Research on Engineering Structures & Materials
www.jresm.org
MIM Research Group

Research Article

Integrated trajectory planning and tracking control of a parallel mechanism: Embedded real-time implementation and processor-in-the-loop validation

Hao Huang [*,a]

School of Mechanical and Aerospace Engineering, Jilin University, Changchun, China

Article Info

Article History:

Received 19 Aug 2026

Accepted 12 Oct 2026

Keywords:

Parallel mechanism;
Time-optimal trajectory planning;
Disturbance observer;
Integrated planning and control;
Embedded real-time implementation;
Processor-in-the-loop validation

Abstract

For rapid pick-and-place operations using parallel mechanisms, separate trajectory planning and control can lead to mismatched constraints, overly conservative safety margins, and inefficient motor utilization. This study proposes an integrated framework in which both the planner and the controller use the same simplified rigid-body dynamic model. A dynamics-constrained time-optimal planner computes the optimal velocity profile offline and provides online reference values through interpolation at 1 kHz. Computed torque feedforward is combined with a generalized momentum disturbance observer that avoids online inversion of the inertia matrix. The framework is deployed on an ARM and EtherCAT embedded platform and validated through processor-in-the-loop (PIL), in which the controller executes all online algorithms in real time on the embedded hardware while the mechanical system is replaced by a high-fidelity virtual prototype that incorporates distributed rod inertia, joint friction, transmission elasticity, and parameter perturbations absent from the design model. For a standard gate-shaped trajectory, the cycle time decreases from 0.475 s to 0.392 s, a 17.5% reduction, compared with S-curve planning with PD control. The peak tracking error increases by only 0.10 mm or 3.3% (3.16 mm vs. 3.06 mm over 50 runs per group), while RMS error decreases by 34.1% to 1.18 mm. Group B (time-optimal planning with PD control only) shows a maximum error increase of 86.3%. The worst-case control-loop execution time is 121 μs, accounting for 12.1% of the cycle period.

© 2026 MIM Research Group. All rights reserved.

1. Introduction

Delta-type parallel mechanisms are preferred for high-speed pick and place operations in electronics assembly, food sorting, and pharmaceutical packaging due to the high rigidity, low inertia, and high dynamic performance of their closed-loop chain limbs [1]. The benefit of base-actuated configuration lies in having an advantageous payload to moving mass ratio, but the closed-loop constraints lead to strongly non-linear coupled dynamics [2]. High acceleration, cycle time reduction conflicting with accuracy, and approaching drive limits while maintaining accuracy have become one of the most important problems with such machinery [3].

Pick-and-place trajectory generation is moving forward from kinematic motion generation to utilizing the dynamics of the system. S-curve and polynomial motion are extensively used in trajectory generation because of their ease of use and availability of continuous acceleration [4,5] and smooth transition of corners is conducted to avoid kinematic discontinuity in the pick-and-place trajectory [6]. Time-optimal path parameterization involves converting torque limits to path-acceleration constraints and determining the optimum velocity profile from the feasible set [7], with kinematic feasibility established using laboratory prototypes [8]. However, despite the fact

*Corresponding author: Hao_Huang123@outlook.com
[a]orcid.org/0009-0003-8595-0012
DOI: http://dx.doi.org/10.17515/resm2026-2220vk0819rs

that motion laws do not have dynamic data but only empirical torque margins, while time-optimal approaches rely on separate model-based designs, dynamic feasibility of near-boundary trajectories in closed-loop control is not assured.

With regard to control, the application of advanced tracking schemes which deal with both uncertainties within the model and external disturbances of parallel mechanisms has been extensively studied. Among these control schemes include fractional order and adaptive terminal sliding mode control schemes [9,10], model-free adaptive sliding mode control scheme [11], predefined time terminal sliding mode control scheme [12], and model predictive control with embedded reference dynamics [13]. While all these schemes can provide substantial improvements in precision, the one thing that these schemes have in common is the same assumption that the trajectory is considered to be a predefined feasible external reference.

In terms of implementation, parts of the control algorithms are validated through offline and non-real-time simulations, not being associated with any specific implementation architecture, so that there is no execution or scheduling data along with the stated precision [14]. Implementation-focused studies are based on real-time Ethernet technology, using optimized open-source EtherCAT masters [15] and EtherCAT-based synchronization for control of Delta mechanisms [16]. Nevertheless, planning and control are still considered in two separate environments or in two separate phases, and the implementation schemes where both of them run in the same control cycle under the same low-cost embedded architecture are scarce.

To fill this gap, an integrated planning and tracking control framework with real-time implementation capability for the Delta parallel mechanism is presented in this study. First, it provides an integrated framework where both planning and control use one same simplified dynamic model for the rigid body. Second, it provides a two-layer planning method with an embedded generalized-momentum-based disturbance observer that eliminates the need for online inversion of the inertia matrix while retaining the online assembly of the inertia matrix for the momentum computation. All the online operations such as profile interpolation, feedforward calculation, and disturbance compensation are performed in a 1 kHz control loop. Third, it provides a full real-time implementation in an ARM and EtherCAT platform with measured worst-case execution times of all the software components and a processor-in-the-loop experiment with four configurations compared with a high-fidelity virtual prototype. The overall architecture of the proposed framework is shown in Fig. 1.

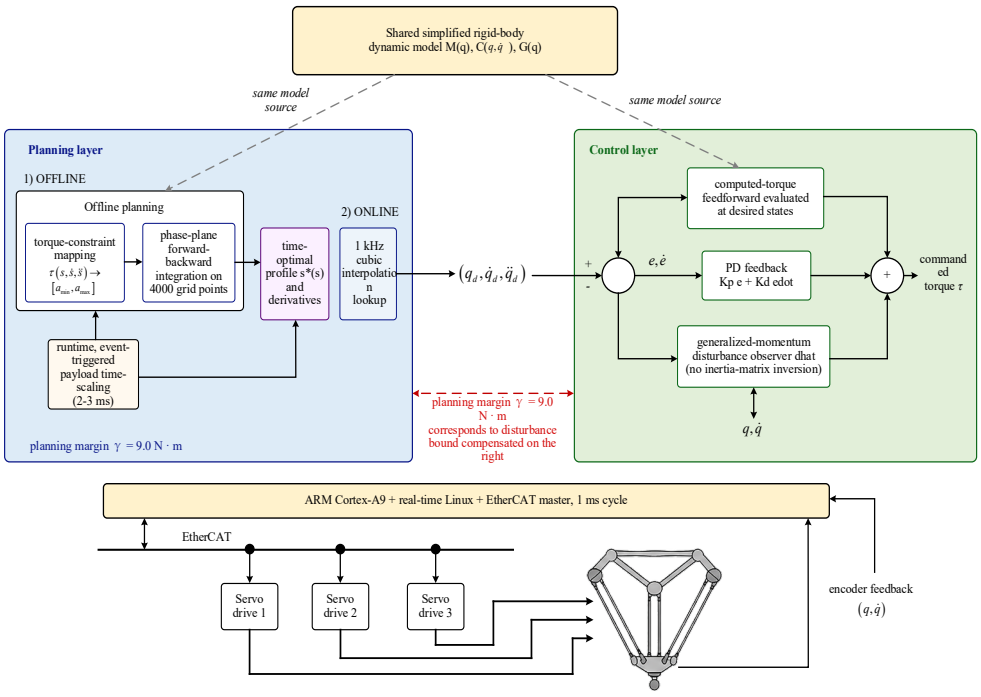


Fig. 1. Overall architecture of the integrated planning and control framework

2. Kinematic and Dynamic Modeling

2.1 Mechanism Description and Coordinate Frames

The object under study is a three-degree-of-freedom translational Delta parallel mechanism, having a fixed base, a movable platform, and three identical arms placed equidistantly at 120° intervals. The mechanism includes an active proximal arm driven by a geared reducer and two distal rods that form a spatial parallelogram ensuring pure translatory motion of the platform [17]. The coordinate system of the base is located at the fixed platform with its z-axis directed downwards (Fig. 2). The values of geometric and inertial parameters selected to match those of experimental robots are shown in Table 1.

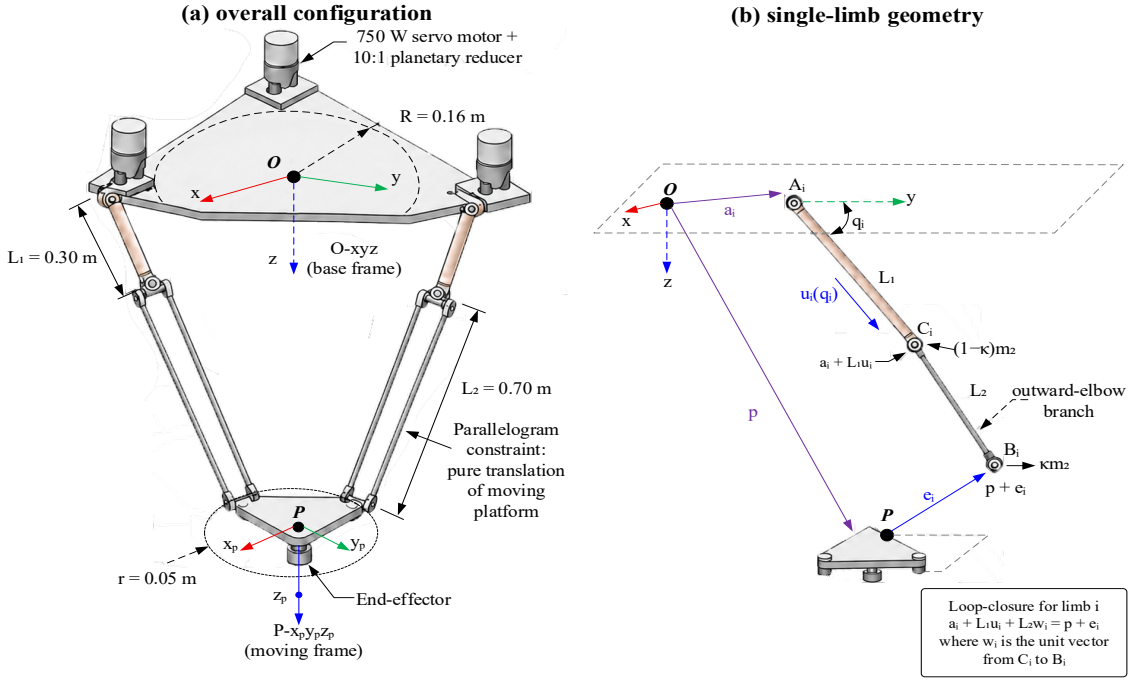


Fig. 2. Schematic of the Delta mechanism and coordinate frames

Table 1. Geometric, inertial, and drive parameters of the Delta virtual prototype (design model)

