Deep reinforcement Learning Challenges and Opportunities for Urban Water Systems.

Ahmed Negm[1], Xiandong Ma[1] and George Aggidis[1, *]

[1]Lancaster University Energy Group, School of Engineering, Lancaster. LA1 4YW, UK

*Corresponding Author: George Aggidis, G.Aggidis@lancaster.ac.uk, +44 (0) 1524 593052, Office C08, Gillow Avenue, Lancaster, LA1 4YW, UK

Abstract

The efficient and sustainable supply and transport of water is a key component to any functioning civilisation making the role of urban water systems (UWS) inherently crucial to the wellbeing of its customers. However, managing water is not a simple task. Whether it is aging infrastructure, transient flows, air cavities or low pressures; water can be lost as a result of many issues that face UWSs. The complexity of those networks grows with the high urbanisation trends and climate change making water companies and regulatory bodies in need of new solutions. So, it comes as no surprise that many researchers are invested in innovating within the water industry to ensure that the future of our water is safe.

Deep reinforcement learning (DRL) has the potential to tackle complexities that used to be very challenging as it relies on deep neural networks for function approximation and representation. This technology has conquered many fields due to its impressive results and can effectively revolutionise UWS. In this article, we explain the background of DRL and the milestones of this field using a novel taxonomy of the DRL algorithms. This will be followed by with a novel review of DRL applications in the UWS which focus on water distribution networks and stormwater systems. The review will be concluded with critical insights on how DRL can benefit different aspects of urban water systems.

Key words: Deep reinforcement learning; leakage; urban water systems; pressure management; stormwater systems.

1. Introduction

Water scarcity is a reality experienced by 2.3 billion people globally that live in water-stressed countries yet water demand is set to increase by 40% by 2030 (Endo *et al.*, 2017). Our water preservation practices are not sustainable and will diminish the availability of clean water. In response to the rising challenges of water distribution in the UK, regulatory bodies such as Ofwat and the Public Accounts committee have been pushing water companies to reimagine the water sector by 2050 (Mace, 2020). Main themes of the sector-wide strategy include to 'Deliver resilient infrastructure systems' and 'achieving net-zero carbon' that will rely on developing better water management within UWS (U.K.W.I.R., 2020). The preservation of the world's most important resource increases in complexity as we consider the outdated infrastructure forced to keep up with the rising customer demands. Tackling such high dimensional scenarios will require more research and extensive efforts from both industry and academia to rectify the mishandling of water distribution networks.

In this paper we explore a specific subfield of machine learning that has overwhelmed the research community and IT companies such as OpenAI (Berner *et al.*, 2019) and Google (Silver *et al.*, 2016) - Deep Reinforcement Learning (DRL). DRL is an emerging field of dynamic computing that has risen through the use of deep neural networks to advance reinforcement learning (Mnih *et al.*, 2015a). Its successes rely on its applicability in real world scenarios that require learning from experience and its failures arise from challenges in instability and environment definition. The appealing nature of finding low-dimensional features the accurately represent high-dimensional real-world problems and experience driven autonomous learning makes DRL a true advancement in AI. As this field grows, researchers have developed numerous deep reinforcement learning algorithms that equip computational methods such as bootstrapping, backups, replay memory and function approximation to overcome any issues that arise and improve results (Li, 2017). In addition to numerous neural network architectures, deep reinforcement learning has quickly grown to become an unclassified jungle of artificial intelligence advancements.

Navigating the field of DRL requires a solid knowledge of its predecessor Reinforcement Learning and the major advancements that were led by the introduction of neural networks which is covered in section two. After reviewing the wider field of research, this paper focuses on a novel review of the application of DRL in urban water systems which includes challenges and opportunities to applying DRL in UWS followed by case studies in water distribution and stormwater management in section three. This in-depth review of the current research in the UWS will lead to an extensive discussion regarding the future of deep reinforcement learning in UWS in section four. This will hopefully unveil unexplored avenues of research to promote the use of DRL in water. A list of abbreviations used is available in Table 1-1 below.

Table 1-1 *List of Abbreviations*

Abbreviation	Name
A2C	Advantage Actor Critic
A3C	Asynchronous Advantage Actor Critic
ACKTR	Actor-Critic using Kronecker-factored Trust Region
ANN	Artificial Neural Networks
ASM-SMP	Activated Sludge Model - Soluble Product
C51	Categorical Deep Quality Network
CMA-ES	Covariance Matrix Adaptation Evolution Strategy
CSO	Combined Sewer Overflow
DDPG	Deep Deterministic Policy Gradient
D-DQN	Dueling – Deep Quality Network
DE	Differential Evolution
DMODRL	Dynamic Multi Objective Deep Reinforcement Learning
DP	Dynamic Programming
DQN	Deep Quality Network
DRL	Deep Reinforcement Learning
DST	Deep Sea Treasure
E-PPO	Exploration enhanced – Proximal Policy Optimisation
ExIt	Expert Iteration
FQF	Fully parameterised Quantile Function
FSSRS	Fixed-Step Size Random Search
GA	Genetic Algorithm
GAE	Generalised Advantage Estimation
GCN	Graph Convolutional Network
GCN-DQN	Graph Convolutional Network – Deep Quality Network
GCN-DRL	Graph Convolutional Network – Deep Reinforcement Learning
GLIE MC	Greedy in the Limit with Infinite Exploration - Monte Carlo
GPU	Graphics Processing Unit
I2A	Imagination-Augmented Agents
IQN	Implicit Quantile Regression
KA-PPO	Knowledge Assisted – Proximal Policy Optimisation
k-NN	k-Nearest Neighbour
MADRL	Multi Agent Deep Reinforcement Learning
MBMF	Model Based priors for Model Free
MB-MPO	Model Based Trust Region Policy Optimisation
MBVE	Model Based Value Estimation
MC-ES	Monte Carlo Exploration Strategy
MDP	Markov Decision Process
ME-TRPO	Model Ensemble Trust Region Policy Optimisation
ML	Machine Learning
MO-MCTS	Multi Objective – Monte Carlo Tree Search
MP-DQN	Multi Policy – Deep Quality Network
MPQ	Multi-Period Quadratic
ORM	Objective Relation Mapping
PDW	Performance Degree
PPO	Proximal Policy Optimisation
PQDQN	Proposed Parity-Q Deep Quality Network
PSO	Particle Swarm Optimisation
Q Learning	Quality Learning
QR-DQN	Quantile Regression – Deep Quality Network
QT-Opt	Quantile Regression for Reinforcement Learning
REINFORCE	REward Increment = Nonnegative Factor × Offset Reinforcement × Characteristic Eligibility
RL	Reinforcement Learning
SAC	Soft Actor Critic
SARSA	State-Action-Reward-State-Action
SCADA	Supervisory Control and Data Acquisition
SRI	System Resilience Index
SSC	Suspended Sediment Control
STEVE	Stochastic Ensemble Value Expansion
SVG	Stochastic Value Gradients
SVM	Support Vector Machine
SWMM	Storm Water Management Model
TD	Temporal Difference
TD3	Twin Delayed Deep Deterministic Policy Gradient

