

Study on Singularity Passing Method for Large Industrial Robot Based on Proper Element Improvement of Inverse Matrix

Ayumu Takeuchi *, Daiki Kato, Masataka Sekioka, Toshiki Hirogaki, and Eiichi Aoyama

Department of Mechanical Engineering, Doshisha University, Kyotanabe-shi, Kyoto, Japan

Email: takeuchi.ayumu5@icloud.com (A.T.); daikidoshisha0106@gmail.com (D.K.); massaa.0811@icloud.com (M.S.); thirogak@mail.doshisha.ac.jp (T.H.); eaoyama@mail.doshisha.ac.jp (E.A.)

*Corresponding author

Abstract—Singularity has become a limitation to the expected diversification of industrial robot applications in production. Here, we propose a novel solution for the inverse kinematics of a 6-DOF manipulator that suppresses rapid joint rotation when passing through a singularity. The proposed method is based on the optimization of the inverse Jacobi cofactor matrix utilized in the inverse kinematics calculation. By optimizing only the elements related to the four and six joints of the cofactor matrix in the inverse matrix, we were able to suppress the divergence of the joint velocities and achieve smooth robot motion when the robot entered a singular posture at the wrist. The developed system application range and tuning gain that minimize the position and orientation error were also determined. Hence, the position and orientation errors were improved by approximately 76%, and rapid rotation of the joints was almost completely suppressed. The method did not require consideration of tools attached to the hand tip, and the hand tip position, orientation, and velocity were all maintained.

Keywords—industrial robot, singularity, cofactor matrix

I. INTRODUCTION

When George C. Devol applied for a patent for an industrial robot arm in 1956 and developed the world's first industrial robot, "Unimate" [1], there was a demand for mass production of a small variety of products. Today, however, variable-mix, variable-volume production is required, and general-purpose machines that can perform tasks according to the situation are needed. In response, teaching methods for robots have shifted from playback to offline teaching [2, 3]. In addition, the role of industrial robots has diversified as smart factories have developed in recent years, and there are high expectations for industrial robots. For example, in additive manufacturing, large-scale AM has been developed employing the flexibility and large workspace of industrial robots [4, 5]. In addition, in piping construction, a novel method was proposed to reproduce the positional relationship between two pipe flanges to be connected utilizing a large industrial robot to

eliminate the gap between the field's measurement stage and the factory's manufacturing stage [6]. However, with such a variety of roles requiring different movements, robots have singular postures; that is, postures in which the robot cannot change the posture of its hand in any direction. Particularly, the 6-DOF serial-link robot, which is the most popular industrial robot, has three singular postures within its range of motion [7], and approaching a singularity causes a sudden rotation of the robot's joints or a robot motion stop, which is an obstacle to trajectory generation. This problem must be solved because the unintended timing of robot motion stoppage reduces the efficiency of the robot motion.

Various methods have been explored to avoid singular postures. The damped least-squares method, which regularizes the Jacobian matrix of a robot, was proposed as one method [8–10]. This method reduces the effect of wrist singularity in nonredundant robots, but introduces position and posture errors. A singular posture-passing algorithm that stretches the Lie algebra was proposed [11]. However, this method cannot guarantee the posture of a robot at singular points and does not apply to motions in which the trajectory of the hand tip is important. A singularity-consistent method was also proposed to accurately track the target trajectory [12]. This method focuses on the inverse Jacobian matrix and sets appropriate joint angular velocities by extracting the relationship between the joint velocities that accurately realize the direction of the tip velocity at a singularity, thereby avoiding excessive joint angular velocities. However, the robot must reach a complete standstill in the vicinity of the singularity; thus, the tip velocity and continuity cannot be maintained. In addition, the operator must command the direction of rotation for each joint, which makes this method impractical. Similar to this method, we also sought a method for passing through singularities by adjusting $\det \mathbf{J}$, which represents the size of the inverse matrix. However, because $\det \mathbf{J}$ affects the angular velocity of all the joints, it is impossible to maintain the hand tip velocity. Recently, a novel method was proposed that considers

tools attached to the hand of the robot [13]. This method extends a nonredundant robot with multiple virtual joints that close the kinematic loop to achieve precise tool contact positions on the workpiece surface. Furthermore, a method was proposed for passing through a singularity by attaching a welding tool with a bent axis to increase redundancy [14]. Although these tooling methods can maintain a robot's hand posture, they are limited by the tools and situations in which they can be utilized. A method that modifies the original path design and does not calculate the inverse kinematics was also proposed, but it has only been put to practical application on two axes [15]. A method using Monte Carlo Method was also proposed to map and define the vicinity of the singularity [16–18]. This method can select a path that is closest to the command path without calculating inverse kinematics. However, this method requires sufficient real data, and it introduces a small amount of position error.

