The obstacle-surmounting analysis of a pole-climbing robot

Du Qiaoling®, Lu Xinpo, Wang Yankai and Liu Sinan

Abstract
Surmounting obstacles during continuously climbing in a complex environment is an important issue for pole-climbing robots. An obstacle-surmounting strategy is presented for a pole-climbing robot. The force and moment applied on the pole-climbing robot in static status were analyzed, and the analysis of pole-climbing robot’s upward vertical climbing was conducted. The climbing execution has four steps: loosening the lower gripper, curling up, striding forward, and clamping the upper gripper. To obtain the information of obstacle crossing accurately, the obstacle-surmounting conditions were analyzed in detail. We modeled the striding linkage with thickness and obtained the Denavit–Hartenberg coordinates of each vertex. The model of the grippers with thickness was proposed and the Denavit–Hartenberg coordinates of each vertex of the grippers were obtained. Then single-step negotiating an obstacle and multistep negotiating an obstacle were proposed. Experiments were conducted to verify the effectiveness of the obstacle-surmounting strategy.

Keywords
Five-bar linkage, pole-climbing robot, obstacle-surmounting, autonomous climbing

Date received: 30 April 2020; accepted: 17 November 2020

Introduction
Pole-climbing robots (PCRobots) can be used to carry out high-risk tasks for human beings in hazardous situations, such as inspection, supervision, and maintenance. Due to their importance in aerial work, PCRobots have been attracting interests of the researchers.

PCRobots have been built and implemented, such as HyDRAS, RiSE V3, Climbot, Shady3D, and BiPCRs. These robots are mainly studied in the respects of structure and climbing gait. Specifically, HyDRAS was a serpentine robot that consisted of a serial chain of actuated universal joints for wrapping around pole-like structures. The methods and considerations for selecting the optimal design parameters (module length, module diameter, helical pitch, and allowable range of motion) were presented for the given task of climbing pole-like structures. RiSE V3 was developed for dynamical, high-speed climbing of uniformly convex cylindrical structures. The climbing machine included novel linkage designs for its legs to propel the body forward during climbing locomotion. Climbot was a biped climbing robot with five joint modules connected in series and two special grippers mounted at the ends. Three climbing locomotions in trusses were proposed based on modular and bioinspired robots. Shady3D had three motive degrees of freedom, which moved along the three-dimensional (3D) truss structure with a fixed section shape. A novel binary approximating method was proposed to compute the graspable region for a BiPCR and

State Key Laboratory on Integrated Optoelectronics, College of Electronic Science and Engineering, Jilin University, Changchun, Jilin, China

Corresponding author:
Du Qiaoling, State Key Laboratory on Integrated Optoelectronics, College of Electronic Science and Engineering, Jilin University, Changchun, Jilin 130012, China.
Email: duq@jlu.edu.cn
to generate the 3D workspace of a manipulator with a constant orientation.\textsuperscript{9,10} A tethered winding-styled soft rod-climbing robot could move on vertical, horizontal, or tilted rods with corners. It consisted of two winding actuators and a telescopic actuator to improve the locomotion efficiency.\textsuperscript{11}

For these robots, the self-continuous climbing is a necessity as the operators may not be able to see the next target to be grasped by the robots all the times. However, how to surmount obstacles for PCRobot during continuously climbing in a complex pole-type environment is a challenging issue.

Dynamic path planning had been presented to generate collision-free strategies for PCRobots moving in obstacle-cluttered environments. Chen et al. designed a mathematic model to make BiPCRs overcome obstacles for transition.\textsuperscript{12} A feasible single-step collision-free climbing mode for BiPCRs was proposed, which could avoid obstacle-crossing gait.\textsuperscript{9} A grip planning strategy was also proposed to generate collision-free grip sequences based on global path planning.\textsuperscript{13} The collision-free path planning method was based on energy optimization. And the collision-free path points were generated following the artificial potential field and sampling searching analysis.\textsuperscript{14} An autonomous obstacle avoidance method was suggested for a robotic manipulator based on an improved RRT algorithm.\textsuperscript{15}

These designs\textsuperscript{13–15} are only applicable to the truss-type environment where there are multiple alternative paths to be selected to avoid obstacles. If there are obstacles that the robots cannot evade, the above methods are not applicable. Some researchers tried to solve this situation by improving the structure or adding additional equipment. Xu and Jiang designed a mechanical structure for a bilateral-wheeled robot to overcome obstacles of semi-surrounding structures.\textsuperscript{16} Lee and Chu developed a three-modular obstacle-climbing robot. The winch mechanism was set to assist the controller to climb obstacles, and the robot controller coordinated the three modules as well as the winch to climb obstacles.\textsuperscript{17} In some cases, it is impossible to add additional equipment to assist obstacle crossing. However, the obstacle can be surmounted by changing the gait of the climbing robot. Therefore, this article focuses on the obstacle-surmounting gait of the PCRobots.

In this article, an obstacle-surmounting strategy was proposed for a PCRobot. An obstacle-surmounting model was firstly built up for computing the enclosed space of five-bar linkages. And the surmountable conditions were proposed. Then single-step negotiating an obstacle and multistep negotiating an obstacle were proposed. Lastly, experiments were conducted with PCRobot to verify the effectiveness of the obstacle-surmounting strategy.

**PCRobot architecture**

**3D model of PCRobot**

The 3D model of PCRobot is shown in Figure 1. It consists of five-bar linkages of $a_0, a_1, a_2, a_3, a_4$, four joints of $j_0, j_1, j_2, j_3$, two supports of $h_1, h_2$, and two grippers of $g_1, g_2$. $j_0$ and $j_3$ are defined as swing joints to realize the relative swing between each two rods. The specific implementation is that one of the two rods is connected with the frame of the steering gear, and the other is connected with the pole. Hence the rotation of the steering gear realizes swing. The functions of the swing joint are similar to that of elbow or knee joints. The functions of $j_0, j_3$ are to adjust the angle of the robot fuselage and grasp, and the functions of $j_1, j_2$ are to adjust the stride distance. The supporting structures of $h_1, h_2$ are formed between the support frame under the gripper and the pole, making PCRobot stable as a whole.

The flipping-over gait of PCRobot is decomposed into two stages: flip and stride, which are described in Qiaoling et al.\textsuperscript{18}

The mechanical structure of PCRobot prototype is shown in Figure 2. Two grippers $g_1$ and $g_2$ are installed at the end of the robot, which are driven by the steering gears $s_0$ and $s_5$ to realize the action of tightening and loosening. $g_1$ and $g_2$ are, respectively, fixed with one end of $a_0$ and $a_4$ with screws, and the other end of $a_0$ and $a_4$ is fixed with
screws to the frame of steering gears $s_1$ and $s_4$, which coordinate the relative motions between $a_0$ and $a_1$ and between $a_4$ and $a_5$, respectively. The steering gears $s_2$ and $s_3$ are connected at both ends of $a_3$. The output ends of $s_2$ and $s_3$ are, respectively, connected with $a_2$ and $a_4$. $b$ is a battery and $c$ is a camera. Lithium-polymer battery parameters are shown in Table 1. The camera is attached on the battery. Its function is to detect the distance between the obstacle and PCRobot. The camera model is XY-SM11.

**Denavit–Hartenberg parameter model**

In Figure 1, $a_0$ and $g_1$, $a_4$ and $g_2$ are fixed connections, which are represented by $l_1$ and $l_2$, respectively. The fixed angle $g_1$ is formed between $l_1$ and the pole due to the support $h_1$. PCRobot rotates $j_0$, $j_1$, $j_2$ to make $j_3$ move to $Q$ to realize the stride action according to the preset step size $d$. Rotation $j_3$ makes $l_2$ reach a certain posture, and then $g_2 (g_2 = g_1)$ between $l_2$ and the pole is formed and then $d$ is parallel to the pole, as shown in Figure 3. In the process, $S$ is regarded as the starting point of the motion as the fixed point, and the control processing is to adjust the angle of $j_0$, $j_1$, and $j_2$ to make $j_3$ reach the target $Q$.

