Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 5s (2025)

Real Time Monitoring and Control of Pump by Deep Reinforcement Learning

Dr. Paromita Goswami [1], R Shalini [2], Dr.M.V.Rajesh [3], Rashmi Hegde [4], Dr.S.Suma Christal Mary Sundararajan [5], Dr. Bramah Hazela [6]

[1] Department of Computer Engineering and Applications, GLA University, Mathura, India. mcanbu.16@gmail.com

[2] Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Green Fileds, Vaddeswaram, India. spshalinikaran@kluniversity.in

[3] Department of IT, Aditya University, Surampalem, India, Rajesh.masina@adityauniversity.in

[4] Nitte (Deemed to be University), NMAM Institute of Technology, Nitte, Karkala, India. rashmihegde@nitte.edu.in

[5] Department of Information Technology, Panimalar Engineering College,Poonamalle, Chennai, India. sumasheyalin@gmail.com

[6] Amity School of Engineering & Technology, Amity University UTTAR PRADESH, LUCKNOW CAMPUS, LUCKNOW, India. bramahhazela77@gmail.com

Article History:

Received: *12-10-2024*

Revised: *28-11-2024*

Accepted: *09-12-2024*

Abstract:

Flood prevention is aided by rainwater pumping stations situated close to cities or agricultural regions. These stations activate a suitable number of pumps, each with a different capacity, in response to the actual amount of rainwater that is falling. Unfortunately, effective control is typically lacking when rule-based pump operations are used in isolation to monitor basin water levels. Diminishing the number of switches that is on or off for the pumps at rainfall stations is just as important as keeping the maximum water level low to avoid floods. Pump switch frequency reduction reduces maintenance expenses by lowering the chance of mechanical failure. This paper presents a method for operating rainwater pumping stations in real-time utilizing Deep Reinforcement Learning (DRL). The goal is to meet all of these operational requirements at the same time, using only data that is presently noticeable like inflow, rainfall, place of storage amount, water level basin and outflow. It was trained using simulated rainfall data created using the Huff technique with different return periods and durations. Experiments were carried out using the Storm Water Management Model (SWMM), which was set up to mimic the rainwater pumping station. Next, the suggested DRL model's efficiency was contrasted with the station's present rule-based pump operation.

Keywords: Rainwater Pumping Station, Floods, DRL, Rainfall, SWMM.

1. Introduction

Rainfall in metropolitan centres and adjacent regions follows intricate drainage patterns before reaching rivers [1]. However, direct input into rivers after a quick rain might flood nearby regions, inflicting major damage [2]. Climate change has increased the frequency of severe rainfall, leading to significant flood damage that burdens national economies [3]. The installation and operation of rainwater pumping stations serves as a buffer to reduce the likelihood of flooding in regions that are next to rivers [4]. By discharging collected rainfall into retention basins, these pumping stations successfully avoid river overflow and flood damage. The capabilities of the pumps vary according to their size [5-7]. In order to activate or deactivate the pumps at rainwater pumping stations [8], the majority of these systems employ a rule-based approach that takes into account the retention basin's water level [9]. Localized heavy rains are becoming more common as a result of climate change, and basic operations based on rule that depend on levels of water or physical management built on operator expertise aren't enough to handle the sudden changes in water intake [10-12]. Methods for improving the efficiency of urban drainage systems have been the subject of several investigations.

A heuristic control system based on genetic algorithms was suggested by the researchers as a means to avoid sewage overflow in real-time [13]. Although heuristic algorithms, dynamic programming, and linear programming are great at discovering optimal solutions, they become computationally intensive to recalculate for each scenario, which limits their use in real-time control as the quantity of variable grows [14-16]. Other methods for control in actual time include heuristic algorithms and dynamic programming. One such method is Model Predictive Control (MPC) [17]. MPC is able to foretell how a system will act in the future and use that information to choose the best control inputs to provide it in real time.

Extensive research has been conducted across engineering domains in response to recent revolutionary advances in deep learning and artificial intelligence [18]. There has been a surge in research into the potential uses of deep learning in urban water management, with potential applications ranging from system operation and prediction to asset evaluation [19], planning and maintenance, anomaly detection, and asset assessment. Based on the data format, availability of labels, and learning technique [2], machine learning methods are often classified as either unsupervised, supervised, or reinforcement learning. As a result, the application's unique traits and goals must be carefully considered while choosing a deep learning model [21]. Because supervised learning models need predetermined right answers for all situations, they are not applicable to the functioning of rainwater pumping stations. when trying to determine the appropriate pump activation strategy based on variables like rainfall and the retention of basin water levels [22-24]. Unsupervised learning, on the other hand, is more suited to tasks like distributing data or feature analysis and is hence not a good fit for this job. The most suitable method for optimizing pump operations in real-time, when the model has to accomplish goals like lowering retaining basin water levels in accordance with changing circumstances, is Deep Reinforcement Learning (DRL) [25]. This is because DRL is able to use reward mechanisms. As a result, research into DRL's potential for controlling storm drainage systems for water in real time has just started.