Therefore, here, utilizing the 6-DOF manipulator SRA166-01 from Nachi Corporation, the goal was to allow the robot to pass through singularities while maintaining the posture and velocity of the end-effector. However, wrist singularity can occur anywhere in the working envelope, and staying within reach of the arm and away from the overhead position avoids overhead and elbow singularities of the robot arm [14, 19]. Therefore, we focused on wrist singularity. The proposed method is based on the optimization of the inverse Jacobi cofactor matrix utilized in inverse kinematics calculations. By optimizing only the elements related to four and six joints in the cofactor matrix related to the inverse matrix, the divergence of the joint velocities when the robot enters the singular wrist posture is suppressed, and the robot attempts to pass the singularity while maintaining hand posture and velocity.

II. EXPERIMENTAL EQUIPMENT

A 6-DOF industrial robot, SRA166-01 (manufactured by NACHI), was utilized for the analysis. The main body mass, payload, and positioning accuracy of the robot were 960 kg, 166 kg, and 0.10 mm, respectively. Fig. 1 presents an overview of the robot. The position of the end effector is represented by three-dimensional coordinates $[X, Y, Z]$, and its orientation is represented by a roll-pitch-yaw $[r, p, y]$.

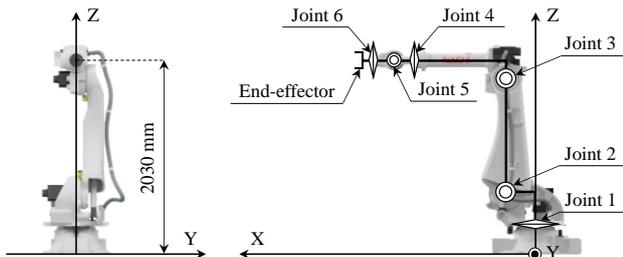


Fig. 1. Robot model.

III. EXPERIMENTAL THEORY

A. Singularity

The forward kinematics are defined in Eq. (1), which determines the position and orientation of the end effector given the joint displacement. where $\theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$ is the displacement of the joint and $p = [X, Y, Z, r, p, y]^T$ is the position and orientation of the end-effector.

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

The Jacobian $J(\theta)$ is defined in Eq. (2). Given the joint speed, we determined the velocity of the end effector.

$$\dot{p} = \frac{\partial f}{\partial \theta_1} \dot{\theta}_1 + \frac{\partial f}{\partial \theta_2} \dot{\theta}_2 + \dots + \frac{\partial f}{\partial \theta_6} \dot{\theta}_6 = J(\theta) \dot{\theta} \quad (2)$$

Inverse kinematics is the problem of determining the displacement of joints, given the position and orientation of the end effector. It plays an important role in controlling end effectors. Inverse kinematics is difficult to solve for a general vertically articulated 6-DOF manipulator because it is a nonlinear system of equations involving inverse trigonometric functions. Therefore, the problem of determining the joint speed, given the end-effector speed defined by Eq. (3) was adopted as a control.

$$\dot{\theta} = J(\theta)^{-1} \dot{p} \quad (3)$$

To get the inverse of the Jacobian, the Jacobian must have full rank, that is, $\det J(\theta) \neq 0$. In contrast, when the Jacobian is singular, which means $\det J(\theta) = 0$, there is no inverse kinematics for velocity. This is called singularity. Fig. 2 illustrates three types of singularities for the analysis target, as presented in Fig. 1.



Fig. 2. Three types of singularity (a) Shoulder singularity (b) Elbow singularity (c) Wrist singularity.

B. Inverse Kinematics and Solution Types

There are eight solution types of inverse kinematics in the manipulator, as illustrated in Fig. 1, which are classified as NONFLIP and FLIP according to the wrist configuration, as depicted in Fig. 3. These are classified depending on the positive or negative Joint 5 angle, with NONFLIP defined as a pose where $\theta_5 > 0^\circ$, and FLIP as a posture where $\theta_5 < 0^\circ$.

