Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.090940 Published Online: 20 November 2026

ARTICLE

Metaheuristic Framework with Learning-Augmentation for Tuning Cascaded Controller Adopted in a Nonlinear Reactor Temperature Regulation

Serdar Ekinci[1,*], Davut Izci[2], Rizk M. Rizk-Allah[3,4], Mahmoud Abdel-Salam[5], Zahraa Tarek[6,7], Beytullah Yağız[8], Mohammad Al-Rabayah[9] and Mohammad Salman[9]

[1]Department of Computer Engineering, Bitlis Eren University, Bitlis, Türkiye
[2]Department of Electrical and Electronics Engineering, Bursa Uludag University, Bursa, Türkiye
[3]Department of Computer Science, VSB-Technical University of Ostrava, Ostrava, Czech Republic
[4]Basic Engineering Science Department, Faculty of Engineering, Menoufia University, Shebin El-Kom, Egypt
[5]Faculty of Computer and Information Science, Mansoura University, Mansoura, Egypt
[6]Department of Computer Engineering and Information, Prince Sattam Bin Abdulaziz University, Al Kharj, Saudi Arabia
[7]Department of Computer Science, Faculty of Computer and Information Science, Mansoura University, Mansoura, Egypt
[8]Department of Science, Van Türk Telekom Science High School, Van, Türkiye
[9]College of Engineering and Technology, American University of the Middle East, Kuwait

*Corresponding Author: Serdar Ekinci. Email: sekinci@beu.edu.tr

Received: 25 August 2026; Accepted: 02 November 2026

ABSTRACT: Temperature regulation in continuous stirred tank reactors (CSTRs) is complicated by nonlinear kinetics, thermal inertia and transport lags, which limit the effectiveness of conventional proportional–integral–derivative (PID)-type schemes. In this study, a series (cascaded) controller composed of a proportional–integral (PI) stage and a proportional–derivative-with-filter (PDN) stage is tuned by an improved golden jackal optimizer (ImGJO), in which Laplacian crossover learning, elite group learning and opposition repair learning strengthen exploration, exploitation, and boundary handling, respectively. A weighted composite cost function balances settling time, overshoot, and the integral of absolute error (IAE). Tuning is carried out on a first-order-plus-dead-time (FOPDT) surrogate whose gain and time constant agree within 2% with those identified from the nonlinear CSTR model over the operating range of interest, and the tuned controller is subsequently verified on the nonlinear model. In simulation, the approach is compared with nine recent metaheuristics under identical settings and with five previously reported CSTR controllers. ImGJO yields the lowest best, mean, median and worst costs over 25 runs, and Wilcoxon signed-rank tests with Bonferroni correction confirm that the differences are statistically significant. With an ideal actuator, zero overshoot is obtained together with the shortest rise and settling times; relative to the metaheuristics, rise time, settling time and IAE fall by up to 46.6%, 15.5% and 17.7%, and relative to the reported controllers, settling time and IAE fall by up to 83.1% and 59.3%. Ablation and robustness tests, together with gain and phase margins of 6.88 dB and 66.1°, are also reported. On the nonlinear model, the unconstrained loop overshoots by 5.9%, whereas coolant-temperature saturation with anti-windup reduces the overshoot to 0.06%; an additional rate limit, however, produces an overshoot of about 7%. The comparative results therefore refer to an idealized actuator, and actuator-aware tuning is identified as the next step.

KEYWORDS: Improved golden jackal optimization; continuous stirred tank reactor; PI–PDN controller; temperature control; time response

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

Temperature regulation in continuous stirred-tank reactors (CSTRs) is confronted with tight safety margins, pronounced thermal inertia and effective transport delays [1–3]. These features arise from the interplay between strongly temperature-dependent kinetics and finite-rate heat exchange, and they degrade the performance of straightforward single-loop tuning. Variable-volume and exothermic operation further accentuates these nonlinear effects and the associated parameter uncertainty [4]. Enhanced PID-family structures, such as two-degree-of-freedom PID [5] and PID-cascade [6] controllers, as well as data-driven tuning based on reinforcement learning [7], have accordingly been examined for stirred-tank heating processes. Real-time fractional-order two-degree-of-freedom control of a process plant [8] and relay-based autotuning of PID controllers for industrial photobioreactors [9] further show that PID-family controllers continue to be preferred in practice for their transparency, while improved structures and tuning procedures are sought to reconcile responsiveness with restraint in the presence of delay.

Three major strands can be distinguished in the literature on CSTR control. First, robust linear designs about operating points have been developed to retain guaranteed margins in the face of model–plant mismatch and variable delay; this thread established that low-order compensators can be effective when loop-shaping is carried out carefully on local models [10]. Second, nonlinear control formulations (including feedback linearization and adaptive/backstepping) have been proposed to address multiple equilibria and parameter drift; these methods widen the region of attraction and offer formal guarantees under stated assumptions but typically require reliable state information and careful gain scheduling [11,12]. Third, model predictive control (MPC) has been employed to handle state and input constraints explicitly; improved transient and constraint management have been reported, though performance depends on model fidelity and on-line computational resources [13].

Recent years have seen a renewed interest in advanced PID-family structures, fractional-order extensions [14–16], and nature-inspired tuning strategies, motivated by the need to improve robustness and transient performance in complex process environments [17]. In parallel, there has been sustained interest in PID-family and cascade solutions tailored to reactor processes. Recent reports include multi-model two-degree-of-freedom (2-DOF) control of a nonlinear CSTR [18] and 2-DOF PID temperature control of CSTRs tuned by quadratic interpolation optimization and differential evolution [19], as well as single-loop designs such as fractional-order PID and PID controllers tuned by nature-inspired optimizers [20]. These studies reaffirm the practical appeal of PID structures while highlighting recurring trade-offs: designs that accelerate the rise frequently incur peaking or noise sensitivity, whereas designs that suppress overshoot often prolong settling and accumulate larger integral errors. Additional journal studies on cascade or advanced PID strategies for CSTRs have emphasized the benefits of coherently tuned inner–outer interactions (e.g., ratio/parallel cascades and noise-aware derivative filtering), again reporting improvements that can depend strongly on the chosen test protocol [21,22]. Adaptive schemes with on-line gain adjustment form a further strand; for example, a Takagi–Sugeno–Kang fuzzy adaptive PID controller combined with static feedforward compensation has been validated experimentally on industrial heating ovens [23]. Such schemes rely on process-specific rule bases, membership functions and, for the feedforward path, a measurable disturbance. Since these elements are not defined for the CSTR considered here, a re-implementation would involve design choices that could not be made on a basis comparable to the other methods, and adaptive fuzzy control is therefore treated in this study as related work rather than as a benchmark.

Taken together, robust linear and nonlinear approaches offer principled guarantees but may require detailed models or on-line adaptation; MPC introduces constraint awareness at additional computational cost; and PID-family and cascade designs remain implementation-friendly, yet are commonly evaluated under heterogeneous step magnitudes, horizons and indices. Two gaps are therefore evident. (i) Joint, closed-loop tuning of both levels of a cascade architecture is not always carried out on a common

surrogate that reflects the inertia-plus-delay character of the reactor. (ii) Optimization objectives are not consistently formulated to balance settling, overshoot and cumulative tracking error within a single interpretable criterion, which complicates fair aggregation and cross-study comparison. A further practical issue is that metaheuristic searches can stagnate or lose diversity when several controller gains must be coordinated simultaneously.

To address these points, a cascaded proportional–integral (PI)—proportional–derivative with filter (PDN) architecture [24] is adopted and tuned as a whole on a first-order-plus-dead-time (FOPDT) surrogate of the CSTR. Following [24], the term cascaded refers here to the series connection of a PI stage and a PDN stage within a single temperature loop (Section 4.1). This structure is selected deliberately. The scenario considered is regulation around a stable operating point, for which the dominant dynamics are well captured by an FOPDT representation (Section 3). Under such conditions, MPC, adaptive or nonlinear controllers, which rely on high-fidelity models, on-line estimation or constraint solvers, offer limited additional benefit relative to their implementation cost. The PI stage provides offset-free tracking, whereas the PDN stage supplies phase lead and filtered derivative shaping of the actuator signal, which is beneficial in lag- and delay-dominated thermal processes. The resulting controller remains compatible with standard industrial control platforms. Keeping the controller structure fixed also allows the contribution of the tuning algorithm to be isolated from that of the control law. The five controller parameters are optimized against a single weighted objective that combines penalties on settling time and overshoot with the integral of the absolute tracking error over a fixed evaluation window.

Tuning is performed with an improved golden jackal optimization (ImGJO) algorithm that augments the baseline golden jackal optimization (GJO) [25] with three complementary mechanisms: Laplacian crossover learning (LCL) [26] to enhance exploration diversity, elite group learning (EGL) [27,28] to intensify exploitation around high-quality candidates, and opposition repair learning (ORL) [29] to handle boundary interactions without collapsing population diversity. The individual and combined contributions of these mechanisms are examined through an ablation analysis.

The study is positioned as a controlled comparative benchmark. Nine metaheuristics (GJO [25], salp swarm algorithm [30], Harris hawks optimization [31], sine cosine algorithm [32], artificial ecosystem-based optimization [33], equilibrium optimizer [34], gradient-based optimizer [35], Runge–Kutta algorithm [36] and artificial rabbits optimizer [37]) are applied under identical population, iteration and bound settings, and their performance is summarized over 25 independent runs and assessed with non-parametric statistical tests. Previously reported CSTR controllers (2-DOF PID, FOPID and single-loop PID designs) are also included, using their published parameters, so that each approach is represented in its intended form. All evaluations are simulation-based. Tuning and comparison are carried out on the FOPDT surrogate with an ideal actuator. The tuned controller is then verified on the nonlinear CSTR model under disturbances and measurement noise, and its behavior under actuator saturation, rate limits and anti-windup is checked on the nonlinear model (Section 5.8). Experimental validation is outside the scope of the present work, and no claim of experimental or industrial performance is made.

In line with the preceding discussion, the main contributions of this work are as follows:

- A jointly tuned series (cascaded) PI–PDN controller for CSTR temperature control on an FOPDT surrogate whose gain and time constant are verified quantitatively against the nonlinear mass and energy balances for the operating range considered (Section 3).
- An ImGJO algorithm that integrates LCL, EGL and ORL to improve search diversity, depth and boundary handling relative to GJO, with the contribution of each mechanism quantified by ablation.
- A weighted composite criterion that jointly accounts for settling time, overshoot and cumulative error, together with an explicit rationale for the weights and the search bounds (Sections 4.2 and 5.5).

- A benchmark against nine metaheuristics and five reported CSTR controllers under common settings, supported by Wilcoxon signed-rank tests with correction for multiple comparisons and by the run-level results and implementation details needed for reproduction. With an ideal actuator, rise time, settling time and IAE are reduced by 1.7%–46.6%, 0.3%–15.5% and 4.7%–17.7%, respectively, relative to the metaheuristics, with zero overshoot; relative to the reported fractional-order and single-loop PID controllers, settling time and IAE are reduced by 43.6%–83.1% and 40.2%–59.3%, respectively.
- A stability and robustness assessment comprising frequency-domain margins, parametric uncertainty, trajectory tracking, an analysis of the controller output, and a verification of the tuned controller on the nonlinear model under disturbances, measurement noise and actuator constraints.

The remainder of the paper is organized as follows. Section 2 describes GJO and the proposed ImGJO. Section 3 presents the nonlinear CSTR model and the FOPDT surrogate, whose parameters are listed in Appendix A. Section 4 details the control architecture, the cost function and the optimization setup. Section 5 reports and discusses the results, and Section 6 concludes the paper.

2 The Proposed Methodology

In this section, the fundamentals of golden jackal optimization (GJO) are first summarized. The algorithm is then extended with three learning strategies, and the resulting variant is referred to as improved GJO (ImGJO).

2.1 Basics of GJO

GJO is a nature-inspired optimization technique based on the cooperative hunting behavior of golden jackals [25]. In GJO, the pursuit behavior is modelled on jackals that move parallel to their prey and eventually overtake it. The hunt comprises two main stages: (1) searching for and approaching the prey, and (2) encircling the prey, harassing it until it stops moving, and finally pouncing on it. A population of jackals is first created (initialization) via Eq. (1):

$$\Delta_{ij} = \Delta_j^{Lb} + U(0,1) \cdot \left(\Delta_j^{Ub} - \Delta_j^{Lb}\right), i = 1, 2, \ldots, N, j = 1, 2, \ldots, n \tag{1}$$

where N and n represent the number of jackals and the number of decision variables, $U(0,1)$ is a uniformly distributed random number in $[0, 1]$. The initial locations are arranged as:

$$Prey = \begin{bmatrix} \Delta_{11} & \Delta_{12} & \ldots & \Delta_{1n} \\ \vdots & \vdots & \Delta_{ij} & \vdots \\ \Delta_{N1} & \Delta_{N2} & \ldots & \Delta_{Nn} \end{bmatrix} \tag{2}$$

$$\Delta_1(t) = \Delta_M(t) - E \cdot |\Delta_M(t) - rl \cdot Prey(t)| \tag{3}$$

$$\Delta_2(t) = \Delta_{FM}(t) - E \cdot |\Delta_{FM}(t) - rl \cdot Prey(t)| \tag{4}$$

where Δ_{ij} defines the jth dimension of the ith jackal. In the prey search phase, the exploration is performed using Eqs. (3) and (4) where t is current iteration, $Prey(t)$ is prey's current location, and Δ_M and Δ_{FM} are the locations of the male and female jackals, respectively. The locations of male and female jackals (renewed ones) are represented by Δ_1 and Δ_2, respectively. The energy (E) related to the evasion of prey resistance is represented as:

$$E = E_0 \cdot E_1 \tag{5}$$

$$E_0 = 2 \cdot r - 1 \tag{6}$$

$$E_1 = 1.5 \cdot (1 - t/T_{\max}) \tag{7}$$

where E_0 represents the initial energy level, E_1 is the prey's diminishing level of energy, r is a random number in $[0, 1]$, and $T_{\max}$ is the maximum number of iterations. The vector rl is given as follows.

$$rl = 0.05 \cdot LF(\Delta) \tag{8}$$

LF is the Lévy flight function:

$$\boldsymbol{LF}(\Delta) = \mathbf{0.01} \cdot (\boldsymbol{\mu} \cdot \boldsymbol{\sigma}) / \left|\boldsymbol{v}^{(1/\beta)}\right|; \boldsymbol{\sigma} = \left\{ \frac{\boldsymbol{\Gamma}(\mathbf{1}+\boldsymbol{\beta}) \sin\left(\frac{\boldsymbol{\pi\beta}}{\mathbf{2}}\right)}{\boldsymbol{\beta} \cdot \boldsymbol{\Gamma}\left(\frac{\mathbf{1}+\boldsymbol{\beta}}{\mathbf{2}}\right) \cdot \mathbf{2}^{\frac{(\beta-1)}{2}}} \right\}^{1/\beta} \tag{9}$$

Finally, the updated location is expressed by means of male and female locations as follows.

$$\Delta(t+1) = \frac{\Delta_1(t) + \Delta_2(t)}{2} \tag{10}$$

In the phase of surrounding and pouncing prey, the exploitation is performed. The mathematical formula for this phase is as follows.

$$\Delta_1(t) = \Delta_M(t) - E \cdot |rl \cdot \Delta_M(t) - Prey(t)| \tag{11}$$

$$\Delta_2(t) = \Delta_{FM}(t) - E \cdot |rl \cdot \Delta_{FM}(t) - rl \cdot Prey(t)| \tag{12}$$

Following this update, the locations of the jackals are once again refreshed using Eq. (10). The pseudo-code for the GJO is evident in Algorithm 1.

