

Adaptive reinforcement learning for energy-efficient high-recovery closed-circuit reverse osmosis

Jeongwoo Moon^{a,1}, Byeongchan Yun^{b,1} , Kiho Park^c , Seong-Su Kim^d, Youngjoo Lee^d, Kwanho Jeong^{e,*} , Kyung Hwa Cho^{b,*}

^a Future and Fusion Lab of Architectural, Civil, and Environmental Engineering, Korea University, Seoul 02841, Republic of Korea

^b School of Civil, Environmental, and Architectural Engineering, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul 02841, Republic of Korea

^c Department of Chemical Engineering, Hanyang University, 222 Wangsimni-ro, Seongdong-gu, Seoul 04763, Republic of Korea

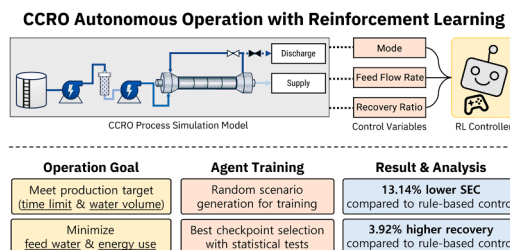
^d Water and Wastewater Research Center, Korea Water Resources Corporation, 125 Yuseong-dearo, 1689beon-gil, Yuseong-gu, Daejeon 34045, Republic of Korea

^e Department of Environmental Engineering, Chosun University, Gwangju 61452, Republic of Korea

HIGHLIGHTS

- Dynamic CCRO environment was implemented and calibrated using operational data
- RL was applied to CCRO environment and the best-performing policy was identified
- Nonparametric tests validated RL policy selection across scenarios
- RL agent improves SEC by 13.14% and recovery by 3.92% relative to rule-based control
- Explainable AI analyses support interpretable RL control

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:

Closed circuit reverse osmosis
Dynamic modeling
Reinforcement learning
Proximal policy optimization
Process control

ABSTRACT

Closed-circuit reverse osmosis (CCRO) achieves high recovery; however, its semi-batch purge-and-refill cycles complicate control and optimization. Unlike static rule-based operation, reinforcement learning is increasingly used for operational optimization to adaptively select real-time control setpoints under changing conditions. In this study, a data-calibrated dynamic CCRO simulator was integrated with a reinforcement learning control framework and evaluated under realistic plant-variable conditions. Plant behavior was stably reproduced by the simulator, achieving low RMSE across permeate flow, circulated concentrate flow, membrane inlet pressure, and permeate concentration. A proximal policy optimization agent was trained across 24 environmental settings with over 10 million steps, and the best-performing policy was identified through evaluation of 375 predefined scenarios and nonparametric statistical analyses. Across the evaluation scenarios, the resulting agent achieved a mean specific energy consumption (SEC) of 0.489 kWh/m³ and a mean recovery rate of 95.5%, outperforming a static rule-based controller by 13.14% in SEC and 3.92% in water recovery through adaptive control. Interpretability and feasibility were further supported by explainable AI and edge execution-time analyses on

Abbreviations: RO, Reverse osmosis; SEC, Specific energy consumption; CCRO, Closed-circuit reverse osmosis; RL, Reinforcement learning; PPO, Proximal policy optimization; SHAP, Shapley additive explanation; CC, Closed-circuit; PID, Proportional-integral-derivative; R2, Coefficient of determination; RMSE, Root mean square error; LLM, Large language model.

* Corresponding authors.

E-mail addresses: khjeong@chosun.ac.kr (K. Jeong), khcho80@korea.ac.kr (K.H. Cho).

¹ These authors contributed equally to this work.

<https://doi.org/10.1016/j.watres.2026.125855>

Received 17 December 2025; Received in revised form 12 March 2026; Accepted 31 March 2026

Available online 1 April 2026

0043-1354/© 2026 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

representative hardware. Overall, the proposed framework provides a promising alternative to conventional rule-based CCRO operation in small-scale decentralized plants where continuous expert supervision is impractical.

1. Introduction

Reverse osmosis (RO) is the dominant membrane process for desalination and potable water reuse, accounting for approximately 85% of the plants and 70% of the installed capacity worldwide (Moon et al., 2025b). RO operation requires a feed pressure that exceeds the osmotic pressure difference between the feed and permeate, as well as hydraulic losses to produce water (Elimelech and Phillip, 2011). Under higher salinity and recovery conditions during RO, the osmotic pressure difference across the membrane inevitably increases, reducing the net driving pressure available for water permeation at a constant feed pressure (Moon et al., 2024). This necessitates either increased feed pressure or operational adjustments to sustain permeate production, thereby increasing pump work and specific energy consumption (SEC) (Kim et al., 2019). Moreover, inland facilities that pursue brine minimization and move toward zero liquid discharge now target recoveries of 90% or higher, which increases the risk of mineral scaling and the cost of brine management (Walker et al., 2014). Conventional RO increases overall recovery by adding serial stages, thereby reducing the brine volume. However, this also increases capital and operating costs due to added system complexity and higher operational demands (Elbassoussi et al., 2025).

Hence, closed-circuit reverse osmosis (CCRO) has been developed to address the inherent limitations of multistage operations (Warsinger et al., 2016). In the CCRO process, the concentrate is continuously recirculated and mixed with the newly pressurized feed (Warsinger et al., 2018). During operation, the applied pressure was gradually increased to compensate for the rising osmotic pressure. Upon reaching a specified recovery or permeate quality setpoint, the system purged the brine and refilled it with fresh feed to initiate a new semi-batch cycle. Reflecting its operational robustness, CCRO has demonstrated high recoveries in brackish water management and potable water reuse pilots, with sustained recoveries of 93–95% under careful operation (Li, 2023a). However, continuous mixing generates entropy, which increases the work required for separation (Mo and She, 2025). At higher salinities, the operation is limited by the maximum feed pressure that the pump can deliver (Sutariya and Raval, 2021). Moreover, CCRO has two separate issues. First, fouling and mineral scaling persist rather than being fully eliminated across the cycles. Second, purge-and-refill transients complicate data normalization and operational control (Gu et al., 2021).

Building on these operational challenges, the complexity and limitations of the CCRO necessitate the use of dynamic simulations to optimize the system design and operation for reliable long-term performance (Gu et al., 2021). Dynamic models incorporate key process elements, including semi-batch mixing, purge events, pressure drops, concentration polarization, and membrane fouling. These models enable quantitative evaluation of recovery, permeate quality, and energy consumption (Li, 2023a). Owing to the capabilities of these dynamic models, optimized operating policies and advanced control strategies can be developed to enhance the CCRO performance (Li, 2021). Existing CCRO studies can be broadly categorized into two groups: analysis-based studies that focus on understanding process behavior and performance, and operation-based studies that focus on guiding operating conditions and practical implementation.

Analysis-based studies have mainly used mechanistic dynamic models to describe the cyclic behavior of CCRO. One study developed a spatiotemporal model coupling water, salt, and momentum balances with pressure drop and concentration polarization to analyze the cyclic steady state. It reported that under brackish conditions, CCRO is less energy efficient than multistage RO at matched fluxes and recoveries,

with competitiveness improving at lower fluxes (Li, 2023a). Another study proposed an improved configuration, in which a volume-varying stirred tank was placed with a free piston upstream of the module. This study evaluated the cyclic dynamics using coupled partial and ordinary differential equations and compared the results with those of the original CCRO and steady RO (Li, 2023b). A third study developed a time-dependent finite-difference model to quantify purge losses, over-pressurization, and reverse flux during purging. It showed purge-period losses of approximately 0.20 kWh/m³ (approximately 9% of the total energy), yielding only modest savings relative to single-stage RO at 50% recovery (Mo and She, 2025). In parallel, operation-based studies have focused on favorable operating conditions and practical implementation of CCRO. For example, the effects of operating pressure, crossflow velocity, and membrane properties on SEC were analytically evaluated to identify favorable operating conditions for semi-batch CCRO (Sutariya and Raval, 2021), while practical operational issues associated with purge-and-refill transients and long-term plant operation were highlighted under CCRO implementation (Gu et al., 2021).

Despite these advances, existing CCRO studies have limitations. Analysis-based studies have provided mechanistic insight into CCRO behavior, but their calibration and validation against plant time-series data have remained limited. Operation-based studies have offered practical guidance on operating conditions and implementation, but they do not provide a sufficiently calibrated dynamic framework for robust control and policy evaluation under realistic plant variability. More broadly, existing studies still show limited tracking of fouling progression and cleaning triggers, and simplified representation of residence time in the recycle loop. These shortcomings hinder their ability to perform robust control and policy evaluations in real-world settings. Control-oriented dynamic simulations are required to address this issue. Such simulations should integrate real plant data, accurately capture residence-time effects in the recycling path, and incorporate membrane fouling with explicit cleaning logic to reliably and quantitatively assess operating policies. Such a framework is also essential for evaluating advanced decision-making methods under realistic CCRO operation.

For the real-time operational optimization of such dynamic processes, recent studies in the field of water treatment have explored advanced techniques, including reinforcement learning (RL) (Yoon et al., 2022). RL focuses on optimizing decision-making processes through trial-and-error and learning from the environment to maximize long-term rewards (Park et al., 2022). This method has proven particularly effective for systems that require dynamic adjustment and control, as it continually refines strategies to improve operational efficiency. Prior RO studies have applied RL to key control tasks. For example, RL has been used to tune transmembrane pressure to achieve a target permeate flux with approximately 99% salt rejection (Bonny et al., 2022), to coordinate feed flow and pump setpoints in two-stage trains to reduce specific energy while maintaining production (Yun et al., 2025). RL has also been employed to select cleaning-in-place thresholds that lower operating costs over extended operational periods (Shim et al., 2025).

Given the unique operational demands of CCRO, particularly the critical need to precisely regulate parameters such as operating pressure and purge timing, RL offers a promising approach to optimizing performance (Song et al., 2025). Nevertheless, RL control studies for CCRO remain limited because CCRO involves nonlinear process dynamics and event-driven purge-and-refill transitions. In practice, CCRO systems are typically operated using predefined rule-based logic, such as fixed recovery targets, concentration-based criteria, or predetermined purge schedules. Although such strategies can provide stable baseline

operation, they are inherently limited in their ability to adapt flexibly to changing feed conditions and plant variability. This limitation is particularly important in CCRO, where the optimal action depends on the evolving process state within each cycle. In this regard, RL is attractive because it can learn a dynamic control policy that adjusts operating actions according to the current process condition rather than following a static rule. In real CCRO facilities, however, safety and operational constraints restrict direct trial-and-error learning; therefore, a calibrated dynamic simulator is essential for the safe development and systematic evaluation of RL-based control strategies. To the best of our knowledge, such a simulator-integrated RL framework has not yet been systematically demonstrated for realistic CCRO plant operation.

This study aims to develop an RL-based control policy for CCRO that reduces specific energy consumption while maintaining high recovery and stable permeate quality under realistic plant dynamics. For this purpose, a dynamic model of a CCRO system was developed to realistically simulate key processes, such as semi-batch mixing, purge events, pressure drops, concentration polarization, recycle residence times, and

membrane fouling. The model was carefully calibrated and validated using data from a full-scale plant running continuous operational cycles. Based on this model, an RL framework was built to enable real-time process control by balancing multiple objectives including permeate production, permeate quality, recovery, energy consumption, and system stability. The performance of the RL-based controller was compared with that of a traditional rule-based approach under the same operating conditions using statistical tests to select the best-performing strategy and to evaluate the significance of the performance differences. To further understand the controller's decision-making process, Shapley additive explanations (SHAP) were applied to rank features and identify the key factors. Thus, this study presents an autonomous RL agent for CCRO operation under realistic plant dynamics. This autonomous control strategy is particularly relevant for small-scale decentralized water treatment plants, where continuous expert supervision is impractical, and it offers a promising alternative to conventional rule-based CCRO operation.