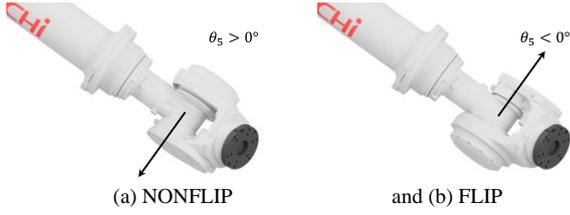


Fig. 3. Two types of inverse kinematics solutions classified by Joint 5 angle.

C. Inverse Jacobian Control

The inverse kinematics of a 6-DOF manipulator are difficult to solve analytically. Obtaining a solution for manipulators with numerous offset links is often impossible. Therefore, the end effector was controlled by an inverse Jacobian. Eq. (4) is a control based on the Newton–Raphson method, where the inverse kinematic solution is the converged value obtained by iterative calculations. where \mathbf{p}_d denotes the command position/orientation of the end effector.

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i + \mathbf{J}(\boldsymbol{\theta}_i)^{-1} \cdot (\mathbf{p}_d - \mathbf{p}) \quad (i = 1, 2, 3, \dots) \quad (4)$$

The inverse Jacobian can be expanded as expressed in Eq. (5) utilizing the determinant $\det \mathbf{J}$ and cofactor matrix $\text{adj} \mathbf{J}$.

$$\mathbf{J}^{-1} = \frac{1}{\det \mathbf{J}} (\text{adj} \mathbf{J}) \quad (5)$$

The $1/\det \mathbf{J}$ is a scalar representing the relationship between the magnitude and direction of the joint speeds. The $\text{adj} \mathbf{J}$ represents the relationship between the joint speeds that realize the direction of the end-effector speeds. At the singularity, the joint speed diverged because $\det \mathbf{J} = 0$. Tsumaki et al. proposed passing through a singularity by changing $1/\det \mathbf{J}$ to an arbitrary scalar value [12]. Because the cofactor matrix is not changed in this method, the joint angles change according to the scalar value while maintaining the ratio of the speeds of each joint angle. This implies that the end effector deviates from the command trajectory at the singularity and slowly changes its position and orientation. This study proposes an algorithm to pass through singularities by appropriately fitting the cofactor matrix to maintain the velocity accuracy of the end effector.

IV. EXPERIMENTAL METHOD

Continuous motion is important in novel fields, such as additive manufacturing, where industrial robots are expected to be applied. Therefore, we consider moving the robot in a straight line without changing its hand posture. As illustrated in Fig. 4, the end-effector position was commanded by linear interpolation from A [1,690, -200, 2,020 mm] to B [1,690, 1,000, 2,020 mm]. The end-effector orientation was always in the Y–Z plane, that was, $[r, p, y] = [-180^\circ, -90^\circ, 0^\circ]$. This path was a trapezoidal velocity trajectory with a maximum velocity of 500 mm/s and maximum acceleration of 2,000 mm/s. The wrist singularity is S [1,690, 0, 2,030 mm], and the end effector passes through the vicinity of the singularity.

Because the robot moves only in the Y-axis direction, the angle of joint 5 is the smallest at Y=0 and is closest to the singularity.

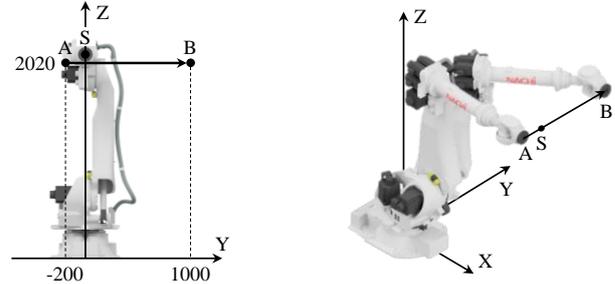


Fig. 4. Linear trajectory of end-effector passing through vicinity of wrist singularity.

Fig. 5 demonstrates the inverse kinematics solution for the end-effector trajectory A→B. Joints 4 and 6 angles in NONFLIP and FLIP are presented in Fig. 5. The robot typically attempts to maintain its wrist form, as illustrated in Fig. 3. Therefore, Joints 4 and 6 turned 180° near the singularity.