TRPO	Trust Region Policy Optimisation
UCB	Upper Confidence Bound
UWOT	Urban Water Optioneering Tool
UWS	Urban Water Systems
WQR	Water Quality Resilience
WWTP	WasteWater Treatment Plant

2. Deep Reinforcement Learning Background

The field of machine learning (ML) has been a trending topic for researchers from diverse backgrounds such as virologist, biologists, engineers, psychiatrists, and more (Libbrecht and Noble, 2015; Nichols, Herbert Chan and Baker, 2019) due to its ability to analyse real world problems using algorithms that tackle more dynamic perspective and improve with experience (Shinde and Shah, 2018). Machine learning begun as researchers hoped to achieve a novel area where instrumentation can achieve innate learning and demonstrate more 'intelligent' behaviour. From the first ML algorithm in 1951 named 'response learning algorithm' until the current day, artificial intelligence has only been empowered by this new field (Shinde and Shah, 2018). Some of the major achievements in ML was the creation of the algorithms Linear Classifier, Naive Bayes, Bayesian Network, Support Vector Machines (SVM), k-Nearest Neighbour (k-NN) and Artificial Neural Networks (ANN) (Shinde and Shah, 2018). ANNs were then adapted further to introduce deep layer and hence the introduction of Deep Learning.

ML has successfully developed the world of artificial intelligence into a true hope for near-human intelligence. Machine learning methods are often split into supervised learning used for classification and regression (Shinde and Shah, 2018; Nichols, Herbert Chan and Baker, 2019) or unsupervised learning methods used for clustering and feature engineering (Libbrecht and Noble, 2015). Where supervised learning depends on our prior knowledge and labelled examples to form an understanding of the model; unsupervised learning aims to learn some hidden structure using feature extraction of the unlabelled dataset. Whilst both forms of learning have greatly advanced their respective fields and widened the scope of artificial intelligence; they fall victim to the curse of time. Overlooking the effect of time can have grave consequences when implementing ML models to sensitive and stochastic applications which is often the case with engineering problems such as urban water management. Hence, the need of a learning approach that incorporates the hidden dimension of time – Reinforcement Learning. **Figure 2-1** highlights the place of RL as a subfield of machine learning. RL's ability to consider the effects of time through semi-supervised learning was the first expression of artificial foresight in machine learning and its closest form to human intelligence.

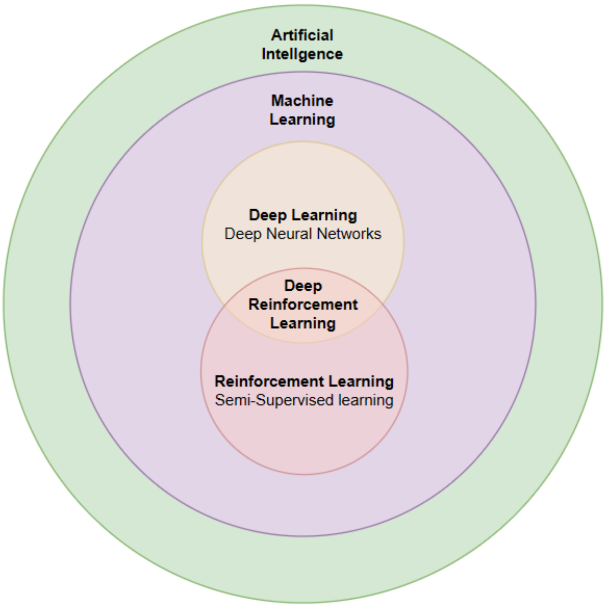

Figure 2-1 *The subfields of machine learning*

In its infancy, the use of reinforcement learning (RL) was an exciting concept that promised an introduction to responsive and continuously-learning AI systems. A behaviourist mathematical approach for experience-driven learning was finally attainable through RL (Sutton and Barto, 2018).This entails a reward-driven learning from interaction with an unmapped environment rather than hard computing or supervised learning where it is near difficult to obtain examples of desirable behaviour. Despite the initial successes of RL (Tesau and Tesau, 1995; Singh *et al.*, 2002; Kohl and Stone, 2004), it could not escape the 'curse of dimensionality' when applied to real life problems. RL was limited by complexity issues ranging from memory complexity, computational complexity and sample complexity (Strehl *et al.*, 2006).

