TIME OPTIMAL ONLINE TRAJECTORY PLANNING OF ELECTROMECHANICAL SYSTEM BASED ON IMPROVED CRITICAL TEST CURVE ALGORITHM

Yang WANG [1 *]

For time-optimal online trajectory planning in electromechanical systems, the study proposes an innovative solution based on an improved Critical Test Curve (CTC) algorithm. The results show that the angle curves of each joint exhibit dynamic changes consistent with trajectory planning requirements, with corresponding and relatively continuous and smooth variations in velocity and acceleration curves. From the comparative plot of the end-effector error curves for the robotic arm, it is evident that the optimized robotic arm exhibits significantly reduced error fluctuations across all axes, with errors tending to be more stable. The lowest Z-axis error is approximately -1.32 mm. In contrast, the unoptimized robotic arm shows large error fluctuations in the X, Y, and Z-axis directions, with the Z-axis error dropping to around -2.5 mm. The overall error variations are severe and cover a wide range. It is clear that the trajectory planning algorithm effectively controls the motion states of each joint of the robotic arm, enabling orderly transitions in joint motion in terms of angle, velocity, and acceleration, and meeting kinematic requirements. This study provides a guarantee for achieving precise and stable trajectory tracking in robotic arm electromechanical systems.

Keywords: electromechanical system, trajectory planning, online planning, kinematic constraints, critical test curve algorithm

1. Introduction

For electromechanical systems, efficient and precise time optimal trajectory planning (TP) ensures the accuracy of system operation while minimizing operation time and improving production efficiency [1-2]. Online TP can dynamically adjust trajectories based on the real-time system state and task requirements, offering greater flexibility and adaptability. However, in practical applications, electromechanical systems are often subject to various kinematic constraints. Online TP can dynamically adjust trajectories according to the real-time system state and task demands, exhibiting stronger flexibility and adaptability. The Critical Test Curve (CTC) algorithm, as an effective TP method, holds certain advantages in handling kinematic constraints. Nevertheless, traditional CTC algorithms still exhibit some limitations, such as strict restrictions

* Corresponding author
[1] College of Electromechanical, Changchun Polytechnic University, Changchun, 130033, China, e-mail: wangyang780211@163.com

on the boundary conditions of parameter acceleration, which limit their application effectiveness in complex scenarios [3].

In existing research, in order to address the issues of traditional fuel vehicle durability evaluation not being applicable to pure electric vehicles and high user data collection costs, Wang et al. used fuzzy clustering to mine user trajectories to achieve distance targets, and combined particle swarm optimization to link user behavior with experimental field development and testing methods. The results show that the iteration time is more than 80% shorter than the polynomial method, and the testing specifications match the actual motor operation [4]. To reduce welding robots' energy consumption while ensuring coordinated joint motion, Zhou et al. used an improved sparrow search algorithm with seventh-order B-spline curves for trajectory construction. They solved optimal energy consumption time sequences via elite opposition-based learning to plan optimal trajectories; simulations showed continuous, smooth control and effective energy reduction [5]. To solve directed energy deposition's poor molding accuracy and material waste from improper process planning, Zhao et al. proposed a composite path and optimized overlap distances. They built a geometric model to analyze edge collapse impact, and a constrained compensation path method. Results showed material utilization rose from 80% to 95.6%, cutting waste and boosting efficiency/quality [6]. To solve low accuracy and instability in multi-UAV collaborative TP, Lou et al. proposed a method combining cuckoo search and golden jackal optimization. They built single/multi-UAV TP models with tent chaotic mapping, cuckoo search position updates, and nonlinear control parameters. Results showed the algorithm enables efficient, accurate trajectory planning with better stability, accuracy, and convergence speed [7]. To solve poor trajectory tracking of electric shovels under harsh environments/self-weight, Chen et al. proposed a Lyapunov-based inverse sliding mode method with its electromechanical coupling model, verified via simulations/experiments. Results showed it achieved tracking (deviations <0.3 m) and revealed track slippage/regenerative braking steering impacts [8]. To tackle time-varying trajectory tracking issues in cable-suspended planar parallel robots caused by nonlinear dynamics, Saber et al. established mechanical/electromechanical models and developed a nonlinear suboptimal controller. Control was realized via reference model formulation, pseudo-linearization, and Riccati equation solving. Simulations and experiments confirmed the controller stabilized the system, with tracking errors asymptotically approaching zero, showing robustness [9].

In summary, researchers have studied sparrow search algorithm, constraint compensation path planning, and model predictive control algorithm for the optimal online TP of electromechanical systems. However, traditional algorithms still have limitations, such as strict constraints on boundary conditions for parameter acceleration; Under the assumption of static boundaries, they cannot

explain the dynamic changes in constraints with motion states under real-world operating conditions. Therefore, this study proposes a time optimal online TP method suitable for electromechanical systems, which innovatively integrates kinematic constraints, comprehensively optimizes the CTC algorithm, and improves the TP performance of electromechanical systems under complex constraint conditions.