Parameter	Symbol	Value	Unit	Source
Base platform radius	R	0.16	m	Representative of the class; consistent with the identified prototypes of [17,19]
Moving platform radius	r	0.05	m	Representative of the class; consistent with the identified prototypes of [17,19]
Proximal arm length	L_1	0.30	m	Representative of the class; consistent with the identified prototypes of [17,19]
Distal rod length	L_2	0.70	m	Representative of the class; consistent with the identified prototypes of [17,19]
Moving platform mass	m_p	0.65	kg	Representative of the class; consistent with the identified prototypes of [17,19]

Proximal arm mass (per limb)	m_1	1.10	kg	Representative of the class; consistent with the identified prototypes of [17,19]
Distal rod-pair mass (per limb)	m_2	0.42	kg	Representative of the class; consistent with the identified prototypes of [17,19]
Platform-side mass allocation coefficient	κ	0.5	–	Uniform rod assumption
Joint-side torque limit	τ_{max}	28	N·m	750 W servo, 10:1 reducer
Joint-side velocity limit	$\dot{q}_{max}$	31	rad/s	750 W servo, 10:1 reducer

Note: the parameter set is representative of the 750 W class of Delta mechanisms rather than measured from one specific machine, and is chosen to lie within the range reported for the identified prototypes of [17,19]. Because every quantitative claim of this work is a within-study comparison across four groups that share the identical virtual prototype, the conclusions depend on the internal consistency of this parameter set and, on the perturbation, range applied to it in the processor-in-the-loop configuration described below, not on its correspondence to any particular commercial robot.

2.2 Inverse Kinematics and Velocity Mapping

Let p denote the platform reference position and q the active joint angles. The distance between the proximal tip of the arm and the joint of the platform for the i-th limb equals the length of the distal rod. This ensures the loop closure property

$$\left\| p+e_i-a_i-L_1 u_i(q_i) \right\|^2 = L_2^2, \quad i=1,2,3 \tag{1}$$

where a_i and e_i are the joint positions on the fixed and moving platforms, u_i is the proximal arm unit vector, and L_1 and L_2 are the arm lengths. This reduces to a univariate trigonometric equation having a closed form solution, where the branch is selected based on the configuration of the elbow being outside the workspace. The solution obtained using analytical inversion gives a deterministic minimum cost and forms the basis of embedded time budgeting, while iterative solutions take a non-deterministic time to execute [18]. Therefore, only the inverse solution and velocity mapping are invoked online. Differentiating the constraint with respect to time gives

$$J_q \dot{q} = J_p \dot{p}, \qquad \dot{q} = J \dot{p}, \qquad J = J_q^{-1} J_p \tag{2}$$

where J_q is diagonal and Jp is composed of the limb geometric vectors. The trajectory considered in this section lies only in the central regular workspace, having the Jacobian condition number of less than 4 and, hence well away from the singularities. The online singularity avoidance in neighborhood regions is addressed among the robustness measures below.

2.3 Simplified Rigid-Body Dynamic Model

The joint space equations of motion are obtained on the basis of the virtual work principle [19]. The fact that the slender distal bars have negligible mass means that the mass of these elements is modeled using two concentrated masses located at the elbow and the platform joints. The coefficient for allocating the mass to the platform side is chosen equal to 0.5 for the approximation of the mass distribution to be uniform. Including the reflected inertia of the motors and reducers, the dynamics read:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{3}$$

where the inertia matrix possesses the structure

$$M(q) = I_q + (m_p + 3\kappa m_2) J^T J^{-1} \tag{4}$$

in which I_q is the diagonal matrix of joint-side equivalent inertias, m_p the platform mass, m_2 the distal rod-pair mass, and κ = 0.5 the allocation coefficient consistent with the velocity mapping defined above, while C and G are the Coriolis-centrifugal and gravity terms derived from the

configuration dependence of M. This model sets the limits of the torque, uses feed-forwarding in the control system, and defines the computational budget to be used. Using one model at each end acts as the mathematical connection for the unified system.

To quantify the modeling error introduced by the two-point mass equivalence, the full inertia matrix Mexact(q) is computed numerically by including the distributed rotational inertia of the distal rods and the configuration-dependent inertia of the proximal arms, and compared with the simplified matrix M(q) of Equation (4). The relative Frobenius norm ‖Mexact(q) − M(q)‖F / ‖Mexact(q)‖F is evaluated over 5,000 uniformly sampled configurations within the reachable workspace. The mean relative error is 4.7%, with a maximum of 8.6% occurring near the workspace boundary where the Jacobian condition number exceeds 15. Along the gate-shaped trajectory used in this study, the relative error ranges from 3.1% to 5.3% with a mean of 3.8%. In the PIL configuration of Section 6, the ±5% parameter perturbation is drawn around the simplified (nominal) parameters rather than around the exact ones, so the perturbation and the simplification error are not independent additive terms. Their combined worst case is bounded as follows: the simplification error of at most 5.3% along the trajectory and the ±5% perturbation compound to a relative inertia error of at most 10.6% (1.053 × 1.05 − 1). At the peak commanded acceleration this corresponds to an inertia-related torque mismatch of at most 2.96 N·m, which, together with the 1.07 N·m friction contribution, exceeds the a priori slow-disturbance allocation of 2.5 N·m by about 1.5 N·m in the worst case. This excess is absorbed by the 3.0 N·m feedback allocation of the margin γ, consistent with the measured margin consumption of 8.1 N·m on average and 9.1 N·m in the worst run (Section 6.3). The simplified model is therefore adequate for the present framework; note that the a priori slow-disturbance allocation of Section 2.4 covers friction and the parameter perturbation, whereas the simplification bias is carried by the feedback allocation.

2.4 Boundedness of Unmodeled Terms

The remaining difference between the model and the plant arises from the two-point equivalence assumption, the unmodeled joint friction, and calibration (parameter) errors; the a priori bound below covers the latter two, while the simplification bias quantified in Section 2.3 is absorbed by the feedback allocation of the margin. Calibration error is considered one of the most important factors limiting the model-based control precision [20]. Lumping the unmodeled dynamics and external disturbances together, the physical system satisfies

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + d(t), \qquad \|d(t)\| \le \bar{d} \tag{5}$$

where d(t) collects the slowly varying residuals and is continuous and bounded. Its magnitude is bounded a priori from the component specifications rather than from any closed-loop measurement: the Coulomb level of the joint friction contributes 0.6 N·m, the viscous term contributes at most 0.47 N·m at the 31 rad/s velocity limit, and a ±5% perturbation of the inertial parameters contributes at most 1.4 N·m at the peak commanded acceleration, giving a slow-disturbance bound of about 2.5 N·m. Two further torque components fall outside this slow bound and must be budgeted separately in the constraint mapping that follows: the transient excursions produced by the unmodeled transmission flexibility, which are fast rather than slowly varying, and the feedback torque that the closed loop necessarily superposes on the feedforward whenever a tracking error is present.

3. Dynamics-Constrained Time-Optimal Trajectory Planning

3.1 Task Geometry and Path Parameterization

The trajectory of the test object is along the standard gate test shape, which is widely used in pick-and-place applications. This test shape includes a lift of 25 mm, a span of 305 mm, and a lowering of 25 mm. The tangential discontinuities formed by the transitions between straight lines cause shocks and vibrations in the case of fast motion, and it is shown that corner smoothing is necessary to achieve time-optimal motion planning [21]. Furthermore, the correspondence between the smoothed trajectory and velocity profile defines the quality of the motion [22]. Therefore, quintic transition splines with a cutting distance of 15 mm and zero end curvature are inserted at the two

corners. The trajectory is parameterized using arc length through numerical data points, and the total trajectory attains C^2 continuity. The orientation angle of the gate plane is taken as 15° with respect to the vertical axis from the symmetry plane of the first arm. Let s denote the path parameter and p(s) the end-effector position; the joint configuration q(s) follows from the inverse kinematics, and the joint velocity and acceleration decompose along the path as

$$\dot{q}=q'(s)\,\dot{s}, \qquad \ddot{q}=q'(s)\,\ddot{s}+q''(s)\,\dot{s}^2 \tag{6}$$

where the prime denotes differentiation with respect to s. The planning problem thus reduces to finding the time-optimal velocity profile along a given geometric path.

3.2 Mapping of Constraints onto the Path Parameter

Substituting the above decomposition into the dynamic model derived above, the joint torque becomes a function of the path parameter

$$\tau(s,\dot{s},\ddot{s})=a(s)\,\ddot{s}+b(s)\,\dot{s}^2+c(s) \tag{7}$$

where a(s), b(s), and c(s) are composed of M, C, G together with q'(s) and q''(s), all precomputable along the path. After deducting the disturbance margin, the torque limits are written as

$$-(\tau_{max}-\gamma)\le a_j(s)\,\ddot{s}+b_j(s)\,\dot{s}^2+c_j(s)\le\tau_{max}-\gamma, \quad j=1,2,3 \tag{8}$$

where the subscript j = 1, 2, 3 indexes the torque constraint of the j-th active joint.