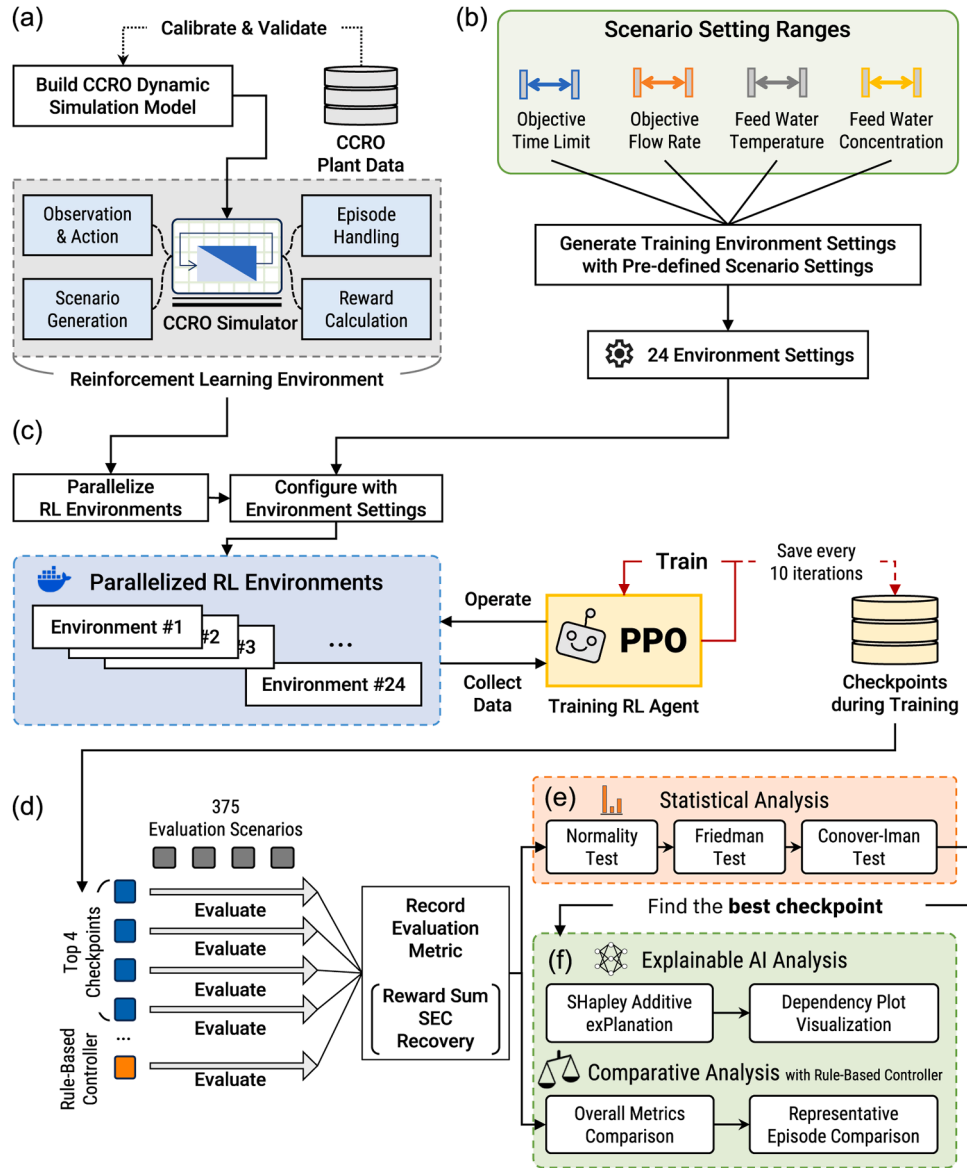


Fig. 1. Flow chart of the study, detailing the sequential steps: (a) construction of the CCRO dynamic simulation model and RL environment components; (b) generation of 24 environment settings for training; (c) parallel training of the RL agent (PPO) and checkpoint saving; (d) evaluation of the checkpoints and rule-based controller using 375 evaluation scenarios; (e) statistical analysis over the four best checkpoint candidates for selection of the best checkpoint and (f) subsequent analysis over the best checkpoint, including explainable AI analysis, and comparative performance analysis with the rule-based controller.

2. Materials and methods

In this study, RL was applied to a full-scale CCRO system using a framework that learns operating policies to meet production targets while respecting quality and energy constraints. The workflow is illustrated in Fig. 1. First, a dynamic CCRO simulation model was built, calibrated, and validated using operating data from a CCRO plant, which served as the environment for RL training and evaluation (Fig. 1a). To train an RL agent with diverse feed water and production target situations, scenario ranges were defined for the production time limit, target flow rate, feed water temperature, and feed water concentration. Combinations of these ranges produced 24 training environment settings (Fig. 1b). These settings configured parallel simulation environments in which the agent was trained using proximal policy optimization (PPO), and checkpoints, including the RL agent's parameters, were periodically saved during training (Fig. 1c).

After training, the saved checkpoints were evaluated for 375 deterministic scenarios. To identify the best-performing checkpoint among the saved checkpoints, the operational performance of the checkpoints was summarized using reward and process metrics, including SEC and

recovery ratio (Fig. 1d). The statistical significance of the differences between the performances of the checkpoints was assessed by testing normality using the Shapiro–Wilk test, applying a Friedman test for overall differences, and conducting a Conover–Iman post-hoc test for pairwise comparisons (Fig. 1e). The best-performing agent loaded with the identified checkpoint was interpreted using SHAP, an explainable AI technique, and compared with a rule-based controller configured to mimic the behavior of the plant data (Fig. 1f). The following sections describe the full-scale CCRO system and simulator, RL environment, training procedure, evaluation scenarios, metrics, statistics, interpretability, and comparative analyses.

2.1. Target CCRO system and process modeling

2.1.1. CCRO system configuration and components

The target process of this study was the CCRO process (Fig. 2a and b). Located in a rural area, the process primarily supplied water by purifying groundwater. In this area, water demand varies with factors such as agricultural seasonality. This necessitates a control strategy that can produce specific water volumes within predefined time limits. In this

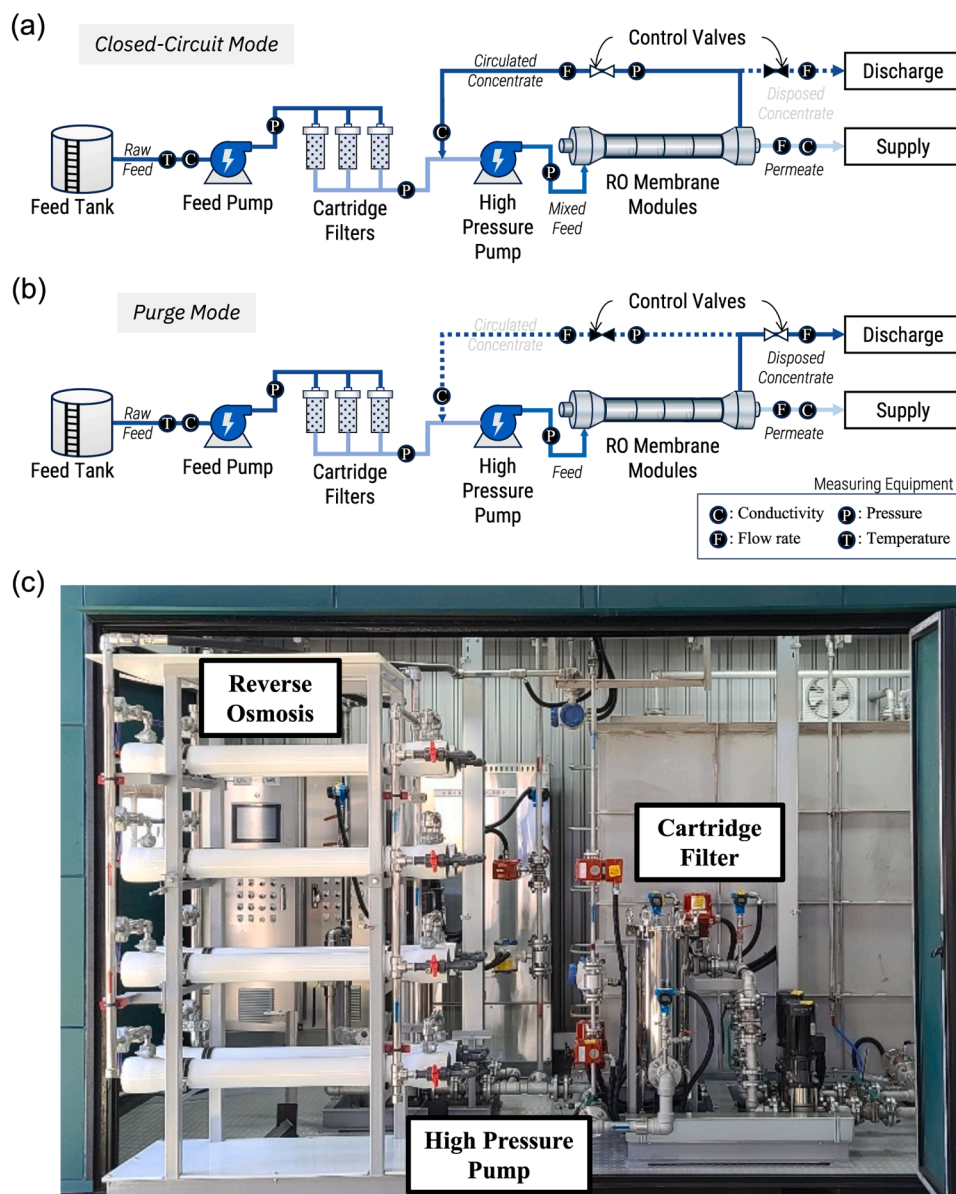


Fig. 2. (a) Schematic diagram of the target CCRO process in closed-circuit (CC) mode and (b) purge mode, and (c) photograph of the CCRO process.

process, raw feed groundwater is transported to cartridge filters to remove impurities that could damage the RO membranes. The process had two modes. If the CCRO process was in the closed-circuit (CC) mode (Fig. 2a), the feed water was mixed with the circulated concentrate to constitute the mixed feed. If the process was in the purge mode (Fig. 2b), the feed flowed through instead. Throughout this study, the feed to the RO pressure vessels was referred to as “mixed feed” for terminological simplicity, although in purge mode, the feed water was delivered directly to the vessels without mixing. The mixed feed was pressurized using a high-pressure pump and delivered to four parallel pressure vessels. Each vessel contained two 4-inch spiral-wound RO elements (BW30–4040, FILMTEC/DuPont) in series. Finally, the permeate was collected for supply, and the concentrate was either circulated (Fig. 2a) or discharged (Fig. 2b), depending on the mode. The actual operational data for the process are provided in Text S1 of the **Supplementary Materials**.

2.1.2. CCRO process simulation through numerical modeling

Based on previous studies (Jeong et al., 2021; Li, 2023a; Park and Davies, 2021), a dynamic numerical model of the CCRO process was implemented in Julia programming language with the ModelingToolkit.jl package. The model simulated water filtration and fouling within the membrane elements, circulation and purging of the concentrate, time lag associated with the residence time in the circulation line, and overall process energy consumption. After the governing ordinary differential equations were formulated, the DP5 solver in DifferentialEquations.jl, based on the Dormand–Prince 5/4 Runge–Kutta method (Dormand and Prince, 1980), was used to solve the system. The parameters were tuned by comparing the simulated solution with the operational data.

In the RO membrane module simulation, two variables played crucial roles— Q_0 (mixed feed flow rate, m^3/h) and R_{sp} (single-pass recovery ratio), as explained in Eqs. (1) and (2) (Li, 2023a).

$$Q_0 = \begin{cases} Q_{\text{feed}} + Q_{\text{circ}} & \text{if mode} = \text{CC} \\ Q_{\text{feed}} & \text{if mode} = \text{purge} \end{cases} \quad (1)$$

$$R_{sp} = \frac{Q_{\text{perm}}}{Q_0} \quad (2)$$

Here, Q_{feed} and Q_{circ} denote the flow rates of the raw feed and recirculated concentrate, respectively (m^3/h). The total system volume was assumed to be constant, and the water in the process was treated as incompressible. Thus, the following equations were derived for the raw feed, concentrates, and permeate, and it was assumed that the raw feed flow was consistently supplied to maintain Q_0 by a low-level controller, such as a proportional–integral–derivative (PID) controller.

$$Q_{\text{feed}} = \begin{cases} Q_0 \times R_{sp} & \text{if mode} = \text{CC} \\ Q_0 & \text{if mode} = \text{purge} \end{cases} \quad (3)$$

$$Q_{\text{circ}} = \begin{cases} Q_0 \times (1 - R_{sp}) & \text{if mode} = \text{CC} \\ 0 & \text{if mode} = \text{purge} \end{cases} \quad (4)$$

$$Q_{\text{disc}} = \begin{cases} 0 & \text{if mode} = \text{CC} \\ Q_0 \times (1 - R_{sp}) & \text{if mode} = \text{purge} \end{cases} \quad (5)$$

$$Q_{\text{perm}} = Q_0 \times R_{sp} \quad (6)$$

where Q_{disc} denotes the flow rate of the discharged concentrate in purge mode (m^3/h). To avoid using different sets of equations for each mode simulation, an integer value M representing the mode and a circulation-rate value α were defined as follows:

$$M = \begin{cases} 0 & \text{if mode} = \text{CC} \\ 1 & \text{if mode} = \text{purge} \end{cases} \quad (7)$$

$$\alpha = (1 - M) \times (1 / R_{sp} - 1) \quad (8)$$

Based on the flow rate and mode variables, differential equations were derived from the mechanics of the RO process. For a detailed explanation of the numerical model and process parameters, refer to Text S2.