The recent surge of deep learning and deep neural networks that has spearheaded the movement in function approximation and representation learning giving hope to unlock the true potential of RL by overcoming the issues of scalability; hence the rise of the field of DRL. This technology gained the interests of companies such as Google and Tesla during their race for driver-less vehicles (Kool, Van Hoof and Welling, 2018; Nazari *et al.*, 2018). It has lent its abilities to the field of robotics (Levine *et al.*, 2016; Nguyen and La, 2019; Zhao, Queralta and Westerlund, 2020), gaming (Mnih *et al.*, 2015a; Silver *et al.*, 2016) and many more sectors (Li, 2017). As deep reinforcement learning gained popularity and developed further, the field of reinforcement learning was quickly populated with novel algorithms. The field of RL has quickly transformed to a forest of methods, architectures and concepts that are difficult to navigate because of its non-modularity. Defining the scopes of RL (and DRL) will help researchers understand the trade-offs involved with algorithm design. Similar work surveying offline reinforcement learning methods with a taxonomy can be found in (Prudencio, Maximo and Colombini, 2022). To highlight the diversity in RL and DRL, we have gathered and classified a novel taxonomy of the algorithms (**Figure 2-2**). This classification tree can serve as a map to new researchers interested in the field of DRL. It classifies the algorithms based on model free vs model based; on policy vs off policy; value-based vs policy-based; gradient based vs gradient free labels. Dotted lines are used to label fields of DRL methods such as dynamic programming, Monte Carlo, temporal difference and distributional RL algorithms. In addition, RL fundamental algorithms are written green, RL methods are in blue and DRL algorithms are written in black. The classification tree aims to introduce a variety of DRL algorithms and methods that might be useful for application in urban water systems.

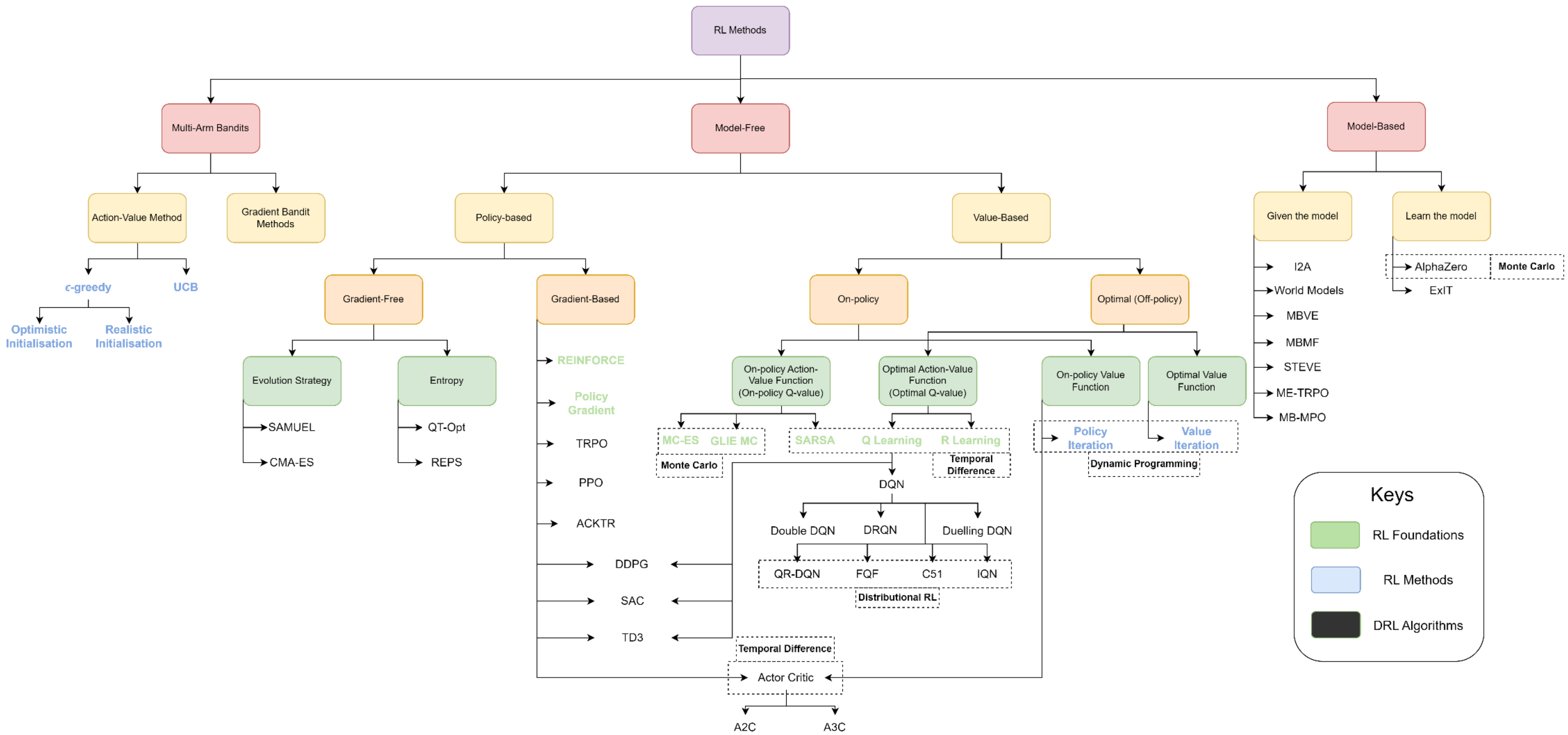

Figure 2-2 *Taxonomy of reinforcement learning algorithms.*

2.1. The Components of DRL

To fully comprehend the aspects and range of methods available in DRL, it is crucial to delve into the formalism that make the RL paradigm. Reinforcement learning tackles its problems as Markov Decision Processes (MDPs) which is a commonly used description in the field of computing that depict real word processes. MDP formalism is based on evaluating the probability of transitions between different states in its process and is sometimes denoted with the five tuple (S,A,P,R,γ) that stand for states (S), actions (A), probabilities/dynamics (P), reward (R) and initial state (γ) (Puterman, 1990; Desharnais *et al.*, 2004). This helps evaluate the sequential interactions between actuators (agents) and their environment to influence both the state of the agent (state, S) and the relevant state of the environment (observation). The agent is then fed the observation data and a reward signal (Reward, R) that serves as an assessor to the new state that this action has led to. The aim of the agent is to find the optimal policy (π) that will maximise the expected reward which is achieved by learning the probability of state transitions attached to a state-action pair. A visual description of this process can be found in **Figure 2-3**. The deep neural network is an addition only found in DRL methods whilst RL methods tend to use a tabular data frame. The components of RL and DRL can be therefore redefined to suit most real-world applications in an organic and straightforward manner.