The margin γ is a torque budget additively assembled from four terms. The slow disturbance bound established above accounts for 2.5 N·m. The transient excursion excited by the transmission compliance is taken as 2.4 N·m, and the observer compensation channel as 1.1 N·m. The discussion of error sources revisits both figures a posteriori, disabling the corresponding effect in the closed loop and differencing the resulting torque demand, and serves to verify that the two a priori allocations were correctly sized rather than to define them. The remaining 3.0 N·m is reserved for the proportional-derivative feedback that acts on the residual tracking error. The four terms sum to 9.0 N·m, about 32% of the torque limit, and γ is set to this value. For a given pair (s, ṡ), the inequalities are affine in s̈, and intersecting them over the three joints yields the feasible acceleration interval

$$\alpha_{min}(s,\dot{s})\le\ddot{s}\le\alpha_{max}(s,\dot{s}) \tag{9}$$

by which the torque constraints are converted into path acceleration constraints. The joint velocity limit of 31 rad/s is translated through q'(s) into a pointwise upper bound on ṡ, and a Cartesian jerk limit of 500 m/s^3 is imposed to suppress residual vibration of compliant components. The structure of the state-constrained problems with limits on jerk has been formally established [23]. In this case, the jerk limit is approximated through a velocity ceiling formulated below, which retains the forward–backward integration approach with only slight suboptimality.

Strictly, a jerk constraint elevates the state space to three dimensions (s, ṡ, s̈), because the Cartesian jerk along the parameterized path decomposes as $j = p'(s)\,\dddot{s} + 3p''(s)\,\dot{s}\,\ddot{s} + p'''(s)\,\dot{s}^3$, where $\dddot{s}$ is the third path derivative. Rather than extending the phase-plane method to three dimensions, a velocity ceiling is constructed by bounding the jerk contributions of the known path geometry and the feasible acceleration range without relying on $\dddot{s}$. At each grid point s, the convective jerk bound is defined as $j_{conv}(s,\dot{s}) = \max_{\ddot{s}\in[\alpha_{min},\alpha_{max}]} \|3p''(s)\,\dot{s}\,\ddot{s} + p'''(s)\,\dot{s}^3\|$. Because the expression inside the norm is affine in s̈, the maximum is attained at one of the interval endpoints αmin or αmax. The jerk-limited velocity ceiling $\dot{s}_j(s)$ is the largest $\dot{s}\ge 0$ satisfying $j_{conv}(s,\dot{s})\le j_{max}$, which reduces to solving a polynomial inequality in ṡ at each of the 4,000 grid points. The effective velocity ceiling used in the forward-backward integration is then $\dot{s}_{max}(s) = \min\{\dot{s}_{torque}(s), \dot{s}_{velocity}(s), \dot{s}_j(s)\}$.

This projection is a heuristic: by omitting the $\dddot{s}$ term, the convective jerk components are kept within jmax, but the total jerk is not formally bounded because the bang-bang switching of s̈

produces impulsive $\dddot{s}$ at the transition points. A post-verification of the final interpolated trajectory confirms that the maximum Cartesian jerk is 463 m/s^3, which remains below the 500 m/s^3 limit; the margin is provided by the cubic interpolation at 1 kHz, which smooths the switching transients in the discretized profile. For the present gate-shaped trajectory, the jerk-limited ceiling $\dot{s}_j(s)$ is binding only at the two corner transitions, where it reduces the peak velocity by 3.7% compared with the torque-only ceiling, confirming that the suboptimality is minor and that the forward-backward integration approach is retained with negligible performance loss.

To assess the sensitivity of the integrated scheme to the choice of γ, the PIL experiment described in Section 6 is repeated for four values of γ spanning the range from aggressive (4.5 N·m) to conservative (12.0 N·m) allocation, with all other parameters held fixed. Reducing γ to 6.0 N·m shortens the cycle by 6.9% relative to the present value but raises the peak tracking error by about 53% to 4.83 mm and causes 312 saturated cycles across the 50 runs. At γ = 4.5 N·m, the behavior exceeds even that of Group B (1,847 vs. 765 saturated cycles, peak error 8.24 mm vs. 5.70 mm). Increasing γ to 12.0 N·m eliminates saturation entirely but extends the cycle time by 10.0% to 0.431 s, while the tracking improvement over 9.0 N·m is marginal (3.02 mm vs. 3.16 mm). The present allocation of γ = 9.0 N·m therefore lies near the knee of the trade-off between cycle time and tracking precision, representing a near-optimal compromise for the given plant and disturbance characteristics.

3.3 Two-Level Solution Strategy

The time-optimal trajectory is formulated inside the phase plane. Through the process of integration of torque and velocity constraints, the trajectory of maximum velocity is generated. The time optimal trajectory takes form from the lower envelope of both integration trajectories and maximum velocity trajectory, making sure that there is an active constraint at any time instant. In the offline stage, the calculations are performed at 4,000 evenly distributed grid points on the parameter s, and the optimal trajectory as well as the first and second derivatives are saved. In the online stage, the desired quantities (q_d, $\dot{q}_d$, $\ddot{q}_d$) are calculated at a rate of 1 kHz using cubic interpolation and table look-up, with constant computational effort per cycle. The velocity trajectory is scaled by a time scaling factor which is found using one-dimensional search for the tightest torque constraint after the load change. Mass of the payload is introduced into the model solely via the mp term. Therefore, the coefficient values in the constraints along the trajectory are recalculated analytically during the search. The scaling operation is executed event-triggered on the non-real-time core, resulting in a measured cost of 2 to 3 ms that consumes no hard real-time budget; its feasibility is verified experimentally under a payload step in the comparative results. The core idea of time-optimal correction at run-time with guaranteed feasibility is consistent with the well-defined approach in scientific publications [24].

3.4 Self-Check of the Planning Result

The gate-shaped path with its C^2 corner transitions and the phase-plane feasible region, maximum velocity curve, and time-optimal profile are shown in Fig. 3. Fig. 4 illustrates the comparison between the time-optimal control profile and the S-curve profile in terms of joint velocity, acceleration, and torque, for the same trajectory, where the margin-reduced boundary limits are added to the torque profile. The torque profiles associated with the time-optimal trajectory keep changing from one limit to the other depending on whether it is the accelerating or decelerating phase, and all this without ever violating any of the constraints. The S-curve profile, by contrast, reaches its planned peak of 19.6 N·m only briefly and remains well below that level over most of the motion, so that the two profiles differ far more in how long they sustain a high torque than in the height of their planned peaks. The closed-loop torques reported in the comparative results are substantially higher for both, because the feedback channel adds the compensation torque that the reserved margin γ was budgeted for. The cycle-time reduction therefore does not originate from commanding a higher planned peak; it originates from sustaining a torque close to the boundary throughout the motion instead of touching it once. For the current geometry of the task, the joint velocity constraint is not active, because the peak joint velocity is only 5.6 rad/s out of the 31 rad/s limitation, so the torque constraints govern the entire profile.

