Computing, Performance and Communication Systems (2026)
Clausius Scientific Press, Canada

DOI: 10.23977/cpcs.2026.100111
ISSN 2371-8889 Vol. 10 Num. 1

Mechanism-Constrained Self-Evolving Optimisation and Safe Trustworthy Closed-Loop Control for Wastewater Treatment Processes

Jiaxiangyi Liu[1,a], Xinzhe Gao[2,b,*]

[1]School of Electronics and Information Engineering, Liaoning University of Science and Technology, Liaoning, China
[2]Artificial Intelligence Division, Beijing Vision Zhiyuan Technology Co., Ltd., Beijing, China
[a]120243103067@stu.ustl.edu.cn, [b]18242691358@163.com
**Corresponding author*

Keywords: Wastewater treatment; Evolutionary computation; Model predictive control; Control barrier function; Runtime assurance

Abstract: Intelligent optimisation algorithms face three obstacles when entering the control loops of wastewater treatment plants (WWTPs): the lack of a trustworthy evaluation environment, the absence of formally provable safety guarantees, and the year-by-year depreciation of once-delivered algorithms. This paper proposes a mechanism-constrained self-evolving optimisation system for wastewater treatment processes, organised as a one-core, two-engine, double closed-loop architecture. The offline evolution engine evolves optimisation strategies through a sample-generate-evaluate-select-archive loop, using a mechanistic model as the trustworthy evaluation field; the online execution engine forms a three-level closed loop with the mechanistic model, a physics-informed neural network (PINN) and model predictive control (MPC). An evolved algorithm may only be issued after passing five one-vote-veto safety gates, namely historical replay, PINN fast screening, high-fidelity mechanistic simulation, shadow-mode operation and graduated grey-release, and is further protected by three layers of formal safety mechanisms: a control barrier function filter, a runtime assurance architecture and uncertainty quantification. Upon any boundary violation the system fuses and rolls back to the gold-medal strategy within ten seconds. Verification shows that the three-level evaluation filtering reduces the evaluation cost by three orders of magnitude, that quantum-inspired search accelerates convergence by about one order of magnitude, and that the system reduces aeration energy consumption by 5%-15% and chemical consumption by 10%-30%, providing a feasible path for artificial intelligence to safely enter wastewater treatment control loops.

1. Introduction

Wastewater treatment processes feature large inertia, strong coupling and frequent disturbances: the influent load fluctuates on a minute scale, whereas manual response operates on an hourly scale, and conservative strategies such as constant excessive aeration and worst-case dosing cause

significant waste. Evolutionary computation has proven effective on combinatorial optimisation problems [1], and variants such as the quantum-inspired evolutionary algorithm further improve convergence speed [2]; however, their application to the control of continuous industrial processes is hindered by two factors: the lack of a trustworthy and affordable evaluation environment, and the lack of formally provable safety guarantees. Model predictive control (MPC) is the mainstream method for constrained multivariable control in the process industry [3], and control barrier functions (CBFs) provide a mathematical tool for the online enforcement of safety constraints [4]. This paper embeds both into a self-evolving framework. The main contributions are: (1) a double closed-loop architecture that separates fast online execution from slow offline evolution; (2) five safety gates combined with a three-layer formal safety mechanism composed of CBF filtering, runtime assurance and uncertainty quantification; (3) a data-reflux-driven generational evolution loop in which the benefit of the algorithm rises year by year instead of decaying.

2. Architecture of the Self-Evolving Optimisation System

2.1 Overall Architecture

The system consists of four layers. The offline evolution layer iterates over generations of strategies, where massive failures are permitted because their cost is merely computing power. The safety verification layer comprises five gates with one-vote veto. The online execution layer runs a three-level closed loop of the mechanistic model, a PINN surrogate and MPC, augmented with fusing and auditing, and permits no failure. The physical layer consists of the plant SCADA, PLCs, instruments and actuators. Operating data flow back from the physical layer to the evolution layer, closing the second loop. The system runs at three tempi: sensing every 5 minutes, execution at the second level, and evolution on a weekly scale.

2.2 Offline Evolution Engine

The evolution engine produces optimisation strategies through population iteration. The fitness function is defined in a multi-objective weighted form

$$f = w_1 \cdot R_{eff} + w_2 \cdot (-E_{sp}) + w_3 \cdot (-C_{chem}) - \lambda \cdot P_{vio} \tag{1}$$

where R_eff is the effluent compliance rate, E_sp is the energy consumption per ton of water, C_chem is the chemical consumption, and P_vio is the penalty term for safety-constraint violations. The selection operator combines Pareto-front ranking with Boltzmann selection to preserve population diversity; the probability of individual i being selected is

$$P(i) = \frac{exp(\frac{f_i}{T})}{\sum_{j=1}^{N} exp(\frac{f_j}{T})} \tag{2}$$

where the temperature parameter T decays over generations, encouraging exploration early and exploitation late. The generation operator performs code-level mutation driven by a large language model (parameter perturbation, logic recombination, operator replacement and structural innovation); the evaluation operator applies three-level filtering of historical replay, PINN fast screening and mechanistic re-verification. Evolution proceeds at three rates: parameter evolution (minute-to-hour scale, e.g. DO setpoints and PID parameters), strategy evolution (day-to-week scale, e.g. control logic and dosing strategies), and model evolution (month-to-quarter scale, e.g. model structures and kinetic parameters). Each generation contains 32-128 candidates, with a routine cycle of 1-3 days.