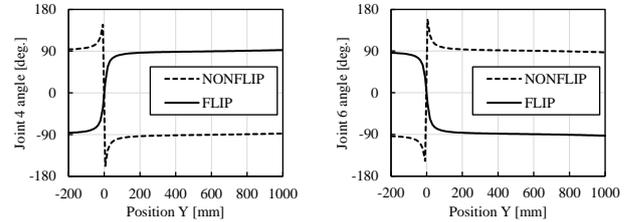


Fig. 5. Inverse kinematics solution of Joints 4 and 6 (a) Joint 4 angle (b) Joint 6 angle

V. EXPERIMENTAL RESULT AND DISCUSSION

A. Trajectory Tracking Error at Singularity

Fig. 6(a) presents Joint 4 angle of the path through the vicinity of the wrist singularity, and Fig. 6(b) illustrates the Y-coordinate of the end effector. From Fig. 6, Joint 4 must change from -90° to 90° when passing through the vicinity of the wrist singularity ($Y = 0$ mm). In the command value, the change is instantaneous. However, in an actual robot, there is an upper limit on the servomotor of the joint, which causes a delay in rotation, resulting in a trajectory tracking error.

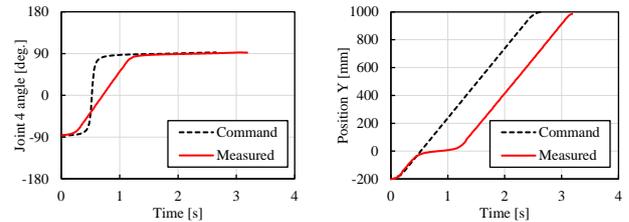


Fig. 6. Trajectory tracking error in the vicinity of the wrist singularity (a) Joint 4 angle (b) Y-coordinate of end effector

B. Fitting Cofactor Matrix

Fig. 7 illustrates a color map of the Jacobian cofactor matrix near the wrist singularity [1,690, 0, 2,020 mm]. Darker colors indicate larger cofactors. We focused on the second row in this experiment because the end effector was operated along the y-axis. Fig. 7 demonstrates that the cofactors related to the y-axis velocities of Joints 4 and 6 were large. Therefore, the singularity was passed by changing the values of a_{42} and a_{62} . where a_{ij} ($i = 1, 2, \dots, 6; j = 1, 2, \dots, 6$) denotes the i -th row and j -th column of the cofactor matrix.

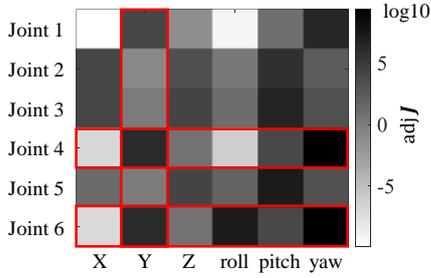


Fig. 7. Colormap of cofactor matrix at wrist singularity.

Because the wrist singularity is $\theta_5 = 0^\circ$, a_{42} and a_{62} are updated, as in Eq. 6 in the range $|\theta_5| = \theta$. where k is the tuning gain and $0 \leq k \leq 1$.

$$\begin{cases} a_{42} \leftarrow ka_{42} \\ a_{62} \leftarrow ka_{62} \end{cases} \quad (6)$$

Fig. 8 illustrates the angles of Joints 4 and 6 with $\theta = 2.5^\circ$ and $k = 0.05$.

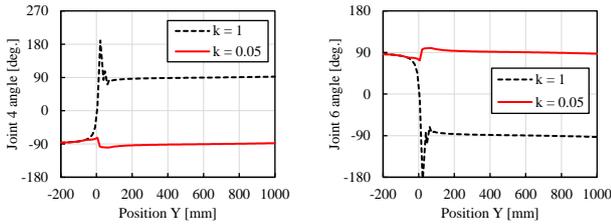


Fig. 8. Joints 4 and 6 angles with fitting cofactor matrix
(a) Joint 4 angle (b) Joint 6 angle.