The Denavit–Hartenberg (D-H) coordinate system of PCRobot is established as shown in Figure 3. The coordinate system of $j_0$ is represented by $x_0$–$z_0$. The coordinate systems of $j_1$, $j_2$, and $j_3$ are established in turn, such as $x_1$–$z_1$, $x_2$–$z_2$, and $x_3$–$z_3$. The parameters are shown in Table 2. $\theta_i$ ($i = 1, 2, 3$) are the angles of linkages, $d$ is the distance between $j_0$ and $j_3$, $a_i (i = 1, 2, 3)$ are the lengths of each common perpendicular line, and $\partial$ is the angle between the axis of the joint. $H_1$ and $H_6$ are the intersections of $l_1$ and $l_2$ and pole, respectively.

The kinematics analysis and inverse kinematics solutions of PCRobot are described in Qiaoling et al.18

**Upright and inverted postures**

In the process of continuous climbing, the robot’s posture is alternately “upright posture” and “inverted posture,” as shown in Figure 4. The upright posture is defined as the state of $g_1$ below $g_2$, otherwise defined as an inverted posture.

In the upright posture, the order of the four joints in the counterclockwise is $j_3$, $j_2$, $j_1$, and $j_0$, as shown in Figure 4 “upright posture.” The D-H coordinate system of PCRobot has been derived elsewhere.18 When the robot turns over, its posture changes to the inverted posture. At this time, the order of the four joints is changed to $j_0$, $j_1$, $j_2$, and $j_3$, as shown in Figure 4 “inverted posture.” The D-H coordinate system of PCRobot has been adjusted accordingly. $S'$ is regarded as the starting point of the motion as the fixed point, and the control processing is to adjust the angle of $j_2$, $j_1$, and $j_0$ to make $j_0$ reach the target $Q'$.

---

**Table 1.** Battery parameters.

| Parameter          | Value                  |
|--------------------|------------------------|
| Battery brand      | ACE                    |
| Battery capacity   | 5200 mAh               |
| Cell combination   | 2 s                    |
| Battery voltage    | 7.4 V                  |
| Battery size       | $132 \times 45 \times 25$ mm$^3$ (L × W × H) |
| Battery weight     | 327 g                  |

**Table 2.** Parameters of D-H coordinate system of PCRobot.

| #     | $\theta_i$ | $d$ | $a_i$ | $\partial$ |
|-------|------------|-----|-------|------------|
| 0–1   | $\theta_1$ | 0   | $a_1$ | 0          |
| 1–2   | $\theta_2$ | 0   | $a_2$ | 0          |
| 2–3   | $\theta_3$ | 0   | $a_3$ | 0          |

D-H: Denavit–Hartenberg; PCRobot: pole-climbing robot.
Modeling and analysis of the obstacle-surmounting

Static analysis of PCRobot

When the robot is in the posture shown in Figure 4, the robot is in static status on the pole. Under static situations, the force of the robot is balanced. Figure 5 illustrates the lateral view of the forces and moments applied on the PCRobot. The equilibrium status of the forces applied on the robot meet the following conditions

\[ \Sigma F_{\text{verti}} = F_{m1v} + F_{m2v} + F_{n1v} + F_{n2v} = mg \]  
\[ \Sigma F_{\text{horiz}} = F_{m1h} + F_{m2h} + F_{n1h} + F_{n2h} = 0 \]  
\[ mgd_G = F_{m1r} + F_{m2r} \]  
\[ mg(d_G + r_a) = F_{n1r} + F_{n2r} \]  
\[ F_{m1v} = F_{m1h} + F_{m1l} \]  
\[ F_{m2v} = F_{m2h} + F_{m2l} \]

where \( F_{\text{verti}} \) is the resultant vertical forces, \( F_{\text{horiz}} \) is the resultant horizontal forces, \( F_{m1v} \) is the friction force produced by \( g_1 \), and \( F_{m1r} \) and \( F_{m1l} \) are the frictions produced by the left and right clamps of \( g_1 \). Similarly, \( F_{m2v} \), \( F_{m2r} \), \( F_{m2l} \) are the frictions produced by \( g_2 \); \( F_{n1v} \) and \( F_{n1h} \) are the vertical and horizontal support forces produced by \( h_1 \); \( F_{n2v} \) and \( F_{n2h} \) are the vertical and horizontal support forces produced by \( h_2 \); \( F_{m1h} \) and \( F_{m2h} \) are the horizontal pull provided by the pole; \( m \) is the mass of the robot; \( \mu \) is the static friction coefficient; \( d_G \) presents the horizontal distance from the axis of the pole to the PCRobot center of mass; and \( r_a \) is the radius of the pole.

Since the mechanism of the robot is axisymmetric and the rod is symmetric, the following conclusions are drawn:

\[ F_{m1v} = F_{m2v} = \frac{mgd_G}{2r_a} \]  
\[ F_{n1v} = F_{n2v} = \frac{mg(d_G + r_a)}{2r_a} \]

From equations (1), (7), and (8)

\[ \mu = \frac{r_a}{2d_G + r_a} \]

The static friction coefficient can be expressed as a function of the position of the robot’s mass center and the radius of the pole. When the radius of climbing poles of different materials is fixed, the decrease of friction coefficient means the increase of \( d_G \), so as to keep the robot in equilibrium.

Dynamic analysis of PCRobot

The joints of PCRobot are driven by the steering gears. In the process of vertical climbing, the performance of the steering gears determines the dynamic performance of PCRobot. The weight of the robot determines the gravity of the robot, which is a key factor in the design of climbing robot. The prototype of PCRobot weighed 1.07 kg. According to the force analysis results, TBS2701 steering gear was adopted. The parameters of the TBS2701 are listed in Table 3. The TBS2701 has a maximum rotation angle
270°, in which \( j_0, j_1, j_2, j_3 \) can be conveniently located at any position.

For upward vertical climbing, the climbing execution is divided into four steps: loosening the lower gripper, curling up, striding forward, and clamping the upper gripper. Take the reversal process from the inverted posture to the upright posture as an example, the climbing control process is as follows: the steering gear \( s_0 \) rotates to loosen the lower gripper \( g_1 \); the steering gears \( s_1, s_2, s_3, \) and \( s_4 \) rotate to curl up at a fixed angle; the steering gears \( s_1, s_2, s_3, \) and \( s_4 \) rotate to stride forward according to the calculated angle (see Qiaoling et al.\(^{18} \) for details), and the steering gear \( s_0 \) rotates to clamping the upper gripper \( g_1 \).

A serial control is adopted for the execution of four steps. The time \( t \) to complete a climb is

\[
t = t_l + t_{cu} + t_s + t_{cl}
\]

where \( t_l \) is the time to complete the loosening, \( t_{cu} \) is the time to complete the curling up, \( t_s \) is the time to complete striding forward, and \( t_{cl} \) is the time to complete the clamping.

The servo system of steering gears is controlled by PWM. The reference signal period of the steering gear is 20 ms, and the pulse width is 1.5 ms. Each rotation time is fixed. The steering speed of the steering gears depends on the difference between the starting angle and the target angle. The greater the difference, the faster the steering gears turn. As approaching the target angle, the rotation slows down. In the process of control, if the speed of the steering gears is too fast, the steering gears are easy to shake and stall. Therefore, the following methods are used to reduce the speed of the steering gears: when the steering gears rotate at a large angle, the program sets it to several small angle progressions and decelerates by delaying the program in the progressive process.

The experimental results show that the minimum controlling time of the steering gears is 2 s. In the loosening and clamping steps, only \( s_0 \) works, so set \( t_l = 2, t_{cl} = 2 \). In the process of curling and striding, \( s_1, s_2, s_3, \) and \( s_4 \) rotate in parallel, so set \( t_{cu} = 2, t_s = 2 \). Then \( t = 8 \). The stride distance of the PCRobot is 8 to 23 cm. Let \( v \) be the climbing speed of the PCRobot. Then \( v \) is 1 to 2.875 cm/s.

