Measuring the Performance of Multi-Objective Reinforcement Learning algorithms - Nile River Case Study

Jakub Kontak[1]
Supervisor(s): Zuzanna Osika[1]**, Pradeep Murukannaiah**[1]
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: Jakub Kontak
Final project course: CSE3000 Research Project
Thesis committee: Zuzanna Osika, Pradeep Murukannaiah, Luis Miranda da Cruz

Measuring the Performance of Multi-Objective Reinforcement Learning algorithms - Nile River Case Study

Jakub Kontak*
Supervisors: Zuzanna Osika, Pradeep Murukannaiah
Examiner: Luis Miranda da Cruz
EEMCS, Delft University of Technology
Delft, Mekelweg 5, 2628 CD

Abstract

This study investigates the use of Multi-Objective Natural Evolution Strategies (MONES) to optimise water management control policies in the Nile River Basin, focusing on four key objectives: minimising irrigation deficits for Egypt and Sudan, maximising hydropower production for Ethiopia, and maintaining water level in the High Aswan Dam (HAD). The developed Nile River simulation was integrated with MONES and used to train an agent for making release decisions. Performance metrics including hypervolume, ε-indicator, and Inverted Generational Distance Plus (IGD+) were employed to compare MONES with the EMODPS baseline. The results indicate that while MONES identifies feasible solutions, it falls short in exploration and overall performance compared to EMODPS. The study contributes to the development of water management strategies by open-sourcing Nile River framework compatible with reinforcement learning and demonstrating the potential and limitations of MONES in multi-objective optimisations.

1 Introduction

Water management is an issue faced by many regions worldwide, involving complex decisions to allocate water resources both efficiently and sustainably. These problems often consider many objectives like balancing the water between agricultural and urban use, hydro-power production and flood control [Giuliani et al., 2014]. Modelling them is typically done by simulating the release of water from dams at various timesteps to meet the objectives.

Reinforcement learning (RL) is a machine learning approach focused on training agents to make a sequence of decisions by maximising cumulative rewards through trial and error. RL agents learn optimal policies by interacting with an environment and receiving feedback in the form of rewards or penalties [OpenAI, 2018]. Standard RL frameworks, such as Gymnasium [Towers et al., 2023], provide structured environments for training, evaluating and comparing RL agents. These environments simulate various problems where an agent takes an action at each timestep and is appropriately rewarded for it depending on the objectives. The water management problems are in essence RL problems, where a sequence of decisions has to be made to optimise for some goal, but there is little RL literature and experiments about it. The water management community created frameworks, like Evolutionary Multi-Objective Direct Policy Search (EMODPS) [Zatarain Salazar et al., 2016], to solve their problems. These frameworks are equivalent to RL algorithms but were developed independently. Because of that, they are not written to be compatible with Gymnasium

*An electronic version of this thesis is available at http://repository.tudelft.nl/

and neither are their simulations. As a result, it is not possible to research and compare how the algorithms in these two fields perform. This hinders both fields as the RL community cannot test their algorithms on real world problems, while water management researchers are not able to benefit from the advancements in RL.

Typically, RL problems are modelled using environments where agents take actions at each timestep and receive rewards based on the outcomes of these actions. Water management problems are similarly modelled, involving actions like releasing water from dams to balance multiple conflicting objectives. However, they often integrate their optimisation algorithm, like EMODPS, directly into simulation [Sari, 2022]. This makes changing the algorithm problematic without rewriting the simulation from scratch.

The identified research problem lies in the discrepancy between the modelling approaches and algorithm usage in the RL and water management communities. This discrepancy hinders the flow of ideas between the two fields. Additionally, benchmarking and development face challenges due to the non-reproducibility and lack of open-source availability of simulations in certain studies. Furthermore, RL research mainly focuses on toy problems like cart pole [Barto et al., 1983], resource gathering [Barrett and Narayanan, 2008] or mountain cart [Moore, 1990], with insufficient testing on real-world scenarios. This presents a significant opportunity to apply and evaluate RL in real-world water management contexts.

This study focuses on bridging the gap between the two fields by introducing a complex, real-world water management simulation of the Nile River Basin as an RL Gymnasium framework and comparing the performance of algorithms from these two fields. Since in the Nile River Basin [Sari, 2022] there are multiple, conflicting objectives our study models it as a multi-objective RL problem in the created simulation. The multi-objective optimisation algorithms compared using this simulation are EMODPS from water management and Multi-Objective Natural Evolution Strategies (MONES) developed in the RL community. Measuring the performance of multi-objective algorithms is non-trivial as there are many different metrics to choose from that focus on different aspects of solution sets. In this study, a successful simulation of the Nile River Basin is created and EMODPS is found to outperform MONES by achieving dominant results.

The rest of the paper is organised as follows: Section 2 introduces the Nile River simulation, Gymnasium, EMODPS, MONES and performance metrics. Section 3 presents the process and methods used for obtaining the results. Responsibility and reproducibility of the research are highlighted in Section 4. Section 5 reports on the experimental results, and Section 6 elaborates on those results, as well as a comparison of some policies between algorithms. Lastly, Section 7 summarises the paper and suggests future work.