2. Methods and materials

2.1. Time optimal online TP of CTC algorithm considering kinematic constraints

The kinematic model is the foundation of robot research, and can be divided into forward kinematics and inverse kinematics based on the direction derived from equations. The traditional offline TP method generates trajectories based on preset operating conditions and cannot cope with sudden interference. In contrast, online planning requires constraint validation and trajectory correction to be completed within milliseconds. In order to solve the time optimal TP problem under complex operating conditions, this paper proposes an innovative solution based on an improved CTC algorithm for solving the time optimal online trajectory planning problem of mechanical and electrical systems such as robotic arms. As a typical mechatronics system, the robot arm integrates mechanical structure, electronic control, and software algorithms, and its trajectory planning needs to meet motion constraints and real-time requirements.

However, in existing research, the handling of kinematic constraints by the CTC algorithm is mostly confined to static boundary assumptions, failing to account for the dynamic variations of constraints with motion states in actual operating conditions [10-11].

In the classical kinematic modeling of electromechanical systems (e.g., robotic arms, manipulators), the hard constraint of joint position is a fundamental and widely accepted formulation [12]. Based on the conventional symbol system in classical robotics (where q_i denotes the position of the i-th joint, $q_{i,\min}$ and $q_{i,\max}$ represent the minimum and maximum physical limits of the i-th joint position, respectively), the classical form is presented in Equation (1).

$$q_{i,\min} \le q_i \le q_{i,\max} \quad (i = 1, 2, \ldots, n) \tag{1}$$

Let the joint variables of the electromechanical system be denoted as $q = [q_1, q_2, \cdots, q_n]^T$, where n represents the number of joints, T represents vector transpose symbol. The joint velocity $\dot{q} = [\dot{q}_1, \dot{q}_2, \cdots, \dot{q}_n]^T$ and acceleration $\ddot{q} = [\ddot{q}_1, \ddot{q}_2, \cdots, \ddot{q}_n]^T$ need to satisfy corresponding constraint conditions. First,

considering the joint velocity constraints, let the upper velocity limit of joint i be $\dot{q}_{i\max}$ and the lower velocity limit be $\dot{q}_{i\min}$. Then, the velocity constraints can be expressed as shown in Equation (2).

$$\dot{q}_{i\min} \le \dot{q}_i \le \dot{q}_{i\max}, \quad i = 1, 2, \cdots, n \tag{2}$$

For acceleration constraints, let the upper acceleration limit of joint i be $\ddot{q}_{i\max}$ and the lower acceleration limit be $\ddot{q}_{i\min}$. The acceleration constraints are expressed as shown in Equation (3).

$$\ddot{q}_{i\min} \le \ddot{q}_i \le \ddot{q}_{i\max}, \quad i = 1, 2, \cdots, n \tag{3}$$

To incorporate these kinematic constraints into the TP process, a parametric trajectory representation is adopted. Assuming the trajectory parameter is τ, the joint variables can be expressed as a function $\mathbf{q}(\tau)$ with respect to the parameter τ. The joint velocity and acceleration can then be obtained by differentiating $\mathbf{q}(\tau)$, as shown in Equation (4).

$$\begin{cases} \dot{\mathbf{q}}(\tau) = \dfrac{d\mathbf{q}(\tau)}{d\tau} \\ \ddot{\mathbf{q}}(\tau) = \dfrac{d^2\mathbf{q}(\tau)}{d\tau^2} \end{cases} \tag{4}$$

By substituting the aforementioned velocity and acceleration expressions into the kinematic constraint conditions, constraint equations with respect to the parameter τ are obtained. The optimal trajectory that satisfies the kinematic constraints is then determined by constructing a CTC. Suppose the objective function of the trajectory is time T, which aims to minimize the integral of $T = \int_0^{\tau_f} d\tau$, where τ_f represents the parameter value corresponding to the trajectory's end - point. A cubic polynomial parameterization is adopted to describe the trajectory, as shown in Equation (5).

$$\mathbf{q}(\tau) = \mathbf{a}_0 + \mathbf{a}_1\tau + \mathbf{a}_2\tau^2 + \mathbf{a}_3\tau^3 \tag{5}$$

In Equation (5), τ_f represents the parameter value corresponding to the trajectory's end-point. Here, $\tau \in [0,1]$ is the normalized time parameter. By differentiating it, the result is as shown in Equation (6).

$$\begin{cases} \dot{\mathbf{q}}(\tau) = \mathbf{a}_1 + 2\mathbf{a}_2\tau + 3\mathbf{a}_3\tau^2 \\ \ddot{\mathbf{q}}(\tau) = 2\mathbf{a}_2 + 6\mathbf{a}_3\tau \end{cases} \tag{6}$$

The relationship between the actual operating time T and the parameter τ is given by $t = T\tau$, and the objective of trajectory optimization is expressed as shown in Equation (7).

$$\begin{aligned} &\min T \\ &\text{s.t.} \quad \mathbf{q}(1) = \mathbf{q}_f, \dot{\mathbf{q}}(1) = \dot{\mathbf{q}}_f \end{aligned} \tag{7}$$

In Equation (7), $\mathbf{q}_f, \dot{\mathbf{q}}_f$ represent the target states. By incorporating the boundary conditions $\mathbf{q}(0) = \mathbf{q}_0$ and $\dot{\mathbf{q}}(0) = \dot{\mathbf{q}}_0$, the relationship between the

coefficient vector $\mathbf{a}_k$ and T can be solved [13-14]. Subsequently, the CTC construction is carried out, and the constraint critical function is defined as shown in Equation (8).

$$g_i(\tau) = \left| \frac{\ddot{\boldsymbol{q}}_i(\tau)}{\ddot{q}_{i,\max}} \right| - 1 \tag{8}$$

In Equation (8), the trajectory touches the constraint boundary when $g_i(\tau) = 0$. By searching for the critical points where $g_i(\tau) = 0$ in the τ-space, a family of CTCs is constructed, and subsequently, the optimal value of T is determined. On the premise of satisfying kinematic constraints, the trajectory is gradually optimized by searching for points on the CTC to achieve time-optimality.