This paper addresses a multiple objective optimizing issue for water pump station functioning under heavy rains. The objective is to reduce water levels and maintenance expenses for pump combination modifications. The issue of multiple-objective optimization is defined, reward functions are proposed, and an improved version of Deep Q-Network, the Dual Deep Q-Network (DDQN), is introduced. There is a dearth of literature on this subject and solutions based on DRL. Some examples of DQN's practical applications include Atari video games, robotic control, and water management systems. DQN is a reinforcement-learning approach that enables agents to choose the optimal behaviours within a given environment. When it comes to Q-learning algorithms, DDQN stands out for its capacity to tackle the prevalent problem of overestimation bias. Stable value estimations are provided by DDQN's independent Q-network as well as Target network architectures, and data efficiency is enhanced by the use of experience replay.

This study utilizes the Gated Recurrent Unit (GRU), a model of deep learning well-suited for data with time series, as the agent, in contrast to most previous studies that use basic deep neural networks without considering the temporal character of data from pump station operations. Furthermore, researchers created synthetic severe rainfall data with various periods of return and rainfall periods using the Huff approach. This will allow for proactive learning and responsiveness to climate change-driven extreme rainfall. By simulating the pumping station's retention basin and pumping configuration with the synthetic rainfall data supplied by the Storm Water Management Model (SWMM), they were able to confirm the method's practical applicability and performance.

This study primarily contributes to the following areas:

- Reducing maintenance expenses caused by pump switching and minimizing retention basin water levels are both taken into account in our definition of a multiple objective optimization problem. This approach was to create a DDQN method to deal with this.
- Integrating a time-series-aware agents and devising an effective incentive function allowed us to get experimental findings that show the proposed model may minimize maintenance costs while maintaining lower levels of water in the retained water basins compared to rule-based pump strategies.
- In order to shed light on possible operational improvements, created an exact model of the pumping station's pumping and retention basin setting and compared the DDQN-based technique with the rule-based one that is presently in use.
- As a consequence of climate change and fast urbanization, they were able to create a system of controls that can successfully react to changes in rainfall. Instead of regular rainfall, this structure was tested in simulated extreme rainfall situations to make sure it would hold up well in harsh weather.

2. Methods and Materials

2.1 Pumping Station Modelling

A simulated ecosystem was created using the Rainwater Pumping Stations in Tamil Nadu. The Pump Station was chosen for this research because of its location in a flood-prone region, making it an ideal site for testing options for flood control. The correlation between the elevation (measured in metres higher than sea level) and the volume of storage capability of the retention reservoir at the Pumping Station, as determined by real measurements. The pumping station has a maximum water level of 10 meters. Equation (1) represents the connection between storage volume and water level for various ranges.

$$h = H_{floor}(v) + \frac{H_{ceil}(v) - H_{floor}(v)}{v_{ceil}(v) - v_{floor}(v)}\left(v - v_{floor}(v)\right) \quad (1)$$

2.2 Rainfall Data

A frequently used method in the construction of rainwater pumping stations, frequency-based probability rainfall was utilized to create the rainfall data used in the trials. Sum of precipitation, length of precipitation, and return period (frequency) make up rainfall probability. The chance of rainfall was distributed over time using Huff's dimensionless cumulative curve approach. This approach, which makes use of rainfall data from the past, is well-known for being dependable and has many real-world uses, including riverbed and smaller stream design criteria. In addition to producing accurate rain designs that represent intensity fluctuations in actual rainfall proceedings, this approach may efficiently generate these patterns, which in turn reduces simulation time and expenses. As a result, this method is also often used in civil engineering and hydrology to evaluate the stability of structures in the face of heavy precipitation. Extreme rainfall patterns were reflected by using probable the greatest rainfall according to periods of return and historical information from the pumping station location. A statistical optimization approach like the Least Squares approach calculates weights for four quantiles. Adjusting weights to simulate various rainfall occurrences allows for accurate modelling of pump station conditions. Rainfall samples were collected over return periods of 10, 20, 30, 50, 80, and 100 years to address the significance of controlling climate-driven severe rainfall occurrences. The samples were categorized according to nine different rainfall durations: 60, 120, 180, 240, 360, 540, 720, 1080, and 1440 minutes.