C. Verification of Errors

Here, we update the cofactors by tuning gain k in the vicinity of a singularity. Therefore, the ranges of k and θ were important for the position and orientation errors of the robot. Therefore, the relationship between the tuning gain k , the range of θ near the singularity, and the error was investigated. The range was set to 0.01 increments for $0 \leq k \leq 1$ and 0.1 increments for $1 \leq \theta \leq 5$ [deg.]. Because the error is maximized near the singular point, we searched for a combination of k and θ for which the maximum error was the smallest. We verified the error in operation for all combinations of k and θ , and calculated the combination of k and θ when the error is the smallest. The position and orientation error norms were calculated with Eq. (7). where ε represents the error and its subscript represents the error direction. The positioning and

orientation error norms were defined as $\varepsilon_{Position_Norm}$ and $\varepsilon_{Orientation_Norm}$, respectively.

$$\begin{cases} \varepsilon_{Position_Norm} = \sqrt{\varepsilon_X^2 + \varepsilon_Y^2 + \varepsilon_Z^2} \\ \varepsilon_{Orientation_Norm} = \sqrt{\varepsilon_r^2 + \varepsilon_p^2 + \varepsilon_y^2} \end{cases} \quad (7)$$

The position and orientation error norms are presented in Figs. 9 and 10, respectively. Both were minimized when $k = 0.03$ and $\theta = 4.6$. The errors were $\varepsilon_{Position_Norm} = 2.08$ mm and $\varepsilon_{Orientation_Norm} = 0.532^\circ$. Without the system, that is, with $k = 1$, the position error norm is 8.78 mm and the orientation error norm is 2.24° ; thus the errors are improved by 76.3% in the position error norm and 76.2% in the orientation error norm, respectively. In addition, as demonstrated in Fig. 9, both the position and orientation errors occurred in the same manner; therefore, the errors were not independent of each other but rather interacted with each other.

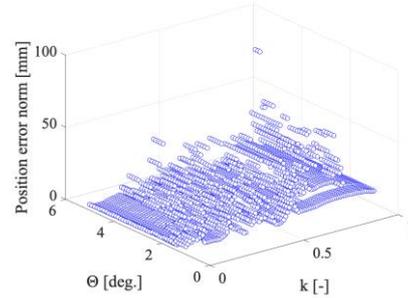


Fig. 9. Influence of k and θ on position error.

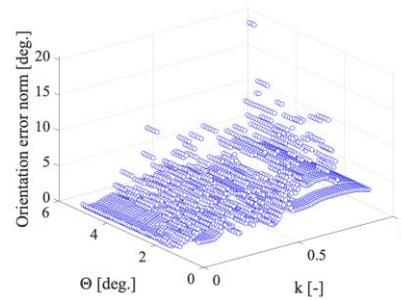


Fig. 10. Influence of k and θ on orientation error.

The rotation of Joints 4 and 6 when utilizing the optimized $k = 0.03$ and $\theta = 4.6$ are presented in Fig. 11. In the figure, $k=1$ is the Command Value (CV), and $k=0.03$ is the Optimized Value (OV). Both joints exhibited almost no rotation.

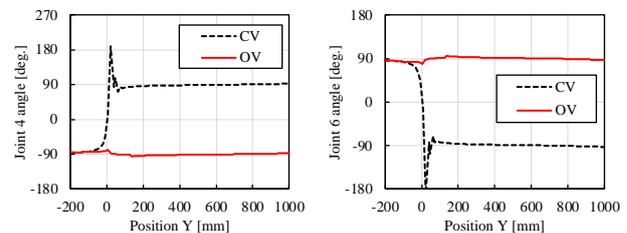


Fig. 11. Joints 4 and 6 angles with optimized cofactor matrix
(a) Joint 4 angle (b) Joint 6 angle.

Fig. 12 illustrates the orientation angles when the robot moved in CV and OV. 180° and -180° may be inverted in roll, but this is not a problem because they are the same wrist posture. In pitch direction, the error was improved by about 53%, with a maximum error of -0.517° . In the yaw direction, however, the error increased by about 83%, but the maximum error was only 0.0456° ; hence, this was not a problem. Fig. 13 presents a three-dimensional view of the robot during operation. Evidently, the error near the singularity is significantly reduced in OV compared with that in CV. Because the robot moves only in the Y-axis direction and the X-axis and Z-axis coordinates are fixed, the errors in these two axes are particularly important. In the X-axis direction, the error was improved by approximately 70.0%, and the maximum error was -0.08 mm. In the Z-axis direction, the error was improved by approximately 52.7%, and the maximum error was -2.04 mm.