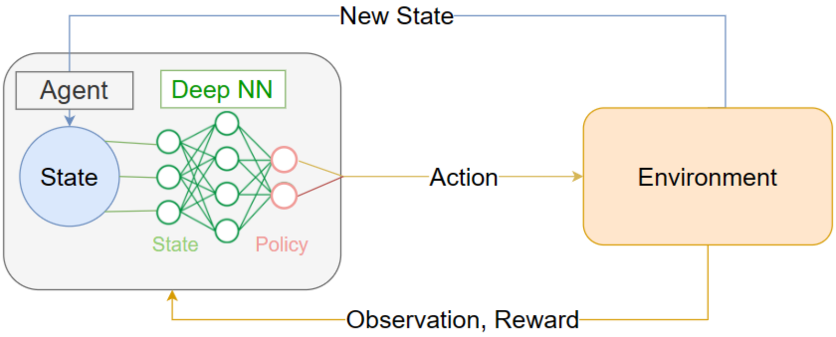

Figure 2-3 *Standard Deep Reinforcement Learning Schematic*

2.1.1. Reward and Return

The reward (r) is the crucial identifier that tells the agent whether their action was beneficial or harmful. The cumulative reward over a trajectory is named the return (R(τ)) and it can be a finite-horizon undiscounted return (Eq. 2-1) or an infinite-horizon discounted return (Eq. 2-2). Finite return is the sum of rewards for a fixed number of steps whilst infinite returns, like the name suggests, is the summation of the sum of all the rewards ever. The infinite returns must include the discount factor γ Є (0,1) used to control how much weight should be placed on the agent's foresight. This helps the infinite sum converge to a finite value.

$$R(\tau) = \sum_{t=0}^{T} r_t. \text{ For finite-horizon undiscounted return.} \quad (2\text{-}1)$$

$$R(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t. \text{ For infinite-horizon discounted return.} \quad (2\text{-}2)$$

This return is usually modified and incorporated into a value function for value-based RL methods or an objective function for policy-based RL methods. Both methods have their advantages and disadvantages; for example policy-based methods are generally less sample efficient than Value based algorithms but can learn stochastic policies and converge faster than their alternative (Lapan, 2019). We discuss this further in the classifiers section below.

2.1.2. Value Based

Value functions are used in almost every RL algorithm. They are a fundamental concept in RL which calculates the expected infinite horizon return to evaluate how beneficial individual states or state-action pairs are. Value functions that solely evaluate the current state without the action are often denoted by the symbol V(s) and named state value functions (Eq. 2-3). Alternatively, state-action value functions are called quality functions, and they provide more of an insight on the trajectory of the agent given its current state-action pair (Eq. 2-4). The Q-value is denoted by the symbol Q(s,a).

$$V(s) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+1+k} \,\middle|\, S_t = s\right] \quad (2\text{-}3)$$

$$Q(s,a) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+1+k} \,\middle|\, S_t = s, A_t = a\right] \quad (2\text{-}4)$$

Where $\mathbb{E}[.]$ is the expected discounted infinite horizon return, s is the state sampled from S_t, a is the action sampled from A_t and t is any time step.

An important property of RL is foresight which enables agents to weight the future consequences of their actions using the expected return hence it is rare to find value functions operating without the incorporation of the bellman equations (Bellman, 1952). Bellman equations are self-consistency equations integral to dynamic programming and MDPs that follow the concept that the value of any starting point is the reward you expect from being at the starting point in addition to the value of the next point (Bellman, 1952; Puterman, 1990). Because the actions taken by an agent depend on the policy that it follows, value functions are often described in relation to its policy. On-policy value functions estimate the expected returns as the agent follows the behavioural policy (π). On-policy value functions can either evaluate a state (state-value function) or a state-action pair (state-action value function or quality function). On-policy state-value functions are denoted by $V^{\pi}(s)$ and evaluates the expected return as the agent acts under behaviour policy (π) and starts with state (s) and is followed by the state (s'). The bellman equation decomposes the value function to the sum of the current value and the future discounted values. Similarly, the Q-value denoted by ($Q^{\pi}(s,a)$) bellman equation is formally defined as the expected return as the agent acts under the behavioural policy (π) starting with the state-action pair (s,a) and followed by the next state-action pair(s',a').

When attempting to find the optimal policy and action for a RL problem, off-policy value functions are used to remove the restrictions of the behavioural policy and allow the agent to explore the value function following the optimal policy This leads to the off-policy state value function and off-policy state-action function. These are also called the optimal value functions ($V^*(s)$ and $Q^*(s,a)$). The main difference between the on-policy and optimal bellman equations is that the optimal uses the maximum rewardable action as shown in the equations below (Eq. 2-5, Eq. 2-6).

$$V^*(s) = \max_{a} \mathbb{E}[r(s,a) + \gamma V^*(s')] \tag{2-5}$$

$$Q^*(s,a) = \mathbb{E}[r(s,a) + \gamma \max_{a'} Q^*(s',a')] \tag{2-6}$$

The optimal action of an RL problem can be extracted by finding the maximum reward argument of the off-policy state-action value function bellman equation (optimal Q-function). In instances where there are multiple optimal actions, the algorithms often select an action at random (Achiam, 2020). Another method to evaluate the value of an action is by using the advantage function (A(s,a)) . This compares how beneficial an action is to the average value of all actions by subtracting the state value from the state-action value under policy (π) (Eq. 2-7).