2 Related work

2.1 Nile River simulation

Nile River Basin simulation was developed in the water management field by Sari [2022]. It simulates the flow of water in the Nile River, as well as additional water inflows, dams, power plants and various demands. The dams are Grand Ethiopian Renaissance Dam (GERD), Roseires, Sennar and High Aswan Dam (HAD), as seen in Figure 1. The first one is in Ethiopia, the next two belong to Sudan and the last is in Egypt. Each of these dams has to release a certain amount of water but it is a complex decision as it has to take into account the goals of different countries. For instance, it should meet the agricultural demands, but also the level of stored water should be high enough as that water might be needed for hydropower production later. The downside of this simulation is that it is connected to the optimisation algorithm, so exchanging it to test different algorithms is difficult. Additionally, it is not written in the Gymnasium framework [Towers et al., 2023] making it incompatible with RL agents.

2.2 Gymnasium

Gymnasium is an open-source library written in Python. It provides a standardised API for RL environment-agent communication [Towers et al., 2023]. The agent is able to take action in the environment and gets back a new state of the environment (also called observation), reward, whether the simulation is finished and some additional information.

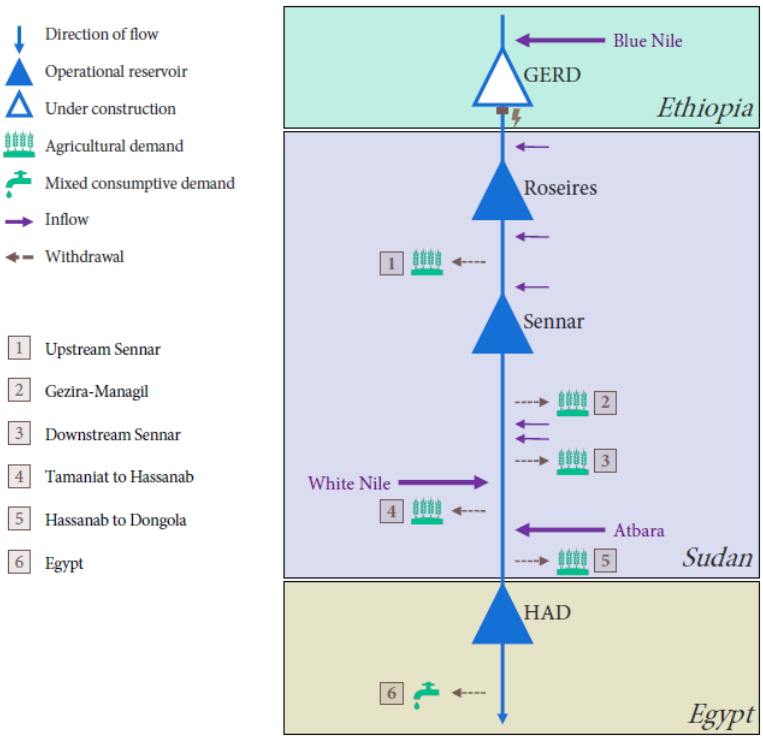

Figure 1: Schematic representation of the Nile River Basin simulation. Source: Sari [2022].

2.3 EMODPS

Evolutionary Multi-Objective Direct Policy Search is an algorithm for solving multi-objective, decision-making problems like water management. It combines Direct Policy Search (DPS) with Evolutionary Algorithms (EAs). DPS is a strategy that optimises the parameters of the decision function, which maps the state of the environment to the actions, instead of optimising the decisions itself [Quinn et al., 2017]. The popular choice of the mapping function is the radial basis function [Sari, 2022, Zatarain Salazar et al., 2016]. The second part of EMODPS is EA, which improves a population of candidate solutions using natural selection mechanisms.

There are numerous papers about multi-objective optimisations using EMODPS in the water management domain. Simulations for rivers like Lower Volta [Owusu et al., 2023], Lower Susquehanna [Witvliet, 2022] and Nile River [Sari, 2022] were developed, and EMODPS baselines were measured for them. Specifically, this study investigates the Nile River simulation and its baseline.

2.4 MONES

Multi-Objective Natural Evolution Strategies (MONES) extend the principles of Natural Evolution Strategies (NES) to address optimisation problems involving multiple conflicting objectives. MONES employ a population-based approach to evolve solutions that balance trade-offs among various objectives, aiming to approximate the Pareto front. It navigates the complex search space, promoting diversity and convergence towards high-quality solutions. MONES follows these four steps [Hayes et al., 2022]:

1. Generate a population of policies based on the current parameters.
2. Execute each policy in the environment.
3. Assess the policies using indicator metric. In this work it is hypervolume.
4. Update the parameters by taking a gradient step to enhance the indicator using the natural gradient.