Figure 1 shows the output of running the SWMM with a 30-years period of return and a 60-minute length rainfall data sample. Figure 1 (a) displays the total rainfall over a period of time, whereas Figure 1(b) shows the rate of inflow into the retained water basin at the same time.

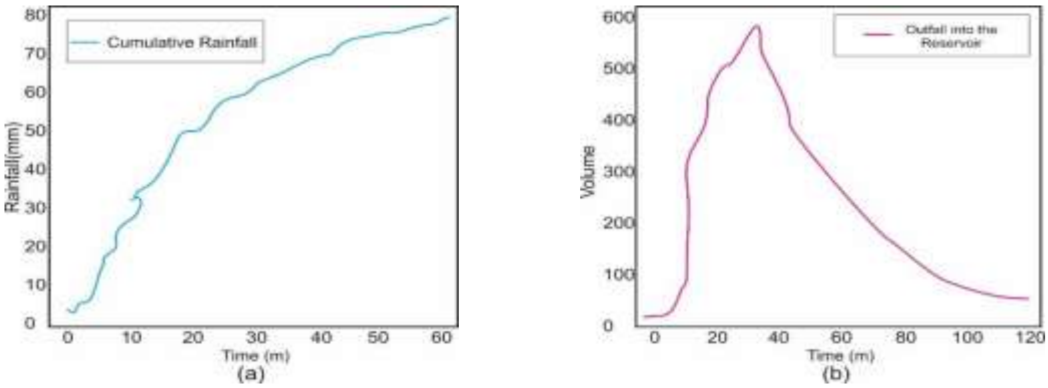

Figure 1 (a) Total Rainfall Over a Period of Time, (b) Rate of Inflow into the Retained Water

2.3 Formulating Problems

A multiple objectives pump combination optimization issue is what the rainwater pump station technology is trying to solve by reducing the quantity of pump switches and diminishing the highest level of water in the retaining basin.

2.3.1 The Problem of Selecting a Multi-Objective Pump Combination

The purpose is to determine the best arrangement of pump combinations to use during a specified time period T given a list of n pumps with varying capacities and a collection of k possible combinations. The goal function should be maximized throughout unit time intervals by this ideal sequence, in order to accomplish optimal operation.

The pump combination functional objective $p_,$ has two goals:

1) Using as few pump switches as possible and

2) maintaining the retention basin water level as low as possible.

In order to prevent the retention basin from filling up to its full capacity, it is necessary to determine a pump sequence that will result in the lowest feasible maximum water level across all unit periods for the whole simulation period T (Equation 2).

$$\text{argmin F}(\pi_i), \pi_i \epsilon \pi^T \qquad (2)$$

$$\text{H}(\pi_i) = max h_{\pi_i}(t), 1 \leq t \leq T$$

On this occasion, hπi (t) stands for the detention basin water level at time t. Reduce the amount of time spent moving between pump combinations at each time unit to keep wear to a minimum. The maintenance cost is the amount that has to be paid to make these modifications. A proxy for the number of pumps switched on or off is used since exactly specifying the maintenance cost might be problematic in reality. Equation (3) shows how to define the minimizing of pump switching.

$$\text{argmin F}(\pi_i), \pi_i \epsilon \pi^T \qquad (3)$$

$$\text{F}(\pi_i) = \sum_{t=1}^{T} \text{f}\pi_i\,(t), 1 \leq t \leq T$$

As of time t, the number of switches that turn the pump on or off is denoted by $\text{f}\pi_i(t)$. For this reason, Equation (4) shows the objective function that is obtained by integrating the two optimization goals with their corresponding weights w_1 and w_2. Finding the ideal activity sequence $\pi^{T\times}$ that fulfils Equation (4) is the last objective.

$$argmin\ \text{O}(\pi_i) = w_1 \times H(\pi_i) + w_2 \times F(\pi_i) + w_3 \times E(\pi_i), \pi_i \in \pi^T \quad (4)$$

In the case of fluctuating pump capacity, the dimension of the result space for potential sequences of pump combinations π over time period T is $2\pi^T$. Finding the best solution becomes more difficult as the time T or number of pumps n rises, as the solution space expands exponentially. This image shows all the potential combinations of pumps throughout time T, but it doesn't take into consideration things like rainfall or the water level in the retention basin. When these things are included, the true solution space is much wider.

