Collision-Free Motion Planning with Obstacle Avoidance for Line Inspection Robots

Xiangli Lu
College of Mechanical Engineering, Hunan Mechanical & Electrical Polytechnic, China
404939913@qq.com

Wenwei Li
College of Mechanical Engineering, Hunan Mechanical & Electrical Polytechnic, China
274063400@qq.com (corresponding author)

Bin Chen
College of Mechanical Engineering, Hunan Mechanical & Electrical Polytechnic, China
307026228@qq.com

Received: 23 December 2025 | Revised: 17 January 2026, 30 January 2026, 6 February 2026, and 8 February 2026 | Accepted: 11 February 2026

 DOI: https://doi.org/10.48084/etasr.17123

ABSTRACT

This study presents a 7-DOF inspection robot designed for overhead high-voltage transmission line maintenance, with a focus on horizontal twin-bundle conductors. The robot features a dual-arm asymmetric suspension, variable-length manipulators, and center-of-mass adjustment. The main challenge addressed is the collision-free motion planning required for traversing transmission lines and avoiding obstacles. A global motion planning approach, based on a known model environment, is employed to establish a collision detection model. ADAMS simulations are used to determine collision-free path nodes, and inverse kinematics solutions are computed using MATLAB for joint variable values. Smooth trajectory curves are generated using cubic polynomial interpolation and serve as drive functions for the joints in ADAMS. The simulation results are validated with physical prototype experiments, confirming the effectiveness of the motion planning strategy.

Keywords-line inspection robot; simulation; motion planning

I. INTRODUCTION

Economically and effectively maintaining and monitoring power systems has become a critical task in the modern power industry. Therefore, the development of reliable live-working transmission line fault inspection devices holds significant practical engineering value [1, 2]. An overhead high-voltage transmission line inspection robot serves as a reliable live-working fault detection device, enabling cost-effective maintenance and monitoring of power systems. A line inspection robot is a complex mechatronic system that involves multiple disciplines such as mechanical structure, automatic control, communications, sensor information fusion, and power supply technology. However, the mechanical structure of its main body forms the foundation of the entire system[3].

In recent years, extensive research has been conducted on line inspection robots. In [4], a power line inspection robot was developed, which was able to operate in hybrid modes. Equipped with a swingable 2D Laser Rangefinder (LRF), the robot could not only detect obstacles but also identify the position and orientation of Overhead Ground Wires (OGW), making it suitable for automated power line inspections [4]. In [5], a novel walking mechanism was developed and optimized to enhance the operational safety of hybrid PTLIRs under strong wind conditions. Under the influence of wind force 7 (wind speed 15.5 m/s), the robot's average deflection angle around the Z-axis remained within the safety threshold, with a maximum deflection angle of 10.38° perpendicular to the robot's windward surface [5]. In [6], a multi-objective optimization model considered numerous constraints associated with FPTLIR during missions. This model offered a thorough examination of execution time and energy usage. The suggested model's average relative error ranged from 0.76% to 3.24%, accurately predicting FPTLIR energy consumption under various flying conditions and trajectories [6].

Motion planning is a crucial issue in robotics research, with its core task being to plan the motion path of a robot from its initial pose to its target pose [7]. Similar joint-space trajectory generation and software-in-the-loop simulation strategies have been reported for robotic manipulators, highlighting the importance of consistency between simulation and controller implementation [8]. In environments with obstacles, a key goal

of motion planning is to ensure that the robotic arm avoids collisions during operation. This paper focuses primarily on collision-free path planning of a line inspection robot based on its structure [9]. In global motion planning, a collision-free obstacle avoidance path is generated and validated through virtual prototype simulations in software and physical prototype experiments.

Although Unmanned Aerial Vehicles (UAVs) have become widely used for transmission line inspection due to their high mobility and rapid deployment, they remain limited in endurance, payload capacity, stability under electromagnetic interference, and suitability for close-contact or long-duration tasks. Ground-controlled transmission line inspection robots, while proposed earlier, are still well suited for stable conductor-based operation, precise motion near complex fittings, and potential maintenance-oriented applications. Therefore, inspection robots and UAVs should be regarded as complementary technologies. This work does not aim to introduce a new robot concept, but focuses on improving collision-free motion planning and obstacle negotiation capabilities to enhance the practical applicability of line inspection robots in complex transmission line environments.

The novelty of this work lies in a solid model-based collision-free motion planning framework developed for a highly constrained 7-DOF transmission line inspection robot, which directly identifies feasible obstacle-crossing poses in ADAMS and generates joint-space trajectories consistent with actuator constraints, thereby avoiding repeated inverse kinematics during planning and ensuring consistency between simulation and physical prototype control. The novelty of this work does not lie in proposing a new inspection robot concept, but in developing a solid model-based collision-free motion planning framework.

II. STRUCTURE OF THE LINE INSPECTION ROBOT

The travel path of the line inspection robot is equipped with various standardized components such as metal fittings and insulators. The robot must possess sufficient flexibility to traverse these obstacles while avoiding collisions with other phase conductors and obstructions on the lines. This inspection robot adopts a dual-arm asymmetrical suspended structure, and its mechanism schematic is shown in Figure 1.

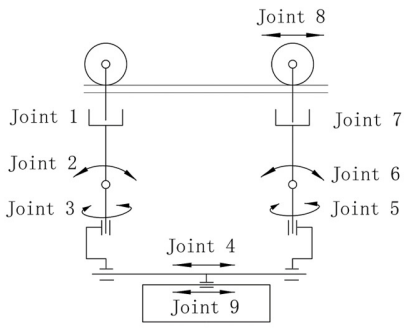


Fig. 1. Simplified structural layout of the line inspection robot.