Algorithm 1: Main steps of GJO

```
Inputs: Insert population size (N), Limit of Max. iterations (T_max)
Initiate a random population of N preys
while (T_max not satisfied) do
     Assess the prey location using the fitness function
     Obtain the best prey's location (male jackal Δ_M)
     Obtain the second-best prey's location (female jackal Δ_FM)
     Renew the evading energy (E) using Eq. (5)
     Renew the Lévy distribution vector (rl) using Eq. (8)
     for i = 1:N do
If   |E| ≥ 1 then  % Explorative step
     Renew the prey's location by Eqs. (3), (4) and (10)
If   |E| < 1  then % Exploitative step
     Renew the prey's location by Eqs.(10)-(12)
end if
end for
t = t + 1
end While
Output: Display Δ_M
```

2.2 The Proposed ImGJO Algorithm

This section is presented to address the improved variant of the GJO aiming to boost its searching capability. As the exploration phase in the GJO follows the male and female jackals, a lack of diversity in solutions can occur, especially if their locations are far from the promising regions. Moreover, random perturbations of the male and female locations during the exploitation phase can lose the guidance pattern, especially if tiny changes are required in the vicinity of the promising solution, leading to stagnation in local optima. Therefore, to alleviate these deficiencies, three enhancement tactics are developed. First, Laplacian crossover learning (LCL) strategy is presented to emphasize the coverage pattern within the search space and therefore enhance the diversity of solutions. Second, elite group learning (EGL) strategy is introduced to enhance the exploitation capabilities by adaptive deep searching around the elite group of jackals and therefore to boost the quality of the promising solution. Third, opposition repair learning (ORL) strategy is proposed to repair locations that fall outside the search boundaries by using opposite directions, which further increases the diversity of solutions.

2.2.1 The LCL Strategy

LCL is introduced here to enhance the diversity of solutions, thereby improving the exploration capability. The LCL begins with two individuals: one is chosen randomly as Δ_r, and the other is the male jackal found so far (Δ_M). The possible search direction is then determined based on Laplacian distribution:

$$k_j = \begin{cases} a - b\log_e(v_j) & u_j \le 0.5 \\ a + b\log_e(v_j) & u_j > 0.5 \end{cases} \tag{13}$$

where u_j and v_j are random numbers drawn from a uniform distribution in $[0, 1]$, and $a \in \mathbb{R}$ and $b > 0$ are the location and scale parameters of the Laplace distribution, respectively. The new candidate positions are generated as follows:

$$\begin{aligned} \Delta_1^{LC,j} &= \Delta_r^j + k_j \left|\Delta_r^j - \Delta_M^j\right| \\ \Delta_2^{LC,j} &= \Delta_M^j + k_j \left|\Delta_r^j - \Delta_M^j\right| \end{aligned} \tag{14}$$

where $\Delta_1^{LC,j}$ and $\Delta_2^{LC,j}$ denote the new locations of jackals according to LCL for the jth dimension.

2.2.2 The EGL Strategy

The EGL strategy is introduced to mitigate the loss of accuracy that arises from repeating operations and enhance the concentration on promising regions. This strategy establishes the elite group by using the fitness function and identifying its best (most fit) and worst (least fit) members (represented as Δ^P and Δ^W, respectively). The update phase of this strategy may be expressed as in Algorithm 2 where Δ_I^t stands for the current solution with I^{th} index inside the elite group.

2.2.3 The ORL Strategy

An opposition repair learning (ORL) strategy is employed to repair locations and extend the search in opposite directions, so that the diversity of solutions is enhanced:

$$\begin{aligned} M(j) &= \left(\Delta_j^{Lb} + \Delta_j^{Ub}\right)/2 \\ \Delta O_j(t) &= \left(\Delta_j^{Lb} + \Delta_j^{Ub}\right) - \Delta_j(t) \end{aligned}$$

$$\Delta_j(t) = \begin{cases} M(j) + \text{rand} \cdot (\Delta O_j(t) - M(j)) & \text{if } \Delta_j(t) < M(j) \\ \Delta O_j(t) + \text{rand} \cdot (\text{M}(j) - \Delta O_j(t)) & \text{Otherwise} \end{cases} \tag{15}$$

where $\Delta O_j(t)$ stands for opposite solution of current solution $\Delta_j(t)$ of jth dimension.

The structure of the proposed ImGJO is depicted in Fig. 1. The three operators are embedded in the GJO cycle as shown in Fig. 1. After the standard position update of Eq. (10), LCL generates two trial positions from a randomly selected jackal and the male jackal (Eq. (14)). Because the Laplace distribution in Eq. (13) has heavier tails than a Gaussian distribution, most trial steps remain close to the parents while occasional long steps are produced, which sustains exploration. Subsequently, with a probability of 0.2, the EGL operator (Algorithm 2) is applied to the elite group: its weakest member is moved towards the best members, and the move is accepted only if the objective improves, which provides greedy local refinement. Otherwise, ORL is applied: instead of clipping a coordinate to the violated bound, the coordinate is re-sampled between the midpoint of its admissible interval and its opposite point (Eq. (15)), so that feasibility is restored while population diversity near the bounds is preserved.

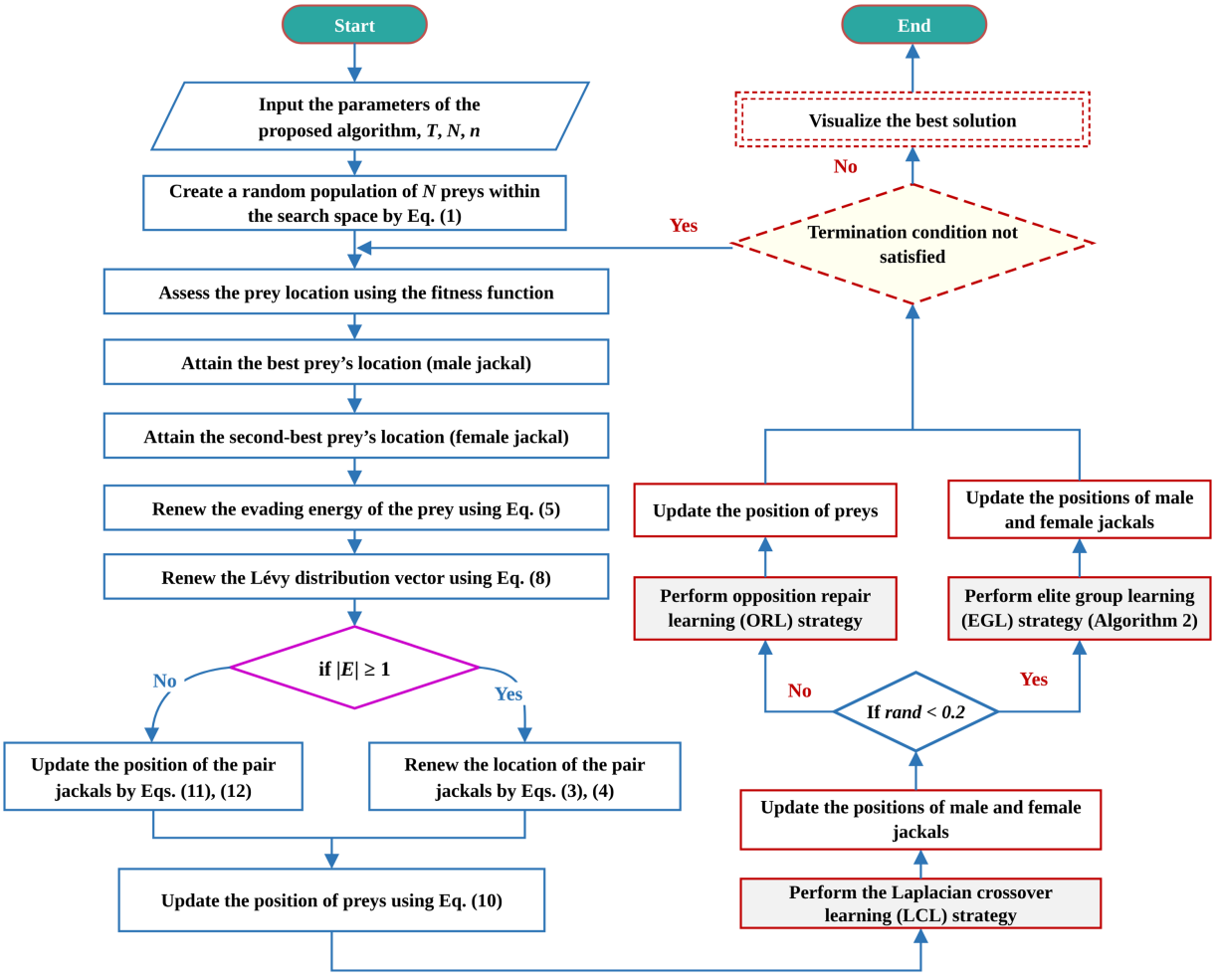

Figure 1: Framework of the suggested ImGJO.

Algorithm 2: The EGL strategy

Inputs: Insert three random numbers (r_1, r_2, r_3), t, Δ_I^t(current location)
$\Delta_1^{EGL} = 2 \times r_1 \times \left(\Delta^P - \Delta^W\right) + \Delta^W$ % Move Δ^W closer to Δ^P
iff $\left(\Delta_1^{EGL}\right) < f\left(\Delta_I^t\right)$
$\Delta_I^t = \Delta_1^{EGL}, f\left(\Delta_I^t\right) = f\left(\Delta_1^{EGL}\right)$
else
$\Delta_2^{EGL} = 2 \times r_2 \times \left(\Delta_{best} - \Delta^W\right) + \Delta^W$ % Move Δ^W closer to Δ_{best}
if $f\left(\Delta_2^{EGL}\right) < f\left(\Delta_I^t\right)$
$\Delta_I^t = \Delta_2^{EGL}, f\left(\Delta_I^t\right) = f\left(\Delta_2^{EGL}\right)$
else
$\Delta_3^{EGL} = 2 \times r_3 \times \left(\left(\Delta^P + \Delta_{best}\right)/2 - \Delta^W\right) + \Delta^W$ % Move Δ^W closer to average of Δ^P and Δ_{best}
if $f\left(\Delta_3^{EGL}\right) < f\left(\Delta_I^t\right)$
$\Delta_I^t = \Delta_3^{EGL}, f\left(\Delta_I^t\right) = f\left(\Delta_3^{EGL}\right)$
end
end
end
Output: Display Δ_I^t

3 Nonlinear Model of Continuous Stirred Tank Reactor

The case study considered in this work is based on a jacketed CSTR in which an irreversible, exothermic, first-order reaction takes place under well-mixed conditions. The reactor is assumed to operate at constant volume, with constant physical and thermodynamic properties, and perfect mixing of both reactants and thermal energy. Heat removal is achieved through an external cooling jacket, which exchanges heat with the reactor contents via a finite-rate heat-transfer mechanism. Such a configuration is representative of a wide class of industrial chemical reactors and has been extensively used as a benchmark in process control studies, particularly for evaluating temperature regulation strategies in the presence of strong nonlinearities and transport delays.

A CSTR undergoing an exothermic, first-order conversion of species A is considered. Under standard modeling assumptions (perfect mixing, constant liquid volume V, and approximately constant density–heat-capacity product δc_p) the macroscopic mass and energy balances are expressed as [38,39]:

$$\frac{dC_A}{dt} = \frac{Q}{V}\left(C_{A,in} - C_A\right) - r(C_A, T) \tag{16}$$

$$\frac{dT}{dt} = \frac{Q}{V}\left(T_{in} - T\right) + \frac{(-\Delta H)}{\delta c_p} r\left(C_A, T\right) - \frac{UA}{V\delta c_p}\left(T - T_j\right) \tag{17}$$

where C_A is the bulk concentration of A in the reactor, T is the reactor temperature, Q is the volumetric flow rate, U is the overall heat-transfer coefficient, A is the heat-exchange area, and T_j is the coolant-side temperature in the external heat-exchange loop. In the control configuration considered, the controlled variable is the reactor temperature, which must be regulated around a desired operating value to ensure safe and efficient reaction conditions. The manipulated variable is the coolant temperature in the jacket, which directly influences the rate of heat removal from the reactor. This choice reflects common industrial practice, where thermal regulation is achieved by adjusting the cooling or heating medium rather than by manipulating reaction kinetics or feed composition. Accordingly, the controller output acts on the

coolant-side temperature, and the resulting heat-transfer dynamics determine the evolution of the reactor temperature. The reaction rate per unit volume is described by [40]:

$$r\left(C_A, T\right) = k_0 e^{\left(-\frac{E_a}{RT}\right)} C_A \tag{18}$$

with k_0 the pre-exponential factor, E_a the activation energy, and R the universal gas constant. These equations capture the essential nonlinearity: heat released by reaction (through $(-\Delta H)/(\delta c_p)$) tends to raise the reactor temperature, while heat removal through the exchange loop (via $-(UA/(V\delta c_p))(T - T_j)$) drives T toward the coolant-side temperature. The inlet concentration and temperature are denoted $C_{A,in}$ and T_{in}, respectively; ΔH is the heat of reaction ($\Delta H < 0$ for the exothermic reaction considered, so that $-\Delta H > 0$); and δc_p is the effective density–heat-capacity product of the reacting mixture, with δ the density and c_p the specific heat capacity. The complete parameter set, with units and sources, and the nominal operating point are listed in Table A1 of Appendix A. Meanwhile, the agitation mechanism is assumed to operate at a constant stirrer speed, as is customary in perfectly mixed CSTR modeling, and therefore does not appear as a dynamic variable in either the nonlinear or linearized representations used in this study.