2.4 Pumping System Double-Deep Q-Network

2.4.1 The Reinforcement Learning

In the reinforcement learning, a computer learns by observing its surroundings and choosing its behaviours in a way that maximizes cumulative rewards. A 5-tuple (S, t, A, R, and Pr) represents a discrete-time stochastic process, and it may be used to simulate reinforcement learning. This process is called a Markov Decision Process (MDP). In this context, t stands for time, S for states, R for rewards, A for actions, and Pr for the likelihood of transitioning between states.

At time step t, the decision-making agent takes into account the present state s and chooses an action a from set A. After step t+1, the system changes to states′ depending on the action a and current state ‘s’. This change in state occurs after the probability of the transition, or the likelihood of the change in state, $pr(s, s') = pr(s_{t+1} = s'|s_t = s)$. A reward or punishment is given to the agent whenever the status changes. Put simply, the agent decides what to do in the environment, which changes the state and gives rise to rewards. In reinforcement learning, the goal of the agent's interactions with its environment is to maximize cumulative rewards.

2.4.2 Double Deep Q Net

Combining Q-learning with deep learning, the DQN is a significant algorithm in reinforcement learning. Agents may learn to perform optimally in many situations with the help of the DQN. To avoid relying on a table, the DQN proposes employing a deep learning method, such DNN, to estimate the Q-values. An agent learns the value of an action in a given state by Q-learning, a basic reinforcement learning method. The projected cumulative future reward is represented by this value, which is known as the Q-value. Using the Bellman Equation, the agent iteratively updates the Q-values in Q-learning.

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left(r + \gamma max Q(s',a')max - Q(s,a)\right) \qquad (5)$$

Here, s represents the present state and s′ the future state. Given the present state, an is an action that is done, r is the right away reward that follows activity a, γ is a factor of discount that is applied to future incentives, as well as α is the learning rate. Updates on Q-learning the agent learns a strategy that optimizes the long-term reward based on the Q values depending on the reward level r and the highest Q value in the following state ‘s′. Using the present state as input, the DQN's deep learning model generates Q-values for every conceivable action. Training this network to provide good approximations of the Q-values is the objective.

An enhanced variant of the conventional DQN used in RL, a DDQN yields better results. The overestimation of action values that DQNs are prone to causes instability and less-than-ideal policies during training; this is one of their major drawbacks that it solves. The goal of a typical DQN is for the agent to achieve the greatest cumulative reward as it moves from one state to another by making decisions based on a learnt Q-function, which assesses the anticipated return for a specific action. To achieve a balance between exploration and exploitation, the agent employs an epsilon greedy strategy, and a deep neural network is used to approximate the Q-function.

2.4.3 Gated Recurrent Unit

The Gated Recurrent Unit is a Recurrent Neural Network (RNN) structure used in deep learning, especially for sequence data. GRUs were presented as a simpler way to replace LSTM networks, another prominent RNN. GRUs aim to address the vanishing gradient issue in standard RNNs while learning long-term dependencies. GRUs use gating mechanisms to govern information flow and retain relevance over lengthy sequences. The reset gate and the update gate are the two primary gates of a GRU. The amount of information from the previous state that is erased by the reset gate and sent to the subsequent step is determined by the update gate, respectively.

2.4.4 Model Configuration

In the DDQN method, the decision-making agent first has a fully connected layer of output with six nodes, then three GRU layers with 32 nodes each. In the output layer, there are six possible combinations of pumps represented by the six nodes. The DDQN and GRU models' structure is determined by hyperparameters, therefore we played around with different settings for them. Since these adjustments did not lead to noticeable performance drops, it was decided with the bare minimum of model setups. The adoption of the replay technique and an epsilon-greedy approach for action selection ensured training stability.

2.4.5 States

Retention basin input, basin volume of water, basin level of water, and outflow make up the five dimensions of the five-dimensional vector that the model uses to make action choices in the pumping station environment based on the state information. The training and inference procedures of the model are fed by this 5-dimensional vector. The inflow values were acquired by conducting SWMM testing at 2 min intervals, while the rainfall measurements were made using simulated rainfall data. At 2-minute intervals, the volume, level, and outflow of water were calculated from the operating results of the chosen pump combinations.

2.4.6 Actions

The GRU agent in the DRL model takes environmental state information into account while deciding one of many potential actions to do in order to maximize rewards. According to Section 2.1, the present Pumping Station controls the retention basin's water level using a total of six distinct pump combinations, including three 100 m3/min pumps and two 170 m3/min pumps. In order to make a fair comparison with the current rule-based operating system and to make it more realistic for real-world operations, the agent in the proposed DRL model is likewise limited to these six combinations.