The robot has a total of 9 joints, among which Joint 8 is a virtual hinge. Joints 1 and 7 are telescopic joints on the two robotic arms, providing vertical movement. Joints 2 and 6 are swing joints in the vertical plane, enabling rotation in that plane. Joints 3 and 5 are rotary joints in the horizontal plane, allowing rotation in that plane. Joint 4 is a horizontal translation joint that provides relative displacement between the two arms (offset motion). Joint 9 is a center-of-mass adjustment joint that regulates the position of the control box to modify the robot's center of gravity. The axes of Joints 1 and 7 are parallel to the axes of Joints 3 and 5, respectively. The axes of Joints 2 and 6 are horizontal and perpendicularly intersect the axes of Joints 3 and 5, respectively. The axis of the translational Joint 4 is horizontal and perpendicularly intersects the axes of Joints 3 and 5.

This configuration enables the following motions: When both arms are suspended, the rolling drive provided by Joint 8 facilitates the robot's overall locomotion. When one arm is fixed to the wire for suspension while the other detaches to traverse obstacles, Joints 1 to 7 provide spatial motion for the obstacle-crossing arm. Joint 4 enables relative displacement between arms (offset motion), and Joint 9 adjusts the center of gravity to improve force distribution and reduce wire deformation. The structural features of the robot include:

1. Capabilities for rolling, crawling, obstacle crossing (avoidance), and posture adjustment in spatial motion, meeting the operational requirements of the working environment.
2. A design with minimal degrees of freedom, incorporating a variable-length arm structure, a dual-arm interactive sliding mechanism, and a center-of-mass adjustment mechanism to traverse obstacles. This approach creates a technical foundation for effectively controlling the robot's weight, energy consumption, and structural dimensions.

III. COLLISION DETECTION MODEL BETWEEN THE LINE INSPECTION ROBOT AND GEOMETRIC OBJECTS

The following section employs the bounding volume representation method to provide a spatial representation of the line inspection robot and obstacles. The establishment method of the collision detection model is described using a simple obstacle (a transmission line) as an example [10]. As shown in Figure 2, the collision detection model is constructed by geometrically abstracting both the obstacle and the end roller of the special arm into enlarged cylinders.

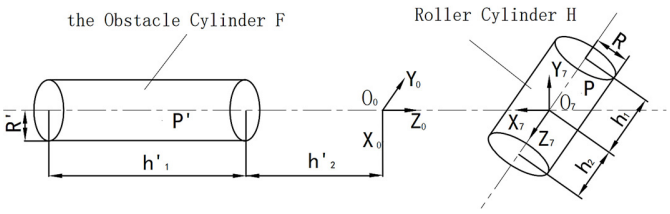


Fig. 2. Collision detection model.

Let the radius of the roller cylinder H be R, and the radius of the obstacle cylinder F be R'. Point P is any point on H, and the homogeneous coordinate of point P in the coordinate system $O_{''}X_{''}Y_{''}Z_{''}$ is $(x_{''} \quad y_{''} \quad z_{''} \quad 1)^T$. The homogeneous coordinate of point P in the coordinate system $O_{'}X_{'}Y_{'}Z_{'}$ is $(x_{'} \quad y_{'} \quad z_{'} \quad 1)^T$. P' is any point on F, and the homogeneous coordinate of point P' in the coordinate system $O_{'}X_{'}Y_{'}Z_{'}$ is $(x'_{'} \quad y'_{'} \quad z'_{'} \quad 1)^T$. The transformation matrix between the coordinate system $O_{''}X_{''}Y_{''}Z_{''}$ and $O_{'}X_{'}Y_{'}Z_{'}$ is ${}_{''}^{'}T$:

$$ {}_{''}^{'}T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

Thus, we have:

$$ (x_{'} \quad y_{'} \quad z_{'} \quad 1)^T = {}_{''}^{'}T \bullet (x_{''} \quad y_{''} \quad z_{''} \quad 1)^T \tag{2} $$

In the coordinate system $O_{''}X_{''}Y_{''}Z_{''}$, the roller cylinder H satisfies the following equation (expressed in cylindrical coordinates):

$$ \begin{cases} x \le R\cos\phi \\ y \le R\sin\phi \qquad (\phi \in [0,2\pi]) \\ -h_1 \le z \le h_2 \end{cases} \tag{3} $$

In the coordinate system $O_{'}X_{'}Y_{'}Z_{'}$, the obstacle cylinder F satisfies the following equation (expressed in cylindrical coordinates):

$$ \begin{cases} x \le R'\cos\phi' \\ y \le R'\sin\phi' \qquad (\phi' \in [0,2\pi]) \\ -(h'_1 + h'_2) \le z \le -h'_2 \end{cases} \tag{4} $$

Therefore, the homogeneous coordinates of point P in coordinate system $(x_{''} \quad y_{''} \quad z_{''} \quad 1)^T$ satisfies (3), and the homogeneous coordinates of point P' in the coordinate system $(x'_{'} \quad y'_{'} \quad z'_{'} \quad 1)^T$ satisfies (4). In the coordinate system $O_{'}X_{'}Y_{'}Z_{'}$, the distance d between points P and P' satisfies:

$$ d^2 = (x_{'} - x'_{'})^2 + (y_{'} - y'_{'})^2 + (z_{'} - z'_{'})^2 \tag{5} $$

From (2), we obtain:

$$ (x_{''} \quad y_{''} \quad z_{''} \quad 1)^T = {}_{''}^{'}T^{-1} \bullet (x_{'} \quad y_{'} \quad z_{'} \quad 1)^T \tag{6} $$