The numerical model was calibrated using the actual operational process data totaling 577 min, presented in Text S1. Among the model parameters, the initial membrane resistance ($R_{m,0}$) was calibrated from the operational data, whereas the other parameters such as the average solute molar mass, geometrics of the process and membrane transport parameters were taken from the water quality analysis, process data-sheet and the membrane manufacturer datasheet. As the model used the control variables Q_0 , R_{sp} and M to simulate the CCRO process, these control variables were derived from the actual operational data for each time step, and the numerical model was integrated with the inputs sequentially. To verify parameter robustness, the calibrated model was further validated on independent operational data comprising 228 consecutive data points, totaling 114 min. Two metrics—the coefficient of determination (R^2) and root mean square error (RMSE)—were calculated for the simulation results of the key process variables—permeate flow rate, circulated concentrate flow rate, pressurized feed water pressure, and permeate concentration, using Eqs. (9) and (10), respectively.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (9)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (10)$$

2.2. RL framework for CCRO process control

2.2.1. RL environment design

This section describes the development of an RL environment for a CCRO. The goal is to develop an RL agent that efficiently controls the CCRO process to meet specific production targets, namely, permeate volume and time limits. To this end, the RL environment was constructed using the simulation model detailed in Section 2.1 and Text S2.

Four key components were designed and implemented to facilitate the RL loop:

1. Observation space: Defined by a vector $\mathbf{o}_t$ (Fig. 3b), which is compiled from nine simulated sensor variables to represent the process state (e.g., pressure, flow rate, and concentration) and two production target variables to represent the operational objective (remaining time and target water volume).
2. Action space: An interface that allows the external controller to pass the action $\mathbf{a}_t$ to manipulate the CCRO process (Fig. 3d) by setting the mixed feed flow rate (Q_0 , 4–6 m^3/h), single-pass recovery ratio (R_{sp} , 0.1–0.3), and operation mode (M , CC or Purge) of the process.
3. Episode: An iteration of the operation that proceeds by providing $\mathbf{o}_t$ to the external controller (e.g., the RL agent) (Fig. 3c), which then decides on the next action based on the learned policy until the termination condition, that is, the completion of the production target, is met.
4. Reward function: A function that reflects key process objectives such as timely production target accomplishment, energy use minimization, recovery maximization, permeate quality maintenance, and control stability. The reward at each timestep was formulated as a weighted sum of six penalty terms corresponding to these objectives (Text S3.4 and Table S6). The maximization of the cumulative reward of an episode (i.e., *return*) is the ultimate objective of training the RL agent.

In addition to these components, a scenario generation module was implemented to introduce operational uncertainties into the feed water

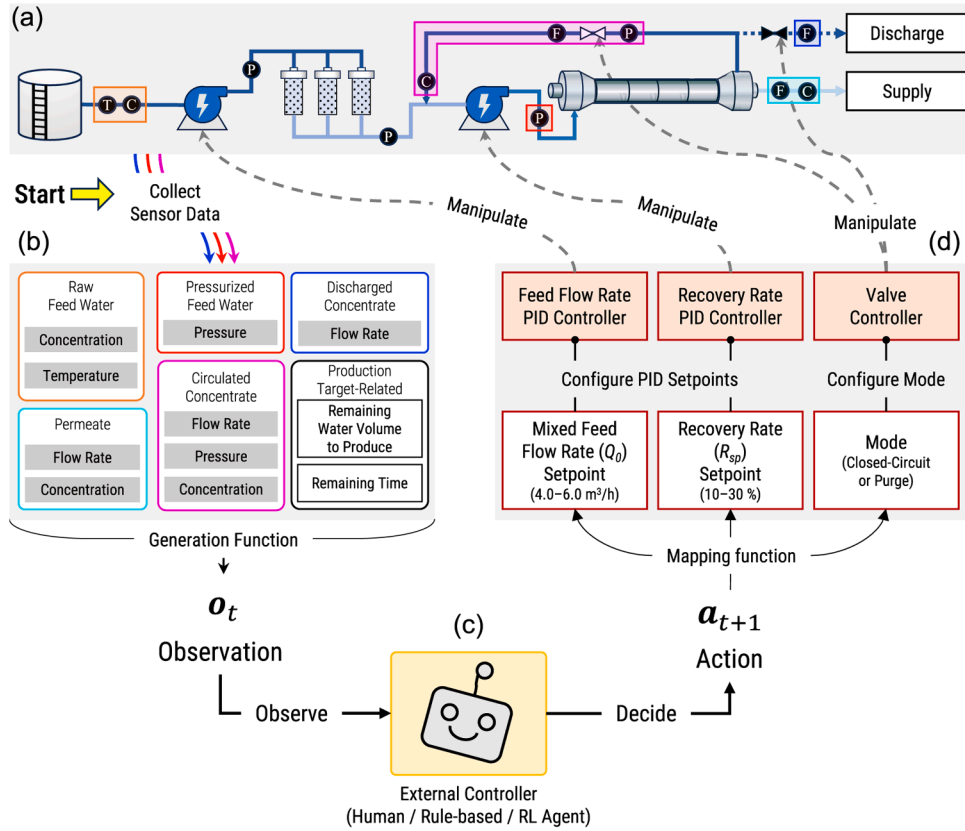


Fig. 3. Schematic diagram of the RL environment for the CCRO process: (a) dynamic numerical model of the CCRO process, (b) observation vector assembled from sensor data and target-related values for input to the external controller, (c) external controller deciding the next action based on the observation, (d) application of the selected action to configure CCRO control variables for the next time step (mixed feed flow rate, single-pass recovery rate setpoint, operating mode).

conditions and production targets. The unit time interval of the environment is set to 60 s. Detailed definitions of the observation and action spaces, episode boundaries, and reward function, including the rationale for observation and action variable selection, action setpoint ranges, and the mathematical formulations of each penalty term and weight assignments, are provided in **Text S3**, and scenario generation is detailed in **Text S4**. The environment was implemented using Julia programming language, Docker, Linux container framework, Python programming language, and Gymnasium Python Library (Towers et al., 2024).

2.3. Implementation and training of RL agent

2.3.1. RL environment setup and parameters

Operational scenarios were generated stochastically within the RL environment to represent variability in production targets and feed water conditions. The production target time limit τ_{obj} (s) and flow rate Q_{obj} (m^3/h) were sampled independently from uniform ranges to define a production target scenario characterized by τ_{obj} , and the corresponding production target volume V_{obj} (m^3). The feed water temperature ($^\circ\text{C}$) and concentration (ppm) time-series scenarios were drawn from Gaussian distributions, with means determined by the environmental setting and standard deviations fixed at $\sigma_{T_{feed}} = 0.25^\circ\text{C}$ and $\sigma_{C_{feed}} = 1.0$ ppm, respectively, manually calibrated to the plant operational data. The sampling process was repeated at the beginning of each episode to construct a unique training scenario. A detailed description of the scenario generation methodology is provided in **Text S4**.

To train a robust agent capable of generalizing across a wide range of operational scenarios, a set of 24 parallel environments was configured, each with unique settings derived from a combinatorial expansion of the key parameters (Fig. 4a). The parameter ranges are defined based on the operational data (Table S2) as follows:

- Production time limits ($\tau_{obj,low}$, $\tau_{obj,high}$): The range from 3.0 to 18.0 h was discretized into eight distinct intervals.
- Objective flow rate ($Q_{obj,low}$, $Q_{obj,high}$): One range from 0.8 m^3/h to 1.6 m^3/h was used, encompassing the observed permeate flow rate range of 0.89–1.28 m^3/h in CC mode.
- Mean feed water temperature ($\mu_{T_{feed}}$): Three discrete mean values (15.0 $^\circ\text{C}$, 16.5 $^\circ\text{C}$, and 18.0 $^\circ\text{C}$) were used, extending beyond the observed feed temperature range of 14.1–15.0 $^\circ\text{C}$.
- Mean feed water concentration ($\mu_{C_{feed}}$): One range from 50 ppm to 100 ppm was used, extending beyond the observed feed concentration range of 75.66–83.94 ppm.

For each of the 24 settings ($8 \times 1 \times 3 \times 1$), a dedicated RL environment was instantiated (Fig. 4b), thereby generating a variety of scenarios for each training iteration. The environments were connected to Python via the HTTP protocol and managed using the Ray Python framework. This parallelized training architecture improves the sample efficiency and stabilizes the learning process by providing the agent with a diverse batch of experiences at each training step.

2.3.2. RL algorithm selection and training procedures

The PPO algorithm, an on-policy gradient method known for its stability and performance, was selected to train the control agent (Schulman et al., 2017). To accommodate the hybrid action space of the CCRO environment, which contains both discrete and continuous actions, a tailored PPO agent implementation in the RLlib Python library was used. The hyperparameters used are listed in **Text S5**.

The PPO has two main components—actor network and critic network, both of which are neural networks that receive the observation vector $\mathbf{o}_t$ (Fig. 4c). The critic network acts as the value function $V(\mathbf{o}_t)$, the expected return from observation $\mathbf{o}_t$, and the actor network

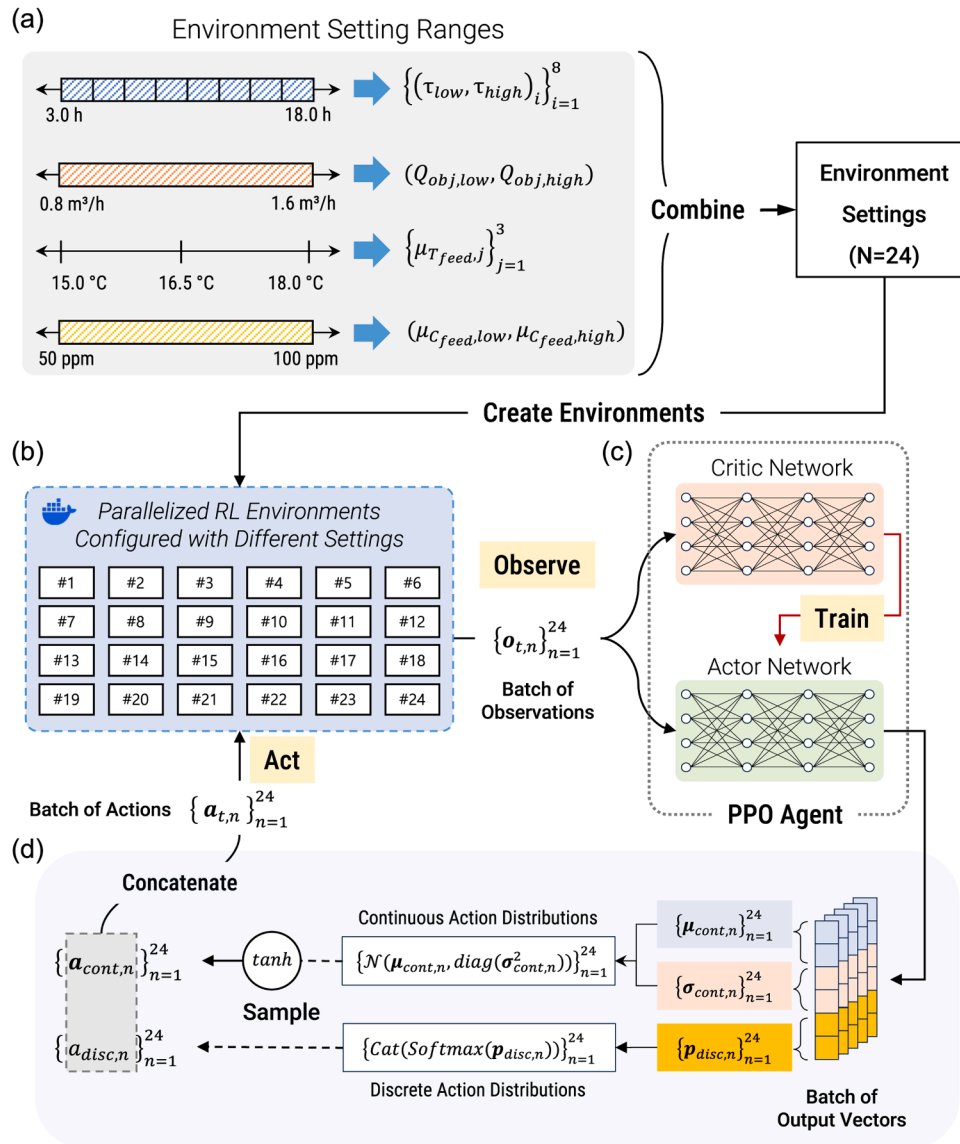


Fig. 4. Schematic diagram of the training procedure: (a) 24 unique environment settings from pre-defined ranges, (b) configuration of an equal number of RL environments, (c) training of the RL controller using the PPO algorithm, (d) hybrid action-space formulation tailored for the controller.

represents the policy $\pi(a_t|o_t)$, a function mapping the observation to probability distributions over actions. The training process was iterative. At each iteration, the agent collected a batch of experience trajectories $(\langle o_t, a_t, r_t, o_{t+1} \rangle)$ by running the policy in 24 parallel environments. The critic's value estimates are then used to compute the advantage estimates using generalized advantage estimation, which quantifies how much better an action is than the baseline expectation (Shim et al., 2024). The actor network was subsequently updated by maximizing the PPO's surrogate objective function with respect to these advantages.

