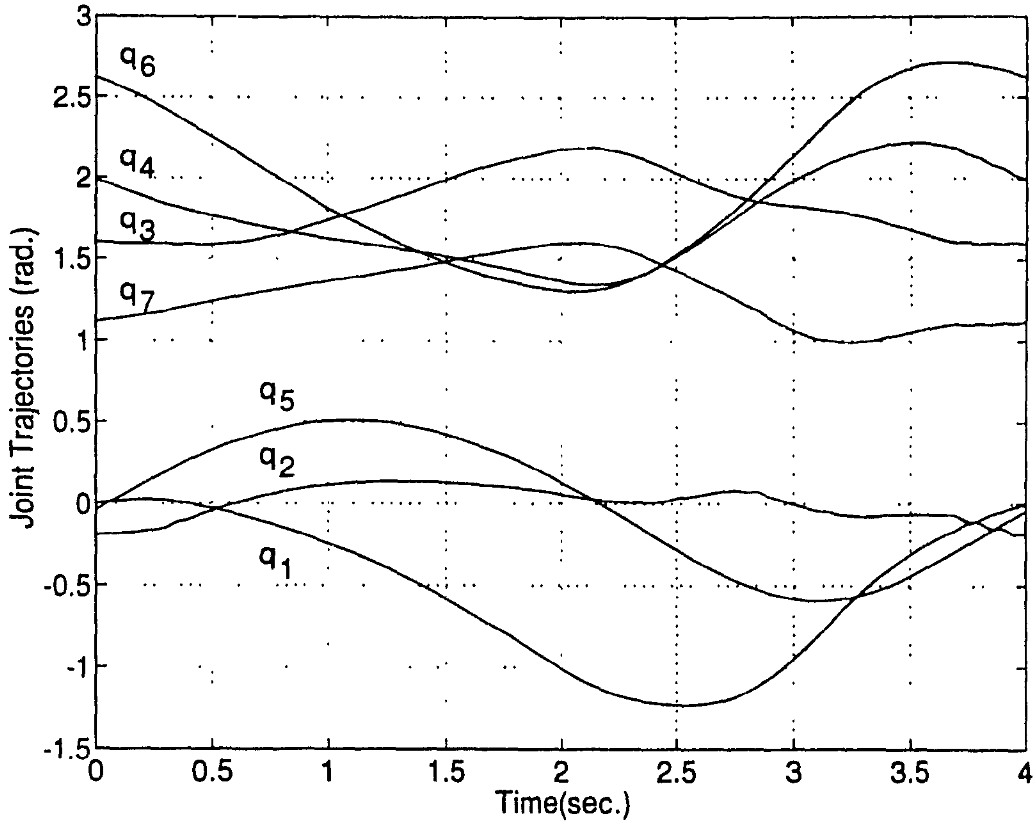


Fig. 5.4 Joint trajectories for cyclic motion of REDIESTRO resulting from RROKM.

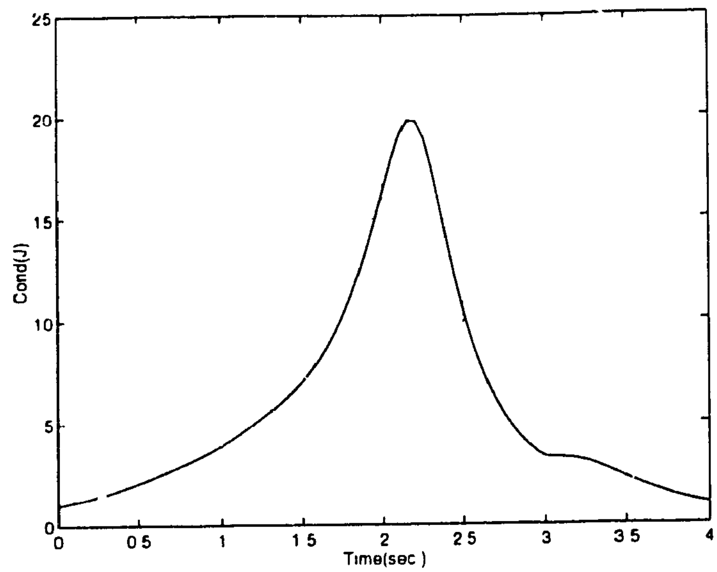


Fig. 5.5 Variation in the condition number of the Jacobian matrix of REDIESTRO during cyclic motion resulting from RROKM.

