COMPLEX SYSTEM MODELING AND SIMULATION
ISSN 2096-9929 02/06 pp 213–223
Volume 2, Number 3, September 2022
DOI: 10.23919/CSMS.2022.0014

Dynamic Scheduling Algorithm Based on Evolutionary Reinforcement Learning for Sudden Contaminant Events Under Uncertain Environment

Chengyu Hu, Rui Qiao, Zhe Zhang, Xuesong Yan*, and Ming Li

Abstract: For sudden drinking water pollution event, reasonable opening or closing valves and hydrants in a water distribution network (WDN), which ensures the isolation and discharge of contaminant as soon as possible, is considered as an effective emergency measure. In this paper, we propose an emergency scheduling algorithm based on evolutionary reinforcement learning (ERL), which can train a good scheduling policy by the combination of the evolutionary computation (EC) and reinforcement learning (RL). Then, the optimal scheduling policy can guide the operation of valves and hydrants in real time based on sensor information, and protect people from the risk of contaminated water. Experiments verify our algorithm can achieve good results and effectively reduce the impact of pollution events.

Key words: evolutionary reinforcement learning; water distribution network; scheduling problem

1 Introduction

Clean drinking water is directly related to people's health and safety. Water distribution networks (WDNs) are the critical infrastructures which convey drinking water from sources to consumers. However, WDNs are widely covered in cities and towns and remain open days and nights, making it extremely prone to pollution events. Harmful chemicals and pathogens can easily enter a WDN because of the breakage of any pipe. Once drinking water pollution occurs, the contaminant will quickly spread in the WDN, causing health problems. In recent years, drinking water pollution incidents have caused major economic losses and bad social impacts on our country. Therefore, it is necessary to find a good way to automatically cope with sudden pollution events and reduce their negative[1, 2].

In a WDN, many water quality sensors are usually deployed[3–5]. They monitor the water quality in real time and quickly raise an early warning once pollution events occur[6, 7]. For sudden pollution events, the intuitive and easiest response is to cut off the water supply of the entire WDN. However, this causes large-scale water cuts and serious economic losses. Another method is to locally operate valves and hydrants and ensure that contaminant is only discharged in the polluted WDN. By closing some valves, contaminated water can be isolated within a certain range. By opening some hydrants, contaminant can be discharged as soon as possible. The last method can effectively reduce the impact of pollution events. However, the challenging problem is which of the valves and hydrants to use so that they will be optimally operated in real time.

Due to the large scale of urban WDNs and many random factors in the complex WDNs, the scheduling problem of valves and hydrants has evolved into a high-dimensional spatial combinatorial optimization problem under uncertain environments. Many researchers have used heuristic optimization algorithms to solve the scheduling problems of valves and

• Chengyu Hu, Rui Qiao, Zhe Zhang, and Xuesong Yan are with the School of Computer Science, China University of Geosciences, Wuhan 430074, China. E-mail: yanxs@cug.edu.cn.
• Ming Li is with the Department of Computer Science, California State University, Fresno, CA 93740, USA.
* To whom correspondence should be addressed.

Manuscript received: 2022-04-17; revised: 2022-06-15; accepted: 2022-07-15

hydrants[8, 9]. However, this problem involves many large-scale computation-intensive tasks, such as hydraulic and water quality simulations, which cause huge computational overhead, so heuristic optimization algorithms cannot meet the requirement of real-time computing[10]. Additionally, the randomness of pollution events makes it hard to gain an optimal solution by the heuristic optimization algorithm[11].

Deep reinforcement learning (DRL) is one of the most rapidly developing methods in the field of artificial intelligence. By combining the learning and expression mechanism of deep learning, DRL fully exploits both the decision-making and perceptual abilities of complex control problems. It has been widely used in many fields[12, 13]. Evolutionary reinforcement learning (ERL) is a hybrid algorithm which combines the advantages of evolutionary computation (EC) and reinforcement learning (RL), including two stages of population optimization and agent policy update. The parameters of the strategy network trained by RL are copied to the population regularly. The excellent experiences generated by EC in the evaluation process fill the replay buffers of the DRL method and help the agent to perform gradient updating, thus, the agent learns the optimal strategy faster.

ERL can fittingly deal with emergency scheduling problems for sudden pollution events. It not only meets the high performance requirements, but it can also schedule valves and hydrants in real time. In order to further enhance the performance of the ERL and make it more suitable to solve dynamic scheduling problems, in this paper, a novel emergency scheduling algorithm based on ERL (ESERL) is proposed to cope with drinking water pollution incidents in a WDN. The algorithm trains the deep neural network offline, and then the learned strategy is applied to the control center to schedule the valves and hydrants in real-time according to incoming sensor information, so as to meet the high real-time requirements of the problem.

Specifically, the main contributions of this paper are as follows:

- The valve and hydrant scheduling problem is modeled as a Markov decision optimization problem, and a novel ERL algorithm which combines EC and RL is proposed. The offline optimal scheduling policy is learned to meet the time requirements of emergency scheduling problems.
- The EPANET simulator is employed to simulate the transform of hydraulic and water quality. Extensive simulation results show that the ERL is a competitive method which can solve the problem of valve and hydrant scheduling and effectively reduce the impact of pollution events.

The rest of this paper is structured as follows. Section 2 introduces the related work of the scheduling of valves and hydrants in a WDN. Section 3 elaborates on valve and hydrant scheduling problems. Section 4 introduces our proposed ERL-based scheduling algorithm. Section 5 presents the experimental results. Finally, the paper is summarized in Section 6.

2 Related Work

2.1 Emergency scheduling in WDNs

WDN emergency scheduling involves opening and closing valves and hydrants to isolate and discharge contaminants. An optimal scheduling policy effectively reduces the impact of pollution events and provides the quick return of a normal water supply. In recent years, heuristic optimization algorithms have been used to compute an optimal scheduling scheme. For example, aiming to minimize the consumption of contaminated water, Gavanelli et al.[14] used heuristic methods to solve the scheduling problem when pollution events occurred, and genetic algorithm (GA) was used as a feasible tool to solve the optimization simulation problem. Rasekh and Brumbelow[15] established a dynamic decision support model that can adaptively simulate the time-varying emergency environment and track the changes of the best health protection response measures at each stage of an emergency in real time. Shafiee and Berglund[16] proposed a sensor-hydrant decision tree. After training, the optimal scheduling policy can be automatically given by the hydrant activated by induction, without the need for information about the characteristics of the pollution source. They also further constructed an adaptive emergency response framework[17]. By integrating decision tree model and agent simulation technology, the framework can easily evaluate the effectiveness of a scheduling policy under different pollution events.