Let

$$ {}_{''}^{'}T^{-1} = \begin{bmatrix} n'_x & o'_x & a'_x & p'_x \\ n'_y & o'_y & a'_y & p'_y \\ n'_z & o'_z & a'_z & p'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7} $$

Then:

$$ \begin{cases} x_{''} = n'_x x_{'} + o'_x y_{'} + a'_x z_{'} + p'_x \\ y_{''} = n'_y x_{'} + o'_y y_{'} + a'_y z_{'} + p'_y \\ z_{''} = n'_z x_{'} + o'_z y_{'} + a'_z z_{'} + p'_z \end{cases} \tag{8} $$

Combining (4-3) and (4-8) yields:

$$ \begin{cases} n'_x x_{'} + o'_x y_{'} + a'_x z_{'} + p'_x \le R\cos\phi \\ n'_y x_{'} + o'_y y_{'} + a'_y z_{'} + p'_y \le R\sin\phi \\ -h_1 \le n'_z x_{'} + o'_z y_{'} + a'_z z_{'} + p'_z \le h_2 \end{cases} \tag{9} $$

Therefore, detecting whether a collision exists between the roller cylinder H and the obstacle cylinder F can be described by the following mathematical model:

For a given $\varepsilon > 0$, determine whether

$$ min\left\{\sqrt{(x_{'} - x'_{'})^2 + (y_{'} - y'_{'})^2 + (z_{'} - z'_{'})^2}\right\} > \varepsilon \tag{10} $$

is valid, where $(x_{'}, y_{'}, z_{'}) \in \Omega_1$, $(x'_{'}, y'_{'}, z'_{'}) \in \Omega_2$

$$ \Omega_1 = \left\{ (x_{'}, y_{'}, z_{'}) \;\middle|\; \begin{cases} n'_x x_{'} + o'_x y_{'} + a'_x z_{'} + p'_x \le R\cos\phi \\ n'_y x_{'} + o'_y y_{'} + a'_y z_{'} + p'_y \le R\sin\phi \\ -h_1 \le n'_z x_{'} + o'_z y_{'} + a'_z z_{'} + p'_z \le h_2 \end{cases} \right\} \tag{11} $$

$$ \Omega_2 = \left\{ (x', y', z') \;\middle|\; \begin{cases} x' \le R'\cos\phi' \\ y' \le R'\sin\phi' \\ -(h'_1 + h'_2) \le z' \le -h'_2 \end{cases} \right\} \tag{12} $$

where $\phi \in [0,2\pi]$.

An enlarged cylinder A is used to encompass the combined obstacle of the suspension insulator string and the straight clamp [11]. An enlarged cylinder B is used to encompass the end effector on the special arm. A parallelepiped C is used to encompass the special arm. Following the method described above, the collision detection model is established between the entire robot and the suspension insulator string.

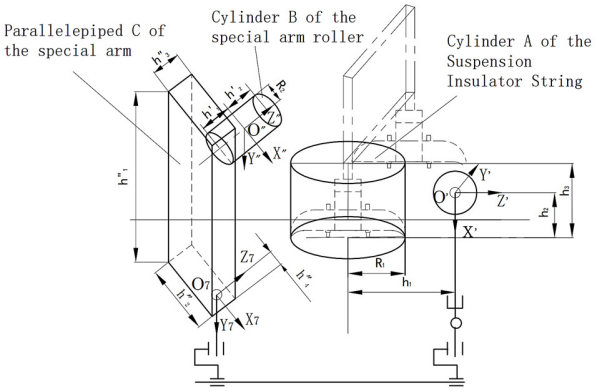


Fig. 3. Collision detection model.

As shown in Figure 3, in the coordinate system $O_{''}X_{''}Y_{''}Z_{''}$, the special arm roller cylinder satisfies:

$$ \begin{cases} x \le R_2\cos\phi' \\ y \le R_2\sin\phi' \quad (\phi' \in [0,2\pi]) \\ -h'_1 \le z \le h'_2 \end{cases} \tag{13} $$

In the coordinate system $O_{'}X_{'}Y_{'}Z_{'}$, the suspension insulator string cylinder A satisfies:

$$ \begin{cases} -(h_3 - h_2) \le x \le h_2 \\ y \le R_1\cos\phi \qquad (\phi \in [0,2\pi]) \\ z + h_1 \le R_1\sin\phi \end{cases} \tag{14} $$

In the coordinate system $O_7X_7Y_7Z_7$, the special arm parallelepiped C satisfies the following equation (expressed in Cartesian coordinates):

$$\begin{cases} -(h''_2 - h''_4) \le x \le h''_4 \\ -h''_1 \le y \le 0 \\ -\frac{1}{2}h''_3 \le z \le \frac{1}{2}h''_3 \end{cases} \quad (15)$$

Then, the collision detection model between the special arm roller and the suspension insulator string is:

For a given $\varepsilon > 0$, determine whether

$$min\left\{\sqrt{(x_{'} - x')^2 + (y_{'} - y')^2 + (z_{'} - z')^2}\right\} > \varepsilon \quad (16)$$

holds true, where $(x_{'}, y_{'}, z_{'}) \in \Omega_1, (x', y', z') \in \Omega_2$

$$\Omega_1 = \left\{(x_{'}, y_{'}, z_{'}) \quad \middle| \begin{cases} -(h_3 - h_2) \le x_{'} \le h_2 \\ y_{'} \le R_1 \cos\phi \\ z_{'} + h_1 \le R_1 \sin\phi \end{cases} \right\} \quad (17)$$

$$\Omega_2 = \left\{(x'_{,}, y'_{,}, z'_{,}) \quad \middle| \begin{cases} (x_{,} \quad y_{,} \quad z_{,} \quad 1)^T = {}_{''}^{'}T \bullet (x_{''} \quad y_{''} \quad z_{''} \quad 1)^T \\ x_{''} \le R_2 \cos\phi' \\ y_{''} \le R_2 \sin\phi' \\ -h'_1 \le z_{''} \le h'_2 \end{cases} \right\} \quad (18)$$

where $\phi \in [0, 2\pi]$.