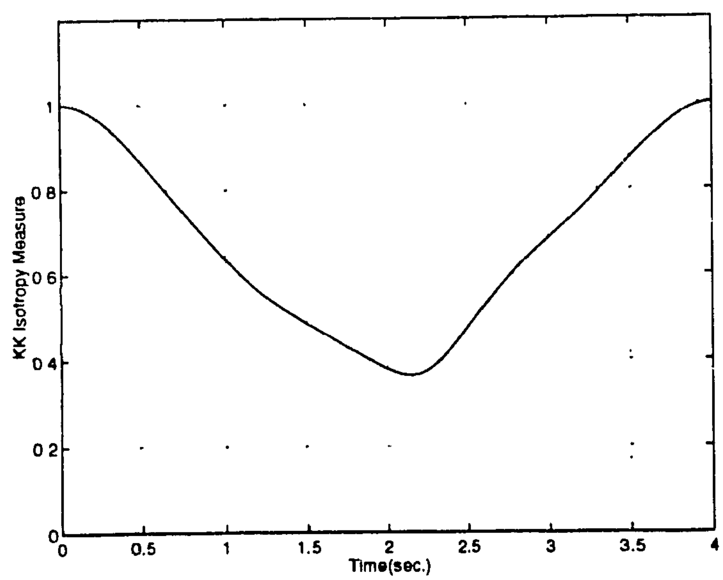


Fig. 5.6 Variation in the KK isotropy measure for REDIESTRO during cyclic motion resulting from RROKM.

5.4 Application of the RRODM Scheme

The dynamics of REDIESTRO can be expressed in the matrix form:

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q), \tag{5.10}$$

where, q, $\dot{q}$, $\ddot{q}$, g and τ are 7×1 vectors, and $M(q)$ and $C(q, \dot{q})$ are 7×7 matrices. Equation (5.10) was computed using the Newton-Euler algorithm, e.g. see [3]. This algorithm enables us to compute the joint torque τ corresponding to a given set of values ($(q, \dot{q}, \ddot{q})$) of the position, velocity, and acceleration of each joint. The inertia tensors ${}^{C_i}I_i$, link masses m_i, and the location of the center of gravity ${}^{i}P_{C_i}$ of each link of REDIESTRO are given in Table 5.2, Table 5.3 and Table 5.4 respectively. These parameters are given in the notation of [2].

Table 5.2 Inertia Tensors (kg m^2)

${}^cI$	Link 1	Link 2	Link 3	Link 4	Link 5	Link 6	Link 7
${}^cI_{xx}$	0.89926	0.02573	1.6620	0.09297	0.8284	0.6541	2.4e-5
${}^cI_{yy}$	0.31342	0.13223	0.7860	0.8881	0.7019	0.6714	1.136e-3
${}^cI_{zz}$	0.62745	0.11099	0.9387	0.8753	0.1317	0.0374	1.135e-3
${}^cI_{xy}$	-2.7e-5	-0.0045	0.0001	-0.1203	0.00009	-0.00839	0.0
${}^cI_{yz}$	0.3698	0.0012	0.1221	-0.0204	0.26852	0.04574	0.0
${}^cI_{xz}$	-1.2e-5	-0.0404	0.0003	0.1411	0.00016	-0.12596	0.0

Table 5.3 Link Masses (kg)

	Link 1	Link 2	Link 3	Link 4	Link 5	Link 6	Link 7
m_i	17.313	5.58	28.586	7.390	5.987	2.557	0.2

Table 5.4 Centers of Gravity (m)

P_c	Link 1	Link 2	Link 3	Link 4	Link 5	Link 6	Link 7
x	0.00048	0.1155	-0.0011	0.3071	0.0	-0.0919	0.06345
y	-0.1607	-0.0036	-0.1176	-0.0408	-0.1326	0.03434	0.0
z	-0.1186	-0.0618	-0.1170	0.0699	-0.3209	0.49	-0.0034

The Newton-Euler algorithm computes the required joint torque for given joint position, velocity, and acceleration. However, for the RRODM scheme, we need to compute the matrices $M(q)$ and $C(q, \dot{q})$, and the vector $g(q)$. Note that the mass matrix $M(q)$ is a function of q only. Therefore, to find $M(q)$, we let $\dot{q} = 0$ and the gravity force = 0 (this results in $g(q) = 0$). In this case, the dynamic equation (5.10) reduces to:

$$\tau = M(q)\ddot{q}. \tag{5.11}$$

Equation (5.11) gives a linear relation between the joint torque and the joint acceleration. Now, if we apply as input a joint acceleration vector $\ddot{q} = v_i$, where v_i denotes the ith column of the 7×7 identity matrix, we can compute the ith column of $M(q)$ using the Newton-Euler dynamics algorithm. Also, let $G = C(q, \dot{q})\dot{q} + g(q)$, we can rewrite the dynamic equation (5.10) as:

