A Constrained Optimization Approach for Path Planning of Redundant Robot Manipulators*

Chih-Jer LIN* and Cha’o-Kuang CHEN**

A redundant manipulator can achieve multiple tasks using the degree of redundancy. In this paper, a redundancy resolution problem is formulated with multiple criteria into a local equality and inequality constrained optimization problem. To achieve multiple tasks, the task with the highest priority can be performed by optimizing a corresponding objective function. The forward kinematic relation and the other tasks with higher priority are performed by satisfying an equality constraint and a set of inequality constraints, respectively. The solution of the path planning is then obtained using an optimization procedure. Different to the conventional approach, the inverse of the Jacobian matrix is not used in the proposed method. The proposed method also provides a unified approach to solve motion planning problems with multiple tasks.

Key Words: Forward Kinematics, Redundant Robots, Permissible Zone, Obstacle Avoidance, Singularity Avoidance, Perturbation Method.

1. Introduction

The motion planning of robot manipulators is one of the challenging problems encountered in the field of robotics. This problem can be treated either in the joint space or in the task space. The motion planning in the task space for a desired end-effector trajectory is usually considered as an inverse kinematics problem. The closed form solution to this problem is only available for certain types of manipulators and it is difficult to obtain. This is particularly true in the case when a robot has extra degree of freedom. A redundant manipulator is a manipulator has more degree-of-freedom (DOF) than those necessary to perform a given task. Consider a redundant manipulator with $n$ degrees of freedom, which manipulates in $m$-dimensional task space ($n > m$), where $r \in R^m$, $\theta \in R^n$ are the task space and the joint space variables, respectively. Then the forward kinematic function $f$ of the robot can be defined as $f: R^n \mapsto R^m$ or

$$ r = f(\theta) \quad (1) $$

Alternatively, Eq. (1) can be represented in its derivative form as:

$$ \dot{r} = J(\theta) \dot{\theta} \quad (2) $$

where $J = \frac{\partial f}{\partial \theta}$ is the $m \times n$ Jacobian matrix.

Therefore, the degree of redundancy (DOR) is represented as the difference $n - m$. The main advantage of a redundant manipulator is that it can perform various additional tasks using the DOR as well as its principle task governed by Eq. (1). These additional tasks include obstacles avoidance$^{5-6}$, singularity avoidance$^{6-10}$, structure limitations$^{11}$, torque minimization$^{12}$, dexterity measures$^{13}$, task priority control$^{13}$, and energy minimization$^{15}$, etc.

The inverse kinematics problem with mechanical redundancies has historically been solved through the Moore-Penrose pseudo-inverse methods, also called the generalized inverse method. This method is first introduced to the robot control problem by Whitney$^{18}$. The fundamentals of this approach are as follows. Consider a redundant manipulator with $n$ DOF, which manipulates in $m$-dimensional task space ($n > m$). The forward kinematic equation is
represented as Eq. (2). Then, the inverse relation of Eq. (2) can be obtained by the pseudoinverse formulation \( J^* \) together with homogeneous solution \((I - J^* J)z \)
shown below:
\[
\dot{\theta} = J^* \dot{\varphi} + (I - J^* J)z
\]
where \( J^* = (J^T J)^{-1} \), and \( \text{Rank}[J] = m \)
\( J \in \mathbb{R}^{n \times n} \) is the Jacobian matrix, \( I \) is an identity matrix, and \( J_0 \) is the projection matrix that projects \( z \)
onto the null space of \( J \). A particular solution with \( \| \dot{\theta} \|_2 \)
minimized can be obtained by the Moore-Penrose pseudo-inverse as Eq. (4). In this way, a
particular solution with \( \| \dot{\theta} \|_2 \) minimized to Eq. (2) is
given by \( \dot{\theta} = J^* \dot{\varphi} \). For some \( \theta(t_0) \) where
\( \text{Rank}[J(\theta(t_0))] < m \), Eq. (4) cannot be used to evaluate
\( J^* \) and the manipulator is said to be in a singular state or singular state or singular configuration[20].
For redundant manipulators the case with singular configuration can be pursued by Eq. (3). To obtain
the solution of \( z \) through certain optimization criterion may be very complicated[57],[93],[109],[117],
and other approaches[118],[119] based on the iterative update of
joint vector have also been proposed.

Singularity in the motion planning problem has
been studied by many researchers[86]–[109]. The obstacle
avoidance and task priority control have also been studied
using the null-space approach. Cheng et al.[49]
proposed the compact quadratic programming (QP)
method, where the redundancy resolution problem
was transformed into a QP problem. They decomposed the variables into basic and free variables
and eliminated the basic variables by free variables,
therefore, the conventional optimization algorithm
can be applied and the problem size is reduced. Chung
et al.[60] proposed a new and computationally efficient
method for redundancy resolution, which can directly
deal with inequality constraints, and it was well suited
to a multiple criteria problem when the number of
additional tasks is larger than that of DOR.

In this article, an approach using the forward kinematics and the perturbation method is proposed
to solve the motion planning problem with task
priority requirements. Different to the conventional
approach, the inverse of the Jacobian matrix is not
used in the proposed method. The solution is obtained
corresponding to a design objective by using an optimi-
zation procedure.