Note that the actor network does not directly return a_{t+1} , but rather returns the parameters of the distributions from which the next actions are sampled. If the action space is continuous, the actor network outputs the mean and standard deviation to parameterize a multivariate Gaussian distribution with a diagonal covariance matrix, from which continuous actions are sampled. Similarly, if the action space is discrete, the network outputs logits that parameterize a categorical distribution. To handle the hybrid action space used in this study, the actor network outputs a 6-length vector (Fig. 4d). The first to fourth elements parameterized a multivariate Gaussian distribution for two continuous actions (Q_0 , R_{sp}), and the fifth to sixth elements parameterized a categorical

distribution for a discrete action (M). Subsequently, each action is sampled independently and combined into the action vector a_t (Fig. 4c).

In this study, a PPO agent interacted with 24 parallel RL environments to collect a batch of experience trajectories, each comprising episodes from a single environment. At each training iteration, the agent is updated using a batch of data. In addition, a checkpoint consisting of the neural network parameters of the agent was saved every 10 training iterations.

2.4. Performance evaluation and analysis

Various uncertainties are involved in training an RL agent, including random scenario generation and action sampling. Although these uncertainties are essential for training a robust agent, they make it difficult to evaluate its performance objectively. Moreover, because of the intrinsic nature of RL training, the performance of an RL agent may fluctuate or even collapse after convergence (Croll et al., 2024; Sutton and Barto, 1998). Therefore, evaluation methods were implemented to reduce uncertainty by generating strictly controlled evaluation scenarios and sampling only the most probable actions from the action distributions, namely the mean values for continuous actions and the

option with the largest logit for discrete actions. The best-performing checkpoint was selected from the saved checkpoints. The agent restored with the best-performing checkpoint is analyzed using SHAP and compared with a rule-based controller (Lundberg and Lee, 2017).

2.4.1. Generation of evaluation scenarios

For objective and reproducible evaluation, scenarios were constructed from predetermined lists of fixed values for production targets and feed water conditions, rather than using stochastic sampling from the environmental settings employed during training. Specifically, the conditions representing the parameter ranges of the production targets and feed conditions employed during training, as explained in Section 2.3.1, were used.

- Production time limits (τ_{obj}) was selected from a set of five evenly spaced values between 3.0 and 18.0 h.
- The objective flow rate (Q_{obj}) was selected from a set of five evenly spaced values in 0.8 and 1.6 m³/h.
- Mean feed water temperature ($\mu_{T_{feed}}$) was selected from [15.0°C, 16.5°C, 18.0°C].
- Mean feed water concentration ($\mu_{C_{feed}}$) was selected from a set of five evenly spaced values between 50 and 100 ppm.

For each parameter, five evenly spaced values were selected to capture both the boundary and interior of the respective training range, while the mean feed water temperature retained its original discrete set. The combination of these discrete sets yielded 375 unique evaluation scenarios (5×5×3×5).

2.4.2. Statistical analysis of results

The RL agents loaded with the saved checkpoints operated under the evaluation scenarios, and the obtained operation trajectories and return values were collected. To identify the best-performing checkpoint, the four that achieved the highest mean return values during the evaluation were selected for subsequent statistical analyses. To determine the appropriate statistical approach, the assumption of normality of the return data in the 375 evaluation scenarios was assessed by performing a Shapiro–Wilk test on the residuals of a general linear model fitted to the return data (Shapiro and Wilk, 1965). Accordingly, an appropriate omnibus test was applied to the data to detect differences between the checkpoints. A parametric analysis of variance test was used when the normality assumption was held, and a nonparametric Friedman test was used when it was not (Fisher, 1970; Friedman, 1937). If the omnibus test indicated significant differences among the checkpoints, a post-hoc test (Tukey's honest significant difference test for parametric data or Conover-Iman test for nonparametric data) was applied to identify the best-performing checkpoint (Conover and Iman, 1981; Tukey, 1949). All statistical tests were conducted at a significance level of 0.001.

2.4.3. Interpretation of learned policy via SHAP analysis

To interpret the trained RL agent's decision-making process, a SHAP analysis was performed on its critic network (Lundberg and Lee, 2017; Tayara et al., 2024). The critic network, which predicts the expected return from a given observation, serves as an effective proxy for the policy because the actor is trained to take actions that maximize this predicted value. Therefore, understanding which observational features the critic considers important can reveal the key drivers of the agent's learning strategy. For this analysis, the model-agnostic KernelSHAP method from the SHAP Python library was employed.

For each prediction, the SHAP analysis assigns a SHAP value to each input feature, representing the specific contribution of the feature in pushing the output (the expected return) away from the baseline average. The sign of the SHAP value indicates the direction of the contribution (positive for a higher return and negative for a lower return), whereas its magnitude indicates the importance of that contribution. In this study, the SHAP values were aggregated across all

evaluation scenarios to generate two key visualizations—summary plots to rank features by global importance (mean absolute SHAP value), and dependence plots for the four most influential features to illustrate the relationship between the feature values and their specific impact.

2.4.4. Comparison with rule-based control

After the best-performing checkpoint was identified, the operations of the RL agent with the checkpoint were compared with those of a rule-based controller in the evaluation scenarios, with the following rules set based on the plant data:

1. Operate for 30 min in CC mode and for 1 min in purge mode.
2. Increase Q_0 gradually during the first 10 min of CC mode from 4.2 m³/h to 5.8 m³/h, then hold until purge mode.
3. Set R_{sp} as 20% throughout an operation.
4. Operate until the production target permeate volume is met, then purge for 1 min after the target is met.

Subsequently, the SEC and recovery ratio for each operation were calculated and compared, because a comparison of returns may be misleading. In addition, one evaluation episode was selected as a representative case to compare the RL agent and the rule-based controller. This episode was configured with the following median parameters—a production target flow rate of 1.2 m³/h, a time limit of 10.5 h, a raw feed temperature of 16.5°C, and a concentration of 75 ppm.

The SEC and recovery ratio at each time step and for the entire operation were calculated using the following formulas:

$$SEC(t) = \frac{Power(t)}{Q_{perm}(t)} \quad (11)$$

$$Total\ SEC = \frac{\sum [SEC(t) \times Q_{perm}(t)]}{\sum Q_{perm}(t)}, \text{ for } t \in [0, t_{termination}] \quad (12)$$

$$Recovery\ ratio(t) = \frac{\sum_{\tau=t_{CC\ start}}^t V_{perm}(\tau)}{\sum_{\tau=t_{CC\ start}}^{t_{purge\ end}} V_{feed}(\tau)}, \text{ for } t \in [t_{CC\ start}, t_{purge\ end}] \quad (13)$$

$$Total\ recovery\ ratio = \frac{\sum V_{perm}(t)}{\sum V_{feed}(t)} \quad (14)$$

where $t_{termination}$ is the time step of the operation termination, $t_{CC\ start}$ and $t_{purge\ end}$ are the start and end time steps of each CC purge cycle, respectively.

In addition to the SEC and recovery ratio, which measure the energy and water resource efficiency, we calculated the intra-episode standard deviations of the action variables, Q_0 and R_{sp} , as operational stability metrics, and compared them between the two controllers using the Wilcoxon signed-rank test (Wilcoxon, 1945).

3. Results and discussion

3.1. Numerical model validation using full-scale CCRO data

After the calibration of the CCRO simulation model, a comparison between the simulated output and observed operational data is presented for four key process variables—permeate flow rate, circulated concentrate flow rate, membrane inlet pressure, and permeate concentration—in Fig. 5 and summarized in Table 1. As expected, the simulated permeate flow rate shows an almost exact match with the observed data (calibration and validation RMSE < 0.001 m³/h) because this variable was a direct target of the model's recovery ratio calculation, as explained in Section 2.1.2 (Fig. 5a). A strong correlation was observed for the flow rate of the circulated concentrate (calibration RMSE = 0.14 m³/h, validation RMSE = 0.25 m³/h) (Fig. 5b). Minor deviations occur primarily during mode transitions, where operational data can reflect a

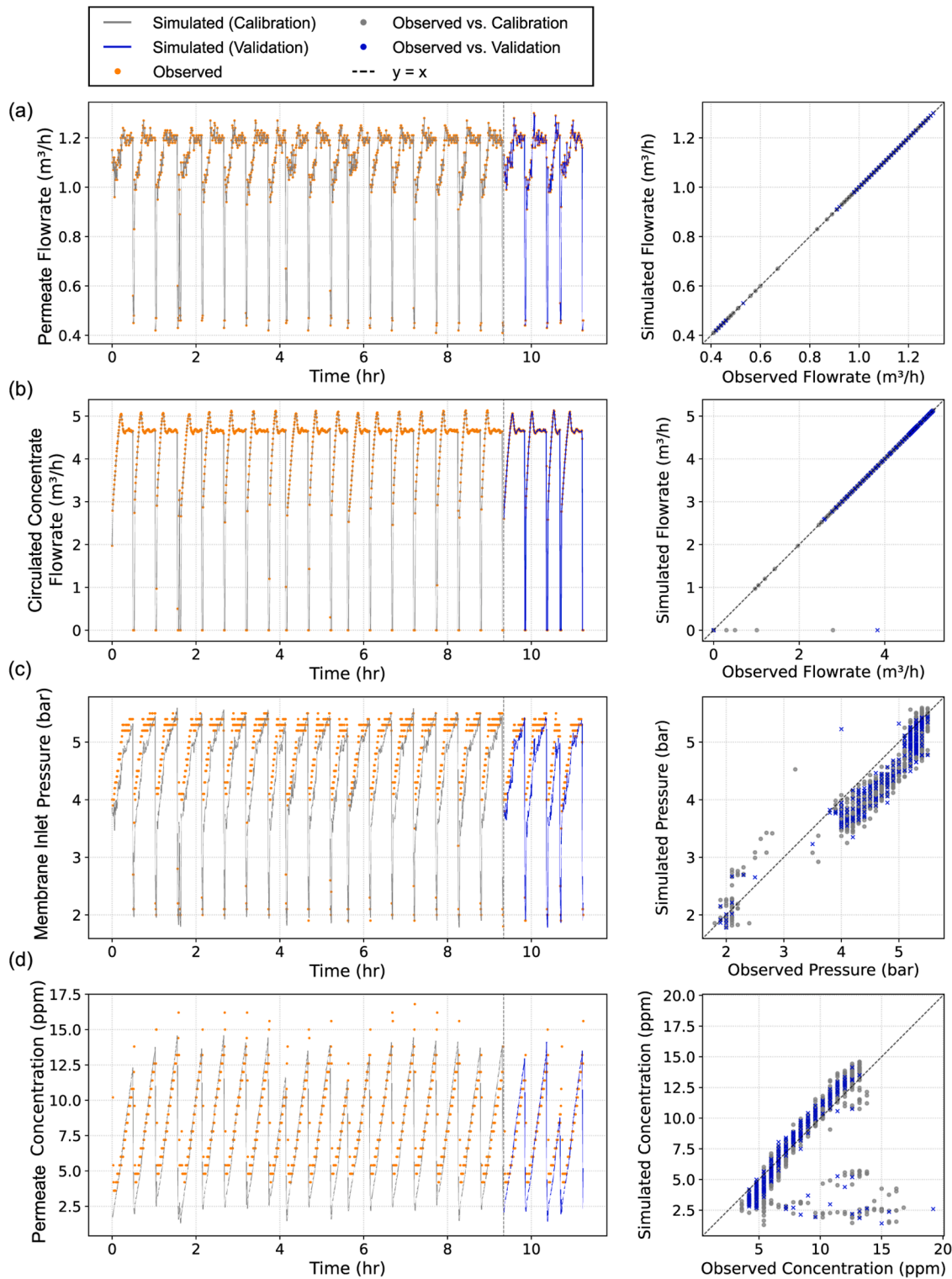


Fig. 5. Calibration and validation results of the CCRO simulation model for four output variables: (a) permeate flow rate, (b) circulated concentrate flow rate, (c) membrane inlet pressure, and (d) permeate concentration. Left panels show time-series comparisons between observed (orange dots) and simulated values (grey lines: calibration; blue lines: validation), with the vertical dashed line indicating the boundary between calibration and validation periods. Right panels show observed-versus-simulated scatter plots for calibration (grey dots) and validation (blue crosses) data points.

blend of the CC and purge modes, a state that cannot occur in the simulation model.