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s) \tag{2-7}$$

The use of advantage function is intuitive as it evaluates the performance of actions relative to an average. It is simpler to compare the consequence of an action with respect to another. Learning the advantage, rather than the quality or state function, has been a recent trend in DRL algorithms (Schulman *et al.*, 2015; Wang *et al.*, 2015; Gu *et al.*, 2016; Mnih *et al.*, 2016). For more details on the basics of value functions, we recommend the following introductory books, papers and articles (Arulkumaran *et al.*, 2017; Li, 2017; Sutton and Barto, 2018; Achiam, 2020).

2.1.3. Policy Driven

Other than value-based algorithms, there are policy driven techniques to solve the reinforcement learning problem and reach an optimal policy. Whilst the value-based methods use a learnt value functions to reach an implicit policy, policy-based methods do not use a value function but directly learns a policy. The value function approach often works well but it is important to be aware of its limitations. Value functions' approach to policy optimisation is focused mostly on deterministic policies which is rare in the real world since optimal policies are often stochastic. They also are subject to high sensitivities as a minor change in the expected value of an action might cause the algorithm to accept or reject it. This has been identified as a key fault that inhibits the convergence of value-based methods such as Q learning, SARSA and dynamic programming methods (Baird, 1995; Gordon, 1995; Bertsekas, Tsitsiklis and Τσιτσικλής, 1996). Policy driven methods bypass these limitations leading to better convergence properties, ability to learn stochastic policies hence more effective algorithms for higher dimensional and continuous action spaces (Sutton *et al.*, 2000). However, these methods can habitually converge to local minimums and are more computationally demanding with higher variance.

Direct policy search methods fine tune a vector of parameters (θ) to select the best action to take for policy $\pi(a|s,\theta)$. The policy π_{Θ} is updated to find the maximum expected return. They can either employ gradient free or gradient based optimisation. Gradient free algorithms often use the concepts of evolution strategies (Gomez and Schmidhuber, 2005; Koutník *et al.*, 2013; Salimans *et al.*, 2017) or the cross entropy function (Kalashnikov *et al.*, 2018). Gradient-free optimisation methods can perform well in low dimensional spaces and update non-differentiable policies but, despite some successes in applying them to neural networks, the favoured method remains gradient-based training for DRL algorithms. Gradient based training methods are more sample efficient when dealing with high parameter policies (Arulkumaran *et al.*, 2017).

The gradient-based policy methods, also called policy gradient, optimise a selected objective function ($J(\pi_{\theta})$) which can be defined by the average reward formulation or start-state formulation (Sutton *et al.*, 2000). Policy function approximation is challenging since gradients cannot be used through samples of a stochastic function hence why use a gradient estimator; the theory of the REINFORCE algorithm (Williams, 1988, 1992; Sutton *et al.*, 2000). The objective function (J) of the parameterised policy (π_θ) is the expected average return (R) under trajectory (τ). The trajectory is defined by parameterised policy.

The aim is to optimise the policy through gradient ascent by numerically defining the gradient of policy performance ($\nabla_\theta J(\pi_\theta)$) also called the policy gradient. A full derivation of the policy gradient can be shown in (Achiam, 2020) however the policy gradient can be redefined as (Eq. 2-8).

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}\left[\sum_{t=0}^{\mathrm{T}} \nabla_\theta \log \pi_\theta(a_t \mid s_t) R(\tau)\right] \quad (2\text{-}8)$$

Where the policy gradient is the expected sum of returns (R(τ)) multiplied by the gradient of the log of the parameterise policy ($\nabla_\theta \log (\pi_\theta (a_t|s_t))$) for timesteps (t) in episode length (T). This is the simplest policy gradient; there are different variations of the policy gradient definition like the Expected Grad-Log-Prob Lemma (Schulman *et al.*, 2015; Achiam, 2020).

Policy-based and value-based RL coincide at the actor-critic algorithms (A2C, A3C, AC, DDPG, SAC) where the actor performs and action using policy-based RL and the critic evaluates the resulting reward using a value function. The critic influences the actor using temporal difference error (TD error) to improve the algorithm's performance.

2.1.4. Other DRL Algorithm Terminology

To fully comprehend DRL algorithms, it is necessary to explain the parlance and methods that form those algorithms. One way to describe DRL algorithms is whether the agent is provided with a state transition function (model-based) or having to learn solely from experience through trial and error (model-free). Agents that have access to a model make use of sample efficiency and display a heightened ability of foresight but can often underperform when applied in real-world applications due to discrepancies between the model used for training and the ground-truth model. Model free methods can be implemented and easily tuned to real world application (Li, 2017). Algorithms can also be trained on sequentially generated data (online mode) or on a pre-set training batch (offline mode).

A commonly used label for RL is whether it is on-policy or off policy. On policy methods evaluate or improve the behavioural policy of the current action-value pair of the current policy (e.g. SARSA) whilst off-policy methods explore the best value policy without necessarily following the current behavioural policy; they are also called optimal methods (e.g. Q-learning) (Arulkumaran *et al.*, 2017; Li, 2017). The value functions used to achieve were highlighted previously.

2.2. Notable DRL Algorithms

Many successes have stemmed from scaling RL using deep neural networks through function approximation. Deep neural networks can be used to approximate the optimal policy (π^*) or the optimal value functions (Q^*, V^*, A^*). In this section, we discuss the current trends and notable deep reinforcement learning algorithms that have progressed the field. This will help contextualise the current state of the research field and expose any future work.