2. Optimal Motion Planning for Redundant Robot

2.1 Formulation of the redundant inverse
kinematics problem

Consider a redundant manipulator with \( n \) degrees
of freedom, which manipulates in \( m \)-dimensional task
space (\( n > m \)). Using the forward kinematics, the
relation between the end-effector position and the
angular position of the joint can be represented as
\( r(i) = f(\theta) \) or
\[
\begin{bmatrix}
r_1(t) \\
r_2(t) \\
\vdots \\
r_m(t)
\end{bmatrix}
= \begin{bmatrix}
f_1(\theta_1(t), \theta_2(t), \ldots, \theta_k(t)) \\
f_2(\theta_1(t), \theta_2(t), \ldots, \theta_k(t)) \\
\vdots \\
f_m(\theta_1(t), \theta_2(t), \ldots, \theta_k(t))
\end{bmatrix}
\]
where \( r(i) \) is the position vector of the end-effector
in the task space, \( \theta(i) \) is the angular position of the \( i \)th
joint, and \( f \) denotes the relation between the end-
effector position and the angular position of the joint.
For further realization on a computer, the discrete
form of Eq. (5) as the following is used.
\[
\begin{bmatrix}
r_1(k) \\
r_2(k) \\
\vdots \\
r_m(k)
\end{bmatrix}
= \begin{bmatrix}
f_1(\theta_1(k), \theta_2(k), \ldots, \theta_k(k)) \\
f_2(\theta_1(k), \theta_2(k), \ldots, \theta_k(k)) \\
\vdots \\
f_m(\theta_1(k), \theta_2(k), \ldots, \theta_k(k))
\end{bmatrix}
\]
Eq. (6) can be considered as a multi-input multi-
output system with \( \theta(k) \) as the input variable and
\( r(k) \) as the output variable. Suppose \( \theta(k+1) \) can
further be represented as
\( \theta(k+1) = \theta(k) + \Delta \theta(k) \)
then Eq. (6) becomes
\[
\begin{bmatrix}
r_1(k+1) \\
r_2(k+1) \\
\vdots \\
r_m(k+1)
\end{bmatrix}
= \begin{bmatrix}
f_1(\theta(k) + \Delta \theta_1(k), \theta_2(k) + \Delta \theta_2(k), \ldots, \theta_k(k) + \Delta \theta_k(k)) \\
f_2(\theta(k) + \Delta \theta_1(k), \theta_2(k) + \Delta \theta_2(k), \ldots, \theta_k(k) + \Delta \theta_k(k)) \\
\vdots \\
f_m(\theta(k) + \Delta \theta_1(k), \theta_2(k) + \Delta \theta_2(k), \ldots, \theta_k(k) + \Delta \theta_k(k))
\end{bmatrix}
\]
In this manner, the discrete-form inverse
kinematics is applied to find \( \theta(k+1) \) such that
\( r(k+1) \) meets the desired end-effector position in
the task space. This means the solution obtained by
the inverse kinematics problem can be achieved by
Eq. (8) if \( \Delta \theta(k) \) can be determined. With this
motivation, the motion planning problem of a redundant
robot can be formulated as an optimization problem
as follows:
\[
\text{Minimize } \Phi = \frac{1}{2} \| \Delta \theta(k) \|^2 R \Delta \theta(k) + \frac{1}{2} (r_{k+1} - r(k+1))
\]
subject to
\( r(k+1) = f(\theta(k) + \Delta \theta_1(k), \theta_2(k) + \Delta \theta_2(k), \ldots, \theta_k(k) + \Delta \theta_k(k)) \)
and
\( |\Delta \theta(k)| \leq M_\theta \Delta t = \bar{M}_\theta \)
where \( r_{k+1} \) is the desired end-effector position
vector at time \( k+1 \), \( r(k+1) \) is the end-effector position
vector at time \( k+1 \), \( \theta(k) \) is the angular position
of the \( i \)th joint at time \( k \), \( \Delta \theta(k) \) is the angular change

JSME International Journal
of the \(i\)th joint at time \(k\), \(H\) and \(R\) are positive definite weighting matrices, \(\bar{M}\) is the maximum angular change of the \(i\)th joint and \(\Delta t\) is the sampling time. In Eq. (9), the first term on the right hand side is introduced to minimize \(|\Delta \theta|\), and the second term is applied to minimize the tracking error. The solution of this optimization problem using the perturbation method can be described in the section 2.2.

