TECHNICAL PAPER

Kinematic performance-based path planning for cable-driven parallel robots using modified adaptive RRT*

Jiajun Xu[1] · Kyoung-Su Park[1]

Abstract

Cable-driven parallel robots (CDPRs) have demonstrated a remarkable potential for a wide range of applications over the past few decades. However, multiple cables introduce various interferences, which is a challenging task for the path planning of CDPRs. In this study, an adaptively optimal rapidly exploring random tree (RRT*) is proposed for CDPRs to generate collision-free paths in cluttered environments. The proposed method considers the kinematic performance of CDPRs by maximizing the feasible wrench volume and optimizing the dexterity of the robot. Furthermore, a sampling method was developed to guide the tree growth efficiently, by adaptively adjusting the sampling space. In the sampling space, the adaptive forward and backward areas are defined for biased sampling, and the probability weight of both areas is determined by the failure rate of the current node expansion. Finally, a post-processing method was adopted to obtain a shorter and smoother path. Simulations and experiments were carried out to verify the proposed path planning method on a self-built 8–6 (8 cables with 6 degrees of freedom) CDPR. The results indicate that the proposed method has much better efficiency than the conventional RRT* method, and the kinematic performance of the CDPR can be guaranteed.

1 Introduction

Cable-driven parallel robots (CDPRs) are a special class of parallel robots in which the end-effector is actuated by multiple cables. Fixed winches attached to the base frame are used to control the cable length (Gosselin 2014). CDPRs have several significant advantages, such as a large workspace, high payload to weight ratio, and low inertia (Duan et al. 2015; Yuan et al. 2015; Gagliardini et al. 2014). Therefore, CDPRs have been applied in rehabilitation training, large radio telescopes, building cleaning, and 3D printing (Qian et al. 2018).

However, CDPRs still face some limitations owing to cable actuation, which makes it challenging to develop a path planning strategy for practical applications (Wischnitzer et al. 2008). Because multiple cables easily cause interference with obstacles in cluttered environments, it is important to ensure the safety of the robot during the motion of the end-effector (Nguyen and Gouttefarde 2015).

The collision phenomenon can cause instability in such robots, causing serious operational problems. In addition, because the cable tension must maintain a positive consistency, the feasible wrenches that the cable can generate are essential for structural stability (Bouchard et al. 2010). The kinematic performance of CDPRs is considered in the proposed path planning method by employing the feasible wrench area and the dexterity indicator (Abdolshah et al. 2017).

Accordingly, researchers have developed different methods for the motion planning of CDPRs. Jiang et al. (2018) proposed dynamic point-to-point trajectory planning for a six-DOF CDPR. More recently, Idà et al. (2019) studied the trajectory planning of CDPRs in the case of rest-to-rest motions, implying that the motion time and path geometry are predefined. However, these studies only considered an underactuated CDPR in an environment without obstacles. Furthermore, the under-actuation of CDPR indicated that there is a unique solution for the cable tension distribution, which easily generates an unbounded oscillatory end-effector (Idà et al. 2021).

In a cluttered environment, path planning is important for CDPRs to avoid obstacles. Lahouar et al. (2009) presented a grid-based path planning method with two modes for a 4-cable-driven parallel manipulator, where the first

✉ Kyoung-Su Park
  pks6348@gachon.ac.kr

[1] Department of Mechanical Engineering, Gachon University, 1342 Seongnamdaero, Sujeong-gu, Seongnam-si 461-701, Gyeonggi-do, Korea

mode leads the robot to move to the target directly, and the second mode generates a feasible path for the robot to avoid the obstacle. However, the grid cell can reduce the diversity of possible paths. To improve this, Bak et al. (2019) proposed a modified rapidly exploring random trees (RRT) method to generate collision-free paths for CDPRs, which utilized goal-biased sampling and post-processing to reduce the path cost. The RRT method performs well in dealing with various interferences during the path planning of CDPRs by sampling the configuration space (C-space) (LaValle 1998). To improve the conventional RRT method, RRT* is proposed to guarantee optimized results by utilizing neighbor nodes to rewire the tree (Karaman and Frazzoli 2011). Several researchers have also investigated path planning for various types of CDPRs. Bury et al. (2019) developed a continuous collision-checking method for a CDPR mounted on a robotic arm. Rasheed et al. (2019) presented an optimization-based method for a reconfigurable mobile CDPR, but it is time consuming.

Although the above methods enable the CDPRs to avoid obstacles, the kinematic performance of the robot is not included. Owing to the special characteristics of the cable, the stability of the robot depends on the feasible wrenches that the cable can generate (Rasheed et al. 2020). Moreover, the kinematic reliability of the entire workspace must be guaranteed. An important metric for evaluating kinematic stability is the dexterity of the robot, which is denoted by the condition number of the kinematic Jacobian matrix. Zhang et al. (2018) designed a novel cost function based on dexterity, and the collision-free path planning of CDPR is completed by combining the RRT* and potential field guided (PFG) methods. More recently, Mishra et al. (2021) proposed an AFG-RRT* method to generate the optimal path for CDPRs in a cluttered environment. The cited method considered not only the dexterity of the robot but also the wrench capability defined by the capacity margin. However, both methods adopted the conventional goal-biased sampling method, which could reduce the convergence speed by sampling an invalid area.

Based on the above discussion, an adaptively optimal RRT* is developed as an offline path planning for CDPRs in a cluttered environment, which considers the interference among cables, end-effectors, and obstacles. To do so, the kinematic performance index of CDPRs is derived which integrates the dexterity and wrench capability metrics. One should note that although kinematic cost minimization is already widely used in the robotic industry, the wrench capability is not included before. The wrench capability aims to maximize the set of wrenches that can be generated by the cables on the end-effector, it is a unique property of CDPRs caused by the cable tension limitation. Moreover, the adaptive sampling method is proposed in the work to reduce the pathfinding time, which has two main parts: forward and backward sampling. These two sampling methods are adaptively selected based on the total expansion failure rate of the current node. Finally, post-processing was adopted to optimize the planned path.

The remainder of this paper is organized as follows. Section 2 presents the kinematic model of CDPR. Section 3 describes the kinematic performance indices in detail. In Sect. 4, the modified adaptive RRT* method is proposed and explained. The effectiveness of the proposed method is verified through simulations and experiments, as described in Sect. 5. Finally, Sect. 6 concludes the study.

2 CDPRs model

The kinematics analysis of CDPRs is a basic premise for studying the path planning problem. In this section, a general model of an n-DOF CDPR with m cables is established and its position kinematics is analyzed. As shown in Fig. 1a, the corresponding general model is illustrated. To simplify the analysis, the cable mass is neglected. Thus, the cable can be regarded as the straight line segment connected between the end-effector and fixed pully. The global coordinate frame O–XYZ is fixed on the base and the local coordinate frame e-xyz is attached to the center of the end-effector. For the CDPR with m cables, $\boldsymbol{a}_i$ ($i = 1, 2, …, m$) denotes the Cartesian coordinates of the cable exit point A_i expressed in O–XYZ, and $^e\boldsymbol{b}_i$ denotes the Cartesian coordinates of the end-effector anchor point B_i in e-xyz. The position and orientation of the end-effector described in O–XYZ are represented by $\boldsymbol{p} = [x, y, z]^T$ and $\boldsymbol{\Theta} = [\theta_x, \theta_y, \theta_z]^T$ repressed by Euler angles, respectively. Thus, the configuration vector of the end-effector can be denoted by $X = [\boldsymbol{p}, \boldsymbol{\Theta}]^T$. The rotation matrix from O–XYZ to e-xyz is denoted as $\boldsymbol{R}$.