In order to deal with randomness in an emergency scheduling problem, Schwartz et al.[18] proposed a limited multistage stochastic programming algorithm for optimal scheduling of pumps and reduced the complexity of classical multi-stage stochastic programming by adding constraints. Khatavkar and

Mays[19], aiming to minimize the difference between the required demand and the satisfied demand, proposed a framework of optimal control to determine the real-time scheduling policy of the WDN.

Although heuristic optimization algorithms can solve some simple scheduling problems, they cannot respond in real-time to random pollution events. The first reason is that the optimization algorithm's online search process takes a long time to evaluate the fitness function, so it is too slow for emergency events when quick responses are needed. The second reason is that the randomness of pollution events and the uncertainty of water demand often make the optimal solution ineffective.

RL is one critical method for dealing with dynamic and real-time sequential decision optimization problem. RL can learn an emergency scheduling policy offline. By conducting scheduling operations on valves and hydrants according to sensor readings, the agent can learn an optimal scheduling policy which can control the valves and hydrants in real time when there are dynamic and uncertain environments. Simulation results show that the RL method can solve the problems of the timeliness of emergency scheduling and the randomness of pollution events[20].

2.2 ERL for scheduling problems

ERL inherits the advantages of EC and RL. On one hand, population can enhance the agent's learning ability and produce excellent policy network parameters, and thus the agent gets high rewards. On the other hand, the agent can accelerate the search speed of the population by random gradient algorithm. The collaborative search can make the ERL have a strong ability of "exploration" and "exploitation"[21].

Khadka et al.[22] proposed an ERL algorithm combining the GA and the deep deterministic policy gradient (DDPG). Experiments showed that this algorithm performed better than RL or EC in MuJuco continuous control tasks. Lü et al.[23] proposed an extensible framework combining the advantages of RL, EC, and imitation learning, which improves the learning process of agents in traditional ERL methods. Gupta et al.[24] introduced a deep ERL framework, in which different agent forms evolve to learn challenging motor and operational tasks in a complex environment.

ERL has demonstrated its performance advantages in many scheduling problems[25, 26]. For example, in the field of job shop scheduling, most problems are solved by heuristic algorithms[27–29]. Wei et al.[30] proposed an iterative optimization framework for a dynamic scheduling system based on RL and GA. This framework is used to schedule jobs in a dynamic job shop. The GA is used to drive parallel search and evolution direction, and the phased Q-learning method is used to realize RL system. Zeng et al.[31] proposed a DRL-based data representation method and an evolutionary job scheduling algorithm based on population optimization. DRL was used to initialize the GA population, and then the final scheduling result was obtained by GA evolution population.

Kamiya et al.[32] combined neural network based RL with the GA in order to improve the online performance of an optimal or near-optimal start-up plan search in power plant operations. The algorithm searchs for optimal or near-optimal start-up plans under a given set of stress limits. In order to improve the efficiency of the multi-robot intelligent warehouse system, Dou et al.[33] combined task scheduling based on the GA with path planning based on RL to form an effective ERL algorithm, which was verified by simulation results.

Ahmed et al.[34] proposed an algorithm integrating RL and the particle swarm optimization algorithm (PSO) for school bus scheduling optimization. Xu et al.[35] proposed an industrial multi-energy scheduling framework to optimize the use of renewable energy and reduce energy costs. To solve this problem, a differential evolution algorithm based on RL determines the optimal mutation strategy and related parameters in an adaptive way. The effectiveness of the algorithm in reducing energy costs in an industrial environment was demonstrated through a case study of real world data. Li et al.[36] improved the GA based on RL and used it for the scheduling of multi-agent tasks in intelligent control systems.

To sum up, ERL algorithms can effectively solve the scheduling problem and have achieved good research results in the field of control and scheduling. Therefore, we have also used a modified ERL algorithm to solve the emergency scheduling problem of the WDN studied in this paper.

3 System Architecture and Formulation

3.1 System formulation

Once a contaminant enters the WDN, it moves through the entire system with water flow over time, affecting large parts of the population. Closing a valve can isolate contaminated water, and opening a valve in uncontaminated areas can ensure the normal use of water by residents. Opening the hydrant can release sewage, and closing it prevents uncontaminated water from being wasted. However, it is challenging to find a good control strategy to close or open valves and hydrants because of the large size of a WDN.

To minimize the impact of pollution, it is better to discharge the contaminant as soon as possible. That is, during the period from the occurrence of pollution to the restoration of the water supply, the goal is to maximize the cumulative amount of contaminant discharged by the hydrant. The optimization model is as follows:

$$\max \sum_{t=1}^{T} \sum_{h=1}^{H} D_h(v_1, v_2, \ldots, v_N, h_1, h_2, \ldots, h_H) \tag{1}$$

where v is the decision variable of the valve; the value is 1 or 0, which means opening and closing the valve, respectively; h is the decision variable of the hydrant, and the value is 0 and 1, which means closing and opening the hydrant, respectively. D_h indicates the quantity of contaminant discharged by the hydrant h by performing the operations on the valve and hydrant, which can be simulated through the open-source software EPANET[37]. N is the number of valves; H is the number of hydrants. t is the simulation step; T is the total simulation time.

Once the water quality sensor issues a warning alarm, we need to quickly respond and give the optimal scheduling policy by operating the valves and hydrants. Therefore, a deep neural network should be trained offline by ERL in advance. A better-trained network can give optimal scheduling schemes in real time according to the information provided by the sensors. The scheduling policy is expressed as follows:

$$\begin{gathered} v_1, v_2, \ldots, v_N, h_1, h_2, \ldots, h_H = \pi(e_1, e_2, \ldots, e_M), \\ a = \pi(s_t) \end{gathered} \tag{2}$$

where e is the information provided by the sensors; all sensor readings at time t are regarded as state s, and the set of valve and hydrant operations is regarded as action a. The scheduling policy π outputs action a according to state s at time t.