The collision detection model between the special arm and the suspension insulator string is:

For a given $\varepsilon > 0$, determine whether

$$min\left\{\sqrt{(x_0 - x''_0)^2 + (y_0 - y''_0)^2 + (z_0 - z''_0)^2}\right\} > \varepsilon \quad (19)$$

holds true, where $(x_0, y_0, z_0) \in \Omega_1, (x''_0, y''_0, z''_0) \in \Omega_3$,

$$\Omega_3 = \left\{(x''_0, y''_0, z''_0) \quad \middle| \begin{cases} (x''' \quad y''' \quad z''' \quad 1)^T = {}_{7}^{'}T \bullet (x_7 \quad y_7 \quad z_7 \quad 1)^T \\ -(h''_2 - h''_4) \le x_7 \le h''_4 \\ -h''_1 \le y_7 \le 0 \\ -\frac{1}{2}h''_3 \le z_7 \le \frac{1}{2}h''_3 \end{cases} \right\} \quad (20)$$

The transformation matrices in (18) and (20) can be obtained through forward and inverse kinematic solutions.

Solving the robot's collision-free path points using the above method involves significant computational challenges and complex processes [12]. In particular, for the obstacle negotiation of the line inspection robot discussed in this paper, both the robot and obstacle mechanisms are relatively complex, making it difficult to obtain a collision-free obstacle avoidance path using the aforementioned method. Therefore, this paper adopts an intuitive approach to compare various obstacle-crossing schemes in ADAMS, ultimately selecting a superior obstacle-crossing path. Subsequently, the steps of the joint space planning method suitable for unconstrained environments are applied to derive the final obstacle-crossing path, which can then be validated in ADAMS.

IV. COLLISION-FREE PATH PLANNING BASED ON SOLID SIMULATION MODEL

The fundamental method for collision-free path planning based on the solid simulation model is as follows: First, establish the obstacle (suspension insulator string) and robot solid models in ADAMS/View. Apply collision detection between entities that may potentially collide. According to the action plan, select different motion drive functions to perform kinematic simulations, and compare the simulation results to determine the poses of the robot's collision-free path points (including start and end points). Second, use the inverse solution program written in MATLAB to obtain the corresponding joint values at each path point. Then, fit smooth functions to the values of the joint path points for each respective joint to derive the trajectory for each joint. Finally, import the obtained joint trajectory data from MATLAB into ADAMS to form SPLINE functions. Apply these SPLINE functions as motion drive functions to the joint motions of the line inspection robot's virtual prototype for kinematic simulation verification, simultaneously obtaining a superior collision-free motion path. The correctness of the motion path is validated through physical prototype experiments [13].

A. *Selection of Collision-Free Path Points*

After importing the robot's 3D model into ADAMS, create an abstract cylindrical model of the suspension insulator string within ADAMS. Apply various motion constraints according to the actual situation. Apply entity-entity collision detection between the special (or even) arm roller and the insulator string where collisions might occur. Combined with the robot's action plan, apply motion drives to each joint to perform a kinematic simulation (Simulate/Kinematics). The simulation results of the action plan are shown in Figure 4.

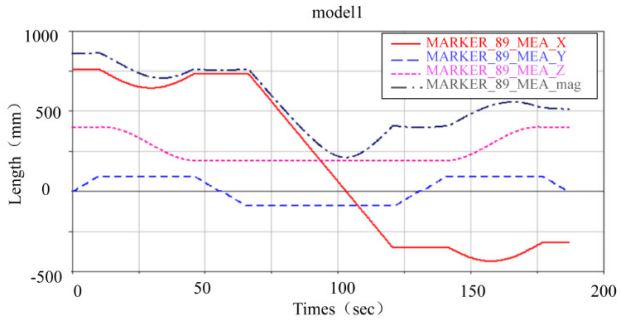


Fig. 4. On the special arm roller during collision-free motion.

Therefore, 10 collision-free path points can be selected according to the planned path. The pose matrices corresponding to each path point in the curve of Figure 4 are denoted $S_0 \sim S_9$:

$$S_0 = \begin{bmatrix} -1 & 0 & 0 & -535 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad S_1 = \begin{bmatrix} -1 & 0 & 0 & -535 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 90 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad S_2 = \begin{bmatrix} 1 & 0 & 0 & -535 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 90 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$S_3=\begin{bmatrix}1&0&0&-535\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\quad S_4=\begin{bmatrix}1&0&0&-535\\0&1&0&0\\0&0&1&-90\\0&0&0&1\end{bmatrix}\quad S_5=\begin{bmatrix}1&0&0&565\\0&1&0&0\\0&0&1&-90\\0&0&0&1\end{bmatrix}$$

$$S_6=\begin{bmatrix}1&0&0&565\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\quad S_7=\begin{bmatrix}1&0&0&565\\0&1&0&0\\0&0&1&90\\0&0&0&1\end{bmatrix}\quad S_8=\begin{bmatrix}-1&0&0&565\\0&-1&0&0\\0&0&1&90\\0&0&0&1\end{bmatrix}$$

$$S_9=\begin{bmatrix}-1&0&0&565\\0&-1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \quad (21)$$

According to the kinematic inverse solution algorithm, the joint variable values corresponding to the pose matrices of each path point can be obtained using the inverse kinematics program written in Matlab, as shown in Table I.

