




To Pump or Not to Pump - Sensor-based Reinforcement Learning for an Optimal Scheduler

Alissa Müller ¹, Paul Stahlhofen ¹, and Barbara Hammer ¹




Abstract: Reinforcement Learning can be a powerful tool for Pump Scheduling in Water Distribution Networks. In comparison to classic optimization it can adapt to unseen situations and find optimal schedules in real-time. In this paper, we consider the optimization of energy efficiency under a pressure constraint. For this purpose, we investigate the effects of different sensory information on the learned scheduling policy. We find that information on pressure, tank levels, daytime, flows and pump energy consumption all boost the performance of the agent. However, sparse pressure readings seem to be sufficient at least in small networks.

Keywords: Pump Scheduling, Water Distribution Networks, Reinforcement Learning

1 Introduction

The successful operation of pumps in Water Distribution Networks (WDNs) constitutes a crucial task for water utilities, as failures can have catastrophic consequences [Mc11]. At the same time, the operational cost of pumping is the largest component in the expense of water operators worldwide [MSS17]. Various methods have been proposed for the optimization of pump scheduling, minimizing cost while adhering to operational constraints [BVS05; OT08; Re24]. As an increasing amount of data is monitored in WDNs by Sensory Control and Data Acquisition (SCADA) systems, a data-driven control through Reinforcement Learning [SB18] becomes a promising goal for water network research. In this paper we train a Reinforcement Learning (RL) agent to minimize the operational cost of pumping while ensuring that the pressure at all consumer nodes in the network stays within a satisfactory range. The main focus of this work is on the effect of different types of sensory input on the performance of the learned pump scheduler. For that purpose, we employ the Soft Actor Critic algorithm [Ha18] to the Anytown benchmark [Wa87] and test its ability to generalize its policy to uncertain demand scenarios.

The rest of the paper is organized as follows: Section 1.1 introduces the basic idea of RL as well as the Soft Actor Critic algorithm. Section 1.2 discusses related work. In Section 2, we formalize the optimization problem in an RL framework and describe our experimental setup. Results are discussed in Section 3, before concluding with a brief summary in Section 4.

¹ Universität Bielefeld, AG Machine Learning, Inspiration 1, 33615 Bielefeld, Deutschland, almueller@techfak.uni-bielefeld.de,  <https://orcid.org/0009-0009-3731-8665>;
pstahlhofen@techfak.uni-bielefeld.de,  <https://orcid.org/0009-0004-7187-4992>;
bhammer@techfak.uni-bielefeld.de,  <https://orcid.org/0000-0002-0935-5591>

1.1 Foundations

Reinforcement Learning is a branch of Machine Learning that aims to train an agent in a dynamic environment. At each time step t , the agent takes an action A_t based on the observed environment state S_t . The environment reacts by providing a reward signal $R_{t+1} \in \mathbb{R}$. The goal of the agent is to select its actions in such a way that the

sum of reward signals (called return) is maximized. The action selection strategy is referred to as policy. A schematic illustration of an interaction between agent and environment is shown in Figure 1. In our experiments, the action is the choice of speed for each pump given a state represented by sensor readings, and the reward is determined by the energy efficiency and satisfactory operating conditions (see Section 2).

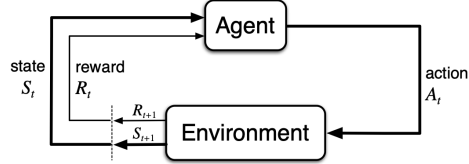


Fig. 1: Schematic illustration of Reinforcement Learning, copied from [SB18]

The Soft Actor-Critic algorithm (SAC) [Ha18] has particularly promising properties for the pump scheduling task: It was designed to handle large and continuous action spaces, which makes it a good candidate for the simultaneous control of multiple variable-speed pumps. SAC employs a replay buffer, meaning that it stores previous experience to be re-used in training, which improves data efficiency. A specialty of SAC is the training objective: In addition to the common RL goal of finding the policy that will maximize the expected return, a separate term for the maximization of entropy in the probabilistic policy is introduced. This leads to better exploration behaviour of the agent and more robust training over all. For a detailed description of SAC, we refer the interested reader to the original paper [Ha18] as well as to the following online resource for an overview².