The CTC algorithm that takes kinematic constraints into account requires the dynamic generation of trajectories by combining real-time states with kinematic constraints in time-optimal online TP. The specific steps are shown in Figure 1.

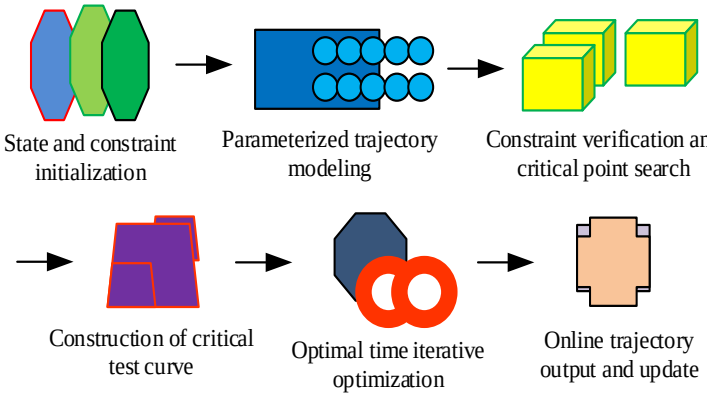


Fig. 1. CTC algorithm considering kinematic constraints for time optimal online TP

Figure 1 illustrates the process of time-optimal online TP using the CTC algorithm that considers kinematic constraints. The steps are carried out in sequence, including state and constraint initialization, parametric trajectory modeling, constraint verification and critical point searching, construction of the CTC, and iterative optimization for the optimal time, ultimately achieving the output and update of the online trajectory.

2.2. Time optimal online TP of electromechanical system considering improved CTC algorithm

A CTC algorithm framework based on static motion constraints was established, but it did not consider the time-varying characteristics of constraint

boundaries. Subsequent research proposes an improved CTC algorithm, which introduces an adaptive constraint estimation mechanism and focuses on establishing a dynamic constraint boundary estimation model and designing an adaptive critical curve search mechanism. When establishing a dynamic constraint boundary estimation model, an extended Kalman filter (EKF) is used to estimate time-varying acceleration constraints, as shown in equation (9).

$$\ddot{q}_{i,\max}(t) = \ddot{q}_{i,0} - k_i \| \dot{\mathbf{q}}(t) \| \tag{9}$$

In Equation (9), $\ddot{q}_{i,0}$ represents the rated acceleration, and k_i denotes the velocity attenuation coefficient. The state equation of the EKF is presented as shown in Equation (10).

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{w}_k \\ \mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \end{cases} \tag{10}$$

In Equation (10), $\mathbf{A}$ represents the state transition matrix, $\mathbf{w}_k$ denotes the process noise vector, $\mathbf{z}_k$ signifies the sensor observation vector, $\mathbf{H}$ indicates the observation matrix, $\mathbf{v}_k$ stands for the observation noise vector, and $\mathbf{x}_k$ represents the state vector, as shown in Equation (11) [15].

$$\mathbf{x}_k = [\ddot{q}_{1,\max}(t_k), \cdots, \ddot{q}_{n,\max}(t_k)]^T \tag{11}$$

Subsequently, an adaptive critical curve search is conducted, and the dynamic critical function is defined as shown in Equation (12).

$$g_i(\tau, t) = \left| \frac{\ddot{q}_i(\tau, T)}{\ddot{q}_{i,\max}(t)} \right| - 1 \tag{12}$$

The Quantum Particle Swarm Optimization (QPSO) algorithm is employed to solve for the optimal value of T, as presented in Equation (13).

$$\begin{cases} m_{ij}(t+1) = \phi m_{ij}(t) + c_1 r_1 (p_{ij}(t) - x_{ij}(t)) + c_2 r_2 (p_g j(t) - x_{ij}(t)) \\ x_{ij}(t+1) = x_{ij}(t) + m_{ij}(t+1) \end{cases} \tag{13}$$

In Equation (13), x_{ij} represents the particle position (corresponding to candidate values of T), p_{ij} denotes the individual best, and p_{gj} signifies the global best. Through the aforementioned improvements, the algorithm can ensure time optimality and enhance the constraint satisfaction rate.

Taking parameter acceleration as an example, in traditional algorithms, the constraints on parameter acceleration are typically assumed to be strict positive upper and negative lower limits, that is, $\ddot{\tau}_{\max} > 0$ and $\ddot{\tau}_{\min} < 0$. However, in actual electromechanical systems, this assumption does not always hold true [16-17]. To handle acceleration constraints more flexibly, research introduces a new acceleration boundary estimation method.

