

Article

Singular Configuration Analysis and Singularity Avoidance with Application in an Intelligent Robotic Manipulator

Helin Wang, Ziqiang Zhou, Xianyou Zhong and Qijun Chen *

Department of Control Science and Engineering, Tongji University, Shanghai 201804, China; whl7228@tongji.edu.cn (H.W.); ziqiang.zhou@tongji.edu.cn (Z.Z.); zhongxianyou@tongji.edu.cn (X.Z.)
* Correspondence: qjchen@tongji.edu.cn

Abstract: Recently, robotic sensor systems have gained more attention annually in complex system sense strategies. The robotic sensors sense the information from itself and the environment, and fuse information for the use of perception, decision, planning, and control. As an important supplement to traditional industrial robots, co-bots (short for co-working robots) play an increasingly vital role in helping small and medium-sized enterprises realize intelligent manufacturing. They have high flexibility and safety so that they can assist humans to complete highly repetitive and high-precision work. In order to maintain robot safe operation in the increasing complex working environment and human–computer intelligent interactive control, this paper is concerned with the problem of applicant accuracy analysis and singularity avoidance for co-bots. Based on the dynamic model with load and torque sensors, which is used to detect the external force at the end of the robot, this paper systematically analyzes the causes of singularity phenomenon in the robot motion control. The inverse solution is obtained by analytical method and numerical method, respectively. In order to ensure the smooth and safe operation in the whole workspace, it is necessary for a robot to avoid singularity. Singularity avoidance schemes are utilized for different control tasks, including point-to-point control and continuous path control. Corresponding simulation experiments are designed to verify the effectiveness of different evasion schemes, in which the advantages and disadvantages are compared and analyzed.

Keywords: co-bot; modeling robotic systems; kinetic inverse solution; singular configuration analysis; singularity avoidance; load and torque sensors



Citation: Wang, H.; Zhou, Z.; Zhong, X.; Chen, Q. Singular Configuration Analysis and Singularity Avoidance with Application in an Intelligent Robotic Manipulator. *Sensors* **2022**, *22*, 1239. <https://doi.org/10.3390/s22031239>

Academic Editor: David Scaradozzi

Received: 12 January 2022

Accepted: 5 February 2022

Published: 6 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Recently, appliance accuracy analysis and human–computer interaction have undergone rapid development in intelligent assembly machine factories, with typical examples including assembly, polishing, dual-arm coordination, or dexterous hand manipulations [1–3]. In order to adapt to small-batch, customized, and short-cycle production, it is necessary for co-bots, a new type of industrial robot that can work with human beings in the same working environment, to develop a hierarchical control algorithm that enables safe and stable cooperative locomotion. With dynamic model and torque sensors, the robot is achieving the drag control by compensations of gravity and friction, which counteract external disturbances. However, it will inevitably lead to the complexity and uncertainty of environment [4,5]. Reaching the singular region makes it easy for a robot to cause various problems, such as instability, poor performance, and so on. Emphasis is placed on the co-bot singularity. Hence, the singularity avoidance is the basis to ensure co-bot stable operation in the man–machine cooperation.

The singular configuration is a phenomenon which hampers the motion of the robot end effector [6,7]. The freedom of the robot end will decrease in a singular configuration, so it cannot be controlled to move in directions. While the robot approaches the singular adjacent region, some joints calculated by the inverse kinematic tend to infinity, which will

cause the decline of tracking expected trajectory [8]. The singularity is the inherent characteristic of an articulated robot, which mainly appears in finding the inverse kinematics solution. The actual performance is that the calculated expected joint speed is much greater than the maximum joint speed which the actual motor can provide. Bohigas [9] gave the general algorithm of robot singular configuration with different mechanical structures. Müller [10] analyzed singular points by using Lie groups separated from the specific structure.

For more general singularity avoidance problems, Maciejewski [11] proposed a damped least square method with a damping term near singular region to obtain approximate solution of joint velocity. The essence lies in limiting the joint velocity near the singularity by adding a damping coefficient. Serial robots conduct seam welding, sealant application, spray painting, polishing, deburring, or other tasks requiring uninterrupted continuous motion. Chiaverini [12] introduced a continuous nonlinear function to characterize the damping factor, which is used to ensure the continuity of joint velocity near singular configuration. However, it will increase the tracking error of robot end in all directions. The damping factor is only added to the minimum singular value of a Jacobian matrix. In addition, Megalingam et al. [13] proposed singular separation to determine the singular direction of Kinova robots, and then avoid joint movement to pass through singular points. References [14,15] also proposed similar analysis methods to analyze singular directions, while these methods are not universal. A large number of SVD decomposition operations are required in the above singular point avoidance methods. In order to reduce amount of computation, Xu et al. [16] proposed the method of “singular separation + damping reciprocal” for PUMA robots. It was the first time to separate the singular problem into two parts of position and attitude. The redundant robots have infinite sets of inverse kinematics solutions, hence the optimal solution (i.e., the solution furthest away from the singular region) can be found by designing some optimization objective. Wang et al. [17] summarized some methods of using redundancy to avoid the singular region with series robot. The ultimate goal is to find the solution of the singular configuration of the robot principle in countless redundant solutions. The optimization directions include the determinant and condition number of Jacobian matrixes. However, the calculation using Jacobian matrix is too complex, and an optimization is proposed in reference [18,19].

All the singularity avoidance methods above are at the cost of tracking accuracy. The parameters of most methods are closely related to experience, especially the determination of damping factor and singular region. In addition to singularity avoidance method satisfying trajectory tracking, Taki et al. [20] proposed a method strictly tracking the desired path. Different from damping avoidance method, the avoidance algorithm proposed is to reduce running speed by reducing the expected speed when approaching the singular region, so as to avoid limiting the motion of the robot end effector caused by singular configuration. However, the strange avoidance method will only produce partial errors in tracking directions.

All these observations motivate the current study. In this paper, we concentrate on singular configuration analysis and singularity avoidance of co-bot, taking the JACO2 robot of Kinova Robotics company as an example [4,13]. It has seven DOF (degrees of freedom), with weight of 5.5 kg, a load of 1.5 kg, and a maximum arm span of 98.4 cm. Each joint motor is equipped with a torque sensor. The rotating potentiometer detects robot joint rotation, and the force sensor reflects ground reaction force information. The main contributions can be summarized as follows.

1. The D-H modeling method is utilized to build a joint coordinate system of the JACO2 robot. The inverse kinematics is solved by analytical method and numerical method, and the operation speed and accuracy of two schemes are compared.
2. The singularity caused by inverse kinematics is analyzed, and the robot singular configuration conditions are based on the block analysis of wrist Jacobian matrix, which are divided into three singular types: internal singularity, external singularity, and wrist singularity.

- According to different task requirements, singularity avoidance schemes are utilized in robot redundancy, damped least squares, and singularity consistency. It is noted that these works focus on the working principle and control methods of robotic sensor system.

2. Materials and Methods

2.1. D-H Model

The kinematics model of the JACO2 robot with D-H model is illustrated in Figure 1, and its parameters are shown in Table 1.