1.2 Related Work

The operation of pumps in WDNs has been optimized for several decades with varying techniques and objectives [MSS17]. A common theme of existing approaches is the goal of reducing the energy cost of pumping. In addition, several other constraints and objectives have been treated in the literature, ranging from pressure bounds at consumer nodes [HPG20] over control of tank levels [BVS05] to robustness against leakage events [GLB13]. Traditional methods optimize a fixed pump schedule for a certain period of time based on a computational model of the WDN and assumptions on consumer demands [OT08]. More recent methods aim to dynamically adapt the schedule to the current state of the network [Pe25]. RL is particularly promising for this direction of research, as it models the pump scheduler as an agent in a dynamic environment, deciding on the next action to apply based

² <https://spinningup.openai.com/en/latest/algorithms/sac.html>

on current observations of the network. Previous approaches included tabular Q-learning [CPA19], usage of deep networks [HPG20; MWW24] and scheduling of valves in large distribution networks under uncertainty [BMM22]. However, most of the existing literature on pump control assumes deterministic demands or knowledge of the demand pattern by the agent. To address a more realistic use case, we evaluate the performance of the agent under uncertainty. Our goal is to minimize the price for pumping energy while making sure that the pressure at consumer nodes in the network stays within acceptable bounds.

The work most closely related to ours is a very recent publication by Pei et al. [Pe25]. The authors optimize an objective similar to ours taking demand uncertainty into account. The observations they provide to the agent differ throughout their experiments: In one setup they assume very limited information comprised only of tank levels and pump status, while in another they assume knowledge of demand forecasts for every node in the network. We argue that the former underestimates available information as knowledge from sensors distributed across the network is not included while the latter might be an overestimation as demand information may not be available or reliable in the real world. In order to further improve the practical use of RL for pump scheduling, we conduct an in-depth analysis of the benefit of different observation types and different amounts of pressure sensors.

2 Case Study: Pump Scheduling for Anytown

We formulate the reward as a weighted sum of two partial rewards

$$r = c_1 \cdot r_{\text{pres}} + c_2 \cdot r_{\text{energy}} \quad \text{s.t.} \quad c_1 + c_2 = 1, \quad c_1, c_2 > 0$$

$$r_{\text{pres}} = \frac{1}{N} \sum_{n=1}^N \mathbb{I}(h_{\min} < h[n] < h_{\max}) \quad r_{\text{energy}} = 1 - \sum_{p=1}^P \frac{E(p)}{E_{\max}(p)}.$$

Here, r_{pres} as in [HPG20] measures the agent’s ability to satisfy pressure constraints at consumer nodes, where \mathbb{I} is the indicator function, $h[n]$ is the pressure at node n and h_{\min}, h_{\max} are lower and upper pressure constraints, respectively. The second term, r_{energy} , reflects the energy consumption of the pumps, where $E(p)$ is the energy consumption of pump p and $E_{\max}(p)$ is its maximum energy consumption, determined empirically before optimization. To ensure constraint satisfaction, we used weights of $c_1 = 0.9$ and $c_2 = 0.1$ throughout the experiments. The action of the agent is a vector of relative speed settings, ranging from 0 to 1 for each pump in the network. To investigate the effects on performance, we trained the agent with different sensory information in the current state. The configurations used are described in Tab. 1 To conduct our experiments, we used the SAC implementation of Stable Baselines 3 [Ra21] with default hyperparameters with the following exceptions: We increased the learning rate to $3 \cdot 10^{-3}$ and fixed the entropy coefficient to 10^{-2} . For our case study, we used a variant of the Anytown network [Wa87] as used in [Re24]. The network contains three pumps connected to the reservoir, which supplies water to 19 junctions, connected by 42 pipes. Three storage tanks are installed as