3.2 System architecture

The valves and hydrants in a WDN are controlled by the control center. When the water quality sensors give an early warning, the control center uses the better-trained deep neural network to generate an optimal policy; it isolates some valves, discharges the contaminant, and minimizes the impact of the pollution event. The scheduling framework is shown in Fig. 1.

Sensors constantly monitor the water quality. When contamination occurs, the control center's scheduling policy outputs actions based on input concentration information. Valves and hydrants in the WDN are opened and closed according to the action information, while the pollutants spread with the water flow and discharge through the hydrant. In this way, the concentration of pollutants in the pipe network changes, and the scheduling policy takes appropriate actions based on the latest concentration information. After a period of time, if the concentration of water detected by the sensors has not returned to normal, the control center will continue to dispatch the valves and hydrants until it does. Sensors are still monitoring water quality information, and the above scheduling process will be repeated once pollution occurs.

4 ERL-Based Scheduling Algorithm

4.1 Markov decision processes

Emergency scheduling optimization for sudden drinking water pollution is not only a high-dimensional space combinatorial optimization problem, but also a discrete control optimization problem with real-time requirements. That is, once a sensor detects a pollution event, an emergency scheduling policy needs to be given immediately. However, due to the huge computational overhead of simulating the transform of hydraulic and water quality, until now, there has not been an efficient method that can solve scheduling

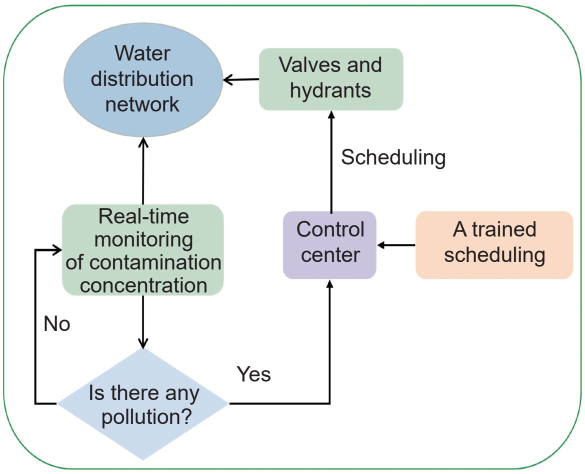


Fig. 1 System architectures of emergency response in a WDN.

optimization problems in time under real environmental conditions.

In recent years, RL has been considered as an alternative method to solve sequential decision tasks[38]. The emergency scheduling problem, by its nature, is sequential decision optimization. Thus, we solve this problem with ERL.

In addition, the Markov decision process (MDP) is a general model for sequential decision tasks[39]. MDP is composed of a four-tuple (s, a, p, r): s represents the state; a represents the action; p represents the state transition probability; and r represents the reward function. In the emergency scheduling problem, MDP is defined as follows:

- s: s is an M-dimensional vector composed of the readings of all water quality sensors in a WDN, representing the state of the agent, and M is the number of sensors.
- a: a represents the operation of valves and fire hydrants in a WDN. The actions of N valves and H hydrants can be represented by an $N+H$ dimensional vector. The first N dimension represents the switching operation of the valves, and the latter H dimension represents the switching operation of the hydrants.
- p: p represents the probability of performing action a in state s and transferring the state to another state. The emergency scheduling problem is a model-free problem. In state s, the probability of transitioning to the next state after action a is executed is unknown.
- r: r represents the reward obtained after performing action a in state s, and the reward function is expressed as $r(s,a)$. The reward is accumulated through continuous interaction with the environment, and the goal of RL is to maximize the accumulated reward. The goal of the emergency scheduling problem is to minimize the impact of pollution. That is, the agent continuously trains, learns the optimal policy, and reduces the impact of pollution events.

The training process is as follows. First, the agent randomly takes action a according to sensor reading state s, then opens or closes the valve and hydrant, obtains the return value r, and transfers to the next state $s\prime$. The process continues in a circular manner until the contaminants are completely discharged and the water supply resumes. After continuous training, the agent learns the optimal policy. Finally, the well-trained deep neural network is deployed to the control center to output optimal policy, which can deal with sudden drinking water pollution events in real-time.

4.2 ERL algorithm

In order to learn an optimal policy, we propose an emergency scheduling algorithm based on ERL (ESERL) to train the deep neural network. ESERL is divided into two parts, which are the cross-entropy method (CEM) and DDPG. The framework of ESERL is shown in Fig. 2.

The agent is parameterized as θ^{π}; the information provided by the sensors is input, and the operation of the valve and hydrant is output. The reward evaluated in the environment is the volume of contaminant discharged from the hydrants, which is calculated by the open-source software EPANET. The reward R is calculated as follows:

$$R=\sum_{t=1}^{T}\sum_{h=1}^{H}D_h(\theta^{\pi}) \tag{3}$$

As shown in Fig. 2, the left side is the cross-entropy algorithm. First, the agent population is initialized and evaluated in the environment, and the reward is obtained according to Eq. (3). Then, half of the individuals with the higher reward are selected to update the mean μ and covariance matrix Σ as in the Eqs. (4) and (5):

$$\mu_{new}=\sum_{i=1}^{k}\lambda_i z_i \tag{4}$$

$$\Sigma_{new}=\sum_{i=1}^{k}\lambda_i(z_i-\mu_{old})(z_i-\mu_{old})^{\mathrm{T}}+\epsilon I \tag{5}$$

where λ_i is the weight coefficient of the individual, commonly chosen with $\lambda_i=\frac{1}{k}$. z_i is the i-th sample. I is the identity matrix. ϵ is an exponentially decaying factor. In Fig. 2, the right side is the DDPG algorithm. An individual in the population is selected as the actor

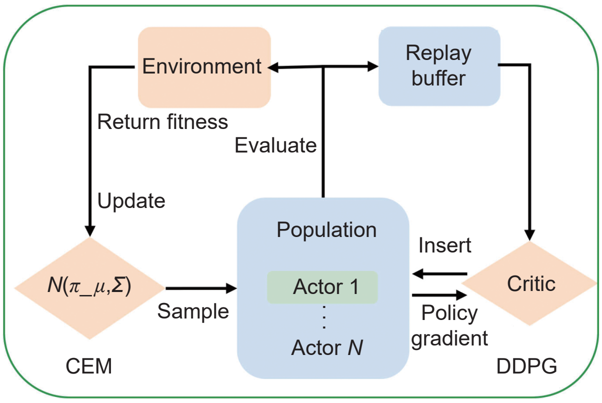


Fig. 2 ERL framework for emergency scheduling problem.