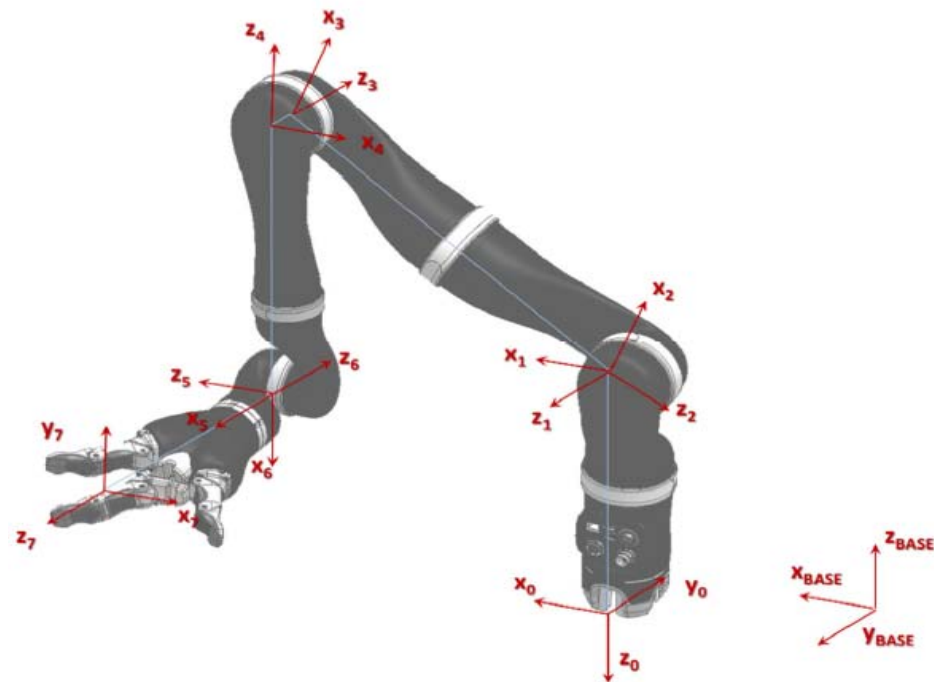


Figure 1. JACO2 robot coordinate system.

Table 1. D-H parameters of the JACO2 robot.

i	α_{i-1} (rad)	α_{i-1} (m)	d_i (m)	θ_i (rad)
1	$\pi/2$	0	-0.2755	θ_1
2	$\pi/2$	0	0	θ_2
3	$\pi/2$	0	-0.41	θ_3
4	$\pi/2$	0	-0.0098	θ_4
5	$\pi/2$	0	-0.3111	θ_5
6	$\pi/2$	0	0	θ_6
7	π	0	-0.2638	θ_7

According to D-H parameters, the transformation matrix between each links is

$${}^{base}T_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^iT_{i+1} = \begin{bmatrix} \cos\theta_{i+1} & 0 & \sin\theta_{i+1} & 0 \\ \sin\theta_{i+1} & 0 & -\cos\theta_{i+1} & 0 \\ 0 & 1 & 0 & d_{i+1} \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^6T_7 = \begin{bmatrix} \cos\theta_7 & \sin\theta_7 & 0 & 0 \\ \sin\theta_7 & -\cos\theta_7 & 0 & 0 \\ 0 & 1 & -1 & d_7 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where $i = 0, 1, 2, \dots, 5$. Thus, the forward kinematics equation of the JACO2 robot can be expressed as

$${}^{base}T_n = {}^{base}T_0 {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_7 = f(\Theta) \quad (1)$$

where ${}^i T_j$ denotes the homogeneous transformation matrix from $\{i\}$ to $\{j\}$ coordinate system, and $\Theta = \{\theta_1, \theta_2 \dots \theta_n\}$ denotes angle information of each joint. The Euler angle (x-y-z)[α, β, γ] and $p = [p_x, p_y, p_z, \alpha, \beta, \gamma]^T$ denote attitude of robot end and end pose, respectively, where $[p_x, p_y, p_z]$ denote robot end position.

2.2. Inverse Kinematics

The analytical solution and numerical solution are generally adopted to resolve robot inverse kinematics. The former uses a robot's geometric configuration by separating the forward kinematics parameters. The latter is based on numerical iteration, which obtains the optimal solution by setting objective function and controlling joint angle to move to the opposite gradient direction.

The JACO2 robot is a seven-DOF spherical revolution spherical configuration series robot. For a certain end pose, the robot has countless sets of inverse kinematics solutions. The redundancy makes the JACO2 robot operate more flexible than a traditional six-DOF series robot but increases the computational complexity. Two methods are used to solve inverse kinematics, respectively.

1. Analytical solution

The JACO2 robot has one more rotating joint at third joint and one more offset at fourth joint. Taking the third joint angle θ_3 as the redundant parameter, other joint angles are

$$\theta_i = f(\theta_3) \quad i = 1, 2, 4, 7 \quad (2)$$

All inverse kinematics solutions can be obtained by traversing all effective value, and only one set of optimal solution can be selected according to cost function. Therefore, the inverse kinematics optimization problem is simplified as

$$\begin{aligned} & \min f(\Theta) \\ & \text{s.t. } {}^{base}T(\Theta) = {}^{base}T^d, \Theta \in [\Theta^{\min}, \Theta^{\max}], \end{aligned} \quad (3)$$

where $\Theta = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5 \ \theta_6 \ \theta_7]^T$ is angle vectors of each joint, ${}^{base}T(\Theta)$ is the forward kinematics equation, ${}^{base}T^d$ is the given end pose, and $[\Theta^{\min}, \Theta^{\max}]$ represents the constraint of each joint.

Substituting (2) into (3), the inverse kinematics optimization problem is turned into a one-dimensional optimization of θ_3 :

$$\min f(\theta_3) \quad \text{s.t. } \Theta \in [\Theta^{\min}, \Theta^{\max}]. \quad (4)$$

Selecting the motion amplitude of each joint as the cost function,

$$f(\Theta) = \sum_{i=1}^7 \lambda_i (\theta_i^{t+1} - \theta_i^t)^2, \quad (5)$$

where λ_i is the weight coefficient, θ_i^{t+1} is the solution of the i -th joint in the current period, and θ_i^t is the solution of the i -th joint in the previous cycle. A greedy search is used to find the optimal solution.

2. Numerical solution

The numerical method focuses on the equation relationship instead of robot geometric configuration. Given this, $f(\Theta) = {}^{base}T_n$, expected position ${}^{base}T_d$, which are to be solved with constraints $\Theta \in [\Theta^{\min}, \Theta^{\max}]$. The Jacobian matrix describes the relationship between robot end velocity and joint velocity:

$$\dot{x} = J\dot{q} \quad (6)$$

where $\dot{x} = [v_x v_y v_z w_x w_y w_z]^T$ represents terminal velocity. $R_{3 \times 3}$ describes the relationship between Euler angle and terminal angular velocity.

$$\dot{p} = \begin{bmatrix} I_{3 \times 3} \\ R_{3 \times 3}^{-1} \end{bmatrix} \dot{x} = \begin{bmatrix} I_{3 \times 3} \\ R_{3 \times 3}^{-1} \end{bmatrix} J \dot{q} = J' \dot{q}, R_{3 \times 3} = \begin{bmatrix} 1 & 0 & \sin\beta \\ 0 & \cos\alpha & -\sin\alpha \cos\beta \\ 0 & \sin\alpha & \cos\alpha \cos\beta \end{bmatrix} \quad (7)$$

The pseudocodes of analytical and numerical inverse kinematics solutions are shown in Algorithms 1 and 2.

