Nuclear Engineering and Technology 55 (2023) 839–849

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original Article

Strategy to coordinate actions through a plant parameter prediction model during startup operation of a nuclear power plant

Jae Min Kim, Junyong Bae, Seung Jun Lee*

Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan, 44919, Republic of Korea

ARTICLE INFO

Article history:
Received 27 June 2022
Received in revised form
13 October 2022
Accepted 18 November 2022
Available online 22 November 2022

Keywords:
Autonomous operation
Reinforcement learning
Soft actor-critic
Long short-term memory
Parameter prediction
Nuclear power plants

ABSTRACT

The development of automation technology to reduce human error by minimizing human intervention is accelerating with artificial intelligence and big data processing technology, even in the nuclear field. Among nuclear power plant operation modes, the startup and shutdown operations are still performed manually and thus have the potential for human error. As part of the development of an autonomous operation system for startup operation, this paper proposes an action coordinating strategy to obtain the optimal actions. The lower level of the system consists of operating blocks that are created by analyzing the operation tasks to achieve local goals through soft actor-critic algorithms. However, when multiple agents try to perform conflicting actions, a method is needed to coordinate them, and for this, an action coordination strategy was developed in this work as the upper level of the system. Three quantification methods were compared and evaluated based on the future plant state predicted by plant parameter prediction models using long short-term memory networks. Results confirmed that the optimal action to satisfy the limiting conditions for operation can be selected by coordinating the action sets. It is expected that this methodology can be generalized through future research.

1. Introduction

With the continuing advances in artificial intelligence (AI) and big data processing, the development of automation technology to reduce human error by minimizing human intervention has accelerated. A representative automation system of a nuclear power plant (NPP) is one that automatically trips the reactor to prevent damage to the core in an emergency situation due to an accident. Other systems mainly employ algorithms to maintain specific variables using proportional-integral-derivative controllers [1]. In many advanced reactor proposals, automation technology is considered from the design stage [2–4]. Further research related to automation in NPPs can be summarized as follows: an expert system for instrumentation and control [5], an automated operating procedure system [6], an intelligent reactor core controller [7], and automated diagnosis of NPP states [8,9]. These methodologies help operators make quick decisions in urgent situations. Research cases using AI have dealt with abnormal conditions [10] and autonomous operation with hierarchical architecture for an NPP in an emergency [11]. However, automation research focusing on the operational modes in which electricity is not generated is lacking, where operators manually perform the operation procedures according to the dynamic situation. The operation status of typical pressurized water reactors (PWRs) can be classified based on the reactor power and the temperature of the reactor coolant system (RCS); Fig. 1 shows the typical operation modes along with their current level of automation.

The criteria for discussing the level of automation in startup and shutdown operations are complex, considering the operation strategies and the ability to respond to dynamic situations. Even if some systems do not require operators to initiate them, such partial automation functions are seen in terms of assisting the operators, and thus it is reasonable to view the overall operational flow as being manual. For example, the setpoint of the signals should be adjusted by operators manually, because during startup and shutdown operations, the variables are low compared to full-power operation. Most of the automatic systems are prepared for power-generating operation conditions, where an NPP is at the highest risk. Therefore, system adjustments according to the situation are performed by the operators even for automatic systems. Referring to a report by the Operational Performance Information System for Nuclear Power Plant, 18% of the cases of unintended shutdowns

* Corresponding author.
E-mail address: sjlee420@unist.ac.kr (S.J. Lee).

https://doi.org/10.1016/j.net.2022.11.012

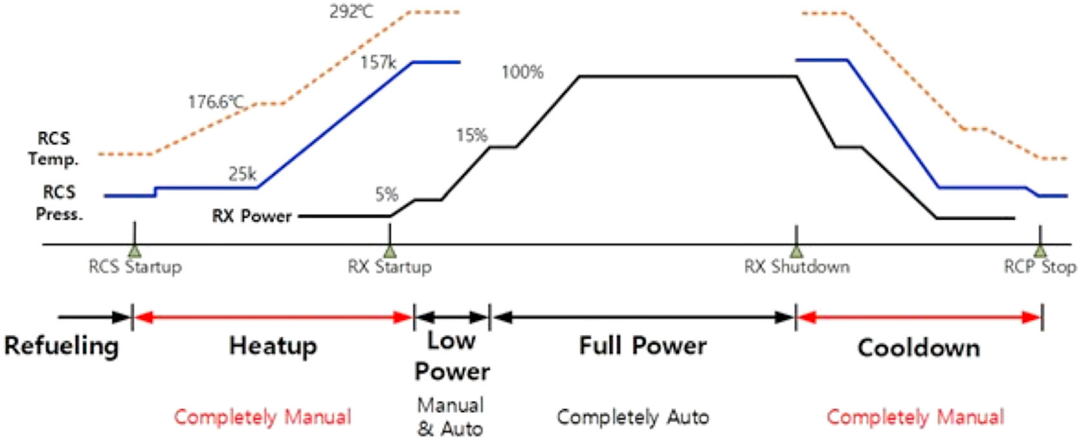

Fig. 1. Automation status according to the operation mode of a typical PWR [12].

during heatup and cooldown operations of domestic Korean NPPs were caused by human error [13]. In this light, autonomous operation performed by AI agents that have obtained operating policies to achieve desired goals could help reduce human error. Studies have shown that NPP operation policies can be learned through reinforcement learning (RL) [14]. In our previous work, we introduced a framework for the development of an autonomous operation system for startup and shutdown operation as shown in Fig. 2 [15].

Operating blocks of the system contain AI agents that execute the startup operation tasks. These blocks are activated when their ancillary condition is satisfied; however, as multiple agents can control the same components, and as individual agents do not consider the overall process or the operating policies of other agents, there may be cases of conflict between the local goals of such agents. This is because each action is selected through the success path for the convergence of RL in the training stage; in other words, the suboptimal actions are not considered by the agents to the extent that the best actions are. But clearly, in a complex multi-agent environment, the influence between agents cannot be ignored, and thus an evaluation method is needed aside from the RL training.

This paper proposes an action coordinating strategy for the autonomous operation system to handle multiple agents. The strategy reflects domain knowledge to align the agents' actions by predicting the future plant state and quantifying it. As a result, it can guide operation that satisfies the limiting conditions for operation (LCOs). To assign values to the actions for comparison, plant parameter prediction models were trained to predict the future outcomes according to the selected actions.

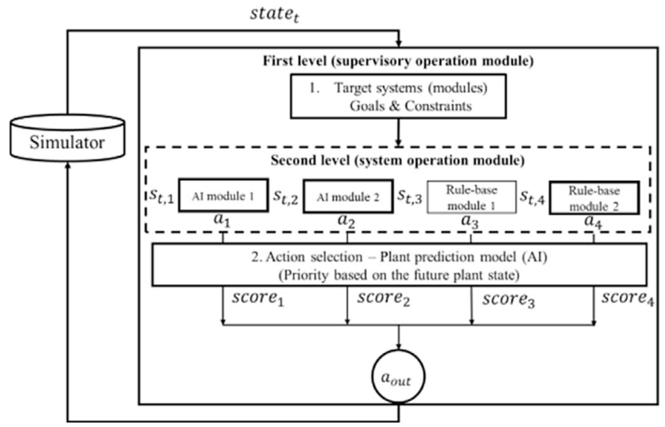
$state_t$
First level (supervisory operation module)
1. Target systems (modules) Goals & Constraints
Simulator
Second level (system operation module)
$s_{t,1}$
AI module 1
$s_{t,2}$
AI module 2
$s_{t,3}$
Rule-base module 1
$s_{t,4}$
Rule-base module 2
a_1
a_2
a_3
a_4
2. Action selection – Plant prediction model (AI)
(Priority based on the future plant state)
$score_1$
$score_2$
$score_3$
$score_4$
a_{out}

Fig. 2. Framework for the autonomous operation system indicating supervisory and system operation modules for startup operation [15].