θ^{π}; the critic θ^{Q}, target actor $\theta^{\pi'}$, and target critic $\theta^{Q'}$ are initialized. The experience of the replay buffer is used to update the actor and critic. Finally, the trained actor randomly replaces one actor in the population, and the iteration continues.

The critic is trained by minimizing the loss function:

$$\begin{aligned} L(\theta^{Q}) &= \frac{1}{T}\Sigma_i(y_i - Q(s_i, a_i \mid \theta^{Q}))^2, \\ y_i &= r_i + \gamma Q'(s_{i+1}, \pi'(s_{i+1} \mid \theta^{\pi'}) \mid \theta^{Q'}) \end{aligned} \tag{6}$$

The Actor is trained using the policy gradient:

$$\nabla_{\theta^{\pi}} J \backsim \\ \frac{1}{T}\Sigma \nabla_a Q(s, a \mid \theta^{Q}) |_{s=s_i, a=a_i} \nabla_{\theta^{\pi}} \pi(s \mid \theta^{\pi}) |_{s=s_i} \tag{7}$$

In the ESERL algorithm, agents and the population cooperatively learn an optimal policy. The agent in DRL can learn a policy in a dynamic environment. Nevertheless, the agent may be stuck on a plateau because of deceptive gradient information. By combining population-based algorithms, such as cross-entropy algorithms, the agent can quickly learn a scheduling policy. The pseudo code of the algorithm ESERL is shown in Algorithm 1.

Algorithm 1 ESERL

```
Initialize a random actor $\pi_\mu$ to be the mean of the CEM algorithm
Initialize the critic $Q^\pi$ and the target critic $Q_t^\pi$
Initialize a population of $pop\_size$ actors $pop$
Initialize the scheduling period $T$ and the maximum number of iterations $G$
for $generation \in [1, G]$ do
    Simulate pollution scene in WDN;
    Observe the initial state $s$
    for $i \in [0, pop\_size]$ do
        for $k \in [0, T]$ do
            $a = \pi(s)$, execute action $a$ in simulator;
            Calculate the quality of the pollutant emitted as reward $r$.
            Observe the next state $s'$.
        end for
    end for
    Update $\pi_\mu$ and $\Sigma$ with the top half of the population based on reward $r$ according to Eqs. (4) and (5)
    Generate the next generation from $N(\pi_\mu, \Sigma)$
    Set the actor policy $\pi$ to $pop[0]$
    Initialize a target actor $\pi_t$ with the weights of $\pi$
    Train $Q^\pi$ and $\pi$ using the policy gradient according to Eq. (6) and Formula (7)
    Reintroduce the weight of $\pi$ in $pop$
end for
```

5 Experiment

5.1 Experimental setup

A typical WDN of Net3_Rossman200 is employed in our simulation which is shown in Fig. 3. The WDN contains 97 nodes, 119 pipes, 2 water sources, and 3 water tanks. Drinking water quality sensors are deployed at nodes 151,111,161, and 207. Hydrants are located at nodes 197,169, and 206. Valves are located at pipes 173,116, and 233. The maximum simulation period T is 24 h; the operation interval is 30 min; the hydraulic and water quality simulation steps are both 5 min, and the discharge volume of a hydrant is 400 gallons per minute.

The structure of the target actor and critic network in the ESERL algorithm is consistent with the actor and critic network. In addition, the network is trained by the Adam optimizer with a learning rate of 1×10^{-3}[40]. The actor and critic neural network parameters are shown in Table 1. The learning rate lr is 1×10^{-2}, the discount rate γ is set to 0.99, and the target weight τ is 5×10^{-3}. The agent population size is five. The number of training iterations is 500.

In the experiment, in order to show the effectiveness of ESERL, the baseline scheduling policies tested are VCHO and VOHO. VCHO is to keep all hydrants open and valves closed during the scheduling period. VOHO is to keep all hydrants open and valves open.

5.2 Single pollution event simulation in WDS

We assume that each pollution event can occur at one point in the WDN. Three single pollution events—PS1, PS2, and PS3 —were randomly selected to show the scheduling effect of the ESERL algorithm. Three

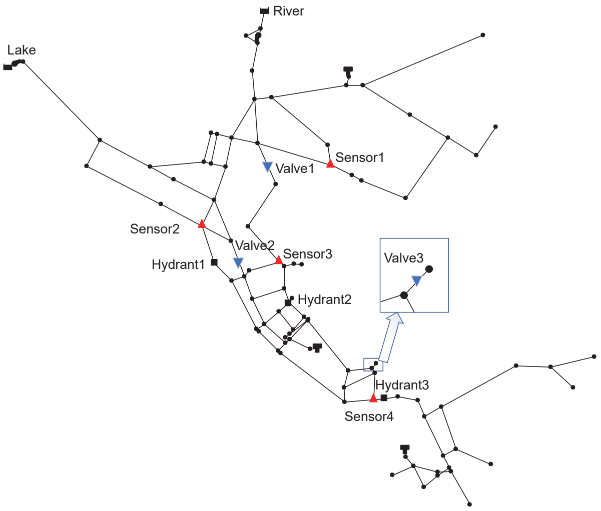


Fig. 3 Water distribution network.

Table 1 Structures of actor and critic network.

Network	Layer	Number of units	Activation function
Actor	Fully connected	64	Tanh
	Fully connected	64	Tanh
	Fully connected	64	Tanh
Critic	Fully connected	64	ReLU
	Fully connected	64	ReLU
	Fully connected	1	Linear

pollution events happened at nodes 105, 120, and 261, respectively. As shown in Fig. 4, different colored stars represent pollution events PS1, PS2, and PS3.