On the online side, the execution engine is centred on receding-horizon MPC, which solves the following constrained optimisation problem at every control period:

$$min_u \sum_{k=0}^{N}(\|y_k - r_k\|_Q^2 + \|\Delta u_k\|_R^2),\ \ s.t. x_{k+1} = F(x_k, u_k), u_{min} \le u_k \le u_{max} \quad (3)$$

where the prediction model F is provided by the mechanistic model or its Koopman global linearisation, Q and R are weighting matrices, and Y_safe denotes the safe output set. For model-free online fine-tuning, extremum seeking control (ESC) is introduced: a small sinusoidal probing signal is superimposed on the setpoint

$$u = \hat{u} + a \cdot sin(\omega t) \quad (4)$$

with an amplitude of ±1%-2% of the setpoint, which is imperceptible to the process; after high-pass filtering and demodulation, the gradient of the performance index with respect to the setpoint is estimated in real time and the setpoint is updated along the gradient direction [5]. Multi-chemical cooperative dosing is modelled as a non-cooperative game among carbon source, phosphorus removal and disinfection agents, whose dosing strategies converge to a Nash-equilibrium point at which no agent can reduce its own cost function by unilateral deviation [6].

2.3 Five Safety Gates and Three-Layer Formal Safety Mechanisms

Evolution encourages exploration, while execution must be conservative. Any algorithm must pass five one-vote-veto safety gates before entering the production closed loop, whose functions and release criteria are listed in Table 1. The gates enforce four no-issue rules: no issue beyond process boundaries, no issue without passing mechanistic simulation, no issue with mismatched authority, and no issue with unexplainable anomalies.

Table 1 Five safety gates

No.	Gate	Function	Release criterion
01	Historical replay	Replay of rainy-season shocks and abnormal conditions	Outperform the best existing baseline
02	PINN fast screening	Second-level evaluation of thousands of candidates	Physically infeasible solutions eliminated
03	High-fidelity mechanistic gate	High-fidelity full-process simulation	No issue without passing simulation
04	Shadow mode	Recommendation only, no execution	Stable wins over consecutive periods
05	Graduated grey-release	Assisted, semi-automatic, automatic promotion	Fusing on violation, rollback < 10 s

Above the procedural defence line, three mathematical defence layers are superimposed. The first layer is a CBF-QP filter: with the safe set defined as C = {x | h(x) ≥ 0}, the nominal command u_nom produced by the evolved algorithm is filtered online by a quadratic programme

$$u^* = argmin_u \|u - u_{nom}\|^2,\ \ s.t. L_f h(x) + L_g h(x) \cdot u + \alpha(h(x)) \ge 0 \quad (5)$$

which returns the minimally modified command that keeps the state inside the safe set [4]. The second layer is a runtime assurance (RTA) architecture originating from fly-by-wire avionics: the primary control channel (the evolved algorithm) and the safety channel (a degraded baseline controller) run in parallel, and the switching logic is formally verified so that takeover capability is always present. The third layer is uncertainty quantification (UQ): Bayesian calibration decomposes the predictive variance into an epistemic component and an aleatoric component; the former triggers degradation when exceeding its threshold, while the latter is converted into a safety margin, so that the model knows what it does not know. A four-level degradation chain (evolved optimal strategy, elite-library stable strategy, conventional automatic control, manual operation) guarantees that a retreat always exists, and the edge node can operate autonomously for no less than 72 hours after disconnection. Before issuance, every command is rehearsed on a digital shadow that fast-forwards

the process two hours into the future; if the exceedance probability crosses the threshold, the command is degraded automatically, and the hashes of strategy evolution, gate approvals and command issuance are chained for tamper-proof auditing.

2.4 Self-Healing of Data and Models

The engineering foundation of self-evolution is the sustained trustworthiness of data and models. The sensor self-healing network cross-validates every instrument using the redundancy among the mechanistic equations: for the i-th instrument, its expected reading ŝ_i is inverted from the remaining observables through the model, and an out-of-threshold residual e_i = s_i − ŝ_i isolates the faulty sensor and issues a work order. The drift watchdog applies CUSUM change-point detection to the prediction residuals

$$S_t = max(0, S_{t-1} + |e_t| - k), S_t > h \Rightarrow recalibration \tag{6}$$

where k is the tolerated drift and h is the alarm threshold [7]. Once drift is captured, the adjoint method completes parameter recalibration within hours, unattended; dirty data cannot enter and stale models cannot linger, so the evolution loop remains trustworthy over the long term.[8]

3. Experiments

3.1 Datasets and Evaluation Criteria

The experiments are based on historical operating data from 27 WWTPs covering seven regions of China: a sampling interval of 5 minutes, 20 monitored fields, approximately 31 million records and a time span of about 10.9 years. The evaluation criteria include the effluent compliance rate, energy consumption per ton of water, chemical consumption, per-candidate evaluation cost, evolution convergence speed and safety-event statistics. The experimental environment is listed in Table 2, and the weights of the fitness function in Eq. (1) are set plant by plant according to compliance priorities.

