Transactions of the Korean Nuclear Society Autumn Meeting
Goyang, Korea, October 24-25, 2019

Framework of two-level operation module for autonomous system of nuclear power plants during startup and shutdown operation

Jae Min Kim and Seung Jun Lee*
Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan, 44911
**Corresponding author: sjlee420@unist.ac.kr*

1. Introduction

Nuclear power plants (NPPs) are equipped with automatic control systems that do not require direct operator intervention. Figure 1 shows typical operation mode and automation level in a pressurized water reactor (PWR) [1].

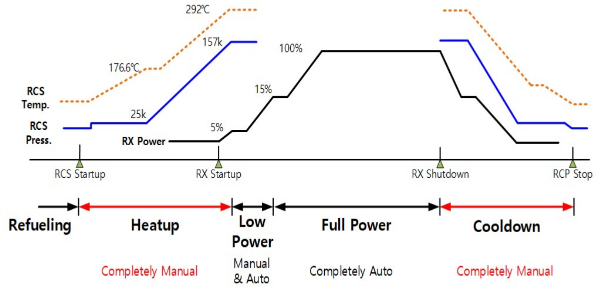

Fig. 1. Typical operation mode and automation level in a PWR [1].

Currently, most control systems in commercial NPPs, such as a pressurizer (PZR) level controller, use a proportional-integral-derivative (PID) controller. However, a PID controller is mainly used to keep a certain variable at setpoint in low and full power modes. In case of heatup (startup) and cooldown (shutdown) operation modes, human operators should operate an NPP by monitoring status of the plant to gradually increase or decrease temperature and pressure according to the situation.

While practical applications are hard to find in commercial NPPs, autonomous operation is one of the main concerns for improving NPP safety. Richard T. Wood *et al.* [2] suggested an autonomous control framework for advanced reactors such as small modular reactors. Advanced reactors are currently in development and can be modified at the design stage to build an automated system. However, it is difficult to apply such framework to already built NPPs. Daeil Lee *et al.* [3] suggested an autonomous operating algorithm using a function-based hierarchical framework and a long short-term memory algorithm for emergency operation. Although this algorithm has shown a successful performance, supervised learning has a problem that a lot of data must be secured in advance. In case of startup and shutdown operations that require an operation time of 20 hours or more, it is difficult to secure a sufficient amount of data.

To overcome this problem, this work proposes an autonomous operation employing both reinforcement learning (RL) algorithm and rule-based logic for startup and shutdown operations. As a part of development, this paper introduces overall framework of two-level operation module for autonomous system. In addition, a simple example of pressurizer level control using a Q-Learning algorithm is presented.

The suggested system is expected to support tasks of operators during long operation time. In addition, based on optimal behaviors learned by an artificial intelligence (AI) model, new ways of operating an NPP might be found.

2. Methodology

2.1 Framework of the autonomous operation module

It is more efficient to divide operating section into modules rather than developing the entire process. Based on a general operating procedure (GOP) for startup and shutdown operations, tasks that need complex judgement and component control are targets to be replaced with AI algorithm. Other than that, a simple rule-base logic covers operating tasks. Figure 2 shows overall framework of autonomous operation module, which is called as two-level operation module.

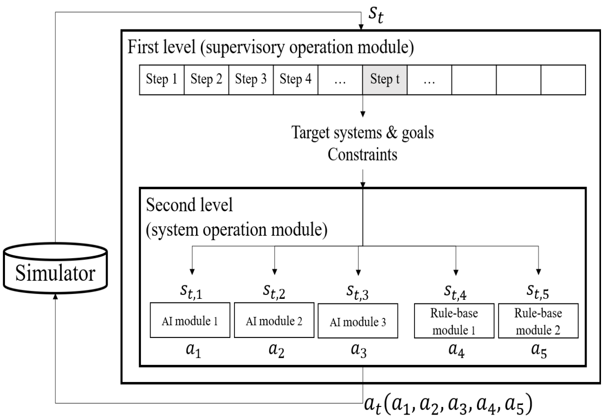

Fig. 2. Overall framework of autonomous operation module.

At the first level, supervisory operation module sets goals and constraints for the second level operation modules. Considering a state ‘s’ of an NPP, the values required by the GOP are determined through rule-based logic. If there are conflicts to control certain components at the second level, supervisory operation module should set priorities among system operation modules.