Figure 5 shows the volume of discharged contaminant by different scheduling policies which were trained by ESERL, VCHO, and VOHO algorithms. As can be seen in Fig. 5, the volume of discharged contaminant is the largest when the scheduling policy trained by ESERL is used. The experimental results showed that the performance of the ESERL algorithm is better than that of VOHO and VCHO for three different single pollution events.

ESERL was compared to the ESPPO algorithm. Figure 6 shows the performance of different policies which were trained by ESERL and ESPPO algorithms.

It can be seen from Fig. 6 that for different pollution events, the performance of ESERL algorithm is better than ESPPO. From the perspective of computational overhead, the number of iterations of the ESERL algorithm is 500, and the population consists of five agents, so the total training step of the ESERL algorithm is 2500. The total training step of the ESPPO algorithm is 5000. As a result, the computational overhead of ESPPO is two times of the ESERL algorithm.

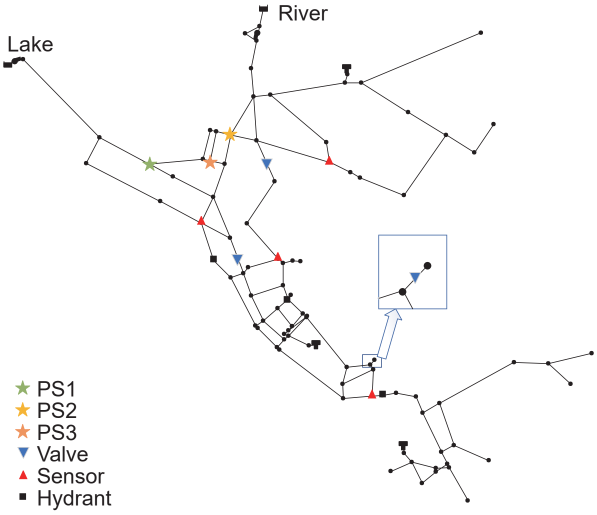


Fig. 4 Three single pollution events PS1, PS2, and PS3.

5.3 Multiple pollution events simulation in WDS

In order to further investigate the ability of the ESERL algorithm to deal with complex pollution events, we assume that each pollution event can occur simultaneously at multiple points in the WDN. In our

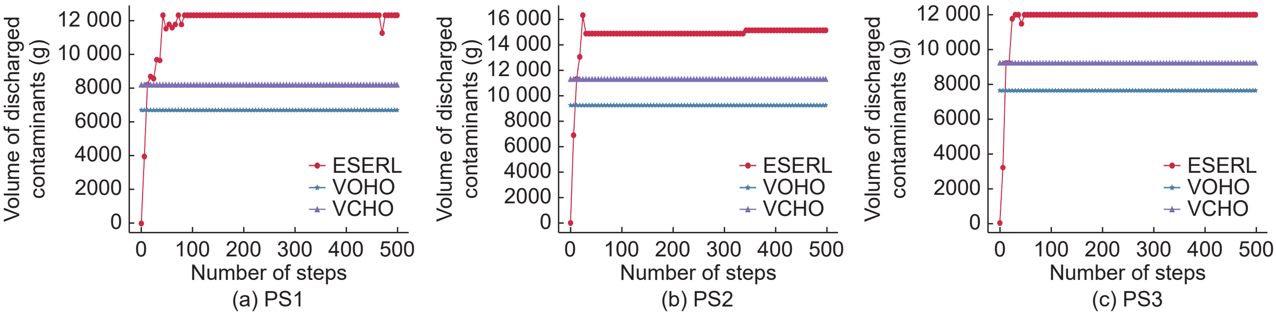


Fig. 5 Volume of discharged contaminants for different single polution events by ESERL, VCHO, and VOHO.

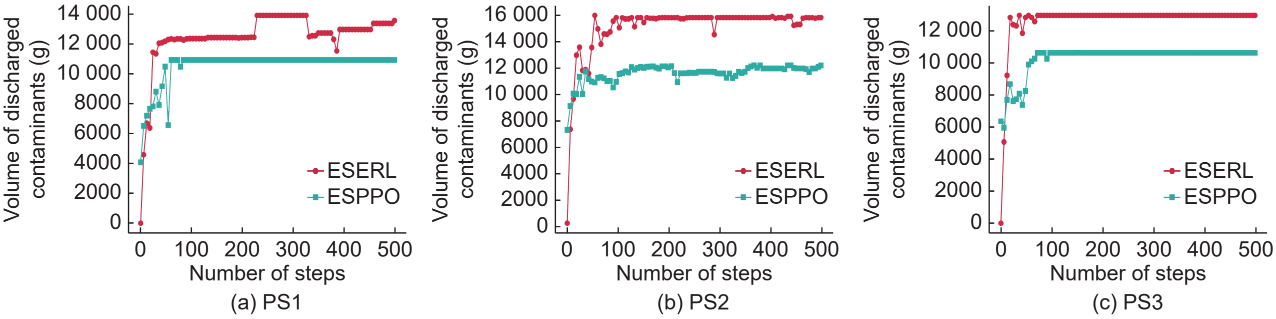


Fig. 6 Volume of discharged contaminants for different single polution events by ESERL and ESPPO.

experiments, PS4 means that pollution events occur simultaneously at the nodes 105, 117, and 261. PS5 means that pollution events occur simultaneously at nodes 120 and 113. In Fig. 7, the green and orange stars represent the contaminant events PS4 and PS5, respectively.

Figure 8 shows the volume of discharged contaminant by different scheduling policies which were trained by ESERL, VCHO, and VOHO algorithms when multiple pollution events occur. It can be seen from Fig. 8 that ESERL algorithm is not stable at initial stages and generally converges in the later stages of the iteration. In addition, it can be seen that the volume of discharged contaminant is the largest when the scheduling policy by ESERL is used, thus we can conclude that the performance of the ESERL algorithm is better than that of VOHO and VCHO for PS4 and PS5.