### Analysis of obstacle-surmounting conditions

#### Enclosed space of five-bar linkages

The enclosed space \( R \) of five-bar linkages is defined as the closed area enclosed by linkages of the robot and pole. \( R \) is a hexagonal region, and its six vertices are, respectively, \( H_1, H_2, H_3, H_4, H_5, \) and \( H_6 \), as shown in Figure 6(a). The shape of the obstacle on the pole can be arbitrary. The front view of the obstacle can be regarded as a convex polygon or a concave polygon approximately. A concave polygon can be decomposed into convex polygons by connecting concave points with their visible vertices.\(^{14} \) A concave polygon can also be surrounded by a convex polygon \( K \), which has \( K_i \) (\( i = 1, 2, 3, \ldots, m \)) vertices by connecting the external vertices of the concave polygon. Figure 6(b) shows an irregular obstacle surrounded by a convex polygon \( K \). \( K \) is set as a rectangle and only locates on one side of the pole. Its vertices are \( K_1, K_2, K_3, \) and \( K_4 \). \( K \) has no contribution to the obstacle negotiated and will not be illustrated in the following text.

The surmountable conditions are as follows:

**Condition 1:** \( R \) and \( K \) are independent of each other.

**Condition 2:** let \( P(x, y) \in K \), \( P(x, y) \notin R \).

Let \( R_s \) denote the segment set of the outline of \( R \), excluding the segment of pole \((H_1 H_6)\), and \( K_s \) denote the segment set of the outline of \( K \).

\[
R_s = \{H_1 H_2, H_2 H_3, H_3 H_4, H_4 H_5, H_5 H_6\}
\]

\[
K_s = \{K_1 K_2, K_2 K_3, \ldots, K_{m-1} K_m, K_m K_1\}
\]

Set \( H_i H_{i+1} \in R_s \) (\( i = 1, 2, \ldots, 5 \)) and \( K_i K_{i+1} \in K_s \) (\( i = 1, 2, \ldots, m \)). If \( H_i H_{i+1} \) and \( K_i K_{i+1} \) are not intersected,
then \( R \) and \( K \) are separated from each other, as shown in Figure 7(a) and (b). When the position relationship between the robot and the obstacle is shown in Figure 7(a), i.e. \( K \subset R \), the robot can surmount the obstacle. When the position relationship between the robot and the obstacle is shown in Figure 7(b), the obstacle is outside of the robot and cannot be surmounted.

In fact, if the position relationship between \( K \) and \( R \) satisfies condition 1, as long as any point \( P(x, y) \) of \( K \) is limited within \( R \), Figure 7(b) can be excluded.

Assume the position relationship between \( K \) and \( R \) satisfies condition 1 and \( K_1 \) is selected as \( P(x, y) \). \( K_1 \) is defined on the climbing pole, so condition 2 can be simplified as “whether point \( K_1 \) is located between \( H_1(x_{H_1}, y_{H_1}) \) and \( H_6(x_{H_6}, y_{H_6}) \).” That is to say, whether the \( x \) coordinate of \( K_1 \) meets the following conditions

\[
x_{H_1} < x < x_{H_6}
\]

\[
x_{H_1} = l \times \sin \gamma_1
\]

\[
x_{H_6} = l \times \sin \gamma_1 + d
\]

where \( l \) is the length of \( H_1H_2 \).

If equations (13) to (15) are established, condition 2 is satisfied. If conditions 1 and 2 are satisfied, then \( K \subset R \). The obstacle \( K \) can be crossed by the robot. In the process of robot turning, it is necessary to dynamically judge whether \( K \) and \( R \) meet the surmountable conditions due to the change of \( R \), as shown in Figure 8.

**Influence of linkages thickness on \( R \)**

The actual robot structure has a certain thickness. The effect of thickness on \( R \) needs to be considered. The space
occupied by the thickness of the robot structure should be removed from $R$. $R'$ considering the thickness of linkages can be obtained, as shown by the dotted line in Figure 9. Obviously, $R'$ is smaller than $R$. Condition 1 is modified to be $K \subset R'$ to ensure the robot can surmount the obstacle.

As shown in Figure 6(a), each linkage is described by only one segment, and the two ends of the segment are the joints. The coordinates of the joints can be directly solved by using the D-H model of the robot linkage structure mentioned in Qiaoling et al.\textsuperscript{18} Because of the thickness of the robot structure, the dynamic description of $R_0$ becomes complex. As shown in Figure 10, the thickness area edge of each linkage needs to be described by multiple segments, and the end points of the segments deviate from the joints, which leads to a complex shape of $R'$. Specifically, if the robot is composed of $n$ bar linkages and the $i$-th linkage is surrounded by $m_i$ segment, then the set of edge segments of all bar linkages is $R'_s$, that is

$$R'_s = \{L_{11}, L_{12}, \ldots, L_{1m_1}, \ldots, L_{n1}, L_{n2}, \ldots, L_{nm_n}\}$$

where $L_{nm}$ is the $m_i$-th edge of the $i$-th bar linkage, $i = 1, \ldots, n$. Set $l_{nm} \in R'_s \ (i = 1, 2, \ldots, n)$ and $k_i \in K \ (i = 1, 2, \ldots, m)$. If $l_{nm}$ and $k_i$ are not intersected, the position relationship between $R'_s$ and $K$ satisfies condition 1. Next is to judge whether equations (13) to (15) are satisfied. If $K \subset R'_s$ is established, then the robot can surmount the obstacle $K$.

**Modeling of the striding linkages with thickness**

To obtain the dynamic model of $R'_s$, it is necessary to solve the vertex D-H coordinate of the linkages with thickness. Firstly, we model the striding linkages with thickness and obtain the D-H coordinates of each vertex.

The striding linkages of PCRobot include $a_1, a_2,$ and $a_3$. The main entities of $a_1, a_2, \text{and } a_3$ are shown in Figure 10(a). To simplify the analysis, the contour of the thickness area of the striding linkage is taken as a rectangle, as shown in Figure 10(b). Let $C_{a_1}, D_{a_1}, E_{a_1}, \text{and } F_{a_1}$ represent the four vertices.

As shown in Figure 10(b), $t_{a_1}$ is the thickness of the striding linkage, $h_{a_1}$ is the distance from $B_{a_1}$ to $C_{a_1}D_{a_1}$, and $s_{a_1}$ is the distance from $A_{a_1}$ to $E_{a_1}F_{a_1}$. Set $\alpha_{a_1}$ is the angle between $A_{a_1}E_{a_1}$ and $O_{a_1}O_{a_1}$, and $\beta_{a_1}$ is the angle between $B_{a_1}D_{a_1}$ and $O_{a_1}O_{a_1}$.

$$\alpha_{a_1} = \arctan \frac{t_{a_1}/2}{s_{a_1}}$$ \hspace{1cm} (17)

$$\beta_{a_1} = \arctan \frac{t_{a_1}/2}{h_{a_1}}$$ \hspace{1cm} (18)

Then the length of $A_{a_1}E_{a_1}, A_{a_1}F_{a_1}, B_{a_1}C_{a_1}$, and $B_{a_1}D_{a_1}$ can be obtained

$$l_{a_1} = l_{a_1} = \frac{s_{a_1}}{\cos \alpha_{a_1}}$$ \hspace{1cm} (19)
In this article, the size of the striding linkage is symmetrical \( (a_n = \beta_n) \). It is obvious that \( \alpha_n = \beta_n \).

There are three striding linkages \( (a_1, a_2, a_3) \) in the PCRobot. In the following, \( a_2 \) is taken as an example to analyze the coordinates of the striding linkage, which is the same as the analyses of \( a_1 \) and \( a_3 \). Transformation matrix \( T_B \) from \( x_0-z_0 \) to \( x_2-z_2 \) can be obtained as shown in equation (21), as shown in Figure 11.

\[
M_B = \prod_{i=1}^{n} T_i \tag{21}
\]

where \( T_i \) is the rotation matrix (see Qiaoling et al.\(^{18}\) for detailed calculation).

Assume \( B_{an}C_{an} \) as a virtual linkage, as shown in Figure 10(c). Transformation matrix from \( B_{an} \) to \( C_{an} \) can be described by D-H parameters in Table 4. Then the rotation matrix \( T_C \) is obtained as shown in equation (22)

\[
T_C = \begin{bmatrix}
\cos \beta_{an} & -\sin \beta_{an} & 0 & l_{B_{an}C_{an}} \cos \beta_{an} \\
\sin \beta_{an} & \cos \beta_{an} & 0 & l_{B_{an}C_{an}} \sin \beta_{an} \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \tag{22}
\]

---

**Figure 10.** Model of striding linkages \( (a_1, a_2, a_3) \) with thickness. (a) Main entities of \( a_1, a_2, a_3; \) (b) Approximate rectangle of the striding linkage with thickness; (c) Solution of vertex coordinates by virtual link method.