The information obtained from the prediction models allows one to know the consequences of actions at a particular moment. Each action is assigned a score by quantifying this future information, and by comparing the scores, we can choose the action that will guide the future states in the desired direction. In this paper, three quantification methods are compared: the first simply compares the last future values, the second takes the average values of variables at 1 min intervals, and the third calculates the area between the target value and the future states. The parameter prediction process is performed in cases with a conflict between the actions of different agents; in such cases, the future state related to reflecting the LCOs quantifies how the main variables change through a regression model to select the optimal action. The highest performance among the quantification methods is assessed by ranking them based on how long target parameter stays within the recommended range that satisfy the LCOs.

As an application, soft actor-critic (SAC) algorithms are implemented for the agents to obtain operating policies, and plant parameter prediction models are developed based on long short-term memory (LSTM) networks [16,17]. Through application to startup operation, the quantification methods were compared to best reflect the LCOs leading to the desired result. It is found that optimal operation is possible when the conflicting actions of the autonomous operation system are handled compared to when no processing is performed.

The remainder of this paper is organized as follows. Section II introduces the framework of the autonomous operation system. Section III covers the related methodologies including AI techniques and evaluation strategies. Section IV covers the application and the results. Finally, Section V provides the conclusions.

2. Framework of autonomous operation system for startup operation

Startup operation is carried out according to general operating procedures (GOPs) covering the normal condition of NPPs. In this study, the first step in the development of the autonomous operation system is to introduce the concept of an operating block that groups operation tasks through GOP analysis. In the case of a PWR, systems must be maintained in a high-temperature and high-pressure state for power generation, and the number of components and systems handled for this purpose is wide and diverse. Such a large number of variables covered in the GOP leads to a large number of features for AI training. As the number of data features increases, the dimension of the data becomes too large to train all tasks with a single agent.

In addition, to satisfy the high safety standards of NPPs, it is necessary to reflect expert knowledge rather than completely relying on AI. If many goals are added in the training phase, too many constraints are defined, which may result in lowering the training efficiency. If all states and components are configured as the training environment, the optimal policy is difficult to converge. Therefore, it is advantageous to configure the environment with simplified information to acquire the optimal policy, which leads to a multi-agent environment where various agents intervene. Based on the concept of grouping the GOP tasks into operating blocks for efficient learning, an autonomous operation framework was previously designed [15]. Fig. 3 is drawn with an emphasis on coordinating actions in the previously proposed framework.

The autonomous system consists of two levels, namely supervisory and system operating modules. The supervisory operating module manages the overall operational process by considering the rise and fall of RCS temperature and pressure, which are the main variables. The system operating module includes operating blocks that perform small operation tasks. The operating blocks are activated to achieve their respective operation goals according to each's entry condition. Simple operation tasks, such as changing the state of a component when a certain condition is reached, can be implemented with a rule-based algorithm in the operating block. For example, when the pressure of the steam generator and the temperature of the RCS reach a certain condition during startup operation, the residual heat removal system is disconnected (or isolated) from the RCS, after which the pressure increases along with the temperature beyond previous limits. In this case, the isolation operation is performed safely without complex control. However, other than such rule-based operating blocks that perform simple tasks, complex tasks are performed by operating blocks with an AI agent trained through RL.

According to this concept, the operating blocks are activated in parallel. As shown in Fig. 4, if there is no conflict between actions, the actions are carried out as is. Here, action candidates are a group of actions held for some period of time to decide whether they should be sent to the simulator. If there is no conflict between actions, the action candidates are undefined because the actions desired by the agent are passed directly to the simulator. However, there are cases where the components that the agents want to control overlap and require opposite actions. In this case, each AI agent only considers the variables related to its local goals when creating its operating policy. For example, the pressurizer (PZR) pressure control block considers the PZR pressure and the status of the valves it controls, while not considering the PZR water level as an input variable, which is controlled by a separate block. The problem is that the valves for controlling the PZR pressure and water level in certain contexts are shared, which could lead to conflicts between the blocks in terms of opening or closing the valves. Despite the operation blocks being based on procedures, the control valves for the charging and letdown water flowrate affect the PZR pressure since the inside of the PZR is in a saturated state with vapor and liquid after bubble formation. Although they are not the dominant components for PZR pressure control, these valves can be seen as common actions to control the dependent variables of pressure and water level by different operation blocks. Accordingly, if the actions to be performed by the operating blocks are in conflict, it is necessary to coordinate a set of actions to globally optimize the entire operation.

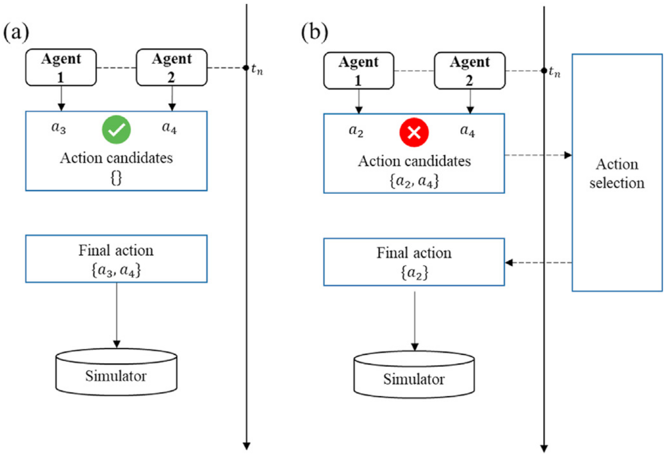

Fig. 4. An example of operation flow (a) without conflict and (b) with conflict.

3. Methodology

This work covers the development of a strategy to coordinate the actions obtained through the operating blocks as a function of

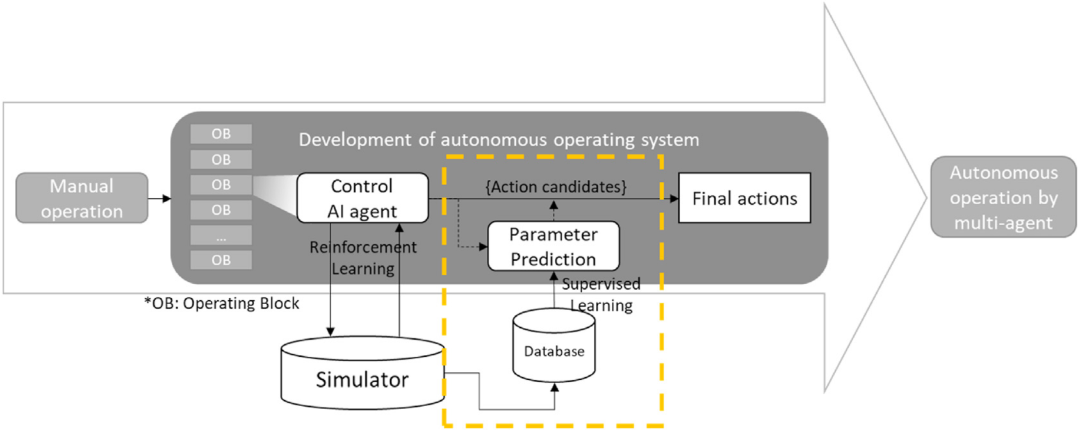

Fig. 3. Overall process to develop the autonomous operation system for startup operation. The step for coordination actions is highlighted in yellow box. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

the autonomous operation system. Unlike the rule-based operating blocks, it is important for the AI-based operating blocks to adopt an algorithm to achieve the optimal policies of each block. To replace a human operator, we created an agent using RL and configured the learning environment to allow the agent to learn the optimal operation policy by repeating trial and error on its own. In addition, we developed a solution, or action coordination strategy, to cover situations in which the actions by operating blocks conflict since the operation blocks can be executed at the same time. Plant parameter prediction models were then created. This section describes the SAC algorithm selected for RL, the LSTM network used in the prediction models, and the three quantification strategies.