$$\tau = M(q)\ddot{q} + G. \tag{5.12}$$

Then setting $\ddot{q} = 0$, we get

$$\tau = G. \tag{5.13}$$

Hence the vector G can also be obtained using the Newton-Euler dynamics algorithm. Note that the mass matrix and vector G are computed discretely in time. If the sampling period is small, the matrix $M(q)$ can be considered constant between two successive sampling instants. Equation (4.53) can be reduced to

$$\dot{y}_1 = y_2,$$

$$\dot{y}_2 = -M^{-1}G - M^{-1}R^{-1}(M^{-1})^T\lambda_2,$$

$$\dot{\lambda}_1 = -\left[\frac{\partial\eta}{\partial y_1}\right]^T Q\eta - J^T Se + \left[\frac{\partial}{\partial y_1}[M^{-1}G]\right]^T \lambda_2,$$

$$\dot{\lambda}_2 = -\lambda_1 + \left[\frac{\partial}{\partial y_2}[M^{-1}G]\right]^T \lambda_2. \tag{5.14}$$

The mass matrix $M(q)$ is updated at each sampling instant.

The desired end-effector trajectory is given by equation (5.2). The end-effector is required to move from

$$p_d(t_0) = \left[0.0618\ 0.2314\ 0.1747\right]^T \tag{5.15}$$

to

$$p_d(t_f) = \left[0.1618\ 0.3314\ 0.4747\right]^T \tag{5.16}$$

in one second. The weighting matrices are chosen as follows: W_1 = diag(1500), W_2 = diag(1500), R^{-1} = diag(50) and S = diag(2000). The sampling period is 1/30 second. The simulation results are shown on Figures 5.8 - 5.12. In Fig. 5.12, the solid line denotes the condition number of $M(q)$ with Q = diag(0.01), while the dashed line represents the condition number of $M(q)$ with $Q = 0$, i.e., without considering DCI. Fig. 5.12 shows that the conditioning $M(q)$ is better when DCI is included in the RRODM scheme.

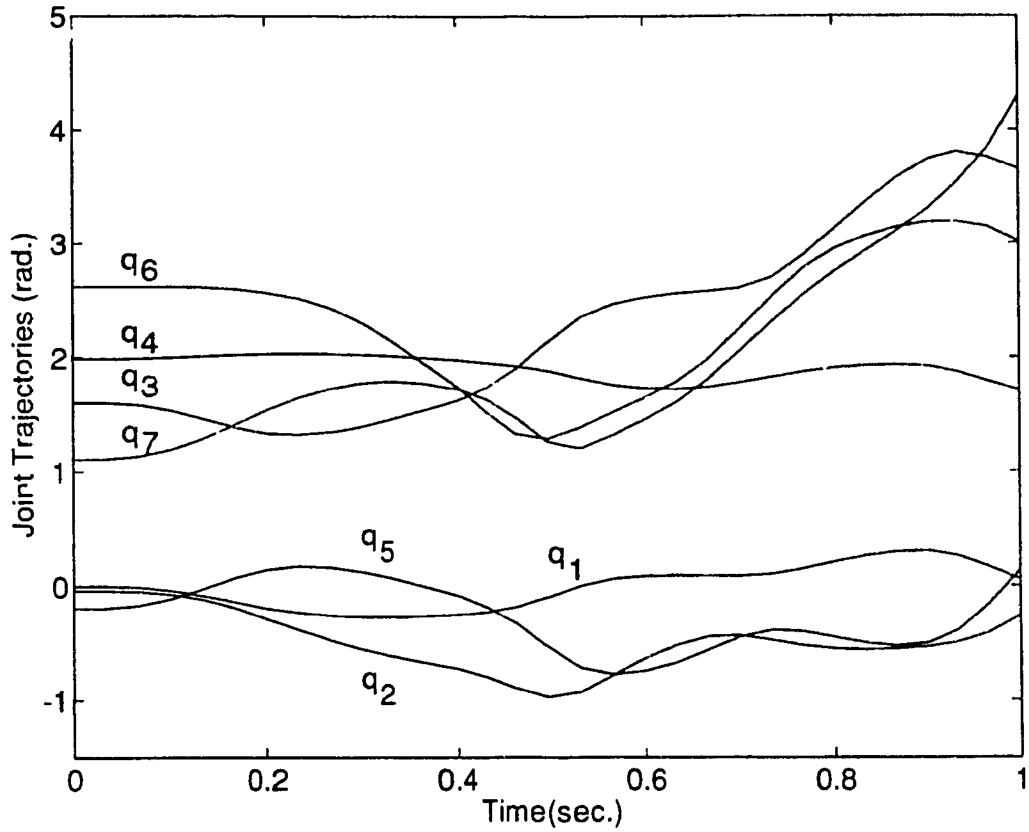


Fig. 5.7 Joint trajectories for REDIESTRO resulting from RRODM.