Algorithm 1: According to the Cost Function (7), the Obtained Optimal Solution θ_3

Input: T^d (End pose to be solved)
 Θ^0 (Current joint angle vector)
 n (Iterative numbers per layer loop)
 ε (Minimum allowable error)
 $[\theta_3^{\min}, \theta_3^{\max}]$ (Possible value range)
Output: Θ^{opt} (Optimal joint angle vector)

- 1: **do**
- 2: $r = (\theta_3^{\max} - \theta_3^{\min}) / (n - 1)$ (Search step of θ_3)
- 3: $S = \{\theta_3^{\min} + (m - 1)r | m = 1 \dots n\}$
- 4: $C_{\max} = \text{inf}$ (Cost)
- 5: **for** r in S **do**
- 6: $\theta_3 = r$
- 7: $\Theta = \text{InvKin}(T^d, \theta_3)$ (Given θ_3 to find inverse kinematics solutions. If the solution is within the joint limit, Θ is returned; otherwise, Φ is returned.)
- 8: **if** $\Theta \neq \Phi$ and $f(\Theta) < C_{\max}$ **then**
- 9: $C_{\max} = f(\Theta)$
- 10: $\Theta^{opt} = \Theta$
- 11: **else**
- 12: **continue**
- 13: **end**
- 14: $\theta_3^{\min} = \theta_3^{opt} - s$
- 15: $\theta_3^{\max} = \theta_3^{opt} + s$
- 16: **while** $\gamma > \varepsilon$

Algorithm 2: φ is the Partial Derivative of the Joint Angle

Input: P^d (Terminal position vector needed to be solved)
 Θ^0 (Current joint angle vector)
 λ (Step size per iteration)
 n_{\max} (Maximum number of iterations)
 ε (Maximum allowable error)
Output: Θ^{opt} (Optimal joint angle vector)

- 1: $\Theta = \Theta^0, n = 1$
- 2: $err = P^d - f(\Theta)$
- 3: **do**
- 4: $\Delta\Theta = J^+ \cdot err + (I_{7 \times 7} - J^+ J) \varphi$ (φ is used to control the iteration direction)
- 5: $\Delta\Theta = \text{normalize}(\Delta\Theta)$ (Normalized to $[-\pi, \pi]$)
- 6: $\Theta = \Theta + \lambda \cdot \Delta\Theta$
- 7: $err = P^d - f(\Theta)$
- 8: $n = n + 1$
- 9: **while** $\|err\| < \varepsilon$ or $n > n_{\max}$
- 10: $\Theta^{opt} = \Theta$

2.3. Singular Configuration Analysis

Both of the two methods illustrated in Section 2.2 will introduce singularity. Whether the robot is in a singular configuration depends on the full rank of the Jacobian matrix,

$$J = \begin{bmatrix} J_p \\ J_w \end{bmatrix} = \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} e_1 \times r_1 & e_2 \times I & \dots & e_7 \times r_7 \\ & e_1 & & e_2 & & & e_7 \end{bmatrix} \tag{8}$$

where e_i is the unit vector of the i -th joint, and r_i is the vector between the origin and end position. The Jacobian matrix at the end and wrist satisfies

$$J_e = \begin{bmatrix} I_{3 \times 3} & U_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix} J_w, U_{3 \times 3} = \begin{bmatrix} 0 & -d_7c_6 & -d_7s_5s_6 \\ d_7c_6 & 0 & d_7c_5s_6 \\ d_7s_5s_6 & -d_7c_5s_6 & 0 \end{bmatrix}. \tag{9}$$

Let c_i and s_i be abbreviations of $\cos\theta_i$ and $\sin\theta_i$, respectively, and that $\det(J_e J_e^T) = \det(J_w J_w^T)$, $\text{Rank}(J_e) = \text{Rank}(J_w)$. According to (8), the Jacobian matrix of wrist is

$$J_w = \begin{bmatrix} J_{113 \times 4} & 0_{3 \times 3} \\ J_{213 \times 4} & J_{223 \times 3} \end{bmatrix}. \tag{10}$$

It is expressed as (11) in the wrist coordinate system:

$${}^w J_w = \begin{bmatrix} {}^{base}R_w^{-1} & 0_{3 \times 3} \\ 0_{3 \times 3} & {}^{base}R_w^{-1} \end{bmatrix} J_w \tag{11}$$

where ${}^{base}R_w$ represents wrist Jacobian matrix.

1. Positional singularity

The unfilled rank of the first three rows of the Jacobian matrix will result in position singularity, which can be described as $\det({}^w J_{11} {}^w J_{11}^T) = 0$,

$$\det({}^w J_{11} {}^w J_{11}^T) = \sum_{i=1}^4 M_i^2 \tag{12}$$

where M_i represents the determinant of cofactor of matrix ${}^w J_{11}$.

$$\begin{aligned} M_1 &= \det([j_1 \ j_2 \ j_3]) = 0, \\ M_2 &= \det([j_1 \ j_2 \ j_4]) = d_3 d_5 s_4 (d_3 s_2 - d_5 s_2 c_4 + c_2 d_5 c_3 s_4 + c_2 d_4 s_3), \\ M_3 &= \det([j_1 \ j_3 \ j_4]) = d_3 d_5 s_2 s_4 (c_3 d_4 - d_5 s_3 s_4), \\ M_4 &= \det([j_2 \ j_3 \ j_4]) = d_3 d_5 s_4 (d_4 s_3 + c_3 d_5 s_4). \end{aligned}$$

The conditions in whether each co-factor equals zero are expressed as Table 2.

Table 2. The condition in which the determinant of the ${}^w J_{11}$ cofactor of each order is 0.

Number	Condition	M_1	M_2	M_3	M_4
1	$s_4 = 0$	✓	✓	✓	✓
2	$d_3 s_2 - d_5 s_2 c_4 + c_2 d_5 c_3 s_4 + c_2 d_4 s_3 = 0$	✓	✓		
3	$s_2 = 0$	✓		✓	
4	$c_3 d_4 - d_5 s_3 s_4 = 0$	✓		✓	
5	$d_4 s_3 + d_5 s_3 s_4 = 0$	✓			✓

When $s_4 = 0$, the θ_4 joint limitation is $32^\circ \sim 328^\circ$, the robot arms stretch out, and the end is at the boundary point, which is called boundary singularity. The θ_2 joint limitation is $49^\circ \sim 311^\circ$, when $d_4 s_3 + d_5 s_3 s_4 = 0$, the JACO2 robot is in a singular state, which is called internal singularity, which is shown in Figure 2. The singularity condition is illustrated in Table 3.

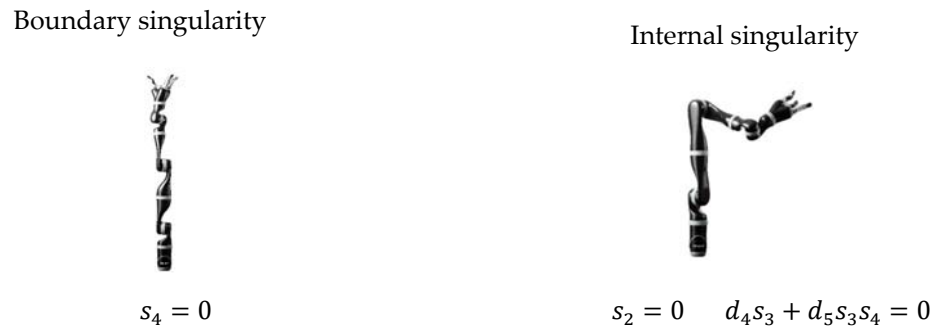


Figure 2. Position singularity of the JACO2 robot.