3.1. Soft actor-critic for operation

Operating blocks with AI agents are implemented by RL. RL is a field of machine learning in which an agent converges to an optimal policy through interaction with the environment [18]. As a preceding study, by dividing the GOP into detailed task groups, a single AI-based operating block succeeded in achieving the desired operation goal [15]. In the current paper, SAC was used as the RL algorithm. SAC combines off-policy updates with an actor-critic method to maximize the expected reward augmented with an action entropy [16]. SAC has an advantageous structure for the convergence of optimal policies even for continuous spaces. In addition, it is possible for agents to learn by grouping advantageous paths for goal achievement due to the off-policy nature of SAC.

From now on, the specific SAC learning environment is explained with specific examples used in this study. An NPP simulator is used because it is not possible to undergo the trial and error necessary to construct the learning environment in an actual NPP. First, we developed a parallel running environment for multiple NPP simulators, similar to Ref. [19], a setting that enables repeated and real-time communication between an agent and an environment, which is necessary for RL. The information on the states and actions of the operating blocks was selected through GOP analysis, and the use of more information than necessary to achieve the operation goals was restricted. Table 1 lists the input states and output actions of two example operating blocks. The listed actions show that the blocks may conflict since they have shared components (i.e., charging flow control valve and letdown control valve).

RL is a process by which an AI agent optimizes policies by interacting with the environment: an AI agent performs an action (a_t) based on a current state (S_t) that maximizes expected rewards [$r_t(s_t, a_t)$], after which it receives a new state (S_{t+1}). In this process, the direction of policy optimization is guided by how the reward function is defined. In this research, we define this function as a success/fail reward and an auxiliary reward. Equations (1) and (2) detail the success/fail and auxiliary rewards, respectively, where x_t is target parameter (e.g., PZR pressure or level), g_x is the target value for x, l_x is the boundary constant for x, and k_1 and k_2 are positive constants. In the case of the PZR pressure control, g_x was randomly sampled from 23–27 kg/cm^2 and l_x was 0.3 kg/cm^2. In the case of the PZR level control, g_x was randomly sampled from 30–70 % and l_x was 1%.

$$r_1 = \begin{cases} k_1, if\ |x_t - g_x| < l_x \\ 0, otherwise \end{cases} \tag{1}$$

$$r_2 = -k_2\,(x_t - g_x)^2 \tag{2}$$

The success/fail reward informs the agent whether the current state is the target one or not. In other words, if the current state is outside the target range, the agent will not receive any feedback on its current action. This sparse feedback can discourage policy optimization. To solve this problem, we adopted the auxiliary reward, which is the negative squared distance from the target value. The scaling factors for the rewards (i.e., k_1 and k_2) were 10 and 0.0004, respectively. The blocks experienced 5000 episodes, and each episode was reset when the block violated the limiting condition of operation or reached the maximum episode length (i.e., 4200 s). The transition $[s_t, a_t, r_t, s_{t+1}]$ buffer size was 1,000,000, and 128 transitions were randomly sampled for each training trial of policy optimization.

The AI agent created in this way outputs the optimal action that can maximize the future reward according to the current state as learned. However, the future state considered at this time does not fully consider other factors. When several agents are involved in the operation at the same time, uncertainty about the future state increases. Therefore, a method is required to observe the effects of the currently performed actions on the future state.

3.2. Long short-term memory for prediction

The object of quantification is based on the future state, not the present state. To predict the future situation, we used LSTM networks to create parameter prediction models for all possible actions. LSTM networks use the concept of a cell state combining the information at each point in time with several linear combinations and then delivering it to the next time step [17]. This allows certain information to be intentionally removed, maintained, or added via forget, input, and output gates.

Regression models can predict how a plant parameter will change in the future based on current actions taken [20]. Among previously tested artificial neural networks, the LSTM network showed the best performance to address the multivariate problem of future parameter trend estimation. Therefore, in the present work, the plant parameter prediction models are regression models that predict future states with the information currently passing through the LSTM networks. The variables to be predicted through

Table 1
Input and output variables for operating blocks using SAC.

	PZR pressure control block	PZR water level control block
State, S_t	PZR pressure Position of charging flow control valve Position of letdown control valve Position of spray flow control valve Target PZR pressure Deviation from the target pressure	PZR level Position of charging flow control valve Position of letdown control valve Target PZR level Deviation from the target level
Action, a_t	Position of charging flow control valve[a] Position of letdown control valve[a] Position of spray flow control valve	Position of charging flow control valve[a] Position of letdown control valve[a]

[a] Shared components.

the regression models are selected by referring to the LCOs. Here, the number of required prediction models corresponds to the number of possible action outcomes at the moment of an action conflict.

The data structure for the prediction model input has a 10 min length per dataset. Random actions are performed for the first 10 s, with the information up to 60 s used as an input value. After that, the future state is predicted up to 9 min at 1-min intervals.

The structure of the prediction model consists of three LSTM layers. Input data for the first layer should be transformed from two-dimensional data with as much information as the number of features for 60 s to one-dimensional data to fit the model structure. The first layer receives an input variable and transmits an output value of the same size to the next layer, which repeats the same process. After going through the last layer, the data shape is restored to equal the number of features.

Fig. 5 shows an example of the model learning process using the specific values covered in the application (Section IV). In this example, *n* datasets with 17 variables are recorded for 6000 s, which are selected considering the LCO. In the original operation records, the last column shows the label information indicating the actions randomly performed for 10 s. The data for each label has a length of 600 s, and in order to prevent the process from starting under the same condition each time, a random action is performed once every 600 s during an episode running for 6000 s. Through this process, 10 operation data for various situations can be collected from one episode. After the data passes through the LSTM layers composed of prediction models, learning is performed in the direction of reducing the loss by comparing the predicted values for 9 time points at 1-min intervals with the true value.

In this work, a total of 9 prediction models were created because there are 9 possible combinations of actions in the application. At any moment, these models provide an advance notice of variable changes resulting from the chosen action. However, there is a need for a quantification method to check whether the operation is being guided in the desired direction through these parameter changes.

3.3. Quantification strategies

To evaluate the actions performed during NPP operations, standards for quantification are needed. For example, one criterion may be the time required to raise the RCS temperature to a desired level. However, if this criterion does not account for potential hazards, such as rapid changes in temperature and pressure destabilizing the RCS fluid and damaging the surrounding structures, it is not proper. Under normal NPP operation without any abnormality, it should be possible to evaluate whether an operation is being performed in the right direction from a long-term perspective, as short-term errors can have ambiguous causal relationships [3]. For example, if we want to increase the water level in a tank by increasing the inlet flow, then the inlet flow control valve should be opened. At this time, the change in the water level in the tank does not appear immediately within 1–2 s (i.e., in the short-term), but rather requires some longer period of time depending on the flow rate. In addition, if the difference between the inlet and outlet flow rates is large, the water level change by manipulation of the valve will require a longer time.