TABLE I. JOINT VARIABLE VALUES OF ATTITUDE MATRIX

Path Point	d_1	θ_2	θ_3	d_4	θ_5	θ_6	d_7
S_0	0	0	–90°	516	–90°	0	0
S_1	0	0	–90°	516	–90°	0	90
S_2	0	0	–90°	516	–270°	0	90
S_3	0	0	–90°	516	–270°	0	0
S_4	–90	0	–90°	516	–270°	0	0
S_5	–90	0	–90°	–584	–270°	0	0
S_6	0	0	–90°	–584	–270°	0	0
S_7	0	0	–90°	–584	–270°	0	90
S_8	0	0	–90°	–584	–90°	0	90
S_9	0	0	–90°	–584	–90°	0	0

B. Joint Trajectory Interpolation Algorithm

During the robot's obstacle avoidance motion, since the joint angle θ_0 corresponding to the starting point is known, and the joint angle θ_f of the endpoint can be obtained through inverse kinematics, the motion trajectory can be described by a smooth interpolation function $\theta(t)$ between the starting point joint angle and the endpoint joint angle[14]. When interpolating the joint trajectory, the trajectory must satisfy a set of constraints, such as requirements for pose, velocity, and acceleration at each node, as well as continuity of joint position, velocity, and acceleration throughout the entire time interval. Under the required constraints, different types of joint interpolation functions can be selected to generate different trajectories. Depending on the chosen joint interpolation function, the main interpolation methods for joint trajectories include cubic polynomial interpolation, cubic polynomial interpolation via path points, high-order polynomial interpolation, linear interpolation with parabolic blends, and linear interpolation via path points with parabolic blends. Among these, the cubic polynomial interpolation method is only suitable for motion where the initial and final joint velocities are zero, which is a special case of the cubic polynomial interpolation via the path points method. High-order polynomial interpolation is applicable when the joint position, velocity, and acceleration are specified at both the start and end points of the path. Considering the control requirements for the line inspection robot's motion, this section adopts the cubic polynomial interpolation via the path points method to interpolate the joints for each action planning unit.

Assume that at the path points, the value and velocity of the joint variable θ are θ_f, respectively, with initial values and velocities being θ_0. The interpolation function is $\theta(t)$,which must satisfy:

$$\begin{cases}\theta(t_0)=\theta_0\\ \theta(t_f)=\theta_f\\ \dot{\theta}(t_0)=\dot{\theta}_0\\ \dot{\theta}(t_f)=\dot{\theta}_f\end{cases} \quad (22)$$

Equation (21) can uniquely determine a cubic polynomial:

$$\theta(t)= a_0+a_1(t-t_0)+a_2(t-t_0)^2+a_3(t-t_0)^3 \quad (23)$$

Then the joint velocity and acceleration on the motion trajectory are given by:

$$\begin{cases}\dot{\theta}(t)=a_1+2a_2(t-t_0)+3a_3(t-t_0)^2\\ \ddot{\theta}(t)=2a_2+6a_3(t-t_0)\end{cases} \quad (24)$$

Substituting (22) and (23) into (24) yields:

$$\begin{cases}\theta_0=a_0\\ \theta_f=a_0+a_1(t_f-t_0)+a_2(t_f-t_0)^2+a_3(t_f-t_0)^3\\ \dot{\theta}_0=a_1\\ \dot{\theta}_f=a_1+2a_2(t_f-t_0)+3a_3(t_f-t_0)^2\end{cases} \quad (25)$$

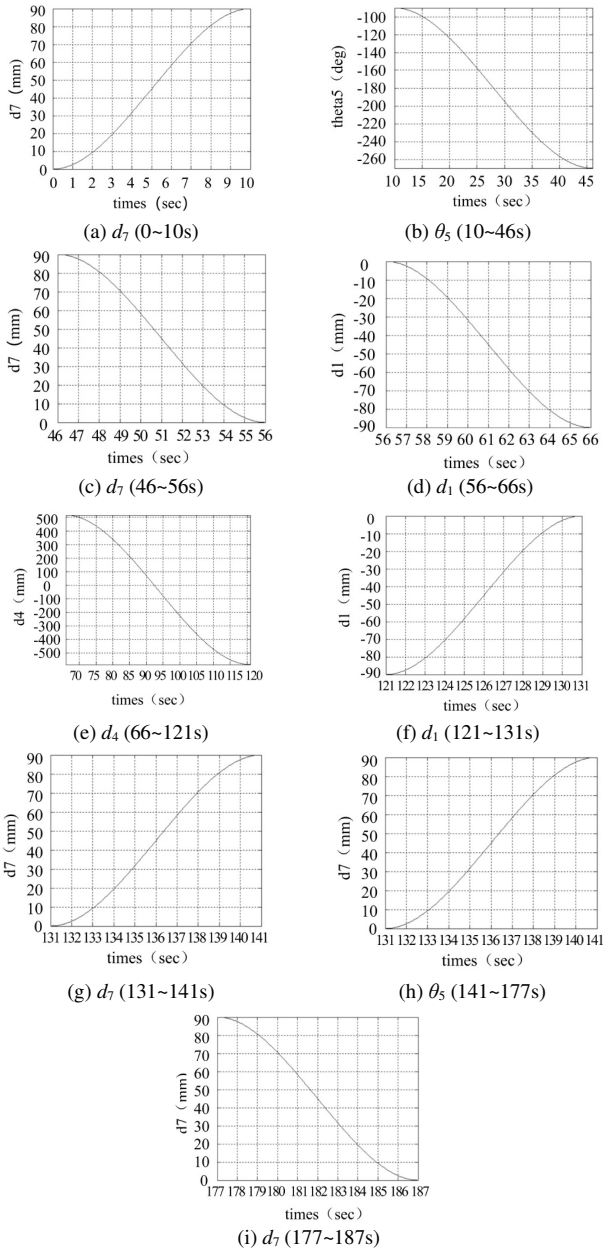
By solving the above system of equations, the coefficients of the cubic polynomial can be obtained as:

$$\begin{cases}a_0=\theta_0\\ a_1=\dot{\theta}_0\\ a_2=\frac{3}{(t_f-t_0)^2}(\theta_f-\theta_0)-\frac{2}{(t_f-t_0)}\dot{\theta}_0-\frac{1}{(t_f-t_0)}\dot{\theta}_f\\ a_3=-\frac{2}{(t_f-t_0)^3}(\theta_f-\theta_0)+\frac{1}{(t_f-t_0)^2}(\dot{\theta}_0+\dot{\theta}_f)\end{cases} \quad (26)$$

Substituting (26) into (23) yields the motion function of the joint variable.

Using the previously mentioned joint trajectory interpolation method, MATLAB programming was used to obtain the joint trajectory curves for each action planning unit corresponding to the plan in Figure 4, as well as the trajectory curves of each joint variable throughout the entire planned path. These are shown in Figure 5 and Figure 6, respectively.

Figure 5(a) represents the variation curve of the joint variable d_7 (driven solely by Joint 7) from waypoint S_0 to S_1; (c) represents from waypoint S_2 to S_3; (g) represents from waypoint S_6 to S_7; (i) represents from waypoint S_8 to S_9. Figure 5(b) represents the variation curve of joint variable θ_5 (driven solely by Joint 5) from waypoint S_1 to S_2; (h) represents from waypoint S_7 to S_8. Figure 5(e) represents the variation curve of joint variable d_4 (driven solely by Joint 4) from waypoint S_4 to S_5. Figure 5(d) represents the variation curve of joint variable d_1 (driven solely by Joint 1) from waypoint S_3 to S_4; (f) represents from waypoint S_5 to S_6.



(a) d_7 (0~10s) (b) θ_5 (10~46s)

(c) d_7 (46~56s) (d) d_1 (56~66s)

(e) d_4 (66~121s) (f) d_1 (121~131s)

(g) d_7 (131~141s) (h) θ_5 (141~177s)

(i) d_7 (177~187s)

Fig. 5. Joint trajectory curves within each action planning unit.

Figure 6 shows the global trajectory curves of each motion joint during the entire process, where the even arm remains fixed, and the special arm avoids the suspension insulator obstacle. These curves provide a basis for subsequent motion simulation and motion control.

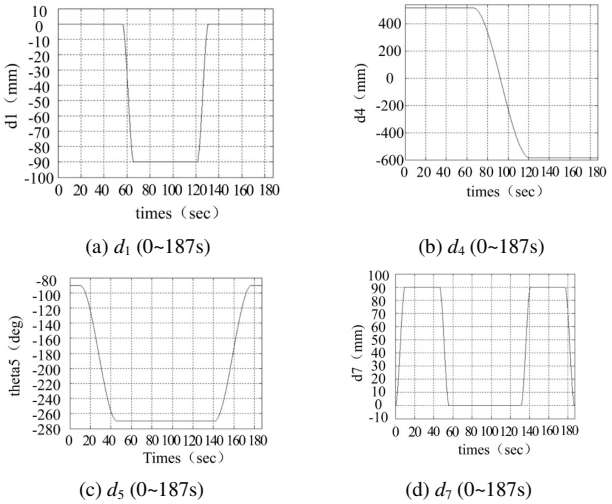
C. Co-Simulation Between MATLAB and ADAMS

The joint trajectory curves obtained in MATLAB (Figure 6) were stored in .txt format. The data of these joint curves were imported into ADAMS to generate SPLINE drive function curves for each joint. Due to differences in the initial position and orientation of the physical model compared to the kinematic model established using the D-H method, modifications were made to the obtained trajectory curves to define the MOTION for each joint.

The drive functions are defined as follows:

- For Fig. 6(a), the drive function applied to Joint 1 is general_motion_8 = –AKISPL(time, 0, SPLINE_1, 0).
- For Fig. 6(b), the drive function applied to Joint 4 is general_motion_4 = AKISPL(time, 0, SPLINE_2, 0) – 516.
- For Fig. 6(c), the drive function applied to Joint 5 is general_motion_3 = AKISPL(time, 0, SPLINE_3, 0) * d + 90d (where d denotes degrees).
- For Fig. 6(d), the drive function applied to Joint 7 is general_motion_1 = –AKISPL(time, 0, SPLINE_4, 0).