---

**Figure 11.** \( A \) and \( B \) in D-H coordinate system of PCRobot. D-H: Denavit–Hartenberg; PCRobot: pole-climbing robot.

**Table 4.** D-H parameters of virtual link structure \( A_{B_{an}C_{an}} \).

| \# | \( \theta \) | \( d \) | \( a \) | \( \alpha \) |
|---|---|---|---|---|
| \( \beta_{an} \) | \( \beta_{an} \) | 0 | \( l_{B_{an}C_{an}} \) | 0 |

D-H: Denavit–Hartenberg.
Table 5. D-H parameters of virtual linkage structure $A_{an}B_{an} \cdots B_{na}D_{na}$.

| # | $\theta$ | $d$ | $a$ | $\alpha$ |
|---|---|---|---|---|
| $B_{an}-D_{an}$ | $-\beta_{an}$ | 0 | $l_{ba,de_a}$ | 0 |

D-H: Denavit–Hartenberg.

Set $S_\beta = \sin \beta_{an}$, $C_\beta = \cos \beta_{an}$, then transformation matrix $M_C$ can be described as follows:

$$M_C = M_B \times T_C = \begin{bmatrix} n_1 C_\beta + a_1 S_\beta & a_1 C_\beta - n_1 S_\beta & 0 & p_x + n_1 l_{ba,de_a} C_\beta + a_1 l_{ba,de_a} S_\beta \\ n_1 C_\beta + a_1 S_\beta & a_1 C_\beta - n_1 S_\beta & 0 & p_y + n_1 l_{ba,de_a} C_\beta + a_1 l_{ba,de_a} S_\beta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{23}$$

According to the definition of coordinate system matrix, the origin coordinate $(x_{C_{an}}, y_{C_{an}})$ of coordinate system $x_{C_{an}}-y_{C_{an}}$ is $C_{an}$:

$$\begin{cases} x_{C_{an}} = p_x + n_1 l_{ba,de_a} C_\beta + a_1 l_{ba,de_a} S_\beta \\ y_{C_{an}} = p_y + n_1 l_{ba,de_a} C_\beta + a_1 l_{ba,de_a} S_\beta \end{cases} \tag{24}$$

The above method to solve the coordinate of $C_{an}$ is defined as “virtual linkage method” in this article. This method can also be used to solve the coordinates of $D_{na}$. The D-H parameters transformed from coordinate system $B_{an}$ to coordinate system $D_{na}$ are shown in Table 5. Then $(x_{D_{na}}, y_{D_{na}})$ can be obtained as shown in equation (25):

$$\begin{cases} x_{D_{na}} = p_x + n_2 l_{ba,de_a} C_\beta + a_2 l_{ba,de_a} S_\beta \\ y_{D_{na}} = p_y + n_2 l_{ba,de_a} C_\beta + a_2 l_{ba,de_a} S_\beta \end{cases} \tag{25}$$

The virtual linkage method is also used to solve the coordinates of $E_{an}$ and $F_{an}$, as shown in Figure 10(c). The rotation matrix $T_A$ and $T_E$ can be obtained by D-H parameters in Table 6. According to Table 6, $T_A$ and $T_E$ are obtained as follows:

$$T_A = \begin{bmatrix} -1 & 0 & 0 & -l_{ba,de_a} \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{26}$$

$$T_E = \begin{bmatrix} \cos \alpha_{an} & -\sin \alpha_{an} & 0 & l_{in,de_a} \cos \alpha_{an} \\ \sin \alpha_{an} & \cos \alpha_{an} & 0 & l_{in,de_a} \sin \alpha_{an} \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{27}$$

Due to $\alpha_{an} = \beta_{an}$, transformation matrix $M_E$ can be described as follows:

$$M_E = M_B \times T_A \times T_E = \begin{bmatrix} -n_1 C_\beta - a_1 S_\beta & n_1 S_\beta - a_1 C_\beta & 0 & p_x - n_1 l_{ba,de_a} C_\beta - a_1 l_{ba,de_a} S_\beta \\ -n_1 C_\beta - a_1 S_\beta & n_1 S_\beta - a_1 C_\beta & 0 & p_y - n_1 l_{ba,de_a} C_\beta - a_1 l_{ba,de_a} S_\beta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{28}$$

According to the definition of coordinate system matrix, the origin coordinate $(x_{F_{an}}, y_{F_{an}})$ of coordinate system $x_{F_{an}}-y_{F_{an}}$ is $E_{an}$:

$$\begin{cases} x_{F_{an}} = p_x - n_3 l_{ba,de_a} - n_3 l_{in,de_a} C_\beta - a_3 l_{in,de_a} S_\beta \\ y_{F_{an}} = p_y - n_3 l_{ba,de_a} - n_3 l_{in,de_a} C_\beta - a_3 l_{in,de_a} S_\beta \end{cases} \tag{29}$$

The contour of the striding linkage with thickness can be described as $R'_{s-an}$.
### Modeling of the gripper with thickness

The shape of the robot’s grippers is irregular. One gripper is with a battery, and the others are not. To simplify the analysis, multiple rectangles are combined to describe the contour of the grippers, as shown in Figure 9.

#### Modeling of the gripper with battery

As shown in Figure 12, the gripper with battery is composed of the support, the steering gear, and the battery. The vertices of three regions are solved, respectively.

1. **Vertex coordinates of the support**

   The four vertices of the support are defined as \( C', D', E', \) and \( F' \), as shown in Figure 13. \( A' \) is the joint of the gripper and the linkage, \( t' \) is the distance from \( A' \) to \( E'F' \), and \( l' \) is the distance from \( A' \) to \( C'D' \). Set \( \alpha' \) as the angle between \( A'F' \) and \( o_1o_1' \), and \( \beta' \) as the angle between \( A'C' \) and \( o_1o_1' \). The fixed angle \( \gamma_1 \) is formed between \( l_1 \) and the pole, which is directly measured by the accelerometer on the gripper with battery.

   According to the geometric relationship in Figure 13, equations (32) to (35) can be obtained

\[
\alpha' = \arctan \frac{t'/2}{s'}
\]

(32)

\[
\beta' = \arctan \frac{t'/2}{l'}
\]

(33)

\[
l_{A'E'} = l_{A'F'} = \frac{s'}{\cos \alpha'}
\]

(34)

\[
l_{A'C'} = l_{A'D'} = \frac{h'}{\cos \beta'}
\]

(35)

### Table 7. D-H parameters of virtual link structure

| \#   | \( \theta \) | \( d \) | \( a \) | \( \alpha \) |
|------|------------|--------|--------|--------|
| \( b_a \) \( A_b \) | \( \pi \)    | 0      | \( l_{b_a} \) | 0      |
| \( A_a \) \( F_a \) | \( -\alpha_a \) | 0      | \( l_{a_a} \) | 0      |

D-H: Denavit–Hartenberg

### Figure 12. Model of the gripper with battery: (a) physical structure and (b) outer contour.

The coordinates of \( C_a \), \( D_a \), \( E_a \), and \( F_a \) are shown in equations (24), (25), (29), and (30), respectively.

### Table 8. D-H parameters of virtual linkages in the support

| \#   | \( \theta \) | \( d \) | \( a \) | \( \alpha \) |
|------|------------|--------|--------|--------|
| \( A' \) \( C' \) | \( -\pi/2 + \gamma_1 + \beta \) | 0      | \( l_{C'C} \) | 0      |
| \( A' \) \( D' \) | \( -\pi/2 + \gamma_1 - \beta \) | 0      | \( l_{D'D} \) | 0      |
| \( A' \) \( E' \) | \( \pi/2 + \gamma_1 + \alpha \) | 0      | \( l_{E'E} \) | 0      |
| \( A' \) \( F' \) | \( \pi/2 + \gamma_1 - \alpha \) | 0      | \( l_{F'F} \) | 0      |

D-H: Denavit–Hartenberg.