Through dynamic analysis of electromechanical systems and monitoring of actual operational data, the actual boundary functions $\ddot{\tau}_{ub}(\tau)$ and $\ddot{\tau}_{lb}(\tau)$ for

parameter acceleration are obtained. These functions are no longer confined to fixed positive and negative values but vary dynamically with the trajectory parameter τ [18-19]. Under such circumstances, the acceleration constraints can be expressed as shown in Equation (14).

$$\ddot{\tau}_{lb}(\tau) \le \ddot{\tau} \le \ddot{\tau}_{ub}(\tau) \tag{14}$$

To achieve TP based on such dynamic acceleration constraints, research has improved the nonlinear filter [20]. Let the transfer function of the filter be $H(s)$, and its parameter vector be $\boldsymbol{\theta}$. By monitoring the trajectory state and acceleration boundary functions in real time, an adaptive algorithm is used to adjust $\boldsymbol{\theta}$, enabling the filter to scale the reference trajectory more effectively to satisfy dynamically changing kinematic constraints [21-22]. The specific adaptive adjustment process is illustrated in Figure 2.

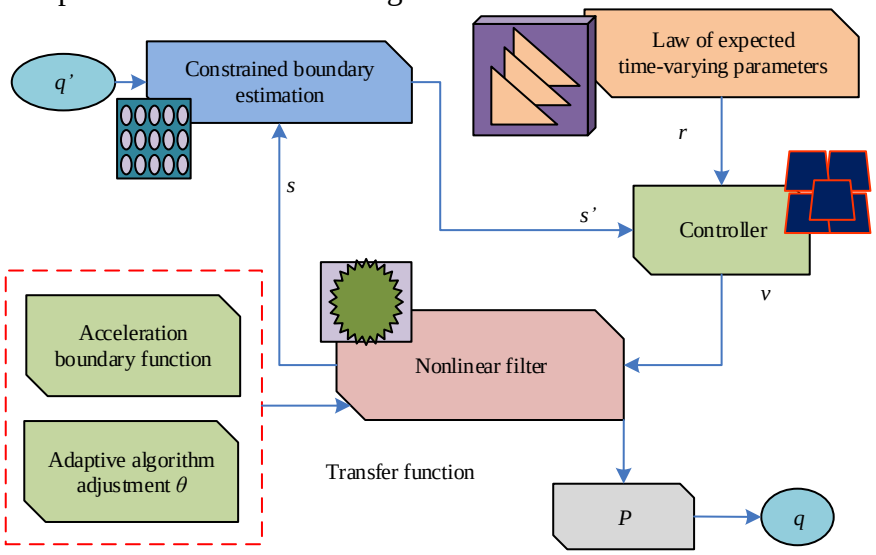


Fig. 2. Specific adaptive adjustment process

As shown in Figure 2, the current trajectory parameter τ and the corresponding acceleration boundaries $\ddot{\tau}_{ub}(\tau)$ and $\ddot{\tau}_{lb}(\tau)$ are obtained in real time. Based on the current trajectory state and acceleration boundaries, the direction and step size for adjusting the filter parameters are calculated [23-24]. For instance, the gradient descent method can be employed to compute the gradient $\nabla J(\boldsymbol{\theta})$ of the objective function with respect to the parameter vector $\boldsymbol{\theta}$. Subsequently, the parameters are updated according to the gradient direction and a preset step size α, as shown in Equation (15).

$$\boldsymbol{\theta}_{m+1} = \boldsymbol{\theta}_m - \alpha \nabla J(\boldsymbol{\theta}_m) \tag{15}$$

In Equation (15), m represents the iteration number [25]. The updated parameter vector $\boldsymbol{\theta}$ is applied to the filter $H(s)$ to scale the reference trajectory, thereby obtaining a new trajectory that satisfies the current kinematic constraints.

In order to achieve dynamic modeling and constraint fusion, a Lagrangian equation was used to establish a dynamic model of a 6-degree-of-freedom serial robotic arm, describing the mapping relationship between joint motion and driving torque, ensuring that trajectory planning can be executed at the dynamic level. Then, the dynamic constraints (driving torque constraints) are combined with the original kinematic constraints and integrated into the optimization objectives of the improved CTC algorithm to ensure that the trajectory simultaneously satisfies "time optimization+kinematic feasibility+dynamic feasibility".

3. Results

3.1. Performance analysis of time optimal online TP method for electromechanical system

The experimental object is a 6-degree-of-freedom serial joint robotic arm (model: UR5e), and the coordinate system of its robotic arm motion is shown in Figure 3. This robotic arm adopts a hybrid structure of rotating joints and moving joints, including a base, upper arm, middle arm, lower arm, wrist, and claw joint combination. The kinematic model is established using the D-H parameter method, and its forward kinematic equations are derived through homogeneous transformation matrices. The inverse kinematics is solved using numerical methods to ensure the accuracy of pose mapping.

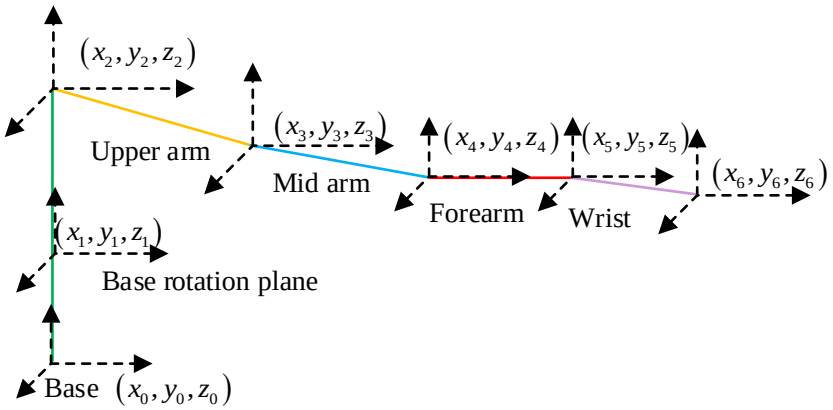


Fig.3. Coordinate system for the movement of the robotic arm of the robot

The experimental subject was an electromechanical system featuring a 6-joint robotic arm. To define the motion trajectory tasks that the robotic arm

needed to execute and provide specific objectives for the experiments, a sequence of waypoints was set. This involved moving from one waypoint to the next to accomplish specific TP tasks. The list of waypoint sequences is shown in Table 1.