2.4.7 Reward Function

An essential part of the DDQN, the function of reward tells the agent how well or poorly it has done in relation to its surroundings. By directing the agent to its objective, it has an immediate impact on learning. In order to accomplish the two goals stated in the issue statement, they created two halves of the reward function.

3. Results and Discussion

Training data consisted of around 2250 of the 3000 rainfall samples, whereas test data included about 25% of the total, or 750 samples. To mimic scenario-based learning and its ability to generate new situations in real time, they utilized all 2250 rainfall situations in the training dataset just once. As a result, unlike when using the same data again, they did not use a technique to determine the ending point by using distinct validation data. Both the training and test datasets were given an equal amount of rainfall data, which was obtained from 5 period of return probabilities and nine durations. That is why there are a total of twelve cases for every possible return time and length in the test dataset. Presented below are the average experimental findings from these twelve test cases.

Figure 2 displays the outcomes of the simulations conducted using a 30-year return period test data sample and a 60-minute rainfall duration scenario as inputs to the trained DDQN model. The time-lapse input data has a sequence length of 3 in this case. Figure 2a shows the volume going into the water reservoir every 2 minutes and the volume going out, according to the model-selected pump operation. The water level in the basin was monitored at 2-minute intervals during the experiment, as shown in Figure 2b.

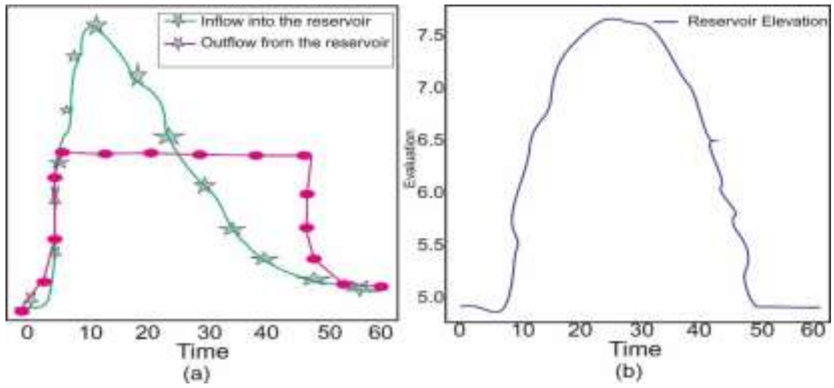

Figure 2 (a) Volume Going into the Water Reservoir, (b) Intervals for Basin Monitoring

Applying the same situation to the rule-driven pump operating mechanism presently utilized at the water pumping station, Figure 3 (a, b) depicts both the inflow and the outflow of the basin.

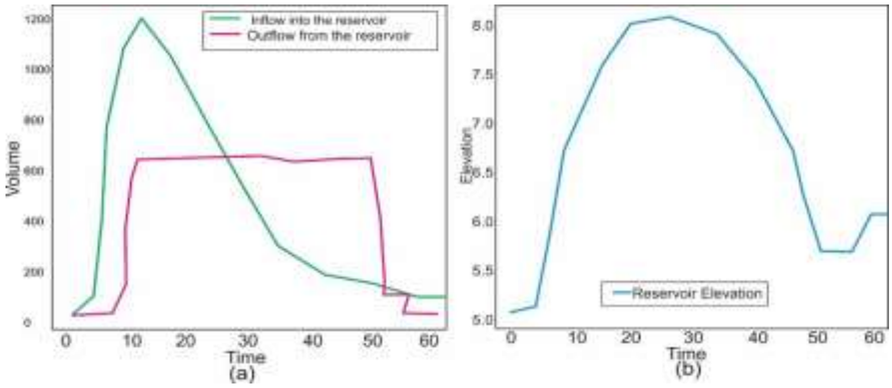

Figure 3 (a) Inflow (b) outflow of the Basin

The rule-based method, a DDQN method that aimed only at reducing the maximum water level, and a DDQN method that considered both the highest possible water level and pump switching minimization were tested with test results for every one of the six period of return probabilities across nine durations. w_e and w_p , the reward function's weights, were subject to the same rigorous testing with a variety of values as the other variables utilized in the DDQN as well as GRU methods. The optimal values for we and wp were 2 and 1, correspondingly.