Set \( A' \) be the pedestal, the matrix expression of the pedestal \( x_0\)–\( z_0 \) is defined as follow

\[
M_0 = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

(36)

The virtual linkage method is used to solve the coordinates of \( C', D', E', \) and \( F' \), as shown in Figure 13. \( A'C', A'D', A'E', \) and \( A'F' \) are defined as virtual links. The ends of the linkages are \( C', D', E', \) and \( F' \), respectively. The coordinate systems of \( C', D', E', \) and \( F' \) are established in turn, such as \( x_{C'}-z_{C'}, x_{D'}-z_{D'}, x_{E'}-z_{E'}, \) and \( x_{F'}-z_{F'} \). The process transformed from \( x_0\)–\( z_0 \) to \( x_{C'}-z_{C'}, x_{D'}-z_{D'}, x_{E'}-z_{E'}, \) and \( x_{F'}-z_{F'} \) can be described by D-H parameters in Table 8. The virtual linkage method is used to solve the coordinates of \( C', D', E', \) and \( F' \), as shown in Figure 13. \( A'C', A'D', A'E', \) and \( A'F' \) are defined as virtual links. The ends of the linkages are \( C', D', E', \) and \( F' \), respectively. The coordinate systems of \( C', D', E', \) and \( F' \) are established in turn, such as \( x_{C'}-z_{C'}, x_{D'}-z_{D'}, x_{E'}-z_{E'}, \) and \( x_{F'}-z_{F'} \). The process transformed from \( x_0\)–\( z_0 \) to \( x_{C'}-z_{C'}, x_{D'}-z_{D'}, x_{E'}-z_{E'}, \) and \( x_{F'}-z_{F'} \) can be described by D-H parameters in Table 8.

\[
\begin{align*}
C'(x_{C'}, y_{C'}, z_{C'}) = D'(x_{D'}, y_{D'}, z_{D'}),
E'(x_{E'}, y_{E'}, z_{E'}),
F'(x_{F'}, y_{F'})
\end{align*}
\]

are solved, respectively.

\[
\begin{align*}
x_{C'} &= l_{A'C'} \sin(\gamma_1 + \beta) \\
y_{C'} &= -l_{A'C'} \cos(\gamma_1 + \beta)
\end{align*}
\]

(37)
The process transformed from $x_0$ to $x_{G_1}$, $x_{G_2}$, $x_{G_3}$, and $x_{G_4}$ can be described by D-H parameters in Table 9.

(2) Vertex coordinates of the steering gear

The four vertices of the steering gears are defined as $G'_1$, $G'_2$, $G'_3$, and $G'_4$, as shown in Figure 14. $p_1$ is the distance from $A'$ to $G'_1$/$G'_2$, $p_2$ is the distance between $G'_2$/$G'_3$ and $o_1o'_1$, and $p_3$ is the thickness of the steering gear. $G'_5$ and $G'_6$ are the intersections of the steering gear and the support. $p_4$ is the distance between $G'_5$/$G'_6$ and $o_1o'_1$.

The virtual linkage method is used to solve the coordinates of $G'_1$, $G'_2$, $G'_3$, $G'_4$, $G'_5$, and $G'_6$. A virtual link $A'$/$G'_1$, $G'_1$/$G'_2$, $G'_2$/$G'_3$, $G'_3$/$G'_4$, $G'_4$/$G'_5$, and $G'_5$/$G'_6$ are defined as virtual links. The ends of the linkages are $G'_1$, $G'_2$, $G'_3$, $G'_4$, $G'_5$, and $G'_6$, respectively. The coordinate systems of $G'_1$, $G'_2$, $G'_3$, $G'_4$, $G'_5$, and $G'_6$ are established in turn, such as $x_{G'_1}$, $x_{G'_2}$, $x_{G'_3}$, $x_{G'_4}$, $x_{G'_5}$, and $x_{G'_6}$. The process transformed from $x_0$ to $x_{G'_1}$, $x_{G'_2}$, $x_{G'_3}$, $x_{G'_4}$, $x_{G'_5}$, and $x_{G'_6}$ can be described by D-H parameters in Table 9.

(3) Vertex coordinates of the battery

The four vertices of the steering gears are defined as $V'_1$, $V'_2$, $V'_3$, and $V'_4$, as shown in Figure 15. $q_1$ is the distance from $A'$ to $V'_1$/$V'_2$, $q_2$ is the distance between $V'_2$/$V'_3$ and $o_1o'_1$, and $q_3$ is the thickness of the battery. $V'_5$ and $V'_6$ are the intersections of the battery and the support. $q_4$ is the distance between $V'_5$/$V'_6$ and $o_1o'_1$.

The virtual linkage method is used to solve the coordinates of $V'_1$, $V'_2$, $V'_3$, $V'_4$, $V'_5$, and $V'_6$. $A'$/$V'_1$, $V'_1$/$V'_2$, $V'_2$/$V'_3$, $V'_3$/$V'_4$, $V'_4$/$V'_5$, and $V'_5$/$V'_6$ are defined as virtual links. The ends of the linkages are $V'_1$, $V'_2$, $V'_3$, $V'_4$, $V'_5$, and $V'_6$, respectively. The coordinate systems of $V'_1$, $V'_2$, $V'_3$, $V'_4$, $V'_5$, and $V'_6$ are established in turn, such as $x_{V'_1}$, $x_{V'_2}$, $x_{V'_3}$, $x_{V'_4}$, $x_{V'_5}$, and $x_{V'_6}$. The process transformed from $x_0$ to $x_{V'_1}$, $x_{V'_2}$, $x_{V'_3}$, $x_{V'_4}$, $x_{V'_5}$, and $x_{V'_6}$ can be described by D-H parameters in Table 10.

Table 9. D-H parameters of virtual link structure in the steering gear.

| #   | $\theta$  | $d$ | $a$ | $\alpha$ |
|-----|------------|-----|-----|-----------|
| A  | $-\pi/2 + \gamma_1$ | 0   | $p_1$ | 0         |
| G$_1$-G$_2$ | $-\pi/2$  | 0   | $p_2$ | 0         |
| G$_2$-G$_3$ | $\pi/2$   | 0   | $p_3$ | 0         |
| G$_3$-G$_4$ | $-\pi/2$  | 0   | $p_4$ | 0         |

D-H: Denavit-Hartenberg.

\[
\begin{align*}
  x_{E'} &= l_{A'D'} \sin(\gamma_1 - \beta) \\
  y_{E'} &= -l_{A'D'} \cos(\gamma_1 - \beta) \\
  x_{E} &= -l_{A'E'} \sin(\gamma_1 + \alpha) \\
  y_{E} &= l_{A'E'} \cos(\gamma_1 + \alpha) \\
  x_{E} &= -l_{A'E'} \sin(\gamma_1 - \alpha) \\
  y_{E} &= l_{A'E'} \cos(\gamma_1 - \alpha)
\end{align*}
\]  

(41)

(42)

(43)

(44)

(45)

(46)
Table 10. D-H parameters of virtual link structure in the battery.

| #   | \( \theta \) | \( d \) | \( a \) | \( \alpha \) |
|-----|-------------|---------|--------|--------|
| A'->V1' | \(-\pi/2+\gamma_1\) | 0       | 0      | 0      |
| V1'/V2' | 0           | \( q_2 \) | 0      | 0      |
| V2'-V3' | \(-\pi/2\)  | 0       | \( q_2 \) | 0      |
| V3'-V4' | \(-\pi/2\)  | 0       | \( q_2 \) | 0      |
| V4'-V6' | \( \pi/2 \) | 0       | \( q_2 \) | 0      |

D-H: Denavit–Hartenberg.

Figure 16. Model of the gripper without battery.