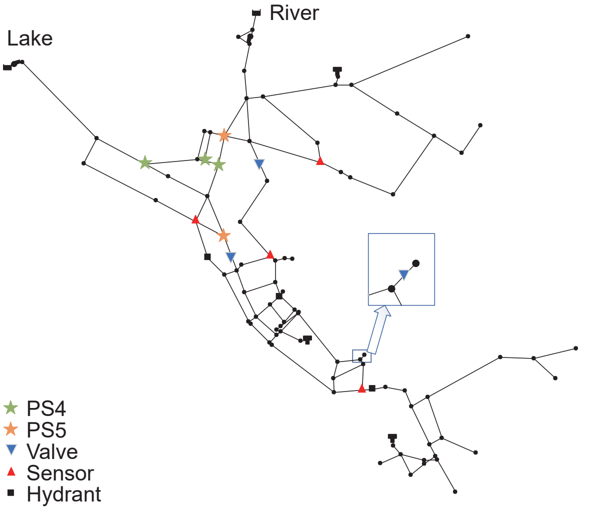
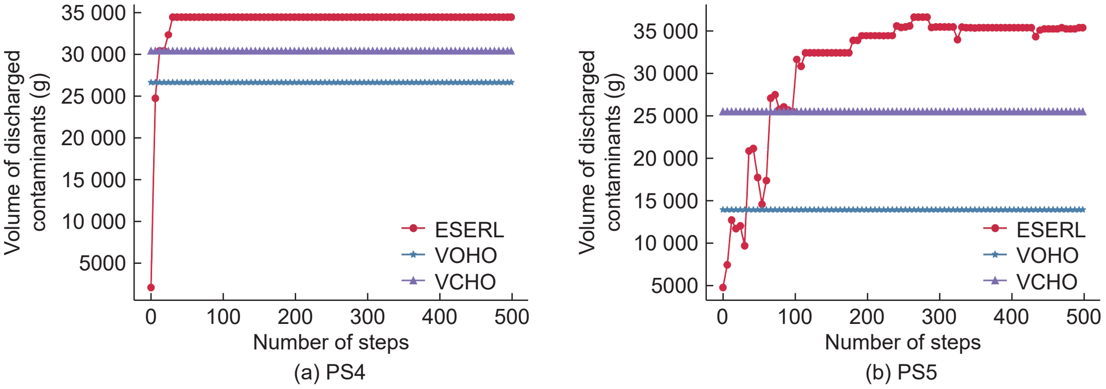


Fig. 7 Multiple pollution events PS4 and PS5.

The ESERL algorithm was also compared with the ESPPO algorithm. As can be seen in Fig. 9, the performance of ESERL algorithm is still better than ESPPO no matter single or multiple pollution events, while the computational overhead of the ESERL is only half of the ESPPO.

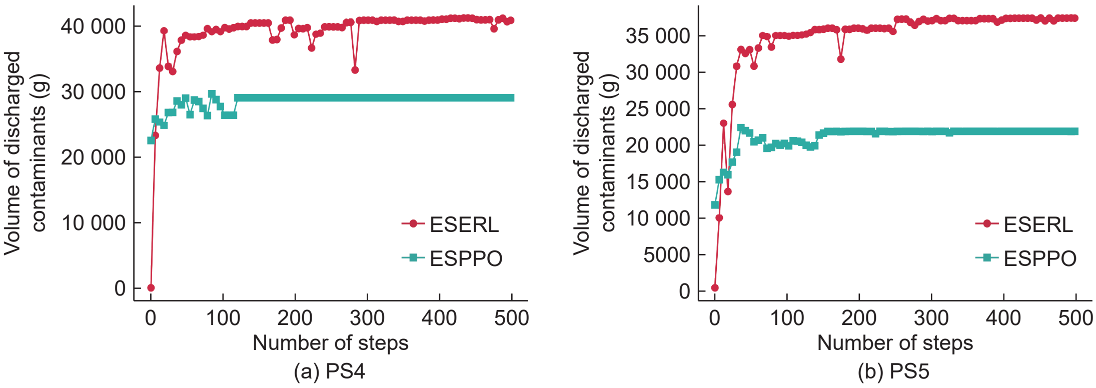
5.4 Control center scheduling process

In order to intuitively observe the effectiveness of the scheduling policy of the ESERL, we deployed the well-trained deep neural network on the control center and used the optimal policy to generate a series of actions according to the real-time sensor data. Table 2 shows



(a) PS4 (b) PS5

Fig. 8 Volume of discharged contaminants for multiple pollution events by ESERL, VOHO, and VCHO.



(a) PS4 (b) PS5

Fig. 9 Volume of discharged contaminants for multiple pollution events by ESERL and ESPPO.

Table 2 Status of sensors, valves, and hydrants for the contaminant event PS5.

Number of simulation steps	Contamination concentration (mg/L)				Operation action					
	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Valve 1	Valve 2	Valve 3	Hydrant 1	Hydrant 2	Hydrant 3
1	0	51.1459	0	0	0	0	0	0	0	0
2	0	168.4680	0	0	1	1	0	1	1	0
3	0	11.3612	0	0	1	1	0	1	1	1
4	1.9820	27.2015	422.1400	0	0	1	1	1	1	1
5	0	3.5965	1.8434	37.0935	0	1	0	0	1	1
6	0	1.6352	1.0588	90.4168	0	1	0	0	1	1
7	0.3989	0	0.2456	30.7884	1	1	0	1	1	1
8	0.3149	10.6553	0	48.0839	0	1	1	1	1	1
9	0	0.9179	0.2823	5.6702	1	1	0	1	1	1
10	0	0	0.9596	19.2035	1	0	0	1	1	1
11	0	0	0	6.3055	1	0	0	1	1	1
12	0	0	0	4.3463	1	0	0	1	0	1
13	0	0	0	0	0	0	0	0	0	0

the status of the sensors, valves, and hydrants for the contaminant event PS5.

It can be seen from Table 2 that 13 dispatch operations were carried out to discharge the contaminant. The reading data from sensor 2 decreased from 51.1459 mg/L to 0. After the contaminant event PS5, the hydrants were gradually turned on, because the goal was to maximize emissions. However, in the dispatching process, the hydrants were not always open, which effectively prevented the waste of unpolluted water. The valves were also not always closed, which prevented large regional water cuts and effectively reduced the impact of pollution events. After the pollution was eliminated, all the valves were turned on, the town resumed normal water supply, and all the fire hydrants were turned off to ensure that the water with normal water quality was not wasted. Therefore, it can be concluded that the emergency scheduling model based on ERL algorithm is feasible and effective for emergency handling of pollution emergencies.