According to Fig. 1a, we select the i-th closed-loop of the CDPR, the cable vector $\boldsymbol{l}_i$ can be described by

$$\boldsymbol{l}_i = \boldsymbol{a}_i - \boldsymbol{p} - \boldsymbol{R} \cdot^e \boldsymbol{b}_i \tag{1}$$

Equation (1) denotes the position relationship between the end-effector and the cable. Therefore, given a specific position of the end-effector, the state of each cable can be obtained.

The overview schematic of the proposed method in shown in Fig. 1b. This work aims to develop a methodology to efficiently generate an optimal path based on the kinematic performance of CDPR in a cluttered environment from a given starting pose to a given ending pose. To do so, the kinematic performance index of CDPRs is derived which integrates the dexterity and wrench capability metrics. Moreover, the adaptive sampling method is proposed to improve the efficiency of the algorithm. The
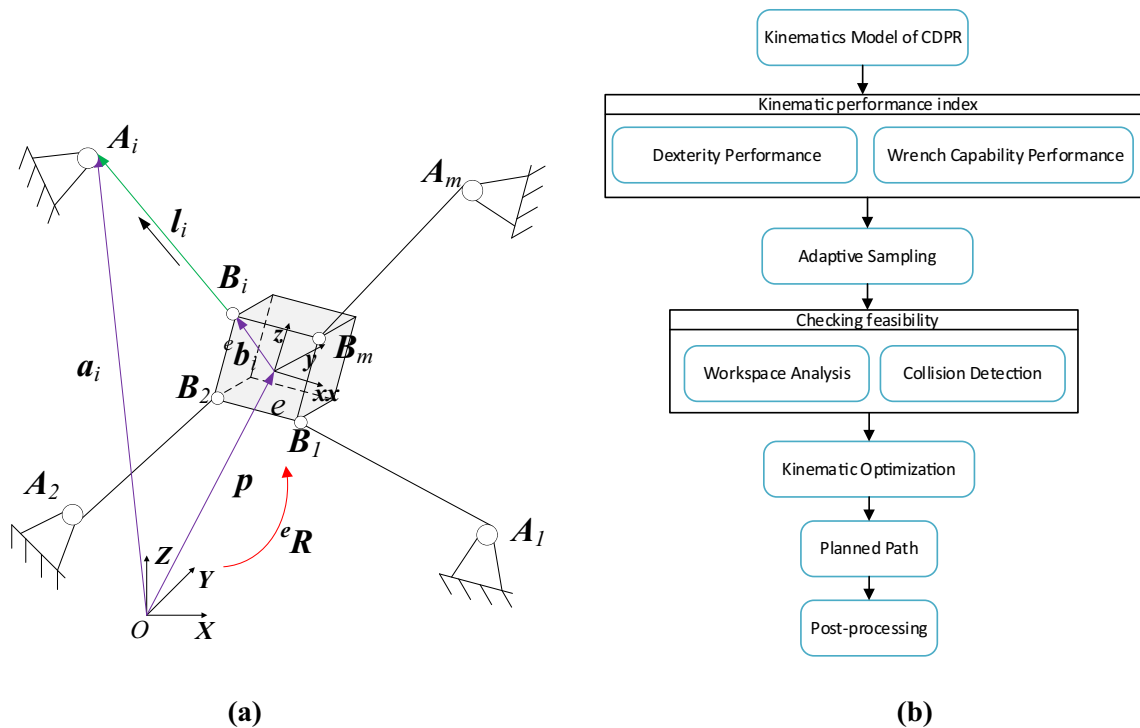
Fig. 1 Overview schematic of CDPRs. **a** Structural diagram of CDPRs; **b** overview schematic of the proposed method

various constraints such as the workspace, cables/obstacles, and end-effector/obstacles collisions are considered. Finally, the path post-processing method is proposed to smooth the final path.

3 Kinematic performance index generation

3.1 Dexterity index of CDPR

The dexterity of a parallel robot is an important indicator for measuring the accuracy of transmission between cable space and the Cartesian space, and the distribution of dexterity in the workspace plays an important role in the kinematic stability of the CDPR. To detect the dexterity of the CDPR, we differentiate Eq. (1) with respect to time, which gives the following

$$\dot{l} = -J^T \dot{X} \tag{2}$$

$$J(X) = \begin{bmatrix} u_1 & \cdots & u_m \\ u_1 \times (R \cdot^e b_1) & \cdots & u_m \times (R \cdot^e b_m) \end{bmatrix} \tag{3}$$

where $\dot{l} = [\dot{l}_1, \ldots, \dot{l}_m]^T$ is the velocity vector of the cables, $\dot{X} = [\dot{p}, \dot{\Theta}]^T$ represents the velocity vector of the end-effector, and u_i denotes the unit vector of the cable. $J \in \mathbb{R}^{m \times n}$ is the kinematic Jacobian matrix of the CDPR at a given pose. Therefore, the dexterity of CDPR is denoted by the

condition number of the kinematic Jacobian matrix, which is expressed as follows

$$k(J) = \|J\| \cdot \|J^{-1}\| \tag{4}$$

It should be noted that k is a number always greater than one, $k \in [1, \infty)$. To facilitate subsequent analysis, the dexterity index of the CDPR γ_d is represented by the normalization of k.

$$\gamma_d = \frac{k(J) - k_{min}}{k_{max} - k_{min}} \tag{5}$$

where k_{max} and k_{min} are the maximum and minimum values of the dexterity of CDPR in the workspace. Thus, a smaller number γ_d indicates that a better kinematic performance of the CDPR system at a given pose.

3.2 Feasible wrench set index of CDPR

According to the principle of virtual work, the static equilibrium equation of CDPRs is denoted as follows

$$JT + w_e = 0 \tag{6}$$

where $T = [t_1, t_2, \ldots, t_m]^T$ represents the cable tension vector. $w_e = [f_e, m_e]^T$ denotes the external wrench applied to the end-effector, f_e is the external force, and m_e is the external moment. Equation (6) indicates that the CDPR must balance the external wrench to maintain the stability of the system. In practice, the external wrench set (EWS)

can be formulated as a set $\mathbb{R}^n$ expressed by $W_e = \{ w_e \in \mathbb{R}^n \}$, which can be geometrically represented as a polytope. Consistent with this, owing to the limitation of cable tension, the feasible wrench set (FWS) that the cable can generate is defined as follows

$$W_f = \left\{ w_f \in \mathbb{R}^n | w_f = \sum_{i=1}^m t_i J_i, \ \underline{t}_i \leq t_i \leq \bar{t}_i \right\} \qquad (7)$$

Further, Eq. (7) can be rewritten as

$$W_f = \left\{ w_f \in \mathbb{R}^n | w_f = \sum_{i=1}^m \alpha_i \Delta t_i J_i + J\underline{t}, 0 \leq \alpha_i \leq 1 \right\} \qquad (8)$$

where $\Delta t_i = \bar{t}_i - \underline{t}_i$ is the feasible interval of the cable tension, J_i and is the i-th column of the Jacobian matrix. Accordingly, Eq. (8) can be expressed as a zonotope, which can be regarded as a special polytope with central symmetry (Fukuda 2004). However, it came to our attention that W_f depends on the pose of the end-effector. The convex hull method can be applied to obtain the FWS of the CDPR (Bouchard et al. 2010).