Table 1

Path point order list

Path point number	Joint angle (JA) 1 (rad)	JA 2 (rad)	JA 3(rad)	JA 4(rad)	JA 5(rad)	JA 6(rad)
1	0	0	0	0	0	0
2	0.5236	0.3491	-0.2618	-0.5236	0.2618	0.1745
3	1.0472	0.6981	-0.5236	-1.0472	0.5236	0.3491
4	1.5708	1.0472	-0.7854	-1.5708	0.7854	0.5236

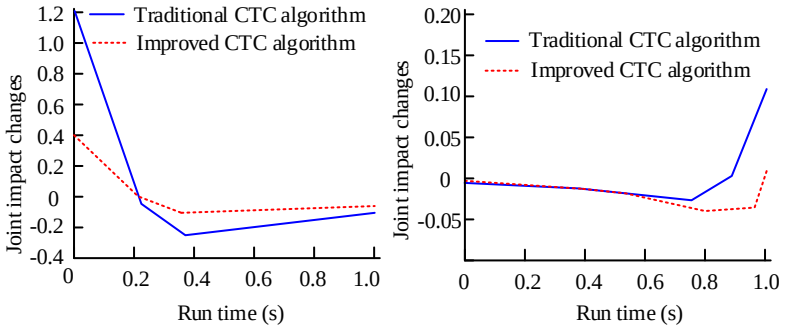
The joint constraint condition table clearly specified the limit ranges for each joint in terms of angle, velocity, and acceleration. The joint constraint conditions for the robotic arm are shown in Table 2.

Table 2

Joint constraints of robotic arm

Joint constraint conditions	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
Angle range (rad)	[-π/2,π/2]	[-π/3,π/3]	[-π/2,0]	[-π,0]	[-π/4,π/4]	[-π/6,π/6]
Speed limit (rad/s)	3.14	3.14	3.14	3.14	3.14	3.14
Upper limit of acceleration (rad/s^2)	6.28	6.28	6.28	6.28	6.28	6.28

Figure 4 illustrates the control of joint impact variation amplitude by the CTC algorithm before and after improvement during TP. “Joint impact” refers to the instantaneous impact load and motion disturbance generated by sudden changes in velocity, acceleration, or trajectory discontinuity during the movement of joints in electromechanical systems. It is a core indicator for measuring the smoothness of joint motion. In Figure 4(a), during the initial motion phase, the traditional CTC algorithm exhibited a large amplitude of joint impact variation with rapid decay, whereas the improved CTC algorithm demonstrated a more gradual change in impact and ultimately approached a stable low value. In Figure 4 (b), the traditional CTC algorithm showed a significant increase in joint impact changes in the later stage, while the improved CTC algorithm maintained stable and relatively low impact values. This indicates that the improved CTC algorithm effectively suppresses joint impact in both the initial and final motion stages, resulting in smoother joint motion.

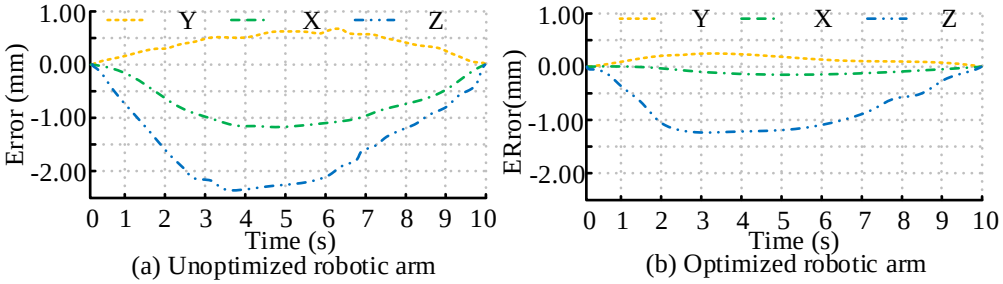


(a) Start period shock time curve (b) End period shock time curve

Fig. 4. Control of joint impact change amplitude in CTC algorithm before and after improvement in TP

3.2. Analysis of robot arm online planning results

Figure 5 presents a comparative plot of the end-effector (EE) error curves for the robotic arm joints. In Figure 5(a), when examining the comparative plot of the EE error curves for the robotic arm joints, it was evident that the unoptimized robotic arm exhibited significant error fluctuations in the X, Y, and Z-axis directions, with the Z-axis error even dropping to approximately -2.5 mm. The overall error variations were severe and covered a wide range. In Figure 5(b), however, the optimized robotic arm demonstrated a noticeable reduction in error fluctuations across all axes, with the lowest Z-axis error reaching around -1.32 mm. Moreover, the errors in each axis tended to be more stable, indicating a significant improvement in the positional accuracy of the EE of the optimized robotic arm. This resulted in enhanced accuracy and stability in motion control, enabling the robotic arm to complete TP tasks with greater precision.