Acronym	Tank-Level	Daytime	Pressure	Flow	Energy Consumption
TD	✓	✓	✗	✗	✗
TP(A)	✓	✗	all nodes	✗	✗
TDP(A)	✓	✓	all nodes	✗	✗
TDP(1)	✓	✓	one random node	✗	✗
TDP(8)	✓	✓	eight random nodes	✗	✗
TDP(A)FE	✓	✓	all nodes	✓	✓

Tab. 1: Acronyms of different state representations used during training

water buffers. The network file and our code are publicly available³. Following the general guidelines for pressure constraints in water networks [GKG16], we used a lower pressure constraint of 28.12m and an upper constraint of 70m. We utilized the EPANET simulator [Ro20] through the EPyT-Flow Python library [Ar24]. Experiments used a 30min time step for a duration of 24 hours. Due to stochasticity in the training process, we repeated each training run five times using different seeds and averaged the results. This includes different sensor placements for the random locations. To test the agent’s ability to generalize to unseen scenarios, we added 5% of uncertainty to the demand pattern at each node. To account for this uncertainty, the results reported below in Table 2 were additionally averaged over ten 24-hour episodes.

3 Results and Discussion

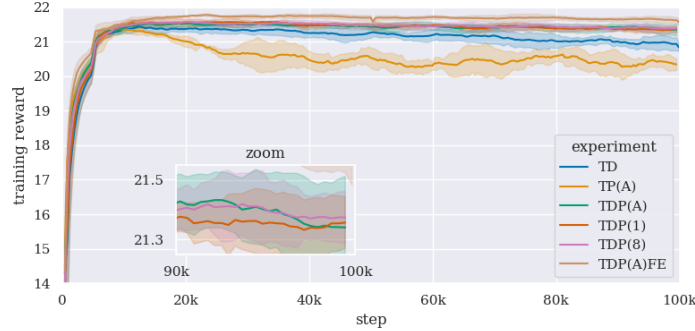


Fig. 2: The training reward during the training process. The shaded area represents the standard deviation over the five repetitions. The lines of the different pressure sensor configurations are included in a zoomed in window, since they overlap heavily.

The episodic returns over the course of all training runs are depicted in Fig. 2. The plotted results reveal, that all observation types are beneficial for the agent to reach a higher reward. However, while adding pressure values to the observation in general results in a clearly

³ Link to code and data on GitHub: https://github.com/HammerLabML/RL4Water_Sensor_Placement_Anytown

visible improvement, the amount of sensors matters relatively little. The reward curves are nearly indistinguishable from one another. Furthermore the standard deviation is small in comparison between the experiments TD and $TP(A)$, so the curves are similar regardless of the random sensor locations.

For a more detailed look into the performance of the fully trained agents we refer to Tab. 2. The table splits the reward into its components of keeping all nodes within pressure bounds and optimizing the operation costs of the pumps. All agents have managed to find a policy to stay within the pressure bounds at all nodes with only small deviations. The price objective gains another noticable boost from the added flow and energy consumption information. However, while inspecting the policy over the course of one day, we found that the agent gained that boost by learning to switch the pump on and off every other hour. Using an additional reward term to discourage this behaviour is part of our ongoing research.

Experiment	Pressure	Price
TD	0.999(.018)	0.607(.159)
TP(A)	0.999(.009)	0.588(.164)
TDP(A)	1.000(.011)	0.606(.172)
TDP(1)	0.999(.012)	0.602(.173)
TDP(8)	1.000(.005)	0.596(.153)
TDP(A)FE	1.000(.005)	0.652(.261)

Tab. 2: This table lists the mean partial rewards the agents reached after training per time step. The standard deviation is given in parentheses.

4 Conclusion

While [Pe25] have achieved good results using only tank levels and pump speeds, we show that it is beneficial to include other sensor information that can be available in real world scenarios such as pressure sensors. Enforcing limited pump switch to improve applicability remains a challenge for future work.

5 Acknowledgments

We gratefully acknowledge funding from the European Research Council (ERC) under the ERC Synergy Grant Water-Futures (Grant agreement No. 951424). We are thankful for the support of André Artelt, maintainer of EPyT-Flow, who implemented various feature requests to make this work possible.