The timeline and milestones that led to the creation of DRL was well illustrated in (Nguyen, Nguyen and Nahavandi, 2020, fig. 1) showing how trial and error learning, TD learning and deep neural networks came together to incentivise the first deep reinforcement learning algorithm – the deep Q-network (DQN). DQN was first introduced by Mnih et al. as they used convolutional neural networks (CNN) to feature engineer images from a series of 49 games (Mnih *et al.*, 2015a). It was then used to tackle MuJoCo physics problems (Duan *et al.*, 2016) and three-dimensional maze problems (Beattie *et al.*, 2016). Following the success of DQN, researchers have built on the existing DQN architecture to improve its performance hence creating new algorithms such as Double DQN (DDQN) and Duelling DQN (D-DQN). Double DQN minimises the effect of noise on DQN by avoiding the overestimation of Q values (Van Hasselt, Guez and Silver, 2016) whilst the duelling network architecture combines two streams of data (the value stream and advantage stream) to produce a more accurate Q function (Wang *et al.*, 2015).

Another milestone was the introduction of the Actor-Critic algorithms that combine the use of value functions and policy gradients to forego the trade-off of variance reduction in policy methods and bias introduction from value functions (Konda and Tsitsiklis, 1999; Schulman *et al.*, 2015). Quickly, the DRL research community has direct their efforts to improve the AC methods. Schulman et al. improves the actor using generalised advantage estimation (GAE) to produce better variance reduction baselines (Schulman *et al.*, 2015). The critic is also improved separately using target network in (Mnih *et al.*, 2015b). Introducing deterministic policy gradients (DPG) in actor-critic algorithms was first observed in (Silver *et al.*, 2014). DPGs allow the use of policy gradients in deterministic policies when they were initially exclusive to stochastic policies. This lowers the computational load as DPGs only integrate over the state space and can therefore tackle large action spaces using less sampling. Stochastic Value Gradients (SVG) are another method to apply standard gradients to stochastic policies by 'reparametrizing' (Kingma and Welling, 2013; Rezende, Mohamed and Wierstra, 2014). This trend was first introduced in (Heess *et al.*, 2015) and created a flexible method capable of being using with and without value function critics and models (Arulkumaran *et al.*, 2017). SVG and DPG provide algorithmic means of improving learning efficiency in DRL.

On the lines of learning efficiency, Google's DeepMind lab released the Asynchronous Advantage Actor Critic algorithm (A3C) (Mnih *et al.*, 2016). This advancement entails the use of an advantage function in an actor-critic architecture through training parallel agents asynchronously yielding high accuracy and applicable in continuous and discrete action spaces (Zhu *et al.*, 2016; Lapan, 2019) hence creating a trend for asynchronous and parallel learning.

2.3. Current DRL Trends

The field of DRL is growing exponentially as researchers ground their understanding of reinforcement learning in human psychology. Using methods that parallel our natural learning trends has helped develop DRL methods further leading to fields such as inverse reinforcement learning (IRL). Moreover, there is more effort on improving algorithms by modelling the reward as a distribution of values similar to our brain's reward system (Dabney *et al.*, 2020). Multi agent reinforcement learning (MADRL) models the real-world nature of multiple agents interacting with the same environment and reward probability. In this section of the review, we focus on current trends in the field of deep reinforcement learning. We explain the recent advancements and highlight notable work and challenges that are being addressed.

Hierarchical Reinforcement Learning

As the field of DRL grows, researchers have learnt how to include biases into the algorithm's learning experience. Hierarchical reinforcement learning (HRL) is a field of DRL dedicated to introducing inductive biases by factorising the final policy into several levels through state or temporal abstractions. This approach allows algorithms to tackle higher and lower level goals simultaneously by allowing top-level policies to focus on the main goal and sub-policies to focus on fine control (Tessler *et al.*, 2017; Vezhnevets *et al.*, 2017). This is how HRL attempts to achieve compositionality; achieving new representations by the combination of primitives (Hutsebaut-Buysse, Mets and Latré, 2022). The challenges faced in HRL stem from the selection of sub-behaviours or policies and how to efficiently learn state abstractions.

Inverse Reinforcement Learning

As humans, we can often learn from others' mistakes and successes. Similarly, researchers have developed methods to bootstrap the learning process using trajectories from other controllers. This is known as imitation learning (also known as behavioural cloning). The success of behavioural cloning lead to the success of an autonomous car using ALVINN in (Pomerleau, 1989).The main challenge with imitation learning is its susceptibility to uncertainties. Imitation learning's inability to adapt can lead the agent down a destructive trajectory hence why it is paired with reinforcement learning. Using RL, the policy can fine-tune whist imitation learning guides the general learning leading to faster convergence properties and better stability properties. Introducing behavioural imitation to DRL births the field of inverse reinforcement learning (IRL). IRL applies behavioural cloning by relying on provided trajectories for the desired solution to approximate the reward function (Ng and Russell, 2000). Intuitively, the motivation behind using IRL usually includes learning behaviour from experts, assisting humans and learning about systems (Adams, Cody and Beling, 2022). Application of IRL are mostly concerned with teaching robots to imitate experts (Adams, Cody and Beling, 2022). Notable work and algorithms in this field include (Ziebart and Fox, 2010; Finn, Levine and Abbeel, 2016; Ho and Ermon, 2016; Levine and Van De Panne, 2018; Paine *et al.*, 2018; Peng *et al.*, 2018).

Distributional Reinforcement Learning

Distributional RL grounds itself in our natural brain reward system (Dabney *et al.*, 2020). Like our natural dopamine system, DRL displays returns as a value probability distribution learned from interacting with the environment. This parallel between distributional RL and our brains opens up opportunities for collaboration between AI and neuroscience (Lowet *et al.*, 2020). This new method of value distribution has shown its usefulness in improving learning speed and stability. The original distributional reinforcement learning algorithm is the categorical DQN (C51) (Bellemare, Dabney and Munos, 2017) where using value distributions the authors have surpassed most gains on the Atari2600 environment thus beating the benchmark DQN and DDQN. Other algorithms include quantile regression DQN (QR-DQN) which uses quantile regression to minimise the Wasserstein metric and improve greatly on the previous C51 in the Atari 2600 (Dabney *et al.*, 2017). Implicit quantile regression (IQR) and fully parameterised quantile function (FQF) are the latest algorithms in distributional RL and they build further on the foundations of QR-DQN (Dabney *et al.*, 2018; Yang *et al.*, 2019).