The model captured the overall dynamic behavior of the membrane inlet pressure and permeate concentration (Fig. 5c and d). As the variables were affected by unmodeled factors, such as measurement

resolution, which excessively affected the R^2 values, only the RMSE values were quantified (Moon et al., 2025a). The model robustly predicted the variables, with a validation RMSE of 0.37 bar and 2.73 ppm, respectively. Although the metrics on the validation set are higher than those on the calibration set (calibration RMSE for membrane inlet

Table 1

Comparison of calibration and validation performance metrics of the CCRO simulation model across four representative variables.

Variable	Calibration R^2	Calibration RMSE	Validation R^2	Validation RMSE
Permeate flow rate	1.00	0.00 m^3/h	1.00	0.00 m^3/h
Circulated concentrate flow rate	0.99	0.14 m^3/h	0.95	0.25 m^3/h
Membrane inlet pressure	-	0.28 bar	-	0.37 bar
Permeate concentration	-	2.56 ppm	-	2.73 ppm

pressure = 0.28 bar, calibration RMSE for permeate concentration = 2.56 ppm), this result indicates the high predictive performance of the simulation model across the ranges of the observed variables (2.0–5.5 bar and 2–16 ppm, respectively). Existing physics-based CCRO modeling studies have generally been validated either qualitatively or against numerical reference solutions, without reporting quantitative prediction metrics against real operational data (Li, 2021; 2023a; b; Mo and She, 2025; Sutariya and Raval, 2021). In contrast, the present work quantitatively evaluates a physics-based CCRO simulation model against full-scale plant data and further verifies parameter robustness using an independent validation dataset.

In summary, the developed numerical CCRO simulation model demonstrated high predictive performance. The model achieved excellent accuracy for the flow rate variables on both the calibration and validation sets. For more complex pressure and concentration dynamics, it successfully captured the essential trends and magnitudes, as confirmed by the low validation RMSE values. While point-to-point correlations (R^2) directly compared with operational data can be

affected by unmodeled noise and minor timing offsets, the model's ability to accurately predict the physical magnitudes of key variables is sufficiently reliable for subsequent RL studies.

3.2. Training results of RL

A PPO-based RL agent was trained in CCRO RL environments across a wide range of environmental settings (Section 2.3). Fig. 6 shows the evolution of the episode return and SEC during training. Training was conducted for 752 iterations, equivalent to 18,048 episodes and 10,910,629 environmental steps. The total wall clock time for the training process is detailed in Text S7.

The average episode return increased rapidly during the initial stage of training, indicating successful learning (Fig. 6a). This rapid increase reflects the agent's discovery and adoption of effective strategies, leading to significant improvements in cumulative rewards. The average return then converged in the intermediate stage, demonstrating stabilization as the agent approached the optimal policy. The SEC did not exhibit a monotonic trend during training (Fig. 6b), unlike the steadily increasing and converging episode returns shown in Fig. 6a. This was expected because the reward function balances multiple objectives, including minimizing SEC, maximizing recovery, and ensuring punctuality (Text S3.4). The recovery ratio steadily increased during the initial improvement stage and stabilized during the convergence stage (Fig. 6c). The remaining target production time, defined as the difference between the objective production time (3–18 h) and the actual time required to reach the target volume of water production, increased rapidly from negative values early in training to avoid penalties for exceeding the time limit (Fig. 6d). However, as the training progressed, the remaining time approached zero rather than increasing further. That is, the agent learned to optimize reward components, such as SEC and recovery, by fully utilizing the assigned time limits to achieve efficiency

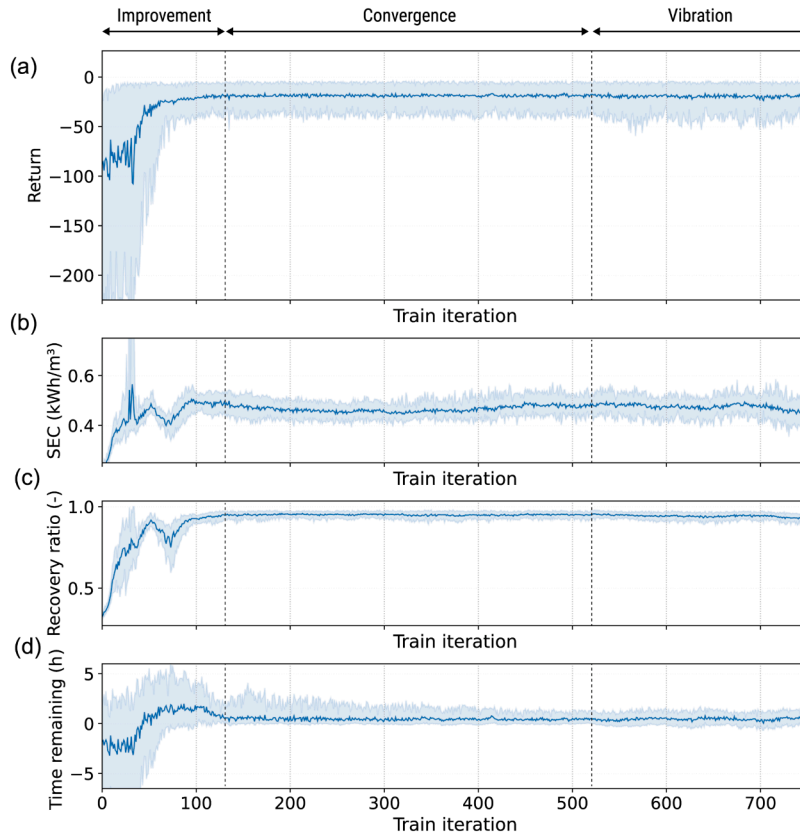


Fig. 6. Evolution of (a) episode return, (b) SEC, (c) recovery ratio, and (d) remaining production target time during the training. Blue solid lines and shaded areas indicate the mean value and the minimum-to-maximum range of the variables at each train iteration.

gains rather than producing the target water volume as early as possible.

In the final stage, minor policy fluctuations were observed, which could be attributed to the interaction between PPO's on-policy updates and the reward function structure. The punctuality penalty (Eq. S49) accumulates linearly per timestep beyond the production time limit, creating a sharp transition in the reward landscape. As the agent rapidly learned to meet the time constraint in the early training stages, the punctuality penalty feedback diminished substantially (Fig. 6d). The agent then exploited this reduced penalty to maximize other objectives such as SEC and recovery by operating near the time constraint boundary, which consequently led to overshooting the time limit and reintroducing punctuality penalty signals. Since PPO updates rely solely on the current policy's experience, we interpret these intermittent penalties as sufficient to perturb the converged policy. This result highlights the importance of evaluating and selecting the best-performing checkpoint among those saved at different training stages (Section 2.4).

3.3. Results of quantitative and qualitative evaluation

To objectively assess the agent's learning outcomes, 75 agent states (i.e., checkpoints) were saved throughout training. Each checkpoint was evaluated across 375 scenarios, using the following key performance indicators—episode return, SEC, recovery ratio, and target production time. These metrics provide comprehensive insights into the efficiency and reliability of an agent's learned policy (Fig. 7).

The episode returns quickly stabilized after the initial phase, reflecting efficient learning and consistent performance (Fig. 7a). The SEC plot shows inherent variability, reflecting the expected trade-offs induced by the multi-objective reward function, which was also observed during the training process (Fig. 7b). In contrast, the recovery ratio (Fig. 7c) exhibited a distinct pattern. Before the agent learned to

operate the system efficiently, the mean recovery ratio was very high. This occurred because the agent erroneously neglected the “purge” operation, so the system never discarded any residual, inflating the recovery ratio while achieving poor overall returns. As the agent improved, it learned to apply the purge function properly, reducing the mean recovery ratio to a more realistic operational value of approximately 0.95. This confirms the agent's successful acquisition of appropriate cycle management and termination policies.

In the plot (Fig. 7d) of the remaining target production time (i.e., the time left until the fixed production period ended), the mean values after checkpoint 50 remained only slightly above zero. Rather than displaying a pronounced overshoot, the policy results in minor positive deviations, meaning the agent completes water production slightly before the maximum allowed time, thereby avoiding penalties for exceeding it. This marginal positive difference indicates stable and efficient control during the evaluation, balancing multiple objectives while meeting punctuality.

The red crosses in Fig. 7 highlight the top four checkpoints (11, 18, 50, and 51) based on their high episode returns, which indicate reasonably low SEC, high recovery ratios, and efficient use of time. The statistics in Table 2 show that Checkpoint 18 achieved the lowest mean SEC (0.489 kWh/m³) and the highest mean recovery ratio (0.955), indicating the most effective control behavior. However, because Checkpoints 50, 51, and 11 exhibit broadly similar performances, with a maximum difference in mean return of only 0.126, selecting Checkpoint 18 as the best checkpoint requires further confirmation to ensure that the chosen policy (i.e., checkpoint) performs robustly better in average performance across diverse operational conditions. Therefore, rigorous statistical tests are conducted to determine whether the observed differences in returns across checkpoints are statistically significant, enabling the objective identification of the single best-performing

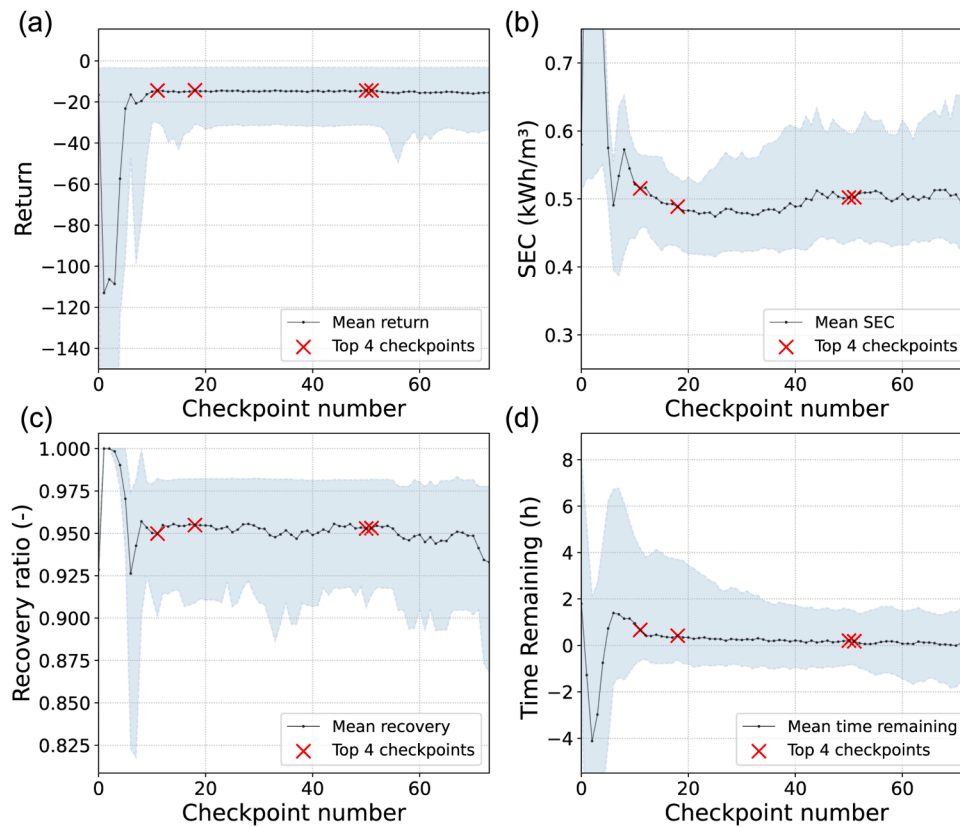


Fig. 7. Evolution of (a) episode return, (b) SEC, (c) recovery ratio, and (d) remaining production target time during the evaluation. Black solid lines and shaded areas indicate the mean and minimum-to-maximum range of the variables at each checkpoint, and red crosses indicate the four checkpoints with the highest average returns.

Table 2
Statistics of metrics for four checkpoints with the highest average returns.

Checkpoint Number		18	50	51	11
Return	Mean	-14.316	-14.434	-14.437	-14.442
	Minimum	-30.720	-31.175	-31.308	-29.863
	Maximum	-3.149	-3.177	-3.182	-3.331
	Standard deviation	7.225	7.552	7.565	6.955
SEC (kWh/m ³)	Mean	0.489	0.502	0.502	0.516
	Minimum	0.430	0.437	0.439	0.457
	Maximum	0.531	0.596	0.596	0.564
	Standard deviation	0.021	0.026	0.026	0.029
Recovery ratio	Mean	0.955	0.953	0.953	0.950
	Minimum	0.909	0.906	0.914	0.901
	Maximum	0.982	0.982	0.982	0.982
	Standard deviation	0.015	0.015	0.015	0.019
Remaining production target time (h)	Mean	0.417	0.205	0.177	0.666
	Minimum	-0.617	-0.833	-0.833	-0.650
	Maximum	3.700	1.550	1.483	4.183
	Standard deviation	1.007	0.653	0.638	1.076

checkpoint.