2.2 The perturbation method for the trajectory planning

For a robot system, the least input increment of the turning angle is limited by the driving system. That is, the change of the angular position \(\Delta \theta_i(k)\) is limited in each step. If the least input increment of the turning angle is \(\eta>0\), the optimization problem described in Eq. (9) –(11) can be modified as:

Maximize or Minimize \(\Phi\) \(\Delta \theta_i(k)\) (12)

subject to

\[ r(k+1) = f(\theta_i(k) + \Delta \theta_i(k), \theta_j(k) + \Delta \theta_j(k), \ldots \theta_n(k) + \Delta \theta_n(k)) \]

\[ \Delta \theta_j(k) = j \cdot \eta - m_i \leq j \leq m_i \]

where \(m_i = \bar{M}_i / \eta\) (13.a)

\[ \Delta \theta_j(k) = j \cdot \eta - m_i \leq j \leq m_i \]

where \(m_i = \bar{M}_i / \eta\) (13.b)

Definition:

A perturbation subset \(\Delta \theta_i(k)\) represents the set \(\Delta \theta_i(k) = [\bar{M}_i, -\bar{M}_i, \eta] = [j \cdot \eta - m_i \leq j \leq m_i]\), where \(\bar{M}_i\) is the upper bound of a perturbation at step \(k\) and \(m_i = \bar{M}_i / \eta\). \(\eta\) is the interval of the perturbation. So the possible solutions at step \(k\) compose a perturbation set

\[ \Delta \theta_i(k) = \Delta \theta_j(k) \times \Delta \theta_j(k) \times \cdots \times \Delta \theta_n(k) \]

This searching algorithm has some drawbacks, such as the high computational load and no flexibility. For example, if there are \(n\) joints for a robot, we must compute \((2m_n+1) \cdot (2m_{n-1}+1) \cdots (2m_1+1)\) times, and this computation load is very large. However, the perturbation set can be strained out the bad solution with the permissible zone and then the computation load can be reduced. Figure 1 describes the relationship between the perturbation set and the permissible zone. For example, a three DOF robot with the least input increment of the angular position being 0.01 degree and \(\bar{M}_i = 0.1\) degree, there are 1000 trials will be calculated for each step. But the use of the permissible zone can reduce the trials to 50 - 100. It can also be used to solve the task priority problem described in the next section.

2.3 Permissible zone

For task priority motion planning problems, e.g., the obstacles avoidance case or singularity avoidance case, there are two main tasks to be achieved. One is to achieve the PTP (point-to-point) motion. The other is to maximize or minimize the cost function corresponding to a priority task, such as the collision avoidance or he manipulability measure maximization. Because different tasks may be in conflict with each other, the permissible zone is introduced to relax the PTP motion task. The concept of this method is to bound the tracking errors of the end-effector in the permissible zone and maximize or minimize the cost function corresponding to the additional task at the same time. The use of the permissible zone is illustrated in the following.

A PTP motion-planning problem with priority task can be formulated as follows.

Maximize or Minimize \(\Phi\) \(\Delta \theta_i(k)\) (14)

subject to

\[ r(k+1) = f(\theta_i(k) + \Delta \theta_i(k), \theta_j(k) + \Delta \theta_j(k), \ldots \theta_n(k) + \Delta \theta_n(k)) \]

\[ \Delta \theta_j(k) = j \cdot \eta - m_i \leq j \leq m_i \]

where \(m_i = \bar{M}_i / \eta\) (15.a)

\[ |r(k+1) - r_{ref}(k+1)| < \rho \quad \text{(permissible zone condition)} \]

where \(r_{ref}(k+1)\) and \(r(k+1)\) are the tracking reference and actual end-effector position vectors at step \(k+1\), respectively. \(\Phi\) is the cost function corresponding to the task to be optimized, such as tracking errors, manipulability measure and the shortest distance between the robot and the obstacles, etc.

From the above formulation, the PTP motion planning problem with additional task can be solved as follows. First, the end-effector position should be located inside the permissible zone, and this guarantees the end-effector tracking the desired path.
with a bounded error. Since the inverse kinematics solution to this permissible zone is not unique, the optimal solution is then determined according to the cost function \( \Phi \) in each step.

For illustrating the permissible zone concept, consider a complex problem of the PTP motion with obstacle avoidance shown in Fig. 2. The dashed line indicates the shortest path for the PTP motion. However, for obstacle avoidance the motion path should be modified as shown by the cross line. The PTP motion then can be considered as a sequential sub-PTP motion, where a group of tracking reference points is specified. Let the distance between the two neighboring reference points be \( \varepsilon \) and the reference end-effector positions keep away from the obstacles with a distance \( d \). Then, the permissible zone for the end-effector, as shown in Fig. 3, corresponding to the \( k \)th tracking reference point and

\[
R = \{ r \| r - r_{\text{ref}}(k) \| < \rho \}
\]

where \( \| \cdot \| \) denotes the distance, \( r_{\text{ref}}(k) \) denotes the \( k \)th tracking reference point and \( \rho < \min(d, 0.5\varepsilon) \)

Condition (17) is applied to achieve obstacles avoidance. It is noted that if \( \rho > 0.5\varepsilon \), the end-effector may move sluggishly and even stay around somewhere, and if \( \rho > d \), the collision occurs.

There are five important coefficients should be assigned in this approach. They are the permissible zone radius \( \rho \), the joint rates bounds \( M_i \), the distance between the two neighbouring reference tracking points \( \varepsilon \), the distance between the reference tracking point and the obstacle \( d \), and perturbation level \( \eta \). It is suggested that the path planning distance \( \varepsilon \) is chosen according to the robot dynamic behaviour and the sampling time. The bounds of the joint rates \( M_i \) is chosen according to the physical limit of the joint. The permissible zone radius \( \rho \) is chosen based on \( \varepsilon \) and \( d \). And \( \eta \) is chosen by the accuracy of the motors.