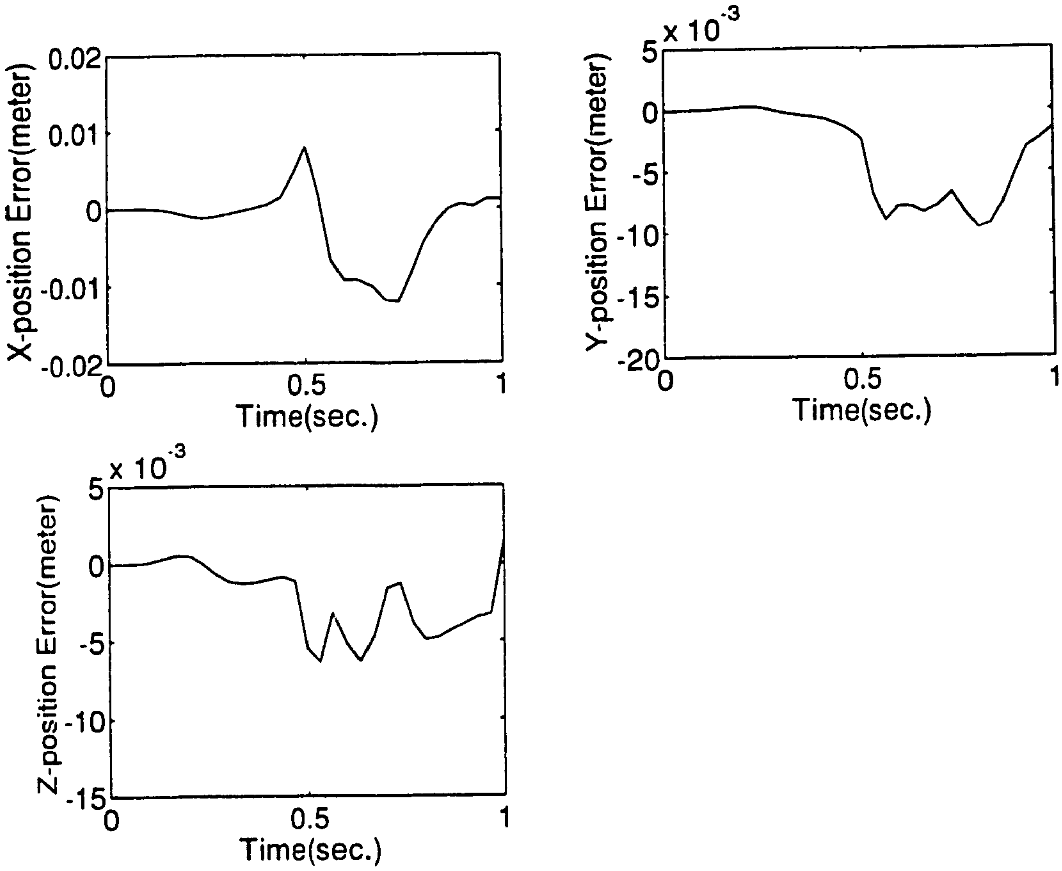


Fig. 5.8 Tracking error resulting from RRODM.