Figure 2a, b shows the EWS and FWS the studied CDPR in this work at a pose $[0, 0, 445]^T$. To simplify the problem and visualize the zonotope in 3D space, the end-effector always remains horizontal in this work. The EWS in this work is a cuboid body due to the external rotational moment around the three axes is neglected. The cable tension was limited to $\underline{t}_i = 0$ and $\bar{t}_i = 100N$. As shown in Fig. 2b, the edge of the zonotope is equal to $\Delta t_i = 100N$. According to the definition of the FWS, the required wrenches in the task can be satisfied if the EWS belongs to FWS, thereby FWS should be as large as possible to balance W_e. Therefore, the feasible wrench index γ_f of CDPR is denoted as the normalization of $1/vol(W_f)$, where $vol(W_f)$ is the volume of the zonotope of W_f.

4 Formulation of modified adaptive RRT*

4.1 modified adaptive RRT*

In this section, the modified adaptive RRT* is proposed to generate a collision-free path in a cluttered environment based on the kinematic performance of CDPR. The structure of the proposed method is presented in Algorithm 1.

Algorithm 1 Modified adaptive RRT*

Input: x_{start}, $\mathcal{G}(x_{goal}, r_g)$, $Obstacle$, r_c
Output: $\mathcal{T}$
1. Initialize tree $\mathcal{T} \leftarrow x_{start}$
2. $x_{new} \leftarrow x_{start}$
3.   **while** $Distance(x_{new}, x_{goal}) \geq r_g$
4. $x_{rand} \leftarrow AdaptiveSample(\mathcal{T}, x_{goal})$
5. $x_{nearest} \leftarrow NearestNeighbor(\mathcal{T}, x_{rand})$
6. $x_{new} \leftarrow Steer(x_{nearest}, x_{rand}, \eta)$
7.   **if** CollisionFree $(x_{new}, x_{nearest})$ **then**
8. $c_{min} \leftarrow KinematicCost(x_{new}, x_{nearest})$
9. $\mathcal{T} \leftarrow InsertNode(\mathcal{T}, x_{new})$
10. $X_{neighbors} \leftarrow Neighbors(\mathcal{T}, x_{new}, r_c)$
11.   **for** each $x_{neighbor} \in X_{neighbors}$ **do**
12. $c_{new} = KinematicCost(x_{new}, x_{neighbor})$
13.   **if** $c_{new} < c_{min}$ and CollisionFree $(x_{new}, x_{neighbor})$
14. $Parent(x_{new}) \leftarrow x_{neighbor}$
15. $\mathcal{T} \leftarrow InsertEdge(\mathcal{T}, x_{neighbor}, x_{new})$
16.   **end if**
17.   **end for**
18.   **end if**
19. **end while**
20. **return** $\mathcal{T}$

In the proposed modified adaptive RRT* method, the tree is initialized with the start point $x_{start,}$ which is then assigned to the first new node x_{new}. The algorithm proceeds with the distance between the new node x_{new} and the goal point x_{goal} larger than a sufficiently small value r_g. The *AdaptiveSample* function generates a random node based on the proposed adaptive sampling method, and then the nearest node $x_{nearest}$ to the tree is found to facilitate the tree growth by the *Steer* function (line 4–6 of Algorithm 1). The Steer function can be expressed as follows
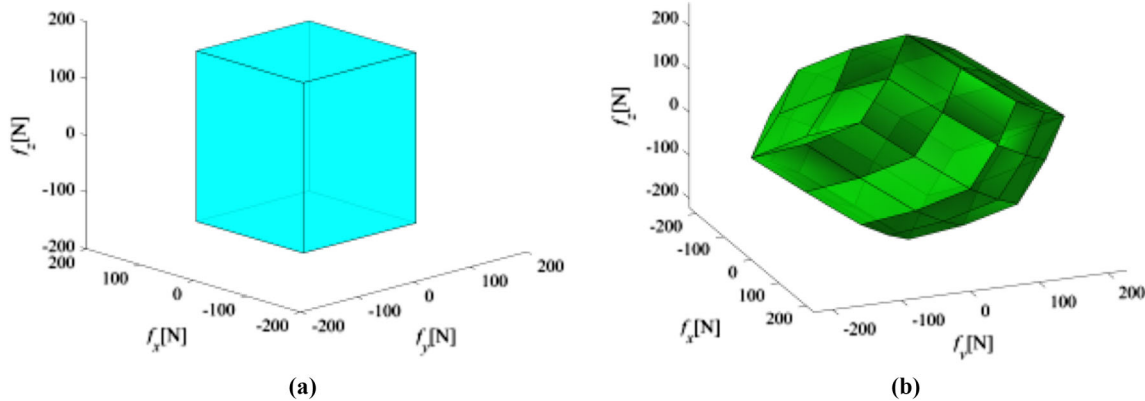

Fig. 2 Geometrical representation of EWS and FWS. **a** Polytope representation of EWS; **b** zonotope representation of FWS

$$x_{new} = x_{nearest} + \eta(x_{rand} - x_{nearest})/\|x_{rand} - x_{nearest}\| \quad (9)$$

If the path from $x_{nearest}$ to x_{new} is collision-free, x_{new} is inserted on the tree $\mathcal{T}$, and the kinematic cost of the new node is obtained based on Sect. 3, which is defined as $c = \gamma_d + \gamma_f$ in this work. Subsequently, an optimized strategy similar to RRT* was applied. The neighbors of x_{new} are obtained with a given radius r_c, and the parent of x_{new} is replaced by the neighbor node with a better kinematic cost (line 10–15 of Algorithm 1). This optimization process is intended to guarantee the kinematic performance of the robot.

4.2 Adaptive sampling

The adaptive sampling method is proposed in this section to reduce the pathfinding time, and the specific process is shown in Algorithm 2.

Algorithm 2 AdaptiveSampling

Input: $\mathcal{T}$, x_{goal}

Output: x_{rand}

1. $r_{rand} \leftarrow rand()$
2. **if** $r_{rand} > e_r$
3. $x_{rand} \leftarrow SamplingForward(\mathcal{T}, x_{goal})$
4. **else**
5. $x_{rand} \leftarrow SamplingBackward(\mathcal{T})$
6. **end**

The proposed sampling method has two main parts: forward and backward sampling. These two sampling methods are adaptively selected based on the total expansion failure rate of the current node e_r. r_{rand} is a random number between 0 and 1. Moreover, e_r can be formulated as the ratio of the number of expansion failures to the number of searches, which reflects the current situation of tree growth. A low e_r indicates that the influence of the obstacles is not powerful and the adaptive sampling method tends to sample the goal region. Nonetheless, it indicates that obstacles have a considerable impact on tree growth, and the sampling method samples the region far from the center of the tree to guide the tree away from the obstacle area.