6 Conclusion

In this paper, we studied the emergency scheduling optimization problem for a sudden drinking water pollution event. In order to solve the need for real-time control and quick response, we proposed the ESERL algorithm, which combines cross-entropy and DDPG. By offline training, a well-trained deep neural network was deployed on the control center to generate optimal policies in real time according to sensor information. Consequently, a series of actions were carried out to discharge the contamination as soon as possible. In order to evaluate the performance of the algorithm, a typical WDN with 97 nodes was employed. Simulations coupled with EPANET were carried out for single and multiple contamination events. The experimental results show that ESERL performed better than the other algorithms.

A large-scale WDN will be employed for simulation in future work; distributed DRL and EC will also be studied to solve other large-scale and real-time scheduling optimization problems.

Acknowledgment

This work was supported in part by the National Science Foundation of China (Nos. 62073300, U1911205, and 62076225).

References

[1] D. Hou, X. Ge, P. Huang, G. Zhang, and H. Loáiciga, A real-time, dynamic early-warning model based on uncertainty analysis and risk assessment for sudden water pollution accidents, *Environmental Science and Pollution Research*, vol. 21, no. 14, pp. 8878–8892, 2014.

[2] Y. Rui, D. Shen, S. Khalid, Z. Yang, and J. Wang, GIS-based emergency response system for sudden water pollution accidents, *Physics and Chemistry of the Earth, Parts A/B/C*, vol. 79, pp. 115–121, 2015.

[3] S. Rathi and R. Gupta, A simple sensor placement approach for regular monitoring and contamination detection in water distribution networks, *KSCE Journal of Civil Engineering*, vol. 20, no. 2, pp. 597–608, 2016.

[4] C. Hu, G. Ren, C. Liu, M. Li, and W. Jie, A spark-based genetic algorithm for sensor placement in large scale

drinking water distribution systems, *Cluster Computing*, vol. 20, no. 2, pp. 1089–1099, 2017.

[5] X. Yan, K. Yang, C. Hu, and W. Gong, Pollution source positioning in a water supply network based on expensive optimization, *Desalination and Water Treatment*, vol. 110, pp. 308–318, 2018.

[6] D. Hou, X. Song, G. Zhang, H. Zhang, and H. Loaiciga, An early warning and control system for urban, drinking water quality protection: China's experience, *Environmental Science and Pollution Research*, vol. 20, no. 7, pp. 4496–4508, 2013.

[7] H. Che, S. Liu, and K. Smith, Performance evaluation for a contamination detection method using multiple water quality sensors in an early warning system, *Water*, vol. 7, no. 4, pp. 1422–1436, 2015.

[8] T. Ren, S. Li, X. Zhang, and L. Liu, Maximum and minimum solutions for a nonlocal *p*-Laplacian fractional differential system from eco-economical processes, *Boundary Value Problems*, no. 1, p. 118, 2017.

[9] T. Ren, H. Xiao, Z. Zhou, X. Zhang, L. Xing, Z. Wang, and Y. Cui, The iterative scheme and the convergence analysis of unique solution for a singular fractional differential equation from the eco-economic complex system's co-evolution process, *Complexity*, no. 3, pp. 1–15, 2019.

[10] A. Preis and A. Ostfeld, Multiobjective contaminant response modeling for water distribution systems security, *Journal of Hydroinformatics*, vol. 10, no. 4, pp. 267–274, 2008.

[11] A. Afshar and E. Najafi, Consequence management of chemical intrusion in water distribution networks under inexact scenarios, *Journal of Hydroinformatics*, vol. 16, no. 1, pp. 178–188, 2014.

[12] L. Wang, Z. Pan, and J. Wang, A review of reinforcement learning based intelligent optimization for manufacturing scheduling, *Complex System Modeling and Simulation*, vol. 1, no. 4, pp. 257–270, 2021.

[13] L. Luo, N. Zhao, and G. Lodewijks, Scheduling storage process of shuttle-based storage and retrieval systems based on reinforcement learning, *Complex System Modeling and Simulation*, vol. 1, no. 2, pp. 131–144, 2021.

[14] M. Gavanelli, M. Nonato, A. Peano, S. Alvisi, and M. Franchini, Genetic algorithms for scheduling devices operation in a water distribution system in response to contamination events, in *Proc. 12th European Conference on Evolutionary Computation in Combinatorial Optimization*, Málaga, Spain, 2012, pp. 124–135.

[15] A. Rasekh and K. Brumbelow, A dynamic simulation–optimization model for adaptive management of urban water distribution system contamination threats, *Applied Soft Computing*, vol. 32, pp. 59–71, 2015.

[16] M. E. Shafiee and E. Z. Berglund, Real-time guidance for hydrant flushing using sensor-hydrant decision trees, *Journal of Water Resources Planning and Management*, vol. 141, no. 6, pp. 04014079-1-04014079-14, 2015.

[17] M. E. Shafiee and E. Z. Berglund, Complex adaptive systems framework to simulate the performance of hydrant flushing rules and broadcasts during a water distribution system contamination event, *Journal of Water Resources Planning and Management*, vol. 143, no. 4, p. 04017001, 2017.

[18] R. Schwartz, M. Housh, and A. Ostfeld, Limited multistage stochastic programming for water distribution systems optimal operation, *Journal of Water Resources Planning and Management*, vol. 142, no. 10, pp. 1–6, 2016.

[19] P. Khatavkar and L. W. Mays, Optimization-simulation model for real-time pump and valve operation of water distribution systems under critical conditions, *Urban Water Journal*, vol. 16, no. 1, pp. 45–55, 2019.

[20] C. Hu, J. Cai, D. Zeng, X. Yan, W. Gong, and L. Wang, Deep reinforcement learning based valve scheduling for pollution isolation in water distribution network, *Mathematical Biosciences and Engineering*, vol. 17, no. 1, pp. 105–121, 2020.