The nominal parameter values correspond to the benchmark jacketed CSTR of Bequette [41], for which time is expressed in minutes and the hydraulic residence time V/Q equals 1 min. With T_j = 280 K, Eqs. (16)–(18) admit the stable steady state C_A = 0.9774 mol/L and T = 304.17 K, which is used as the initial condition of all nonlinear simulations. This point is taken as the nominal operating condition for regulatory control, since the study focuses on rejecting disturbances and tracking moderate setpoint changes in its neighborhood rather than on wide-range or multi-equilibrium transitions. Linearization of Eqs. (16) and (17) about this point, with the coolant temperature T_j as input and the reactor temperature T as output, yields the second-order transfer function $G_{lin}(s) = 2.0921(s + 1.0231)/[(s + 1.0295)(s + 2.6386)]$ with s in min^{-1}. Both poles are real and negative, which confirms that the selected operating point is locally stable and non-oscillatory. More importantly, the zero at $s = -1.0231\ \text{min}^{-1}$ almost coincides with the slower pole at $s = -1.0295\ \text{min}^{-1}$. The two therefore nearly cancel, and the coolant-to-reactor-temperature pathway is dominated by a single real pole. Consequently, although the balance equations are of second order, a first-order input–output description is appropriate at this operating point. On this basis, the following first-order-plus-dead-time (FOPDT) surrogate is adopted for controller design:

$$G_{cstr}(s) = \frac{0.85e^{-0.0135s}}{0.4355s + 1} \tag{19}$$

In Eq. (19), K = 0.85 K/K is the steady-state gain from coolant-temperature deviation to reactor-temperature deviation, τ = 0.4355 min is the time constant and L = 0.0135 min is the dead time; these values are those of the FOPDT representation of this benchmark adopted in previous studies [19,20,40]. Because the surrogate is used for tuning, its parameters were checked against Eqs. (16)–(18). The coolant step that raises the steady-state reactor temperature from 304.17 to 314.17 K, i.e., the 10 K setpoint change studied in Section 5, is ΔT_j = 11.58 K, which corresponds to a large-signal (chord) gain of 0.864 K/K. A least-squares fit of a first-order response to the nonlinear temperature trajectory produced by this step gives K = 0.864 K/K and τ = 0.428 min, so that the adopted gain and time constant differ from the identified values by 1.6% and 1.7%, respectively. The adopted gain is larger than the small-signal gain $G_{lin}(0)$ = 0.788 K/K, because the heat of reaction raises the effective gain as the temperature increases; the surrogate therefore represents the upward 10 K transition rather than infinitesimal deviations. The dead time is not generated by Eqs. (16) and (17), which contain no transport term. It represents the transport and measurement delay of the coolant and sensing paths assumed in the benchmark [19,20,40], and the same delay is therefore appended to the nonlinear and linearized models whenever these are simulated in closed loop.

The nonlinear model, its linearization and the FOPDT surrogate were also compared quantitatively in open loop for coolant steps of ±5.88 K and ±11.76 K, with the dead time appended to the nonlinear and linearized models. For the upward 10 K transition, the surrogate reproduces the nonlinear step response with a normalized root-mean-square error (NRMSE) of 1.8%, a maximum error of 0.21 K and a steady-state gain error of −1.8%, whereas the linearization gives 5.6%, 0.88 K and −9.0%, because its small-signal gain does not capture the rise of the effective gain with temperature. These figures support the use of the surrogate for tuning. Its accuracy is lower for downward moves, for which the gain is overestimated by up to 13% and the linearization is the better description, and at very short times, where its initial slope $K/\tau = 1.95\ \mathrm{min}^{-1}$ is about 7% below the value $UA/(V\delta c_\mathrm{p}) = 2.09\ \mathrm{min}^{-1}$ of the balance equations. The time unit of minutes is used consistently for τ, L and all time-domain results, and frequencies are accordingly expressed in rad/min. While the FOPDT model in Eq. (19) captures the dominant lag and delay near the selected operating point, it remains a local surrogate of the nonlinear mass–energy balances, and state-dependent effects such as the temperature sensitivity of the reaction rate are attenuated in it.

The closed-loop behavior obtained on the surrogate therefore reflects regulation in the neighborhood of the nominal condition, which is consistent with temperature control of CSTRs operating at fixed setpoints. The consequences of this approximation are examined explicitly: the tuned controller is verified on the nonlinear model, both without and with actuator saturation, rate limits and anti-windup (Section 5.8). The schematic representation of the cross-section of a CSTR is shown in Fig. 2. The primary purpose of the surrogate is to permit systematic, reproducible tuning and a fair comparison of optimization strategies, rather than to represent the full nonlinear operating envelope of the reactor.

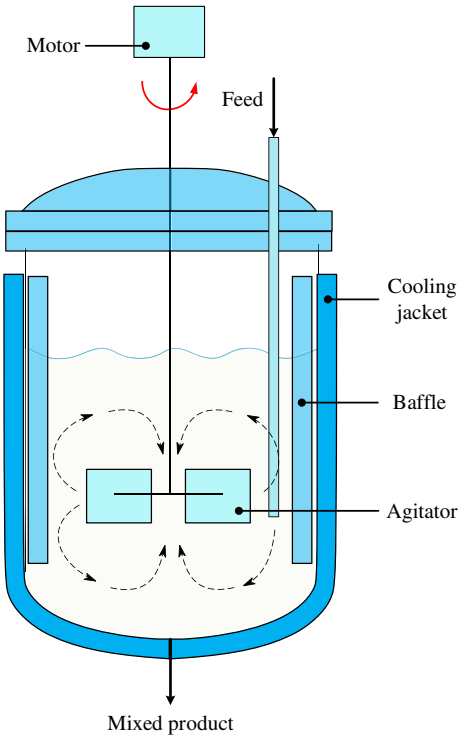

Figure 2: Schematic representation of the cross-section of a CSTR.

4 Proposed Control Architecture

Section 4 presents the proposed control architecture for regulating the CSTR and explains how it is deployed in practice. A series (cascaded) PI–PDN strategy is introduced in which steady-state accuracy is ensured by the PI stage while responsiveness and noise awareness are shaped by the PDN stage. The performance objective guiding the tuning is described in plain terms as a balanced measure of tracking quality and transient behavior, and practical bounds are set so the search remains meaningful. The implementation workflow is then outlined: the controller is tuned in the closed loop by an improved metaheuristic, the evaluation–update cycle is repeated until convergence, and the procedure is validated through multiple independent runs to ensure reliable results.

4.1 The Structure of the Controller

The proposed controller regulates the reactor temperature by manipulating the coolant-side temperature through two controller blocks connected in series, as shown in Fig. 3: a proportional–integral (PI) stage followed by a proportional–derivative stage with a filtered derivative (PDN) [24,42]. In line with [24,42], the term cascade is used in this paper for this series connection of two controller blocks within a single feedback loop. It does not denote a classical two-loop (master–slave) cascade: no secondary process variable is measured, no inner feedback loop is closed, and the only measured and fed-back signal is the reactor temperature T.

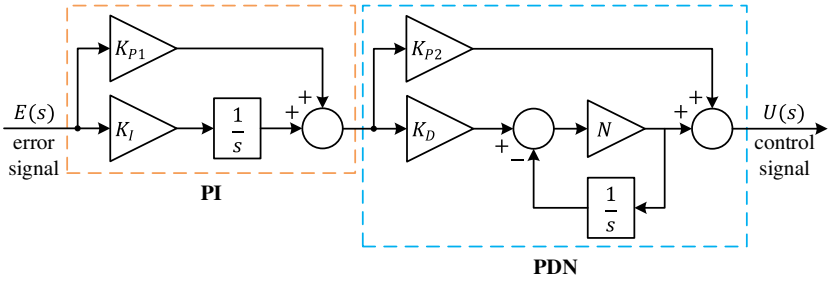

Figure 3: Block diagram of the series (cascaded) PI–PDN controller: The PI stage acts on the temperature error $E(s)$, and the PDN stage acts on the PI output to produce the control signal $U(s)$.

The overall objective is to regulate T so that it follows its reference with negligible steady-state error and limited transient excursion, while maintaining adequate phase margin against the process lag captured by Eq. (19). The signal flow is as follows. The error $e(t) = r(t) - y(t)$, where $y(t)$ is the measured reactor temperature, is processed by the PI stage, which produces the intermediate signal $v(t)$. The PDN stage processes $v(t)$ and produces the manipulated variable $u(t)$, i.e., the deviation of the coolant temperature from its nominal value, which is applied to the plant $G_{\mathrm{cstr}}(s)$. The PDN stage therefore acts as a series lead (filtered-derivative) compensator on the PI output rather than as the controller of a secondary loop. Eq. (20) defines the transfer function $C_{\mathrm{PI}}(s) = V(s)/E(s)$ of the PI stage. It comprises a proportional term, K_{P1}, that reacts immediately to the temperature error and an integral term, K_{I}, that accumulates past error to eliminate steady-state offset.

$$C_{PI}(s) = K_{P1} + \frac{K_I}{s} \tag{20}$$

Eq. (21) gives the transfer function $C_{\mathrm{PDN}}(s) = U(s)/V(s)$ of the PDN stage. It consists of a proportional term, K_{P2}, and a derivative branch, K_D, equipped with a first-order filter whose bandwidth is set by N (min^{-1}).

The PDN output $U(s)$ is the manipulated variable supplied to the process $G_{\mathrm{cstr}}(s)$ in Eq. (19).

$$C_{PDN}(s) = K_{P2} + K_D \frac{N \cdot s}{s + N} \tag{21}$$

Since the two stages act in series on the same error path, the overall controller is the product $C(s) = C_{\mathrm{PI}}(s) \cdot C_{\mathrm{PDN}}(s) = U(s)/E(s)$ [42], i.e., $C(s) = (K_{\mathrm{P1}}s + K_{\mathrm{I}})[(K_{\mathrm{P2}} + K_{\mathrm{D}}N)s + K_{\mathrm{P2}}N]/[s(s + N)]$, which is a filtered PID controller in series (interacting) form with two real zeros, an integrator and a filter pole at $s = -N$. In the time domain, the controller is realized with one integrator state in the PI stage and one filter state in the PDN stage, as depicted in Fig. 3. With the plant model $G_{\mathrm{cstr}}(s)$ of Eq. (19), the loop is closed through the single temperature measurement, and the reference-to-output transfer function $G_{\mathrm{cl}}(s)$ takes the standard feedback form of Eq. (22).

$$G_{cl}(s) = \frac{C(s)G_{cstr}(s)}{1 + C(s)G_{cstr}(s)} \tag{22}$$

4.2 New Cost Function and Adopted Parameter Boundaries

To tune the cascade controller, a single scalar objective is minimized so that competing transient requirements are traded off within one criterion:

$$CF = \sigma_1 n_{st} + \sigma_2 n_{os} + (1 - \sigma_1 - \sigma_2) \int_0^{t_{sim}} |e(t)| dt \tag{23}$$

where n_{st} is the 2% settling time (min), n_{os} is the percentage overshoot, $e(t) = r(t) - y(t)$ is the tracking error (K), and the integral term is the IAE (K · min) over the evaluation horizon $t_{sim} = 2$ min. The rise time, denoted n_{rt} (min, 10%–90%), is not part of Eq. (23) but is reported for comparison in Section 5. The weights are set to $\sigma_1 = \sigma_2 = 0.1$ so that 80% of the weight is assigned to the IAE. The effect of σ_1 and σ_2 on the comparison of the tuned controllers is discussed in Section 5.5.

The three terms are not dimensionally identical. Expressed in the above units, however, they are of comparable numerical magnitude for the responses of interest (settling times of 10^{-2}–10^{-1} min, overshoots of at most a few percent, and IAE values of about 0.2 K · min for a 10 K step), so the weights act as both scaling and priority factors. The largest weight is assigned to the IAE, because sustained deviations from the setpoint directly affect conversion and product quality. With $\sigma_2 = 0.1$, an overshoot of 1% adds 0.1 to the cost, which is about half of the IAE contribution of a well-tuned response. Overshoot is therefore strongly penalized, in line with the need to avoid temperature peaks in an exothermic reactor. The settling-time term, whose contribution is an order of magnitude smaller, mainly discriminates between candidates with similar IAE and no overshoot. Because the IAE is integrated over the whole window, trajectories that rise promptly and settle cleanly are favored over those that merely reach the setpoint by the end of the window.

The horizon $t_{sim} = 2$ min was chosen so that the complete closed-loop transient of every controller considered in this study is contained within the integration window. It corresponds to about 4.6 open-loop time constants of the surrogate, about 33 times the settling time of the proposed design and more than five times the longest settling time among all compared controllers. The tracking error is therefore negligible at the end of the window for all controllers, and extending the horizon would change the IAE values only marginally without affecting the ranking. A substantially shorter window, in contrast, could truncate the tails of the slower responses. The same window is used for all controllers and for the disturbance scenario adopted in this study.

The optimization is carried out over the search region specified in Table 1, and the same bounds were applied to all algorithms. The bounds were fixed before the comparative runs on the basis of preliminary

simulations and of three considerations. (i) All lower bounds are strictly positive, so that the five actions of the PI–PDN structure are retained in every candidate. (ii) The upper bound $N \leq 1000$ min^{-1} keeps the time constant of the derivative filter at or above 0.001 min, i.e., more than an order of magnitude below the dead time L, and thereby caps the high-frequency gain $K_{P1}(K_{P2} + K_D N)$ of the controller and the derivative kick on setpoint changes. (iii) The remaining upper bounds were chosen large enough to contain the well-damped region of the parameter space, but not so large that the search is dominated by unstable candidates. The last point was quantified by Monte Carlo sampling of 2×10^4 uniformly distributed parameter vectors: 83.5% of the box in Table 1 yields a closed loop that is unstable (79.7%) or not settled within the evaluation window (3.8%; error above 0.2 K after 1.5 min), and this fraction increases to 95.4% and 99.6% when all upper bounds are multiplied by two and five, respectively. Excessively wide bounds would therefore spend most of the evaluation budget on infeasible candidates. At the same time, the selected bounds do not dictate the tuning outcome. Of the five parameters of the ImGJO design obtained in Section 5.4, K_{P1}, K_I, K_{P2} and K_D lie at 10%, 24%, 39% and 34% of their admissible ranges, and only N lies close to its upper bound. With the other four gains fixed, raising N from 992.66 to 2000 and 5000 min^{-1} lowers the cost only from 0.1667 to 0.1665 and 0.1664, whereas the initial controller output rises from 1778 to 3321 and 7919 K. The bound on N thus acts as the intended limit on the filter bandwidth and on kick amplification rather than as an active restriction on performance. Within the selected limits, the cost function further supports stable tuning: oscillatory and divergent responses receive high cost values, and the search is steered towards well-damped parameter combinations.

Table 1: The boundaries adopted for the controller parameters during the optimization process.

Bound	K_{P1}	K_I	K_{P2}	K_D	N
Lower	0.1	0.1	0.1	0.005	5
Upper	20	20	30	0.2	1000

4.3 The Implementation of the Proposed Control Architecture

The proposed control architecture is executed as an optimization process wrapped around the closed-loop CSTR control system, as summarized in Fig. 4. A reference signal $R(s)$ is compared with the measured output $Y(s)$ to produce the error $E(s)$, which the PI–PDN controller converts into the actuator command $U(s)$ applied to the plant model of Eq. (19). The controller is parameterized by the vector $[K_{P1}, K_I, K_{P2}, K_D, N]$, whose elements are adapted by the ImGJO outside the loop. The search is conducted with a population size of 30 and a maximum of 100 iterations, and 25 independent runs are carried out for every algorithm considered in the comparisons. Within each run, candidate solutions that leave the admissible box of Table 1 are returned to it: in ImGJO this is performed by the ORL operator, whereas the competing algorithms use the boundary-clipping rule of their original implementations.