4.2.1 Sampling forward

The *SamplingForward* function uniformly samples inside a hyperspherical cone centered on the goal point. The hypersphere cones in an *n*-dimensional space can be defined as follows

$$\begin{cases} x_1 = r\cos(\theta_1) \\ x_2 = r\sin(\theta_1)\cos(\theta_2) \\ x_3 = r\sin(\theta_1)\sin(\theta_2)\cos(\theta_3) \\ \qquad\qquad\qquad\qquad\vdots \\ x_{n-1} = r\sin(\theta_1)\sin(\theta_2)\cdots\sin(\theta_{n-2})\cos(\theta_{n-1}) \\ x_n = r\sin(\theta_1)\sin(\theta_2)\cdots\sin(\theta_{n-2})\cos(\theta_{n-1}) \end{cases} \quad (10)$$

where $0 \leq r \leq R$, $-\theta \leq \theta_1 \leq \theta$, $0 \leq \theta_2 \leq \pi$, …, $0 \leq \theta_{n-2} \leq \pi$, $0 \leq \theta_{n-1} \leq 2\pi$. In Eq. (9), the parameters R and θ are determined by the node x_c, which has a maximum expansion failure rate, x_{start}, and x_{goal}. R and θ can be obtained using the following equations

$$\begin{cases} R = \|x_c - x_{goal}\| \\ \theta = \arccos(u_s \cdot u_c) \end{cases} \quad (11)$$

where u_s and u_c are the unit vectors from x_{goal} to x_{start}, and x_{goal} to x_c, respectively. The basic sampling forward process is illustrated in Fig. 3.

As shown in Fig. 3a, the forward area is the line between the starting point and the target point, which ensures that the tree grows toward the goal point effectively. After colliding with an obstacle, the forward area is expanded to a cone to continuously guide the tree toward the goal point.

4.2.2 Sampling backward

In this work, the backward region is defined as a hyperball, and the center of the hyperball is the middle point of the current tree denoted by x_{center}, which is the mean value of all node coordinates. The radius r is defined as the distance from x_{center} to x_c. Therefore, the random node x_{rand} in the backward region is limited by the following equation.

$$\begin{cases} x_{center} = \sum_{i=1}^{N} x_i/N \\ r = \|x_{center} - x_c\| \\ \|x_{center} - x_{rand}\| > r \end{cases} \quad (12)$$

Figure 3b represents a Bug trap problem with high planning difficulty (Yershova et al. 2005). where the start point is inside a concave obstacle and cannot directly connect to the goal point. The proposed sampling method makes several unsuccessful attempts in the forward region and then samples the backward region. Thus, the nodes in the area surrounded by obstacles are sparse inside and dense outside. Compared to conventional sampling methods, the proposed sampling method can obtain fewer extensions and collision detections.

Fig. 3 Schematic of the proposed adaptive sampling method. **a** Schematic representation of sampling forward; **b** schematic representation of sampling backward

(a)

(b)
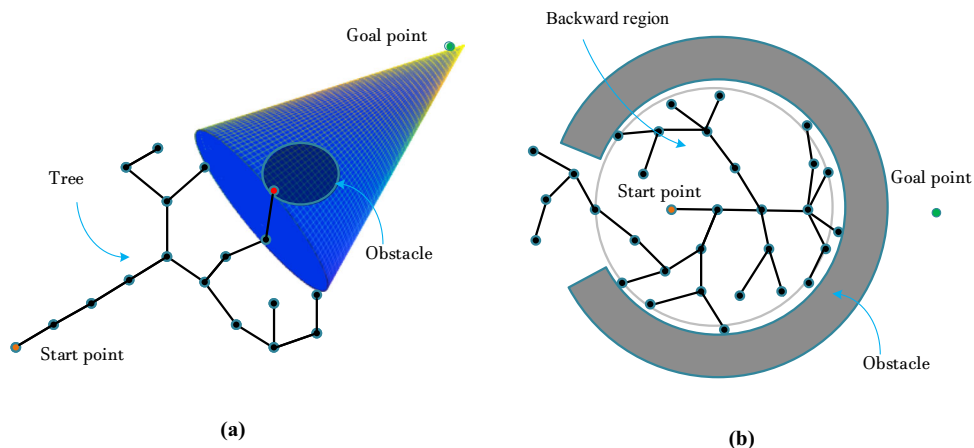
4.3 Post-processing

In general, the path generated by the modified adaptive RRT* is not smooth. In this section, post-processing is adopted to optimize the path, which is summarized in Algorithm 3.

Algorithm 3 Post-processing
Input: $\mathcal{T}$,σ
Output: σ*
1. $N = |σ|$
2. $i = 1$
3. σ* ←σ(i)
4. **while** $i ≠ N$
5. $j = N$
6.   **while** $j ≠ i$
7.   **if** *not* CollisionFree(σ(i), σ(j)) **then**
8. $i = j$
9.   σ* ←σ* ∪ σ(i)
10.   **break**
11.   **end**
12. $j = j\text{-}1$
13. **end**
14. **end**

As shown in Fig. 4a, the post-processing starts from the initial node on the initial path σ, and moves to the goal node progressively. Each child node is traversed until a collision-free path with minimum path cost is found. After post-processing, an optimal path σ*is generated, which does not have additional nodes that can be directly connected. As shown in Fig. 4b, the green line indicates the

optimized path σ*, it can be seen that the path smoothness is achieved.

5 Simulation and experimental results

5.1 Consideration

In this section, the proposed modified adaptive RRT* algorithm is simulated and tested on a self-built 8–6 (8 cables with 6 DOF) CDPR, as shown in Fig. 5. The end-effector is a cube with a size of 130 mm × 130 mm × 130 mm and a weight of 0.4 kg. The coordinates of the cable outlet point A_i ($i = 1,…,m$) are listed in Table 2 (Table 1).

The workspace is a significant constraint on the end-effector motion. In this work, the typical wrench closure workspace (WCW) is used, which is defined as a set of poses of the end-effector in which the cables can balance any external wrench. The maximum cable tension is the same as that in Sect. 3. For simplification, the cables were pre-tensioned in the initial pose. Thus, the cable tension was not controlled during the motion of the end-effector.