The experimental findings for the 30-years period of return probability are shown in Figure 4, which cover all rainfall periods. Figure 4a shows that compared to the rule-based model, both DDQN models were able to attain a smaller maximum water level. The DDQN structures constantly chose pumping combinations with maximum capacity, thus even in circumstances of very long rainfall periods, the maximum water level remained much lower than in cases of shorter durations. Furthermore, the two DDQN frameworks performed similarly up to the maximum water level.

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 5s (2025)

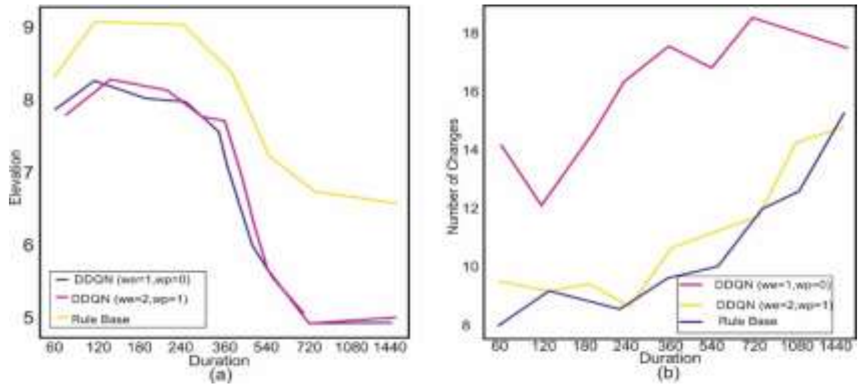

Figure 4 (a) Compared to the Rule-based Method, (b) Frequency of Pumping Switches

Figure 4b shows that with regard to the frequency of pumping switches, the DDQN model with switch minimization maintained the number of changes low, much as the rule-based method, while the approach without switch minimization had an elevated frequency of pump switches. Thus, it is possible to keep the water level low without significantly raising the quantity of pump adjustments when switch minimization is included in the objective function. However, when rainfall lasts for a long time, more often than not, it requires pump combinations with large capacities, which means to turn on the pumps more often.

The length of the sequence indicates how long the observed data was that the model was fed. When the sequence length is 3, for example, the model trains and infers using state data from three separate 2-minute periods. When the DDQN model alone considers the minimization of the maximum water level, Figure 5a, c demonstrates variations in the number of pump switches and the maximum water level for various sequence durations. Considering both the maximum water level (w_e) and the pump switch (w_p) as targets has an effect on performance, as shown in Figure 5b, d.

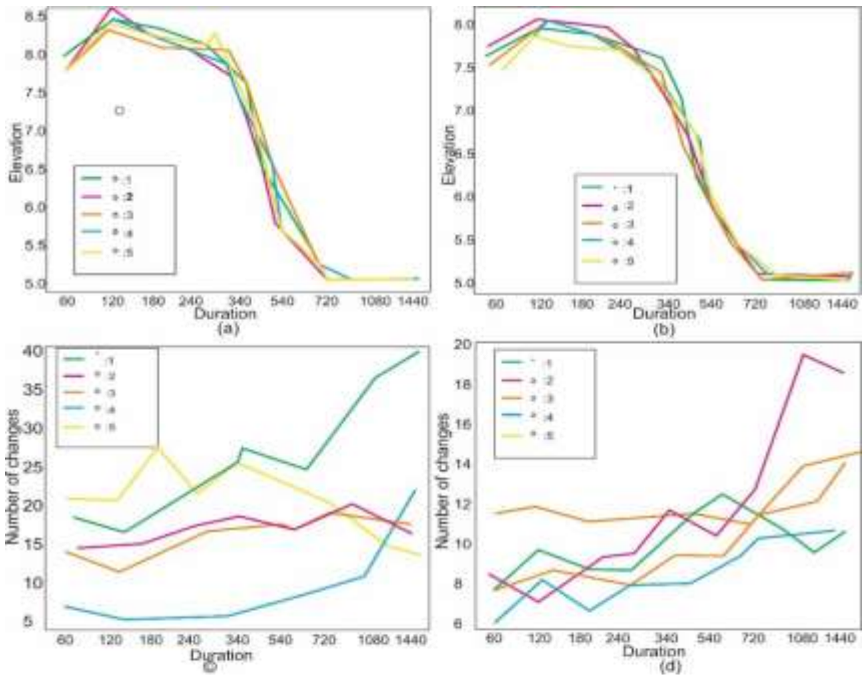

Figure 5 (a, c) Variations in the Number of Pump Switches and Maximum Water Level (b, d) Both the maximum water level

Results for both models show that maximum water level is unaffected by sequence length. The DDQN model that takes into account both goals have a less effect on the number of pump changes compared to the method that simply takes into account maximal water level minimization. In general, longer sequences aid in reducing pump switches more efficiently than shorter ones, while there is no obvious association between the length of the sequence and the amount of switching. The DDQN method that uses w_p in conjunction with

us reaches its minimal switch count at 5 sequence lengths, while the model that uses, alone reach its minimum at 4 sequence lengths.