Each candidate is evaluated by simulating the closed-loop response to a 10 K reference step from zero initial conditions (deviation variables) over $t_{sim} = 2$ min. The simulations were performed in MATLAB/Simulink R2026b. The 2% settling time, the overshoot and the 10%–90% rise time are computed from the sampled response, and the IAE is obtained by trapezoidal integration. As an independent check, all costs and time-domain indices in Sections 5.5 and 5.12 were recomputed with a fixed-step implementation ($\Delta t = 10^{-5}$ min) in which the plant and the controller are discretized exactly under a zero-order hold, the dead time is realized as a delay of $L/\Delta t = 1350$ samples, and the fractional operators of the FOPID controller are realized with a fifth-order Oustaloup approximation over 10^{-3}–10^3 rad/min. The settling times, overshoots and IAE values were reproduced to within 0.001 min, 0.002% and 0.0003 K·min, respectively, and the same

implementation was used for the nonlinear verification of Section 5.8, the bound analysis of Section 4.2 and the model comparison of Section 3.

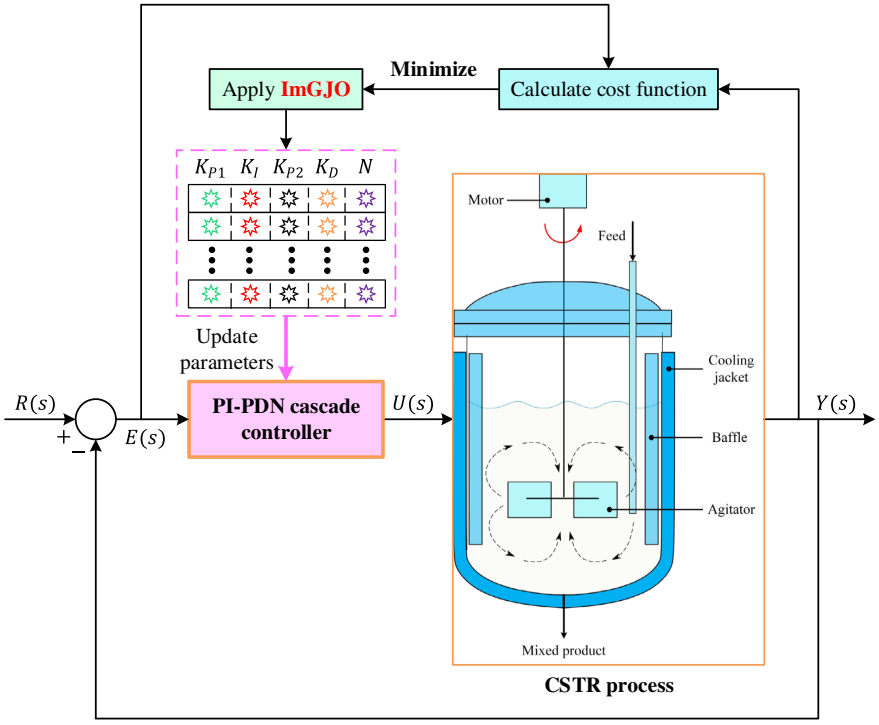

Figure 4: Implementation of the proposed tuning procedure: The ImGJO updates the controller parameters [K_{P1}, K_I, K_{P2}, K_D, N], each candidate is evaluated in a closed loop with the CSTR model, and the cost function of Eq. (23) is returned to the optimizer.

The series PI–PDN controller admits a clear physical interpretation. The PI stage provides offset-free tracking of the reactor-temperature reference, whereas the PDN stage shapes the coolant-temperature command and supplies phase lead to compensate for the dominant lag and delay of the thermal pathway. The first-order filter on the derivative limits the amplification of high-frequency measurement noise, although, as shown in Section 5.8, the large high-frequency gain of the tuned controller still transmits considerable noise to the manipulated variable. The design is developed within a regulatory framework centered on a fixed operating point, and the controller parameters are optimized on the FOPDT surrogate of Section 3, which is a local approximation of the nonlinear CSTR model. All comparisons in Section 5 assume an ideal actuator, and their results should be read with this assumption in mind; the effect of saturation, rate limits and anti-windup on the proposed controller is examined in Section 5.8.

5 Results and Discussion

The performance of the ImGJO-based PI–PDN controller is assessed in this section through statistical analyses, convergence behavior, transient response, control-signal behavior, robustness tests, and comparisons with metaheuristic baselines and previously reported CSTR controllers. All results are obtained in simulation. Sections 5.1–5.7 and 5.9–5.12 use the FOPDT surrogate of Eq. (19) with an ideal (unconstrained) actuator, and Section 5.8 uses the nonlinear model of Eqs. (16)–(18), including a check of the proposed controller under actuator constraints. Throughout this section, times are expressed in minutes and frequencies in rad/min.

5.1 Initial Statistical Evaluation

An initial statistical comparison was conducted between the proposed ImGJO and nine competing algorithms (golden jackal optimization (GJO) [25], salp swarm algorithm (SSA) [30], Harris hawks optimization (HHO) [31], sine cosine algorithm (SCA) [32], artificial ecosystem-based optimization (AEO) [33], equilibrium optimizer (EO) [34], gradient-based optimizer (GBO) [35], Runge–Kutta (RUN) algorithm [36] and artificial rabbits optimizer (ARO) [37]) using repeated runs to reveal both the central tendency and the dispersion of the objective values (lower is better). To ensure a fair assessment, every algorithm was run with the default parameter values recommended in its original publication, without additional tuning, and the additional coefficients of ImGJO were likewise fixed *a priori*, so that no algorithm benefits from problem-specific tuning. The same controller parameter bounds listed in Table 1 were applied to all methods, and 25 independent runs were conducted per algorithm. Tuning the parameters of individual algorithms could change the absolute values in Table 2, but the common population size, iteration budget, bounds, and number of runs ensure that all methods are compared under identical conditions.

Table 2: Comparative numerical statistics.

Algorithm	Best	Mean	Median	Worst	Standard Deviation
ImGJO	0.1667	0.1757	0.1752	0.1907	0.0062
GJO	0.1749	0.1857	0.1844	0.2019	0.0070
SSA	0.2083	0.2228	0.2211	0.2491	0.0105
HHO	0.2012	0.2118	0.2086	0.2323	0.0087
SCA	0.2185	0.2324	0.2321	0.2503	0.0083
AEO	0.1914	0.2027	0.2001	0.2242	0.0086
EO	0.1996	0.2120	0.2098	0.2334	0.0082
GBO	0.2005	0.2114	0.2114	0.2277	0.0071
RUN	0.1872	0.1950	0.1943	0.2067	0.0050
ARO	0.1809	0.1913	0.1902	0.2034	0.0056

The boxplots in Fig. 5 place the proposed method's distribution visibly below the others, with a compact interquartile range and short upper whiskers, indicating consistently low costs across trials; competing methods exhibit higher medians and wider spreads, reflecting less reliable outcomes. The numerical summary in Table 2 corroborates this visual impression. ImGJO attains the lowest best (0.1667), mean (0.1757), median (0.1752), and worst (0.1907) values among all contenders, demonstrating strong performance not only on average but also in the tails of the distribution. Relative to its baseline counterpart GJO, the mean cost is reduced by about 5%, with comparable improvements in the best and worst cases, underscoring the net benefit of the enhancements. Although ImGJO's standard deviation (0.0062) is not the smallest (RUN (0.0050) and ARO (0.0056) are slightly tighter) its variability remains low while its central tendency is markedly better, yielding a more favorable overall trade-off between performance and consistency. In Fig. 5, the central line of each box marks the median, the box edges the 25th and 75th percentiles, and the whiskers the most extreme values not regarded as outliers; the red "+" markers denote outliers, i.e., runs whose final cost lies more than 1.5 times the interquartile range beyond the box.

The computational cost of the algorithms was also evaluated under identical settings (population size of 30 and 100 iterations), and the average execution time per run was recorded over the 25 independent runs. All runs were executed sequentially in a single MATLAB session on a Windows 11 computer with a 13th Gen Intel i7 (2.40 GHz) processor and 16 GB of RAM. The average execution times are 83.7881 s for GJO, 85.5713 s

for GBO, 85.9674 s for ImGJO, 89.2413 s for ARO, 90.8816 s for SSA, 91.6018 s for SCA, 92.1600 s for RUN, 93.2373 s for AEO, 96.7660 s for EO and 103.5855 s for HHO. The computational cost of ImGJO is therefore competitive: it is lower than that of seven of the nine competitors, and the three learning mechanisms add only about 2.18 s per run relative to the baseline GJO. Since the PI–PDN controller is tuned offline, prior to deployment, an execution time of about 86 s is entirely practical. Once tuned, the fixed-gain controller imposes no additional computational load during operation.

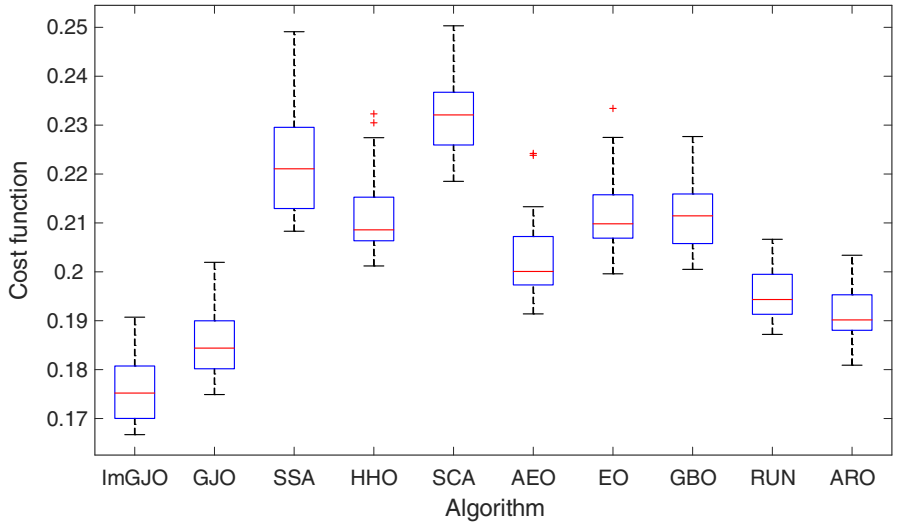

Figure 5: Boxplot analysis of ImGJO with respect to GJO, SSA, HHO, SCA, AEO, EO, GBO, RUN and ARO over 25 independent runs (red "+": outliers beyond 1.5 times the interquartile range).

5.2 Non-Parametric Statistical Test

To assess whether the differences observed in Table 2 are statistically significant, the Wilcoxon signed-rank test (MATLAB signrank function) was applied to the final objective values obtained in the 25 independent runs, with run k of ImGJO paired with run k of each competing algorithm ($k = 1, \ldots, 25$). The runs of each algorithm were initialized independently, so that the run index does not link the two samples and the pairing is, in effect, arbitrary. The test nevertheless remains valid in this setting. Under the null hypothesis that two algorithms produce identically distributed costs, the difference between two independent outcomes is distributed symmetrically about zero, which is the condition on which the signed-rank statistic relies; an arbitrary pairing may reduce the power of the test, but it does not inflate its type-I error. The null hypothesis H_0 states that there is no performance difference between ImGJO and the compared algorithm (i.e., the median of the paired differences is zero), whereas the alternative hypothesis H_1 states that a significant difference exists. The two-sided test was evaluated at a significance level of $\alpha = 0.05$ and, since $n = 25$, the p-values were obtained from the normal approximation of the signed-rank statistic. To address the multiple testing arising from the nine pairwise comparisons, a Bonferroni correction was additionally considered, which yields an adjusted significance threshold of $\alpha_{\text{adj}} = 0.05/9 \approx 0.0056$.

The results are summarized in Table 3. All p-values lie between 1.2290×10^{-5} and 5.7563×10^{-5}, i.e., more than two orders of magnitude below α_{adj}. The null hypothesis is therefore rejected in all nine cases, and the improvements achieved by ImGJO remain statistically significant even under this conservative correction. The identical p-value of 1.2290×10^{-5} obtained in seven comparisons is the smallest value that the two-sided test can return for 25 pairs; it occurs whenever ImGJO yields the lower objective value in all 25 paired runs, so that the smaller signed-rank sum equals zero ($z = -4.372$). For ARO and GJO, the competitor was better

in a few runs by small margins, giving $p = 1.3898 \times 10^{-5}$ and 5.7563×10^{-5}, respectively. For SSA, HHO, SCA, AEO, EO and GBO, moreover, the worst ImGJO cost (0.1907) is lower than the best cost of the competitor (Table 2); the two samples are completely separated, and every possible pairing of the runs gives the same result for these six comparisons. These results are consistent with the separation of the distributions in Fig. 5 and Table 2.

Table 3: Wilcoxon signed-rank test results (two-sided, $n = 25$; Bonferroni-adjusted $\alpha_{adj} \approx 0.0056$).

Comparison	*p*-Value	Decision
ImGJO vs. GJO	5.7563E−05	ImGJO better
ImGJO vs. SSA	1.2290E−05	ImGJO better
ImGJO vs. HHO	1.2290E−05	ImGJO better
ImGJO vs. SCA	1.2290E−05	ImGJO better
ImGJO vs. AEO	1.2290E−05	ImGJO better
ImGJO vs. EO	1.2290E−05	ImGJO better
ImGJO vs. GBO	1.2290E−05	ImGJO better
ImGJO vs. RUN	1.2290E−05	ImGJO better
ImGJO vs. ARO	1.3898E−05	ImGJO better

5.3 Ablation Test

To investigate the individual and combined contributions of the proposed enhancement strategies, an ablation analysis was carried out by progressively augmenting the baseline GJO algorithm with LCL, EGL, and ORL. Five algorithmic variants were therefore examined: standard GJO, GJO + LCL, GJO + EGL, GJO + ORL, and the full improved version (ImGJO). All variants were evaluated using identical parameter settings, search bounds, and termination criteria, and each experiment was repeated over 25 independent runs. The numerical results of this ablation study are summarized in Table 4. It can be observed that all single-strategy variants outperform the baseline GJO in terms of best, mean, and median objective values, confirming that each enhancement contributes positively to the search process. Among these variants, GJO + LCL achieves the largest reduction in mean cost, indicating that Laplacian crossover learning plays a key role in improving exploration capability and preventing premature convergence.

Table 4: Ablation analysis for performance evaluation with different improvement approaches.

Algorithm	Best	Mean	Median	Worst	Standard Deviation
ImGJO	0.1667	0.1757	0.1752	0.1907	0.0062
GJO	0.1749	0.1857	0.1844	0.2019	0.0070
GJO + LCL	0.1701	0.1788	0.1763	0.1968	0.0073
GJO + EGL	0.1723	0.1825	0.1803	0.2015	0.0081
GJO + ORL	0.1682	0.1796	0.1784	0.2014	0.0091

GJO + EGL and GJO + ORL also yield consistent performance improvements over GJO, though with slightly higher dispersion, reflecting their respective roles in exploitation intensification and boundary handling. In contrast, the full ImGJO consistently achieves the best performance across all statistical indicators, attaining the lowest best (0.1667), mean (0.1757), median (0.1752), and worst (0.1907) values, together with a reduced standard deviation (0.0062). These results demonstrate that the three enhancement

mechanisms are complementary rather than redundant. While each strategy individually improves the baseline algorithm, their coordinated integration within ImGJO produces a more balanced exploration–exploitation trade-off and leads to superior convergence reliability and solution quality. The ablation analysis therefore supports the design rationale of ImGJO for the present tuning problem.

