Automated Reward Shaping via Meta-Reinforcement Learning for Autonomous Pressurizer Level Control in Nuclear Power Plants

Yoo Joon Seoung[a], Seung Jun Lee[b]

[a] Ulsan National Institute of Science and Technology, Ulsan, South Korea, yjs0427s@unist.ac.kr

[b] Ulsan National Institute of Science and Technology, Ulsan, South Korea, sjlee420@unist.ac.kr

Abstract: This paper presents an automated reward shaping framework for autonomous pressurizer level control in nuclear power plants (NPPs). Designing effective reward functions for reinforcement learning (RL)-based NPP controllers requires substantial domain expertise and iterative trial-and-error. To address this, we propose a dual-loop meta-reinforcement learning (Meta-RL) architecture in which an outer meta-policy automatically optimizes the parameters of a potential-based reward shaping (PBRS) function, while an inner Soft Actor-Critic (SAC) agent learns the continuous control policy. The PBRS potential is represented as a 5-point piecewise linear function of the controlled variable, parameterized by seven learnable scalars. The meta-policy is trained using reward-weighted regression, a stable alternative to REINFORCE that avoids score function variance issues. Experiments are conducted on the KAERI Compact Nuclear Simulator (CNS) with four objective weight configurations (1:0, 1:10, 1:20, 1:30 for tracking vs. undershoot penalty) applied to both upward and downward level transitions to a target of 50%. Results demonstrate that the meta-learned shaping consistently accelerates convergence and, in certain configurations, simultaneously reduces both tracking error and safety violations. Notably, the 1:10 configuration for the downward task achieves the lowest final mean squared error (MSE = 266.8) alongside near-zero undershoot (0.011), outperforming the MSE-only baseline on both metrics. Analysis of converged potential shapes reveals interpretable lead-compensation strategies learned by the meta-policy, wherein the shaping peak is placed asymmetrically relative to the true target to account for system dynamics.

1. INTRODUCTION

Autonomous operation of nuclear power plants (NPPs) has attracted increasing interest as a means to reduce operator workload, improve consistency, and enhance safety margins. The pressurizer, which regulates primary coolant pressure and water level, is among the most operationally sensitive components: precise level control is required to maintain pressure setpoints while avoiding safety system actuation. Conventional proportional-integral-derivative (PID) controllers perform adequately at steady state but are limited in handling transients and multi-variable interactions [1].

Reinforcement learning (RL) has emerged as a promising alternative for continuous NPP control tasks. RL agents learn control policies through interaction with a simulator without requiring explicit plant models. Soft Actor-Critic (SAC) [2], an entropy-regularized off-policy algorithm, has demonstrated particular effectiveness in continuous action spaces with stable training properties. Recent studies have applied SAC and related deep RL methods to NPP control, including pressure regulation, coolant temperature management, and multi-variable operations [3, 4].

A central challenge in applying RL to NPP control is reward function design. The reward function encodes the control objective and safety constraints, and its shape critically determines both the learning speed and the final policy quality. Hand-crafted reward functions require extensive domain knowledge and often require iterative revision when the control objective changes. Moreover, the relative weighting between tracking accuracy and safety violation penalties introduces additional degrees of freedom that are difficult to set a priori.

Potential-based reward shaping (PBRS) [7] provides a theoretically grounded approach to accelerating RL learning without altering the optimal policy of the underlying Markov decision process (MDP). By augmenting the environment reward with a potential function difference $\gamma\Phi(s') - \Phi(s)$, shaping guides the agent toward promising regions of the state space. However, designing the potential function itself still requires human expertise.

To overcome this limitation, we propose automating the design of the PBRS potential function through a meta-reinforcement learning (Meta-RL) outer loop. Inspired by approaches such as Hindsight Experience Replay [5] and recent meta-learning frameworks for NPP applications [6], our framework trains a meta-policy that observes recent episode performance statistics and outputs the parameters of a

5-point piecewise linear potential function. The meta-policy is updated via reward-weighted regression, providing stable gradient estimates.

This paper focuses on the pressurizer level control task and makes the following contributions: (i) a dual-loop Meta-RL framework for automated PBRS parameter optimization; (ii) a 5-point piecewise potential function tailored for single-variable process control; (iii) empirical evaluation across four objective weight configurations and two transition directions on the KAERI CNS; and (iv) interpretable analysis of converged potential shapes.