In this aspect, it is insufficient to simply use only RCS temperature and pressure as evaluation criteria. We therefore apply three quantification methods to determine whether the future NPP state according to specific actions complies with the LCOs. In plant GOPs, the LCOs to be satisfied to safely operate the NPPs are described according to the operating situation.

The LCOs described in the GOP can be classified into two types. One includes the conditions in which certain values must not be exceeded, such as heating rate or flow rate limits. As long as the conditions are not violated, all operating strategies are allowed, and thus the score quantified for this LCO type can be binary. The second type includes the conditions in which certain variables must be kept within a certain range during the operation. When training AI agents, the variables that need to be maintained have higher rewards the further they are from the boundary values. But conversely, in the quantification stage, the closer the variables are to the boundary values, the higher the score should be assigned because high scores represent actions that require a quick response with the highest priority.

In order to compare the quantification scores between variables of different scales, each variable is divided discretely and assigned scores corresponding to specific values. In this paper, we compare three quantification methods to determine how and which of the scores assigned to future states will be retrieved. The first method compares only the variables at the last moment (comparison), the second method compares the mean of predicted values at regular time intervals (average), and the third method compares the area between the predicted values and the median at regular time intervals (area).

Fig. 6 depicts a visualized example of the three quantification methods. P and L represent different variables, the x-axis is the timeline, and the y-axis is the numerical value of each variable. The green region represents the recommended range for the particular

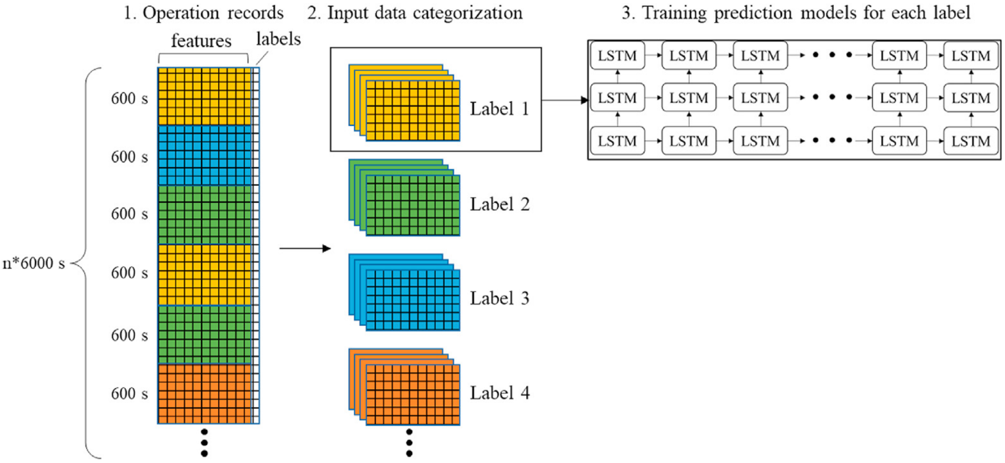

Fig. 5. Learning flow diagram of a plant parameter prediction model that learns by picking one case from the entire dataset.

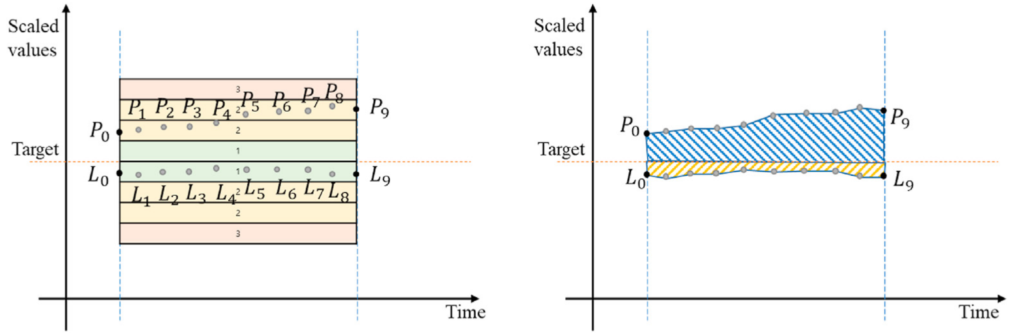
Scaled values
Target
Time
P_0 P_1 P_2 P_3 P_4 P_5 P_6 P_7 P_8 P_9
L_0 L_1 L_2 L_3 L_4 L_5 L_6 L_7 L_8 L_9
3
2
1
1
2
3
Scaled values
Target
Time
P_0 P_9
L_0 L_9

Fig. 6. Visualized example of LCO-based quantification methods. Left: comparison and average methods. Right: area method.

LCO, the yellow region is the acceptable range, and the red region near the boundary conditions stands for LCO violation. Following an operation, the variable trends in terms of these regions at every timestep are monitored.

The comparison method only compares P9 and L9, while the average method takes the average of the predicted moments from 1 to 9 as a score. The area method treats the area of the variable trend as a score based on the yellow region. Based on these methods, when actions for P and L conflict, the actions are compared via scores and the optimal action is selected.

The equations used for this calculation are listed as follows:

$$score_{avg} = \frac{\sum_1^n X_i}{9} \tag{3}$$

$$score_{area} = \sum\nolimits_1^{n-1} \frac{(X_i - target) + (X_{i+1} - target)}{2} \tag{4}$$

$$score_{comp} = X_n \tag{5}$$

Equations (3)–(5) correspond to the three quantification methods based on the information provided by the plant parameter prediction model. X can be any variable considered for the LCO. As shown in Eq. (6), the total score is calculated by adding the scores for pressure and water level, for example, to an index that gives −10 points if the heating rate limit is exceeded when considering future variables, and 0 points otherwise.

$$score_{total} = score_{heatrate} + score_{pressure,k} + score_{level,k} \tag{6}$$

A performance metric of operation with or without quantification methods is expressed as a ratio of the time the relevant variables are maintained in the different regions divided based on the LCO. A successful control can be evaluated as that from which the target variable stayed in the green region for the total operation time. Equation (7) gives the indicators for the performance evaluation used in the results of comparative experiments.

$$X_{\%} = \frac{X_{green}}{X_{red} + X_{yellow} + X_{green}} \tag{7}$$

Another thing to consider when evaluating actions with future information is the extent to which action candidates are to be determined. Fig. 7 shows an example when only actions determined by the agents are counted as candidates, and Fig. 8 shows the decision process when all actions are considered as candidates. In Fig. 8, all prediction models are activated, and by ranking their scores, it is found that neither conflicting action is the optimal action. This implies that there might be new actions that can compensate for the conflicting desired actions from the agents, which should be confirmed through experiments.

4. Application

4.1. Experimental settings

A compact nuclear simulator (CNS) provided the training data and simulation environment. Developed by the Korea Atomic Energy Research Institute (KAERI), the CNS models a 993 MWe three-loop Westinghouse PWR [21]. In the simulation, an action set was activated every 20 s because it is not realistic to control components every second. The application concerns an operation in which the PZR pressure and water level are simultaneously adjusted after a PZR bubble is formed, that is, after the water level has dropped from the full water level.

The agents of the two AI-based operating blocks controlling the PZR pressure and water level in this application were created based on two SAC algorithms. One agent handles the charging water flow control valve and letdown flow control valve for adjusting the water level of the PZR, and the other agent handles the PZR spray control valve in addition to the same valves as the first agent for adjusting the pressure of the PZR. We set up the environment such that an agent gives a valve open or close signal similar to an operator operation. When the signal is activated, it is reset after a certain time step after which the corresponding signal is removed. The change in opening degree during one step is approximately 1.5% for each valve.