The ablation results can also be related quantitatively to the intended role of each mechanism. LCL, which targets exploration, shifts the entire distribution downwards: relative to GJO, the mean, median, and worst values are reduced by 3.7%, 4.4%, and 2.5%, respectively, whereas the standard deviation remains almost unchanged (0.0073 vs. 0.0070). This uniform shift is consistent with a more diverse population that reaches better basins more often. EGL, which intensifies the search around the elite group, mainly improves the central part of the distribution (median −2.2%, mean −1.7%) but leaves the worst value practically unaffected (−0.2%). This is the expected signature of a greedy refinement operator, which cannot by itself rescue runs that remain in poor regions. ORL yields the best single-strategy best value (0.1682, −3.8%) but also the largest standard deviation (0.0091). This behavior is related to the structure of the present problem: the optimal filter coefficient lies close to its upper bound (N = 992.66 with $N \leq 1000$, Table 5), so the treatment of boundary violations directly affects how precisely this region is exploited. Opposition-based repair keeps such candidates inside the search space without collapsing them onto the bound, which benefits the best runs, whereas its stochastic re-sampling increases run-to-run variability when used alone. When the three operators are combined, the worst value (−5.5%) and the standard deviation (−11.4%) improve considerably more than with any single operator. This indicates that the exploratory effect of LCL and the refinement of EGL compensate for the variability introduced by ORL. These links are inferred from the distributions of the final objective values; population-diversity metrics were not recorded in the present study.

Table 5: Obtained best controller parameters via different algorithms after optimization process.

Algorithm	K_{P1}	K_I	K_{P2}	K_D	N
ImGJO	2.1645	4.9460	11.8453	0.0708	992.6618
GJO	1.0617	2.4319	22.9478	0.1471	872.8815
SSA	2.4739	5.7762	8.5754	0.0367	724.2938
HHO	4.3470	10.6235	5.6875	0.0363	930.5748
SCA	7.1661	17.4642	3.1566	0.0163	764.9355
AEO	2.8614	6.5667	7.7628	0.0498	828.9378
EO	3.1634	7.2170	6.7764	0.0469	512.2818
GBO	1.5694	3.7908	15.3381	0.0973	998.8295
RUN	1.0567	2.4469	21.9125	0.1305	949.8281
ARO	1.9157	4.4185	12.3918	0.0761	685.5582

5.4 Cost Function Minimization and Obtained Controller Parameters

The behavior of the algorithms during optimization is first examined through the convergence curves in Fig. 6, where the cost function in Eq. (23) is progressively reduced as iterations proceed. A clear separation among methods is observed: the proposed approach descends rapidly at early iterations and settles to a lower terminal value, whereas the alternatives plateau at higher costs. This pattern indicates that a more favorable region of the search space is located sooner and refined more effectively, yielding a consistently improved minimum. The best controller parameters recovered at the end of the optimization are summarized in Table 5.

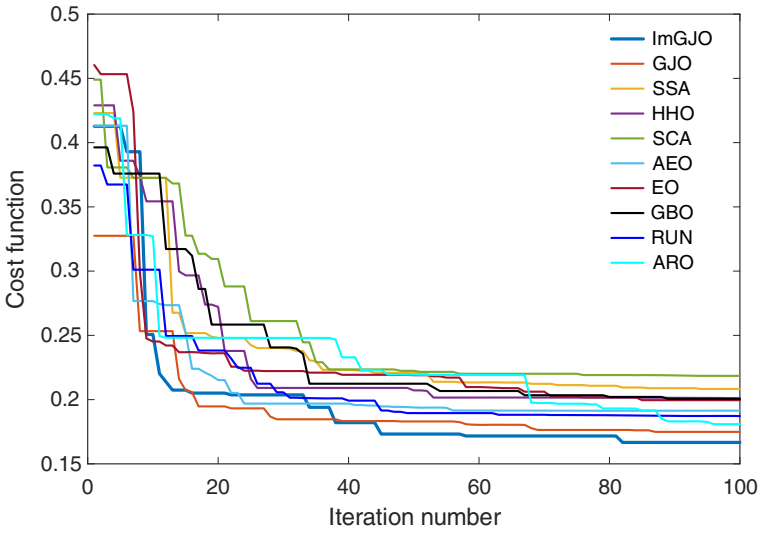

Figure 6: Comparative convergence curves demonstrating the cost function minimization.

All five decision variables remain within the admissible bounds set earlier, and distinct tuning signatures emerge across methods. The proposed strategy selects moderate PI-stage gains together with a mid-range PDN-stage proportional action and a small derivative time paired with a high derivative-filter factor. In contrast, the baseline method emphasizes a larger PDN-stage proportional gain and a longer derivative time with a somewhat smaller N, while several other competitors distribute effort differently across the PI and PDN stages (for example, favoring larger PI-stage gains with weaker PDN-stage action, or *vice versa*). These patterns suggest that improved closed-loop tracking is achieved here by combining decisive PDN-stage shaping with restrained integral buildup, a choice that aligns with the inertia-plus-delay character of the plant introduced earlier.

5.5 Transient Response

Transient behavior was assessed for a temperature step from 304 to 314 K. In Fig. 7, the proposed controller's trajectory rises promptly and settles without visible peaking, while the competing methods either respond more slowly or exhibit a pronounced crest; the zoomed view in Fig. 8 highlights the early-time segment where these differences emerge most clearly. The side-by-side display was therefore used to verify that the apparent visual advantage is not an artifact of axis scaling but reflects genuine improvements in rise and damping. These impressions are confirmed by the quantitative summary in Table 6.

The proposed ImGJO-based design achieves the fastest rise time (0.0117 min) and the shortest settling time (0.0607 min) among all contenders, while maintaining zero overshoot; it also attains the lowest integral of absolute error (IAE), given in Eq. (24), over the evaluation window (0.2008), indicating smaller cumulative tracking error through the transient. Taken together, these four metrics point to a response that is both swift and well-damped.

$$IAE = \int_0^{t_{sim}} |e(t)|dt \tag{24}$$

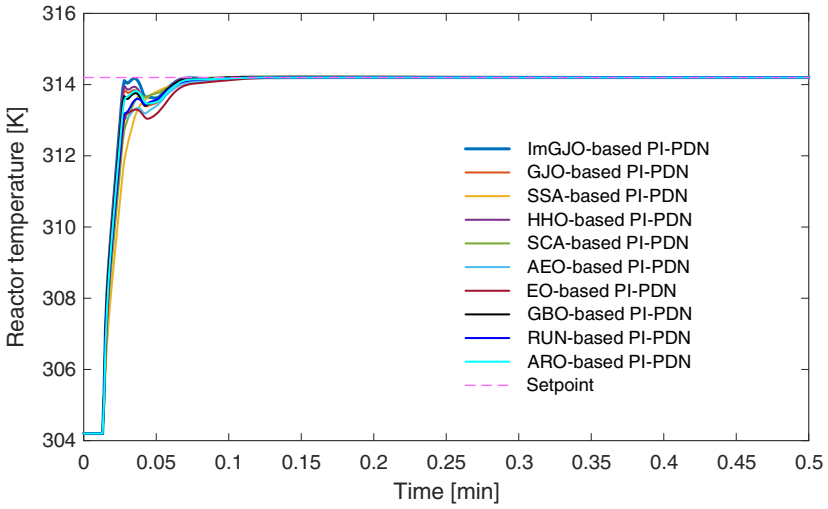

Figure 7: Comparative transient response analysis for reactor temperature changing from 304 to 314 K.

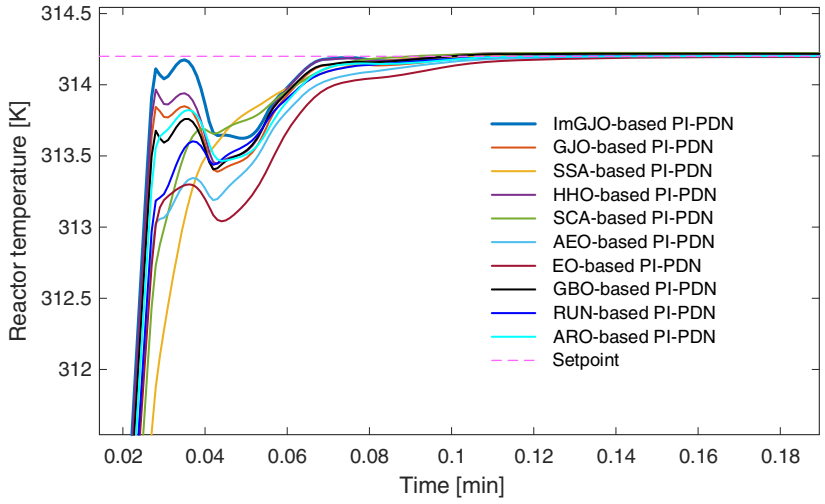

Figure 8: Zoomed view of Fig. 7.

Table 6: Transient response performance metrics and obtained minimum IAE values.

Algorithm	n_{rt} **(min)**	n_{st} **(min)**	n_{os} **(%)**	***IAE***
ImGJO	0.0117	0.0607	0	0.2008
GJO	0.0121	0.0623	0	0.2108
SSA	0.0219	0.0618	0.0689	0.2441
HHO	0.0119	0.0609	0.2272	0.2155
SCA	0.0177	0.0623	0.2389	0.2354
AEO	0.0189	0.0673	0	0.2308
EO	0.0162	0.0718	0	0.2405
GBO	0.0125	0.0618	0.1857	0.2197
RUN	0.0147	0.0638	0.0296	0.2223
ARO	0.0128	0.0638	0.0149	0.2163

Several contrasts are noteworthy. The baseline GJO also settles without overshoot but is slower on both rise and settling and accumulates a higher IAE (rise time (n_{rt}) = 0.0121 min, settling time (n_{st}) = 0.0623 min, IAE = 0.2108), suggesting that its additional delay translates into a modest but persistent error penalty. By comparison, HHO approaches the best settling time (0.0609 min) yet incurs a substantial overshoot (0.2272%) and a larger IAE (0.2155), indicating that a quicker approach alone does not guarantee economical error when peaking is present. Methods such as SCA and SSA show longer rise/settling and nonzero overshoot, which is mirrored by their higher IAE values. Algorithms whose overshoot is nominally zero (e.g., AEO and EO) nonetheless display longer settling and larger IAE than the proposed approach (AEO: n_{st} = 0.0673 min, IAE = 0.2308; EO: n_{st} = 0.0718 min, IAE = 0.2405), indicating that peaking avoidance alone does not suffice when convergence is comparatively slow. The remaining competitors (GBO, RUN, ARO) occupy intermediate positions: their rise and settling are slower than ImGJO's and their IAEs are higher, with occasional small overshoot in GBO and ARO.

To complement the numerical values in Table 6, the relative improvements achieved by the proposed method were also calculated with respect to the baseline algorithms. When compared with the standard GJO, the ImGJO reduced the rise time from 0.0121 to 0.0117 min (≈3%), the settling time from 0.0623 to 0.0607 min (≈3%), and the IAE from 0.2108 to 0.2008 (≈5%), while both controllers maintained zero overshoot. Larger margins were observed against the remaining competitors. For example, relative to SSA, the rise time was reduced by approximately 47%, the settling time by 2%, and the IAE by 18%, together with the elimination of overshoot (0.0689%). Compared with HHO, the proposed controller achieved similar settling performance but eliminated the 0.2272% overshoot and reduced the IAE by about 7%. Against SCA, the rise time decreased by roughly 34%, the settling time by 3%, and the IAE by 15%, while overshoot was fully removed. The remaining algorithms (AEO, EO, GBO, RUN, and ARO) exhibited slower rise/settling and higher cumulative error, with IAE reductions ranging from 7% to 16% when measured against the proposed design. These percentage comparisons confirm that the proposed ImGJO-based PI–PDN cascade delivers consistently superior transient behavior across all metrics.

Because Eq. (23) is a weighted sum, the effect of σ_1 and σ_2 on the comparison of the obtained controllers can be stated exactly. Table 6 shows that the ImGJO design has the shortest settling time, the lowest IAE and zero overshoot, so that no competing design is better on any of the three terms of the cost function. For any non-negative weights, its cost is therefore lower than that of every other design in Table 6 and, by the same argument, of every reported controller in Section 5.12; the only exception is the degenerate choice in which overshoot alone is penalized, for which all non-overshooting designs tie at zero. This property concerns the controllers obtained with $\sigma_1 = \sigma_2 = 0.1$. It does not imply that the algorithms would rank identically if the tuning were repeated with other weights, which was not examined in this study, and no claim of weight independence is made for the tuning process itself. The ISE, ITAE and ITSE results of Section 5.10 provide complementary evidence on alternative error measures.

5.6 Control Signal Behavior Analysis

In addition to output tracking, the control signal generated by each controller was examined, since excessive or oscillatory actuator demands may compromise practical feasibility. Fig. 9 shows the controller outputs of the ten tuned PI–PDN controllers for the 10 K setpoint step. The output represents the deviation of the coolant temperature from its nominal value (K), and no actuator limit is imposed in this section. Three phases can be distinguished. At $t = 0^+$, every controller produces a sharp initial peak whose magnitude, $u(0^+) = 10{\cdot}K_{P1}(K_{P2} + K_D N)$, is set by the high-frequency gain of the series PI–PDN structure multiplied by the step amplitude. Because all tuned controllers combine a sizeable derivative gain with a large filter coefficient, these peaks range from 870 K (SSA) to 1778 K (ImGJO), the peak of the proposed design being the

largest in the group. The peak is, however, extremely short-lived. It decays with the filter time constant and falls below 500 K within about 0.002 min. The output then remains at a plateau of about 250–280 K until the end of the dead-time interval, since the plant output has not yet responded. Thereafter, it decreases rapidly, shows a brief negative excursion around $t \approx 0.027$ min, and converges without sustained oscillation to the steady-state value of about 11.8 K required by the process gain. The competing controllers follow a similar pattern and differ mainly in the damping of the correction after the dead time. The reported controllers of Section 5.12 behave in the same way: the two 2-DOF PID controllers produce initial outputs of about 1026–1045 K, the Oustaloup realization of the CBOA-based FOPID about 994 K, and the DA- and WCA-based PID controllers, which contain an ideal derivative, a Dirac impulse at the setpoint change. The initial peak is thus a direct consequence of applying an ideal step reference to a controller with derivative action on the error. Such a coolant-temperature excursion cannot be realized by a physical jacket, and the comparisons of Sections 5.5 and 5.12 should therefore be regarded as comparisons of tuning quality under an idealized actuator. In industrial implementations, proportional and derivative kicks are routinely avoided by applying the derivative (and optionally the proportional) action to the measured output rather than to the error, by ramping or pre-filtering setpoint changes, and by including actuator limits with anti-windup protection [43]. The behavior of the proposed controller under coolant-temperature saturation, rate limits and anti-windup is examined in Section 5.8.