2. METHODOLOGY

2.1 Soft Actor-Critic

SAC [2] is an off-policy, maximum-entropy RL algorithm that optimizes a stochastic policy π to maximize the expected return augmented by an entropy bonus:

$$J(\pi) = E[\Sigma_t \; r_t \; + \; \alpha \, H(\pi(\cdot|s_t))] \quad (1)$$

where α is the temperature parameter that balances exploration and exploitation, and $H(\cdot)$ is the Shannon entropy. The actor network outputs a Gaussian policy, and a single critic with a target network (soft update $\tau = 0.001$) approximates the state-action value function. All networks use three hidden layers of 256 units with ReLU activations and L2 regularization.

2.2 Potential-Based Reward Shaping

PBRS [7] augments the task reward r with a shaping term derived from a potential function $\Phi: S \rightarrow \mathbb{R}$:

$$r'(s, a, s') = r(s, a, s') + [\gamma\Phi(s'; \theta) - \Phi(s; \theta)] \, / \, \beta\_scale \quad (2)$$

Under this formulation, the optimal policy of the shaped MDP is identical to that of the original MDP, guaranteeing policy invariance [7]. The shaping term is applied online during SAC critic updates; the environment returns only the task reward, which is stored in the replay buffer. This separation ensures that the base reward signal is not contaminated by shaping.

The potential function $\Phi(z; \theta)$ is a 5-point piecewise linear function of the controlled variable z (pressurizer level, %) parameterized by $\theta = [x_1, x_2, x\_target, x_3, x_4, y_2, y_3]$. The peak value $y\_top = 1000$ is fixed, creating an inverted-V shape centered at x_target. The ordering constraint $x_1 < x_2 < x\_target < x_3 < x_4$ is enforced with a minimum gap of 0.5, and all x-values are bounded within [1, 100] (%). $\beta\_scale = 100$ normalizes the shaping magnitude.

2.3 Meta-Policy and Reward-Weighted Regression

The meta-policy $g_\psi: \mathbb{R}^2 \rightarrow [0,1]^7$ is a feedforward network (Dense 128 → 64 → 7, sigmoid output) that maps a 2-dimensional meta-observation to normalized potential parameters. The meta-observation is:

$$o\_meta = [\tilde{m}\_tracking, \; \tilde{m}\_violation] \quad (3)$$

where $\tilde{m}\_tracking = MSE \, / \, MSE\_ref$ and $\tilde{m}\_violation = Violation \, / \, Violation\_ref$ are normalized performance statistics averaged over the most recent $K = 10$ episodes. The meta-reward is:

$$r\_meta = -(\tilde{w}_1 \cdot \tilde{m}\_tracking \; + \; \tilde{w}_2 \cdot \tilde{m}\_violation \; + \; c\_fail) \quad (4)$$

where $\tilde{w}_1 + \tilde{w}_2 = 1$ are fixed objective weights, and c_fail is a collapse penalty for safety boundary violations. The meta-policy is trained by reward-weighted regression (RWR):

$$L(\psi) = E[\bar{v}\_i \cdot \| g_\psi(o\_meta,i) - \theta\_norm,i \|^2] \quad (5)$$

where $\bar{v}\_i = softmax(advantage\_i)$ are normalized importance weights computed from advantage estimates. RWR avoids the high-variance score function estimator of REINFORCE, yielding stable meta-policy updates.

2.4 Dual-Loop Framework

The overall system operates as a two-timescale loop. The **inner loop** runs the SAC agent on the CNS environment, collecting transitions and updating the actor and critic every episode. The **outer loop** activates every $K = 10$ episodes: it collects episode statistics, computes the meta-observation, queries the meta-policy for new potential parameters θ, and propagates them to idle environment workers. Running episodes receive a delayed credit tag ($theta\_id$) that associates their meta-reward contribution

with the correct θ that generated them, avoiding credit assignment errors during asynchronous parallel training.