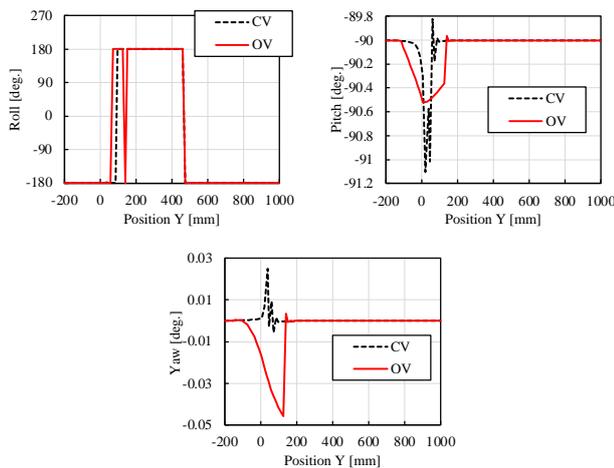


Fig. 12. Orientation angles with optimized value.
(a) Roll (b) Pitch (c) Yaw.

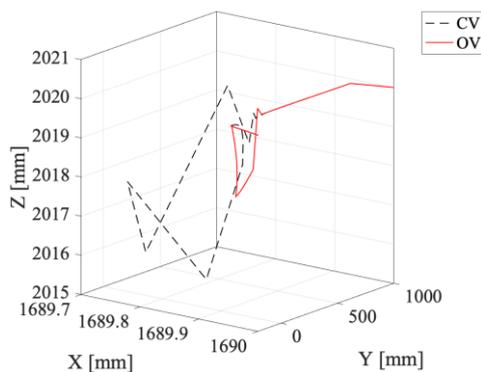


Fig. 13. 3D view of robot's experimental behavior.

Finally, the method described in this report enables the robot to pass near the singularity without considering the tool attached to the hand, whereas previous studies [13, 14] that considered hand tools limited the fields of utilization of robots because the tool's shape was important. In previous studies [8–11] that utilized mathematical methods, the position and posture of a robot's hand tip were significantly affected by the tool's shape, which

caused problems in trajectory tracking. In previous studies [8–11] that utilized mathematical methods, the robot had problems with trajectory tracking owing to a large effect on the position and posture of the robot's hand tip; however, here, the robot was able to move while maintaining its position and posture by optimizing the combination of the range near the singularity and the tuning gain. The singularity-consistent method [12] requires the robot to stop moving once at a singularity and the operator to command the direction of movement; however, this method does not require such a command, and the tip speed can be maintained at a constant level.

One problem with this method is that it is time-consuming because all combinations of the tuning gain and range near the singularity must be calculated before the operation. In the robot utilized here, the axes of Joints 4 and 6 intersected in a straight line. Therefore, when maintaining a constant hand posture, as in this experiment, Joints 4 and 6 rotate by the same amount, and the rapid rotation of the joints is suppressed with almost no postural error. Therefore, it is necessary to determine the tuning gain for robots with an offset structure between Joints 4 and 6.

VI. CONCLUSION

Here, we developed a method that focuses on a cofactor matrix to pass through a singularity that hinders the expected expansion of the application fields of industrial robots. The values for Joints 4 and 6 were large near the singularity, and a method for tuning that element was considered. Because the range in which the system is applied is also important, we calculated a combination of the tuning gain and the range in which the system is applied to minimize the error near the singularity.

Consequently, the error was significantly improved. This method allows passing through a singularity while maintaining the position, attitude, and velocity without considering the hand tools.

In the future, we intend to study robots with more complex structures and systems that can handle more complex movements.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Ayumu Takeuchi summarized experiments and data, and wrote a dissertation; Kato assisted in reviewing experimental results and discussions and revising the paper; Sekioka assisted and advised on the experiments; Hirogaki and Aoyama made a policy decision as an experiment advisor; all authors had approved the final version.

REFERENCES

- [1] O. R. Gerard, "Unimation," *Pillars of Computing*, pp. 219–223, 2015.