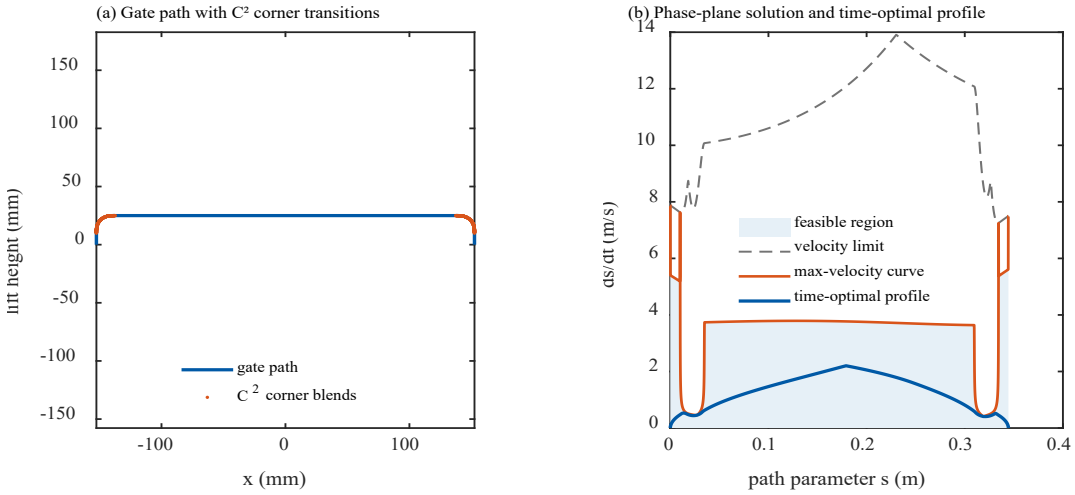


Fig. 3. Task geometry and phase-plane solution

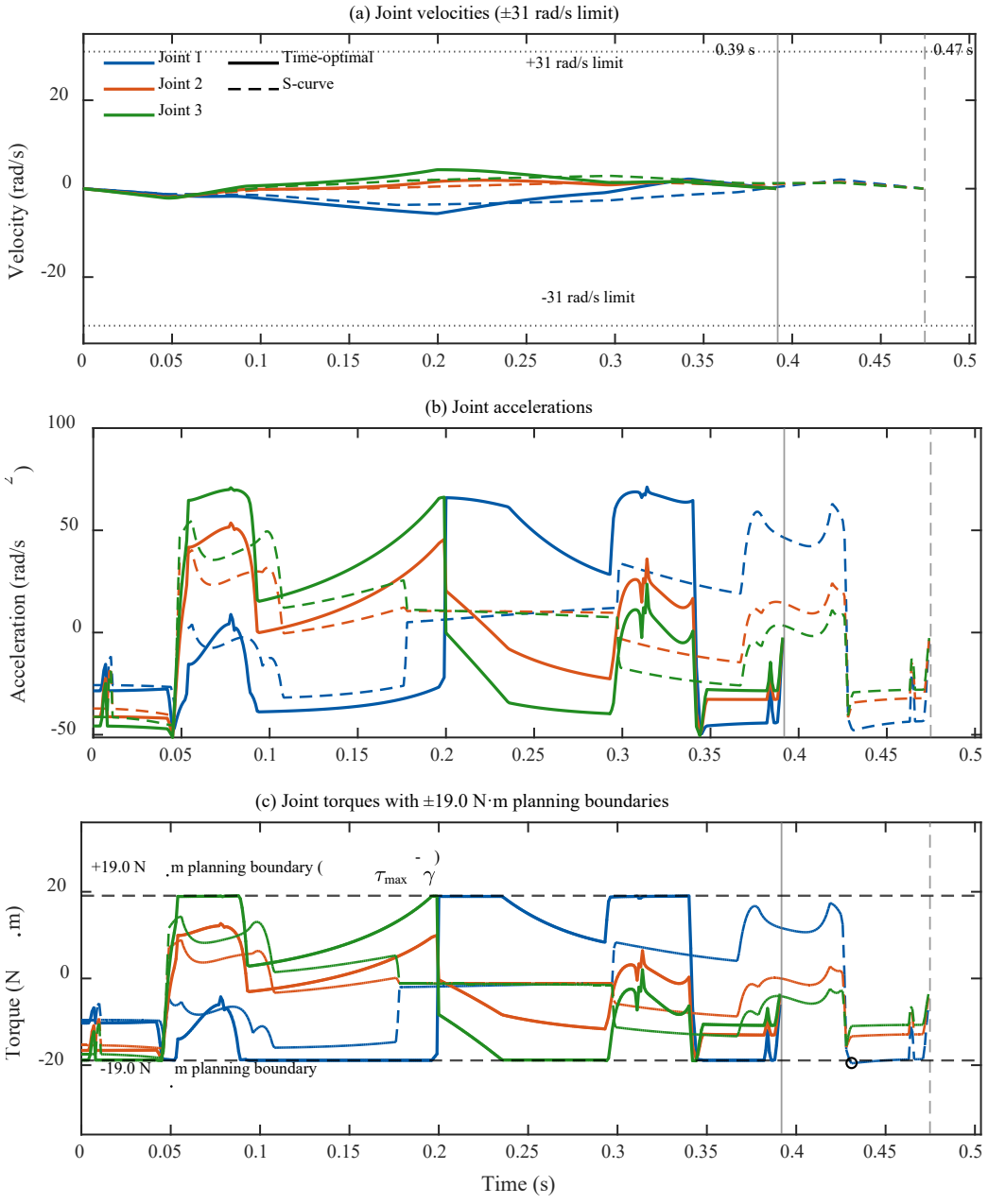


Fig. 4. Planned joint profiles of the time-optimal and S-curve plans

4. Tracking Control Design

4.1 Control Law Structure

Despite the fact that adaptive control and neural-network compensation schemes have excellent ability to deal with uncertainty, the online computational demands of such methods exceed the constraints of the embedded cycle budget [25]. Thus, a deterministic approach combining computed torque feedforward, PD feedback and disturbance compensation is used. Defining the tracking error e = qd − q, the control law reads

$$\tau=M(q_d)\ddot{q}_d+C(q_d,\dot{q}_d)\dot{q}_d+G(q_d)+K_p e+K_d\dot{e}+\hat{d} \tag{10}$$

The three feedforward terms are evaluated at the desired states taken directly from the planning layer's interpolation output, using the shared rigid-body model introduced above. Unlike computed torque evaluated at measured states, desired-state feedforward keeps measurement velocity noise out of the torque command; the resulting model mismatch is bounded whenever the errors are bounded and is lumped into the disturbance handled by the compensation channel.

4.2 Generalized-Momentum Disturbance Observer

Define the generalized momentum p = M(q)q̇. The Coriolis–centrifugal matrix C (q, q̇) is derived from the simplified M(q) in Equation (4) via the Christoffel symbols of the first kind: c_{ijk} = ½ ($\partial m_{ij}/\partial q_k + \partial m_{ik}/\partial q_j - \partial m_{jk}/\partial q_i$). For any smooth, symmetric, positive-definite matrix function M(q), this construction guarantees that Ṁ − 2C is skew-symmetric regardless of whether M(q) represents the full or the simplified dynamics. Indeed, with $C_{ij} = \Sigma_k c_{ijk}\dot{q}_k$, one obtains $(\dot{M} - 2C)_{ij} = \Sigma_k (\partial m_{jk}/\partial q_i - \partial m_{ik}/\partial q_j)\dot{q}_k$, which changes sign when i and j are interchanged. The requirement is only that C be computed from M via Christoffel symbols, which is the standard procedure followed here [19,26]. Exploiting this structural property $\dot{M} = C + C^T$, the momentum derivative satisfies

$$\dot{p}=C^T(q,\dot{q})\,\dot{q}-G(q)+\tau-d \tag{11}$$

which contains neither the joint acceleration nor the inverse of the inertia matrix. The observer is constructed as

$$\hat{d}=K_o\left[\int_0^t\left(C^T\dot{q}-G+\tau-\hat{d}\right)dt-p(t)+p(0)\right] \tag{12}$$

Differentiating this expression and substituting the momentum equation yields the first-order dynamics of the estimation error d̃ = d − d̂

$$\dot{\tilde{d}}=\dot{d}-K_o\tilde{d} \tag{13}$$

so, the observer acts as a first-order low-pass tracker of the disturbance, with the diagonal gain matrix Ko whose entries are the observation bandwidths. The usefulness of generalized momentum in eliminating the requirement to measure acceleration and in avoiding online inversion of the inertia matrix is validated in previous studies [26]. The online computation in each cycle comprises the assembly of M(q) and C (q, q̇) ^{T} from the measured joint angles, the corresponding matrix–vector products, and one numerical integration step; the inertia matrix itself is never inverted, the only linear solve being the 3×3 Jacobian factorization used to form $J^{-T}J^{-1}$ (the same operation listed for the feedforward module in Table 2).

4.3 Closed-Loop Stability

Consider the Lyapunov function candidate

$$V=1/2\,e^TK_pe+1/2\,\dot{e}^TM(q)\dot{e}+1/2\,\tilde{d}^TK_o^{-1}\tilde{d}+\varepsilon\,e^TM(q)\dot{e} \tag{14}$$

where the cross term with a sufficiently small ε > 0 keeps V positive definite, which holds whenever $\varepsilon < \min\{\lambda_{min}(K_d)/\lambda_{max}(M), \sqrt{(\lambda_{min}(K_p)/\lambda_{max}(M))}\}$. The analysis rests on three assumptions: (A1) $M(q)$ is uniformly bounded with $0 < \lambda_{min}(M) \le \lambda_{max}(M) < \infty$ over the traversed workspace, which the condition-number bound of the kinematic model above guarantees; (A2) the disturbance

derivative is bounded, $\|\dot{d}\| \le \mu$, so that the friction and flexibility residuals are slowly varying relative to the observer bandwidth; and (A3) the feedforward mismatch caused by evaluating the model at the desired rather than the measured state satisfies $\|\Delta\| \le k_1\|e\| + k_2\|\dot{e}\| + \delta_0$ with known constants, which follows from the local Lipschitz continuity of M, C and G. Substituting the control law and the observer error dynamics, and using the skew symmetry of $\dot{M} - 2C$ to cancel the inertia-derivative cross terms, the derivative of V along the closed-loop trajectories is bounded above by $\dot{V} \le -\sigma\|z\|^2 + \mu\|z\|/\lambda_{min}(K_o)$, where $z = [e^T, \dot{e}^T, \tilde{d}^T]^T$ and $\sigma > 0$ is guaranteed provided ε is chosen as above and the gains satisfy $\lambda_{min}(K_d) > k_2 + \varepsilon\lambda_{max}(M)$ and $\lambda_{min}(K_p) > k_1/\varepsilon$. The derivative is therefore negative outside the residual set $\|z\| \le \mu/(\sigma\lambda_{min}(K_o))$, so all closed-loop signals are uniformly ultimately bounded for any compact set of initial errors, and the ultimate bound is proportional to the disturbance variation rate μ and inversely proportional to the observation bandwidth $\lambda_{min}(K_o)$, consistent with existing momentum-observer analyses [27].

The stability analysis above assumes that the commanded torque is applied without saturation. Under actuator saturation, the observer update in Equation (12) uses the torque actually applied to the plant, τapplied = sat (τcmd, −τmax, τmax), rather than the commanded torque before clipping. This is implemented by reading the applied torque from the servo drive's process data object at each EtherCAT cycle. Because the observer integrates the applied torque, no integrator windup occurs: the estimation error dynamics of Equation (13) remain valid with τ replaced by τapplied, and the disturbance estimate tracks the actual closed-loop behavior. In Group C, actuator saturation occurs in only 2 of the 19,600 control cycles across the 50 runs, with an unsaturated demand of at most 28.1 N·m against the 28.0 N·m limit. The clipping magnitude is therefore less than 0.4% of the torque limit, and the Lyapunov derivative is violated by a negligible margin for two control cycles, after which the system immediately returns to the nominal operating region of the residual set.