As shown in Fig. 6, the green part represents the WCW and the red line is the frame formed by cable outlet points. In this work, the obstacle/end-effector/cable and the WCW are modeled as the convex body, and the Gilbert–Johnson–

Fig. 4 Schematic of the path smoothness process. **a** Schematic diagram of post-processing. **b** Final optimized path
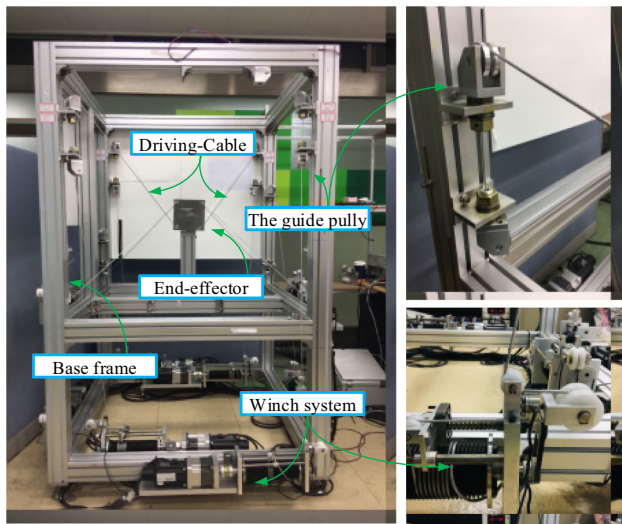

(a)

(b)
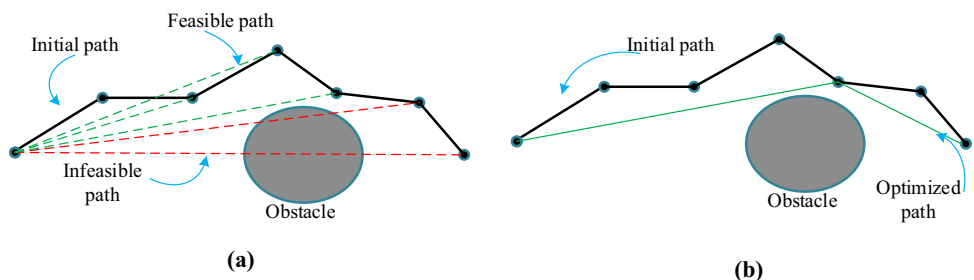
Fig. 5 8–6 CDPR prototype

Table 1 Coordinates of cable outlet points

Cable outlet points	Coordinates (mm)
A_1	[− 427.08, 524.32, 864.74]
A_2	[439.72, 525.94, 860.87]
A_3	[− 435.14, − 521.91, 839.17]
A_4	[437.89, − 523.83, 859.46]
A_5	[− 535.04, 466.89, 79.72]
A_6	[531.98, 451.43, 80.32]
A_7	[− 543.04, − 463.81, 71.49]
A_8	[529.05, − 448.63, 84.36]
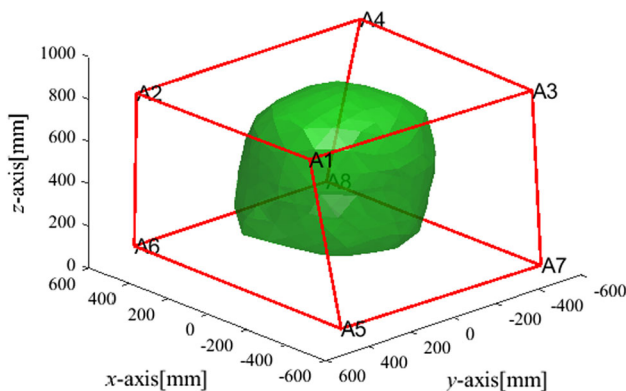
Fig. 6 Wrench closure workspace of the studied CDPR

Keerthi (GJK) algorithm is applied for the collision detection (Ong et al. 1997).

Based on the above analysis, the distribution of kinematic performance indices can be demonstrated in the workspace. From Fig. 7, it can be seen that the cost of the dexterity index gradually decreases as the end-effector approaches the center of the workspace. However, the cost of the feasible wrench set increases under the same conditions. Thus, it is necessary to consider the kinematic cost to balance this contradictory phenomenon.

5.2 Simulation

In this section, several simulations are conducted to demonstrate the proposed path-planning method. The simulation environment was MATLAB 2020a with an Intel i7-9750H processor and 32 GB RAM. The simulation parameters are listed in Table 2.

To validate the proposed method, three environments were designed in which the number of obstacles increased from one to three. These obstacles were modeled as a 3D axis-aligned bounding box, and located in [− 150, 0, 225], [200, − 150, 175], and [− 200, 200, 210], respectively. The closest proximity between the CDPR and the obstacles was defined as 10 mm. For contrasting conditions, the proposed method without adaptive sampling was simulated, in which the goal-biased sampling strategy with 0.5 probability was used to sample the goal point.

Figure 8 shows the simulation results, where the green, red, and blue lines denote the initial path σ, optimized path σ*, and tree, respectively. It is apparent that the proposed modified adaptive RRT* is capable of generating a collision-free path for CDPR, and the post-processing in Algorithm 3 completes the path smoothness with less path cost. It is also clear from the figure that the tree generated using the proposed adaptive sampling method has fewer nodes. Therefore, the computational efficiency of the proposed method is improved.

5.3 Evaluation

We should note that comparisons with other advanced methods can only be made if other conditions are consistent, however, computing the kinematic cost during the RRT sampling process is unique in this paper. To evaluate the reliability of the proposed method, 100 simulations were conducted for each environment, as illustrated in Sect. 5.2. These simulations used the same parameters listed in Table. 2. However, we used different start and goal points with the same path length to reproduce the simulation under identical conditions (Xu et al. 2020).
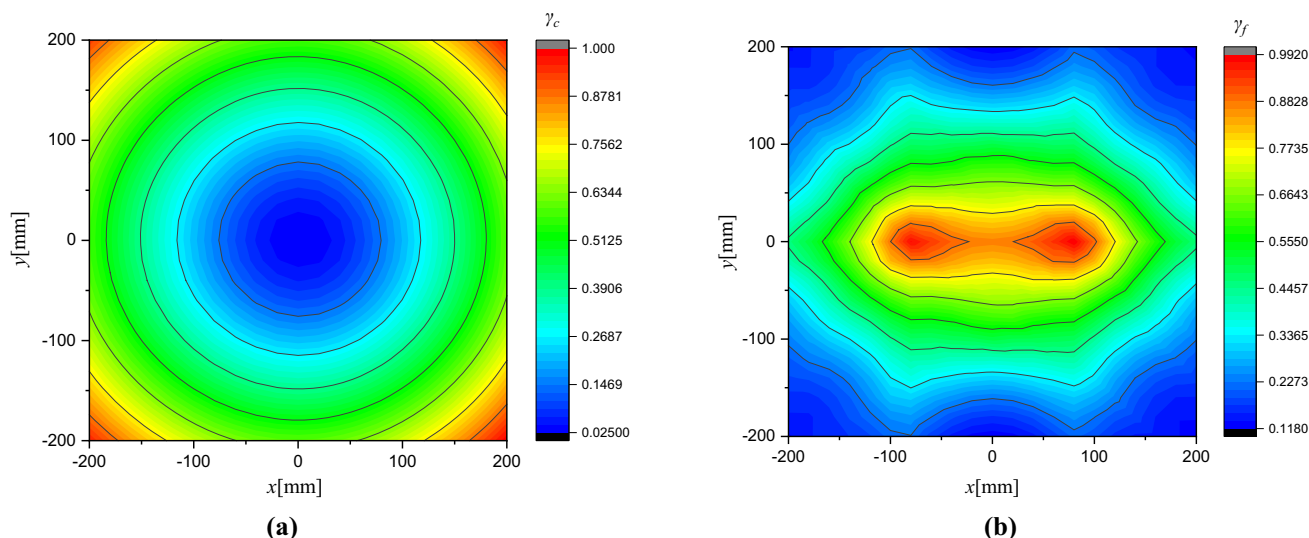
Fig. 7 Distribution of CDPR kinematic performance indexes in the workspace section ($z = 445$ mm). **a** Dexterity index. **b** Feasible wrench set index