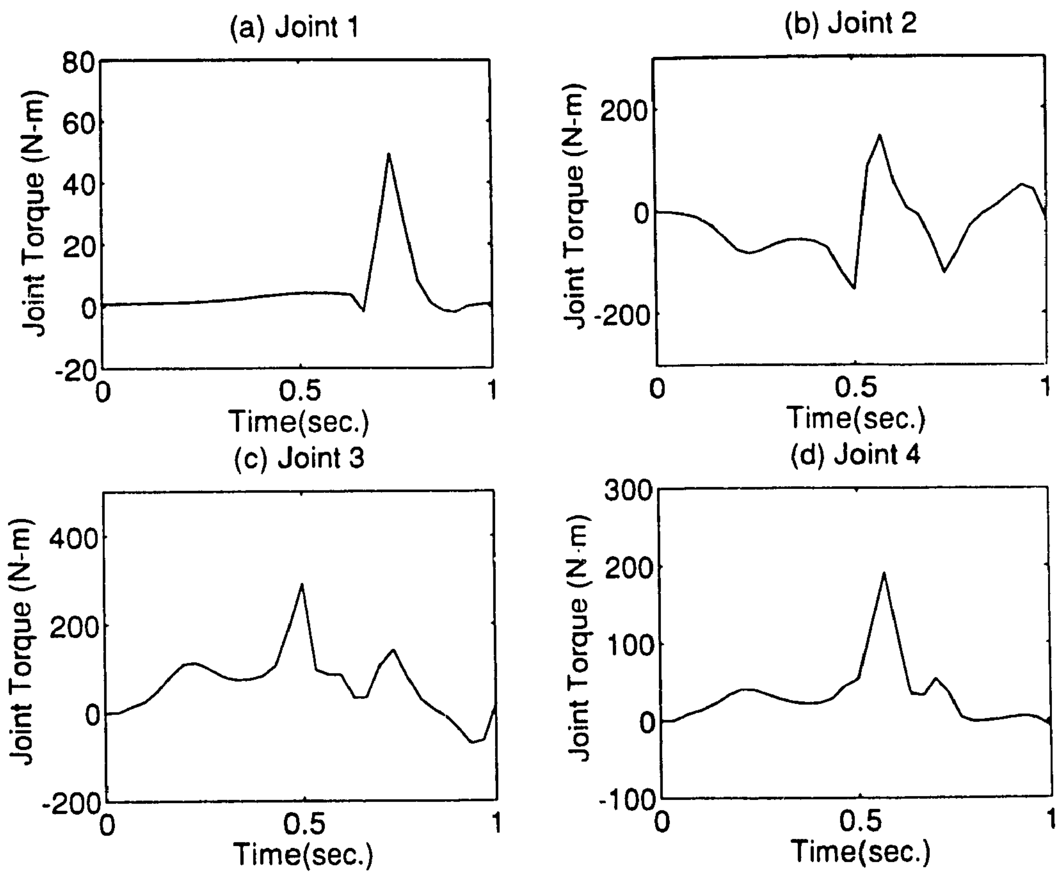


Fig. 5.9 (a-d) Joint torque history resulting from RRODM.