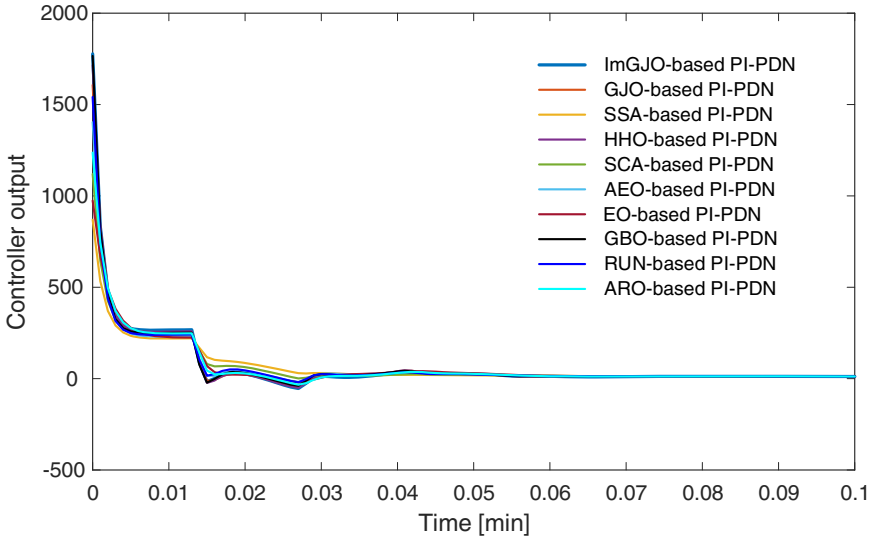

Figure 9: Comparative controller outputs (coolant-temperature deviation, K).

5.7 Trajectory Tracking Performance

In practical CSTR operation, the temperature setpoint is rarely constant for extended periods and may vary due to changes in production requirements, feed composition, or downstream specifications. Consequently, evaluating controller performance exclusively under single step changes may not fully reflect its capability to handle realistic operating scenarios. To address this limitation, the trajectory tracking performance of the proposed ImGJO-tuned PI–PDN cascade controller was investigated using a time-varying reference profile, as illustrated in Fig. 10. The applied reference consists of multiple successive setpoint variations with different magnitudes and directions, thereby emulating typical operational adjustments in reactor temperature. As observed in Fig. 10, the reactor temperature closely follows each setpoint transition with negligible steady-state error and without inducing oscillatory behavior. Each change in the reference is tracked smoothly, and the output converges rapidly to the new operating level, demonstrating that the

controller preserves its regulation quality beyond simple step-response conditions. Notably, no overshoot is observed during either upward or downward transitions, even when the setpoint changes occur in close succession. This behavior confirms that the proposed cascade structure, together with the ImGJO-based tuning, provides sufficient damping and anticipatory action to accommodate dynamic reference variations. The PDN stage shapes the actuator command during rapid transitions, while the PI stage maintains offset-free tracking, resulting in coherent closed-loop behavior across the entire trajectory. Compared with conventional step-response testing, the trajectory tracking analysis provides a more stringent assessment of controller robustness and adaptability. The ability of the proposed controller to maintain fast convergence, smooth transitions, and accurate tracking under varying setpoints indicates that the reported performance improvements are not limited to a single operating condition. Instead, the controller exhibits consistent behavior across a range of reference changes, which is essential for practical temperature regulation in CSTR systems.

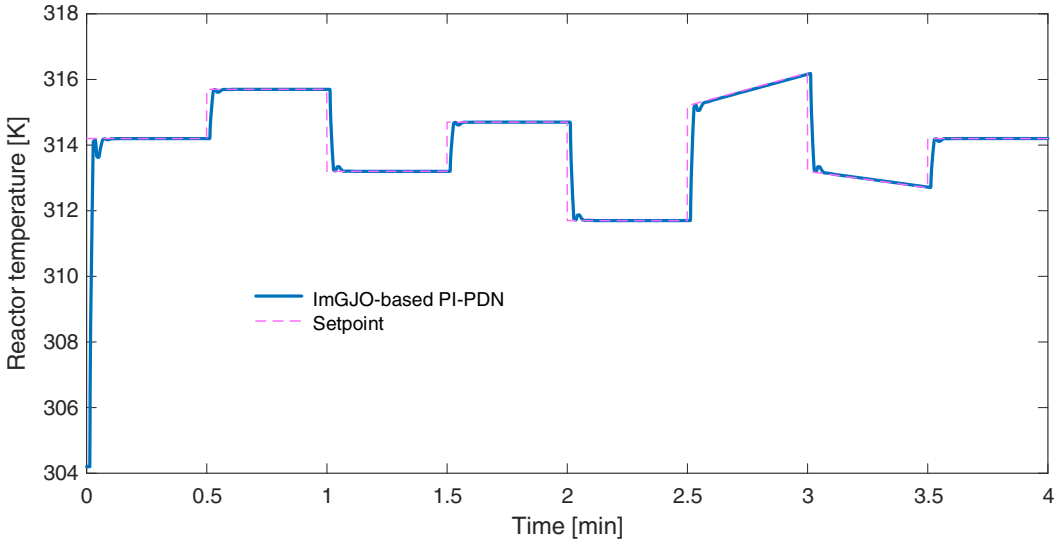

Figure 10: Performance of the proposed method for varying setpoint.

5.8 Performance for External Disturbance and Measurement Noise

To further examine the robustness of the proposed control strategy, the ImGJO-tuned PI–PDN controller was evaluated on the nonlinear CSTR model of Eqs. (16)–(18) under simultaneous external disturbance and measurement noise. The model was initialized at the steady state of Section 3 (C_A = 0.9774 mol/L, T = 304.17 K, T_j = 280 K), the controller output was applied as a deviation of T_j from 280 K, the dead time L of Eq. (19) was appended to the coolant input, and the equations were integrated with a fixed-step fourth-order Runge–Kutta scheme ($\Delta t = 10^{-5}$ min). The disturbance profile is shown in Fig. 11: a rectangular pulse of +0.8 K is added to the reactor-temperature output between 0.5 and 1.0 min, followed by a −0.7 K pulse between 1.0 and 1.5 min. These step-like variations emulate abrupt thermal disruptions caused by coolant-side fluctuations or feed changes. A zero-mean measurement noise, uniformly distributed within ±0.3 K and held constant over intervals of 0.001 min (generator seed 1), is superimposed on the fed-back temperature only, representing sensor-level uncertainty. No actuator limits are applied in this scenario. The resulting reactor temperature is presented in Fig. 12. The response to the setpoint change differs from that obtained on the FOPDT surrogate (Fig. 7): the temperature rises within about 0.011 min and reaches a peak of 314.76 K at $t \approx 0.028$ min, i.e., an overshoot of 5.9%, before settling within 0.060 min. The overshoot is

caused by the very large initial controller output discussed in Section 5.6 acting on a plant whose initial slope, $UA/(V\delta c_p)$ = 2.09 min^{-1}, is about 7% larger than that of the surrogate (K/τ = 1.95 min^{-1}). The linearized model with the same delay gives an almost identical overshoot (5.7%), which shows that the effect originates from this high-frequency gain mismatch rather than from the reaction nonlinearity. The loop nevertheless remains stable and well damped, in agreement with the gain margin of 6.88 dB (Section 5.11).

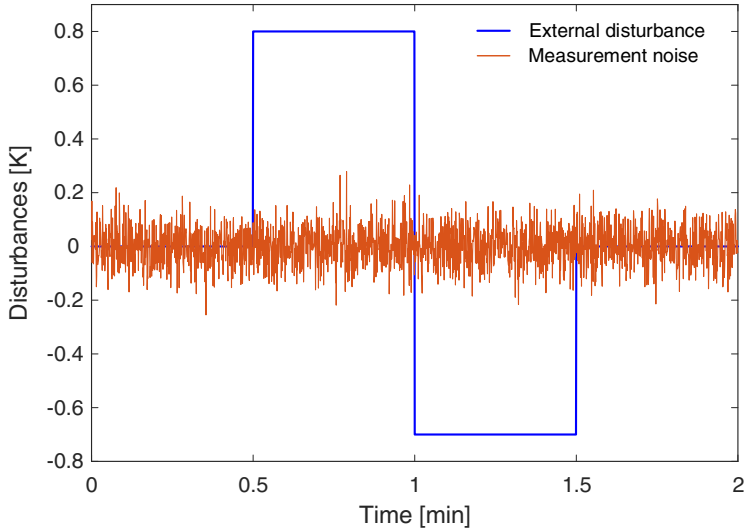

Figure 11: Output disturbance and measurement noise applied in the nonlinear simulation.

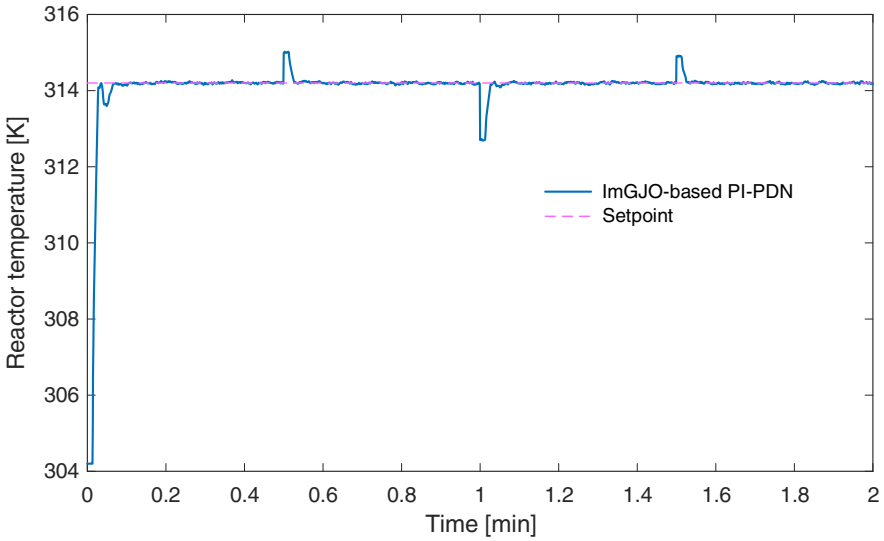

Figure 12: Reactor temperature on the nonlinear CSTR model (with dead time L) for a setpoint change from 304 to 314 K under the output disturbance and measurement noise of Fig. 11 (no actuator limits).

The output disturbances produce deviations of +0.79, −1.50 and +0.70 K at 0.5, 1.0 and 1.5 min, and each deviation is brought back within ±0.2 K of the setpoint in about 0.02 min without sustained oscillation. The measurement noise is strongly attenuated in the controlled temperature, whose standard deviation between 0.3 and 0.5 min is 0.05 K. It is, however, transmitted to the manipulated variable, whose standard deviation over the same interval is about 25 K, because the high-frequency gain of the tuned controller,

$K_{P1}(K_{P2} + K_D N) \approx 178$, amplifies the noise. A measurement filter or a smaller filter coefficient N would therefore be required in practice, at the expense of some transient performance. To examine actuator feasibility, the setpoint change on the nonlinear model was repeated without disturbance and noise for two actuator models. In the first (Case A), the coolant-temperature deviation is limited to $|u| \leq 30$ K, i.e., $T_j \in [250, 310]$ K, which provides about 2.5 times the steady-state coolant change of 11.76 K required for the step, and integrator windup is prevented by back-calculation with a tracking time constant equal to the integral time, $T_t = K_{P1}/K_I$ [43]. In the second (Case B), the rate of change of the coolant temperature is additionally limited to 300 K/min. In Case A, the derivative kick is clipped, the actuator saturates immediately after the setpoint change, and the response settles in 0.212 min with an overshoot of only 0.06%. In Case B, an overshoot of 7.3% appears, because the rate-limited coolant temperature cannot be lowered quickly enough once the reactor temperature approaches the setpoint. Within the limits of the simulation study, these results show that the controller obtained from surrogate-based tuning remains stable on the nonlinear model under abrupt disturbances, noisy feedback and actuator limits, whereas the zero overshoot obtained on the surrogate depends on the idealized actuator. It is noted that the nonlinear disturbance and noise analysis of this section is intended as a verification of the proposed controller rather than as a comparative benchmark. All competing controllers are optimized and compared on the same linear surrogate to ensure fairness and reproducibility.

5.9 Robustness Analysis

To evaluate the robustness of the proposed ImGJO-tuned PI–PDN cascade controller, its performance was examined under structured parametric variations of the CSTR model. Rather than considering a single disturbance realization, robustness was assessed by perturbing the key parameters of the first-order-plus-dead-time representation (namely the process gain K, time constant τ, and dead time L) around their nominal values. These variations emulate modeling uncertainty, operating-point drift, and thermal-transfer changes that commonly arise in practical reactor operation. The investigated cases are summarized in Table 7, which includes the nominal model and two perturbed scenarios. Scenario I corresponds to a moderate increase in process gain and time constant combined with a reduced dead time, while Scenario II reflects a reduced gain, increased dead time, and altered time constant. For each case, the rise time (n_{rt}), settling time (n_{st}), percent overshoot (n_{os}), and integral of absolute error (IAE) were computed. As reported in Table 7, the proposed controller maintains stable and well-damped behavior across all considered cases. In Scenario I, both rise time and settling time are slightly reduced relative to the nominal case, with only a marginal overshoot (0.0897%) and a small decrease in IAE, indicating that the controller adapts favorably to increased process responsiveness. In Scenario II, where the dead time is increased and the gain is reduced, a modest overshoot (2.9892%) is observed together with a moderate increase in settling time and IAE. Importantly, the closed-loop response remains stable and convergent, with no sustained oscillations or divergence.

Table 7: Robustness of the proposed method for different cases.

Case	n_{rt} (min)	n_{st} (min)	n_{os} (%)	*IAE*
Nominal (K = 0.85, τ = 0.4355, L = 0.0135)	0.0117	0.0607	0	0.2008
Scenario I (K = 0.90, τ = 0.45, L = 0.0125)	0.0112	0.0573	0.0897	0.1985
Scenario II (K = 0.80, τ = 0.42, L = 0.0145)	0.0121	0.0649	2.9892	0.2186

The corresponding temperature trajectories are illustrated in Fig. 13, which confirms that all perturbed cases track the reference smoothly and return to steady operation within a bounded time window. Although

performance degradation is evident as parameters deviate from their nominal values, the changes are gradual rather than abrupt, indicating low fragility of the tuned controller. Within the range of variations considered (about ±6–7% of the nominal values), the ImGJO-based PI–PDN controller therefore maintains stable and well-damped regulation in simulation. The absence of instability and the bounded variation of the performance indices indicate that the controller is not overly sensitive to moderate modeling errors. This robustness analysis complements the disturbance-and-noise verification presented earlier.

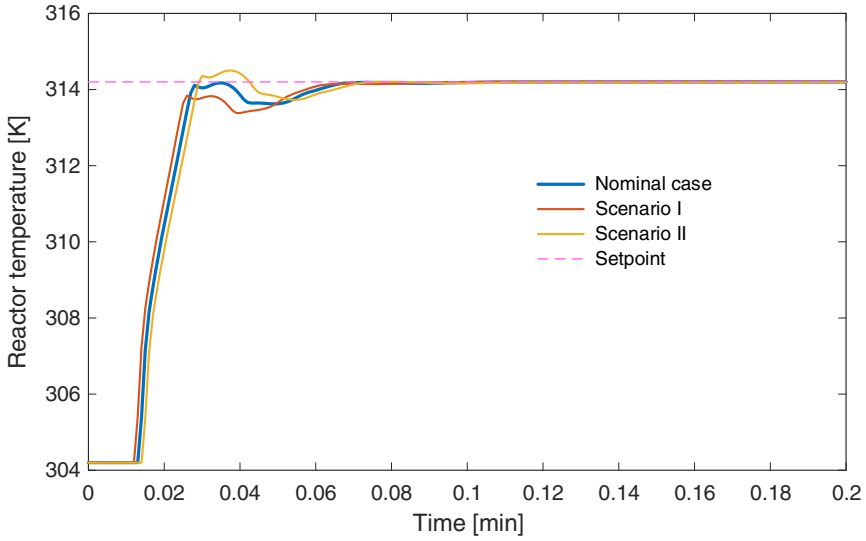

Figure 13: Reactor temperature change with respect to different cases listed in Table 7.