Table 2 Simulation parameters

Parameter	Value (mm)
r_g	50
r_c	80
η	40
$\boldsymbol{x}_{start}$	[180, 180, 300]
$\boldsymbol{x}_{goal}$	[− 180, − 180, 400]
Obstacle 1	$80 \times 80 \times 450$
Obstacle 2	$220 \times 125 \times 350$
Obstacle 3	$220 \times 125 \times 420$

Figure 9a shows the average number of nodes on the tree and the corresponding standard deviation of the simulation evaluation for different environments. It can be observed that the average number of nodes is reduced by 20.5% compared to the goal-biased sampling with a 50% goal probability. Moreover, a lower standard deviation indicates that the proposed method is more stable. To further verify the improvement in kinematic performance, the kinematic cost in Algorithm 1 is replaced by the conventional path length cost, and the comparison result is shown in Fig. 9b. It can be seen that the proposed method obtains a lower kinematic cost, and the average kinematic performance increases by 18.2%. The smaller standard deviation of the kinematic cost with the proposed method can result in a more stable kinematic performance.

5.4 Experimental demonstration

The effectiveness of the proposed method was verified by a real-world experiment. The task was to allow the CDPR to avoid an obstacle with a collision-free path obtained in Fig. 8a. The experimental setup is shown in Fig. 10, which corresponds to environment 1.

We used the April tag to obtain the real position of the end-effector (Olson 2011). Furthermore, the end-effector was maintained at a low speed of 20 mm/s to avoid oscillation. From Fig. 11a, it can be seen that the proposed method generates a collision-free path such that the closest proximity between the CDPR and obstacle is consistently larger than the predefined 20 mm. The end-effector performs obstacle avoidance when the CDPR and obstacle are close to each other, as shown in Fig. 11b. The experimental results are shown in Fig. 12.

6 Conclusion

In this study, a modified adaptive RRT* method was developed for CDPRs to find a collision-free path in a cluttered environment while considering the kinematic performance. The proposed method developed an adaptive sampling method, minimized kinematic cost, and achieved fast convergence by utilizing post-processing. The proposed method was simulated in different environments. The simulation results show that the proposed method can
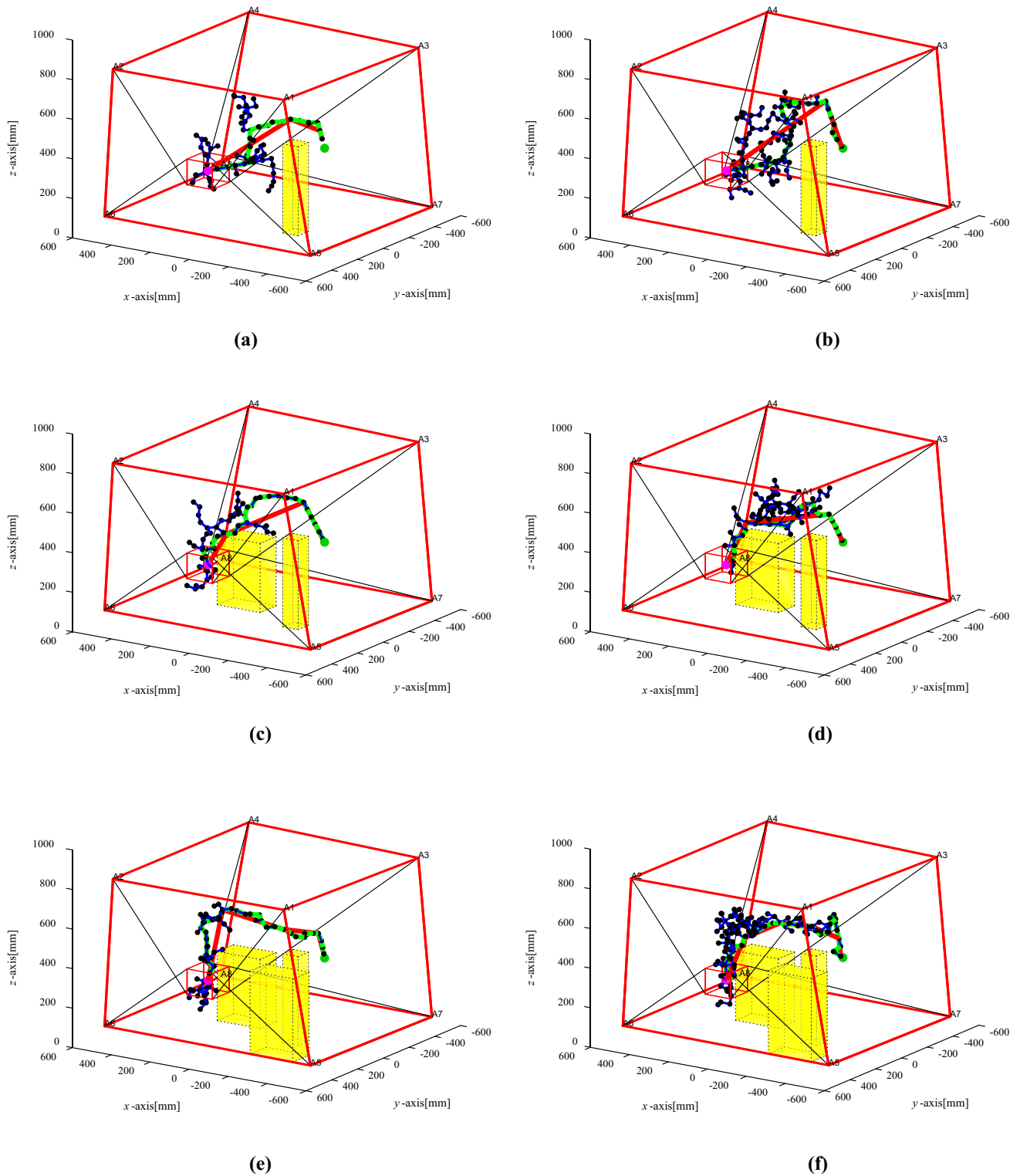
(a)

(b)

(c)

(d)

(e)

(f)