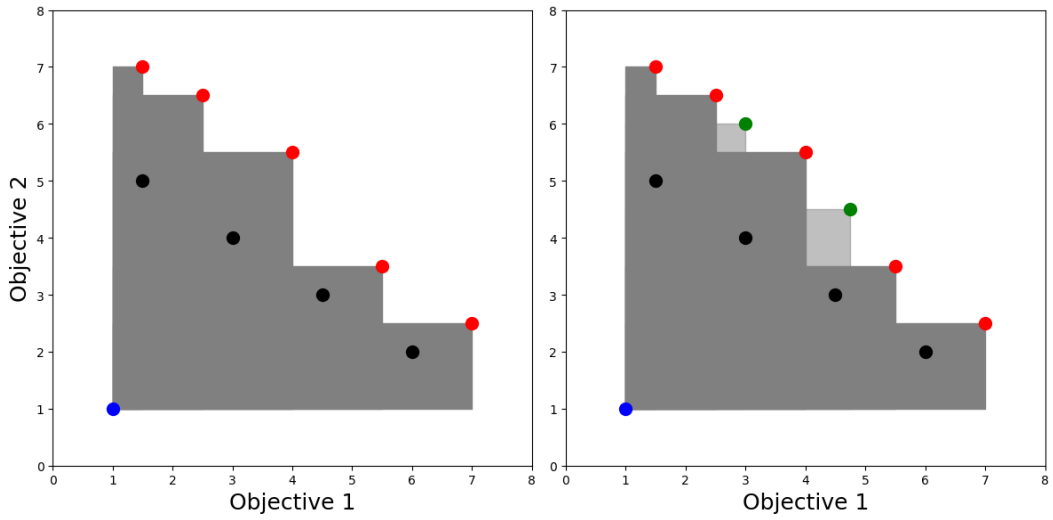

Figure 2: Graphical representation of the hypervolume in a two-objective maximisation problem. The bottom left (blue) point is a reference point. The top (red) points represent non-dominated solutions, while (black) points in the shaded area are dominated solutions. In the right figure, two additional non-dominated (green) points increase the hypervolume. Adapted from: Hayes et al. [2022]

2.5 Performance metrics

MORL is a branch of reinforcement learning that aims to optimise multiple conflicting objectives simultaneously, addressing real-world problems that often involve such conflicts [Liu et al., 2015]. A comprehensive paper on MORL provides various metrics for comparing policies, including hypervolume and the additive ε-indicator [Hayes et al., 2022]. Additionally, the Inverted Generational Distance Plus (IGD+) indicator, proposed by Ishibuchi et al. [2015], can be useful for algorithm comparison. The following section describes each performance metric in detail.

2.5.1 Hypervolume

Hypervolume [Zitzler and Thiele, 1999] calculates the multi-dimensional volume spanned by the non-dominated points of each solution set with respect to the reference point. For a 2-objective problem, it can be visualised as an area, as seen in Figure 2. The left figure represents one solution set, while the right one represents the same solution set with two additional non-dominated points, resulting in an increase in hypervolume. This metric provides a general comparison between two sets. However, there exists a body of problems associated with it. Namely, differences in the hypervolume are not intuitive as they do not map easily to any real-world utility [Hayes et al., 2022]. Additional, non-dominated points in any extreme can significantly contribute to the metric while being useless from the point of balancing objectives [Hayes et al., 2022]. Further, non-dominated points near other solutions, offering potentially beneficial trade-offs, provide little increase in hypervolume

2.5.2 Additive ε-indicator

The additive ϵ-indicator was introduced for minimisation problems. It serves as a quantitative comparison between different multi-objective algorithms, specifically their solution sets [Zitzler et al., 2003]. The formula for the maximisation problems used in this paper is as follows:

$$I_{\epsilon^+} = \inf_{\varepsilon \in \mathbb{R}} \{\forall \mathbf{V}^{\pi} \in PF, \exists \mathbf{V}^{\pi'} \in S : V_i^{\pi} \leq V_i^{\pi'} + \varepsilon, \forall i \in \{1, \ldots, n\}\},$$

where the number of objectives is n, V^{π} is the value of policy π, S is the potential solution set and PF is a Pareto front [Hayes et al., 2022]. In other words, the additive epsilon indicator identifies the smallest factor by which the objective vectors of the Pareto front must be decreased such that each point in the modified Pareto front is weakly dominated by at least one solution from the solution set S.

2.5.3 Inverted Generational Distance Plus indicator

The Inverted Generational Distance Plus indicator is given by the following formula:

$$IGD^{+}(A) = \frac{1}{|Z|}\left(\sum_{i=1}^{|Z|} {d_i^+}^2\right)^{1/2},$$

where Z is a set of reference points approximating Pareto front, A is a solution set to calculate the metric for and $d_i^+ = max\{a_i - z_i, 0\}$ represents the modified distance from $z_i \in Z$ to the closest point in A for the minimisation problem [Ishibuchi et al., 2015]. This metric is weakly Pareto compliant [Ishibuchi et al., 2015], meaning that if the solution set A weakly dominates solution set B the metric has to be lower for the solution set A. Thus, if set A has a lower value than set B we can be sure that set B does not dominate set A.

3 Methodology

3.1 Process

The first phase of the project involved familiarisation with the Gymnasium environments and the simulation that needed to be rewritten. After a thorough analysis and coming up with a scalable structure, the implementation began. The next step was to integrate MONES with the simulation. This required reviewing relevant literature on the algorithm and making modifications to the simulation itself. The connection with MONES was verified by examining the hypervolume progress to check whether the agent was learning. However, the agent was not performing as expected, which led to a debugging phase. After a close examination, the issue was identified: the simulation was clipping the agent's actions to minimal release levels because its decisions were too small. This prevented any learning, as the agent kept getting the same rewards. Multiplying the actions by 100 resolved this, pushing them into a range where they had a real impact on rewards.

Further, the longer experiments were run comparable to EMODPS. To save time, parallelism in the MONES algorithm was implemented to run it on multiple local cores. However, the experiments were computationally expensive and were moved to the DelftBlue supercomputer. This process involved some challenges, including familiarisation with the software and long waiting times that extended the feedback loop. Ultimately, several agents were trained. With the results, another round of literature investigation took place to find and understand methods for comparing different sets of policies. The outcomes of the above process are detailed further in this paper.