3.3.1. Statistical performance analysis

To ensure statistical precision when selecting the best checkpoint, it is important to validate the underlying assumptions of the data distribution. For this purpose, a Shapiro–Wilk test was performed on the evaluation of return residuals using a general linear model and a strict significance threshold. The test result ($p < 0.001$) and the accompanying quantile–quantile (Q–Q) plot (Fig. 8a), where data points diverge from the straight diagonal line, clearly indicate that the residuals deviate from a normal distribution. As standard parametric statistical tests assume normality, these findings necessitate applying nonparametric tests for any further comparisons between checkpoints.

To test for statistically significant differences across the checkpoints, the Friedman test was applied as an omnibus test, using 375 evaluation scenarios as repeated measures. The Friedman test yielded a highly significant result ($p = 1.11 \times 10^{-26}$), indicating that the performance distribution of at least one checkpoint differed statistically from the others (Piadeh et al., 2023). To identify the specific checkpoint pairs that exhibited significant differences, a post-hoc Conover–Iman test was conducted (e Silva et al., 2016). The resulting p -values for these pairwise comparisons are presented as a heat map in Fig. 8b. The analysis demonstrated that Checkpoint 18 significantly outperformed all other candidates in most evaluation scenarios, with higher mean return values

and a p -value lower than 0.001 in the Conover–Iman test with the other checkpoints. Notably, Checkpoints 50 and 51, which were saved immediately before the minor fluctuations observed in the training progress, did not differ significantly ($p = 0.976$), confirming their convergence prior to this time difference. Conversely, Checkpoint 11, the earliest saved model, was significantly outperformed by all others.

Based on the detailed statistical analysis, Checkpoint 18 was identified as the best-performing checkpoint. This checkpoint not only had the highest mean return but also demonstrated statistically significant superiority over the other candidates, thereby justifying its selection for further study.

3.3.2. Performance comparison with rule-based control

A comparative analysis between the RL agent (Checkpoint 18) and the conventional rule-based controller clearly demonstrates its superiority in terms of efficiency and adaptability. Both controllers were evaluated across 375 operational scenarios, identical to those used in the checkpoint evaluations, and the result is summarized in Fig. 9. The scenarios reflect varying raw feed-water concentrations (50–100 ppm) typical of actual practice, a critical process variable that directly influences CCRO energy consumption and recovery performance.

The rule-based controller operated with static decision rules, maintaining a fixed single-pass recovery rate setpoint of 20%, an unvarying feed flow rate ramp, and CC-purge operation strategies, regardless of the situation (Fig. 9a). Consequently, as the feed concentration increased, SEC increased sharply (orange violin distribution in Fig. 9b), whereas the recovery ratio remained constant (orange violin distribution in Fig. 9c). The overall mean performance of the rule-based controller was 0.563 kWh/m³ SEC and 91.9% total recovery rate. By contrast, the RL agent demonstrated both dynamic and adaptive control. By adjusting operating parameters in response to changing feed concentration and compensating the recovery rate for high-concentration feed scenarios to improve energy efficiency (blue violin distributions in Fig. 9b and c), it achieved superior performance (0.489 kWh/m³ SEC and 95.5% recovery rate). Compared with the rule-based approach, this resulted in a 13.14% reduction in SEC and a 3.92% increase in the recovery rate.

The ability of the RL agent to adjust its actions flexibly according to the current feed concentration is attributed to the design of the multi-objective reward function, which balances two competing goals—energy efficiency (minimizing SEC) and water recovery (maximizing product yield from feed water) (Text S3.4). Consequently, the RL agent implicitly learns an effective Pareto-optimal policy, which implies that any improvement in one objective requires accepting modest changes in the other. This trade-off behavior is evident in the agent's response to varying feed-water concentrations.

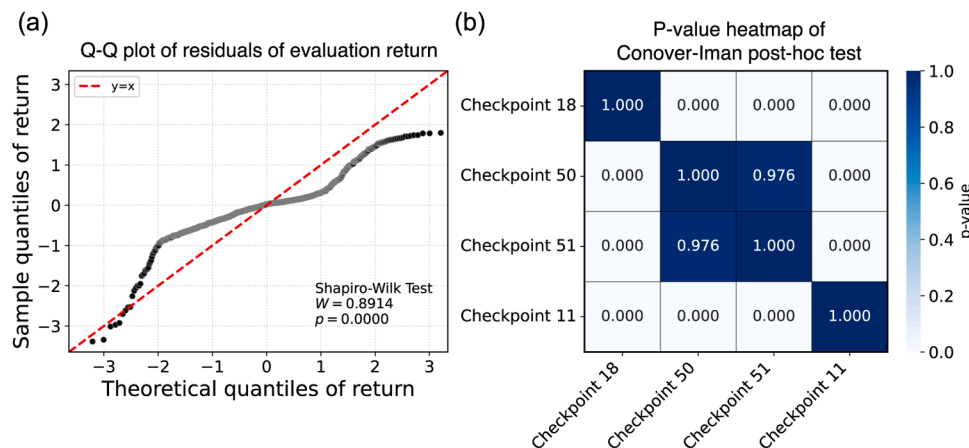


Fig. 8. Statistical comparison of the top four agent checkpoints: (a) Quantile–quantile (Q–Q) plot for the evaluation return residuals, (b) Heatmap of p -values from the post-hoc Conover–Iman test for pairwise comparisons.

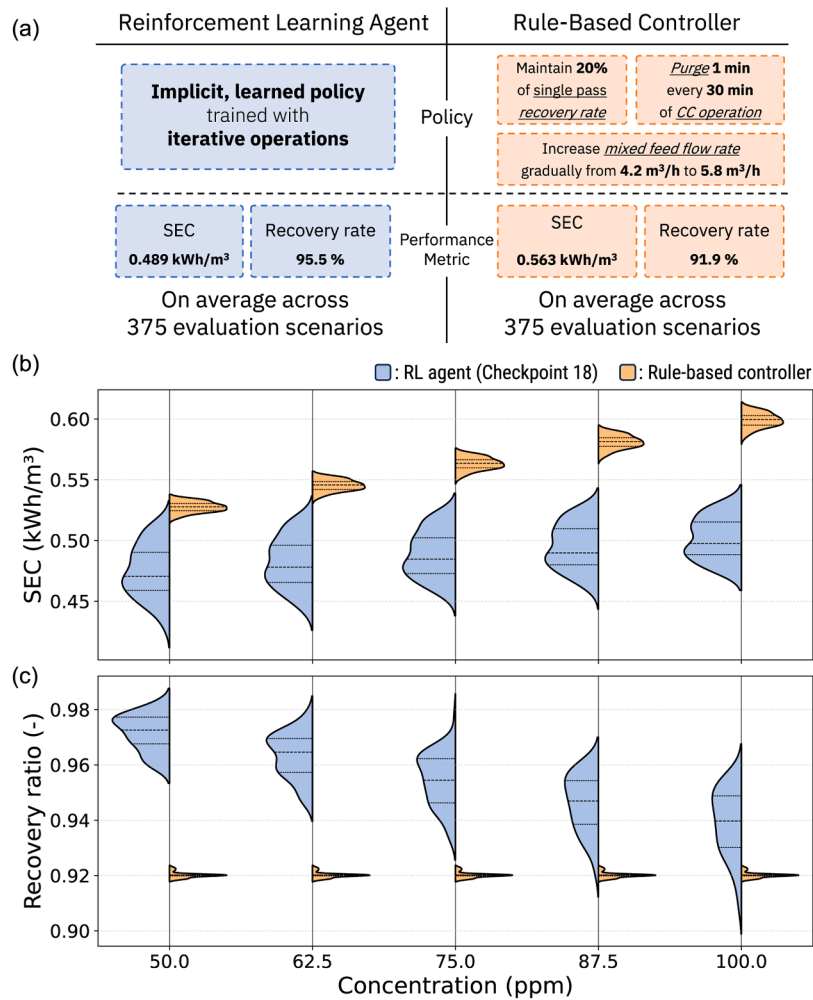


Fig. 9. Comparison of the best-performing RL agent (Checkpoint 18) and a rule-based controller: (a) policy summary and aggregate metrics across 375 scenarios; (b) violin plots of SEC as a function of mean feed water concentration of the evaluation scenarios; and (c) violin plots of recovery ratio as a function of mean feed water concentration of the evaluation scenarios.

- At low concentrations, filtration is energetically less demanding. The agent prioritized a high water recovery rate (~97–98%) by accepting slightly higher SEC values, given that energy costs were inherently low.
- However, filtration at high concentrations is expensive. The RL agent strategically reduced the recovery rate to ~94%, maintaining a low SEC by adjusting the transmembrane pressure and flow rate, thereby minimizing energy penalties.

Beyond the energy and recovery advantages, the control action stability metrics—standard deviation values of the action variables—reveal distinct control characteristics. The rule-based controller followed a predetermined feed flow rate ramp with substantial Q_0 variability (standard deviation: 0.531 m³/h), while maintaining a fixed R_{sp} throughout each episode. In contrast, the RL agent adaptively adjusted R_{sp} (standard deviation: 0.010) while maintaining considerably lower Q_0 variability (standard deviation: 0.247 m³/h; Wilcoxon signed-rank test $p < 0.001$), partly attributable to the stability penalty term in the reward function (Text S3.4). The result of the comparison on the metrics is summarized in Table S8.

To summarize, unlike fixed-setpoint controllers, which offer no operational flexibility when process conditions change unexpectedly, the RL agent demonstrated robust, real-time decision-making derived from simulated operational experience with substantial differences in the efficiency and stability, allowing it to handle real-world operational

variability. Furthermore, the agent's computational efficiency is validated for real-time execution on-site edge hardware, as shown in Text S7, demonstrating its practical feasibility for on-site implementation.

3.3.3. Operation comparison with a rule-based controller

Fig. 10 illustrates the operation of the RL agent and rule-based controllers during a representative evaluation episode. Among the evaluation episodes, the episode with the median production target flow rate (1.2 m³/h), production target time limit (10.5 h), feed water temperature (16.5°C), and concentration (75 ppm) was selected as the representative episode. Note that Fig. 10 presents the key operational variables; the complete set of observation and action variables during the episode is detailed in Text S6.

The rule-based controller, shown as orange markers in Fig. 10, operated with the pre-defined rules—mixed feed flow rate increasing from 4.2 m³/h to 5.8 m³/h (Fig. 10a), consistent single-pass recovery ratio of 0.2 (Fig. 10b), and a fixed cycle of a 30-min CC mode followed by a 1-min purge mode (Fig. 10c), as explained in Section 2.4.3. As expected, the controller demonstrated highly repetitive cyclic behavior, with process observations remaining within the expected operational bounds throughout the operation.

In contrast, the RL agent, shown as blue markers in Fig. 10, demonstrated adaptive actions with higher single-pass recovery ratios, lower mixed feed flow rates, and longer CC phases than the rule-based controller (Fig. 10a–c). The agent demonstrated a gradually