\( V_4' \) (\( x_{V_4'}, y_{V_4'} \)), \( V_5' \) (\( x_{V_5'}, y_{V_5'} \)), and \( V_6' \) (\( x_{V_6'}, y_{V_6'} \)) are shown in equations (47) to (52), respectively.

\[
\begin{aligned}
\begin{cases}
\quad x_{V_1'} = q_1 \sin \gamma_1 \\
\quad y_{V_1'} = -q_1 \cos \gamma_1 
\end{cases}
\end{aligned}
\]

(47)

\[
\begin{aligned}
\begin{cases}
\quad x_{V_2'} = q_1 \sin \gamma_1 + q_2 \cos \gamma_1 \\
\quad y_{V_2'} = -q_1 \cos \gamma_1 + q_2 \sin \gamma_1 
\end{cases}
\end{aligned}
\]

(48)

\[
\begin{aligned}
\begin{cases}
\quad x_{V_3'} = q_1 \sin \gamma_1 + q_2 \cos \gamma_1 + q_3 \cos \gamma_1 \\
\quad y_{V_3'} = -q_1 \cos \gamma_1 + q_2 \sin \gamma_1 + q_3 \sin \gamma_1 
\end{cases}
\end{aligned}
\]

(49)

\[
\begin{aligned}
\begin{cases}
\quad x_{V_4'} = (q_1 + q_3) \sin \gamma_1 \\
\quad y_{V_4'} = -(q_1 + q_3) \cos \gamma_1 
\end{cases}
\end{aligned}
\]

(50)

\[
\begin{aligned}
\begin{cases}
\quad x_{V_5'} = q_1 \sin \gamma_1 + q_4 \cos \gamma_1 \\
\quad y_{V_5'} = -q_1 \cos \gamma_1 + q_4 \sin \gamma_1 
\end{cases}
\end{aligned}
\]

(51)

\[
\begin{aligned}
\begin{cases}
\quad x_{V_6'} = q_1 \sin \gamma_1 + q_4 \cos \gamma_1 + q_5 \sin \gamma_1 \\
\quad y_{V_6'} = -q_1 \cos \gamma_1 + q_4 \sin \gamma_1 - q_5 \cos \gamma_1 
\end{cases}
\end{aligned}
\]

(52)

The coordinates of the vertices of \( g_2 \) are calculated by the method of virtual linkage. These coordinates are used to describe the edge segments of \( g_1 \) as \( R_{s-g_1} \).

\[
R_{s-g_1} = \{ C'D', D'G_0', G_0'G_1', G_1'G_2', G_2'G_3', G_3'E', E'F', F'V_5', V_5'V_2', V_2'V_3', V_3'V_6', V_6'C' \}
\]

(53)

Modeling of \( g_2 \). As shown in Figure 16, \( g_2 \) is composed of the support and the steering gear. The structure of the gripper without battery is consistent with the support and the steering gear of the \( g_2 \).

The virtual linkage method is used to solve the coordinates in Figure 17. Set \( A_1' \) as the pedestal. The four vertices of the support are defined as \( C_1' \), \( D_1' \), \( E_1' \), and \( F_1' \). \( t_1' \) is the thickness of the support, \( s_1' \) is the distance from \( A_1' \) to \( E_1'F_1' \), and \( l_1' \) is the distance from \( A_1' \) to \( C_1'D_1' \). Set \( \alpha_1' \) as the angle between \( A_1'F_1' \) and \( o_{11}o_{11}' \), and \( \beta_1' \) is the angle between \( A_1'C_1' \) and \( o_{11}o_{11}' \). The fixed angle \( \gamma_2 \) (\( \gamma_2 = \gamma_1 \)) is formed between \( l_2 \) and the pole, which is directly measured by the accelerometer on the gripper without battery.

The four vertices of the steering gears are defined as \( G_{11}' \), \( G_{21}' \), \( G_{31}' \), and \( G_{41}' \), as shown in Figure 18. \( p_{11} \) is the distance from \( A_1' \) to \( G_{11}'G_{21}' \), \( p_{21} \) is the distance between \( G_{21}'G_{31}' \) and \( o_{11}o_{11}' \), and \( p_{31} \) is the thickness of the steering gears. \( G_{51}' \) and \( G_{61}' \) are the intersections of the steering gears and the support. \( p_{41} \) is the distance between \( G_{51}'G_{61}' \) and \( o_{11}o_{11}' \).

The solution process described in this section is the same as the one in section vertex coordinates of the support. \( A_1'C_1', A_1'D_1', A_1'E_1', A_1'F_1', A_1'G_1', G_1'G_2', G_2'G_3', G_3'G_4', G_4'G_5', and G_5'G_6' \) are defined as virtual links. The ends of the linkages are \( C_1', D_1', E_1', F_1', G_1', G_2', G_3', G_4', G_5', and G_6' \), which are shown in equations (54) to (63), respectively.
\[
\begin{align*}
\begin{cases}
  x_{C} = l_{A'C'} \sin(\gamma_{1} + \beta) \\
  y_{C} = -l_{A'C'} \cos(\gamma_{1} + \beta)
\end{cases} \\
\begin{cases}
  x_{D} = l_{A'D'} \sin(\gamma_{1} - \beta) \\
  y_{D} = -l_{A'D'} \cos(\gamma_{1} - \beta)
\end{cases} \\
\begin{cases}
  x_{E} = -l_{A'E'} \sin(\gamma_{1} + \alpha) \\
  y_{E} = l_{A'E'} \cos(\gamma_{1} + \alpha)
\end{cases} \\
\begin{cases}
  x_{F} = l_{A'F'} \sin(\gamma_{1} - \alpha) \\
  y_{F} = l_{A'F'} \cos(\gamma_{1} - \alpha)
\end{cases}
\end{align*}
\]

The coordinates of the vertices of \( g_{2} \) are calculated by the method of virtual linkage. These coordinates can be used to describe the edge segments of \( g_{2} \) as \( R'_{5-g_{2}} \)

\[
R'_{5-g_{2}} = \{ C_{1}D_{1}', D_{1}'G_{61}', G_{61}'G_{51}', G_{51}'G_{21}', \ G_{21}'G_{51}', G_{51}'E_{1}', E_{1}'F_{1}', F_{1}'C' \}
\]

\( \text{Analysis on grippers in upright and inverted posture.} \) There are differences in the structure of \( g_{1} \) and \( g_{2} \). \( g_{1} \) is the structure with the battery and \( g_{2} \) is the structure without the battery. When the robot is in the upright posture, \( g_{1} \) is the lower gripper as shown in Figure 19(a). And when the robot turns over, \( g_{2} \) is the lower gripper in inverted position, as shown in Figure 19(b). During the stride locomotion (see Qiaoling et al.\cite{18}), the shape of \( R' \) is constantly changing as shown in Figure 8. In the sections Modeling of the striding linkages with thickness - Modeling of the grippers with thickness, the virtual link method can be used to calculate the vertex coordinates of each part. The virtual linkage method is based on D-H coordinate, so the vertex coordinates obtained can be updated in real time according to the linkage locomotion.

Set \( R'_{\text{up}} \) denote the segment set of the outline of \( R' \) in the upright posture, and \( R'_{\text{in}} \) in the inverted posture.

\[ R'_{\text{up}} = \{ H_{1}V_{6}', V_{6}'V_{1}', V_{1}'V_{2}', V_{2}'V_{3}', V_{3}'F', \ G_{51}'G_{21}', G_{51}'G_{51}'E_{1}', E_{1}'F_{1}', F_{1}'C' \} \]

\[ R'_{\text{in}} = \{ H_{1}'F_{1}', C_{1}F_{a}, C_{a1}F_{a2}, C_{a1}F_{a1}, E_{1}'G_{5}', \ G_{2}'G_{2}', G_{2}'G_{3}', G_{3}'H_{6}' \} \]

\( \text{Decision condition} \)

The internal space enclosed by the fuselage structure is \( R' \) due to the thickness of the robot structure. \( R'_{\text{up}} \) is the segment set of the outline of \( R' \) in the upright posture, and \( R'_{\text{in}} \) is the segment set in the inverted posture. Then the surmountable conditions in Enclosed space of five bar linkage are modified as follows:

1. **Condition 1:** \( R'_{\text{up}} \) and \( K \) are independent of each other in the upright posture, or \( R'_{\text{in}} \) and \( K \) are independent of each other in the upright posture.

2. **Condition 2:** let \( P(x, y) \in K \), \( P(x, y) \in R' \).
Obstacle-surmounting control

Single-step negotiating an obstacle

To facilitate the analysis, $K$ is set as a rectangle. Its vertices are $K_1, K_2, K_3,$ and $K_4,$ as shown in Figure 20. $h_k$ is the height of $K,$ $w_k$ is the width of $K,$ $d_k$ is the length of $H_1K_1,$ $d'_k = w_k \times \tan \gamma_1$ is the length of $H'K_k,$ and $\gamma_1$ is formed between the gripper and the pole.