(a) Unoptimized robotic arm (b) Optimized robotic arm

Fig. 5. Comparison chart of joint end error curves of robotic arms

Figure 6 demonstrated the performance of time-optimal online TP for electromechanical systems based on the improved CTC algorithm. It was observable that, in Figures 6(a) to 6(f), the curves representing JAs (q), velocities (v), and accelerations (a) were continuous and smooth. This indicated that the

algorithm effectively planned time-optimal trajectories while satisfying kinematic constraints, such as joint velocity and acceleration limits. Compared to the traditional CTC algorithm, the improved algorithm enabled smoother joint motion through dynamic constraint estimation and adaptive search, avoiding impacts and abrupt changes.

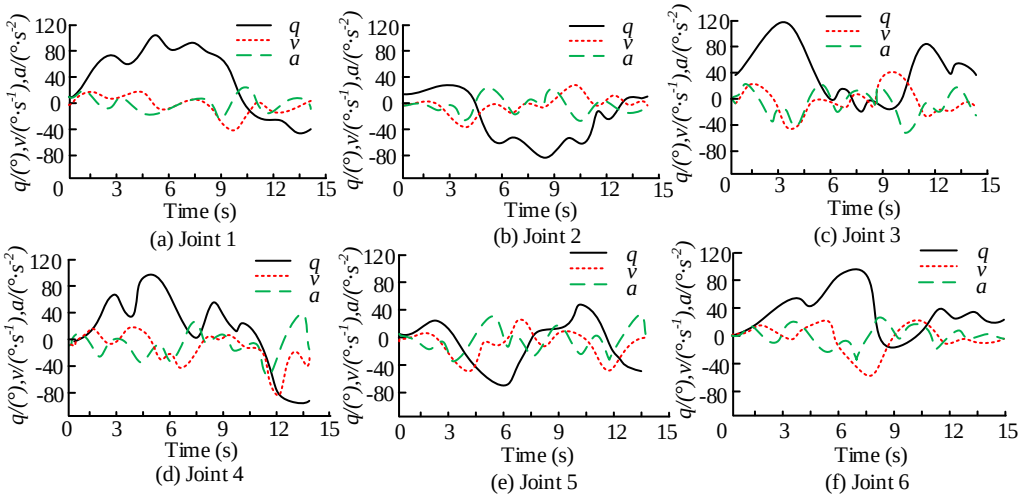


Fig. 6. The effect of time optimal online TP for electromechanical systems based on improved CTC algorithm

Figure 7 shows the variation curves of the angles (unit: rad) of each joint of the robot with running time. In Figure 7 (a), the angles (rad) of different joints (Joint1-6) show different trends with running time (0s-300s): for example, the angle of Joint2 decreases in the early stage, tends to flatten in the later stage, and the angle of Joint3 increases continuously; In Figure 7 (b), after optimization, the overall trend of angle (rad) changes in each joint is similar to Figure 7 (a), but some joints (such as Joint3) have a slower rate of angle growth between 200-300 seconds, and the angle curve is smoother. The stability of joint angle (rad) changes has been improved after optimization.

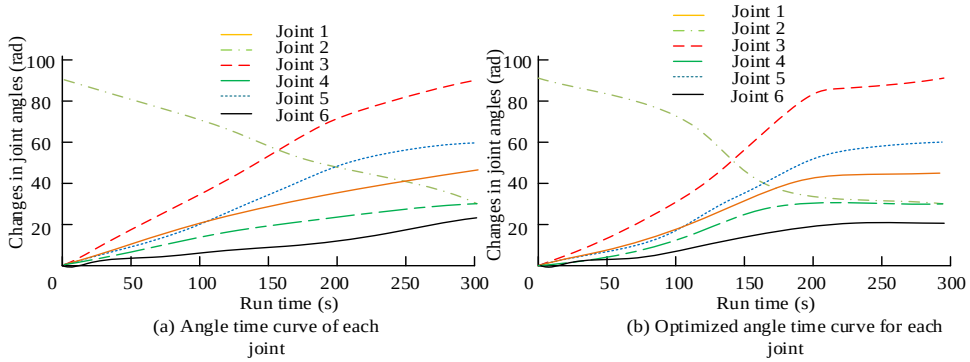
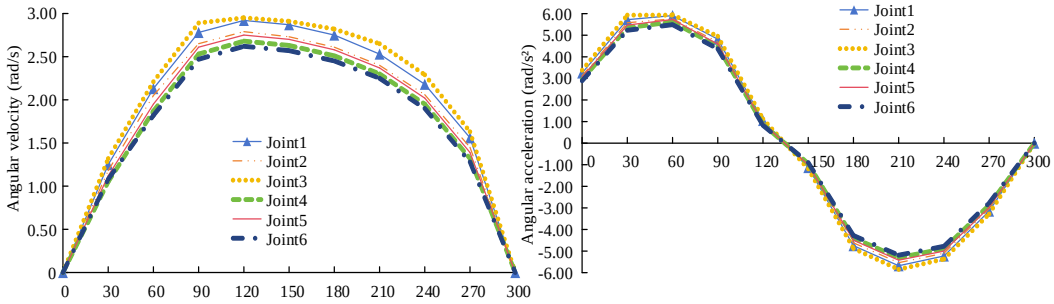