3.2 Simulation

The recreated version of the simulation from the Nile River Basin paper [Sari, 2022] consists of the same four basic components encapsulated in classes: Dams (or Reservouirs), Irrigation Districts, Power Plants, and Catchments. These are divided into controlled and uncontrolled facilities on which an action can and cannot be performed, respectively. Dams are an example of a controlled facility for which we can specify how many m^3/s of water should be released. The Irrigation Districts expect a certain amount of water, which they consume for each month. The Power Plants are attached to the Dams and produce power from water releases if the water is above a certain threshold. Lastly, Catchments are used to model water entering the flow either from rain or other branches of the river. Each of the facilities can have an objective that is going to be calculated for every step. The connections between the facilities are modelled as graph edges and are represented by the Flow class. What is more, we created a Water Management System class that extends the Gymnasium environment. This class can encapsulate the system of connected facilities like the one in the Nile River Basin case study [Sari, 2022] and allow for the connection of RL algorithms, compatible with the gymnasium environments. The implementation of the simulation is open-source and can be found at `https://github.com/jkontak13/MONES-for-Nile-River-Basin`.

The simulation, as seen in Figure 1, includes four dams on the Nile River: GERD, Roseires, Sennar, and HAD. These dams are situated in three different countries: Ethiopia, Sudan, and Egypt. Additionally, the main river receives significant inflows from (but is not limited to) the Blue Nile, White Nile, and Atbara Rivers, which are modelled as catchments within the simulation. Beyond

Table 1: Countries and their corresponding objectives MONES uses for training.

Country	Objective	Direction
Egypt	Irrigation demand deficit	Minimisation
Egypt	Minimum HAD water level	Maximisation
Sudan	Irrigation demand deficit	Minimisation
Ethiopia	Hydropower production	Maximisation

Table 2: Representation of objective units for MONES before conversion and EMODPS units.

Objective	MONES units	EMODPS units
Egypt irrigation deficit	Demand deficit in cubic meters per second (minimisation)	Average yearly demand deficit in billions cubic meters per year (minimisation)
Egypt min HAD level	Number of months equal or above minimum power generation level (maximisation)	Frequency of months below minimum power generation level (minimisation)
Sudan irrigation deficit	Demand deficit in cubic meters per second (minimisation)	Average yearly demand deficit in billions cubic meters per year (minimisation)
Ethiopia hydropower	Total hydroenergy generation in MWh (maximisation)	Yearly hydroenergy generation in TWh (maximisation)

water sources, the simulation also accounts for agricultural and consumptive demands, represented as irrigation districts. Furthermore, there is a power plant at GERD. Since the simulation operates within a Gymnasium environment actions can be taken at each timestep. These actions are represented by a four-element array specifying water releases from each dam, ordered in the river flow direction.

3.3 Objectives

There are four objectives MONES optimises for, as seen in Table 1. They are adapted from the Sari [2022] paper.

Egypt irrigation demand deficit (minimisation) is the total sum of deficits per month in Egypt's irrigation districts. Each monthly deficit is multiplied by the number of days in that month. This is done to make the results comparable with Sari [2022].

Egypt minimum HAD water level (maximisation) is the number of months when the water in HAD is above the power generation level, which is 159 meters above sea level.

Sudan irrigation demand deficit (minimisation) is the same as Egypt. Sum of deficits per month is calculated and each monthly deficit is multiplied by the number of days in that month.

Ethiopia hydropower generation (maximisation) is the total energy generated in MWh across the whole simulation.

3.4 Comparison

To compare solutions obtained from MONES to EMODPS there are two steps to take: 1) Scale the MONES rewards to use the same units as EMODPS, so the solution sets are comparable. 2) Normalise both solution sets using normal and reversed scale min-max normalisation, depending on the calculated metric. This is done to scale each objective to the same range, making their contributions to metrics equal.

(1) The results from both algorithms are given in units seen in Table 2. For a proper comparison, MONES results are converted into the EMODPS units. The irrigation deficit for both Egypt and Sudan is converted using the equation below:

$$MONES\_result' = MONES\_result \cdot \frac{3600 \cdot 24}{20 \cdot 1.000.000.000}$$

The minimum HAD level is converted with the following formula:

$$MONES\_result' = \frac{(240 - MONES\_result)}{240}$$

The Ethiopia hydropower is converted as follows:

$$MONES\_result' = \frac{MONES\_result}{20 \cdot 1.000.000}$$

(2) After the rewards are converted to the same units the normalisation takes place. Min-max normalisation is performed with the formula below:

$$x' = \frac{x_{min} - x}{x_{min} - x_{max}},$$

which results in the mapping of the best value to 1, the worst to 0 and the other values respectively in the range between 1 and 0. This normalisation is used for the calculation of ε-indicator. The reversed scale min-max normalisation follows below formula:

$$x' = \frac{x_{max} - x}{x_{max} - x_{min}},$$

which converts the best objective values to 0, while the worst ones to 1. It is used for the hypervolume and IGD+ calculation. The reversed scale is needed since pymoo [Blank and Deb, 2020] considers minimisation problems instead of maximisation.

The above two steps, unit conversion and normalisation, allow for the calculation of hypervolume, ε-indicator and IGD+ in comparable units for both solution sets.

3.5 Performance metrics

To evaluate and compare the performance of EMODPS and MONES for the Nile River Basin case study, we calculate several performance metrics. These metrics help us understand the effectiveness of each algorithm in achieving the four objectives: Sudan water deficit, Egypt water deficit, Ethiopia power production, and the number of days with minimal water level in the HAD.