Set the coordinate systems of $o$ be $x_0y_0.$ Then the coordinates of $K_1, K_2, K_3,$ and $K_4$ are $(x_{H_1} + d_k, y_{H_1}), (x_{H_1} + d_k, y_{H_1} + w_k),$ $(x_{H_1} + d_k + h_k, y_{H_1} + w_k),$ and $(x_{H_1} + d_k + h_k, y_{H_1}),$ respectively, where $x_{H_1} = l \times \sin \gamma_1$ and $y_{H_1} = l \times \cos \gamma_1.$ The set $K_{s-4}$ can be obtained

$$K_{s-4} = \{K_1K_2, K_2K_3, K_3K_4, K_4K_1\} \quad (67)$$

The stride distance $d'$ should satisfy equation (68)

$$d' \geq (d'_k + h_k + d_k) \quad (68)$$

Due to the limitation of robot structure, there are the minimum and the maximum of a stride. Set $d'_{\text{max}}$ be the max of $d'.$ Set $d'_{\text{min}}$ be the minimum of $d'.$ Assume the vertex coordinates of obstacles known in advance. According to the D-H coordinate, whether the boundary of the obstacle and the linkages of the robot meet the conditions 1 and 2 can be judged in real time. If the surmountable conditions are satisfied and the calculated distance $d'$ is less than $d'_{\text{max}},$ the single-step locomotion can be used to negotiate the obstacle.

In the process of negotiating, the vertex coordinates of the linkages are calculated by the virtual link method. The attitude of the robot is adjusted continuously so that the surmountable conditions are satisfied between the linkages of the robot and the boundaries of the obstacle. Finally, the robot strides over the obstacle in one step.

Multistep negotiating an obstacle

In some scenarios, robots cannot cross obstacles in one step, even if the obstacle is the same as shown in Figure 20. If the robot climbs up and stops under the obstacles, $d_k > d'_{\text{max}} - d'_k - h_k,$ then the robot cannot complete the task of negotiating the obstacle in one step. To solve this problem, we can adopt multistep negotiating an obstacle.

Multistep negotiating an obstacle consists of three steps: climbing down, climbing up, and negotiating.

Firstly, the robot climbs down the pole for $m$ times, as shown in Figure 21(a). Set the step backward be $d_m$

$$d_k = d_m \times m + d_{km} \quad (69)$$

$$d'_{\text{min}} \leq d_m \leq d'_{\text{max}} \quad (70)$$

Secondly, the robot climbs up the pole for $n$ times, as shown in Figure 21(b). Set the step upward be $d_n$

$$d'_{\text{min}} \leq d_n \leq d'_{\text{max}} \quad (71)$$

$$d_{kg} = d_k - d_n \times n \quad (72)$$

$$(d_k - d'_{\text{max}} \times n) \leq d_{kg} \leq (d_k - d'_{\text{min}} \times n) \quad (73)$$

Thirdly, the robot negotiates the obstacle in the form of one step.

Although the reduction of $d_{kg}$ can help the robot to negotiate the obstacle, $d_{kg}$ has a minimum $d'_{\text{min}}$ due to the structure of PCRobot. When the robot approaches the obstacle, the upper gripper may collide with the obstacle. The critical situation is shown in Figure 22. The end of the grab just contacts $K_2.$ In this case, $d_{\text{kg}}$ is defined as a “critical distance.”

$l_h$ is the length of the gripper. $d_{\text{kg}}$ can be obtained as shown in equation (74)

$$d_{\text{kg}} = \sqrt{h^2 - (|y_{H_1} - w_k|^2 - x_{H_1})} \quad (74)$$

$$\max((d_k - d'_{\text{max}} \times n), d_{\text{km}}) \leq d_{\text{kg}} \leq (d_k - d'_{\text{min}} \times n) \quad (75)$$

Obstacle-surmounting control process

Based on the single-step and multistep negotiating an obstacle, the robot can realize the obstacle-surmounting. Suppose that the obstacle moves relative to the robot, the velocity of the robot is $v,$ and the velocity of the obstacle is $w_k.$

$$v = \frac{d'}{t_k} \quad (76)$$

$$8 < d_l - v_k t_k < d' \quad (77)$$

where $d'$ is the stride distance, $d_l = d_k + h_k,$ and $t_k$ is the time from the downward movement of the obstacle from $K_4.$
Constrained by the robot’s current structure, $d'$ is 8 to 23 cm.

From equation (77)
\[ \frac{8}{t_k} < \frac{d_l}{t_k} - v_k < v \]  

$tk$ can be equal to $t$ (e.g. $t_k = 8\ s$), which means that the robot meets and crosses obstacles in a period. Let $v_{\min} = \frac{8}{t_k}$, equation (79) can be obtained
\[ \frac{d_l}{t_k} - v < v_k < \frac{d_l}{t_k} - v_{\min} \]  

In addition to the surmountable condition, the obstacle-surmounting also needs to satisfy equation (79).

The robot’s obstacle-surmounting process is shown in Figure 23. Firstly, the camera $c$ samples the image information, $d_k$ and $v_k$ of obstacles. Through $v_k$ identifies whether the obstacle is stationary or moving, so that static obstacle crossing and dynamic obstacle crossing are performed, respectively. The obstacle size is extracted. According to the above D-H coordinate transformation, we can judge whether the surmountable conditions are satisfied. Next, judge whether the single-step negotiating an obstacle condition is satisfied. If it is satisfied, single-step negotiating an obstacle will be executed; otherwise, multi-step negotiating an obstacle will be executed.

**Prototype and test**

We designed PCRobot prototype as shown in Figure 2. The prototype weighed 1.07 kg (including a battery) and it was 60 cm long when fully extended. The specific size is shown in Table 11. Constrained by the robot’s current structure, a single span of the robot was 8 to 23 cm. The parameters of existing climbing robots are listed in Table 12. PCRobot is much lighter than other robots. Its shape is similar to a line.
In terms of volume, PCRobot is much smaller than the other robots in Table 12. The size of the smallest robot in Table 12 is 72 \( \times \) 50 \( \times \) 22 cm\(^3\), while that of PCRobot is 60 \( \times \) 7.1 \( \times \) 5.4 cm\(^3\).

Table 11. Dimensions of PCRobot components.

| Component | \( a_0 \) | \( a_1 \) | \( a_2 \) | \( a_3 \) | \( a_4 \) | \( h_1 \) | \( h_2 \) | \( g_1 \) | \( g_2 \) |
|-----------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| Length (mm) | 82 | 84 | 81 | 84 | 82 | 40 | 40 | 110 | 110 |

To enable the robot to sample the angle relative to the ground in real time, MEMS three-axis accelerometers (MPU6050; InvenSense Inc., Sunnyvale, CA, United States) were equipped on \( l_1, a_1, a_2, a_3, l_2 \), respectively, named \( A_1, A_2, A_3, A_4, \) and \( A_5 \) (see Qiaoling et al.\(^{18}\)).

Table 12. Comparison of climbing robots.

| Parameters | Mass (kg) | Total length (m) | Velocity (cm/s) |
|------------|-----------|------------------|-----------------|
| Climbing robot\(^2\) | 15.6 | 1.0 | 80 |
| RISE V3\(^6\) | 5.4 | 0.98 | 21 |
| Climbot\(^7\) | 42 | 2.5 | 3.7 |
| Tree-climbing robot\(^{20}\) | 6.8 | — | 7.5 |
| Tree-climbing robot\(^{21}\) | 100 | — | 5.0 |
| Pole-climbing robot\(^{22}\) | 10.5 | 0.72 | 2.9 |
| Our robot | 1.07 | 0.6 | 5.75 |

In terms of volume, PCRobot is much smaller than the other robots in Table 12. The size of the smallest robot in Table 12 is 72 \( \times \) 50 \( \times \) 22 cm\(^3\), while that of PCRobot is 60 \( \times \) 7.1 \( \times \) 5.4 cm\(^3\).

The size of the obstacle negotiating an obstacle

Set the lateral view of the obstacle crossed be rectangular. The test records the size of the lateral view of the obstacle that can be negotiated. The green area is the size of the lateral view of obstacles that PCRobot can negotiate in Figure 24. It shows that the width of the lateral view of the obstacle is inversely proportional to its height. The maximum height was 13.5 cm, and the maximum width was 15.5 cm.