Fig. 7. Curve of robot joint angles changing with running time

To further verify that the improved CTC algorithm meets the kinematic constraints in Table 2, the time series curves of angular velocity and angular acceleration for each joint are shown in Figure 8 (with a time interval of 30 seconds, covering the entire motion cycle from 0 to 300 seconds). It can be seen that the maximum angular velocity of all joints is 2.95 rad/s (Joint 3, 90s), which does not exceed the upper limit of 3.14 rad/s specified in Table 2; The maximum angular acceleration is ± 5.95 rad/s ² (Joint 3, at 60s/210s), strictly within the constraint range of ± 6.28 rad/s ². The continuous curves drawn from the data in the table show that the angular velocity and angular acceleration curves are smooth and without sudden changes, proving that the algorithm fully satisfies the system's kinematic constraints while achieving optimal time planning, ensuring the smoothness of joint motion



(a) Angular velocity (b) Angular acceleration

Fig. 8. Time series curves of angular velocity and acceleration of each joint

The improved CTC algorithm proposed in this article can be directly applied to robot arm trajectory planning in the field of directed energy deposition (DED) additive manufacturing. In the DED process, the robot arm needs to carry a laser nozzle to move along the complex part contour, requiring the trajectory to have time optimality and high smoothness. The algorithm proposed in this article can accurately match the process requirements of DED through dynamic constraint estimation and adaptive search. During the forming process of thin-walled parts, the trajectory acceleration fluctuation is controlled within ± 6.28 rad/s ² to avoid uneven wall thickness caused by nozzle vibration.

4. Discussion and conclusion

A study is conducted on the core problem of the mismatch between the static constraint assumption of traditional CTC algorithm and complex dynamic working conditions in the time optimal online trajectory planning of electromechanical systems. An improved CTC algorithm that integrates dynamic constraint estimation and adaptive search mechanism is proposed, and a complete technical framework of "kinematic constraint modeling dynamic boundary estimation adaptive trajectory optimization" is constructed. Experiments showed

that, in the robotic arm's EE error curve comparison, the unoptimized arm had significant X/Y/Z-axis error fluctuations (Z-axis error down to ~-2.5 mm), with severe, wide-ranging overall errors. Pre-optimization, Joint 1 and 5 angles fluctuated noticeably and unstably; post-optimization, Joint 1 angles rose rapidly, smoothly and stably, while Joint 5 angles also became smoother with better variation patterns. The improved CTC algorithm better met electromechanical systems' demands for time-optimal, smooth and precise TP under complex time-varying conditions, supporting efficient, stable operation. This study enriches the theoretical system of "constrained dynamic modeling optimization solution" for trajectory planning of electromechanical systems, providing a new paradigm for time optimal planning under dynamic constraints. The improved algorithm has good engineering portability and has successfully adapted to the trajectory planning requirements of robotic arms in scenarios such as directional energy deposition additive manufacturing and industrial welding. However, the study had limitations: practical electromechanical systems may be affected by complex factors like dynamics and external disturbances, which were not fully considered in the model and algorithm. Future research could expand the model to include these factors, refining the TP algorithm for better system efficiency in complex real environments.

REFERENCES

[1] *J. Yang, J. Wang, J. Li, X. Meng, X. Jiang, C. Lu,* Sobol sequence RRT* and numerical optimal joint algorithm-based automatic parking trajectory planning of four-wheel steering vehicles, Robot. Auton. Syst., Vol. 186, Iss. 1, 2025, DOI: 10.1016/j.robot.2024.104909.

[2] *Z. Liu, Y. Huang, W.J. Tan,* Trajectory planning of large redundant manipulator considering kinematic constraints and energy efficiency. Robotica: International Journal of Information, Educ. Res. Robot. Artif. Intell., Vol. 41, Iss. 11, 2023, DOI: 10.1017/S0263574723001157.

[3] *C. Xu, W. Qi, Z. Zhang, S. Ge, X. Zhang, G. Wang, F. Peng,* Rapid trajectory planning under multiple constraints in unknown environments, Computing, 2025, Vol. 107, Iss. 4, 2025, DOI: 10.1007/s00607-025-01431-0.

[4] *X. Wang, Y. Cheng, T. Yu, B. Song,* Research on the durability test method of electric driving systems based on fuzzy clustering and particle swarm algorithm, Proc. Inst. Mech. Eng., Part D: J. Autom. Eng., Vol. 238, Iss. 9, 2024, DOI: 10.1177/09544070231167891

[5] *Y. Zhou, G. Han, Z. Wei, Z. Huang, X. Chen, J. Wu,* Optimal trajectory planning of robot energy consumption based on improved sparrow search algorithm, Meas. Contr., Vol. 57, Iss. 7, 2024, DOI: 10.1177/00202940231220080.

[6] *T. Zhao, Z. Yan, H. Liu, B. Zhang, R. Pan, J. Xiao, F. Jiang, S. Chen,* Integration of direct energy deposition systems with an optimized process planning algorithm, J. Ind. Inform. Integrat., Vol. 46, Iss. 1, 2025, DOI: 10.1016/j.jii.2025.100875.

[7] *T. Lou, Y. Wang, Z. Yue, Z. Zhao,* Multi-UAV collaborative trajectory planning for 3D terrain based on CS-GJO Algorithm, Complex Syst. Model. Simul., Vol. 4, Iss. 3, 2024, DOI: 10.23919/CSMS.2024.0013.