Table 3. The position singularity condition.

Number	Condition Combination in Table 2 (Serial Number)	Simplified Conditions
1	1	$s_4 = 0$
2	2 + 3 + 5	$s_2 = 0, d_4s_3 + d_5s_3s_4 = 0$
3	2 + 4 + 5	Not established (4 and 5 cannot be met at the same time).

2. Attitude singularity

Similarly, the attitude singularity can be judged by analyzing whether the last three rows of the wrist Jacobian matrix are less than the rank. Substituting each item of Jacobian matrix, the robot attitude singularity condition is shown in Table 4, and its wrist singularity condition is shown in Figure 3.

$$\det({}^w J_{22} {}^w J_{22}^T) = s_6^2 = 0, \quad \det({}^w J_w {}^w J_w^T) = 2d_3^2 d_5^2 s_2^2 s_4^2 (d_4 s_5 + d_3 s_4 c_5)^2 = 0.$$

Table 4. The robot attitude singularity condition.

Number	Attitude Singularity Satisfying Condition	Note
1	$s_6 = 0 \quad s_2 = 0$	The position singularity is also satisfied.
2	$s_6 = 0 \quad s_4 = 0$	
3	$s_6 = 0 \quad d_4 s_5 + d_3 s_4 c_5 = 0$	

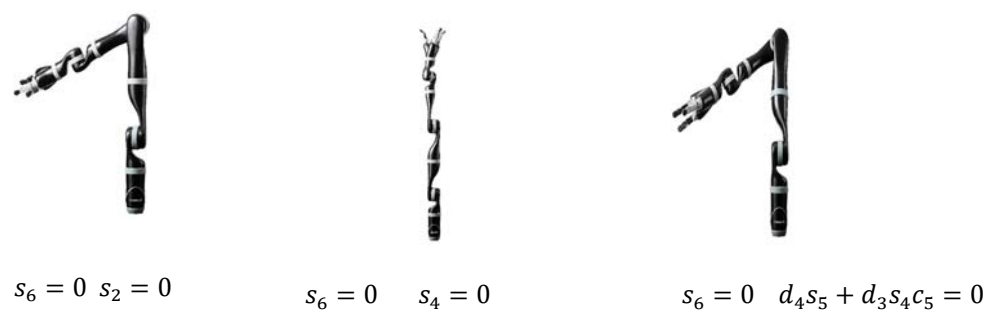


Figure 3. JACO2 wrist singularity.

3. Theoretical Analysis

The main purpose of singularity avoidance is to keep the stable, continuous, and bounded running speed of each joint when the robot is in the singular region. The relationship between joint velocity and end velocity is described as

$$\dot{q} = J^{-1} \dot{x} \tag{13}$$

Since the Jacobian matrix of the JACO2 robot is a nonsquare matrix, (13) is turned into

$$\dot{q} = J^+ \dot{x} + \alpha (I - J^T J^+) \nabla \varphi \tag{14}$$

where $J^+ = J^T(JJ^T)^{-1}$ represents the pseudo-inverse of the Jacobian matrix J , α is coefficient, and $\nabla\varphi$ is an arbitrary vector. In order to ensure that the joint velocity is bounded, it is necessary to change Jacobian matrix J or end running velocity \dot{x} .

3.1. Avoidance Strategy with End Velocity Constant

1. Internal singularity

To keep the end velocity constant, according to (15), only the Jacobian matrix J needs to be changed. The cost function (10) is introduced to obtain the optimal solution. The function is designed:

$$f(\Theta) = \frac{1}{\sqrt{\det(JJ^T)}} \quad (15)$$

where J represents Jacobian matrix corresponding to each group of possible inverse kinematics solutions. Then, $\det(JJ^T)$ will be maximized to avoid singularity. The cost function can be designed as

$$f(\Theta) = \frac{1}{\sigma_r^2} \quad (16)$$

$$f(\Theta) = \frac{\sigma_r^2}{\sigma_1^2} \quad (17)$$

where σ_r and σ_1 represent minimum and maximum singular value, respectively. The convergence direction can be controlled by selecting an appropriate vector $\nabla\varphi$, which controls the gradient of cost function with respect to joint angle Θ . Hence, the equivalent effect of avoiding singularity with the analytical rule is

$$\nabla\varphi = \frac{\partial f(\Theta)}{\partial \Theta} \quad (18)$$

However, considering that all redundant solutions of the JACO2 robot make the manipulator elbow operate on a spatial circle, the above avoidance method can only have a good avoidance effect on the internal singularity and yet have nothing to do with external singularity and pose singularity.

2. Other singular configurations

Other singular configurations cannot be simply avoided by redundant solutions. When the robot elbow is extended, which belongs to external singularity, the robot end is close to the workspace boundary. The robot cannot find the solution far away from the singularity (the fourth joint angle of all solutions approaches π). For six-DOF series industrial robots (most are PUMA type), the method of "singular separation + damping coefficient" adopted in literature [16,21] can obtain good results and analyze the corresponding errors, while it is not suitable for the JACO2 redundant robot, since singular separation effect cannot be realized directly. A damped least squares method is adopted for singular avoidance in this section, which combines the merit of Newton and steepest descent method. It not only ensures the convergence of iterative calculation, but also speeds up the convergence speed.

$$\dot{q} = J^\# \dot{x} \quad (19)$$

where $J^\#$ is the pseudo-inverse of damped Jacobian matrix,

$$J^+ = J^T(JJ^T + \lambda^2 I)^{-1} \quad (20)$$

Then, minimizing the cost function, and regularizing velocity term by coefficient,

$$f(\Theta) = \sum_{i=1}^7 (\theta_i^{t+1} - \theta_i^t)^2 + \sum_{i=1}^7 \lambda^2 (\dot{\theta}_i^{t+1})^2 \quad (21)$$

Then, we regularize of velocity term by coefficient. The minimum singular value of the Jacobian matrix is used as the criterion to judge whether the robot enters the singular region. Equation (20) can be transformed into

$$J = \sum_{i=1}^6 \sigma_i u_i v_i \quad (22)$$

$$J^\# = \sum_{i=1}^6 \frac{\sigma_i}{\sigma_i^2 + \lambda^2} v_i u_i^T \quad (23)$$

$$\dot{\Theta} = J^\# \dot{x} = \sum_{i=1}^6 \frac{\sigma_i}{\sigma_i^2 + \lambda^2} v_i u_i^T \quad (24)$$

$$\text{where } \lambda^2 = \begin{cases} 0, & \sigma_6 > \varepsilon \\ \left(1 - \left(\frac{\sigma_6}{\varepsilon}\right)^2\right) \lambda_m^2, & \sigma_6 \leq \varepsilon \end{cases}$$