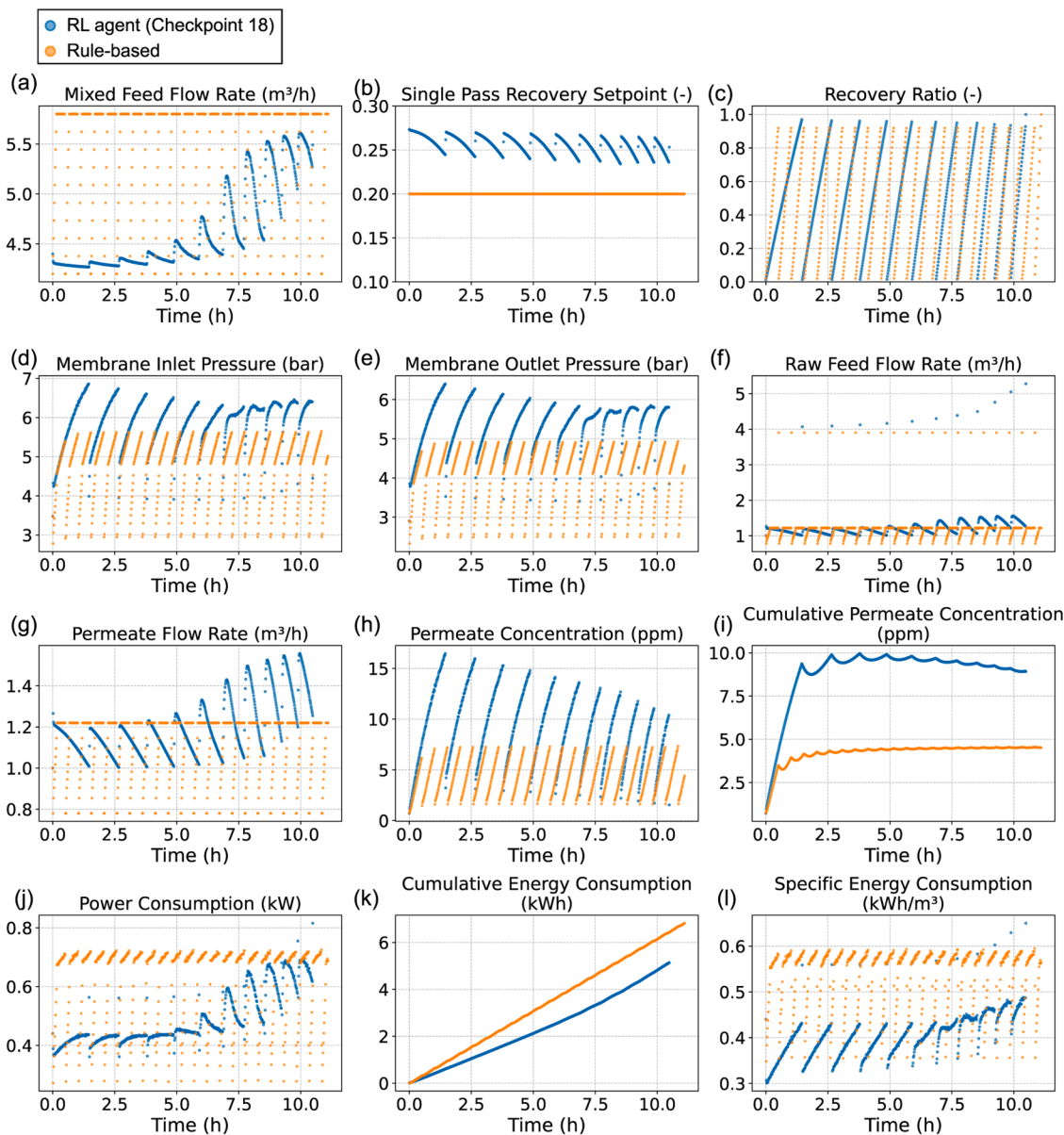


Fig. 10. Operation history comparing the best-performing RL agent (blue) and the rule-based controller (orange) during a representative evaluation episode: (a) mixed feed flow rate setpoint, (b) single-pass recovery ratio setpoint, (c) recovery ratio, (d) membrane inlet pressure, (e) membrane outlet pressure, (f) raw feed flow rate, (g) permeate flow rate, (h) permeate concentration, (i) cumulative permeate concentration, (j) power consumption, (k) cumulative energy consumption and (l) specific energy consumption throughout the operation.

decreasing single-pass recovery ratio and raw feed supply within each CC phase (Fig. 10b and f), and a reasonable energy and feed water minimization strategy as the mixed feed concentration increased. Consequently, the cumulative power consumption and specific energy consumption of the RL agent were lower than those of the rule-based controller throughout the operation (Fig. 10j–l). The membrane inlet and outlet pressures were higher during the operation of the RL agent, as was the permeate concentration (Fig. 10d, e, and h), which was a reasonable outcome of the longer CC phases of the RL agent. Nevertheless, the pressures were significantly lower than the membrane's maximum operating pressure (41 bar), demonstrating that the adaptive policy of the RL agent successfully maximized energy efficiency and recovery under safe operating conditions.

In the perspective of operational reliability, the RL agent presented smoother mixed feed flow rate trajectory throughout the episode. In contrast, the rule-based controller followed a predetermined feed flow rate ramp that produced substantially larger setpoint swings (Fig. 10a),

consistent with the higher standard deviation reported in Section 3.3.2. While the rule-based controller maintained a fixed single-pass recovery ratio throughout the episode (Fig. 10b), the RL agent adaptively modulated this setpoint with minimal fluctuation (standard deviation: 0.010). Owing to the more rapid concentration buildup driven by the higher single-pass recovery ratio of the RL agent (Fig. 10b), the instantaneous permeate concentration was higher in both magnitude and rate of change (Fig. 10h). However, the cumulative permeate concentration (Fig. 10i) remained below the 12 ppm, the quality threshold in the reward function (Table S5). These results indicate that the RL agent did not compromise operational stability and permeate quality while maximizing the efficiency.

3.3.4. SHAP-based model interpretation

A SHAP analysis was conducted to examine whether the learned value function—a function estimating the expected sum of future rewards from the current state—of the best-performing RL agent (Section

3.3.1) captured an accurate understanding of the underlying process dynamics and operational constraints, and the result was summarized in Table 3. A SHAP value represents how much each input feature contributes to the model's predicted cumulative return. From the summary plots (Fig. 11a), we found that the agent's value estimates were primarily dominated by two production target-related features: remaining production volume (mean absolute SHAP value = 6.148) and remaining production time (mean absolute SHAP value = 1.425), as detailed in Table 3. Among the process-related features (Fig. 11b), feed water concentration (mean absolute SHAP value = 0.286) and membrane inlet pressure (mean absolute SHAP value = 0.123) exerted the most significant influence. This ranking is intuitive, as these four variables directly indicate progress toward the production target and are the primary determinants of energy consumption. This highlights that the agent prioritized physically meaningful process indicators.

The dependence plots for the four variables, presented in Fig. 12, provide deeper insight into the feature-value relationships learned by the agent. For the remaining production volume (Fig. 12a), a strong negative correlation was observed (Pearson $r = -0.98$, Spearman $\rho = -1.00$): as the remaining volume decreases toward zero (i.e., as the task approaches completion), SHAP values increase consistently. This relationship validates that the agent correctly internalized the logical principle that task completion yields high expected returns, which is a fundamental requirement for successful operation. Similarly, feed water concentration (Fig. 12c) exhibits a strong negative correlation with SHAP values (Pearson $r = -0.95$ and Spearman $\rho = -0.96$), reflecting the agent's learned understanding that higher feed concentration increases energy demand and thereby reduces overall episode returns. This inverse relationship directly indicates that the energy consumption is proportional to feed water salinity. These findings suggest that the RL agent internalized a physics-aware mechanism that appropriately captures the underlying physical constraints.

3.4. Practical implications, limitations, and future directions

In this study, a framework for training a reinforcement learning agent to autonomously control the CCRO process was developed and evaluated. The combination of a parallelized training architecture with a

Table 3

Summary of mean absolute SHAP values and correlation coefficients for observation features.

Feature	Mean absolute SHAP value	Pearson correlation coefficient (r)	Spearman correlation coefficient (ρ)
Production target volume remaining	6.148	-0.98	-1.00
Production target time remaining	1.425	0.87	0.91
Feed water concentration	0.286	-0.95	-0.96
Membrane module inlet pressure	0.123	-0.98	-0.99
Permeate flow rate	0.086	0.99	0.99
Membrane module outlet pressure	0.070	0.95	0.98
Circulated concentrate concentration	0.042	0.96	0.98
Circulated concentrate flow rate	0.027	-0.18	-0.15
Feed water temperature	0.022	0.84	0.87
Permeate concentration	0.010	0.13	0.10
Discharged concentrate flow rate	0.008	-0.49	-0.12

simulation model calibrated using real-world operational data is of particular practical relevance. This configuration enables efficient exploration of operational uncertainties common in small-scale, decentralized plants, where tight environmental regulations may be infeasible and sustained expert human oversight impractical, while simultaneously accelerating training. Consequently, the proposed framework for training a robust target-driven autonomous agent is a promising alternative to the conventional rule-based operations in such settings.

The observed training dynamics also provided important insights into reward function design. During training, the agent tended to exploit the production time limit, resulting in performance fluctuations in later stages as it learned to operate near the time constraint to maximize energy and water efficiency, necessitating a post-hoc statistical identification of the best checkpoint. This highlights the importance of carefully engineered reward functions (Dewey, 2014). For example, a reward formulation that gradually increases the penalty as the time limit approaches could guide the agent toward a more stable policy optimization, as the agent would receive incremental penalty signals before reaching the hard constraint boundary. Alternatively, off-policy RL algorithms such as Deep Q-Network or Soft Actor Critic, which maintain replay buffers of past experience, could mitigate the effect of intermittent penalties by retaining earlier penalty transitions in training data (Haarnoja et al., 2018; Mnih et al., 2015). However, on-policy algorithms, such as PPO used in this study, offer advantages in stable policy optimization, representing a meaningful trade-off. Therefore, such reward engineering and algorithm selection tailored to a specific operational context are expected to be crucial steps in future real-world deployment.

Several modeling limitations should also be considered. The numerical simulation, although validated against plant data, assumed an idealized low-level control of the flow and recovery setpoints. Incorporating more realistic models of industrial controllers, such as software-defined PID controllers or model predictive controllers, would enhance the simulation fidelity and enable the investigation of hierarchical control structures, in which a high-level RL agent must cope with noisy, delayed responses from low-level actuators.

The transferability of the proposed framework to other CCRO configurations, in terms of the simulation model and RL training framework, warrants discussion. Adapting the simulation model proposed in this study to a different CCRO system requires re-calibration with operational data from the target plant, and update of the process-specific parameters such as process geometry and feed water property accordingly. Transfer of the RL framework to a new system involves re-selecting or adjusting observation variables to match available sensor configurations of the target process, and adjusting reward function to align with the specific operational objectives. The proposed approach can be systematically adapted to diverse CCRO configurations through the steps described above. Nevertheless, as the present study was conducted using a single groundwater plant, application on other configurations, such as seawater desalination, remains an important direction for future work.

The concept of a target-driven local agent has significant potential for broader hierarchical management systems. A future scenario in which a high-level controller, such as a large language model (LLM)-based system, processes regional data and events, including the farming season, to predict water demand and issue production targets, may be feasible. In such a scenario, an autonomous agent like the one developed in this study, would be especially effective.

4. Conclusions

This study integrated a data-calibrated dynamic CCRO simulator with an RL control framework and quantitatively evaluated the integrated system for plant-realistic tasks under variable operating conditions. The major findings are summarized as follows:

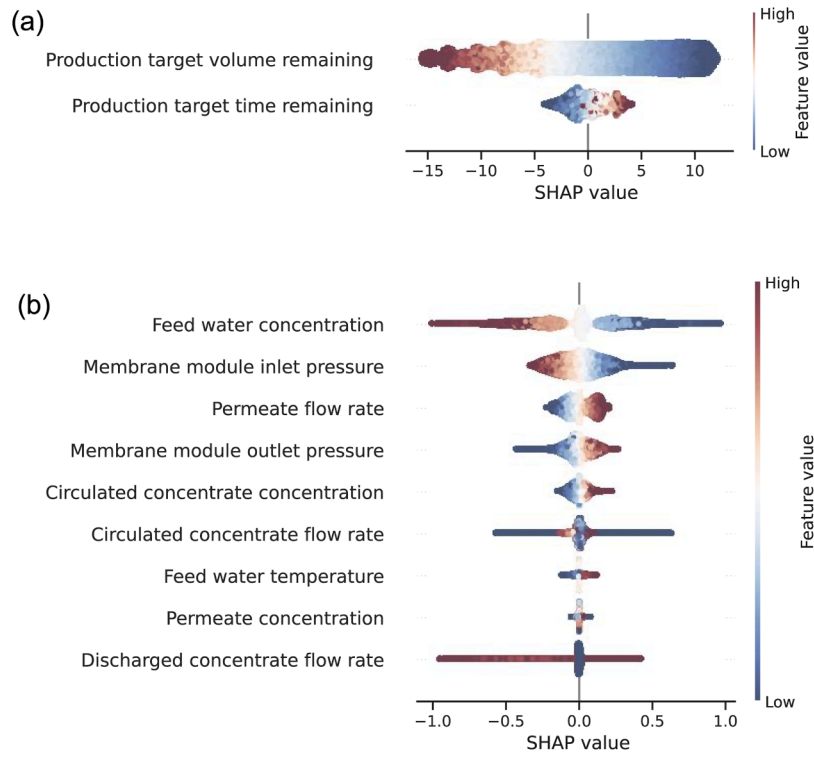


Fig. 11. Summary plots of SHAP analysis on the critic network of the best-performing RL agent on (a) production target-related features and (b) process-related features.