Plant parameter prediction models were created using LSTM networks to predict 17 variables, as listed in Table 2. The variables used in the prediction models were based on the LCOs described in the GOPs of the CNS we used, and all variables mentioned in the LCOs were added.

The number of models derives from a combination of two valves to simplify the problem. The charging water flow control valve and letdown flow control valve, respectively tagged FV122 and HV142 in the simulator, have 9 possible combinations: none 00, open 01, and close 10. To distinguish them simply, the combinations are displayed with these digits. Therefore, a total of 9 prediction models were created. For the data generation, random actions were performed every 10 min during 6000 s of operation to allow for different operating situations. This allowed us to obtain data with varying patterns in different situations for each action label. As a result, there are 10,000 datasets with an interval of 600 s. Other hyperparameters were empirically set as follows: the number of cells in each layer is 200, the dropout fraction of the units to drop for the linear transformation of the inputs is 0.1, the number of epochs, or iterations over all datasets, is 100, the Adam optimizer is used with a learning rate of 0.001 to minimize the loss function, and loss is calculated through mean square error [22].

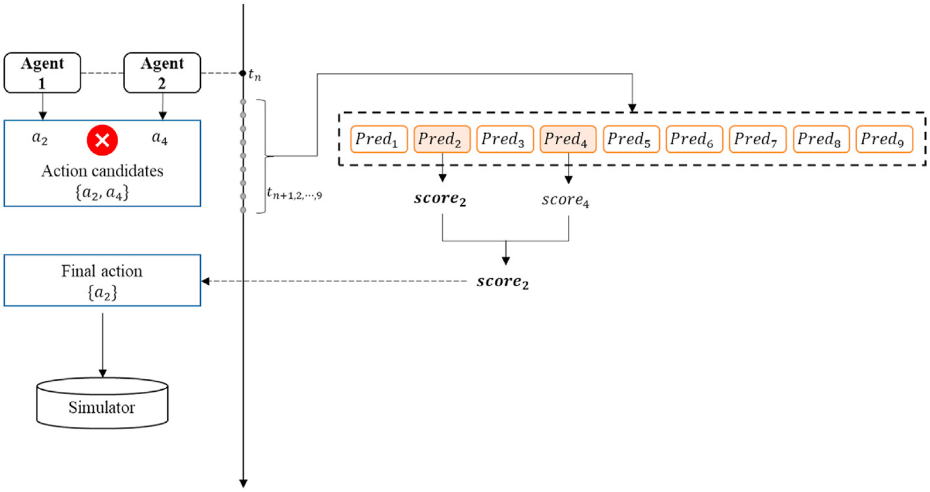

Fig. 7. Example of operation flow considering agents' actions as candidates.

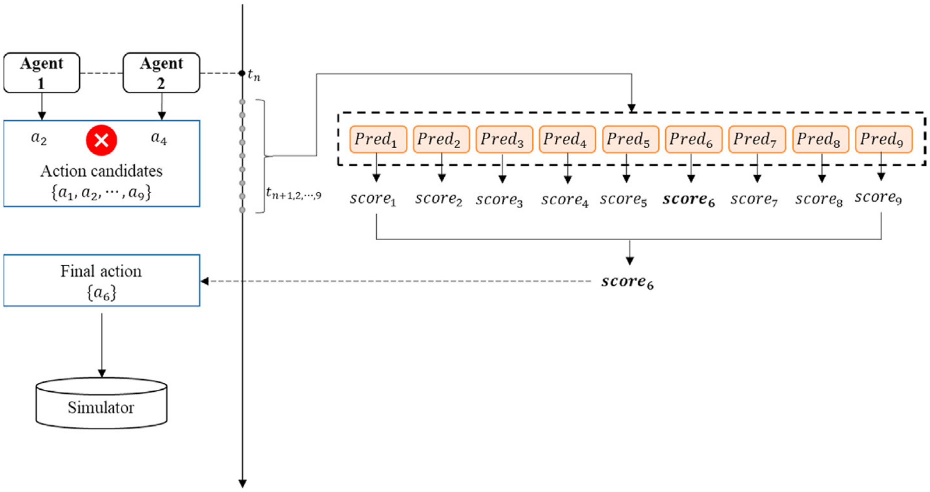
Agent 1
Agent 2
t_n
a_2
a_4
Action candidates
$\{a_1, a_2, \cdots, a_9\}$
$t_{n+1,2,\cdots,9}$
$Pred_1$ $Pred_2$ $Pred_3$ $Pred_4$ $Pred_5$ $Pred_6$ $Pred_7$ $Pred_8$ $Pred_9$
$score_1$ $score_2$ $score_3$ $score_4$ $score_5$ $score_6$ $score_7$ $score_8$ $score_9$
Final action
$\{a_6\}$
$score_6$
Simulator

Fig. 8. Example of operation flow considering all actions as candidates.

Table 2
Input and output variables for the prediction models.

No.	Description
1	PZR TEMPERATURE.
2	LETDOWN BACK PRESSURE
3	LETDOWN FLOW
4	CHARGING FLOW
5	PZR PRESSURE(NARROW RANGE)
6	LOOP 3 AVERAGE TEMP
7	LOOP 2 AVERAGE TEMP
8	LOOP 1 AVERAGE TEMP
9	PZR LEVEL
10	VOLUME CONTROL TANK LEVEL.
11	VCT PRESSURE
12	RCP SEAL INJECTION FLOW
13	RCP SEAL NO.1 DELTA PRESSURE
14	RCP SEAL NO.1 RETURN FLOW
15	S/G 3 PRESSURE
16	S/G 2 PRESSURE
17	S/G 1 PRESSURE

*VCT: Volume control tank, RCP: Reactor coolant pump, S/G: Steam generator.

The plant parameter prediction models predict the future states of the variables up to 9 min according to the 9 action combinations. For example, if the PZR pressure control model selects the action of opening FV122 while the PZR water level control model selects the action of closing FV122, then the actions are ranked by scores based on the model prediction and the optimal action is chosen. Fig. 9 shows an example of the model notation and process to select the final action assuming simple numbers. The first two digits represent signals for controlling FV122, and the remaining numbers represent signals for controlling HV142. In this example, FV122 receives signals in opposite directions by the two agents. The prediction models then calculate the future states to get scores through the coordinating strategy, and ultimately the closing FV122 signal is selected as the optimal action with the higher score. After performing autonomous operation without human intervention through the experiment, the PZR pressure and water level records were classified according to the criteria presented in the LCOs.

The pressure and water level of the PZR, which are the objects of the LCOs, were divided into three regions at regular intervals: recommended, acceptable, and violation. The pressure of the PZR

Table 3
Discrete scoring for PZR pressure and water level.

Level (%)	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40
Pressure (kg/cm^2)	30	29.5	29	28.5	28	27.5	27	26.5	26	25.5	25	24.5	24	23.5	23	22.5	22	21.5	21	20.5	20
Score	10	9	8	7	6	5	4	3	2	1	1	1	2	3	4	5	6	7	8	9	10

Table 4
RMSE for the plant parameter prediction models.