3.5.1 Hypervolume

After the reversed scale min-max normalisation described in Section 3.3, where the best value is turned to 0 and the worst value to 1 the hypervolume is calculated. It is done using the Python pymoo package [Blank and Deb, 2020]. The reason for using a reversed scale instead of normal min-max normalisation is that the package considers minimisation problems, instead of maximisation ones. The reference point is set to (1.2, 1.2, 1.2, 1.2), so that it is larger than all the other solutions, this is also a value used in the literature [Sari, 2022].

3.5.2 ε-indicator

Another metric to compare the solution sets of EMODPS and MONES is the additive ε-indicator. To calculate this metric min-max normalisation is used, which turns the best value into 1 and the worst value into 0. This is done because ε-indicator is defined for the maximisation problem [Hayes et al., 2022]. Even though the equation in Section 2.5 uses the true Pareto front we can substitute it with set Z - the non-dominated points from the union of EMODPS and MONES solution sets. Given solution set S and the Pareto front substitution Z we calculate the ε-indicator by looking for the first ε value (given some precision) such that for all points z_i in Z there exists a point s_i in S such that $z_i \leq s_i + \varepsilon$ for all objectives.

3.5.3 Inverted Generational Distance Plus indicator

This metric, described in detail in Section 3.3, is calculated using the Python pymoo package [Blank and Deb, 2020] after the reversed scale min-max normalisation. The reasons for this normalisation are the same as in hypervolume calculation. In the context of this paper, the true Pareto front isn't known. Therefore the substituted set Z is constructed in the same way as for the ε-indicator, using non-dominated points from the union of both solution sets.

4 Responsible research

4.1 Ethical aspects

This research is part of the important but complex topic of managing water resources on the Nile River. It assumes cooperation, goodwill and willingness for compromises between the involved countries. The policies created in this paper are by no means optimal and should not be used as official guidelines. However, this work contributes to resolving the problem of managing water on the Nile River, by exploring new ways of finding good approximations of optimal policies. These potential policies are found in an unbiased and independent way and do not prioritise any country above the others. All the results, methods and the code are publicly available to maximise the transparency.

Important to note is that most simulations use data and this one is no different. The quality of data will impact the policies, which might be faulty in case the data is not correct. The data used in this paper comes from Sari [2022].

4.2 Reproducibility

To fully contribute to the knowledge base we make our code, as well as, this paper fully open-source so that anyone can potentially rerun the calculations. In addition, Section 3 reports in detail on how the metrics are calculated, so the code can also be verified whether it works, as intended in the paper. The experiment design is presented in Section 5.1, which describes exactly the training time, hyperparameters used for the training, and number and model of CPUs. Given all that information the results are easily reproducible and verifiable.

5 Results

5.1 Experiment design

The group implementation of the simulation allows for an individual, experimental phase. It consists of connecting a specific, multi-objective reinforcement learning algorithm, which for this study is MONES, to the simulation. Further, using MONES an agent is trained in order to approximate the Pareto front for the Nile River Basin problem. Such an agent is represented as a small neural net, consisting of a single hidden layer with 50 neurons, *tanh* activation function and Xavier uniform initialisation for each layer, following closely Hayes et al. [2022]. Important to note is that the final neural network outputs are multiplied by 100 to aid the learning process, ensuring the actions are in the range where they have an impact on the rewards. Such an agent, given five observation points, returns four actions. Observation points represent the amount of water in each dam given in m^3 and the month number. Whereas the returned actions indicate how much water each dam should release in m^3/s. The training uses a hypervolume indicator, described in subsection 2.5.1, as the performance metric.

Training of the agent takes place on the DelftBlue supercomputer using 32 CPUs (2x Intel Xeon E5-6248R 24C 3.0GHz | 192 GB). It is run three times with the following random seeds 42, 1410, 2137 and each run takes around 10 hours. The other hyperparameters are set to 270 iterations, with a population size of 128 and a number of runs per population equal to 1. This gives 34,560 number of function evaluations (NFEs) making it comparable to the 50,000 NFEs used for the training of EMODPS [Sari, 2022]. The above parameters were constrained by the limited computational resources and time constraints.

5.2 Multi-Objective Natural Evolution Strategies

In this section, a trained agent with a seed equal to 1410 is described as it achieves the highest hypervolume as can be seen in Figure 3. This agent from the MONES algorithm produced a set of 18 non-dominated solutions, while the EMODPS baseline had 222 points. The best values for each objective are summarised in Table 3. Specifically, the lowest irrigation deficit for Egypt was 3.9 billion cubic meters, and the highest average annual energy production for Ethiopia was 3.9 TWh. Additionally, for the irrigation deficit in Sudan and the frequency of months with a water level below the minimum in the HAD, some solutions achieved a perfect score of 0.0.

Table 3: Comparison of best performance between MONES and EMODPS for each of the objectives.