- [2] Z. Pan, J. Polden, N. Larkin *et al.*, “Recent progress on programming methods for industrial robots,” *Robotics and Computer-Integrated Manufacturing*, vol. 28, pp. 87–94, 2012.
- [3] S. Mitsi, K. D. Bouzakis, G. Mansour *et al.*, “Off-line programming of an industrial robot for manufacturing,” *The International Journal of Advanced Manufacturing Technology*, vol. 26, pp. 262–267, 2005.
- [4] M. Filip, P. Luka, G. Marko *et al.*, “A Riemannian metric for geometry-aware singularity avoidance by articulated robots,” *Robotics and Autonomous Systems*, vol. 145, pp. 103865, 2021.
- [5] M. Schwicker and N. Nikolov, “Development of a fused deposition modeling system to build form-fit joints using an industrial robot,” *International Journal Mechanical Engineering and Robotics Research*, vol. 11, pp. 51–58, 2022.
- [6] T. Yamauchi, T. Hirogaki, E. Aoyama *et al.*, “In-situ modification of position and posture motion of AN industrial robot to recreate space technology in a pipe joint manufacturing system with the laser trucker sensor,” *International Journal Mechanical Engineering and Robotics Research*, vol. 9, pp. 378–386, 2022.
- [7] L. D. Evjemo, T. Gjerstad, E. I. Grotli *et al.*, “Trends in smart manufacturing: Role of humans and industrial robots in smart factories,” *Current Robotics Reports*, vol. 1, pp. 35–41, 2020.
- [8] C. W. Wampler, “Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, pp. 93–101, 1986.
- [9] S. Chiaverini, B. Siciliano, and O. Egeland, “Review of the damped least-squares inverse kinematics with experiments on an industrial robot manipulator,” *IEEE Transactions on Control Systems Technology*, vol. 2, pp. 123–134, 1994.
- [10] Y. Huang, Y. S. Yong, R. Chiba, J. Ota *et al.*, “Kinematic control with singularity avoidance for teaching-playback robot manipulator system,” *IEEE Transactions on Automation Science Engineering*, vol. 13, pp. 729–742, 2016.
- [11] K. Sugimoto and M. Wakasa, “Algorithm for serial robots passing through the singular point spanning a lie algebra,” *Journal of the Robotics Society of Japan*, vol. 25, pp. 1118–1124, 2007. (in Japanese)
- [12] Y. Tsumaki, S. Kotera, N. D. Nenchev *et al.*, “Singularity-constant teleoperation of a 6-DOF manipulator,” *Journal of Robotics Society of Japan*, vol. 16, pp. 195–204, 1998. (in Japanese)
- [13] P. Milenkovic, “Continuous path for optimal wrist singularity avoidance in a serial robot,” *Mechanism and Machine Theory*, vol. 140, pp. 809–824, 2019.
- [14] P. Milenkovic, “Wrist singularity avoidance with a robot end-effector adding an oblique, redundant axis,” *Mechanism and Machine Theory*, vol. 162, 104355, 2021.
- [15] A. A. Pouyan, H. T. Shandiz, S. Arastehfar *et al.*, “Eliminating redundancy and singularity in robot path planning based on masking,” *Expert Systems with Applications*, vol. 37, pp. 6213–6217, 2010.
- [16] Y. Jianjun, S. Xiaojie, Z. Shiqi *et al.*, “Monte Carlo method for searching functional workspace of an underwater manipulator,” in *Proc. 2018 Chinese Control and Decision Conf.*, 2018.
- [17] A. Elnasr, H. M. Bahaa, and O. Mokhiamar, “Novel use of the Monte-Carlo methods to visualize singularity configurations in serial manipulators,” *Journal of Mechanical Engineering and Sciences*, vol. 15, pp. 7948–7963, 2021.
- [18] T. Stejskal and J. Svetlik, “Mapping robot singularities through the monte carlo method,” *Applied Sciences*, vol. 12, 8330, 2022.
- [19] X. Wang, D. Zhang, C. Zhao *et al.*, “Singularity analysis treatment for a 7R 6-DOF painting robot with non-spherical wrist,” *Mechanism and Machine Theory*, vol. 126, pp. 92–107, 2018.

Copyright © 2024 by the authors. This is an open access article distributed under the Creative Commons Attribution License ([CC BY-NC-ND 4.0](https://creativecommons.org/licenses/by-nc-nd/4.0/)), which permits use, distribution and reproduction in any medium, provided that the article is properly cited, the use is non-commercial and no modifications or adaptations are made.