3.2. Avoidance Strategy with End Velocity Changed

The external singularity and wrist singularity are mainly discussed here. The avoidance strategy of end speed change is to change the expected end speed in (15). A path parameter p is introduced to describe the trajectory, and end operation trajectory can be expressed as

$$x = x(p) \quad (25)$$

The tracking of desired trajectory is expressed by mathematical expression,

$$h(\tilde{q}) = k(p) - x(p) = 0 \quad (26)$$

where $k(p)$ represents the forward kinematics solution process, \tilde{q} is the set of path parameters p and joint angles, i.e., $\tilde{q} = (q, p)$. The tracking target will satisfy

$$dh(\tilde{q}) = H(\tilde{q})d(\tilde{q}) = 0 \quad (27)$$

$$H(\tilde{q}) = \frac{\partial h(\tilde{q})}{\partial \tilde{q}} = [J(q) \quad -S] \quad (28)$$

where S is a unit vector, which satisfies $dx(p) = dpS$, and $\|dv\| = \|dv(x)\|$ direction, S is the vector of the end running speed, and parameter P is the end running speed. The general solution of (27) can be expressed as

$$d\tilde{q} = bf(\tilde{q}) \quad (29)$$

where b is constant, $f(\tilde{q}) \in \ker H(\tilde{q})$,

$$f(\tilde{q}) = \left[f^T(q) \quad \det J(q) \right]^T \quad (30)$$

where $f(\tilde{q}) = (adjJ(q))S(p)$. Equation (29) can be expanded to

$$dq = bf(\tilde{q}) \quad (31)$$

$$dp = b \det J(q) \quad (32)$$

As long as the trajectory parameters are designed to satisfy (32) and give the joint speed of formula (31), the robot end trajectory can be always followed. It can be found that when the robot approaches the singular configuration, the determinant of the Jacobian matrix

approaches 0, and the given constant and joint velocity both tend to infinity. Therefore, the singular phenomenon can be avoided. The constant b is determined by

$$b^* = \frac{dq_{\max}}{f(\tilde{q})} \quad (33)$$

Reprogramming end velocity based on (32),

$$dp^* = b^* \det J(q) \quad (34)$$

In that the JACO2 robot has redundant joints, (31) is deformed as

$$dq = b_{ep}n_{ep} + b_{sf}n_{sf} \quad (35)$$

$$dp = b_{ep}\det J_3 \quad (36)$$

where b_{ep} and b_{sf} are all constants, J_3 is the Jacobian matrix removing cofactor of column 3, $n_{ep} \in \text{row}J$, $n_{sf} \in \ker J$. However, when internal singularity occurs, the symbol of $\det J_3$ and dp will change, making the expected trajectory reverse, which is equivalent to an obstacle.

$$n_{ep} = \sum_{i=1}^7 b_i n_i \quad (37)$$

$$dp = b_{ep} \sum_{i=1}^7 b_i \det J_i \quad (38)$$

where J_i is the Jacobian matrix removing the cofactor of column i , the i th item of n_i is always 0, then remove n_i item, there is $Jn_i = C_i \det J_i$. The i th item of C_i is 1, and other items are 0.

Let $b_i = \det J_i$, then $dp = b_{ep} \sum_{i=1}^7 b_i \det J_i = b_{ep} \det(JJ^T)$, then the symbol dp will not change.

4. Experimental Results and Comparisons

To demonstrate the performance of the proposed algorithm, a series of simulations and comparisons are conducted. The simulations include a variety of avoidance strategies which basically cover most of the actual situations. Four experimental studies including inverse kinematics solutions and singularity analysis are carried out to test the feasibility of practical application. The symbolic operation toolbox in MATLAB is used for dynamic modeling.

The robot home position is selected as the starting point (joint angle [283.24; 162.71; 0.; 43.58; 265.23; 257.52; 288.14], pose position [0.2113; -0.2656; 0.5065; 1.6477; 1.1081; 0.1282]).

4.1. Two Inverse Kinematics Solutions

Let the robot end run at a speed of 0.1 m/s along the y -axis for 3 s, and control period be 0.1 s. A total of 200 trajectory middle points are inserted. The angles of each joint are calculated by two inverse kinematics solution algorithms. Figure 4 shows the trajectories of two kinematics algorithms. The red points represent the desired points, and blue and green describe trajectories with numerical and analytical solutions, respectively. Their solution errors are shown in Figures 5 and 6.

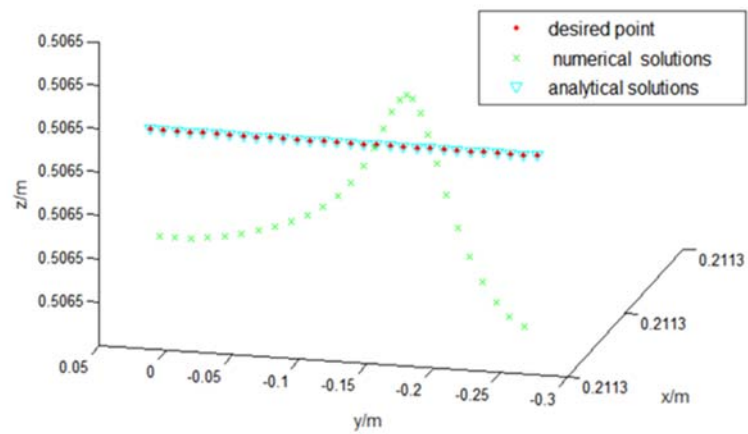


Figure 4. Comparisons of two inverse kinematics solutions.

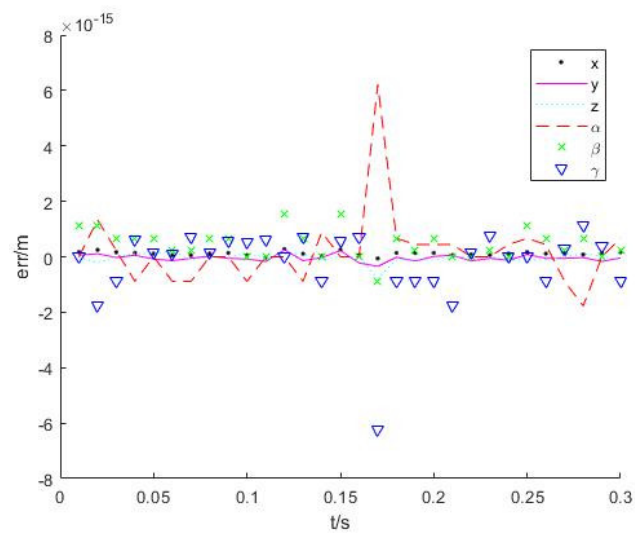


Figure 5. Analytical inverse solution error.

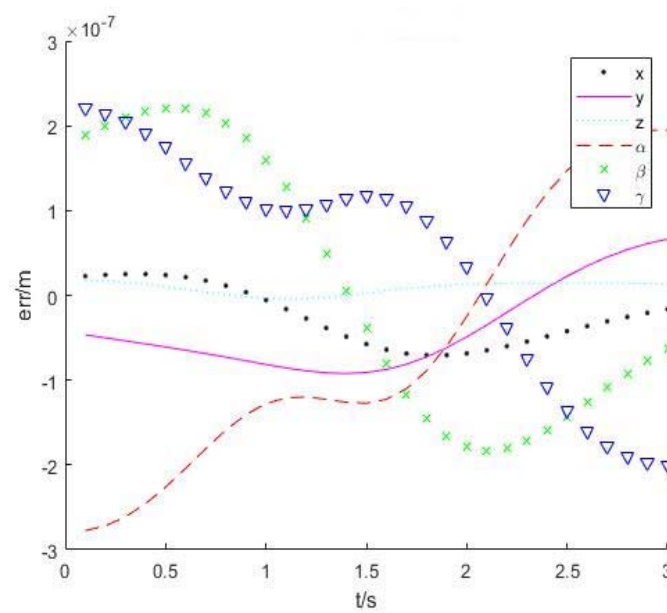


Figure 6. Numerical inverse solution error.