Table 2 Experimental environment

Item	Configuration
Deployment architecture	Cloud-fog-edge collaboration (cloud evolution, edge execution)
Edge node	Domestic AI edge server, NPU ≥ 20 TOPS
Offline autonomy	≥ 72 hours
Access mode	Non-intrusive access, 4 h on-site, 48 h observation window
Rollout procedure	Five-stage acceptance, 90 days from modelling to semi-automatic closed loop

3.2 Evolution Efficiency Experiments

Taking the joint optimisation of aeration and dosing as the task, different evaluation and search configurations are compared in Table 3.

Table 3 Comparison of evolution efficiency

Configuration	Per-candidate evaluation cost	Convergence generations	Remark
Pure high-fidelity mechanistic evaluation	Baseline (1×)	Baseline	Highest accuracy, unaffordable cost
Three-level filtering (replay, PINN, re-verification)	approx. 10^{-3} × baseline	Baseline	Evaluation cost reduced by 3 orders of magnitude
Three-level filtering + quantum-inspired search	approx. 10^{-3} × baseline	approx. 1 order of magnitude fewer	Convergence accelerated by about one order of magnitude

3.3 Optimisation Benefits and Safety Verification

The system was rolled out plant by plant under the five-stage acceptance procedure; the key milestones are listed in Table 4 and the optimisation benefits are summarised in Table 5.

Table 4 Milestones of the five-stage acceptance procedure

Stage	Period	Milestone criterion	Result
1 Survey and modelling	Weeks 1-3	Simulation deviation < 10%	Achieved
2 Shadow operation	Weeks 4-8	Agreement with human operation ≥ 70%	Achieved
3 Assisted recommendation	Weeks 9-12	Strategy adoption rate ≥ 60%	Achieved
4 Semi-automatic trial	From week 13	30 consecutive days without fusing	Achieved
5 Acceptance	Month 6	Aeration energy consumption ↓ ≥ 8%	Achieved

Table 5 Summary of optimisation benefits

Indicator	Benefit	Verification basis
Aeration energy consumption	↓5%-15%	Pilot calibre (acceptance line ≥ 8%)
Chemical consumption	↓10%-30%	Pilot calibre
Chemical phosphorus removal dosing	↓approx. 18%	Dynamic control vs fixed dosing, same-period control; effluent orthophosphate ≤ 0.2 mg/L
Blower energy consumption	↓18%-25%	Stable terminal DO control; ammonia nitrogen compliance rate ≥ 99.5%
Safety events	0 boundary-violating issuances	Algorithms failing the gates never reach the PLC; rollback < 10 s on violation

Unlike traditional buy-out algorithms whose effect decays year by year, the evolution mode keeps operating data flowing back to drive re-evolution, so the benefit index rises year by year and the algorithm becomes an appreciating asset. The evolution paradigm has also been verified in four industrial scenarios—quay-crane scheduling at ports (solution time reduced from hours to minutes), automotive drag simulation, injection-moulding scheduling (improved by 6.31%) and steel-plate cutting utilisation (98.2%)—demonstrating the transferability of the method.

4. Conclusion

This paper has proposed a mechanism-constrained self-evolving optimisation system for wastewater treatment processes. The fast-slow separated architecture of offline evolution and online execution, together with five safety gates and the CBF-RTA-UQ three-layer formal safety mechanism, unifies bold exploration in evolution with absolute conservatism in execution, and the data-reflux-driven generational evolution turns the algorithm into an appreciating asset. Experiments show that, with the evaluation cost reduced by three orders of magnitude, the system reduces aeration energy consumption by 5%-15% and chemical consumption by 10%-30%, with zero boundary-violating issuances throughout. Future work will proceed along multi-plant island evolution with gene migration and cross-plant federated learning.

References

[1] Back T, Fogel D B, Michalewicz Z. Handbook of Evolutionary Computation. Oxford University Press, 1997.
[2] Han K H, Kim J H. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. IEEE Transactions on Evolutionary Computation, 2002, 6(6): 580-593.
[3] Camacho E F, Bordons C. Model Predictive Control. Springer, London, 2007.
[4] Ames A D, Grizzle J W, Tabuada P. Control barrier function based quadratic programs with application to adaptive cruise control. In Proceedings of the 53rd IEEE Conference on Decision and Control, 2014: 6271-6278.
[5] Krstic M, Wang H H. Stability of extremum seeking feedback for general nonlinear dynamic systems. Automatica, 2000, 36(4): 595-601.

[6] Nash J F. Equilibrium points in n-person games. Proceedings of the National Academy of Sciences, 1950, 36(1): 48-49.
[7] Page E S. Continuous inspection schemes. Biometrika, 1954, 41(1/2): 100-115.
[8] Raissi M, Perdikaris P, Karniadakis G E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 2019, 378: 686-707.