At the second level, system operation modules operate a system to achieve the defined goal under the constraints. The modules can use either a RL algorithm or rule-based logic. In case of a RL algorithm, each module needs input variables and target components as an output. The modules determine their actions 'a' according to the goals and constraints handed over from the first level.

2.2 Q-Learning

RL is an algorithm that learn how to maximize future rewards through interaction between an agent and its environment as shown in Fig. 3.

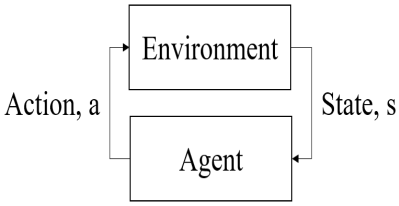

Fig. 3. Interaction between an environment and an agent

Q-Learning is one of RL algorithms for computing an optimal policy with Q function [4]. Q function is initialized at the beginning, and repeat following process by terminal state for each episode:

1. Initialize state s
2. Choose an action from s using policy derived from Q function according to epsilon greedy algorithm
3. Take an action and observe reward and next state
4. Update Q function based on reward and next state
5. Repeat from step 1

An action is selected according to the epsilon greedy algorithm that has decaying probability to select random action at the early stage of training. Thus, an agent obtains information from an environment by choosing random actions until Q function is sufficiently updated. Q function is updated as follow:

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha[R(s,a) + \gamma V(s')] \quad (1)$$

$$V(s) \leftarrow \max_{a \in A} Q(s,a) \quad (2)$$

where α is a learning factor, γ is a discounting factor, s is the current state, s' is the next state, V is a worth function, R is reward by current action a, and A is action space. Reward function should be defined for each model according to its purpose to achieve an optimal policy quickly.

3. Application

A test was conducted employing Q-Learning algorithm. Compact nuclear simulator (CNS) was used for the application. The CNS models three loops Westinghouse PWR, 993MWe, developed by Korea Atomic Energy Research Institute (KAERI) [5].

Figure 4 shows a target system, PZR level control. Experimental settings including hyper parameters are listed in Table 1.

Table 1. Experimental setting for pressurizer level control

Number of episodes	530
Initial level	62.2%
Goal	58%
Input variable	PZR level
Output variable	FV122, HV142, HV603, HV40
Discount factor	0.99
Learning rate	0.1

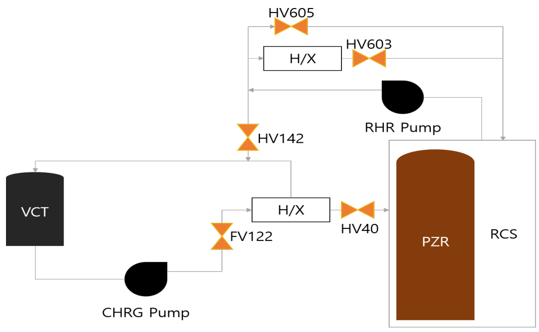

Fig. 4. Simplified system design related with PZR level control

3.1 Input & output variables

PZR level was discretized with 23 intervals that have a smaller size as it neared the target value in one-hot-encoding as shown in Fig. 5. For example, if PZR level is 62%, state is defined as follow:

(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0)

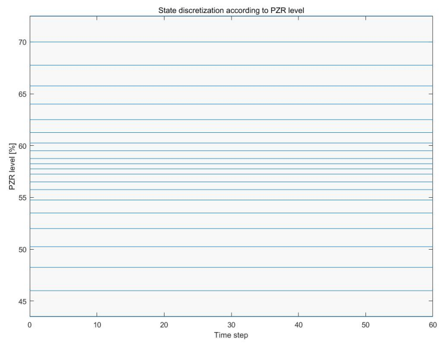

Fig. 5. Discretization of PZR level

When the agent selects an action based on policy function Q, the action is expected to maximize reward in the next state. To provide an action space, each valve is controlled with activation of a toggle, not directly change the parameters. For example, if FV122 should open, the agent selects a command to adjust the valve by a certain amount.

There are a total 11 actions with change of FV122, HV142, HV603 and HV40. FV122, HV142 and HV603 are control valves with a change of 1.5%, and HV40 is a check valve that can only be opened and closed. In addition, there is commands for 'no change' and combined control of FV122 and HV142 simultaneously.