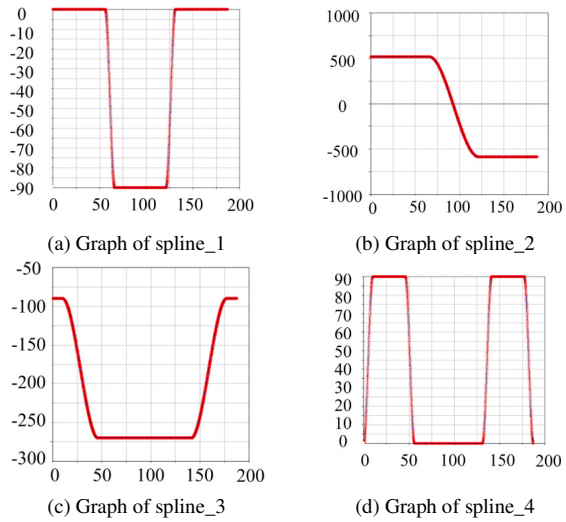
These drive functions are illustrated in Figure 7.



(a) d_1 (0~187s) (b) d_4 (0~187s)

(c) d_5 (0~187s) (d) d_7 (0~187s)

Fig. 6. Trajectory curves of each robot joint in the global path.



(a) Graph of spline_1 (b) Graph of spline_2

(c) Graph of spline_3 (d) Graph of spline_4

Fig. 7. Drive functions for each joint.

In ADAMS, the drive function curves for each joint were applied as their respective motion drives. Entity-to-entity collision detection was implemented between objects where collisions might occur. Kinematic simulations were performed, and the displacement and velocity curves of Marker 80 on the special arm roller were obtained, as shown in Figures 8 and 9, respectively. The results indicate that under the drive functions derived from the cubic polynomial interpolation method, the motion of the line inspection robot remains relatively smooth and stable.

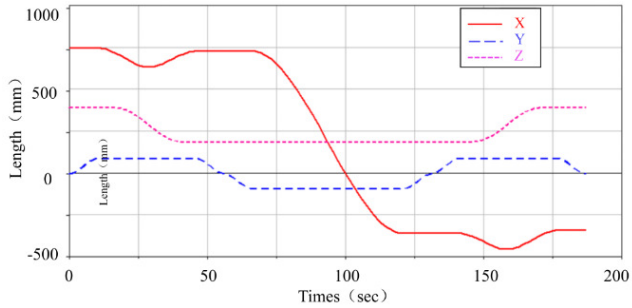


Fig. 8. Displacement curve of marker 80 on the special arm roller in the planned path.

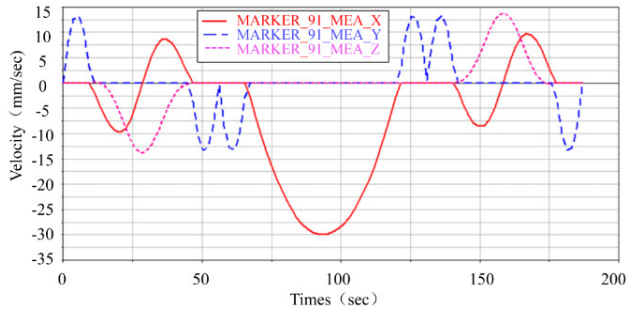


Fig. 9. Velocity curve of marker 80 on the special arm roller in the planned path.

Figure 10 shows a screenshot during the motion planning process of the line inspection robot, where (a) is the abstract cylinder representing the suspension insulator string, (b) is the conductor wire, and (c) is the line inspection robot.

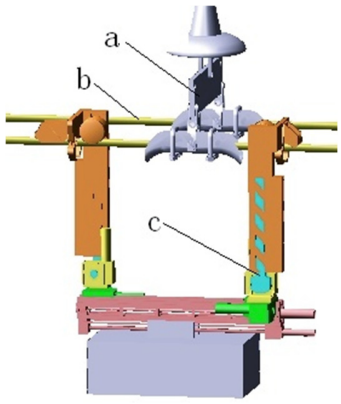


Fig. 10. Path planning simulation diagram.

This planned path passes through the given waypoints, and simulation results confirm that no collisions occur throughout the entire path. The motion is smooth, and the path length is relatively short. Therefore, this planned path is a superior collision-free solution.

D. Physical Prototype Validation

The collision-free obstacle avoidance motion plan obtained above can be validated through physical prototype experiments. A 1:1 simulated test line was constructed in the laboratory to simulate the physical structure and load current of a 220kV horizontal twin-bundle conductor line, as shown in Figure 11. Figure 12 shows partial actions of the physical prototype of the line inspection robot crossing the suspension insulator string. According to the obstacle negotiation path planned in this paper, the physical prototype successfully navigated the obstacle in the experiment. The entire motion process validates the correctness of the motion planning.

Fig. 11. Laboratory simulation of an actual power line.

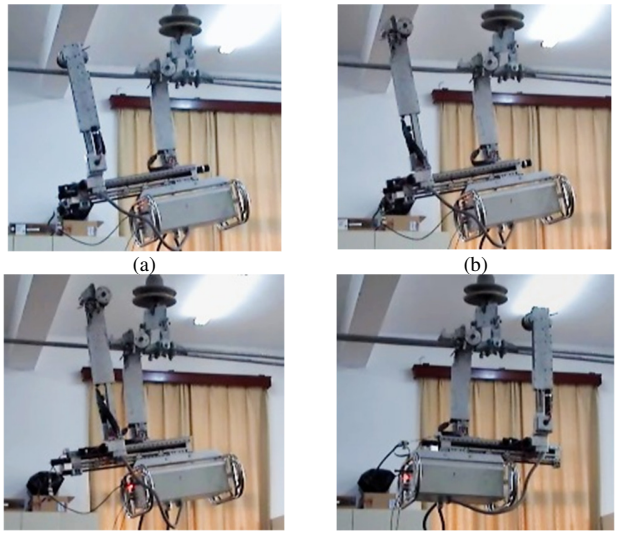


Fig. 12. Physical prototype obstacle negotiation plan. (a) Even arm lifting, (b) even arm rotating 180° counterclockwise, (c) special and even arm offset, (d) even arm rotating 180° counterclockwise.