4.2. Singularity Avoidance

To verify the effectiveness proposed in Section 3, singularity avoidance strategies based on robot redundancy, damped least squares, and singularity consistency are realized, respectively. Let the robot end run at the speed of 0.1 m/s along the positive direction along the z-axis for 7.5 s and then turn back, and continue to run along the negative direction along the z-axis for 7.5 s. Set a control cycle as 0.01 s. The joint state and tracking error are plotted with a sampling period of 0.1 s. The compared simulations are as follows.

1. Taking no avoiding measures

Figure 7 shows the position change curve of each joint, Figure 8 shows the velocity variation curve (4.5 s~4.9 s), and Figure 9 shows the tracking errors of position and orientation (4.5 s~4.9 s).

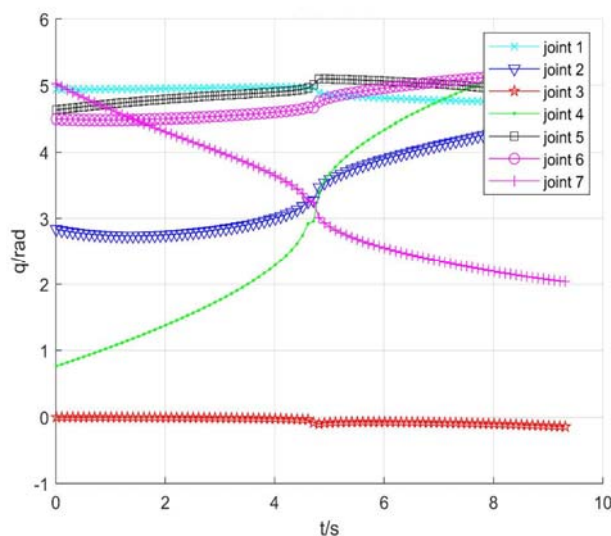


Figure 7. Position of each joint.

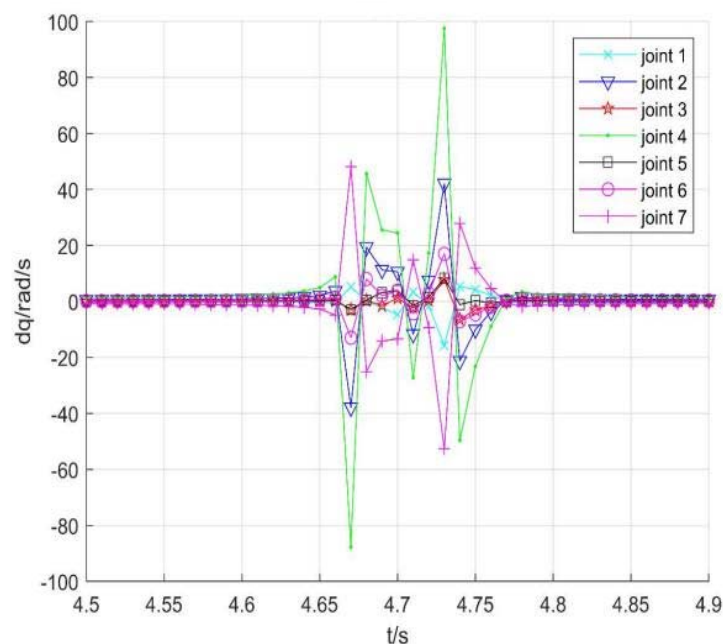


Figure 8. Velocity of each joint.

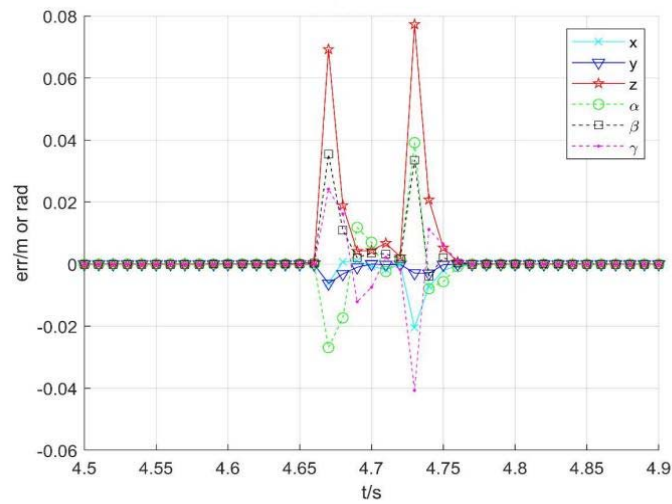


Figure 9. The tracking errors of position and orientation.

2. Using damping Jacobian matrix to avoid singularity

Assume that $\epsilon = 0.1$ is the boundary of the singular region, and $\lambda_m^2 = 0.1$ is the damping coefficient. Figure 10 shows position change curve of each joint, Figure 11 shows the speed change, and Figure 12 shows tracking errors of position and orientation, respectively.

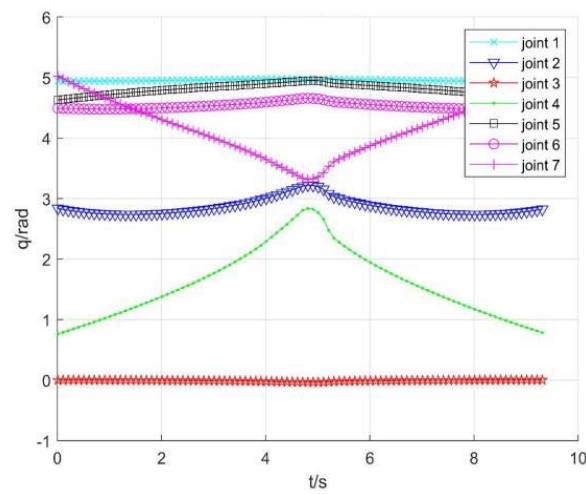


Figure 10. Position of each joint.

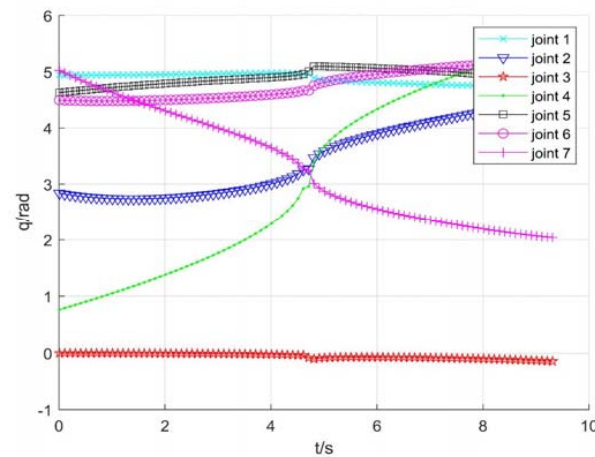


Figure 11. Velocity of each joint.