Multi Agent Reinforcement Learning

With the rising complexity of real-world systems, deep reinforcement learning algorithms often play catch-up to be able to process and scale their models. Most of the methods devised for DRL algorithms aim to simplify complex environments and feature extraction. On the other hand, multi agent DRL introduces complexity in its algorithms by introducing several agents in the algorithms that simultaneously interact with the environment. This represents having multiple employees working as a team to carry out a desired goal (or policy) on the same system. The complexity of the algorithms brings forth multiple challenges that are currently the focus of the research community with the promise to solve more complex environments and real-world problems. There have been different approaches to tackle MADRL including sending signals to the agents, having bidirectional channels between the agents and an all-to-all channel (Arulkumaran *et al.*, 2017). Major challenges in the field stem from non-stationarity, partial observability, complexity in training schemes, application in continuous action spaces and transfer learning (Nguyen, Nguyen and Nahavandi, 2020). Previous reviews and surveys include (Nguyen, Nguyen and Nahavandi, 2020) that provides a review of MADRL challenges, solutions, applications and perspectives;

(Buşoniu, Babuška and De Schutter, 2008) evaluates stability and a taxonomy of MADRL algorithms; (Bloembergen *et al.*, 2015) surveys dynamical models devised for multi agent systems; (Hernandez-Leal, Kartal and Taylor, 2019) bridges the gap between DRL and MADRL including benchmarks for MADRL. Other notable reviews include (Da Silva, Taylor and Costa, 2018; Hernandez-Leal, Kartal and Taylor, 2018).

3. Urban Water Systems (UWS)

Urban water systems are a collection of complex infrastructure and processes that supply, treat, transport, and manage water and wastewater within urban environments. These systems are crucial for managing the supply of clean drinking water as well as treating wastewater and controlling storm water. Henceforth, they are paramount for the sustainability and well-being of cities. Effective management of UWS through sustainable practice aims to ensure a resilient supply of clean water despite climate change and seasonality. It should also minimise water loss through leakage and energy consumption through inefficient water supply and distribution. The key processes in UWS can be split into four major systems which are raw water treatment plants, water distribution networks, wastewater treatment plants, and stormwater systems (Loubet *et al.*, 2014; Etikala, Madhav and Somagouni, 2022) . Some of the processes involved in each function are displayed below in **Figure 3-1**.

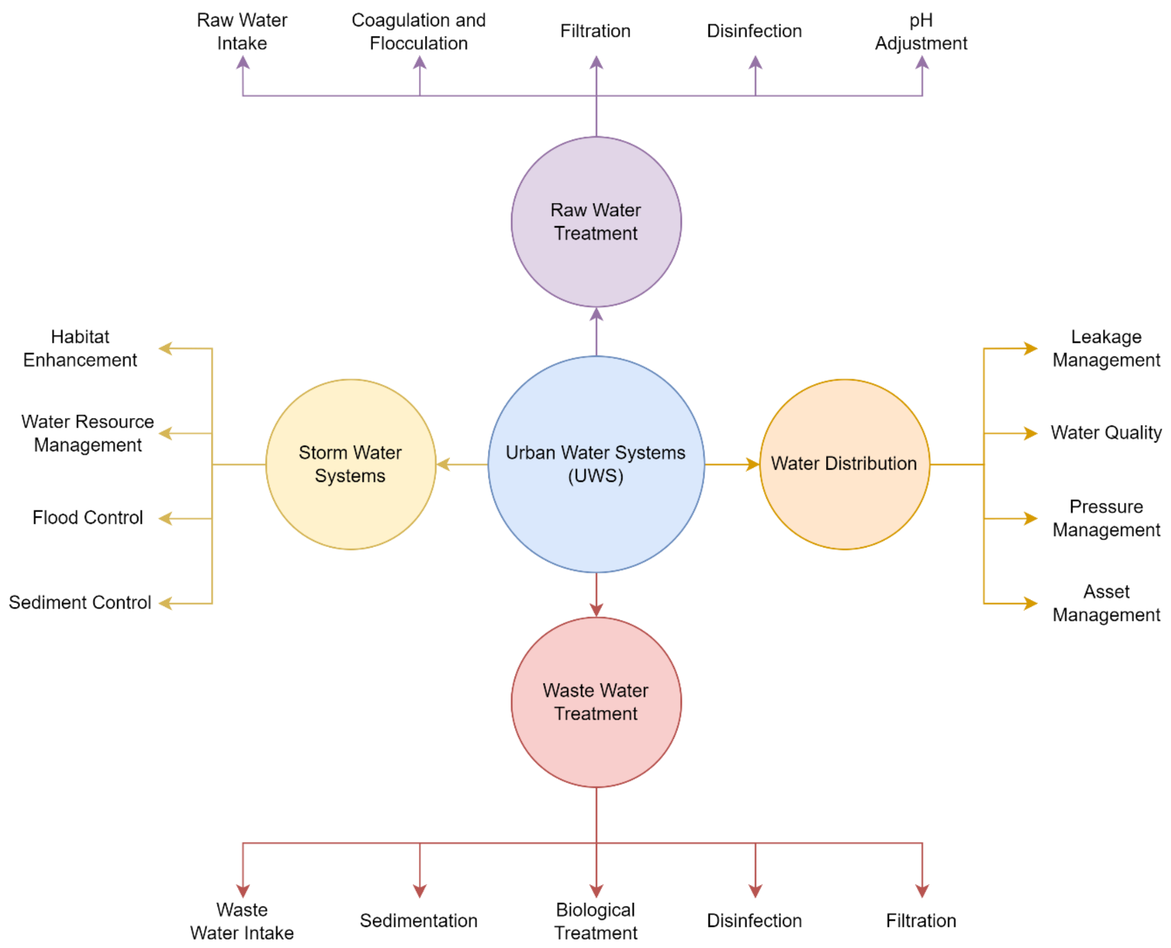

Figure 3-1 *Urban Water Systems*

Urban areas often obtain their water from several resources such as rivers, lakes, groundwater and desalination plants which are managed by raw water treatment plants. Raw water goes through several treatment processes to remove impurities, and contaminants. The main treatment methods used in raw water treatment plants include screening through mesh filters or screens, coagulation, flocculation, sedimentation, filtration, disinfection, corrosion control, pH adjustment, fluoridation, and quality monitoring (Benjamin, 2014; Jiang, 2015; Teodosiu *et al.*, 2018; Lipps, Braun-Howland and Baxter, 2022).