Objective	MONES	EMODPS
egypt irrigation deficit	3.852	3.499
egypt min HAD level	0.0	0.0
sudan irrigation deficit	0.0	0.0
ethiopia hydropower	3.852	15.125

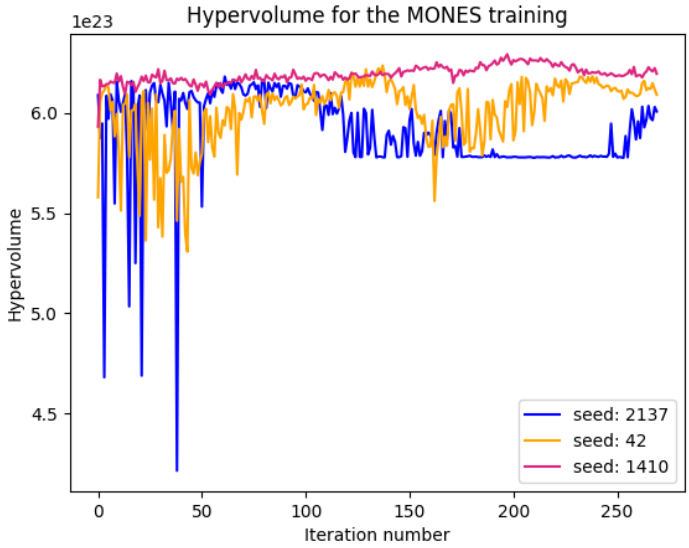

Figure 3: Hypervolume indicactor for not normalised objectives, calculated during the training of the MONES agent. Values for different random seeds are presented.

Although the values for the minimum water level in HAD and the irrigation deficits for both Sudan and Egypt are comparable between MONES and EMODPS, there is a significant difference in hydropower production, with MONES not approaching the values achieved by EMODPS. In Section 6 this difference is explored using an example policy from each algorithm.

The training process for each seed can be seen in Figure 3, where the hypervolume for each training iteration is plotted. It is visible for the agent with seed 2137 that around the 125th iteration, the algorithm reaches a local optimum and stays in it for another 125 iterations. We can also see that some seeds have much higher results variability than others, for instance, seed 1410 is more stable than 2137.

5.3 Performance metrics

Multiple performance metrics are calculated to compare the solution sets of MONES and EMODPS. Firstly, a combined Pareto front Z is constructed. Set Z contains 223 points, where 222 are from the EMODPS algorithm and only 1 comes from the MONES algorithm. This combined with the size of the solution sets shows the superiority of the former algorithm. However, for completeness, other metrics are also presented.

After reversed scale mix-max normalisation, described in Section 3.4, a hypervolume is calculated. Larger values of hypervolume often represent solution sets that have a wider spread and better-performing policies. The values are 0.32 for MONES and 2.03 for EMODPS, as seen in Table 4. For the context, the maximum possible hypervolume in this scenario is 2.0736, while the minimum is 0.0016. This does not prove that EMODPS produces near-optimal solutions, because the normalisation determines the minimum and maximum values. However, it points to the fact that EMODPS produces better results than MONES.

Table 4: Comparison of performance metrics between MONES and EMODPS. The arrow near the metric name signals whether larger or smaller values are more desirable.

Metric	MONES	EMODPS
Hypervolume ↑	0.32	2.03
Additive ε-indicator ↓	1.0	0.005
IGD+ ↓	0.84	0.00002

For the ε-indicator min-max normalisation is used, described in Section 3.4. The values for this indicator are presented in Table 4 and are as follows: 1.0 for MONES and 0.005 for EMODPS. The former value is caused by the fact that MONES find solutions with the least amount of energy produced, so after the normalisation they are set to 0. To make any of them weakly dominate the points in set Z with the maximum power production it must be increased by at least 1.0, which is the solution we get. The value of 0.005 for EMODPS shows us that there indeed has to be at least one point from MONES in the set Z. However, this point is close to the EMODPS solution set and after decreasing the point by 0.005 it is weakly dominated by some solution from EMODPS.

Lastly, the IGD+ indicator is presented. It also measures the results after the reversed scale min-max normalisation. The value for MONES is 0.84, while for EMODPS it is 0.00002. This value represents an average Euclidean distance from each point in the Pareto front to the closest point in the solution set. The EMODPS solution set is equal to the Pareto front (excluding one point), which is observed in the metric being close to 0. The MONES is further apart giving an average distance of 0.84. This value seems to be largely influenced by the poor performance in hydropower production. The worst possible value after normalisation for this Pareto front is 1.65, positioning the MONES solution set around the middle.

6 Discussion

6.1 Exploration analysis

The analysis of the parallel plot in Figure 4 reveals insights into MONES performance. The Figure 4 presents MONES solutions, with a highlighted, non-dominated policy. Additionally, it displays compromises proposed by Sari [2022]. The MONES results are clustered around similar values for Sudan and Egypt deficits, as well as, for Ethiopia hydropower production. While, the objective with highest variability is the minimum water level in the High Aswan Dam. It seems the algorithm explores around similar policies, which might indicate an inability to extensively explore the solution space. For comparison, EMODPS finds solutions that are highly diverse, which is presented by the Best Ethiopia Hydropower policy, seen in Figure 4. This policy sacrifices Sudan and Egypt's deficit objectives for the hydropower and the HAD level, presenting a different potential solution. Given MONES solutions we can observe the trade-off between minimising Egypt's deficit and keeping the minimum water level in HAD. Interestingly, this trade-off was also found by Sari [2022].

The lower level of exploration is also visible in the amount of non-dominated points in the final solution set found by both algorithms, the difference between 222 and 18. Additionally, only one of the MONES solutions is non-dominated by the EMODPS set. This suggests that while MONES can identify some feasible solutions, it lacks the depth required to find high-quality policies.

6.2 Policy analysis

MONES and EMODPS water levels in GERD (dam in Ethiopia) are analysed because of the discrepancy in the hydropower production between the two algorithms. Figures 5 and 6 present the GERD water level for policies best performing in hydropower production for MONES and EMODPS respectively. A strategy of the latter algorithm is to fill the dam first, by storing the water in the beginning, and only after to release it cyclically. Whereas, MONES seems to have not explored such a scenario and starts with the cyclic releases from the beginning. This again can point to too little exploration on the MONES side.