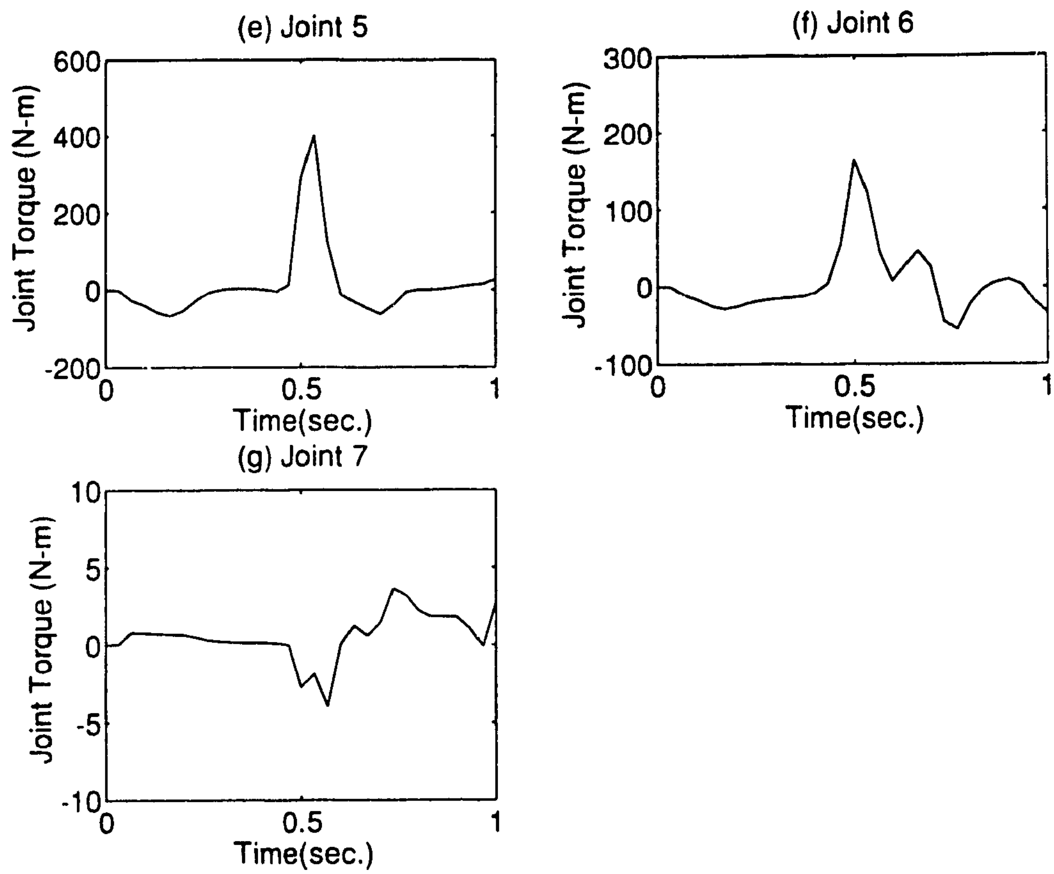


Fig. 5.10 (e-g) Joint torque history resulting from RRODM.

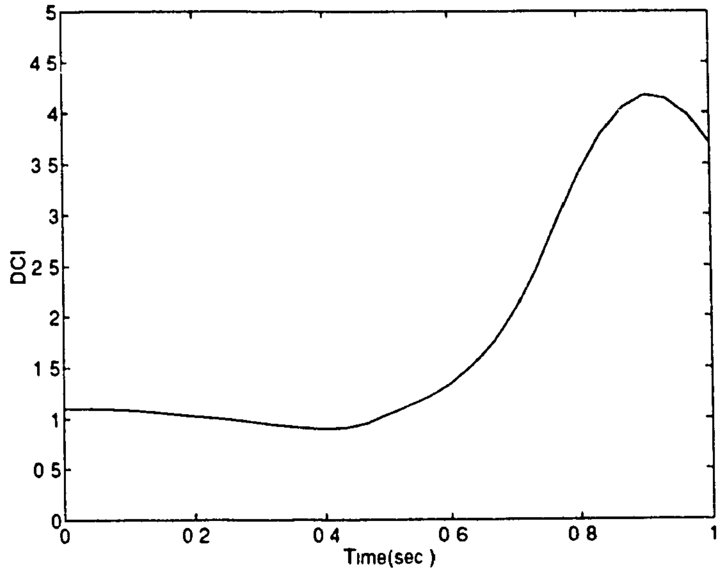


Fig. 5.10 Variation in DCI for REDIESTRO resulting from RRODM.

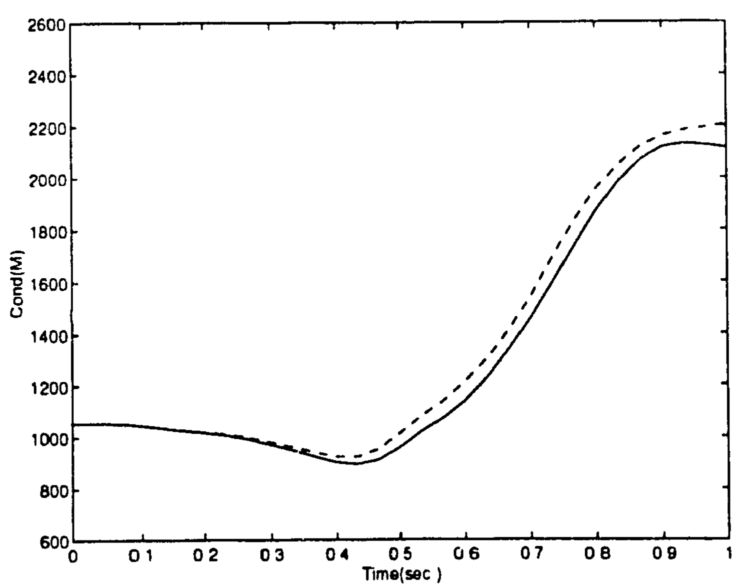


Fig. 5.11 Variation in the Condition number of *M(q)* for REDIESTRO resulting from RRODM with DCI (—) and without DCI (----).

References

[1] J. Angeles, F. Ranjbaran and R. V. Patel. "On the Design of the Kinematic Structure of Seven-Axis Redundant Manipulators for Maximum Conditioning," *Proc. IEEE Int. Conf. Robotics and Automation*, 1992, pp. 494-499.

[2] R. P. Paul, *Robot manipulators: mathematics, programming, and control*, MIT Press, Cambridge, MA, 1981.

[3] K. S. Fu, R. C. Gonzalez and C. S. G. Lee, *Robotics: control, sensing, vision, and intelligence*, McGraw-Hill, New York, 1987.