5.10 Performance Evaluation with Respect to Different Performance Indices

While the optimization framework in this study primarily employs the integral of absolute error (IAE) as part of the weighted composite cost function, it is important to verify that the observed performance improvements are not specific to a single metric. To this end, the proposed controller was additionally evaluated using a family of classical performance indices derived from the generalized performance indicator (PI) defined as:

$$PI = \int_0^{t_{sim}} |e(t)|^m t^n dt \tag{25}$$

where $e(t)$ denotes the tracking error, and the exponents m and n determine the relative emphasis on error magnitude and time weighting. By selecting appropriate values of m and n, commonly used indices are obtained: integral of squared error (ISE) for $m = 2$ and $n = 0$, integral of time-weighted absolute error ($ITAE$) for $m = 1$ and $n = 1$ and integral of time-weighted squared error ($ITSE$) for $m = 2$ and $n = 1$.

The comparative results for these indices are reported in Table 8. As observed, the proposed ImGJO-based PI–PDN controller achieves the lowest values across all three indices, indicating superior performance regardless of whether error magnitude, late-time behavior, or a combination of both is emphasized. Specifically, ImGJO yields the smallest ISE (1.6336), reflecting reduced sensitivity to large transient deviations, as well as the lowest ITAE (0.0029), which confirms rapid elimination of residual errors at later stages of the response. Likewise, the minimum ITSE value (0.0140) demonstrates that the proposed controller effectively suppresses both large and persistent deviations throughout the transient. In comparison, all benchmark algorithms exhibit higher values across at least one of the indices, indicating either slower convergence, larger transient excursions, or increased sensitivity to sustained error. For instance, SSA and SCA produce

noticeably higher ITAE and ITSE values, reflecting prolonged error persistence, while HHO and GBO, despite ISE values close to that of ImGJO, yield ITAE values more than twice as large, owing to their overshoot and slower final convergence. These results confirm that the performance advantages of the proposed ImGJO-tuned PI–PDN controller are consistent across multiple classical performance criteria and are not an artifact of selecting a particular metric. The use of IAE in the primary cost function is therefore justified as a balanced indicator for regulatory temperature control, while the additional indices demonstrate that similar conclusions are reached when alternative emphases are applied. Control-effort metrics were not incorporated into the objective, since the problem was formulated as a regulation task.

Table 8: Comparative performance evaluation for performance indices of ISE, ITAE and ITSE.

Algorithm	*ISE*	*ITAE*	*ITSE*
ImGJO	1.6336	0.0029	0.0140
GJO	1.6512	0.0032	0.0144
SSA	1.8607	0.0053	0.0188
HHO	1.6401	0.0078	0.0142
SCA	1.7585	0.0085	0.0165
AEO	1.7174	0.0037	0.0160
EO	1.7486	0.0048	0.0168
GBO	1.6554	0.0072	0.0146
RUN	1.6997	0.0040	0.0154
ARO	1.6874	0.0034	0.0151

5.11 Frequency Response Analysis of the Proposed Approach

In addition to time-domain investigations, frequency-domain analysis provides valuable insight into the stability margins, robustness, and disturbance-attenuation characteristics of the closed-loop system. To this end, the frequency response of the CSTR controlled by the proposed ImGJO-tuned PI–PDN cascade controller was examined using Bode analysis. The resulting magnitude and phase characteristics are shown in Fig. 14. As observed in Fig. 14, the closed-loop frequency response exhibits a well-shaped magnitude roll-off and a smoothly varying phase profile, indicating a stable and well-damped system. From a robustness perspective, the frequency-domain stability margins further substantiate the reliability of the proposed design. The gain margin is measured as 6.8825 dB, indicating that the open-loop system can tolerate a substantial increase in loop gain before instability occurs. In addition, the phase margin of 66.1194° provides a comfortable buffer against phase uncertainty, modeling errors, and unmodeled dynamics. Such a phase margin is generally regarded as indicative of strong robustness and well-damped transient behavior in industrial process control applications. The favorable frequency-domain characteristics can be attributed to the coordinated action of the cascade structure and the optimized controller parameters. The PI stage ensures adequate low-frequency gain for offset-free tracking, while the PDN stage introduces phase lead that improves the phase margin, with the derivative filter limiting, though not eliminating, the amplification of measurement noise. The ImGJO-based tuning effectively balances these contributions, yielding a frequency response that satisfies both performance and robustness requirements.

5.12 Comparative Assessment with Respect to Reported Approaches

A head-to-head comparison was conducted against approaches reported in the literature (quadratic interpolation optimizer (QIO)-based 2-degree-of-freedom (2-DOF) PID controller [19], differential evolution (DE)-based 2-DOF PID controller [19], coot bird optimization algorithm (CBOA)-based

fractional-order PID (FOPID) controller [20], dragonfly algorithm (DA)-based PID controller [20] and water cycle algorithm (WCA)-based PID controller [20]). Their published controller settings were adopted without modification, as summarized in Table 9, to ensure that the evaluation reflected the configurations advocated by the original studies. Structural differences across these designs (two-degree-of-freedom weighting, fractional orders, and ideal PID) were retained so that each method was represented in its intended form.

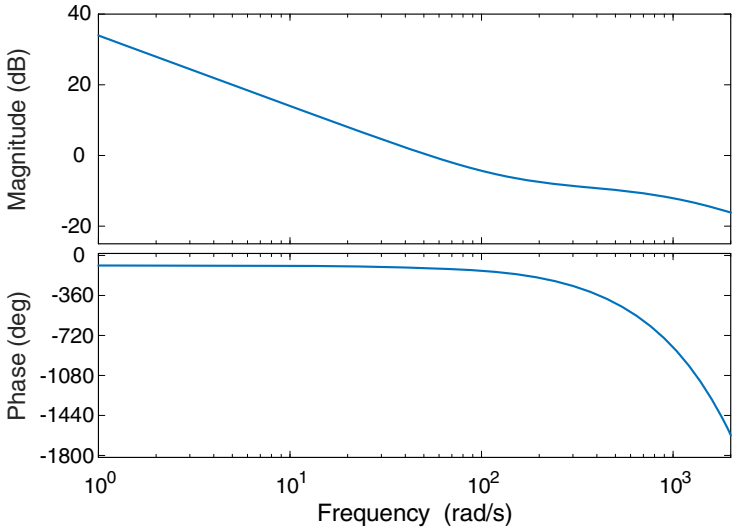

Figure 14: Bode diagram of ImGJO-based PI-PDN controlled system.

Table 9: Best controller parameters reported in relevant studies.

Controller Type	K_P	K_I	K_D	λ	μ	N	α	β
QIO-based 2-DOF PID	20.6040	57.9826	0.1137	—	—	402.5572	0.9989	1.8332
DE-based 2-DOF PID	16.1738	44.0146	0.1347	—	—	333.0291	1.0087	1.9234
CBOA-based FOPID	16.334	36.206	0.168	0.964	0.898	—	—	—
DA-based PID	13.299	40.130	0.268	—	—	—	—	—
WCA-based PID	12.146	39.840	0.157	—	—	—	—	—

The transient overlays in Fig. 15 (with a zoom in Fig. 16) show a consistent pattern: the PI–PDN tuned by ImGJO tracks the step change promptly and returns to the new operating level without peaking, whereas the comparative methods either converge more slowly or exhibit an overshoot crest before settling. The zoomed panel makes the early-time separation evident—ImGJO closes the gap fastest while maintaining a smooth approach, indicating a favorable trade-off between speed and damping. These visual impressions are supported by the quantitative results in Table 10. The ImGJO-based PI–PDN achieved the shortest settling time (0.0607 min) and fastest rise time (0.0117 min) while maintaining zero overshoot and the lowest IAE (0.2008). Against the best two-degree-of-freedom baseline (QIO), settling was reduced from 0.0669 to 0.0607 min (~9%), rise time from 0.0120 to 0.0117 min (~3%), and IAE from 0.2068 to 0.2008 (~3%), with both remaining non-overshooting. Relative to the DE-based 2-DOF PID, settling decreased from 0.0984 to 0.0607 min (~38%), overshoot was eliminated (0.1661 → 0), and IAE dropped from 0.2501

to 0.2008 (~20%). Larger gaps were observed against the single-loop comparators: for example, CBOA-FOPID (0.1076 min settling time, IAE = 0.3359), DA-PID (0.1483 min, 1.7114% overshoot, IAE = 0.4422), and WCA-PID (0.3601 min, 2.475% overshoot, IAE = 0.4928) all trailed the proposed design by wide margins.

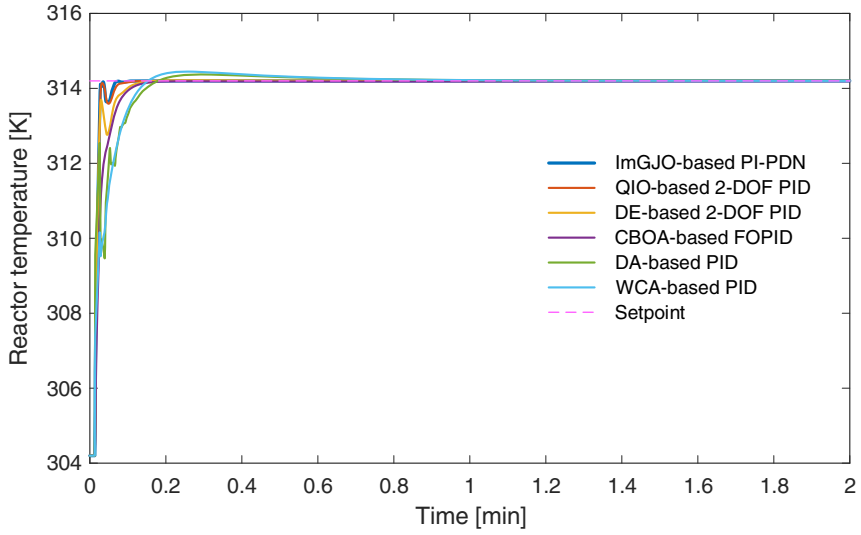

Figure 15: Comparative transient response analysis with respect to reported approaches (for reactor temperature changing from 304 to 314 K).

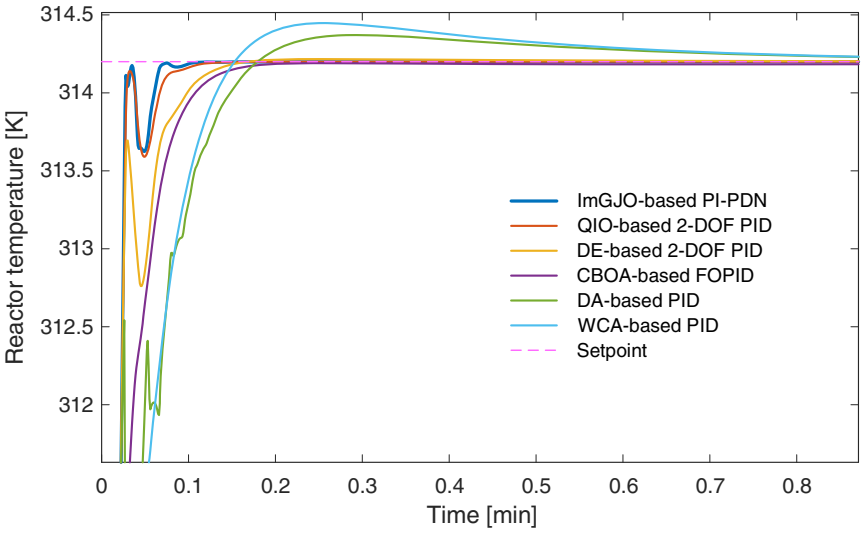

Figure 16: Zoomed view of Fig. 15.

Table 10: Transient response performance metrics and obtained minimum IAE values with respect to reported approaches.

Controller Type	n_{rt} (min)	n_{st} (min)	n_{os} (%)	*IAE*
ImGJO-based PI-PDN (proposed)	0.0117	0.0607	0	0.2008
QIO-based 2-DOF PID	0.0120	0.0669	0	0.2068
DE-based 2-DOF PID	0.0124	0.0984	0.1661	0.2501
CBOA-based FOPID	0.0476	0.1076	0	0.3359
DA-based PID	0.0833	0.1483	1.7114	0.4422
WCA-based PID	0.0775	0.3601	2.4750	0.4928

Percentage improvements were also quantified relative to the reported controllers summarized in Table 10. When compared with the best performing two-degree-of-freedom benchmark (QIO-based 2-DOF PID), the proposed controller reduced the settling time from 0.0669 to 0.0607 (≈9%), the rise time from 0.0120 to 0.0117 min (≈3%), and the IAE from 0.2068 to 0.2008 (≈3%), with both methods exhibiting zero overshoot. Much larger gains were observed against the DE-based 2-DOF PID, where settling time decreased by approximately 38% (0.0984 min → 0.0607 min), overshoot was eliminated (0.1661% → 0), and the IAE was reduced by about 20% (0.2501 → 0.2008). The fractional-order and single-loop comparators exhibited still wider gaps: relative to the CBOA-FOPID controller, the settling time was shortened by 44% and the IAE by 40%, while against the DA- and WCA-based PID designs, the proposed method reduced settling time by approximately 59% and 83%, respectively, and lowered the IAE by more than 54% in both cases. Under the idealized actuator assumed in this comparison, the ImGJO-tuned PI–PDN therefore outperforms the reported designs on all four transient response metrics; whether this advantage persists under actuator limits was not examined for the reported controllers and is left for future work.

6 Conclusion and Potential Future Works

This study presented an improved golden jackal optimizer (ImGJO) for tuning a series (cascaded) PI–PDN controller for CSTR temperature regulation. Three complementary operators (LCL, EGL and ORL) were incorporated to strengthen the population diversity, local refinement and boundary handling of the baseline GJO. The FOPDT surrogate used for tuning was first checked against the nonlinear mass and energy balances; for the upward 10 K transition considered, its gain and time constant agree with the identified values within 2%, and its step response deviates from the nonlinear one by an NRMSE of 1.8%. On this surrogate, the ImGJO-tuned controller attained the lowest best, mean, median and worst cost values over 25 runs, and Wilcoxon signed-rank tests with Bonferroni correction confirmed that the improvements over all nine competing metaheuristics are statistically significant. With an ideal actuator, zero overshoot was obtained together with the shortest rise and settling times. Relative to the metaheuristics, rise time, settling time and IAE were reduced by 1.7%–46.6%, 0.3%–15.5% and 4.7%–17.7%, respectively. Relative to previously reported CSTR controllers, the settling time was shortened by 9.3% with respect to the best 2-DOF PID design and by 43.6%–83.1% with respect to the fractional-order and single-loop PID designs, with IAE reductions of up to 59.3%. The ablation study showed that each operator improves the baseline and that their combination gives the largest reductions in worst-case cost and dispersion. Gain and phase margins of 6.88 dB and 66.1° were obtained, and stable behavior was maintained under parameter variations of about ±6–7%, time-varying references, and disturbances and measurement noise applied to the nonlinear CSTR model. On the nonlinear

model, the unconstrained loop overshot by 5.9% because of the large derivative kick, whereas coolant-temperature saturation with anti-windup reduced the overshoot to 0.06%; an additional rate limit led to an overshoot of about 7%.