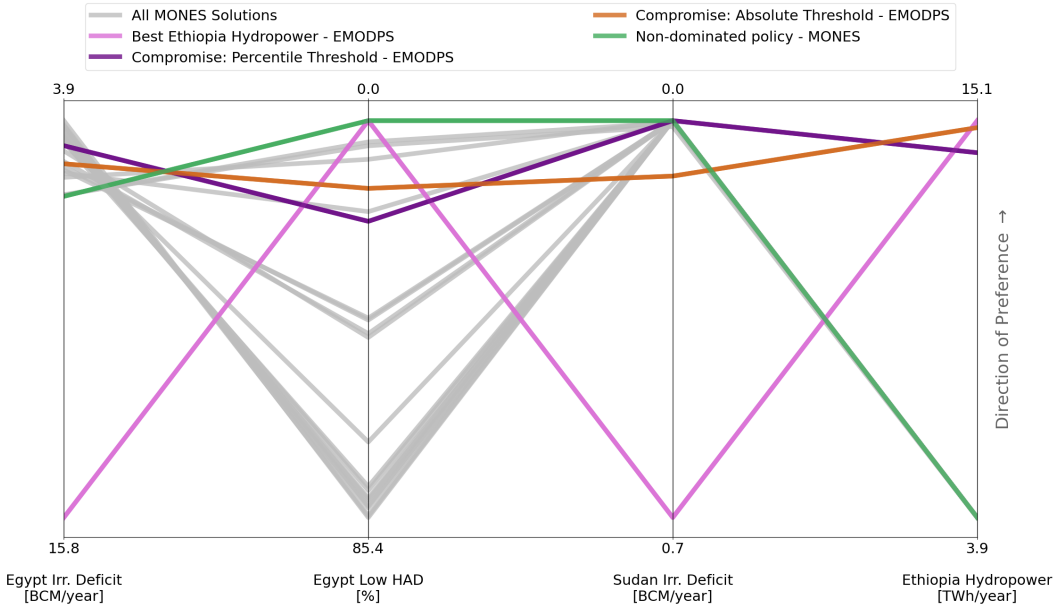

Figure 4: Parallel plot for the MONES solution set. Additionally added chosen solutions from EMODPS [Sari, 2022]. The percentile compromise is the policy that is in at least the 45th percentile for each objective. The absolute compromise achieves at least a better value for all objectives than 0.82 on the scaled normalised using the EMODPS solution set.

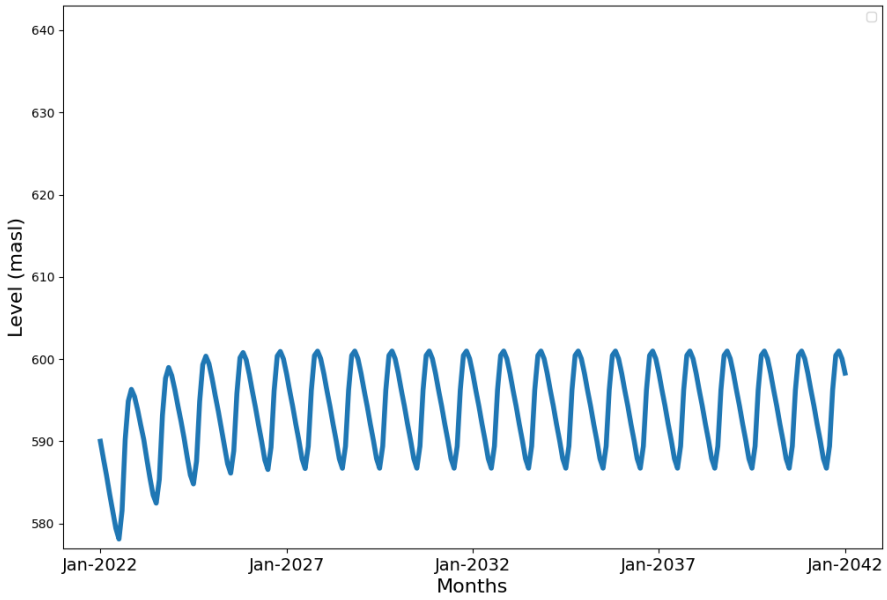

Figure 5: Water level in GERD based on MONES policy achieving the best hydropower production.