4.4 Discrete Implementation and Gain Tuning

The controller gain and the observer gains are discretized with a sampling period of 1 kHz by using the zero-order hold technique, whereas the observer integrator is calculated through forward Euler approach. The gains are set to Kp = diag {2200} N·m/rad and Kd = diag {42} N·m·s/rad. Considering that the equivalent inertia of joint side is about 0.25 kg·m^2 based on the prototype parameters, the stiffness of the resulting closed loop system provides a natural frequency of around 94 rad/s and a damping factor of about 0.9.

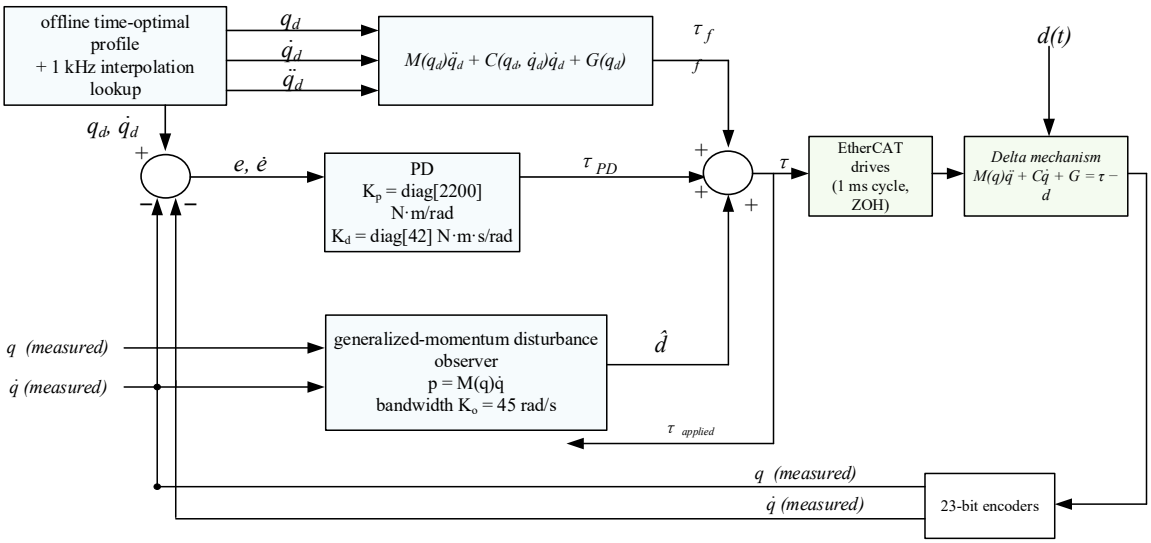


Fig. 5. Block diagram of the tracking control system

The observation bandwidth is set to 45 rad/s, below one fiftieth of the 6283 rad/s sampling angular frequency so that the discretization phase lag stays small, and roughly one eighth of the 346 rad/s first structural flexible mode of the plant, so that no unmodeled elasticity is excited. All bandwidths are quoted as angular frequencies throughout, the flexible mode being 55 Hz in ordinary frequency. The control system architecture diagram is shown in Fig. 5. In the implementation, the torque value

fed into the observer integrand of Equation (12) is the saturated (applied) torque read back from the drive, not the pre-saturation commanded torque, so that the observer state remains consistent with the actual plant input under clipping.

5. Embedded Real-Time Implementation

5.1 Hardware Architecture

The embedded controller employs dual-core ARM Cortex-A9 CPU with a frequency of 800 MHz, which is running under Linux kernel patched with real-time capabilities, whereas the Ethernet interface on board accommodates EtherCAT master. For processor-in-the-loop testing, the three servo drives are emulated by a real-time plant simulator joining the network as slaves in a line topology through ESC-based interface modules; the emulated drives close their current loops locally and advance the high-fidelity model detailed below, exchanging torque commands and encoder feedback cycle by cycle through process data objects. Distributed clock synchronization uses the first slave as reference clock, with a 1 ms control cycle. Real-time Linux provides bounded scheduling latency in ARM architecture under load conditions [28]. Master-slave clock synchronization accuracy determines command simultaneity for multiple axes [29].

5.2 Dual-Rate Software Architecture

The software is organized into two layers, hard real-time and event-triggered. Core 0 runs the 1 kHz control task at the highest real-time priority, pinned by CPU affinity for the whole run; the chain comprises feedback acquisition, interpolation, feedforward and observer computation, and EtherCAT frame exchange. Core 1 is tasked with the event-driven profile scaling and with system health monitoring. The cores share their trajectory through lock-free double buffer mechanism using atomic pointer flip to avoid any locking overheads and ensure safety of segment switch.

5.3 Measured Computational Load

Each module of the control task chain is timed cycle by cycle over 10^6 consecutive cycles, and the statistics are listed in Table 2.

Table 2. Measured real-time performance over 10^6 consecutive 1 ms cycles

Item	Mean (μs)	Max (μs)
Inverse kinematics (analytical)	18	19
Jacobian construction	12	13
Feedforward torque (incl. 3×3 factorization + 2 back-substitutions)	27	29
Generalized-momentum observer	15	16
Interpolation lookup	6	6
EtherCAT frame processing	38	40
Total control-task chain	116	121 (WCET)
Cycle jitter	3	19
Distributed-clock deviation	—	0.8
Deadline overruns (10^6 cycles)	0	0

Note: per-module maxima do not occur simultaneously; the measured chain WCET (121 μs) is therefore smaller than the sum of individual maxima (123 μs).

The feedforward torque computation, the costliest module at 27 μs, includes the assembly of M(qd) along the J−TJ−1 structure with one third-order factorization and two triangular back-substitutions. The chain averages 116 μs with a worst case of 121 μs, occupying 12.1% of the 1 ms cycle and leaving nearly eightfold headroom. The execution-time and cycle-jitter distributions over the 10^6 cycles are shown in Fig. 6.

The 1 kHz control frequency is determined by the plant-simulator slave firmware, which updates its model state and exchanges process data at a fixed 1 ms rate in the present PIL configuration, rather than by the computational budget of the controller. Within this 1 ms

period, the closed-loop natural frequency of 94 rad/s corresponds to a ratio of 1:67 relative to the 6,283 rad/s sampling angular frequency, and the observer bandwidth of 45 rad/s corresponds to 1:140. The zero-order-hold discretization phase lag is less than 2.7° for the closed loop and 1.3° for the observer, both negligible relative to the stability margins. Increasing the loop rate to 2 kHz or 4 kHz is feasible from the master-side computation budget (the 121 µs chain would fit in a 250 µs period) and would reduce the observer discretization phase lag from 1.3° to 0.65° or 0.32°, respectively. However, since the residual tracking error is dominated by the unmodeled transmission flexibility rather than by the sampling rate, the expected accuracy improvement is modest. The transition to a higher rate is planned for the physical prototype, where the servo drives support cyclic-synchronous-torque mode at 0.5 ms or shorter. The remaining 88% CPU headroom at 1 kHz is reserved for future extensions such as online vision-guided replanning that would share the hard real-time core.

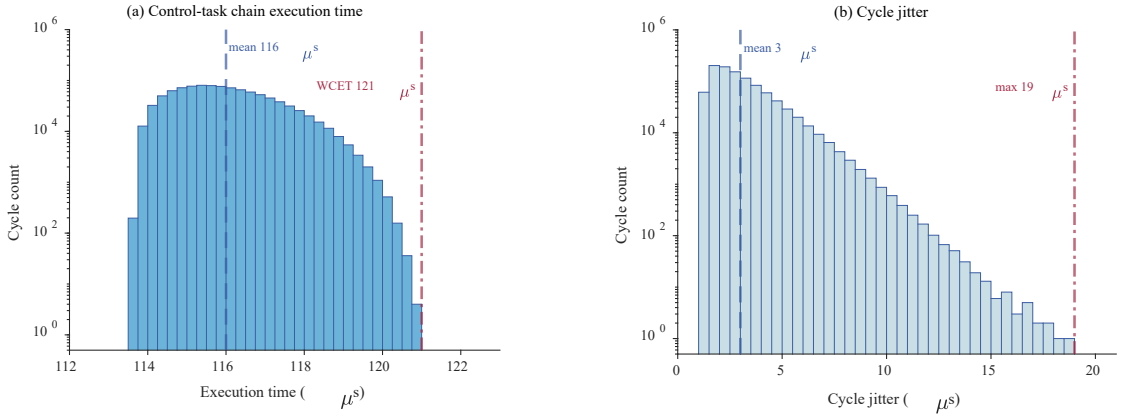


Fig. 6. Measured distributions of the control-task execution time and cycle jitter over 10^6 consecutive cycles

5.4 Robustness Measures and Real-Time Quality

The Jacobian condition number is monitored every cycle, and exceeding a threshold of 30 triggers singularity protection, which decelerates the profile progressively with the growth of the condition number. This criterion is a defensive provision for non-nominal commanded paths such as manual jogging. It is not exercised anywhere in the present study, because the condition number along the nominal gate-shaped trajectory stays below 4 against the threshold of 30, so the mechanism is reported here as an implemented safeguard whose activation behavior remains to be characterized rather than as a validated result. Both cycle jitter and synchronization are determined alongside calculation times, resulting in the highest cycle jitter value of 19 µs and the highest distributed clock offset value of 0.8 µs, with no overruns in 10^6 cycles. The measured 0.8 µs distributed-clock offset contributes less than 0.1% of the 1 ms control period and is absorbed within the cycle-jitter envelope without affecting the control performance.