### 3. Point to Point Motion with Optimal Singularity Avoidance

For the multiple task problem, the cost function should be determined according to the additional task with higher priority. The additional task to be optimized can be achieved by maximizing the cost function \( W(\theta) \) and the other additional tasks can be performed by satisfying a set of inequality constraints\(^6\) represented as Eq. (18):

\[
G_i(\theta) \leq 0, \text{ or } G_i(\theta) \geq 0 \quad i = 1, \ldots, k
\]

Then, the redundancy resolution problem with multiple criteria can be formulated as the following local constrained optimization problem:

\[
\text{Max or Min} \ W(\theta) \quad \text{(the task to be optimized)}
\]

subject to

\[
\tau = \dot{q}(\theta) \quad \text{(the forward kinematics)}
\]

\[
g(\theta) \leq 0 \text{ or } g(\theta) \geq 0 \quad \text{(additional tasks to be achieved)}
\]

where

\[
g(\theta) = [G_1(\theta), G_2(\theta), \ldots, G_k(\theta)]^T
\]

For example, a task with avoiding singularity as higher priority, the cost function is in the form of the manipulability measure\(^7\), and other additional tasks, such as obstacle avoidance, can be formulated as an inequality constraint. While for a task with obstacle avoidance as higher priority, the cost function can be defined as the shortest distance between the robot and the obstacle instead of the manipulability measure.

### 3.1 The proposed algorithm

A motion planning with obstacle avoidance and optimal singularity avoidance can be formulated as follows.

\[
\text{Maximize} \quad \Phi = H(\theta) = \sqrt{\det(J^T J)}
\]

(singularity avoidance condition)

subject to

\[
r(k+1) = f(\theta(k) + \Delta \theta(k), \theta(k) + \Delta \theta(k), \ldots, \theta(k) + \Delta \theta(k))
\]

(24)

\[
G(\theta) \leq 0 \quad \text{(obstacle avoidance inequality)}
\]
\[ |r(k+1) - r_{ref}(k+1)| < \rho \]  
(permissible zone condition) \hfill (26)

and
\[
\Delta \theta_i(k) = \{ j \cdot \eta | -m_i \leq j \leq m_i \}
\hfill (27)
\]

where \( m_i = \bar{M}_i / \eta \)

where \( r_{ref}(k+1) \) and \( r(k+1) \) are the tracking reference and the end-effector position vectors at the step \( k+1 \), respectively. The obstacle avoidance condition is defined according to the robot configuration. \( \bar{M}_i \) is the maximum angular change of the \( i \)th joint at each step.

### 3.2 The path planning procedure by the perturbation method

Assume that initially the end-effector is at rest on the desired trajectory and the initial values of the joint variables are \( \theta_0 \) and the final position of the end-effector is \( X_0 \). Let the number of path planning points is "N", and then the path planning points can then be obtained from the tracking trajectory as \( X_i(k) = X_0 + \frac{X_{r0} - X_0}{N} \cdot k \), \( k = 1 \cdots N \). The optimization procedure is performed for each time step until the robot reaches the final state, and the sampling time "\( \Delta t \)" is determined according to the robot dynamics. In the optimization procedure, the searching procedures are proceeded as follows:

1. At the \( k \)th step, perturb the angular position change to generate the perturbation set, \( \Delta \theta_i(k) = \{ j \cdot \eta | -m_i \leq j \leq m_i \} \).
2. Calculate the position of the end-effector \( r(k+1) = f(\theta(k) + \Delta \theta(k)) \) through the forward kinematic of the robot, where \( f(\bullet) \) is the function of mapping the joint angles to the task space variables.
3. Strain out the bad solutions with the permissible zone.
4. Check whether the additional constraint is satisfied.
5. Select \( \Delta \theta_i, i = 1 \cdots n \) among \( |\Delta \theta_i(k)| \leq \bar{M}_i \), as the resulting joint angular position change, such that the cost function \( \Phi(r(k+1), \theta(k+1)) \) is optimized.
6. Determine the local optimal solution and calculate the position of the end-effector at time \( k+1 \). Then \( k = k+1 \) and return to step 1.

From the above procedure, each local optimal solution can be determined at each time step and the angular position change of each joint at each time step is bounded. Figure 4 shows the flow chart of the above procedure.