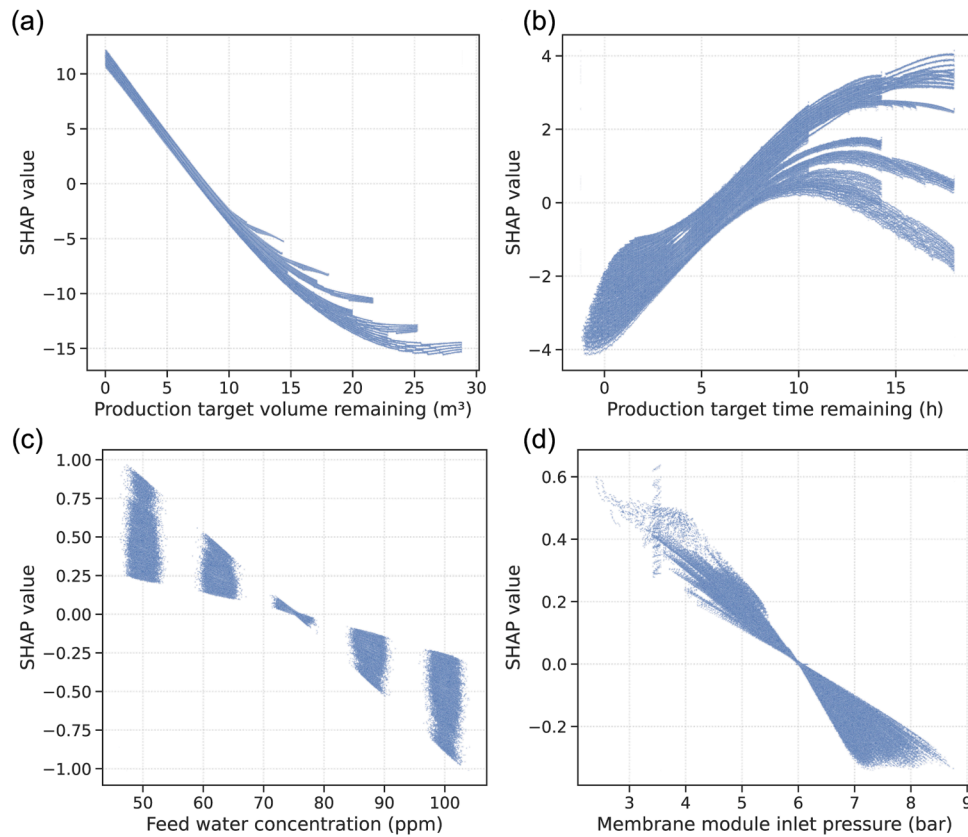


Fig. 12. Dependence plots for the four most influential features: (a) production target volume remaining, (b) production target time remaining, (c) feed water concentration and (d) membrane module inlet pressure.

- The simulator reproduced key process variables, including the permeate flow rate, circulated concentrate flow rate, membrane inlet pressure, and permeate concentration, with low RMSE values, indicating that the dynamic model adequately captured the underlying plant behavior.
- The RL environments with a hybrid action space comprising mixed feed flow rate, single-pass recovery ratio, and operation mode were configured across 24 environmental settings. Based on training on more than 10 million environmental steps and evaluating 375 deterministic scenarios, Checkpoint 18 was identified as the best checkpoint among those saved during training using statistical tests, including Shapiro–Wilk for normality, a Friedman omnibus test, and Conover–Iman post-hoc comparisons.
- Checkpoint 18 demonstrated adaptive policies across feed concentration conditions, which achieved a mean SEC of 0.489 kWh/m³ with a mean recovery ratio of 0.955 over the evaluation set. The resulting agent consistently outperformed the static rule-based controller, achieving an average SEC reduction of 13.14% and a 3.92% higher recovery ratio across all evaluation scenarios.

These results demonstrate that a data-calibrated CCRO simulation, combined with RL control, can achieve high recovery while maintaining stable water quality and reducing energy demand under variable operating conditions. Thus, the proposed framework for training a robust target-driven autonomous agent is a promising alternative to conventional rule-based operations in small-scale decentralized water treatment plants, where continuous expert oversight is impractical. More broadly, the proposed simulation-to-control framework is transferable within CCRO by updating membrane and feed-scenario specifications in the simulator, and it can be extended to other high-recovery membrane processes such as forward osmosis or membrane distillation when an appropriate dynamic process model is available. Future work should incorporate more realistic controller models by moving beyond the idealized low-level control of flow and recovery setpoints and instead representing industrial control behavior, such as software-defined PID or model predictive control, together with advanced RL algorithms and hierarchical control structures that account for noisy and delayed actuator responses. In addition, high-level supervisory controllers, such as LLM-based systems that translate regional information into production targets for local RL-based CCRO agents, could enhance adaptability and facilitate practical implementation in decentralized water treatment networks.

CRedit authorship contribution statement

Jeongwoo Moon: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Byeongchan Yun:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Kiho Park:** Writing – review & editing, Validation, Methodology, Investigation. **Seong-Su Kim:** Resources, Investigation, Data curation. **Youngjoo Lee:** Resources, Investigation, Data curation. **Kwanho Jeong:** Writing – review & editing, Validation, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Kyung Hwa Cho:** Writing – review & editing, Validation, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

The authors declare the following potential competing interests: Prof. Kyung Hwa Cho serves as a Guest Editor for this special issue. To ensure the integrity of the review process, Prof. Cho will be completely excluded from the peer-review handling and editorial decision-making

for this manuscript. All other authors declare no competing interests. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2024-00406500) and the NRF grant funded by the Korean government (MSIT) (No. RS-2023-00252518).

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.watres.2026.125855.

Data availability

Data will be made available on request.

References

- Bonny, T., Kashkash, M., Ahmed, F., 2022. An efficient deep reinforcement machine learning-based control reverse osmosis system for water desalination. *Desalination* 522, 115443.
- Conover, W.J., Iman, R.L., 1981. Rank transformations as a bridge between parametric and nonparametric statistics. *Am. Stat.* 35 (3), 124–129.
- Croll, H.C., Ikuma, K., Ong, S.K., Sarkar, S., 2024. Unified control of diverse actions in a wastewater treatment activated sludge system using reinforcement learning for multi-objective optimization. *Water Res.* 263, 122179.
- Dewey, D., 2014. Reinforcement learning and the reward engineering principle.
- Dormand, J.R., Prince, P.J., 1980. A family of embedded Runge-Kutta formulae. *J. Comput. Appl. Math.* 6 (1), 19–26.
- e Silva, P.P.G., Nobre, C.R., Resaffe, P., Pereira, C.D.S., Gusmão, F., 2016. Leachate from microplastics impairs larval development in brown mussels. *Water Res.* 106, 364–370.
- Elbassoussi, M.H., Kaoud, O.G., Zubair, S.M., 2025. A dimensionless approach to minimizing energy consumption in a single-pass reverse osmosis desalination system. *Water Res.*, 123885.
- Elimelech, M., Phillip, W.A., 2011. The future of seawater desalination: energy, technology, and the environment. *Science* 333 (6043), 712–717.
- Fisher, R.A., 1970. Breakthroughs in Statistics: methodology and Distribution. Springer, pp. 66–70.
- Friedman, M., 1937. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *J. Am. Stat. Assoc.* 32 (200), 675–701.
- Gu, H., Plumlee, M.H., Boyd, M., Hwang, M., Lozier, J.C., 2021. Operational optimization of closed-circuit reverse osmosis (CCRO) pilot to recover concentrate at an advanced water purification facility for potable reuse. *Desalination* 518.
- Haarnoja, T., Zhou, A., Abbeel, P., Levine, S., 2018. Soft Actor-Critic: off-policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. *Pmlr*, pp. 1861–1870.
- Jeong, K., Son, M., Yoon, N., Park, S., Shim, J., Kim, J., Lim, J.-L., Cho, K.H., 2021. Modeling and evaluating performance of full-scale reverse osmosis system in industrial water treatment plant. *Desalination* 518.
- Kim, J., Park, K., Yang, D.R., Hong, S., 2019. A comprehensive review of energy consumption of seawater reverse osmosis desalination plants. *Appl. Energy* 254, 113652.
- Li, M., 2021. A spatiotemporal model for dynamic RO simulations. *Desalination* 516, 115229.
- Li, M., 2023a. Cyclic simulation and energy assessment of closed-circuit RO (CCRO) of brackish water. *Desalination* 545.
- Li, M., 2023b. An improved closed-circuit RO (CCRO) system: design and cyclic simulation. *Desalination* 554, 116519.
- Lundberg, S.M., Lee, S.-L., 2017. A unified approach to interpreting model predictions. *Advanc. Neural. Informat. Process. Syst.* 30.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D., 2015. Human-level control through deep reinforcement learning. *Nature* 518 (7540), 529–533.
- Mo, Z., She, Q., 2025. Analysing practical inefficiencies of closed-circuit reverse osmosis (CCRO) seawater desalination via finite-difference time-dependent model. *Desalination*, 119406.
- Moon, J., Jeong, K., Chae, S.H., Shim, J., Kim, J., Cho, K.H., Park, K., 2025a. Robust deep learning model combined with missing input data estimation: application in a 1000 m³/day high-salinity SWRO plant. *Desalination* 603, 118678.
- Moon, J., Kim, J.H., Park, K., 2024. Cost-based optimization, feasibility study, and sensitivity analysis of forward osmosis/crystallization/reverse osmosis with high-

- temperature operation for high-salinity seawater desalination. *Desalination* 580, 117531.
- Moon, J., Son, S., Kim, J., Park, K., 2025b. Critical challenges in high-salinity seawater reverse osmosis systems: technical, energy, and environmental reviews. *Desalination*, 118811.
- Park, K., Davies, P.A., 2021. A compact hybrid batch/semi-batch reverse osmosis (HBSRO) system for high-recovery, low-energy desalination. *Desalination* 504.
- Park, S., Shim, J., Yoon, N., Lee, S., Kwak, D., Lee, S., Kim, Y.M., Son, M., Cho, K.H., 2022. Deep reinforcement learning in an ultrafiltration system: optimizing operating pressure and chemical cleaning conditions. *Chemosphere* 308, 136364.
- Piadeh, F., Behzadian, K., Chen, A.S., Kapelan, Z., Rizzuto, J.P., Campos, L.C., 2023. Enhancing urban flood forecasting in drainage systems using dynamic ensemble-based data mining. *Water Res.* 247, 120791.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. *arXiv preprint*.
- Shaphiro, S., Wilk, M., 1965. An analysis of variance test for normality. *Biometrika* 52 (3), 591–611.
- Shim, J., Lee, S., Park, S., Moon, J., Lee, C., Cho, K.H., 2025. Optimizing membrane cleaning strategy of industrial reverse osmosis process using long sequence time-series forecasting. *Desalination*, 119193.
- Shim, J., Lee, S., Yun, N., Son, M., Chae, S.H., Cho, K.H., 2024. Autonomous real-time control for membrane capacitive deionization. *Water Res.* 262, 122086.
- Song, Y.-P., Wang, W.-Z., Wang, Y.-Q., Yin, W.-X., Chen, J.-J., Xu, H.-R., Cheng, H.-Y., Ma, F., Wang, H.-T., Wang, A.-J., 2025. Data-driven differentiable model for dynamic prediction and control in wastewater treatment. *Water Res.*, 123772.
- Sutariya, B., Raval, H., 2021. Analytical study of optimum operating conditions in semi-batch closed-circuit reverse osmosis (CCRO). *Sep. Purif. Technol.* 264, 118421.
- Sutton, R.S., Barto, A.G., 1998. Reinforcement learning: an introduction. MIT press Cambridge.
- Tayara, A., Shang, C., Zhao, J., Xiang, Y., 2024. Machine learning models for predicting the rejection of organic pollutants by forward osmosis and reverse osmosis membranes and unveiling the rejection mechanisms. *Water Res.* 266, 122363.
- Towers, M., Kwiatkowski, A., Terry, J., Balis, J.U., De Cola, G., Deleu, T., Goulão, M., Kallinteris, A., Krimmel, M., KG, A., 2024. Gymnasium: a standard interface for reinforcement learning environments. *arXiv preprint*.
- Tukey, J.W., 1949. Comparing individual means in the analysis of variance. *Biometrics* 99–114.
- Walker, W.S., Kim, Y., Lawler, D.F., 2014. Treatment of model inland brackish groundwater reverse osmosis concentrate with electrodialysis—part I: sensitivity to superficial velocity. *Desalination* 344, 152–162.
- Warsinger, D.M., Tow, E.W., Maswadeh, L.A., Connors, G.B., Swaminathan, J., 2018. Inorganic fouling mitigation by salinity cycling in batch reverse osmosis. *Water Res.* 137, 384–394.
- Warsinger, D.M., Tow, E.W., Nayar, K.G., Maswadeh, L.A., 2016. Energy efficiency of batch and semi-batch (CCRO) reverse osmosis desalination. *Water Res.* 106, 272–282.
- Wilcoxon, F., 1945. Individual comparisons by ranking methods. *Biometrics. Bullet.* 1 (6), 80–83.
- Yoon, N., Park, S., Son, M., Cho, K.H., 2022. Automation of membrane capacitive deionization process using reinforcement learning. *Water Res.* 227, 119337.
- Yun, B., Shim, J., Moon, J., Lee, S., Jeong, K., Cho, K.H., 2025. Towards autonomous operation of two-stage reverse osmosis water treatment system with multi-agent reinforcement learning. *Desalination* 609, 118870.