6. Processor-in-the-Loop Results and Discussion

6.1 Processor-in-the-Loop Configuration and High-Fidelity Virtual Prototype

Validation is conducted processor-in-the-loop: the embedded controller described above executes the online algorithms in real time while the mechanism is replaced by a high-fidelity virtual prototype on the plant simulator, with the 1 ms loop closed over the physical EtherCAT network so that all timing and synchronization effects are those of the real platform. The distal rods carry their full distributed inertia instead of the two-point equivalence. The joints are designed to experience both Coulomb friction and viscous friction, where Coulomb friction equals 0.6 N·m while the viscous coefficient equals 0.015 N·m·s/rad, whose ceiling at the joint velocity limit stays within the lumped disturbance bound established earlier. A lumped transmission elasticity places the first flexible natural frequency at 55 Hz, that is 346 rad/s, well above the 45 rad/s observation

bandwidth of the observer design. The inertial parameters are perturbed independently about their nominal values by draws from a uniform distribution on ±5%, resampled once per run from a fixed random seed sequence so that the four groups are exposed to identical perturbation realizations run by run; the Coulomb friction level is likewise resampled uniformly within ±20% of its nominal value. The command torques are limited to 28 N·m, and joint angles are measured using simulated encoders with 23-bit precision. The plant model is coupled with a fixed step of 20 kHz, which is essentially one twentieth of the control period. The payload is 0.5 kg of standard material, while each configuration completes 50 picks and places.

Angular velocity is estimated by first-order backward difference of the 23-bit encoder position: $\dot{q}(k) = [q(k) - q(k-1)] / Ts$, where Ts = 1 ms. With 23-bit resolution (8,388,608 counts per revolution), the velocity quantization at the maximum joint velocity of 31 rad/s is 7.49×10^{-4} rad/s, or 0.0024% of full scale, and the RMS quantization noise of the backward difference is 3.06×10^{-4} rad/s. No additional low-pass filter is applied to the velocity estimate for two reasons. First, the feedforward torque is evaluated at desired (interpolated) states rather than measured states, so velocity measurement noise does not enter the feedforward channel. Second, the velocity noise propagates into the disturbance estimate through $p = M(q)\dot{q}$ with a gain of at most $Ko \cdot \lambda max(M) \cdot \sigma v \approx 45 \times 0.57 \times 3.06 \times 10^{-4} \approx 0.008$ N·m (where 0.57 kg·m^2 is the maximum eigenvalue of M(q) over the traversed workspace, while the 0.25 kg·m^2 quoted in Section 4.4 is the joint-side equivalent inertia alone), which is negligible relative to the disturbance magnitudes of order 1 N·m. The backward difference itself introduces a half-sample delay of 0.5 ms, but this is already lumped into the discrete-time observer dynamics and contributes 1.3° of phase at 45 rad/s, which is negligible relative to the stability margins.

6.2 Comparative Study Design

Four configurations are compared. Group A is the baseline, combining S-curve planning with PD feedback. Group B replaces the planner with dynamics-constrained time-optimal planning while keeping the PD control unchanged, isolating the contribution of planning. Group D applies the same time-optimal trajectory as Groups B and C, with computed-torque feedforward evaluated at desired states and PD feedback, but without the disturbance observer, representing a standard feedforward-PD controller. Group C is the complete integrated scheme. The four groups share the same virtual prototype, payload, perturbation sampling, and evaluation procedure, so B minus A is attributable to planning, D minus B to feedforward compensation, C minus D to disturbance observation, and C minus A to the integrated scheme.

6.3 Comparative Results

The quantitative indices of the four groups are summarized in Table 3, and the tracking-performance comparison is shown in Fig. 7.

With respect to cycle time, Group A needs 0.475 s, while Group B and Group C each need 0.392 s, implying a 17.5% improvement that is completely due to the planning process and irrespective of the control topology. With regard to accuracy, Group A shows a peak error of 3.06 mm and RMS error of 1.79 mm, a level set by the compliance of the PD loop itself, which must generate the whole dynamic torque from the tracking error. Although the S-curve profile plans a peak of only 19.6 N·m, the closed-loop torque of Group A peaks at 27.4 N·m, or 98.0% of the actuator limit. The 7.8 N·m difference is feedback torque, synthesized from a nonzero tracking error because no feedforward channel exists. Group B degrades most despite the time gain, its maximum error reaching 5.70 mm and its RMS error 2.93 mm, worse than the baseline by 86.3% and 63.7% respectively. The sustained near-boundary torque demand of the time-optimal path intensifies the inertial and coupling torque excitation beyond what the fixed-gain PD loop can supply, and the torque saturates momentarily at the reversals in every one of the 50 runs, 765 clipped cycles in total, with an unsaturated demand averaging 30.1 N·m and peaking at 32.2 N·m against the 28 N·m limit. Group C, on the identical aggressive trajectory, brings the maximum error back to 3.16 mm and the RMS error down to 1.18 mm, the latter 34.1% below the baseline and 59.7% below Group B. Welch two-sample tests over the 50 per-run maxima separate every pair of groups at $p < 0.001$, including the 0.10 mm offset between Group C and Group A ($t = 16.2$, $df = 97$). With a between-run scatter of only

0.03 mm, however, statistical detectability is reached by offsets far below any practically meaningful threshold, so the informative quantity is the size of the offset rather than its significance.

6.3.1 Gate-Shaped Trajectory

Group D, which shares the time-optimal trajectory and the feedforward channel with Group C but omits the disturbance observer, achieves a peak error of 3.11 mm and an RMS error of 1.28 mm. Comparing D with B isolates the contribution of the feedforward channel: the peak error reduction from B to D is 2.59 mm (45.4%), and the RMS reduction is 1.65 mm (56.3%), confirming that desired-state feedforward accounts for the dominant accuracy recovery from the PD-only configuration. Comparing C with D isolates the net contribution of the disturbance observer: the RMS error decreases by a further 0.10 mm (7.8%), while the peak error increases by 0.05 mm because the observer compensation torque of 1.1 N·m raises the total closed-loop demand closer to the actuator limit. The observer's primary role is therefore to compensate the slowly varying friction and parameter-mismatch residuals that affect the mean tracking floor rather than the peak transient error, consistent with its first-order low-pass dynamics.

Table 3. Quantitative comparison of the four groups (50 runs each)

Index	Group A (S-curve + PD)	Group B (Time-optimal + PD)	Group C (Full scheme)	Group D (Time-optimal + FF + PD)
Cycle time (s)	0.475	0.392	0.392	0.392
Cycle time reduction vs. A	–	−17.5%	−17.5%	−17.5%
Maximum tracking error (mm)	3.06	5.70	3.16	3.11
RMS tracking error (mm)	1.79	2.93	1.18	1.28
Change in max error vs. A	–	+86.3%	+3.3% (0.10 mm)	+1.6% (0.05 mm)
Std. of per-run max error (mm)	0.03	0.11	0.03	0.03
Closed-loop peak joint torque (N·m)	27.4 (planned peak 19.6)	28.0 (clipped; planned peak 19.0)	27.1 (planned peak 19.0)	26.0 (planned peak 19.0)
Torque limit utilization	98.0%	100% (saturated at reversals)	96.9%	92.9%
Saturated cycles / peak demand	6 in 3/50 runs; 28.2 N·m	765 in 50/50 runs; 32.2 N·m	2 in 1/50 runs; 28.1 N·m	0 in 0/50 runs; 26.0 N·m

On that measure the two ablations differ in order: Group B raises the peak error by 2.64 mm, 86.3% of the baseline and 88 times the per-run standard deviation of the baseline group, whereas Group C raises it by 0.10 mm, or 3.3%, less than the 0.15 mm spread of the per-run maxima within Group A itself. The supported statement is therefore that Group C holds peak accuracy within 3.3% of the baseline while removing 17.5% of the cycle time. The peak torque of Group C is 27.1 N·m when averaged over the 50 runs, above the 19.0 N·m planning boundary yet below the 28 N·m actuator limit, giving a mean limit utilization of 96.9%. Of the 19,600 control cycles executed across the 50 runs of each group, that is 392 cycles of 1 ms per run, two cycles within a single run reached the limit in Group C, with an unsaturated demand of at most 28.1 N·m, and six cycles across three runs did so in Group A, with an unsaturated demand of at most 28.2 N·m; both remain within 1% of the limit. Averaged over the 50 runs it amounts to 8.1 N·m against the 9.0 N·m budget allocated in the planning stage, a consumption of 90%; in the worst single run it reaches 9.1 N·m and marginally exceeds the budget, which is exactly where the two saturated cycles occur. Fig. 8 shows the

distribution of torque utilization and margin consumption across the 50 runs rather than the mean alone. Stepping the workpiece from 0.5 kg to 1.2 kg raises the platform-side moving mass by a factor of 1.61, and the one-dimensional search returns a time-scaling factor of 1.12, well below the square root of the mass ratio because the joint-side reflected inertia does not scale with the payload. The recomputation completes in 2.6 ms on the non-real-time core, the hard real-time task registers no deadline overrun across the transition, and the rescaled profile settles at a peak torque of 27.5 N·m, inside the actuator limit.