Model (action)	Train_score RMSE	Val_score RMSE
00 00	0.16440	0.16498
00 01	0.17851	0.18993
00 10	0.16424	0.16695
01 00	0.28892	0.28315
01 01	0.12891	0.12905
01 10	0.12082	0.12033
10 00	0.14195	0.14212
10 01	0.17615	0.17994
10 10	0.11743	0.11613

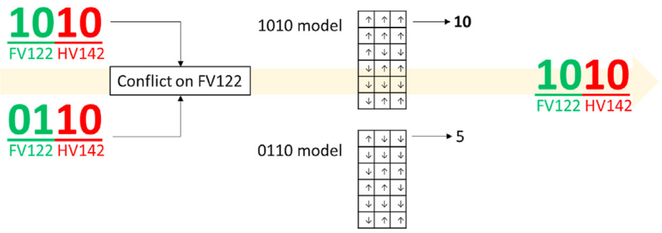

Fig. 9. Example of the model notation and simple process to select the final action when an action conflict exists. The arrows in the tables are simple representations of variable increases or decreases relative to its current value.

ranges from 20 kg/cm^2 to 30 kg/cm^2, and the water level of the PZR ranges from 40% to 60%, with proportionally distributed regions. The scores assigned to specific values of PZR pressure and water level are shown in Table 3.

4.2. Results

When bubbles are formed in the PZR, the water level drops from 100%. In this test, the two agents operate in parallel after a 70% water level is reached without intervening immediately after the water level drops from 100%. The goals of this operation are to keep the PZR pressure between 25 kg/cm^2 and 28 kg/cm^2 and the PZR water level around 50% to comply with the conditions suggested in the LCOs.

Table 4 shows the root mean square error (RMSE) used as an index indicating the learning degree of the prediction models. Since an RMSE score is calculated by expressing the difference between true values and predicted values as a distance, low RMSE values indicate high accuracy of the regression models. According the table, the 9 prediction models have been trained properly. As mentioned in Sect. 4.1, there are 9 prediction models corresponding to the 9 action combinations, actions for FV122 and HV142 in order, and the remaining numbers represent signals for controlling PZR spray valve. Fig. 10 shows that the pressure change is predicted differently depending on the action selected at a specific time. An example of the predictions by the 4th model (01 00), related to the action for opening FV122 only, showing the highest error among the models, can be seen in Fig. 11.

Tables 5 and 6 list the results of comparing the three quantification methods according to evaluation indicators when

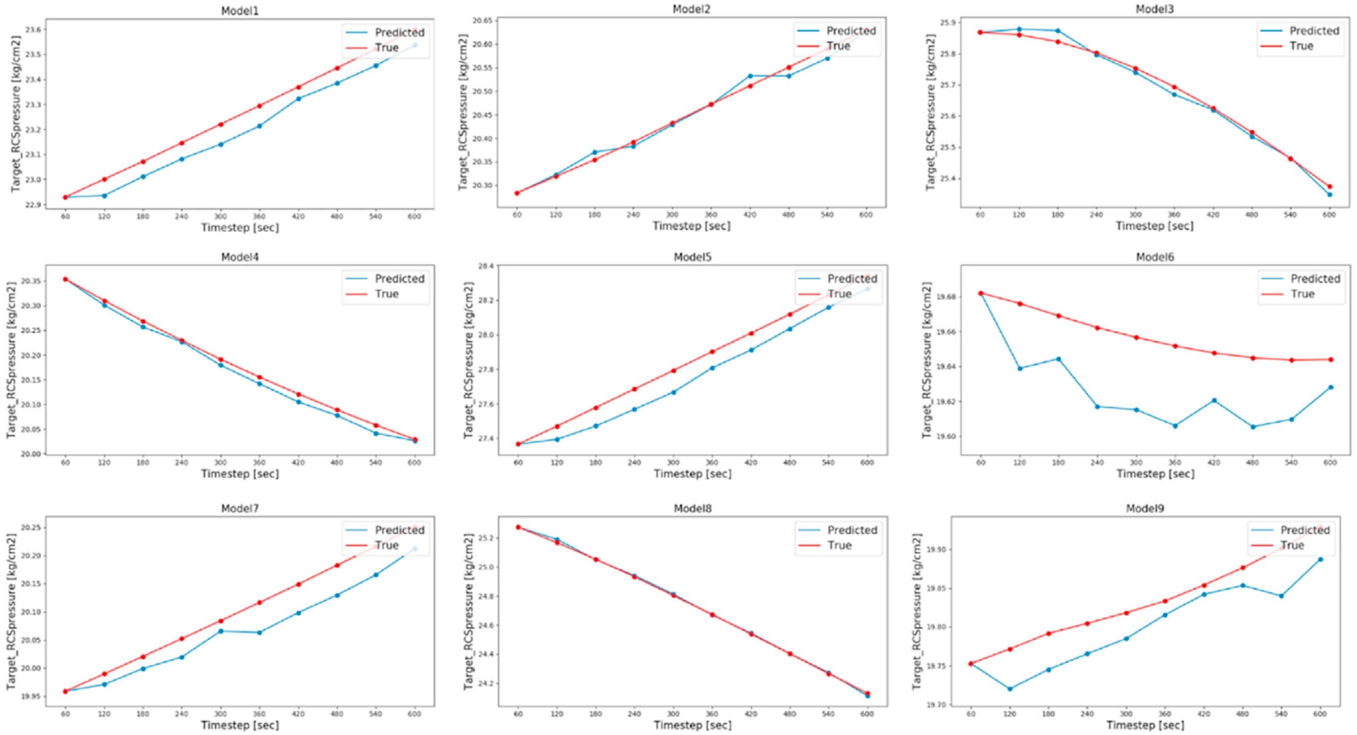

Fig. 10. Examples of future states by the 9 prediction models for PZR pressure. In all graphs, the y-axis represents the PZR pressure with a small scale, and the x-axis represents the time from the moment of measurement to 600 s. The model number of each graph follows the model order in Table 4.