3. CASE STUDY: PRESSURIZER LEVEL CONTROL

3.1 Simulation Environment

Experiments are conducted on the KAERI Compact Nuclear Simulator (CNS), a real-time simulation platform for a two-loop pressurized water reactor. The CNS models primary coolant circuit dynamics including pressurizer heater and spray, reactor coolant pumps, and steam generators. Two initial conditions are used: IC19 (high initial level, ≈70%) and IC20 (low initial level, ≈30%), randomly selected at each episode reset to ensure policy generalization. Simulation is advanced at 12-second intervals with an episode time limit of 14,400 seconds (4 hours).

3.2 MDP Formulation

The state vector contains five variables observed from the CNS:

$s = [ZINST63,\ BFV122,\ BHV142,\ (ZINST63 - L^*),\ L^*]$

where ZINST63 is the pressurizer level (%), BFV122 and BHV142 are the positions of two coolant makeup valves (0–1), $(ZINST63 - L^*)$ is the level deviation, and $L^* = 50\%$ is the fixed target. The action vector is $a = [a\_BFV122,\ a\_BHV142] \in [-1, 1]^2$, commanding incremental valve position targets converted to pulse signals. An auxiliary spray controller independently maintains pressure within [22, 28] bar during level transitions.

The task reward (returned by the environment without shaping) is:

$$r\_task = [+10 \text{ if } |L - L^*| < 1\%] - w\_mse \cdot (L - L^*)^2 - w\_viol \cdot V(L) \quad (6)$$

where $V(L) = max(0, L^* - L)^2$ is the squared undershoot penalty (level falling below target). The episode terminates on level collapse (< 17% or > 99%), PT-curve violation, or timeout.

3.3 Experimental Configuration

Four objective weight ratios $w_1 : w_2$ are evaluated, where $\tilde{w}_1 = 1/(1+x)$ and $\tilde{w}_2 = x/(1+x)$, with $x \in \{0, 10, 20, 30\}$. These are applied independently to both **down** (level descending from ≈70% to 50%) and **up** (level ascending from ≈30% to 50%) transition scenarios, yielding eight experiments in total. Each experiment runs for approximately 10,000 episodes using 10 parallel CNS instances. The meta-policy begins training after a 100-episode burn-in period.

4. RESULTS

4.1 Training Performance

Table 1 summarizes the final performance of all eight experiments, computed as the mean of the last 200 episode statistics.

Table 1: Final Training Performance (Mean of Last 200 Episodes)

Experiment	Direction	w_1:w_2	$\tilde{w}_1$	Final MSE	Final Violation	Viol. Type
down 1:0	Down	1:0	1.000	303.5	1.414	Undershoot
down 1:10	Down	1:10	0.091	266.8*	0.011*	Undershoot
down 1:20	Down	1:20	0.048	293.2	0.517	Undershoot
down 1:30	Down	1:30	0.032	270.9	0.188	Undershoot
up 1:0	Up	1:0	1.000	50.7*	4.056	Undershoot
up 1:10	Up	1:10	0.091	67.6	3.655	Undershoot
up 1:20	Up	1:20	0.048	88.4	2.904	Undershoot
up 1:30	Up	1:30	0.032	94.1	1.432*	Overshoot

** Best value in the respective group (Down or Up).*

For the **down** direction, the 1:10 configuration uniquely dominates all others: it achieves the lowest final MSE (266.8 vs. 303.5 for the MSE-only baseline) while simultaneously reducing undershoot to near zero (0.011 vs. 1.414 for 1:0). This finding indicates that the violation penalty, mediated through the Meta-RL shaping mechanism, provides additional learning signal that improves tracking rather than degrading it. Beyond x = 10, further increasing the violation weight degrades MSE (293.2 for x = 20) without proportional improvement in violation suppression.
For the **up** direction, a clear monotonic trade-off exists: increasing *x* reduces violation (from 4.056 at x = 0 to 1.432 at x = 30) at the cost of higher MSE (50.7 to 94.1). No single configuration dominates on both metrics, indicating that the up task presents a genuine multi-objective trade-off. The x = 30 experiment additionally measures overshoot rather than undershoot, reflecting a change in asymmetric penalty direction.

4.2 Converged Reward Shape Analysis

Table 2 presents the final potential parameters θ averaged over the last 20 meta-updates, revealing three systematic patterns in the meta-policy's learned shaping strategy.