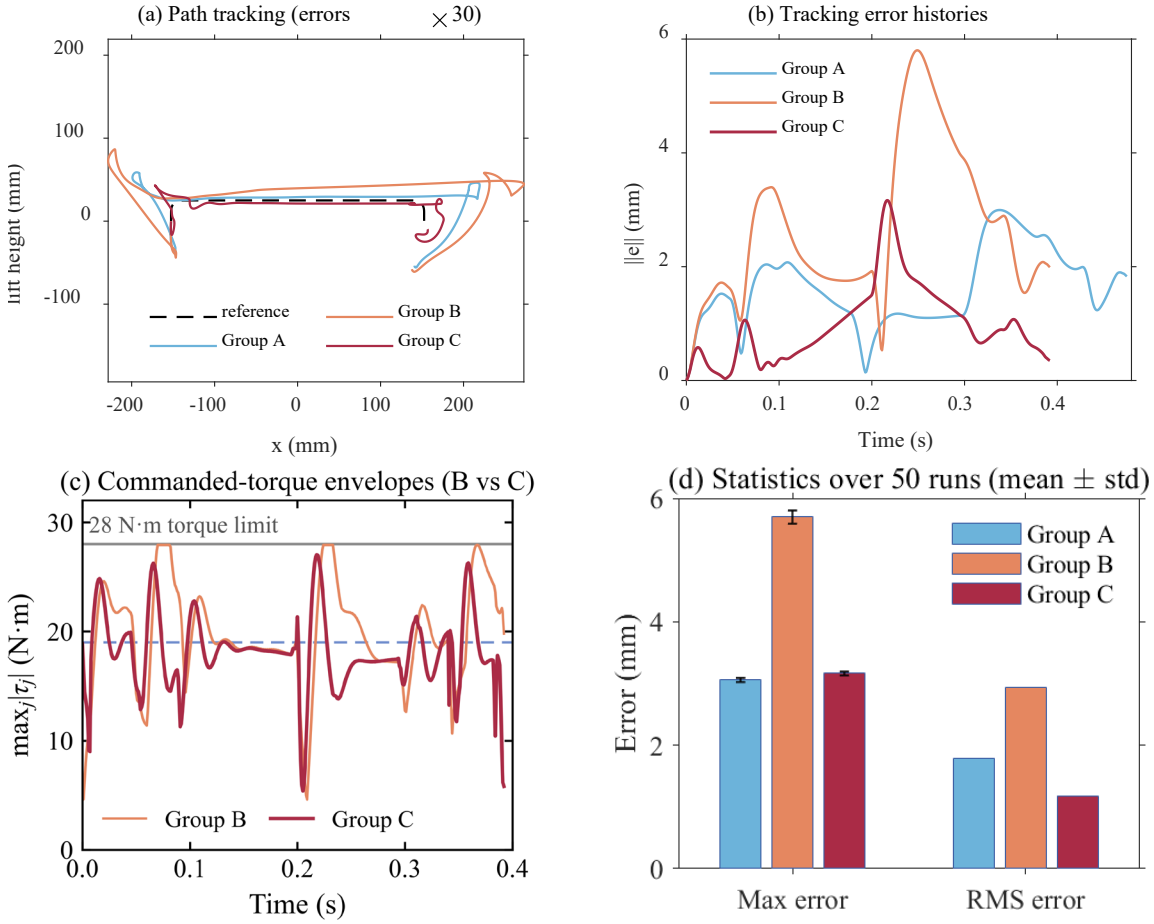


Fig. 7. Tracking performance comparison of Groups A–C under the processor-in-the-loop configuration (Group D results are reported in Table 3)

6.3.2 Semicircular Arc Trajectory

To assess generalization beyond the gate-shaped path, the comparison is repeated on a semicircular arc trajectory: a 180° arc of radius 120 mm in the horizontal plane at a depth of 350 mm, with symmetric 30 mm vertical transitions at both endpoints blended by 12 mm quintic splines, and the arc plane oriented at 60° from the first arm symmetry plane. The total path length is approximately 437 mm, and the acceleration direction rotates continuously along the arc rather than being confined to two discrete corner transitions.

On the semicircular arc, Group A requires 0.538 s with a peak error of 2.74 mm and RMS error of 1.48 mm. Group B reduces the cycle to 0.446 s but increases the peak error to 5.12 mm (+86.9%). Group C achieves the same 0.446 s cycle time (−17.1%) with a peak error of 2.91 mm (+6.2%) and an RMS error of 1.05 mm (−29.1%). The trends are consistent with those of the gate trajectory and confirm that the integrated scheme generalizes across trajectory geometries. The slightly larger relative increase in peak error (+6.2% vs. +3.3%) reflects the higher centripetal acceleration at the arc midpoint, which intensifies the coupling torque excitation and the associated flexibility response.

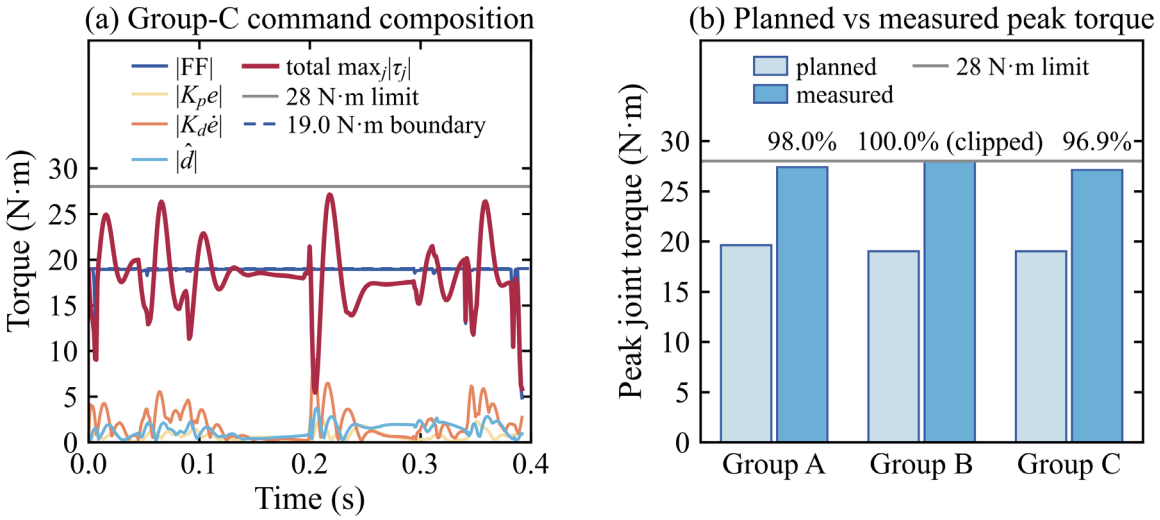


Fig. 8. Torque utilization and margin consumption of the integrated scheme

6.4 Discussion

The error sources of Group C are listed based on their contributions. The dominant source is structural flexibility. Stiffening the transmission so that the first flexible mode moves from 55 Hz to 400 Hz lowers the peak error from 3.16 mm to 2.25 mm and the torque demand from 27.1 N·m to 24.7 N·m, so this factor alone accounts for 29% of the peak error and for the 2.4 N·m flexibility term of the planning margin. Disabling the compensation channel (the Group D configuration) changes the peak error only from 3.16 mm to 3.11 mm while reducing the demand from 27.1 N·m to 26.0 N·m, which both confirms that the residual error is governed by the flexible dynamics rather than by the compensation path and fixes the 1.1 N·m observer term of the same budget. The unmodeled friction is a secondary contributor. It is not isolated by a dedicated diagnostic run here, so its share is reported as a residual bound rather than as a measured value: the two diagnostics above attribute 0.91 mm of the 3.16 mm peak error to transmission flexibility and 0.05 mm to the compensation channel, and the perturbation and discretization terms below account for less than 0.1 mm, which leaves at most 2.20 mm distributed between friction and the kinematic tracking floor of the loop itself. Within that residual, friction is identifiable in the logged error records as characteristic spikes at the velocity reversal of each joint, consistent with the friction-induced errors reported for measured parallel mechanisms [30].

The sampled parameter perturbations and the discretization residual of the 1 kHz implementation rank last, together shifting the statistical indices by less than 0.1 mm across the 50 runs. Several restrictions apply. The rigid-body assumption loses validity at higher accelerations, raising the flexible share of the residual error, and the load-scaling strategy presumes the workpiece mass to be declared by the task command, so unknown or time-varying loads would require online identification. The validation is processor-in-the-loop against a virtual prototype, so real-machine effects absent from the plant model—gear backlash, axis misalignment, cable forces, high-frequency frame vibrations, and non-linear friction characteristics beyond the Coulomb-viscous model—are not captured. Although the controller runs on the target hardware over a physical EtherCAT network with real timing, synchronization, and scheduling effects, the mechanical plant is simulated. Confirmation on a physical Delta robot prototype is therefore essential before the quantitative claims can be transferred to an industrial setting, and is the primary objective of ongoing work. A further scope restriction concerns the comparison baseline. Groups A and B use a fixed-gain PD law, which represents the separated architecture that this work targets but does not represent the state of the art in parallel-mechanism tracking control; the sliding-mode and predictive schemes of [9-13] would very likely narrow the accuracy gap that Group C shows over Group B.

7. Conclusions

An integrated planning and tracking control framework has been developed and implemented in an embedded real-time system to resolve the constraint mismatch that arises when planning and control are designed separately for high-speed parallel mechanisms. The framework is unified by a shared simplified rigid-body dynamic model that serves both the offline time-optimal planner and the online controller, ensuring that the torque budget allocated at the planning stage is consumed consistently by the feedforward, feedback, and disturbance-compensation channels in the closed loop.

Processor-in-the-loop verification against a high-fidelity virtual prototype on two trajectories—a standard gate-shaped path and a semicircular arc—shows that the integrated scheme reduces the cycle time by 17.5% and 17.1%, respectively, compared with the separated S-curve-plus-PD baseline. On the gate trajectory, the peak tracking error increases by only 0.10 mm (3.3%), while the RMS error decreases by 34.1% to 1.18 mm. The four-group ablation study confirms that the feedforward channel accounts for the dominant accuracy recovery (45.4% peak-error reduction from Group B to Group D), and the disturbance observer provides a further 7.8% RMS improvement by compensating slowly varying friction and parameter-mismatch residuals. The ablation group without feedforward (Group B) degrades the peak error by 86.3%, underscoring the necessity of the integrated architecture. A sensitivity study of the planning margin γ demonstrates that the present value of 9.0 N·m lies near the knee of the cycle-time-versus-accuracy trade-off, with lower margins leading to widespread torque saturation and higher margins yielding diminishing accuracy returns.