Fig. 8 Simulation results. **a** Environment 1 with the proposed method. **b** Environment 1 without adaptive sampling. **c** Environment 2 with the proposed method. **d** Environment 2 without adaptive sampling. **e** Environment 3 with the proposed method. **f** Environment 3 without adaptive sampling
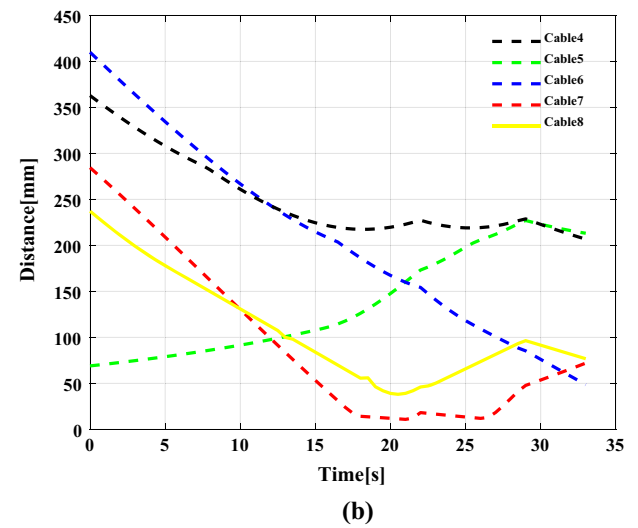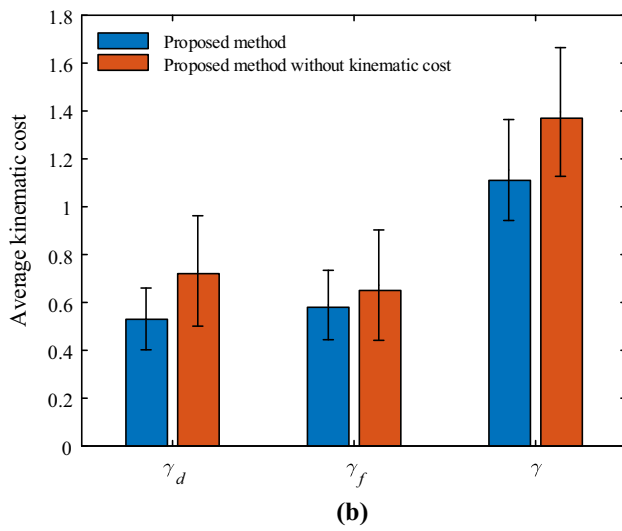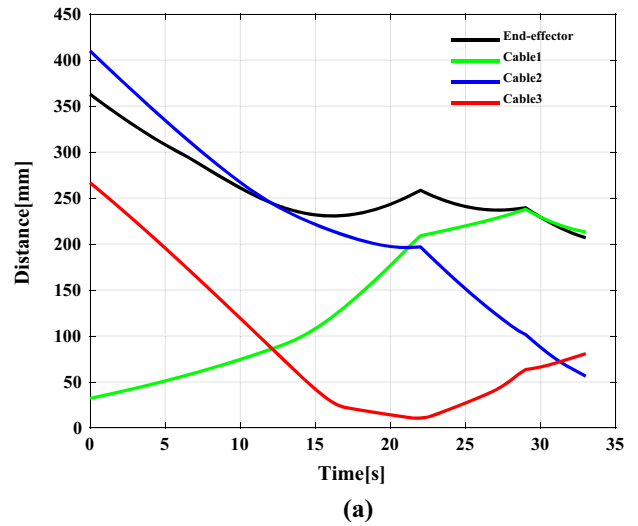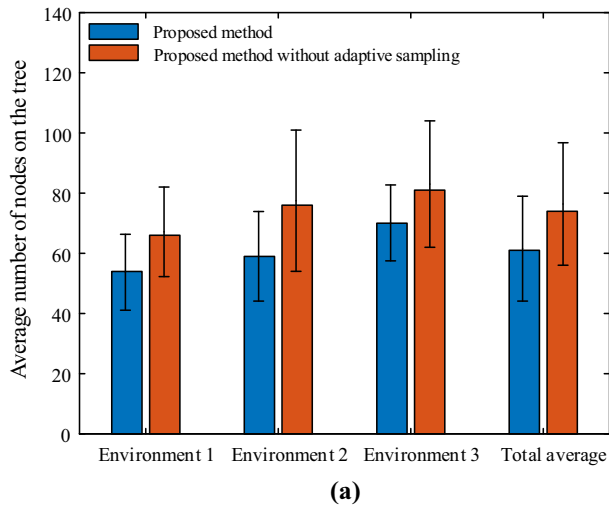
(a)

(b)

Fig. 9 Evaluation results. **a** Number of nodes comparison with different method. **b** Average kinematic cost comparison with different method
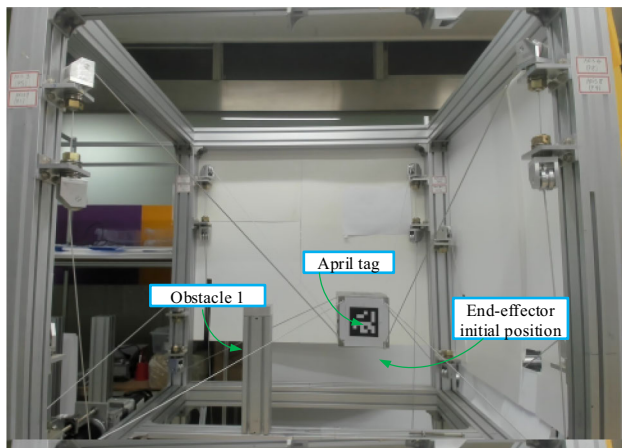

(a)

(b)