### 3.3 Study cases

In this section, the performance of the proposed method is evaluated through two cases\(^{(6)}\). Consider the 3R planar manipulator shown in Fig. 5. The forward kinematic equation is given as

\[
x(k+1) = f(\theta(k+1))
\]

where
\[
s_1 = \sin \theta_1, \quad s_{12} = \sin(\theta_1 + \theta_2), \quad s_{123} = \sin(\theta_1 + \theta_2 + \theta_3)
\]
\[ c_1 = \cos \theta_1, \quad c_{12} = \cos(\theta_1 + \theta_2), \]

and

\[ c_{12} = \cos(\theta_1 + \theta_2). \]

Differentiating Eq.(28) yields:

\[
\begin{bmatrix}
\frac{dx}{dt} \\
\frac{dy}{dt}
\end{bmatrix} =
\begin{bmatrix}
-l_s \sin \theta_1 - l_s \sin \theta_2 - l_{s12} - l_{s12} \\
\frac{l_c_1}{l_c_1 + l_c_{12} + l_c_{12}} + \frac{l_c_{12}}{l_c_{12} + l_c_{12}}
\end{bmatrix} \frac{d\theta_1}{d\theta_2} + \frac{d\theta_1}{d\theta_2}
\]

\[ = J(\theta) \begin{bmatrix}
\frac{d\theta_1}{d\theta_2} \\
\frac{d\theta_1}{d\theta_2}
\end{bmatrix} \] (29)

where \( J \) is the Jacobian matrix.

**Case 3.1**

The basic motion task is given as follows:

\[ x(t) = -1.0 \cos (2\pi t) + 3.0 \]

\[ y(t) = -1.0 \sin (2\pi t) \quad \text{for} \quad t \in [0, 1] \] (30)

The lengths of the links are \( l_s = 3.0 \quad l_c = 2.5 \quad l_b = 2.0 \) m. The end-effector starts at \([2.0, 0.0]^T\) to trace a circle trajectory counterclockwise in 1 second, and back to the starting point. It is also required to maximize the manipulability measure, \( H(\theta) = \sqrt{\det (J^T J)} \), for singularities avoidance. The manipulability measure is a nonnegative value to measure the singularity, and it becomes zero only at the singular points. The initial configuration is \([-1.521, 1.951, 1.353]^T \text{(rad)}\), which corresponds to \([2.0, 0.0]^T\) in the task space. If the obstacle avoidance is also required, then the DOR is smaller than the number of the additional tasks. To this problem, the obstacle avoidance can be achieved by satisfying the inequality constraints.

**Definition: the forbidden zone**

The tracking problem with obstacle avoidance is to track a reference trajectory without passing a forbidden zone. The forbidden zone is defined as an area includes the obstacles themselves and the region that can not be intersected for safety reasons. The shape of forbidden zone can be a circle or a rectangle, for simplicity.

In this case, suppose that the forbidden zone is modeled as a circle with radius of \( \sqrt{2} \) and center of \([4.3, -3.0]^T\). The constrained optimization problem corresponding to the PTP tracking with obstacle avoidance and optimal singularity avoidance can be formulated as follows.

Maximize \( \Phi = H(\theta) = \sqrt{\det (J^T J)} \)

(singularity avoidance condition) (31)

subject to

\[ r(k+1) = f(\theta(k) + \Delta \theta(k), \theta(k) + \Delta \theta(k), \ldots \theta(k) + \Delta \theta(k)) \]

\[ G(\theta) = -(l_c_1 + l_c_{12} - 4.3)^2 -(l_s_1 + l_s_{12} + 3.0)^2 \]

\[ + (\sqrt{2})^2 \leq 0 \quad \text{(obstacle avoidance inequality)} \]

\[ |r(k+1) - r_{ref}(k+1)| < \rho \quad \text{(permissible zone condition)} \]

\[ \Delta \theta_i(k) = |j \cdot \eta| - m_i \leq j \leq m_i \]

where \( m_i = M_i / \eta \)

where \( r_{ref}(k+1) \) and \( r(k+1) \) are the tracking reference and the end-effector position vectors at the step \( k+1 \), respectively. \( M_i \) is the maximum angular change of the \( i \)th joint.

In this task, the kinematic parameters of the proposed method are as follows. The number of path planning points is 1000 so that \( \epsilon = \frac{2\pi}{1000} = 0.0062832 \).

The radius of the permissible zone \( \rho = 0.001 \leq \frac{\epsilon}{2} \), the joint rates bounds \( M_i = 0.5 \text{ deg/step} \), the sampling time is 0.001 sec, and \( \eta = 0.05 \) deg. For illustrative purpose, the results for this task without considering obstacle avoidance are also demonstrated. The results obtained by the proposed method are shown in Fig. 6 and 7. Figure 6 shows that the manipulator successfully performs two additional tasks as well as the tracking motion task. Results for the joint angles
of these two cases are shown in Fig. 8 and the manipulability measures for two cases are shown in Fig. 9. The initial and final states are shown in the Fig. 10. The result also shows that the proposed approach has a cyclic property, and this is because the manipulability measure is minimize. The actual and desired positions of end-effector are given in Fig. 11.