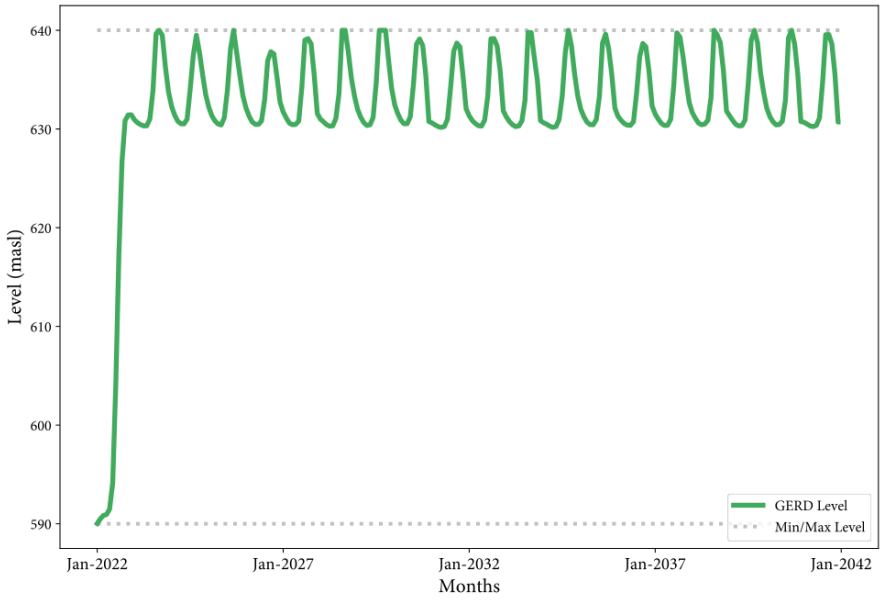

Figure 6: Water level in GERD based on EMODPS policy achieving the best hydropower production. Source: Sari [2022].

7 Conclusions

Integrating reinforcement learning techniques into water management has the potential to significantly advance both fields. For the RL community, real-world water management problems offer valuable benchmarks for their algorithms and stimulate the development of new techniques tailored to these challenges. Conversely, water management can benefit from the influx of RL researchers and the application of various multi-objective RL algorithms, which may outperform current state-of-the-art methods.

In this study, we adapted a Nile River Basin simulation to be compatible with the Gymnasium framework, marking a step towards this integration. Additionally, we tested MONES, a multi-objective RL algorithm, as an alternative to the standard EMODPS commonly used in water management. Although MONES did not outperform EMODPS, it delivered respectable results, indicating promising potential for RL applications in water management. To facilitate further research, we made the results and the simulation easily reproducible and open-source.

The identified research gap can be further addressed in the future by developing an EMODPS implementation compatible with the Gymnasium framework. It would enable EMODPS to be evaluated on benchmark problems from the RL community. Such work would determine whether EMODPS's strong performance is specific to the water management problems or if it is a versatile algorithm, well-performing in the field of multi-objective RL.

References

Leon Barrett and Srini Narayanan. Learning all optimal policies with multiple criteria. pages 41–47, 01 2008. doi: 10.1145/1390156.1390162.

Andrew G. Barto, Richard S. Sutton, and Charles W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5):834–846, 1983. doi: 10.1109/TSMC.1983.6313077.

J. Blank and K. Deb. pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509, 2020.

M. Giuliani, J. D. Herman, A. Castelletti, and P. Reed. Many-objective reservoir policy identification and refinement to reduce policy inertia and myopia in water management. *Water Resources Research*, 50(4):3355–3377, 2014. doi: https://doi.org/10.1002/2013WR014700. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2013WR014700.

C. F. Hayes, R. Radaulescu, and Bargiacchi. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 2022.

Hisao Ishibuchi, Hiroyuki Masuda, Yuki Tanigaki, and Yusuke Nojima. Modified distance calculation in generational distance and inverted generational distance. In António Gaspar-Cunha, Carlos Henggeler Antunes, and Carlos Coello Coello, editors, *Evolutionary Multi-Criterion Optimization*, pages 110–125, Cham, 2015. Springer International Publishing. ISBN 978-3-319-15892-1.

Chunming Liu, Xin Xu, and Dewen Hu. Multiobjective reinforcement learning: A comprehensive overview. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(3):385–398, 2015. doi: 10.1109/TSMC.2014.2358639.

Andrew William Moore. Efficient memory-based learning for robot control. *University of Cambridge Computer Laboratory*, 1990.

OpenAI. Welcome to spinning up in deep RL, 2018. URL https://spinningup.openai.com/.

A. Owusu, J. Zatarain Salazar, M. Mul, P. van der Zaag, and J. Slinger. Quantifying the trade-offs in re-operating dams for the environment in the lower volta river. *Hydrology and Earth System Sciences*, 27(10):2001–2017, 2023. doi: 10.5194/hess-27-2001-2023. URL https://hess.copernicus.org/articles/27/2001/2023/.

Julianne D. Quinn, Patrick M. Reed, and Klaus Keller. Direct policy search for robust multi-objective management of deeply uncertain socio-ecological tipping points. *Environmental Modelling & Software*, 92:125–141, 2017. ISSN 1364-8152. doi: https://doi.org/10.1016/j.envsoft.2017.02.017. URL https://www.sciencedirect.com/science/article/pii/S1364815216302250.

Yasin Sari. Exploring trade-offs in reservoir operations through many objective optimisation: Case of Nile River basin. 2022.

Mark Towers, Jordan K. Terry, Ariel Kwiatkowski, John U. Balis, Gianluca de Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Andrew Tan Jin Shen, and Omar G. Younis. Gymnasium, March 2023. URL https://zenodo.org/record/8127025.

Mark Witvliet. Multi-sector water allocation: The impact of nonlinear approximation network hyperparameters for multi-objective reservoir control. 2022.

Jazmin Zatarain Salazar, Patrick M. Reed, Jonathan D. Herman, Matteo Giuliani, and Andrea Castelletti. A diagnostic assessment of evolutionary algorithms for multi-objective surface water reservoir control. *Advances in Water Resources*, 92:172–185, 2016. ISSN 0309-1708. doi: https://doi.org/10.1016/j.advwatres.2016.04.006. URL https://www.sciencedirect.com/science/article/pii/S0309170816300896.

E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999. doi: 10.1109/4235.797969.

Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2), 2003.