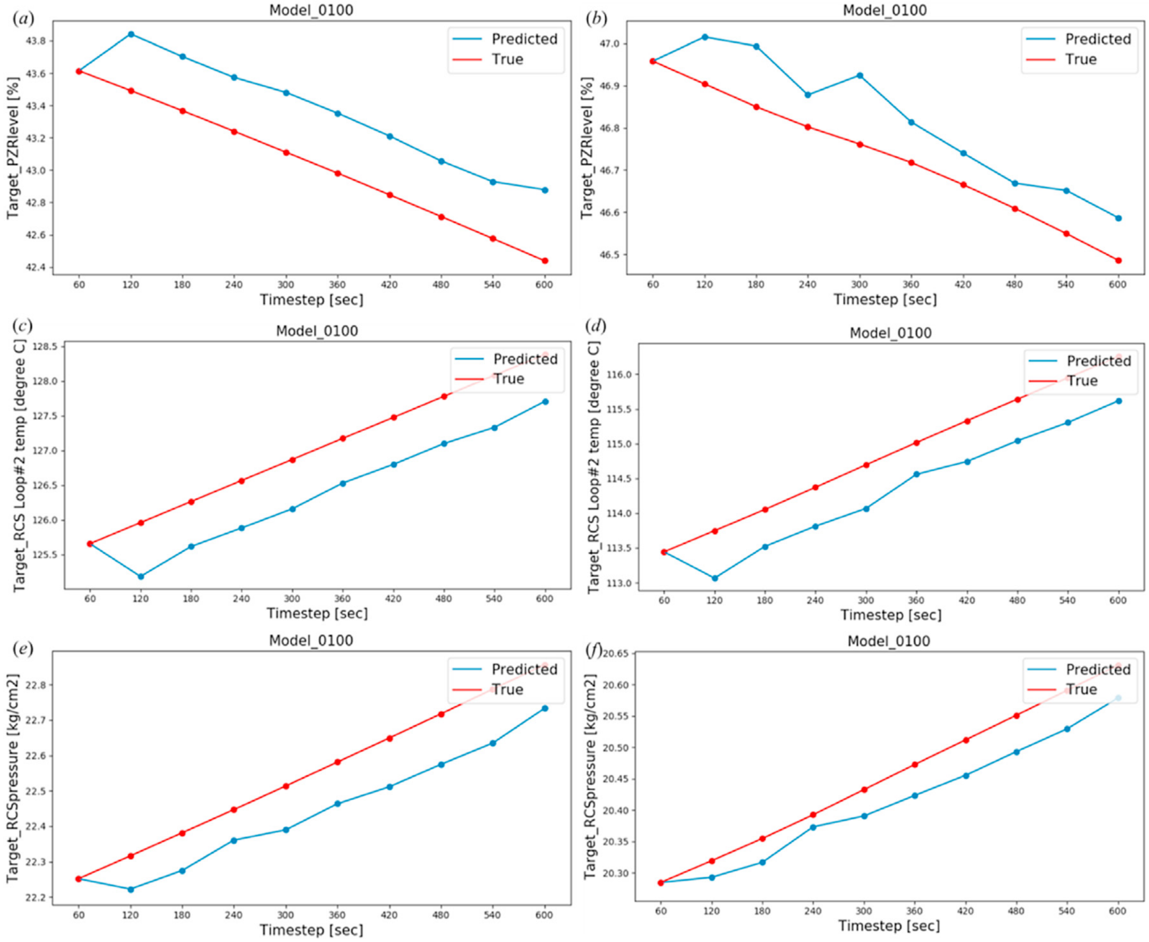

Fig. 11. Prediction results of model #4, where only the charging flow control valve open signal is received (True: actual value, Predicted: predicted value). (a,b) PZR level. (c,d) RCS loop#2 average temperature. (e,f) PZR pressure.

Table 5
Comparative results of each quantification method with all actions as action candidates.

Method	Temp	P_{green}	P_{yellow}	P_{red}	P%	L_{green}	L_{yellow}	L_{red}	L%
Without coordination	170.84	14210	5788	2	71.1%	14344	887	0	94.2%
Comparison	172.17	15254	4744	2	76.3%	6076	8012	0	43.1%
Average	172.25	16853	2741	406	84.3%	14306	837	0	94.5%
Area	172.32	15326	4673	1	76.6%	14004	1254	0	91.8%

considering all actions and agents' actions as candidates, respectively. When all actions were considered, the quantification method using the average of the 9 future states showed the most improved results for PZR pressure and water level compared to the uncoordinated case. The case where only the agents’ action sets were considered as candidates showed lower performance than the uncoordinated case for all methods. In all methods, the heating rate did not exceed the limit (28 °C/h), and the final RCS cold leg #2 temperatures for all methods were similar.

4.3. Discussion

From the experimental results, it can be seen that quantifying the whole trend with the average and area methods rather than comparing the final values only was advantageous to achieve the desired operation goal. Among the quantification methods, it was

Table 6
Comparative results of each quantification method with agents’ action sets as action candidates.

Method	Temp	P_{green}	P_{yellow}	P_{red}	P%	L_{green}	L_{yellow}	L_{red}	L%
Without coordination	170.84	14210	5788	2	71.1%	14344	887	0	94.2%
Comparison	171.92	13444	6555	1	67.2%	8620	6819	0	55.8%
Average	170.98	15360	4639	1	76.8%	1640	13616	8	10.7%
Area	170.53	15859	4140	1	79.3%	2068	12694	12	14.0%

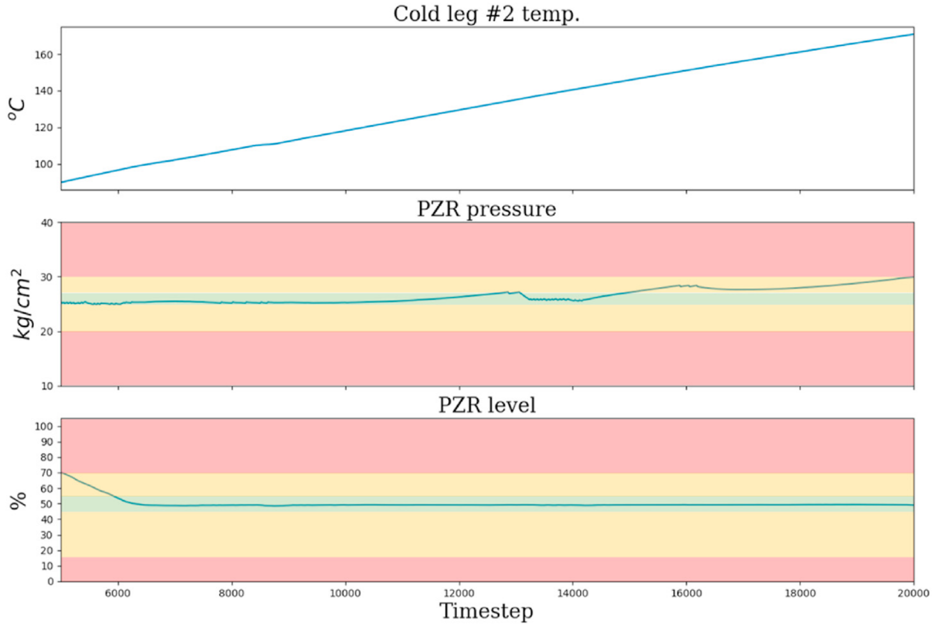

Fig. 12. Representative variable trends with LCO-based regions following coordinated operation by the two agents controlling PZR pressure and water level (average case).

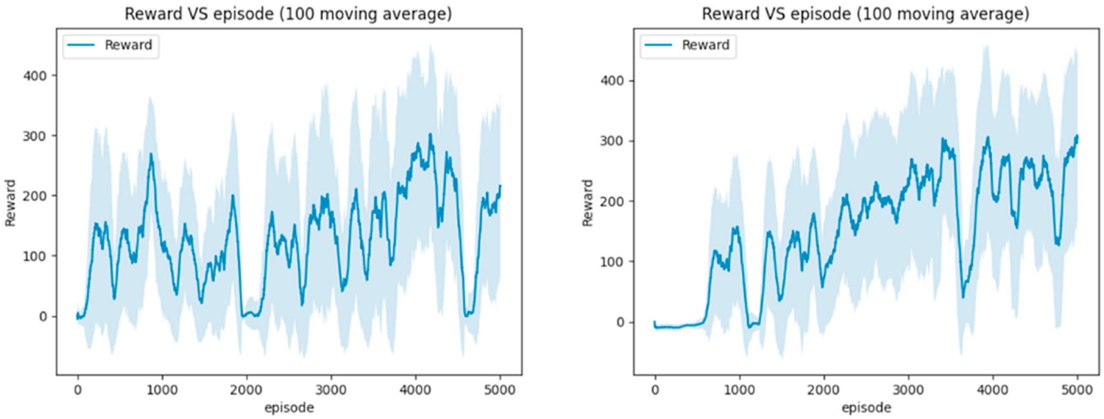

Fig. 13. Accumulated reward for each episode for the PZR water level control agent (left) and PZR pressure control agent (right).

confirmed that the use of the average showed the greatest performance improvement, with the method considering the area also showing improvement compared to the uncoordinated case.

As shown in Fig. 10, the variable shows different trends depending on the selected actions. In a future study, we plan to extend the scope to maintain the accuracy of the plant parameter prediction models while increasing the prediction time range so that more dynamic changes can be seen.