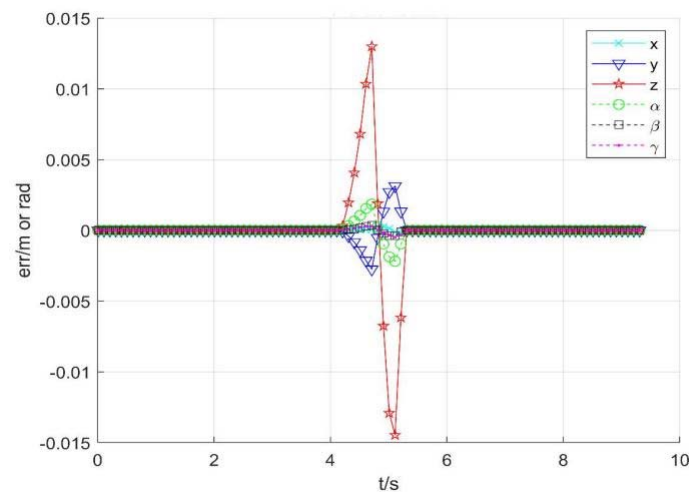


Figure 12. The tracking errors of position and orientation.

3. Using singular consistency to avoid singularity

Figures 13 and 14 show the position and speed change curve of each joint, respectively. Figure 15 shows the running speed of the robot end, Figure 16 shows each item changes of constant b , and Figure 17 shows the tracking errors of position and orientation.

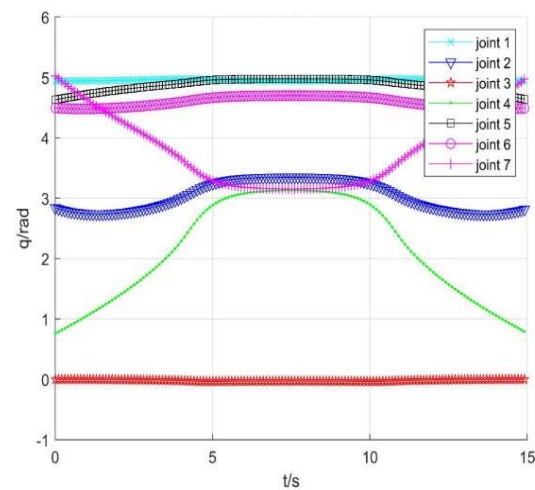


Figure 13. Position change of robot joints.

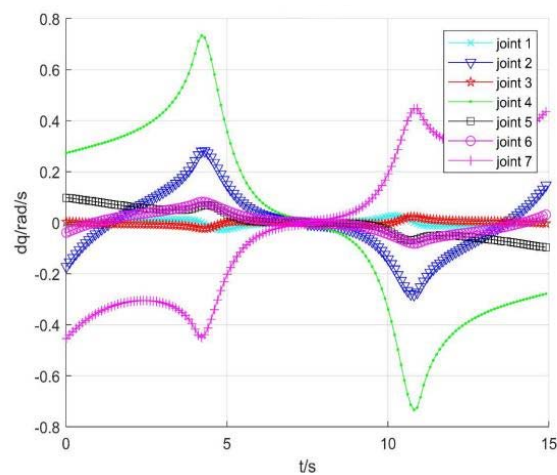


Figure 14. Velocity variation of robot joints.

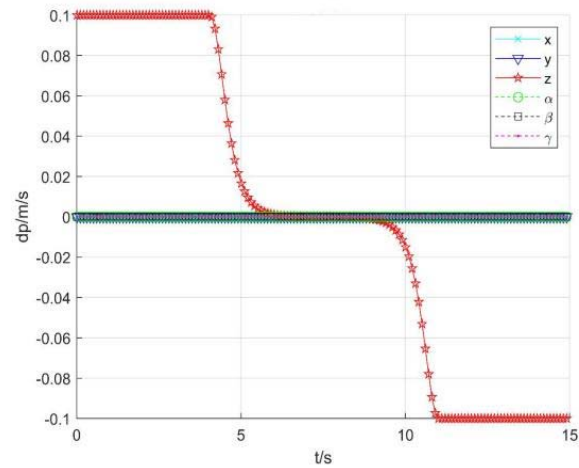


Figure 15. Robot end speed change.

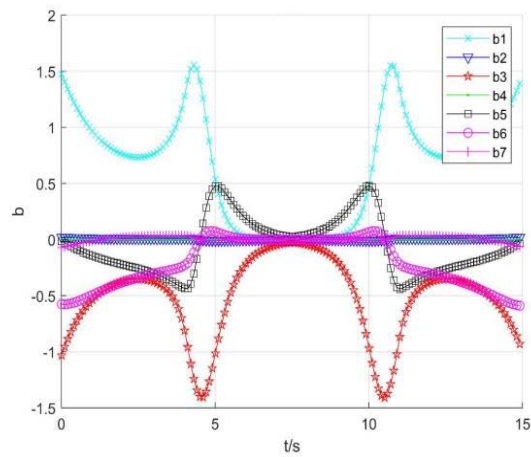


Figure 16. The constant b .

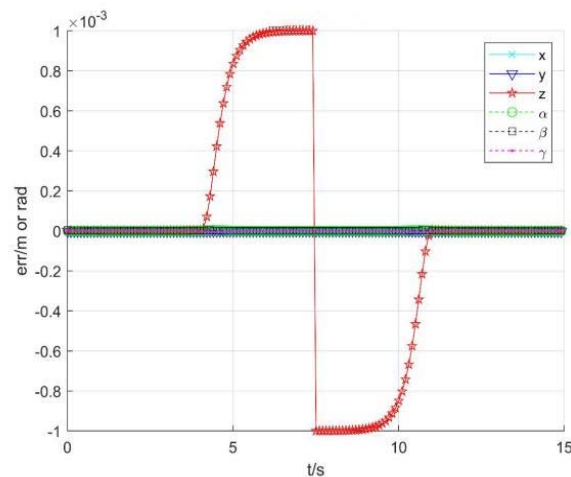


Figure 17. The tracking errors of position and orientation.

5. Results Discussion

From the simulation results in Section 4.1, it is noted that both accuracies can meet the requirement of inverse kinematics. The average calculation time of using the numerical method to calculate the inverse kinematics solution is 0.0021 s, and using the analytical method is 0.0014 s. The calculation time of two methods is in the same order of magnitude.

However, when the robot is in the singular configuration, the numerical method takes three times more time than the numerical method due to more iterations needed.

Figures 7–9 show variation curves of joint position, velocity, and tracking error with time under no avoiding measures. It is obvious that the robot enters the singular region in about 4.65 s. At this time, the angle value of joint 4 is close to π and meets the condition of external singularity. Due to the ill condition of the Jacobian matrix, the obtained joint angle has a sudden change, and the joint speed is particularly huge, which is impossible for the motor to realize in practice. Meanwhile, the robot tracking error end pose becomes larger after entering the singular region, since the upper motor speed cannot be tracked in practical operation.

From Figures 10–12, it can be seen that with damping Jacobian matrix (21), the joint angle changes smoothly, and the whole joint speed can be below 1.5 rad/s. After entering the singular region at about 4.65 s, the pose tracking error will also increase to a certain extent. However, the avoidance method has a certain effect on limiting joint speeds, it is too sensitive to the selection of parameters. Hence, it is necessary to design and select appropriate control parameters for different task requirements.