[4] D. E. Whitney, "The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators," *ASME J. Dynam. Syst., Meas., Contr.*, Vol. 94, No. 14, 1972, pp. 303-309.

[5] C. A. Klein and C. H. Huang, "Review of Pseudoinverse Control for Use with Kinematically Redundant Manipulators," *IEEE Trans. Systems, Man, and Cybernetics*, SMC-13(2), 1983, pp. 245-250.

[6] D. R. Baker and C. W. Wampler, II, "Some Facts Concerning the Inverse Kinematics of Redundant Manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1987, pp. 604-609.

[7] C. W. Wampler, "Inverse Kinematic Functions for Redundant Manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1987, pp. 610-617.

[8] O. Ma and J. Angeles, "The Concept of Dynamic Isotropy and its Applications to Inverse Kinematics and Trajectory Planning," *Proc. IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 481-486.

CHAPTER 6 CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

The main contributions of this thesis are two new redundancy resolution schemes based on optimization of kinematic and dynamic conditioning: redundancy resolution based on optimization of kinematic measures (RROKM) and redundancy resolution based on optimization of dynamic measures (RRODM). With these approaches we can see how conditioning measures affect errors. Both approaches have been applied (via simulation) to REDIESTRO, a seven joint kinematically isotropic manipulator.

The RROKM uses local constrained optimization techniques, i.e., the solution is optimal at each sample point on the desired trajectory but not necessary optimal over the whole trajectory. This results in an efficient computational scheme. When the Jacobian matrix is used as a transformation matrix, the condition number of the Jacobian matrix affects the propagation of errors such as numerical rounding errors and measurement errors. In the RROKM approach, a conditioning measure is optimized while the end-effector tracks a desired trajectory. This ensures that the condition number remains as small as possible along the trajectory. Hence error magnification resulting from the conditioning of the Jacobian is reduced. In addition, since a singularity is characterized by an infinite condition number of the Jacobian matrix, constraining the condition number ensures that the manipulator is kept as far away from singularities as possible. This in turn reduces sensitivity with regard to velocity and torque computations, and provides better control of the manipulator.

The RRODM approach is a global optimization approach based on optimal control theory. An integral type of performance index is chosen so that the resulting solution is optimal over the whole trajectory. The optimal control strategy results in a two-point boundary-value problem (TPBVP) which is solved numerically using a complete nonlinear dynamic model of the manipulator. In general, extensive computation is required to solve the TPBVP. The RRODM optimizes a dynamic conditioning index (DCI) while the end-effector tracks a desired trajectory. This ensures that the DCI is as close to isotropy (optimum conditioning of the mass matrix) as possible while the end-effector tracks the desired trajectory. This improves the performance of dynamic controllers that involve transformations using the mass matrix. The RRODM approach requires significantly more computations than the RROKM approach. The performance of the scheme is characterized by several weighting matrices which are selected using several simulation runs. The choice of these weighting matrices allows a designer to achieve a balance between dynamic conditioning, tracking errors, and magnitude of the required control torque.

Simulation results for a 3 DOF revolute joint planar manipulator and REDIESTRO, a seven DOF revolute joint kinematically isotropic manipulator, are given for both RROKM and RRODM schemes.

6.2 Suggestions for Future Work

Manipulator designs considering both kinematic and dynamic dexterity measures to utilize redundancy have been investigated for several years. However, manipulator control that takes into account various conditioning measures as they affect transformation accuracy and system sensitivity has not received much attention. This thesis has provided an approach that shows the feasibility of developing such control schemes. A major issue is the amount of computation required by the proposed schemes. Future work is required to develop strategies (perhaps suboptimal ones) to reduce the amount of computation.

In Chapter 3, we note that the computation of the KK isotropy measure is more efficient than that of the condition number. This was shown by comparing the expressions for calculating the KK isotropy measure and the condition number in the case of a three-link planar manipulator. But comparisons between computation of the KK isotropy measure and that of the condition number for general manipulators would be useful in manipulator design as well as control. The RROKM scheme in Chapter 3 is a local optimization scheme. For completeness, it would be useful to develop a global scheme where the kinematic measure is optimized over the whole trajectory.

As discussed in Chapter 4, the performance of optimal control is characterized by weighting matrices. One can achieve a trade-off between tracking error, control input energy, and dynamic conditioning by selecting appropriate values of the weighting matrices. The choice of these matrices is not particularly straightforward. In this thesis they were selected based on computer simulations of a few sample runs. Therefore, it is necessary to develop more systematic procedures for selecting these weighting matrices. Finally, the solution of the TPBVP is very computation intensive so that online (real-time) solution of RRODM is not possible. However, it may be possible to generate a class of trajectories for which the solutions of the TPBVP may be considerably simplified.