**Case 3.2**

A more complicated case of a planar manipulator with four joints is considered in this case study. The length of the fourth link is 1.5 m and the others are the same as the previous. The basic motion task is to trace a straight line from $[6.0, 0.0]^T$ to $[2.0, 0.0]^T$ in 1 second. The additional task with higher priority is to avoid singularity by maximizing the manipulability measure. The initial optimal configuration is $[-0.914, 0.662, 1.180, 0.267]^T$ (rad). Suppose that the forbidden zone is modeled as a circle with radius of $\sqrt{2}$ and center of $[-0.3, -4.2]^T$. The corresponding optimization problem can be formulated as follows:

Maximize $\Phi = H(\theta) = \sqrt{\det(J^TJ)}$

(Singularity avoidance condition) \hspace{1cm} (32)

subject to

$r(k+1) = f(\theta(k)) + \Delta \theta(k)$, $\theta(k) + \Delta \theta(k)$,$\ldots$,$\theta(k) + \Delta \theta(k)$

$O(\theta) = -(l_1 c_1 + 0.3)^2 -(l_2 c_2 - 1)^2 + (\sqrt{2})^2 \leq 0$

(Obstacle avoidance inequality)

$|r(k+1) - r_{ref}(k+1)| < \rho$

(Permissible zone condition)

and

$\Delta \theta(k) = (j, \eta) - m_j \leq j \leq m_i$, where $m_i = M_i/\eta$

where $r_{ref}(k+1)$ and $r(k+1)$ are the tracking reference and the end-effector position vectors at the step $k+1$, respectively. $M_i$ is the maximum angular change of the $i$th joint. Further, the forward kinematic equations are:

$$
\begin{bmatrix}
    x \\
    y
\end{bmatrix} = f(\theta) = 
\begin{bmatrix}
    l_1 c_1 + l_2 c_2 + l_3 c_3 + l_4 c_4 \\
    l_1 s_1 + l_2 s_2 + l_3 s_3 + l_4 s_4
\end{bmatrix}
$$

(33)

In this task, the kinematic parameters of the proposed
method are as follows. The number of path planning points is 1000, so that $\epsilon = \frac{2\pi}{1000} = 0.0062832$. The radius of the permissible zone $\rho = 0.001 < \frac{\epsilon}{2}$, the joint rates bounds $M_t = 0.5 \text{deg/step}$, and $\eta = 0.05 \text{deg}$. Figure 12-14 show the results of these two cases obtained by the proposed method. Figure 15 shows the manipulability measures for both cases. In the resulting operation, the mission of singularity avoidance has higher priority and it is required to be as large as possible when the robot is away from the forbidden zone, which corresponds to time period $t \leq 0.65$ second. When the manipulator collides the forbidden zone, the mission of obstacle avoidance has higher priority and the manipulability measure is to be maximized subject to the obstacle avoidance condition.

4. Point to Point Motion with Optimal Obstacle Avoidance

In this section, the performance of the proposed method is applied to another two simulations. Different to the previous section, the task of obstacle avoidance is required to be optimal instead of avoiding singularity. Therefore, the cost function is chosen as $\Phi = \min(D_o)$, where $D_o$ represents the shortest distance between the $i$th joint and the $j$th obstacle. In this way, $\min(D_o)$ represents the shortest distance between the robot and the obstacle. The task of singularity avoidance is formulated as an inequality constraint, and the PTP motion problem with singularity avoidance and optimal obstacle avoidance can be formulated as follows.

4.1 The problem formulation

Not only a motion planning problem with avoiding an obstacle but also avoiding singularity can be
formulated as the following optimization problem.

\[
\text{Maximize } \Phi = \min_i (D_o) \quad \text{(Obstacle avoidance condition)}
\]

subject to

\[
\begin{align*}
    r(k+1) &= f(\theta(k) + \Delta \theta(k), \theta(k) + \Delta \theta(k), \ldots \theta_n(k) + \Delta \theta_n(k)) \\
    G(\theta) &= (\sqrt{\det(JF^T) - M_s}) \geq 0 \\
    \left| r(k+1) - r_{ref}(k+1) \right| &< \rho \\
    \text{(Singularity avoidance inequality)} \\
    \text{(Permissible zone condition)}
\end{align*}
\]

and

\[
\Delta \theta(k) = \{j \cdot \eta | m_i \leq j \leq m_i\},
\]

where \( m_i = \hat{M}_i / \eta \)

where \( r_{ref}(k+1) \) and \( r(k+1) \) are the tracking reference and the end-effector position vectors at the step \( k+1 \), respectively. \( \hat{M}_i \) is the maximum angular change of the \( i \)-th joint, and \( M_s \) is a lower bound for the manipulability measure, it is used to avoid the singularity.

The path planning procedure is the same as that in section 3.2, but the cost function is defined to be the shortest distance between the robot and the obstacles. The manipulability measure is bounded for singularity avoidance. To maximize this cost function will keep the robot away from the obstacle as far as possible. Further, the singularity inequality make the robot avoid singularity.

4.2 Distance between the obstacle and the robot arm

The obstacle can be described by a series of boundary points. These boundary points can be defined to represent the distinct feature of the obstacle. For example, a planar rectangular obstacle can be represented by its extreme points. In the case where the edge-length of the obstacle is greater than the link length, extra boundary points on the edge need to be specified so that any two boundary points are smaller than the link-length. Then the distance between the obstacle and the robot arm can be defined as the shortest distance between each boundary point and the robot arm link. The distance between the \( i \)-th link and the \( j \)-th boundary point denoted as \( D_{ij} \) is defined as follows.