3.2 Reward

A reward function was empirically determined by taking an inverse of the difference between the goal and the current PZR level. In addition, a small constant *C* is added to prevent a denominator from becoming zero.

$$R = \frac{1}{abs(goal - level) + C} \tag{3}$$

4. Result & Discussion

The test was conducted with Python 3.6.8 version and tensorflow 1.12.0 version. Q-learning algorithm finished its training in 530 episodes. Operating time per episode is ten minutes with 60 timesteps that represents 5 seconds of interval.

Figure 6 shows the final PZR level at the end of each episode. After 487-th episode, final level of PZR converged around goal, which was set as 58%. To maximize the reward, the agent tends to stay 58% as shown in Fig. 7.

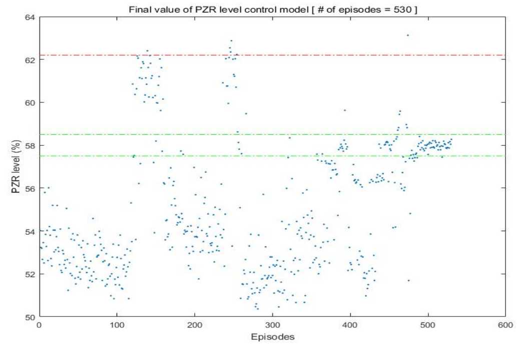

Fig. 6. Final level of PZR for 530 episodes

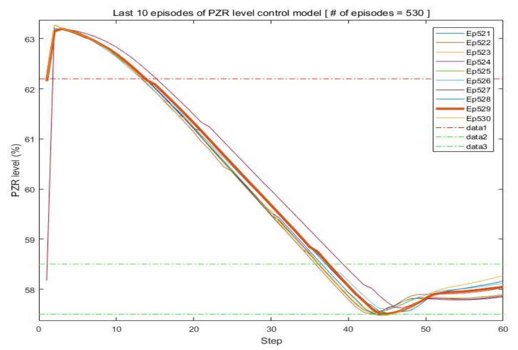

Fig 7. PZR level change during operating time for the last 10 episodes

The results proved that Q-Learning algorithm can be successfully applied to autonomous operation of an NPP. However, the algorithm is not applicable to all cases that require autonomous operation. Through lots of tests, the input variable was preprocessed by one-hot-encoding. In addition, adjustment of hyperparameters requires a lot of effort because the adjustment should be made while analyzing the actual result.

5. Conclusion

This paper proposed framework of autonomous operation for startup and shutdown operations of an NPP. Two-level operation module was suggested having a supervisory module at the first level and operation modules that perform each operation task at the second level. To overcome lack of the number of data to be secured, RL algorithm is employed, Q-Learning algorithm in this paper. The results showed that RL algorithm can be successfully applied to autonomous operation with fine tuning of operating modules.

While completely autonomous system is both technically and legally challenging, it is expected that research on this area will support operators who might take high workload over long periods of operating time.

6. Acknowledgement

This work was supported by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Trade, Industry & Energy (MOTIE) of the Republic of Korea (No. 20171510102040).

REFERENCES

[1] Poong Hyun Seong, Hyun Gook Kang, Man Gyun Na, Jong Hyun Kim, Gyun Young Heo and Yoen Sub Jung: Advanced MMIS toward substantial reduction in human errors in NPPs, Nuclear Engineering and Technology Volume 45, Issue2, pp.125-140 2013.

[2] Richard T. Wood, Belle R. Upadhyaya, Dan C. Floyd, An autonomous control framework for advanced reactors, Nuclear Engineering and Technology, Volume 49, Issue 5, pp. 896-904, 2017.

[3] Daeil Lee, Poong Hyun Seong, Jonghyun Kim, Autonomous operation algorithm for safety systems of nuclear power plants by using long-short term memory and function-based hierarchical framework, Annals of Nuclear Energy, Volume 119, pp.287-299, 2018.
Pages 287-299,

[4] Watkins, C.J. & Dayan, P. Technical Note: Q-Learning, Machine Learning vol. 8, Issue 3-4, pp.279-292, 1992.

[5] KAERI, Advanced Compact Nuclear Simulator Textbook, Nuclear Training Center in Korea Atomic Energy Research Institute, 1990.