The results show that choosing the right hyperparameters, especially the weights, is still a tough job that needs a lot of trial and error. The efficiency of the model is greatly affected by the weights in the function of reward and the length of the sequence of state observations; this highlights the need for more study on appropriate techniques of value selection. A relatively modest factor of discount for future prizes helped minimize the maximum water level, which was an additional benefit. Since this research only uses observable data from the past and present about rainfall, it follows that prioritizing immediate water level management over future levels is more beneficial for this purpose. Alternate choices for the discount factor may provide more accurate predictions of future rainfall if they were included. To improve the performance and practicality of the schemes, further research is needed in these areas.

The actual cost of maintenance might be hard to predict, thus the reward function substituted the quantity of pump switches for it. Making a cost estimating model that more faithfully represents actual maintenance expenses is essential for developing a field-deployable model. To account for actual needs, the training model has to include not just the maintenance expenses of pump switching but also a number of other practical requirements. Important considerations include, for instance, synchronizing several pumping stations to control flood hazards and reducing the amount of electricity needed to operate pumps. As a result of these demands, the pump operating issue is likely to become more complicated, calling for new ways of solving it. Nevertheless, no matter how complicated the situation becomes, the DDQN's design permits modifications to the status and action spaces according to the unique pump station environment and structure. To further accommodate growing system complexity, hyperparameters like the model's layer and node counts may be adjusted.

However, reliable real-time input data, such as exact rainfall and water retention basin inflow/outflow measurements, is crucial for putting a DRL based operational technique into practice, therefore supporting equipment like sensors is an important consideration. It is also crucial to train and test using real data, create models that are highly congruent with field settings, and modify the system to account for hardware restrictions in the field. Creating multiple objectives solutions to coordinate many pumping stations in the same area, each working under different circumstances, to jointly control flood risk is another topic that needs additional investigation. Possible future directions for research include developing novel training techniques to improve performance and conducting a thorough evaluation of the performance of different DRL methods that are relevant to pump station operations.

4. Conclusion

This research established a model for automating pumping operations at a water pump station in Tamil Nadu using a deep reinforcement learning technique called DDQN. To train and test the model using the created synthetic rainfall, we used the SWMM to simulate the basin and pump system. The Huff technique was used to generate rainfall data, which included five period of return probabilities and nine durations. The suggested approach was compared to the present rule-based pump operating technique at the pumping station to verify its performance. Compared to the rule-based system, the DDQN model with both reward functions showed 15% improvement in average maximal level of water and 6% in average pumping changes. This work is the first to use deep reinforcement learning to pump operation in a water pumping station. The use of a GRU structure as the agent proved that this method improved model performance and learning efficacy. GRUs are ideal for time-series analysis on visible sequential state data. The model was able to successfully avoid basin overflow without adding to the maintenance expenses linked with pump switching by creating a multiple objectives reward function that aimed to minimize both the maximum water level and the number of switches used to turn them on and off. This method proved to be more effective than rule-based procedures in preventing floods in experimental studies. Nevertheless, a more comprehensive cost model that takes into consideration pump maintenance costs beyond maximum water levels is required for a more accurate reflection of real-world situations. Intelligent pumps that are well-suited for use in the field might be developed with the help of this improved cost model if its impacts are analysed after being included into the reward function.

Reference:

[1] Zhang, Mofan, et al. "Evaluation of uncertain signals’ impact on deep reinforcement learning-based real-time control strategy of urban drainage systems." Journal of Environmental Management 324 (2022): 116448.

[2] Shao, Kai, et al. "Efficient Pump Scheduling for Large-Scale Multiproduct Pipelines Using Deep Reinforcement Learning." International Journal of Adaptive Control and Signal Processing (2024).

[3] Zhang, Ruichao, Dechun Chen, and Liangfei Xiao. "Intelligent Method to optimize the frequency modulation for beam pumping system based on deep reinforcement learning." ACS omega 8.10 (2023): 9475-9485.