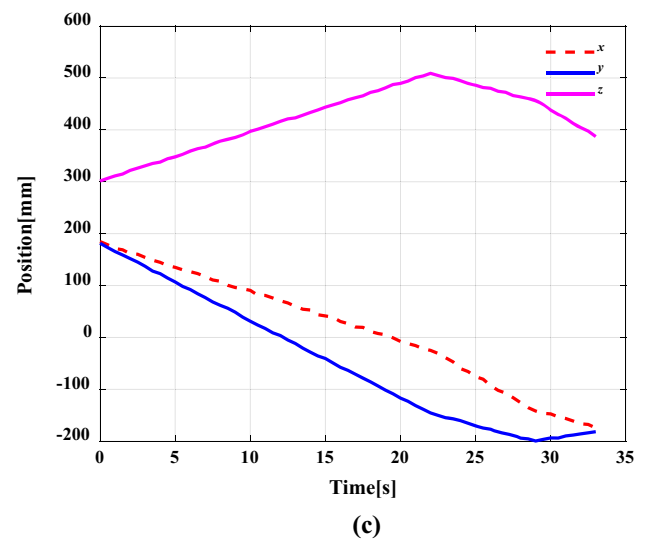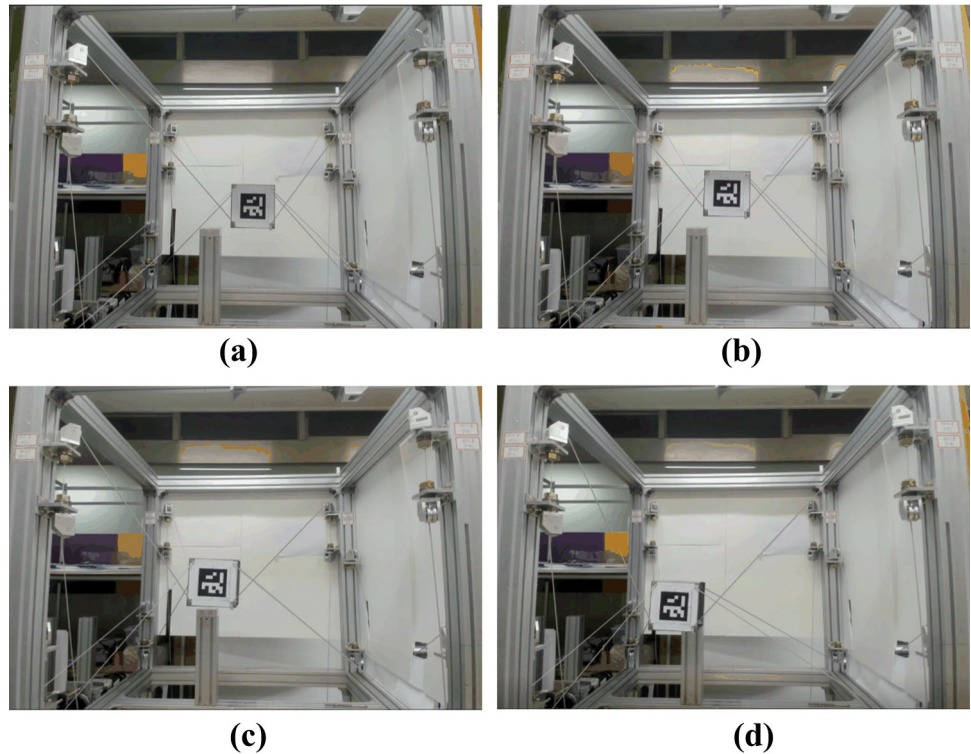Fig. 10 Schematic of experiment setup with environment 1

(c)

Fig. 11 Experiment results. **a**, **b** Distance change between the CDPR and obstacle. **c** Position change of the end-effector

Fig. 12 Experiment visualization. **a** $t = 15$ s. **b** $t = 20$ s. **c** $t = 25$ s, **d** $t = 35$ s

generate a collision-free path for CDPRs. Moreover, the average number of nodes on the tree is reduced by 20.5% compared to that without adaptive sampling, and the average kinematic performance increased by 18.2%. Finally, the developed method was fully validated and demonstrated through an experiment using a self-built 8–6 CDPR.

References

Abdolshah S, Zanotto D, Rosati G, Agrawal SK (2017) Optimizing stiffness and dexterity of planar adaptive cable-driven parallel robots. J Mech Robot 9(3):031004

Bak JH, Hwang SW, Yoon J, Park JH, Park JO (2019) Collision-free path planning of cable-driven parallel robots in cluttered environments. Intel Serv Robot 12(3):243–253

Bouchard S, Gosselin C, Moore B (2010) On the ability of a cable-driven robot to generate a prescribed set of wrenches

Bury D, Izard JB, Gouttefarde M, Lamiraux F (2019) Continuous collision detection for a robotic arm mounted on a cable-driven parallel robot. In: 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, pp 8097–8102

Duan Q, Vashista V, Agrawal SK (2015) Effect on wrench-feasible workspace of cable-driven parallel robots by adding springs. Mech Mach Theory 86:201–210

Fukuda K (2004) From the zonotope construction to the Minkowski addition of convex polytopes. J Symb Comput 38(4):1261–1272

Gagliardini L, Caro S, Gouttefarde M, Wenger P, Girin A (2014) Optimal design of cable-driven parallel robots for large industrial structures. In: 2014 IEEE international conference on robotics and automation (ICRA). IEEE, pp 5744–5749

Gosselin C (2014) Cable-driven parallel mechanisms: state of the art and perspectives. Mech Eng Rev 1(1):DSM0004

Idà E, Bruckmann T, Carricato M (2019) Rest-to-rest trajectory planning for underactuated cable-driven parallel robots. IEEE Trans Rob 35(6):1338–1351

Idá E, Briot S, Carricato M (2021) Natural oscillations of underactuated cable-driven parallel robots. IEEE Access 9:71660–71672

Jiang X, Barnett E, Gosselin C (2018) Dynamic point-to-point trajectory planning beyond the static workspace for six-dof cable-suspended parallel robots. IEEE Trans Rob 34(3):781–793

Karaman S, Frazzoli E (2011) Sampling-based algorithms for optimal motion planning. Int J Robot Res 30(7):846–894

Lahouar S, Ottaviano E, Zeghoul S, Romdhane L, Ceccarelli M (2009) Collision free path-planning for cable-driven parallel robots. Robot Auton Syst 57(11):1083–1093

LaValle SM (1998) Rapidly-exploring random trees: a new tool for path planning

Mishra UA, Mishra U, Métillon M, Caro S (2021) Kinematic stability based AFG-RRT* path planning for cable-driven parallel robots. In: The 2021 IEEE international conference on robotics and automation (ICRA 2021).

Nguyen DQ, Gouttefarde M (2015) On the improvement of cable collision detection algorithms. Cable-Driven Parallel Robots. Springer, Cham, pp 29–40

Olson E (2011) AprilTag: a robust and flexible visual fiducial system. In: 2011 IEEE international conference on robotics and automation. IEEE, pp 3400–3407

Ong CJ, Gilbert EG (1997) The Gilbert-Johnson-Keerthi distance algorithm: a fast version for incremental motions. In Proceedings

of international conference on robotics and automation, vol 2. IEEE, pp 1183–1189

Qian S, Zi B, Shang WW, Xu QS (2018) A review on cable-driven parallel robots. Chin J Mech Eng 31(1):1–11

Rasheed T, Long P, Roos AS, Caro S (2019) Optimization based trajectory planning of mobile cable-driven parallel robots. In: 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, pp 6788–6793

Rasheed T, Long P, Caro S (2020) Wrench-feasible workspace of mobile cable-driven parallel robots. J Mech Robot 12(3)

Wischnitzer Y, Shvalb N, Shoham M (2008) Wire-driven parallel robot: permitting collisions between wires. Int J Robot Res 27(9):1007–1026

Xu J, Park KS (2020) A real-time path planning algorithm for cable-driven parallel robots in dynamic environment based on artificial potential guided RRT. Microsyst Technol 26(11):3533–3546

Yershova A, Jaillet L, Siméon T, LaValle SM (2005) Dynamic-domain RRTs: efficient exploration by controlling the sampling domain. In: Proceedings of the 2005 IEEE international conference on robotics and automation. IEEE, pp 3856–3861

Yuan H, Courteille E, Deblaise D (2015) Static and dynamic stiffness analyses of cable-driven parallel robots with non-negligible cable mass and elasticity. Mech Mach Theory 85:64–81

Zhang B, Shang W, Cong S (2018) Optimal RRT* planning and synchronous control of cable-driven parallel robots. In: 2018 3rd international conference on advanced robotics and mechatronics (ICARM). IEEE, pp 95–100