[8] *Z. Chen, W. Guan, J. Guo, D. Xue, Z. Liu, G. Wang, L. Quan,* Electric shovel trajectory tracking with inversion sliding mode based on Lyapunov functions, Autom. Constr.: Int. Res. J., Vol. 162, Iss. 6, 2024, DOI: 10.1016/j.autcon.2024.105364.
[9] *O. Saber, B. Yazadan, F. Sirwan,* Time-varying trajectory tracking controller design for cable-suspended planar parallel robots, J. Intell. Robot. Syst., Vol. 108, Iss. 4, 2023, DOI: 10.1007/s10846-023-01936-w.
[10] *G.F. Igneczi, E. Horvath, R. Toth, K. Nyilas,* Curve trajectory model for human preferred path planning of automated vehicles, Automot. Innov., Vol. 7, Iss. 1, 2024, DOI: 10.1007/s42154-023-00259-8
[11] *J. Wang, Z. Chen, R. Chen, Y. Zhao, Y. Bai,* Real-time critical safety curve and predictive control based trajectory planning for complex shaped rotating spacecraft proximity, Adv. Space Res., Vol. 75, Iss. 4, 2025, DOI: 10.1016/j.asr.2024.12.025.
[12] *Y. Wang, Y.Z. Wang, J.J. Xie, H. Dou, S.H. Wu, Y. He, W. Li, C.J. Zhang,* Research on the structural design and path tracking control algorithm of an intelligent road traffic marking robot, Sci. Bull. Ser. D: Mech. Eng., Vol. 87, Iss. 2, 2025, 85-102.
[13] *W. Liu, T.Y. Zhou, C. Jing, Y.H. Ma,* Simultaneous obstacle prediction and real-time adaptive trajectory planning for robotic manipulators in dynamic environments, Sci. Bull. Ser. D: Mech. Eng., Vol. 84, Iss. 3, 2022, 53-68.
[14] *T. Li, S. Meng, C. Lu, Y. Wu, J. Liu,* A novel BIM and vision-based robotic welding trajectory planning method for complex intersection curves, Meas., Vol. 253, Iss. 1, 2025, DOI: 10.1016/j.measurement.2025.117587.
[15] *J. Wang, Z. Li, C. Pan,* Energy-efficient trajectory planning with curve splicing based on PSO-LSTM prediction, Contr. Eng. Pract., Vol. 150, Iss. 1, 2024, DOI: 10.1016/j.conengprac.2024.106009.
[16] *S. Yiit, A. Sezgin,* Trajectory planning with A* Algorithm and tracking with multiple controllers, Opt. Control-Appl. Meth., Vol. 46, Iss. 3, 2025, DOI: 10.1002/oca.3243.
[18] *J. Cheng, S. Hao, R. Wang, F. Wu, J. Yu,* A Method for Kinematic Analysis and Trajectory Planning of a Dredging Robot Based on Screw Theory and Quaternion, J. Mech. Sci. Technol., Vol. 39, Iss. 5, 2025, DOI: 10.1007/s12206-025-0447-3.
[19] *A. Amouri, H. Merabti, A. Cherfia, Y. Laib Dit Leksir,* Nonlinear model predictive control for trajectory tracking of a class of continuum robots, Sci. Bull. Ser. D: Mech. Eng., Vol. 84, Iss. 3, 2022, 19-32.
[20] *Z. Shu, X. Gong, Z. Wei, Z. Huang,* Stable Control for the Planar P-R Type Underactuated Robot Based on Trajectory Planning and Intelligent Algorithm, Lect. Notes Electr. Eng., Vol. 1, Iss. 1, 2025, DOI: 10.1007/978-981-96-2260-3_40.
[21] *Q.C. Zheng, Z. Peng, P.H. Zhu, Y.Y. Zhao, W.P. Ma,* Path optimization for dynamic obstacle avoidance of pioneer robot based on DDPG smart agent, Sci. Bull. Ser. D: Mech. Eng., Vol. 85, Iss. 2, 2023, 29-42.
[22] *H.F. Zhao, Y. Guo,* Design of quick docking method for end effector of industrial robot, Sci. Bull. Ser. D: Mech. Eng., Vol. 85, Iss. 3, 2023, 15-28.
[23] *K. Chen, R. Chai, R. Zhang, Z. Xing, Y. Xia, G. Liu,* A data-driven real-time trajectory planning and control methodology for UGVs using LSTMRDNN, IEEE/CAA J. Autom. Sin., Vol. 11, Iss. 5, 2024, pp. 1292-1294, DOI: 10.1109/JAS.2024.124269.
[24] *X. Wang, Y. Cheng, T. Yu, B. Song,* Research on the durability test method of electric driving systems based on fuzzy clustering and particle swarm algorithm, Proc. Inst. Mech. Eng., Part D: J. Automob. Eng., Vol. 238, Iss. 9, 2024, pp. 2829-2842, DOI: 10.1177/09544070231167891.
[25] *Y. Guo, Y. Liu, B. Wang, P. Huang, H. Xu, Z. Bai,* Trajectory planning framework for autonomous vehicles based on collision injury prediction for vulnerable road users, Accid. Anal. Prevent., Vol. 203, Iss. 8, 2024, DOI: 10.1016/j.aap.2024.107610.