In the following derivations, it is assumed that the position of the boundary point \( o_j(x_{o_j}, y_{o_j}) \) is prespecified or can be obtained by the sensory system. As shown in Fig. 16, the obstacle avoidance point \( x_o \), on \( i \)-th link \( \overline{A_jB_j} \) with \( l_i \), is closest to the \( j \)-th boundary point \( o_j(x_{o_j}, y_{o_j}) \). From Fig. 17, the projection of \( \overline{A_jo_j} \) on \( \overline{A_jB_j} \) can be obtained by

\[
P_j = \overline{A_jo_j} \cdot \overline{A_jB_j} / l_i
\]

Different values of \( P_j \) denote different obstacle-link relationships, which can be divided into three cases as follows.

**Case 1** \( 0 \leq P_j < l_i \)

This case corresponds to the situation shown in Fig. 17, where the obstacle avoidance point \( x_o \) is among the \( i \)-th link and will be on \( C_i \). Therefore, the shortest distance between the \( j \)-th boundary point \( o_j \) and the link \( i \), denoted \( D_{ij} \), is:

\[
D_{ij} = o_jC_i = \sqrt{A_jo_j^2 - P_j^2}
\]

**Case 2** \( P_j < 0 \)

This case corresponds to the situation shown in Fig. 18, where \( x_o \) is assigned to be on point \( A_i \),
therefore, $D_0$ can be determined as
\[ D_0 = A_{i0} \]  
(37)

Case 3 $P_0 \geq l_i$

This case corresponds to the situation shown in Fig. 19, where $x_{oi}$ is assigned to be on point $B_i$, therefore $D_0$ can be obtained as
\[ D_0 = \sqrt{O_i C_i^2 + (P_0 - l_i)^2} \]  
(38)

4.3 Study case

In this section, the performance of the proposed method is evaluated by the following study cases. The first case has the same basic task and configuration as Case 4.1, where the task of obstacle avoidance is to be optimized.

Case 4.1

The basic motion task in the first simulation is described as follows:
\[ x(t) = -1.0 \cos(2\pi t) + 3.0 \]
\[ y(t) = -1.0 \sin(2\pi t) \quad \text{for} \quad t \in [0, 1] \]  
(39)
The end-effector starts from $[2.0, 0.0]^T$ to trace a circle trajectory counterclockwise in 1 second, and stops at the starting point. The corresponding initial state of the robot joint is $[-1.521, 1.951, 1.353]^T$ (rad). The task of singularity avoidance is formulated as an inequality constraint. In this case, the forbidden zone is modeled as a circle with radius of $\sqrt{2}$ and center of $[4.3, -3.0]^T$. So there are only one obstacle point $O_1 = [4.3, -3.0]^T$, and the shortest distance between the obstacle and the robot arm is $\text{Min}(D_0)|_{i=1, j=1}$. The constrained PTP motion problem with singularity avoidance and optimal obstacle avoidance can be formulated as follows.

Maximize $\Phi = \text{Min}(D_0)|_{i=1, j=1}$

(obstacle avoidance condition)  
(40)
such that
\[ r(k+1) = f(\theta(k) + \Delta \theta(k), \theta(k) + \Delta \theta(k), \ldots, \theta(k) + \Delta \theta(k)) \]
\[ C(\theta) = (\sqrt{\det(J^TJ)} - M_j) \geq 0 \]
(singularity avoidance inequality)
\[ |r(k+1) - r_{ref}(k+1)| < \rho \]
(Permissible zone condition)

and
\[ \Delta \theta(k) = \{j, \eta | -m_i \leq j \leq m_i\}, \]
where $M_i = \tilde{M}_i/\eta$

where $r_{ref}(k+1)$ and $r(k+1)$ are the tracking reference and the end-effector position vectors at the step $k+1$, respectively. $\tilde{M}_i$ is the maximum angular position change of the $i$th joint. $M_i$ is the manipulability measure bound.

In this simulation, the initial configuration is $X_0 = [2.0, 0.0]^T$ (m) and the final configuration is $X_2 = [2.0, 0.0]^T$ (m). The number of path planning points $N$ is 1000 so that $\epsilon = \frac{\pi \times 2}{N} = \frac{6.2832}{1000} = 0.0062832$ (m), the radius of the permissible zone $\rho = 0.001 < \frac{\pi}{2}$, the joint rates bounds $\tilde{M}_i = 0.5 \text{deg/step}$, the singularity bound $M_x = 5.5$, and $\eta = 0.05 \text{deg}$. Figure 20 shows that the manipulator successfully avoids the obstacle, and the robot is away from the obstacle as far as possible. The resulting joint angular positions obtained by the proposed method are shown in Fig. 21. Figure 22 shows that the basic motion task is also performed well. The manipulability measures for these two cases are shown in Fig. 23. From Fig. 23, the manipulability measure is less than 5.5 when $t > 0.95$ second so that the task of avoiding singularity is dominate within this period of time.

Case 4.2

In the second case, a PTP motion planning with obstacles avoidance is presented. In this case, it is required that the end-effector moves from one position (with a corresponding configuration) to the other position with a rectangular obstacle in between the two points. The configuration of the robot and the required PTP path planning tasks are described as follows.