All results were obtained in simulation, and several limitations should be noted. The controller was tuned on a local FOPDT surrogate whose accuracy decreases for downward moves and away from the nominal operating point. The comparisons with the other metaheuristics and with the reported controllers assume an ideal actuator; the large derivative and proportional kicks of the tuned controller make its response sensitive to actuator rate limits, and its high-frequency gain transmits measurement noise to the coolant command. The tuning was performed with fixed cost-function weights, and the effect of repeating it with other weights was not examined. No experimental validation has been performed, and no claim of experimental or industrial performance is made. Future work will therefore include the actuator model (saturation, rate limits and anti-windup) and a noise-sensitivity term in the cost evaluation, a comparison of all controllers under such constraints, derivative-on-measurement or setpoint-weighted variants of the PI–PDN structure [43], a weight-sensitivity study based on repeated tuning, tuning over wider operating regions and multiple steady states, a comparison with adaptive schemes such as the fuzzy-adaptive PID controller with feedforward compensation of [23] on the same process under a common protocol, and implementation on a laboratory- or pilot-scale reactor. The proposed tuning methodology is general and may be transferred to other processes characterized by inertia and time delay.

Acknowledgement: Not applicable.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Davut Izci: conceptualization, methodology, software, investigation, visualization, writing—original draft. Serdar Ekinci: conceptualization, methodology, software, formal analysis, writing—original draft. Rizk M. Rizk-Allah: validation, data curation, writing—review & editing. Mahmoud Abdel-Salam: validation, resources, writing—review & editing. Zahraa Tarek: project administration, supervision, writing—review & editing. Beytullah Yağız: formal analysis, validation, writing—review & editing. Mohammad Al-Rabayah: investigation, resources, writing—review & editing. Mohammad Salman: supervision, validation, writing—review & editing. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: All related data are available within the manuscript. The MATLAB code of ImGJO, the cost function and the simulation models are available from the corresponding author on reasonable request.

Ethics Approval: Not applicable. This study did not involve human participants, human data or animals.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A Parameters of the Nonlinear CSTR Model

Table A1 lists the parameters of the nonlinear CSTR model of Eqs. (16)–(18) and the nominal operating point used throughout the study. The parameter set is that of the benchmark jacketed CSTR of Bequette [41], which has also been used in [19,20,40].

Table A1: Parameters and nominal operating point of the nonlinear CSTR model.

Symbol	Description	Value	Unit
Q	Volumetric flow rate	100	L/min
V	Reactor volume	100	L
$C_{A,in}$	Inlet concentration of A	1	mol/L
T_{in}	Inlet temperature	350	K
k_0	Pre-exponential factor	7.2×10^{10}	min^{-1}
E_a/R	Activation energy divided by the gas constant	8750	K
$-\Delta H$	Heat of reaction (exothermic)	5×10^4	J/mol
δ	Density of the reacting mixture	1000	g/L
c_p	Specific heat capacity	0.239	J/(g·K)
UA	Overall heat-transfer coefficient × area	5×10^4	J/(min·K)
$T_{j,0}$	Nominal coolant temperature	280	K
$C_{A,0}$	Steady-state concentration (initial condition)	0.9774	mol/L
T_0	Steady-state reactor temperature (initial condition)	304.17	K
L	Transport/measurement delay appended to the model [19,20,40]	0.0135	min

References

1. Prabhakaran D, Thirumarimurugan M. A novel optimization of fractional order PID controller using chaotic maps based atomic search optimization for pH control in continuous stirred tank reactor. J Vib Eng Technol. 2022;10(8):3059–87. doi:10.1007/s42417-022-00538-4.
2. Calise F, Cappiello FL, Cimmino L, Dentice d'Accadia M, Vicidomini M. Dynamic analysis and investigation of the thermal transient effects in a CSTR reactor producing biogas. Energy. 2023;263(8):126010. doi:10.1016/j.energy.2022.126010.
3. Omeke WC, Mbamalu EE, Uwagbaoje OF. Design and mechanical analysis of a continuous stirred tank reactor (CSTR) for the optimum operation and production of propylene glycol from propylene oxide hydrolysis. Sustain Chem Eng. 2024:367–83. doi:10.37256/sce.5220244713.
4. Muhirwa JP, Mbalawata SI, Masanja VG. Markov chain Monte Carlo analysis of the variable-volume exothermic model for a continuously stirred tank reactor. Eng Technol Appl Sci Res. 2021;11(2):6919–29. doi:10.48084/etasr.3962.
5. Ekinci S, Izci D, Gider V, Bajaj M, Blazek V, Prokop L. Quadratic interpolation optimization-based 2DoF-PID controller design for highly nonlinear continuous stirred-tank heater process. Sci Rep. 2025;15(1):16324. doi:10.1038/s41598-025-01379-3.
6. Mahmood QA, Nawaf AT. Performance analysis of continuous stirred tank heater by using PID-cascade controller. Mater Today Proc. 2021;42(5):2545–52. doi:10.1016/j.matpr.2020.12.577.
7. Veerasamy G, Balaji S, Kadirvelu T, Ramasamy V. Reinforcement learning based adaptive PID controller for a continuous stirred tank heater process. Iran J Chem Chem Eng. 2025;44:265–82. doi:10.1109/chilecon.2017.8229632.
8. Bingi K, Ibrahim R, Karsiti MN, Hassan SM, Harindran VR. Real-time control of pressure plant using 2DOF fractional-order PID controller. Arab J Sci Eng. 2019;44(3):2091–102. doi:10.1007/s13369-018-3317-9.
9. Caparroz M, Soltesz K, Hägglund T, Guzmán JL, Berenguel M. A new approach to relay-based autotuning PID controllers and their evaluation in pH control of industrial photobioreactors. Control Eng Pract. 2025;164(5):106520. doi:10.1016/j.conengprac.2025.106520.
10. Doyle FJ, Packard AK, Morari M. Robust controller design for a nonlinear CSTR. Chem Eng Sci. 1989;44(9):1929–47. doi:10.1016/0009-2509(89)85133-4.

11. Gopaluni RB, Mizumoto I, Shah SL. A robust nonlinear adaptive backstepping controller for a CSTR. Ind Eng Chem Res. 2003;42(20):4628–44. doi:10.1021/ie020412b.
12. Alvarez J, Alvarez J, González E. Global nonlinear control of a continuous stirred tank reactor. Chem Eng Sci. 1989;44(5):1147–60. doi:10.1016/0009-2509(89)87014-9.
13. Pinheiro CIC, Kershenbaum LS. Model predictive control of reactor temperature in a CSTR pilot plant operating at an unstable steady-state. Comput Chem Eng. 1999;23(4):S859–62. doi:10.1016/S0098-1354(99)80211-7.
14. Panneer Selvam AMD, Hussin FA, Ibrahim R, Bingi K, Nagarajapandian M. Optimal fractional-order predictive PI controllers: for process control applications with additional filtering. Singapore: Springer Nature; 2022. doi:10.1007/978-981-19-6517-3.
15. Bingi K, Ibrahim R, Karsiti MN, Hassan SM. Fractional order set-point weighted PID controller for pH neutralization process using accelerated PSO algorithm. Arab J Sci Eng. 2018;43(6):2687–701. doi:10.1007/s13369-017-2740-7.
16. Devan PAM, Hussin FA, Ibrahim R, Bingi K, Abdulrab H. Design of fractional-order predictive PI controller for real-time pressure process plant. In: Proceedings of the 2021 Australian & New Zealand Control Conference (ANZCC); 2021 Nov 25–26; Gold Coast, Australia. p. 86–91. doi:10.1109/anzcc53563.2021.9628377.
17. Devan PAM, Hussin FA, Ibrahim RB, Bingi K, Nagarajapandian M, Assaad M. An arithmetic-trigonometric optimization algorithm with application for control of real-time pressure process plant. Sensors. 2022;22(2):617. doi:10.3390/s22020617.
18. Tamboli D, Chile R. Multi-model approach for 2-DOF control of nonlinear CSTR process. Int J Model Identif Control. 2018;30(2):143. doi:10.1504/ijmic.2018.094208.
19. Ekinci S, Izci D, Jabari M, Ma'arif A. Enhanced temperature control of continuous stirred tank reactors using QIO-based 2-DoF PID controller. J Robot Control. 2025;6(3):1340–6. doi:10.18196/jrc.v6i3.26586.
20. Begum KG. Coot bird optimization algorithm for the temperature control of continuous stirred tank reactor process. Asia-Pacific J Chem Eng. 2023;18(1):e2787. doi:10.1002/apj.2787.
21. Siddiqui MA, Anwar MN, Laskar SH. Control of nonlinear jacketed continuous stirred tank reactor using different control structures. J Process Control. 2021;108:112–24. doi:10.1016/j.jprocont.2021.11.005.
22. Mukherjee D, Raja GL, Kundu P, Ghosh A. Improved fractional augmented control strategies for continuously stirred tank reactors. Asian J Control. 2023;25(3):2165–82. doi:10.1002/asjc.2887.
23. Rospawan A, Angelina CL, Setiawan IMA, Saputra Z, Ocsirendi O, Febriansyah A, et al. Adaptive PID control using TSK elliptic fuzzy and static feedforward for observable disturbance rejection in industrial heating ovens. J Mechatron Electr Power Veh Technol. 2026;17(1):194–205. doi:10.55981/j.mev.2026.1094.
24. Izci D, Artun F, Ekinci S, Ghandour R, Salman M, Ghith E. Design of PI–PDN controlled DFIG system via a new objective function and starfish optimizer. In: Proceedings of the 2025 9th International Symposium on Innovative Approaches in Smart Technologies (ISAS); 2025 Jun 27–28; Gaziantep, Turkiye. p. 1–5. doi:10.1109/isas66241.2025.11101946.
25. Chopra N, Mohsin Ansari M. Golden jackal optimization: a novel nature-inspired optimizer for engineering applications. Expert Syst Appl. 2022;198(5):116924. doi:10.1016/j.eswa.2022.116924.
26. Deep K, Thakur M. A new crossover operator for real coded genetic algorithms. Appl Math Comput. 2007;188(1):895–911. doi:10.1016/j.amc.2006.10.047.
27. Yang Q, Zhang KX, Gao XD, Xu DD, Lu ZY, Jeon SW, et al. A dimension group-based comprehensive elite learning swarm optimizer for large-scale optimization. Mathematics. 2022;10(7):1072. doi:10.3390/math10071072.
28. Yang Q, Song GW, Gao XD, Lu ZY, Jeon SW, Zhang J. A random elite ensemble learning swarm optimizer for high-dimensional optimization. Complex Intell Syst. 2023;9(5):5467–500. doi:10.1007/s40747-023-00993-w.
29. Rahimi I, Gandomi AH, Chen F, Mezura-Montes E. A review on constraint handling techniques for population-based algorithms: from single-objective to multi-objective optimization. Arch Comput Meth Eng. 2023;30(3):2181–209. doi:10.1007/s11831-022-09859-9.
30. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM. Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv Eng Softw. 2017;114:163–91. doi:10.1016/j.advengsoft.2017.07.002.

31. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H. Harris Hawks optimization: algorithm and applications. Future Gener Comput Syst. 2019;97:849–72. doi:10.1016/j.future.2019.02.028.
32. Mirjalili S. SCA: a sine cosine algorithm for solving optimization problems. Knowl Based Syst. 2016;96(63):120–33. doi:10.1016/j.knosys.2015.12.022.
33. Zhao W, Wang L, Zhang Z. Artificial ecosystem-based optimization: a novel nature-inspired meta-heuristic algorithm. Neural Comput Appl. 2020;32(13):9383–425. doi:10.1007/s00521-019-04452-x.
34. Paliwal N, Srivastava L, Pandit M. Equilibrium optimizer tuned novel FOPID-DN controller for automatic voltage regulator system. Int Trans Electr Energy Syst. 2021;31(8):e12930. doi:10.1002/2050-7038.12930.
35. Daoud MS, Shehab M, Al-Mimi HM, Abualigah L, Abu Zitar R, Shambour MKY. Gradient-based optimizer (GBO): a review, theory, variants, and applications. Arch Computat Methods Eng. 2023;30(4):2431–49. doi:10.1007/s11831-022-09872-y.
36. Izci D, Ekinci S, Mirjalili S. Optimal PID plus second-order derivative controller design for AVR system using a modified Runge Kutta optimizer and Bode's ideal reference model. Int J Dyn Control. 2023;11(3):1247–64. doi:10.1007/s40435-022-01046-9.
37. Abualigah L, Izci D, Ekinci S, Abu Zitar R. Optimizing aircraft pitch control systems: a novel approach integrating artificial rabbits optimizer with PID-F controller. Int J Robot Control Syst. 2023;4(1):354–64. doi:10.31763/ijrcs.v4i1.1347.
38. Sinha A, Mishra RK. Temperature regulation in a continuous stirred tank reactor using event triggered sliding mode control. IFAC-PapersOnLine. 2018;51(1):401–6. doi:10.1016/j.ifacol.2018.05.060.
39. Kamalakkannan PE, Vinoth Kumar B, Kalamani M. Optimal nonlinear fractional-order proportional-integral-derivative controller design using a novel hybrid atom search optimization for nonlinear continuously stirred tank reactor. Therm Sci Eng Prog. 2024;54(5):102862. doi:10.1016/j.tsep.2024.102862.
40. Izci D, Jabari M, Ekinci S, Ghandour R, Salman M. Spider wasp optimizer-based PID control approach for temperature management in continuous stirred tank reactors. In: Proceedings of the 2025 9th International Symposium on Innovative Approaches in Smart Technologies (ISAS); 2025 Jun 27–28; Gaziantep, Turkiye. p. 1–5. doi:10.1109/ISAS66241.2025.11101837.
41. Bequette BW. Computer applications in process dynamics and control courses. Comput Appl Eng Educ. 1998;6(3):193–200. doi:10.1002/(sici)1099-0542(1998)6:33.3.co;2-s.
42. Izci D, Ekinci S, Rizk-Allah RM, Alribdi NI, Smerat A, Alzahrani A, et al. Master–slave architecture enhanced and improved GBO tuned cascaded PI-PDN controller for speed regulation of DC motors. Optim Control Appl Methods. 2025;46(5):2137–52. doi:10.1002/oca.3313.
43. Åström KJ, Hägglund T. Advanced PID control. 2006 [cited 2026 Jan 1]. Available from: https://onlinelibrary.wiley.com/doi/chapter-epub/10.1002/9781394442102.fmatter.