Table 2: Converged Potential Shape Parameters (Mean of Last 20 Meta-Updates)

Experiment	x_1	x_2	x_target	x_3	x_4	y_2	y_3	Shape
down 1:0	44.6	48.5	48.98	51.4	99.7	106	104	Narrow sym. peak
down 1:10	1.5	2.2	54.90	99.3	100	106	107	Broad ramp, flat right
down 1:20	1.3	2.0	45.23	99.2	99.7	108	993	Wide, steep right wall
down 1:30	44.3	44.8	54.74	55.3	99.1	991	984	Narrow tall peak
up 1:0	44.8	48.6	54.93	55.4	98.5	110	994	Peak above L*, tall right
up 1:10	44.4	44.9	45.79	54.2	99.6	155	161	Peak below L*
up 1:20	1.4	48.4	48.85	99.2	99.8	109	106	Broad flat ramp
up 1:30	1.3	2.4	54.83	99.4	100	993	990	Wide tall ramp

Three patterns emerge from the converged shapes. **First**, the meta-policy consistently places *x_target* $\neq L^*$ = 50%. For the down task at 1:10, *x_target* = 54.90, placing the shaping peak above the true target. This acts as a lead-compensation: the descending agent overshoots slightly in terms of shaping incentive, inducing it to decelerate precisely at 50%. Conversely, for up 1:10, *x_target* = 45.79 (below L*), pulling the rising agent toward the target from below.
Second, high-violation-weight experiments (x = 30) converge to near-rectangular potential shapes with $y_2, y_3 \approx 990–1000$, creating a uniform high-potential plateau across the entire state space. This maximizes the shaping gradient everywhere, consistent with the strong safety constraint that dominates the meta-reward at x = 30.
Third, asymmetry in y_2 versus y_3 encodes directional preference. For example, up 1:0 has $y_3 = 994 >> y_2 = 110$, creating a steep right-side potential wall that discourages the agent from overshooting above 50%.

5. CONCLUSIONS

This paper presented a dual-loop Meta-RL framework for automated potential-based reward shaping applied to pressurizer level control in a nuclear power plant simulator. A 5-point piecewise potential function parameterized by seven scalars is optimized by a meta-policy trained via reward-weighted regression, enabling objective-aware shaping without manual tuning.

Experiments across four tracking-violation weight configurations and two transition directions demonstrate that: (i) the 1:10 weight configuration for the downward task uniquely achieves simultaneous improvement in both MSE and safety violation, outperforming the MSE-only baseline on both metrics; (ii) the upward task exhibits a genuine MSE-violation trade-off, with x = 30 achieving the lowest violation (1.432) at the cost of higher MSE (94.1); and (iii) the meta-policy learns interpretable shaping strategies including lead-compensation (asymmetric x_target placement) and high-gain plateau shapes at large violation weights.

These results suggest that Meta-RL-based automated reward shaping is a viable approach for NPP control applications where manual reward engineering is time-consuming and objective weights are scenario-dependent. Future work will extend the framework to the aggressive cooldown case study, incorporate multi-task meta-learning across different target setpoints, and investigate policy transfer to higher-fidelity simulators.

Acknowledgements

This work was supported by Korea Institute of Energy Technology Evaluation and Planning(KETEP) grant funded by the Korea government(MOTIE)(RS-2024-00403194, Next-Generation Nuclear Technology Creation IP-R&D Talent (Human Resources) Development Project)

References

[1] M.C. Kim and D.W. Jerng. "Automated startup control of nuclear power plants using model predictive control," Ann. Nucl. Energy, vol. 68, pp. 89–95, 2014.

[2] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," Proc. ICML, PMLR, pp. 1861–1870, 2018.

[3] D. Lee et al. "Reinforcement learning for nuclear power plant control," Energies, vol. 15, p. 2834, 2022.

[4] J. Bae et al. "Autonomous control of pressurized water reactor using deep reinforcement learning," Nucl. Eng. Technol., vol. 55, pp. 3277–3290, 2023.

[5] M. Andrychowicz et al. "Hindsight experience replay," Adv. Neural Inf. Process. Syst., vol. 30, 2017.

[6] K.H.N. Nguyen et al. "Meta-reinforcement learning for nuclear plant control," J. Nucl. Eng., vol. 5, pp. 209–225, 2024.

[7] A.Y. Ng, D. Harada, and S. Russell. "Policy invariance under reward transformations: Theory and application to reward shaping," Proc. ICML, pp. 278–287, 1999.