The two-level planning strategy and the inversion-free generalized-momentum observer compress all online computation into a deterministic worst-case execution time of 121 μs, occupying 12.1% of the 1 ms EtherCAT control cycle and leaving substantial headroom for future algorithmic extensions. The ARM and EtherCAT realization achieve a mean torque-limit utilization of 96.9%, with the reserved margin consumed to 90% of its budgeted value on average and saturation confined to two cycles of one run out of fifty.

The residual tracking error is governed primarily by the unmodeled transmission flexibility and secondarily by joint friction effects. Several limitations apply. The validation is processor-in-the-loop against a virtual prototype; real-machine effects absent from the plant model, including gear backlash, axis misalignment, cable forces, and higher-order structural flexibility, are not captured. The control frequency of 1 kHz is constrained by the plant-simulator slave firmware in the present PIL configuration rather than by the computational budget; the transition to 2–4 kHz is planned for the physical prototype whose servo drives support shorter cyclic-synchronous-torque periods. Confirmation of the reported performance on a physical Delta robot prototype is therefore the primary objective of future work. Additional future directions include modeling the rigid-flexible coupling dynamics within the shared model, online load identification for unknown or time-varying payloads, and leveraging the existing 88% CPU headroom for vision-guided replanning within the hard real-time control cycle.

References

[1] Elghitany MN, Ahmed A, Zaki D, Behhit D, Hosni H, Nour H, et al. Advancements in design, kinematics, and control: a comprehensive review of Delta robot research. Advanced Sciences and Technology Journal. 2024;1(2):1-38. https://doi.org/10.21608/astj.2024.345254.1028

[2] Mahdi SM, Abdulkareem AI, Humaidi AJ, Al Mhdawi AK, Al-Raweshidy H. Comprehensive review of control techniques for various mechanisms of parallel robots. IEEE Access. 2025;13:63381-63416. https://doi.org/10.1109/ACCESS.2025.3557937

[3] Dang HV, Bui TM, Nguyen TVA, Nguyen DH. Experimental validation of dynamic models for high-speed Delta robots. International Journal of Dynamics and Control. 2025;13:146. https://doi.org/10.1007/s40435-025-01651-4

[4] Wu P, Wang Z, Jing H, Zhao P. Optimal time–jerk trajectory planning for Delta parallel robot based on improved butterfly optimization algorithm. Applied Sciences. 2022;12(16):8145. https://doi.org/10.3390/app12168145

[5] Wu G, Zhang S. Real-time jerk-minimization trajectory planning of robotic arm based on polynomial curve optimization. Proceedings of the IMechE Part C: Journal of Mechanical Engineering Science. 2022;236(21):10852-10864. https://doi.org/10.1177/09544062221106632
[6] Su T, Liang X, Zeng X, Liu S. Pythagorean-Hodograph curves-based trajectory planning for pick-and-place operation of Delta robot with prescribed pick and place heights. Robotica. 2023;41(6):1651-1672. https://doi.org/10.1017/S0263574722001898
[7] Li W, Du Y, Dong G. Time-optimal velocity planning of Delta robot dynamic feasible domain based on acceleration constraints. Information and Control. 2024;53(3):329-338. https://doi.org/10.13976/j.cnki.xk.2024.3030
[8] Ayazbay AA, Balbayev G, Orazaliyeva S, Gromaszek K, Zhauyt A. Trajectory planning, kinematics, and experimental validation of a 3D-printed Delta robot manipulator. International Journal of Mechanical Engineering and Robotics Research. 2024;13(1):113-125. https://doi.org/10.18178/ijmerr.13.1.113-125
[9] Alizadeh B, Hajipour A, Tavakoli H, Nasrabadi A. Robust trajectory tracking of Delta parallel robot using fractional-order sliding mode control. IEEE Access. 2023;11:86397-86412. https://doi.org/10.1109/ACCESS.2023.3302525
[10] Zhu D, He Y, Li F. Trajectory tracking of Delta parallel robot via adaptive backstepping fractional-order non-singular sliding mode control. Mathematics. 2024;12(14):2236. https://doi.org/10.3390/math12142236
[11] Fateh A, Momeni H. Model-free adaptive task-space sliding mode control of a Delta robot using a novel reaching law. ISA Transactions. 2024;149:115-123. https://doi.org/10.1016/j.isatra.2024.04.032
[12] Wu P, Zhao PF, Cheng LX, Shi Y, Wang ZY. Research on trajectory tracking control of Delta high-speed parallel robot based on PTNTSMC. Journal of the Brazilian Society of Mechanical Sciences and Engineering. 2024;46(9):562. https://doi.org/10.1007/s40430-024-05075-7
[13] Bettega J, Richiedei D, Trevisani A. Precise path following and trajectory tracking in cable-driven parallel robots through model predictive control with embedded reference dynamics. International Journal of Advanced Manufacturing Technology. 2025;136:3465-3490. https://doi.org/10.1007/s00170-024-14960-3
[14] Zhao A, Toudeshki A, Ehsani R, Viers JH, Sun JQ. Evaluation of neural network effectiveness on sliding mode control of Delta robot for trajectory tracking. Algorithms. 2024;17(3):113. https://doi.org/10.3390/a17030113
[15] Zhang J, Xia M, Li H, Li S, Shi J. The design and real-time optimization of an EtherCAT master for multi-axis motion control. Electronics. 2024;13(15):3101. https://doi.org/10.3390/electronics13153101
[16] Hung CW, Tseng YH, Jiang GY, Song CC. An EtherCAT based Delta robot synchronous control application. Journal of Robotics, Networking and Artificial Life. 2022;9(2):183-186. https://doi.org/10.57417/jrnal.9.2_183
[17] Zheng L, Chen Y. Kinematics and workspace analysis of the Delta robot. IOSR Journal of Mechanical and Civil Engineering. 2023;20(5 Ser III):39-45. https://doi.org/10.9790/1684-2005033945
[18] Zhao A, Toudeshki A, Ehsani R, Sun JQ. Data-driven inverse kinematics approximation of a Delta robot with stepper motors. Robotics. 2023;12(5):135. https://doi.org/10.3390/robotics12050135
[19] Kim TH, Kim Y, Kwak T, Kanno M. Metaheuristic identification for an analytic dynamic model of a Delta robot with experimental verification. Actuators. 2022;11(12):352. https://doi.org/10.3390/act11120352
[20] Urrea C, Domínguez C, Kern J. Modeling, design and control of a 4-arm delta parallel manipulator employing type-1 and interval type-2 fuzzy logic-based techniques for precision applications. Robotics and Autonomous Systems. 2024;175:104661. https://doi.org/10.1016/j.robot.2024.104661
[21] Zhao S, Zheng T, Wang C, Yang Z, Xu T, Zhu Y, et al. Sigmoid angle-arc curves: enhancing robot time-optimal path parameterization for high-order smooth motion. Robotics and Computer-Integrated Manufacturing. 2025;92:102884. https://doi.org/10.1016/j.rcim.2024.102884
[22] Wang Y, Hu C, Li Z, He Z, Lin S, Wang Y, et al. On the consistency of path smoothing and trajectory planning in CNC machining: a surface-centric evaluation. Robotics and Computer-Integrated Manufacturing. 2025;92:102873. https://doi.org/10.1016/j.rcim.2024.102873
[23] Wang Y, Hu C, Li Z, Lin S, He S, Zhu Y. Time-optimal control for high-order chain-of-integrators systems with full state constraints and arbitrary terminal states. IEEE Transactions on Automatic Control. 2025;70(3):1499-1514. https://doi.org/10.1109/TAC.2024.3455382
[24] Wang Y, Hu C, Li Y, Yu J, Yan J, Liang Y, et al. Online time-optimal trajectory planning along parametric toolpaths with strict constraint satisfaction and certifiable feasibility guarantee. International Journal of Machine Tools and Manufacture. 2026;215:104355. https://doi.org/10.1016/j.ijmachtools.2025.104355
[25] Li X, Gao X, Sun L, Zheng D, Shi H, Lei C, et al. The adaptive neural network fuzzy sliding mode control for the 3-RRS parallel manipulator. Advances in Mechanical Engineering. 2022;14(9). https://doi.org/10.1177/16878132221126112

[26] Long S, Dang X, Sun S. An improved adaptive super-twisting momentum observer to estimate external torque for a robot manipulator. Journal of Intelligent & Robotic Systems. 2023;107:25. https://doi.org/10.1007/s108846-023-01814-5
[27] Li Z, Wei H, Liu C, Zhang H, Wei Y, Liu G. A high-order finite-time observer for external force estimation of collaborative robots. International Journal of Robust and Nonlinear Control. 2025;35(8):2952-2967. https://doi.org/10.1002/rnc.7816
[28] Adam GK, Petrellis N, Doulos LT. Performance assessment of Linux kernels with PREEMPT_RT on ARM-based embedded devices. Electronics. 2021;10(11):1331. https://doi.org/10.3390/electronics10111331
[29] Bui TT, Ngo HQT. Enhancing time synchronization between host and node stations using a fuzzy proportional-integral approach for EtherCAT-based motion control systems. PLOS ONE. 2025;20(6):e0324939. https://doi.org/10.1371/journal.pone.0324939
[30] Liu Q, Shen S, Ma Y, Li B. A friction compensation approach of a 6 axis hybrid robot by considering joint inertia change. iScience. 2025;28(4):112173. https://doi.org/10.1016/j.isci.2025.112173