[21] W. Gong, Z. Liao, X. Mi, L. Wang, and Y. Guo, Nonlinear equations solving with intelligent optimization algorithms: A survey, *Complex System Modeling and Simulation*, vol. 1, no. 1, pp. 15–32, 2021.

[22] S. Khadka, S. Majumdar, T. E. Nassar, Z. Dwiel, E. Tumer, S. Miret, Y. Liu, and K. Tumer, Collaborative evolutionary reinforcement learning, in *Proc. 36th International Conference on Machine Learning*, Long Beach, CA, USA, 2019, pp. 3341–3350.

[23] S. Lü, S. Han, W. Zhou, and J. Zhang, Recruitment-imitation mechanism for evolutionary reinforcement learning, *Information Sciences*, vol. 553, pp. 172–188, 2021.

[24] A. Gupta, S. Savarese, S. Ganguli, and F. -F. Li, Embodied intelligence via learning and evolution, *Nature Communications*, vol. 12, pp. 1–12, 2021.

[25] B. Wu, H. -J. Jiang, C. Wang, and M. Dong, Knowledge and behavior-driven fruit fly optimization algorithm for field service scheduling problem with customer satisfaction, *Complexity*, no. 11, pp. 1–14, 2021.

[26] B. Wu, T. Yuan, Y. Qi, and M. Dong, Public opinion dissemination with incomplete information on social network: A study based on the infectious diseases model and game theory, *Complex System Modeling and Simulation*, vol. 1, no. 2, pp. 109–121, 2021.

[27] X. Han, Y. Han, Q. Chen, J. Li, H. Sang, Y. Liu, Q. Pan, and Y. Nojima, Distributed flow shop scheduling with sequence-dependent setup times using an improved iterated greedy algorithm, *Complex System Modeling and Simulation*, vol. 1, no. 3, pp. 198–217, 2021.

[28] W. Zhang, W. Hou, C. Li, W. Yang, and M. Gen, Multidirection update-based multiobjective particle swarm optimization for mixed no-idle flow-shop scheduling problem, *Complex System Modeling and Simulation*, vol. 1, no. 3, pp. 176–197, 2021.

[29] X. Wu, Z. Cao, and S. Wu, Real-time hybrid flow shop scheduling approach in smart manufacturing environment, *Complex System Modeling and Simulation*, vol. 1, no. 4, pp. 335–350, 2021.

[30] Y. Wei, X. Jiang, P. Hao, and K. Gu, Multi-agent co-

evolutionary scheduling approach based on genetic reinforcement learning, in *Proc. 2009 Fifth International Conference on Natural Computation*, Tianjian, China, 2009, pp. 573–577.

[31] D. Zeng, J. Zhan, W. Peng, and Z. Zeng, Evolutionary job scheduling with optimized population by deep reinforcement learning, *Engineering Optimization*, doi: 10.1080/0305215X.2021.2013479.

[32] A. Kamiya, H. Kimura, M. Yamamura, and S. Kobayashi, Power plant start-up scheduling: A reinforcement learning approach combined with evolutionary computation, *Journal of Intelligent & Fuzzy Systems*, vol. 6, no. 1, pp. 99–115, 1998.

[33] J. Dou, C. Chen, and P. Yang, Genetic scheduling and reinforcement learning in multirobot systems for intelligent warehouses, *Mathematical Problems in Engineering*, vol. 2015, pp. 1–10, 2015.

[34] E. K. Ahmed, Z. Li, B. Veeravalli, and S. Ren, Reinforcement learning-enabled genetic algorithm for school bus scheduling, *Journal of Intelligent Transportation Systems*, vol. 26, no. 3, pp. 269–283, 2020.

[35] Z. Xu, G. Han, L. Liu, M. Martínez-García, and Z. Wang, Multi-energy scheduling of an industrial integrated energy system by reinforcement learning-based differential evolution, *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 3, pp. 1077–1090, 2021.

[36] Z. Li, X. Wei, X. Jiang, and Y. Pang, A kind of reinforcement learning to improve genetic algorithm for multiagent task scheduling, *Mathematical Problems in Engineering*, nos. 43–45, pp. 1–12, 2021.

[37] L. A. Rossman, Epanet 2: Users manual, https://www.epa.gov/water-research/epanet, 2022.

[38] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT press, 2018.

[39] M. V. Otterlo and M. Wiering, Reinforcement learning and Markov decision processes, in *Reinforcement Learning*, M. Wiering and M. Otterlo, eds. Berlin, Germany: Springer, 2012, pp. 3–42.

[40] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv: 1412.6980, 2014.

Chengyu Hu received the BEng and MS degrees in automation and control from Wuhan University of Technology in 2000 and 2003, respectively, and the PhD degree in automation control from Huazhong University of Science and Technology in 2010. He is currently a professor and vice dean of the School of Computer Science, China University of Geosciences, Wuhan, China. His current research interests include evolutionary algorithm, swarm intelligence, and reinforcement learning.

Rui Qiao is currently pursuing the MS degree at the School of Computer Science, China University of Geosciences, Wuhan, China. Her research interests include reinforcement learning and evolutionary computation.

Zhe Zhang is currently pursuing the BS degree at the School of Computer Science, China University of Geosciences, Wuhan, China. His research interests include reinforcement learning and evolutionary computation.

Xuesong Yan received the BEng degree in computer science and technology and the MEng degree in computer application from China University of Geosciences in 2000 and 2003, respectively. He received the PhD degree in computer software and theory from Wuhan University in 2006. He is currently an associate professor in the School of Computer Science, China University of Geosciences, Wuhan, China and was as a visiting scholar with the Department of Computer Science, University of Central Arkansas, Conway, AR, USA. His research interests include evolutionary computation, data mining, and computer application.

Ming Li is currently a professor and chair in the Department of Computer Science, California State University, Fresno, CA, USA. Prior to that, he was an assistant professor from August 2006 to 2012. He received the MS and PhD degrees in computer science from University of Texas at Dallas in 2001 and 2006, respectively. His research interests include QoS strategies for IEEE 802.11 wireless LANs, mobile ad-hoc networks, heterogeneous wired and wireless networks, multimedia streaming over wireless networks, body area networks, and robot swarm communications.