APPENDIX SYSTEM SENSITIVITY AND ERROR ANALYSIS

I Sensitivity and Condition Number of a Square Matrix

Let A be a square nonsingular matrix, so that A^{-1} exists and is unique. Consider the linear system

$$Ax = b. \tag{A1}$$

The exact (unique) solution of (A1) is given by

$$x = A^{-1}b. \tag{A2}$$

Suppose that b in (A1) is changed to δb. This small change may come from experimental data, rounding errors, noise or perturbations. Then, the solution of the system is changed from x to $x + \delta x$:

$$A\,(x+\delta x) = b+\delta b. \tag{A3}$$

Equation (A3) implies that

$$A\delta x = \delta b \tag{A4}$$

and hence

$$\delta x = A^{-1}\delta b. \tag{A5}$$

Therefore, an upper bound on δx is given by

$$\|\delta x\| \le \|A^{-1}\|\|\delta b\|. \tag{A6}$$

where $\|\bullet\|$ denotes an appropriate vector and matrix norm, such as the 2-norm [1]. Note that the relation $Ax = b$ implies that

$$\|b\| \le \|A\|\|x\|. \tag{A7}$$

From (A6) and (A7) we obtain the following important result concerning relative errors:

$$\frac{\|\delta x\|}{\|x\|} \le Cond(A)\frac{\|\delta b\|}{\|b\|}, \tag{A8}$$

where the condition number of A is defined as

$$cond(A) = \|A^{-1}\|\|A\|. \tag{A9}$$

Equation (A8) shows that the relative error in the solution $\frac{\|\delta x\|}{\|x\|}$ can be a magnification by a factor *cond(A)* of the relative perturbation $\frac{\|\delta b\|}{\|b\|}$. When A is close to being singular, i.e.

the $cond(A)$ is very large, the system may be very sensitive in which a small change in $\frac{\|\delta b\|}{\|b\|}$ causes a large change in $\frac{\|\delta x\|}{\|x\|}$. Note that $cond(A) \geq 1$, where the equality holds for a matrix with the best possible conditioning such as an orthogonal matrix or a matrix of the form αI where α ia a scalar and I is the identity matrix.

II Sensitivity and Condition Number of a Non-Square Matrix

When A is a general $m \times n$ matrix of full rank m, the solution of the linear system $Ax = b$ can be found by the pseudoinverse method:

$$x = A^{\dagger} b + (I - A^{\dagger} A) z, \tag{A10}$$

where $A^{\dagger}$ denotes the pseudoinverse and

$$A^{\dagger} = A^T (AA^T)^{-1}. \tag{A11}$$

Let x^* be the minimum-norm solution of the system $Ax = b$. That is,

$$x^* = A^{\dagger} b \tag{A12}$$

or

$$Ax^* = b \tag{A13}$$

and note that

$$\|b\| \le \|A\| \|x^*\| . \tag{A14}$$

Again suppose that b is perturbed to $b + \delta b$, and the corresponding solution is changed to $x^* + \delta x$, i.e.

$$x^* + \delta x = A^{\dagger} (b + \delta b) . \tag{A15}$$

This implies that

$$\delta x = A^{\dagger} \delta b \tag{A16}$$

and

$$\|\delta x\| \le \left\|A^{\dagger}\right\| \|\delta b\| . \tag{A17}$$

Combining (A14) and (A17) and rearranging, we obtain

$$\frac{\|\delta x\|}{\|x^*\|} \le \left\|A^{\dagger}\right\| \|A\| \frac{\|\delta b\|}{\|b\|} = cond(A) \frac{\|\delta b\|}{\|b\|} . \tag{A18}$$

This implies that the relative error $\frac{\|\delta x\|}{\|x^*\|}$ can be as large as the relative perturbation $\frac{\|\delta b\|}{\|b\|}$ multiplied by $cond(A)$. Thus, the condition number of a non-square matrix plays the same role as that of a square matrix [2].

References

[1] G. H. Golub and C. F. Van Loan, *Matrix computations*, 2nd ed., Johns Hopkins University Press, Baltimore, 1989.

[2] P. E. Gill, W. Murray and M. H. Wright, *Numerical Linear Algebra and Optimization*, Vol. 2, Addison-Wesley, Reading, MA, 1991.