References

- [Ar24] Artelt, A. et al.: EPyT-Flow: A Toolkit for Generating Water Distribution Network Data. Journal of Open Source Software 9 (103), p. 7104, 2024.
- [BMM22] Belfadil, A.; Modesto, D.; Martin Hernandez, J. A.: DRL-Epanet: Deep reinforcement learning for optimal control at scale in Water Distribution Systems. In: Deep Reinforcement Learning Workshop. 2022.

- [BVS05] Barán, B.; Von Lücken, C.; Sotelo, A.: Multi-objective pump scheduling optimisation using evolutionary strategies. en, *Advances in Engineering Software* 36 (1), pp. 39–47, 2005.
- [CPA19] Candelieri, A.; Perego, R.; Archetti, F.: Intelligent Pump Scheduling Optimization in Water Distribution Networks. In (Battiti, R. et al., eds.): *Learning and Intelligent Optimization*. Vol. 11353, Series Title: *Lecture Notes in Computer Science*, Springer International Publishing, Cham, pp. 352–369, 2019.
- [GKG16] Ghorbanian, V.; Karney, B.; Guo, Y.: Pressure Standards in Water Distribution Systems: Reflection on Current Practice with Consideration of Some Unresolved Issues. en, *Journal of Water Resources Planning and Management* 142 (8), p. 04016023, 2016.
- [GLB13] Giustolisi, O.; Laucelli, D.; Berardi, L.: Operational Optimization: Water Losses versus Energy Costs. en, *Journal of Hydraulic Engineering* 139 (4), pp. 410–423, 2013.
- [Ha18] Haarnoja, T. et al.: Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In: *PMLR*. Vol. 80, pp. 1861–1870, 2018.
- [HPG20] Hajgató, G.; Paál, G.; Gyires-Tóth, B.: Deep Reinforcement Learning for Real-Time Optimization of Pumps in Water Distribution Systems. *Journal of Water Resources Planning and Management* 146 (11), p. 04020079, 2020.
- [Mc11] McKee, K. et al.: A review of major centrifugal pump failure modes with application to the water supply and sewerage industries. In: *ICOMS asset management proceedings*. 2011.
- [MSS17] Mala-Jetmarova, H.; Sultanova, N.; Savic, D.: Lost in optimisation of water distribution systems? A literature review of system operation. en, *Environmental Modelling & Software* 93, pp. 209–254, 2017.
- [MWW24] Ma, H.; Wang, X.; Wang, D.: Pump Scheduling Optimization in Urban Water Supply Stations: A Physics-Informed Multiagent Deep Reinforcement Learning Approach. en, *International Journal of Energy Research* 2024 (1), ed. by Guan, W., p. 9557596, 2024.
- [OT08] Ostfeld, A.; Tubaltzev, A.: Ant Colony Optimization for Least-Cost Design and Operation of Pumping Water Distribution Systems. en, *Journal of Water Resources Planning and Management* 134 (2), pp. 107–118, 2008.
- [Pe25] Pei, S. et al.: Real-Time Pump Scheduling in Water Distribution Networks Using Deep Reinforcement Learning. en, *Journal of Water Resources Planning and Management* 151 (6), p. 04025012, 2025.
- [Ra21] Raffin, A. et al.: Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research* 22 (268), pp. 1–8, 2021.
- [Re24] Reis, A. L. et al.: Cost-Efficient Pump Operation in Water Supply Systems Considering Demand-Side Management. en, *Journal of Water Resources Planning and Management* 150 (6), p. 04024017, 2024.
- [Ro20] Rossman, L. A. et al.: *EPANET 2.2. User Manual*. U.S. Environmental Protection Agency, Washington D.C., 2020.
- [SB18] Sutton, R.; Barto, A.: *Reinforcement Learning - An Introduction*. The MIT Press, Cambridge, Massachusetts, 2018.
- [Wa87] Walski, T. M. et al.: Battle of the Network Models: Epilogue. en, *Journal of Water Resources Planning and Management* 113 (2), pp. 191–203, 1987.