In terms of the performance evaluation, the classified regions do not mean that operation has failed or is wrong even when the yellow (acceptable) or red (violation) regions have high values. In all operating simulations, the LCO was sufficiently satisfied, and by adding a quantification method, both variables were mostly maintained in the green (recommended) region. Considering the scale, discretely dividing the regions where the variables need to be maintained and assigning scores is considered appropriate as a general approach for application to other operations. Fig. 12 shows the result to obtain the indicators for performance evaluation used in Eq. (7) with colored regions. The PZR pressure and water level graphs were maintained within the regions suggested by the LCOs. At a high temperature, the PZR pressure rises together within the range that does not deviate from the PT curve, but in this experiment, the pressure was controlled to the end.

Fig. 13 displays graphs of the accumulated rewards by the PZR pressure and water level control agents at each episode. As the episodes progress, it can be seen that each agent was generally able to find the optimal policy to increase the cumulative reward. However, each agent does not consider the goals of the other, only focusing on the action candidates that achieve their local goal. Table 5 indicates that when actions conflict, it is better to consider all actions as candidates to achieve the overall operation goal rather than considering only the action sets suggested by the agents. Because a new operation strategy that reflects the LCOs in a multi-agent environment is required, a new optimal policy is needed [23,24].

In order to apply RL to the operation of an NPP, an agent can be assigned to a single component (e.g., one valve) as the smallest unit, or an agent that controls multiple components can be created as discussed in this paper. As the number of input variables and actions corresponding to outputs increases, the size of the dimension on which the agent should converge increases. In this study, we limited the size of the dimension by efficiently training two agents and separating the learning environment from the complex conditions required for achieving multiple goals. However, by researching cases of finding the optimal policy in a more complex environment with the latest deep learning techniques, it is expected that single agents can be created that cover all operations in the future.

5. Conclusion

This paper introduced an action coordinating strategy for developed AI agents considering domain knowledge as part of the development of an NPP autonomous operation system for startup operation. AI agents were implemented using SAC algorithms to achieve the optimal operation policy. However, since each agent was trained in a local environment, a strategy is required to lead the desired operation goal in a multi-agent environment. For this, LSTM networks were employed to make plant parameter prediction models that provide future information according to potential actions so that the autonomous operation system can quantify the LCOs. Three quantification strategies were compared to assess the candidate actions, where the method comparing the average values of the variable trends showed the best results. Ultimately, it was possible to coordinate the actions between multiple agents by dividing the variables related to the LCOs into discrete domains, giving them scores, and selecting the optimal action with the highest score.

The goal of this study was to explore how to successfully perform startup operation by coordinating various operating blocks with AI agents that perform subdivided operation tasks. While completely autonomous systems in the nuclear energy field are both technically and legally challenging, it is expected that continued research into this area can provide support to operators with high workloads over long periods.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported by a Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korean government (MOTIE) (No. 20211510100020) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT). (No.RS-2022-00144042).

References

[1] Stuart Bennett, The past of pid controllers, Annu. Rev. Control 25 (2001) 43–53.
[2] H. Basher, Autonomous Control of Nuclear Power Plants, 2003 (ORNL/TM–2003/252)," United States.
[3] R.E. Uhrig, L.H. Tsoukalas, Multi-agent-based anticipatory control for enhancing the safety and performance of gen4 NPPs during long-term semi-autonomous operation, Prog. Nucl. Energy 43 (1–4) (2003) 113–120.
[4] R.T. Wood, B.R. Upadhyaya, D.C. Floyd, An autonomous control framework for advanced reactors, Nucl. Eng. Technol. 49 (Issue 5) (2017) 896–904.
[5] J.T. Kim, K.C. Kwon, I.K. Hwang, D.Y. Lee, W.M. Park, J.S. Kim, S.J. Lee, Development of advanced I&C in nuclear power plants: ADIOS and ASICS, Nucl. Eng. Des. 207 (1) (2001) 105–119.
[6] S.J. Lee, P.H. Seong, Development of automated operating procedure system using fuzzy colored petri nets for nuclear power plants, Ann. Nucl. Energy 31 (8) (2004) 849–869.
[7] M. Boroushaki, M.B. Ghofrani, C. Lucas, M.J. Yazdanpanah, An intelligent nuclear reactor core controller for load following operations, using recurrent neural networks and fuzzy systems, Ann. Nucl. Energy 30 (1) (2003) 63–80.
[8] S.J. Lee, K. Mo, P.H. Seong, Development of an integrated decision support system to aid the cognitive activities of operators in main control rooms of nuclear power plants, in: 2007 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making, 2007, pp. 146–152.
[9] M.H. Hsieh, S.L. Hwang, K.H. Liu, S.F.M. Liang, C.F. Chuang, A decision support system for identifying abnormal operating procedures in a nuclear power plant, Nucl. Eng. Des. 249 (2012) 413–418.
[10] J.M. Kim, G. Lee, C. Lee, S.J. Lee, Abnormality diagnosis model for nuclear power plants using two-stage gated recurrent units, Nucl. Eng. Technol. 52 (9) (2020) 2009–2016.
[11] J. Kim, D. Lee, J. Yang, S. Lee, Conceptual design of autonomous emergency operation system for nuclear power plants and its prototype, Nucl. Eng. Technol. 52 (2) (2020) 308–322.
[12] P.H. Seong, H.G. Kang, M.G. Na, J.H. Kim, G. Heo, Y. Jung, Advanced mmis toward substantial reduction in human errors IN NPPS, Nucl. Eng. Technol. 45 (2) (2013) 125–140.
[13] KINS, Safety and Operational Status of Nuclear Power Plants in Korea", vol. 21, KINS/ER-035, 2020.
[14] D. Lee, A.M. Arigi, J. Kim, Algorithm for autonomous power-increase operation using deep reinforcement learning and a rule-based system, IEEE Access ume 8 (2020) 196727–196746.
[15] J.M. Kim, H.J. Yang, S.J. Lee, Autonomous startup operation for nuclear power plants using a reinforcement learning algorithm, in: International Symposium on Future I&C for Nuclear Power Plants, 2021.
[16] T. Haarnoja, A. Zhou, P. Abbeel, S. Levine, Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor, in: International Conference on Machine Learning (ICML), PMLR, 2018, pp. 1861–1870.
[17] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[18] R.S. Sutton, A.G. Barto, Reinforcement Learning: an Introduction, 2nd, The MIT Press, 2018.
[19] D. Lee, S. Koo, I. Jang, J. Kim, Jonghyun, Comparison of deep reinforcement learning and PID controllers for automatic cold shutdown operation, Energies 15 (Issue 8) (2022) 2834.
[20] J. Bae, G. Kim, S.J. Lee, Real-time prediction of nuclear power plant parameter trends following operator actions, Expert Syst. Appl. 186 (2021), 115848.
[21] KAERI, Advanced Compact Nuclear Simulator Textbook, 1990.
[22] Diederik P. Kingma, Ba Jimmy, Adam: A Method for Stochastic Optimization, 2014 arXiv preprint arXiv:1412.6980.
[23] D.M. Guisi, R. Ribeiro, M. Teixeira, A.P. Borges, F. Enembreck, Reinforcement learning with multiple shared rewards, Procedia Comput. Sci. 80 (2016) 855–864.
[24] G. Dulac-Arnold, D. Mankowitz, T. Hester, Challenges of Real-World Reinforcement Learning, 2019 arXiv preprint arXiv:1904.12901.