The lengths of the robot links described in Fig. 5

---

Fig. 19 Schematic diagram for the case $P_0 \geq l_i$

Fig. 20 The results with inequality constraint obtained by the proposed method
Fig. 21  The resulting joint angular positions obtained by the proposed method

Fig. 22  The results of the actual and desired position of the end-effector

Fig. 23  The manipulability measures for two cases

Fig. 24  The initial state and the forbidden zone

Fig. 25  The resulting joint angular positions

Fig. 26  The resulting PTP path planning by the proposed method

are $l_1=5, l_2=5, l_3=3$ (m). The rectangular obstacle has its four corners located at [4, 2], [4, -0.5], [8, 2] and [8, -0.5], respectively. The PTP guidance law is generated as follows. (1) Define the forbidden zone with a safety distance $d=0.3$ from the obstacle, and it is shown in Fig. 25. (2) Make the shortest path for the PTP motion indicated by a dashed line as shown in Fig. 25. (3) Move the end-effector of the robot along the solid straight line until it reaches the forbidden zone. (4) Move the end-effector to the another side of the obstacle along the edges of the forbidden
zone until it avoids the obstacle. (5) Move it along the shortest path to the destination point shown by a straight path.

In this task, there are four obstacle points to be avoided and the number of robot links is three so that the cost function \( \Phi \) is chosen as \( \text{Min}(D_i)_{i=1,2,3,4} \). The initial position of the end-effector \([x(t_0), y(t_0)]=[2.40761, 1.86044]\) which corresponds to the joint configuration \([\theta_i(t_0), \theta_j(t_0), \theta_k(t_0)]=[1.81776, -1.88069, -1.97855]\) (rad) and the desired end-effector destination position \([x(t_f), y(t_f)]=[10.0, 0.0]\). The kinematic parameters of the proposed method are chosen as follows. The bound of the joint rate \( \bar{M}=0.5 \) deg/step, \( \epsilon=0.015 \), \( d=0.3 \) and the permissible radius \( \rho=0.003<\min\left(\frac{\pi}{2}, d\right) \), and \( \eta=0.05 \) degree. Using the proposed approach, this PTP motion planning task with optimal obstacle avoidance is formulated as follows.

\[
\text{Maximize } \Phi = \text{Min}(D_i)_{i=1,2,3,4} \\
\text{(obstacle avoidance condition)} \quad (41)
\]

subject to

\[
r(k+1) = f(\theta(k) + \Delta \theta(k), \theta(k) + \Delta \theta(k), \ldots, \theta(k) + \Delta \theta(k))
\]

and

\[
|r(k+1) - r_{ref}(k+1)| < \rho
\]

(permmissible zone condition)

under

\[
\Delta \theta(k) = \{j \cdot \eta | -m_i \leq j \leq m_i\},
\]

where \( m_i = \bar{M}/\eta \)

The results of the proposed approach are shown in Fig. 24-26. It shows that the proposed path planning method for redundant robot manipulator results smooth PTP motion.

---

5. The Computation Performance of the Proposed Method

The problem of robotic control is usually solved by a two-stage optimization procedure. The first stage considers a motion-planning (or path-planning) problem, and the second stage handles the trajectory-tracking problem. The motion planner receives spatial path description and the environment constraints, from which it calculates a time history of the desired positions. Then, the trajectory controller tracks the desired position and velocity. So the motion planning can be on-line or off-line to support the trajectory controller.

The computation time for the previous study cases using the proposed method is shown in Table 1. It is programmed in C++ on Pentium 133. From Table 1, all cases can be used on-line except case (d) and (e). It shows that case (b) and (d) spent more computation time than cases (c) and (e) due to the introduction of the inequality constraint. Comparing case (f) with case (g), the more boundary points the more computation time required. For case (d) and (e), the motion planning may be proceeded off-line as a reference trajectory, controller is applied of force the robot track this trajectory on-line. In the case when small computation time is requested, the proposed algorithm can be implemented on a parallel processing system.

6. Conclusions

In this article, an optimization approach to solve the inverse kinematic problem of the non-redundant and redundant manipulators with multi-task is proposed. It showed that a redundancy resolution
problem with multiple cost functions can be formulated by a local equality and inequality constrained optimization problem, and can be solved by the proposed forward search method. The angular position change in each time interval are bounded and the local tracking errors are bounded in the radius of the permissible zone using the proposed approach. Further, the equality constraint is imposed for the PTP motion task, and the cost function is chosen according to the additional task to be optimized. While the other additional tasks are performed by satisfying a set of inequality constraints. For the problem of tracking manipulator trajectories with optimal obstacle avoidance, the singularity avoidance has also been achieved. The proposed method uses local optimal methods, forward kinematics, and a permissible zone concept to solve the inverse kinematic problem instead of the inverse Jacobian matrix. This leads to its robustness to the singularities and its feasibility to solve multi-task problem by defining a corresponding optimization problem.

Acknowledgement

The authors would like to thank the National Science Council of the Republic of China for the support of this work under the Grant No. NSC86-2212-E-006-037.

References