V. CONCLUSION

This study developed a collision-free obstacle avoidance motion planning method for an overhead transmission line inspection robot by integrating solid model-based collision detection in ADAMS, inverse kinematics computation, and joint-space trajectory interpolation. Simulation and physical

prototype experiments confirmed that the proposed approach can generate smooth and feasible joint trajectories, enabling the robot to safely cross or avoid typical transmission line obstacles, thereby demonstrating its correctness and engineering applicability for autonomous inspection tasks. The method directly considers joint constraints and can be conveniently implemented in both simulation and practical control systems; however, it currently relies on a known and static environment and does not address real-time replanning in dynamic conditions. Future work will focus on incorporating online sensing, improving computational efficiency, and extending the approach to more complex and uncertain operational environments to further enhance the autonomy and robustness of line inspection robots.

REFERENCES

[1] J. Guo, Q. Wu, K. Guo, S. Xiong, W. Feng, and J. Xue, "Study on the Construction and Application of Digital Twins on High Voltage Transmission Line Live Working Scenes," *IEEE Access*, vol. 9, pp. 111587–111594, 2021, https://doi.org/10.1109/ACCESS.2021.3097179.

[2] V. Banthia *et al.*, "A prototype telerobotic platform for live transmission line maintenance: Review of design and development," *Transactions of the Institute of Measurement and Control*, vol. 40, no. 11, pp. 3273–3292, July 2018, https://doi.org/10.1177/0142331216687021.

[3] M. A. Gulzar, K. Kumar, M. A. Javed, and M. Sharif, "High-voltage transmission line inspection robot," in *2018 International Conference on Engineering and Emerging Technologies (ICEET)*, Feb. 2018, pp. 1–7, https://doi.org/10.1109/ICEET1.2018.8338632.

[4] W. Chang, G. Yang, J. Yu, Z. Liang, L. Cheng, and C. Zhou, "Development of a power line inspection robot with hybrid operation modes," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2017, pp. 973–978, https://doi.org/10.1109/IROS.2017.8202263.

[5] P. Jin, X. Qin, J. Lei, Y. Wang, W. Jia, and S. Zhang, "Design and stability performance optimization of a novel hybrid inspection robot walking device for smart grid applications," *Alexandria Engineering Journal*, vol. 109, pp. 970–988, Dec. 2024, https://doi.org/10.1016/j.aej.2024.09.067.

[6] Y. Wang *et al.*, "Multiobjective Energy Consumption Optimization of a Flying–Walking Power Transmission Line Inspection Robot during Flight Missions Using Improved NSGA-II," *Applied Sciences*, vol. 14, no. 4, Feb. 2024, https://doi.org/10.3390/app14041637.

[7] M. G. Tamizi, M. Yaghoubi, and H. Najjaran, "A review of recent trend in motion planning of industrial robots," *International Journal of Intelligent Robotics and Applications*, vol. 7, no. 2, pp. 253–274, June 2023, https://doi.org/10.1007/s41315-023-00274-2.

[8] M. B. Ayed, L. Zouari, and M. Abid, "Software In the Loop Simulation for Robot Manipulators," *Engineering, Technology & Applied Science Research*, vol. 7, no. 5, pp. 2017–2021, Oct. 2017, https://doi.org/10.48084/etasr.1285.

[9] H. Z. Ting, M. H. M. Zaman, M. F. Ibrahim, and A. M. Moubark, "Kinematic Analysis for Trajectory Planning of Open-Source 4-DoF Robot Arm," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, 2021, https://doi.org/10.14569/IJACSA.2021.0120690.

[10] Z. Du, G. Y. Ouyang, J. Xue, and Y. Yao, "A Review on Kinematic, Workspace, Trajectory Planning and Path Planning of Hyper-Redundant manipulators," in *2020 10th Institute of Electrical and Electronics Engineers International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, Oct. 2020, pp. 444–449, https://doi.org/10.1109/CYBER50695.2020.9279171.

[11] P. J. Ryle, "Two transmission-line problems—suspension insulators for industrial areas in Great Britain; conductor vibration," *Journal of the Institution of Electrical Engineers*, vol. 69, no. 415, pp. 805–827, July 1931, https://doi.org/10.1049/jiee-1.1931.0085.

[12] M. Hoy, A. S. Matveev, and A. V. Savkin, "Algorithms for collision-free navigation of mobile robots in complex cluttered environments: a survey," *Robotica*, vol. 33, no. 3, pp. 463–497, Mar. 2015, https://doi.org/10.1017/S0263574714000289.

[13] S. Wei, H. Wu, L. Liu, Y. Zhang, J. Chen, and Q. Li, "A CPG-based gait planning and motion performance analysis for quadruped robot," *Industrial Robot: the international journal of robotics research and application*, vol. 49, no. 4, pp. 779–797, Jan. 2022, https://doi.org/10.1108/IR-08-2021-0181.

[14] S. Tajima, S. Iwamoto, and H. Yoshioka, "Kinematic Tool-Path Smoothing for 6-Axis Industrial Machining Robots," *International Journal of Automation Technology*, vol. 15, no. 5, pp. 621–630, 2021, https://doi.org/10.20965/ijat.2021.p0621.