Figures 13–17 show variation curves of joint position, velocity, end position, and tracking error with singular consistency to avoid singularity. It can be seen that the robot enters the singular region at about 4.1 s. By changing constant value b_i , which makes the following $\det J_i$ smaller, the desired robot end speed dp are forced to change (Figure 15), to limit the speed of each joint (Figure 14). Compared with the singularity avoidance methods in Section 3.1, the joint operation is more stable, and the operation speed of each joint can also transit smoothly when entering into singularity region. Moreover, it will not cause an error deviating from desired trajectory. Therefore, when the robot end velocity is constant, the damping least square avoidance method is selected, while the singular consistency avoidance method is more appropriate when the end velocity changes.

6. Conclusions

The emergence of co-bots is an important supplement to traditional industrial robots, especially for new potential users such as small and medium-sized enterprises and increasingly complex control tasks in 3C electronic products, which determines that co-bots should have characteristics of safe use, simple operation, and convenient deployment. Taking the JACO2 robot as a carrier, this paper provides research on singularity analysis and avoidance strategies to realize safe control and smooth operation of a co-bot in the increasingly complex working environment and man–machine intelligent interactive control. The research results are summarized as follows.

The instability and potential safety hazards of the robot in the singular region are analyzed in detail. The conditions for singularity of the robot are deduced by means of Jacobian matrix separation. Then, the singular configuration conditions are analyzed in detail by means of block analysis of robot wrist Jacobian matrix, which can be divided into three singular types: internal singularity, external singularity, and wrist singularity. For different types of singularity, three singularity avoidance schemes are realized based on robot redundancy and different control tasks. This will ensure safe and stable operation of the co-bot in the whole workspace, and it eliminates safety problems in the man–machine cooperation.

In the future work, it is necessary to study some security issues through other sensors to ensure safety from the perspective of smooth operation, for example, replanning tasks to avoid collision, and adjusting the robot running state according to the working state of the operator. In addition, compliance control can also be studied to improve the robot adaptability with complex industrial tasks.

Author Contributions: Conceptualization, H.W.; Data curation, Z.Z. and X.Z.; Formal analysis, H.W.; Funding acquisition, Q.C.; Investigation, H.W.; Methodology, H.W.; Project administration, H.W. and Q.C.; Resources, H.W., Z.Z. and X.Z.; Software, Z.Z. and X.Z.; Supervision, Q.C.; Validation, H.W.; Visualization, Z.Z.; Writing—original draft, H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by National Natural Science Foundation of China (61733013, 61573260, 61773289, U1713211).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Prusaczyk, P.; Kaczmarek, W.; Panasiuk, J.; Besseghieur, K. Integration of robotic arm and vision system with processing software using TCP/IP protocol in industrial sorting application. *Proc. AIP Conf.* **2019**, *2078*. [\[CrossRef\]](#)
2. Nikolaos, S.; Charalampos, V.; Vassilis, C.M. Experimental verification of optimized anatomies on a serial metamorphic Manipulator. *Sensors* **2022**, *22*, 918. [\[CrossRef\]](#)
3. Jaberzadehansari, R.; Karayiannidis, Y. Task-based role adaptation for human-robot cooperative object handling. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3592–3598. [\[CrossRef\]](#)
4. Golluccio, G.; Gillini, G.; Marino, A.; Antonelli, G. Robot dynamics identification: A reproducible comparison with experiments on the Kinova aco. *IEEE Robot. Autom. Mag.* **2020**, *28*, 128–140. [\[CrossRef\]](#)
5. Sidiropoulos, A.; Karayiannidis, Y.; Dougeri, Z. Human-robot collaborative object transfer using human motion prediction based on cartesian pose dynamic movement primitives. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 3758–3764. [\[CrossRef\]](#)
6. Nakamura, Y.; Hanafusa, H. Inverse kinematic solutions with singularity robustness for robot manipulator control. *J. Dyn. Syst. Meas. Control.* **1986**, *108*, 163–171. [\[CrossRef\]](#)
7. Milenkovic, P. Wrist singularity avoidance with a robot end-effector adding an oblique, redundant axis. *Mech. Mach. Theory* **2021**, *162*, 1–18. [\[CrossRef\]](#)
8. Hamed, K.A.; Kamidi, V.R.; Ma, W.L. Hierarchical and safe motion control for cooperative locomotion of robotic guide dogs and humans: A hybrid systems approach. *IEEE Robot. Autom. Lett.* **2019**, *5*, 56–63. [\[CrossRef\]](#)
9. Bohigas, O.; Zlatanov, D.; Ros, L. A general Method for the Numerical Computation of Manipulator Singularity Sets. *IEEE Trans. Robot.* **2014**, *30*, 340–351. [\[CrossRef\]](#)
10. Müller, A. On the Manifold property of the set of singularities of kinematic mappings: Modeling, Classification, and Genericity. *J. Mech. Robot.* **2012**, *4*, 011006. [\[CrossRef\]](#)
11. Maciejewski, A.A.; Klein, C.A. Numerical filtering for the operation of robotic manipulators through kinematically singular configurations. *J. Robot. Syst.* **1988**, *5*, 527–552. [\[CrossRef\]](#)
12. Chiaverini, S. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Trans. Robot. Autom.* **1997**, *13*, 398–410. [\[CrossRef\]](#)
13. Megalingam, R.K.; Sree, G.S.; Reddy, G.M. Speech and kinova arm-based interactive system with person tracking. In *Computer Networks and Inventive Communication Technologies*; Smys, S., Ed.; Springer: Singapore, 2021; Volume 58, pp. 257–267. [\[CrossRef\]](#)
14. Mou, J. Research on Singular Configuration Analysis and Control Method of Robot. Master Thesis, Northern Jiaotong University, Beijing, China, 2001.
15. Wu, G.; Zhang, G.L.; Yang, F. A singularity avoidance algorithm based on improved damped reciprocal for space robot. *J. Astronaut.* **2016**, *37*, 325–333. [\[CrossRef\]](#)
16. Xu, W.F.; Liang, B.; Liu, Y. A new singularity avoidance algorithm for Puma type robot. *J. Autom.* **2008**, *34*, 6.
17. Wang, X.; Zhang, D.; Zhao, C. Singularity analysis and treatment for a 7R 6-DOF painting robot with non-spherical wrist. *Mech. Mach. Theory* **2018**, *126*, 92–107. [\[CrossRef\]](#)
18. Fang, Y.; Tsai, L. Feasible motion solutions for serial manipulators at singular configurations. *J. Mech. Des.* **2003**, *125*, 61–69. [\[CrossRef\]](#)
19. Dahlin, A.; Karayiannidis, Y. Adaptive trajectory generation under velocity constraints using dynamical movement primitives. *IEEE Control. Syst. Lett.* **2019**, *4*, 438–443. [\[CrossRef\]](#)
20. Taki, S.; Nenchev, D.N. A novel singularity-consistent inverse kinematics decomposition for S-R-S type manipulators. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 5070–5075.
21. Peng, H. Singularity avoidance and vibration suppression in trajectory planning of 6-DOF series robot. Master's Thesis, Harbin Institute of Technology, Harbin, China, 2015.