Anti-sliding load capacity test of PCRobot

The anti-sliding load capacity of the robot was tested on wooden and PVC climbing pole, respectively. The robot was controlled in three states. Posture 1, the upper and
lower grippers grip the pole; posture 2, the lower gripper grips the pole and the upper gripper is released; posture 3, the upper gripper grips the pole and the lower gripper is released. In each state, a downward pull force is applied at $s_1$, $s_2$, $s_3$, and $s_4$ steering gears to test the anti-sliding load capacity of the robot. The force gauge is Handpi.
Instrument, HP–500, division value 0.1 N. The results of the anti-sliding load capacity are listed in Table 13.

It can be obtained from Table 13 that the friction coefficient of iron relative to wood material is small, so the anti-sliding load capacity of PCRobot on iron climbing pole is reduced. For the same material climbing pole, the different cross section will also affect the anti-sliding load capacity of the PCRobot. For the gripper of PCRobot, the anti-sliding load capacity of the robot on the square climbing pole is strong.

**Experiment of single-step negotiating the obstacle**

Experiments were done to test the single-step and multistep negotiating an obstacle, respectively, and the dynamic obstacle-surmounting.

![Figure 26. (a) to (l) Multistep negotiating the obstacle.](image-url)
The cross section of the climbing pole was rectangular 2.7 × 1.7 cm². The cube obstacle was 6 × 4 × 13 cm³. PCRobot was placed in the initial position, and the distance between the upper gripper and the obstacle was $d_k = 4$ cm. The controlling time of the steering gear was 2 s. According to (10), $t = 8$ s. The stride distance of the PCRobot was 20 cm. The velocity of the robot was 2.5 cm/s.

The robot strides over obstacles in one step, as shown in Figure 25(a) to (d).

**Experiment of multistep negotiating the obstacle**

The distance between the upper gripper and the obstacle was $d_k = 47$ cm. Other conditions were the same as above. The robot strode over obstacles in multistep, as shown in Figure 26. Firstly, set a stride distance of 23 cm to climb up. The velocity of the robot was 2.875 cm/s. After climbing twice, the robot reached the position shown in Figure 26(a). Secondly, the robot turned backward in a stride distance of 10 cm as shown in Figure 26(a) to (d). The velocity of the robot was 1.25 cm/s. Back to the position shown in Figure 26(d), the robot began to turn up in a stride distance of 23 cm, as shown in Figure 26(e) to (i). The velocity of the robot was 2.875 cm/s. When the robot reached the position shown in Figure 26(i), it crossed the obstacle in one step, as shown in Figure 26(i) to (l). The process of crossing obstacles was 8 s. The robot successfully surmounted the obstacle. The stride distance of the PCRobot was 20 cm. The velocity of the robot was 2.5 cm/s.

**Experiment of dynamic obstacle-surmounting**

The distance between the upper grippers and the obstacle was $d_k = 30$ cm, then $d_l = 34$ cm. Other conditions were the same as above. Set $d' = 20$ cm, $v_{\text{min}} = 1$ cm/s and $t_k = 8$ s. Then $v = 2.5$ cm/s. Equation (80) can be obtained

$$1.75 < v_k < 3.25$$

In the experiment, a simple structure of the pulley was used to control the obstacle to slide down at $v_k = 2$ cm/s. The robot strode over obstacles in one step, as shown in Figure 27(a) to (e).

**Conclusions**

In this article, the obstacle-surmounting strategy of the climbing robot is proposed. This strategy can solve the problem that the robot cannot stride over obstacles in one step in some scenarios. Firstly, the obstacle surmounting model of five-bar linkage in closed space is established. Secondly, the surmountable conditions were proposed. Thirdly, single-step obstacle crossing and multistep obstacle crossing are proposed. In the process of obstacle surmounting, the camera is used to collect the information of obstacles, and the single-step or multistep obstacle crossing gait is determined according to the sampling information. The strategy does not need to add additional external auxiliary equipment and can cross obstacles on the only way. The strategy proposed in this article can be used as a reference for climbing robot to surmount obstacles.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) received no financial support for the research, authorship, and/or publication of this article.

**ORCID iD**

Du Qiaoling https://orcid.org/0000-0002-4521-8041

**Supplemental material**

Supplemental material for this article is available online.
References

1. Schmidt D and Berns K. Climbing robots for maintenance and inspections of vertical structures—a survey of design aspects and technologies. Robot Autonom Syst 2013; 61(12): 1288–1305.
2. Li P, Duan X, Sun G, et al. Design and control of a climbing robot for inspection of high mast lighting. Assem Autom 2019; 39(1): 77–85.
3. Bogue R. Climbing robots: recent research and emerging applications. Indu Robot Int J Robot Res Appl 2019; 46(6): 721–727.
4. La HM, Dinh TH, Pham NH, et al. Automated robotic monitoring and inspection of steel structures and bridges. Robotica 2019; 37(5): 947–967.
5. Gabriel G and Dennis H. Considerations for finding the optimal design parameters for a novel pole climbing robot. In: 32nd annual mechanisms and robotics conference. New York, NY, 03–06 August 2008.
6. Clark HG, Khripin A, Lynch G, et al. Rapid pole climbing with a quadrupedal robot. In: IEEE international conference on robotics and automation. Kobe, Japan, 12–17 May 2009, pp. 2767–2772.
7. Guan Y, Jiang L, Zhu H, et al. Climbot: a bio-inspired modular biped climbing robot—system development, climbing gaits, and experiments. J Mech Robot 2016; 8(2): 1–17.
8. Yoon Y and Rus D. Shady3D: a robot that climbs 3D trusses. In: IEEE international conference on robotics and automation. Rome, Italy, 10–14 April 2007, pp. 4071–4076.
9. Chen S, Zhu H, Guan Y, et al. Collision-free single-step motion planning of biped pole-climbing robots in spatial trusses. In: IEEE international conference on robotics and biomimetics. Shenzhen, China, 12–14 December 2013, pp. 280–285.
10. Zhu H, Guan Y, Wu W, et al. A binary approximating method for graspable region determination of biped climbing robots. Adv Robot 2014; 28(21): 1405–1418.
11. Branyon C and Menguc Y. Soft snake robots: investigating the effects of gait parameters on locomotion in complex terrains. In: 25th IEEE/RSJ international conference on intelligent robots and systems. Madrid, Spain, 01–05 October 2018, pp. 1–9.
12. Chen W, Gu S, Zhu L, et al. Representation of truss-style structures for autonomous climbing of biped pole-climbing robots. Robot Autonom Syst 2018; 101: 126–137.
13. Gu SC, Zhu HF, Li H, et al. Optimal collision-free grip planning for biped climbing robots in complex truss environment. Appl Sci-basel 2018; 8(12): 1–22.
14. Luo LF, Wen HJ, Lu QH, et al. Collision-free path-planning for six-dof serial harvesting robot based on energy optimal and artificial potential field. Complexity 2018; 2018: 1–12.
15. Wei K and Ren B. A method on dynamic path planning for robotic manipulator autonomous obstacle avoidance based on an improved RRT algorithm. Sensors 2018; 18(2): 1–15.
16. Xu F and Jiang Q. Dynamic obstacle-surmounting analysis of a bilateral-wheeled cable-climbing robot for cable-stayed bridges. Ind Robot Int J Robot Res Appl 2019; 46(3): 431–443.
17. Lee C and Chu B. Three-modular obstacle-climbing robot for cleaning windows on building exterior walls. Int J Precis Eng Manuf 2019; 20(8): 1371–1380.
18. Qiaoling D, Yan L and Sinan L. Design of a micro pole-climbing robot. Int J Adv Robot Syst 2019; 16(3): 1–11.
19. Guo Z, Cui T, Lv X, et al. Inverse kinematics of 6-DOF manipulator. Agr Mech Res 2016; 38(12): 51–55.
20. Gui PF, Tang LQ and Mukhopadhyay S. A novel robotic tree climbing mechanism with anti-falling functionality for tree pruning. J Mech Robot-Trans Asme 2018; 10(1): 1–8.
21. Li Z, Ge Z, Cheng F, et al. Design and implementation of a new type of climbing robot for living trees. Wood Proc Mach 2016; 27(01): 32–34pp6.
22. Fauroux J and Morillon J. Design of a climbing robot for cylindro-conic poles based on rolling self-locking. Ind Robot 2010; 37(3): 287–292.