[4] Patel, Harsh, et al. "Hybrid Reinforcement Learning for Optimizing Pump Sustainability in Real-World Water Distribution Networks." arXiv preprint arXiv:2310.09412 (2023).

[5] Gan, Tian, et al. "Research on low-energy consumption automatic real-time regulation of cascade gates and pumps in open-canal based on reinforcement learning." Journal of Hydroinformatics (2024): jh2024020.

[6] Xu, Jiahui, et al. "Zone scheduling optimization of pumps in water distribution networks with deep reinforcement learning and knowledge-assisted learning." Soft Computing 25 (2021): 14757-14767.

[7] Kumar, T. Rajasanthosh, G. Laxmaiah, and S. Solomon Raj. "A Framework of Intelligent Manufacturing Process by Integrating Various Function." AI-Driven IoT Systems for Industry 4.0. CRC Press 241-254.

[8] Xu, Jin, Han Li, and Qingxin Zhang. "Adaptive control for circulating cooling water system using deep reinforcement learning." Plos one 19.7 (2024): e0307767.

[9] Mewada, Shivlal, et al. "Smart diagnostic expert system for defect in forging process by using machine learning process." Journal of Nanomaterials 2022.1 (2022): 2567194.

[10] Rodrigues, Henrique Donãncio Nunes. Deep reinforcement learning through imitation learning and curriculum learning: application to pump scheduling in water distribution networks. Diss. Normandie Université, 2023.

[11] Bowes, Benjamin D., et al. "Reinforcement learning-based real-time control of coastal urban stormwater systems to mitigate flooding and improve water quality." Environmental Science: Water Research & Technology 8.10 (2022): 2065-2086.

[12] Zhang, Zhiyu, Wenchong Tian, and Zhenliang Liao. "Towards coordinated and robust real-time control: a decentralized approach for combined sewer overflow and urban flooding reduction based on multi-agent reinforcement learning." Water Research 229 (2023): 119498.

[13] Golabi, Arash, et al. "Optimal operation of reverse osmosis desalination process with deep reinforcement learning methods." Applied Intelligence (2024): 1-21.

[14] Nguyen, Khang HN, et al. "Reinforcement Learning-Based Control Sequence Optimization for Advanced Reactors." Journal of Nuclear Engineering 5.3 (2024): 209-225.

[15] SANJANA, KB SAI, K. SAI MOUNIKA, SANTHOSH KUMAR T RAJA, and DV SRIKANTH. "THERMAL ANALYSIS OF ADVANCED IC ENGINE CYLINDER."

[16] Wang, Shangting, et al. "Aortic Pressure Control Based on Deep Reinforcement Learning for Ex Vivo Heart Perfusion." Applied Sciences 14.19 (2024): 8735.

[17] Han, Gwangwoo, et al. "Data-driven heat pump operation strategy using rainbow deep reinforcement learning for significant reduction of electricity cost." Energy 270 (2023): 126913.

[18] Salaorni, Davide. "Optimal real-time control of a water distribution system undergoing cyber-attacks: a reinforcement learning." (2020).

[19] Hu, Chengyu, et al. "multi-objective deep reinforcement learning for emergency scheduling in a water distribution network." Memetic Computing 14.2 (2022): 211-223.

[20] Abdalla, Ramez, et al. "Machine learning approach for predictive maintenance of the electrical submersible pumps (ESPS)." ACS omega 7.21 (2022): 17641-17651.

[21] Fu, Qiming, et al. "Predictive control of power demand peak regulation based on deep reinforcement learning." Journal of Building Engineering 75 (2023): 106992.

[22] Tian, Wenchong, et al. "Flooding and overflow mitigation using deep reinforcement learning based on Koopman operator of urban drainage systems." Water Resources Research 58.7 (2022): e2021WR030939.

[23] Bingol, Mustafa Can, and Omur Aydogmus. "A Reinforcement Learning Approach to Robust Control in an Industrial Application." Arabian Journal for Science and Engineering (2024): 1-12.

[24] Zhu, Taiyu, Kezhi Li, and Pantelis Georgiou. "Offline deep reinforcement learning and off-policy evaluation for personalized basal insulin control in type 1 diabetes." IEEE Journal of Biomedical and Health Informatics (2023).

[25] P. Patro, R. Azhagumurugan, R. Sathya, K. Kumar, T. R. Kumar and M. V. S. Babu, "A hybrid approach estimates the real-time health state of a bearing by accelerated degradation tests, Machine learning," 2021 Second International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE), Bengaluru, India, 2021, pp. 1-9, doi: 10.1109/ICSTCEE54422.2021.9708591.