Once treated, clean water is distributed from the plants to the customers through a network of pipes, valves, pumps and reservoirs. This process requires advanced pressure and asset management to minimise leakage and contamination. Due to the varying elevations, demand and climate change, the distribution of water increases in complexity and leakage has become a natural phenomenon in water distribution networks (Xu *et al.*, 2014; Barton *et al.*, 2019).

Similar to raw water treatment, wastewater treatment plants are concerned with treating wastewater collected through a sewer pipeline network. Treatments include a variety of physical and chemical processes. Physical methods of screening, grit removal, sedimentation, and filtration remove heavier contaminants and large contaminants. Water is then treated biologically in the secondary treatment by using microorganisms to break down organic matter in wastewater (Hussain *et al.*, 2021). Coagulant and flocculants help remove fine particles and dissolved contaminants during the tertiary advanced chemical treatment. A final step of disinfection could use chemicals such as chlorine and UV to remove harmful pathogens (Kentish and Stevens, 2001; Crini and Lichtfouse, 2019).

During detrimental events such as floods and storms, stormwater management controls the impact on the environment and infrastructure (Ahiablame and Shakya, 2016; Aryal *et al.*, 2016; Jefferson *et al.*, 2017). Stormwater management deal with several high-level objectives such as flood control, water quality monitoring, erosion/sediment control, groundwater recharge (Jotte, Raspati and Azrague, 2017).

3.1. Challenges and Opportunities in Urban Water Systems

UWS include a wide range of processes that are riddled with unique dependencies and impacting factors. However, the preservation and use of water is a holistic process that incorporates the wider ecosystem, climate, and wildlife as much as human use. Understandably, UWS share challenges that stem from external factors and opportunities to adapt deep reinforcement learning techniques. In this section, common current challenges that plague UWS processes are discussed and how DRL can provide innovative solutions. This is followed by challenges that researchers might encounter when applying DRL algorithms to UWS.

High trends of urbanisation globally increase the stress and demand on UWS with 60% of the world's population expected to live in urban areas by 2030 (UN-Water, 2012). This rise in demands causes heavier loads and more uncertainty throughout all processes in UWS due to increased supply and network expansions (Sharma *et al.*, 2010). Navigating these uncertainties can be challenging for meta-heuristic decision making algorithms (Maier *et al.*, 2014) in comparison to DRL algorithms that learn from experience and are able to act in real time (Fu *et al.*, 2022). DRL provides a method for managing uncertainties that outperforms traditional decision-making algorithms and can learn from experience which allows it to adapt to the rise in urbanisation.

Another challenge that plagues UWS is the energy consumption and carbon emissions associated with operating water systems (Nair *et al.*, 2014; Xu *et al.*, 2014). It was estimated that 1-18% of all energy consumed in urban areas is due to UWS (Olsson, 2012) which in return produces a lot of carbon emissions. The negative effects of high energy consumption lie beyond the financial impacts as it promotes climate change and global warming. The circular effect of carbon emissions, water scarcity and energy consumption is displayed in the water-energy-green house nexus (Nair *et al.*, 2014, fig. 1). DRL has had a proven record of improving energy management within the water systems (Hernández-Del-olmo *et al.*, 2016; Hernández-del-Olmo *et al.*, 2018) and in system efficiency (Kılkış *et al.*, 2023).

UWS often deal with a heterogeneously aging infrastructure that add to the complexity of asset health management. The aging pipes, pumps, valves, and other system components can lead to high non-revenue water and effect the systems' overall resilience. Hence why, it is essential to provide decision making algorithms that can deal with high-level dependencies and complexities. A challenge that manifests with decision making algorithms is the high computational costs associated with this complexity thus why deploying DRL agents can benefit UWS as they rely on function approximators to lower the computational load (Sutton and Barto, 2018). Furthermore, asset management for UWS operations can be achieved by leveraging DRL for optimal design, strategic planning and predictive maintenance (Fu *et al.*, 2022). This area of research requires more experimentation and social proof despite its clear advantages.

In most pipeline infrastructure, it is necessary to quantify leakage and asset health. Managing leakage effectively is an ongoing battle that effects UWS especially water distribution systems. The use of DRL for leakage management is an unrealised opportunity but has been recommended by reviews and surveys (Moselhe *et al.*, 2020; Fu *et al.*, 2022). The use of a tabular Q-learning method for leakage reduction using pressure management in water distribution networks was tested in (Negm, Ma and Aggidis, 2023b) and whilst the results were positive, it was clear that using DRL would enhance it further and overcome the curse of dimensionality.

3.1.1. Challenges of DRL in UWS

Building DRL algorithms is a science. In this section we build on the challenges and trade-offs underlined in the previous sections inherent in algorithm design. It is crucial to note that the field of RL research, much like the algorithms, has been expanded by experience followed by theory. In essence, some challenges were identified but not completely understood such as the deadly triad issue (Sutton and Barto, 2018).

In DRL algorithm design, most researchers will make use of some form of function approximation, bootstrapping or off-policy. Function approximation uses examples to generalise an entire function hence it aids with the scalability and generalisation issue that riddles tabular algorithms and is the main tide driving the success of deep neural networks in reinforcement learning (DRL). On the other hand, bootstrapping used in DP and TD fields help with improving the algorithm's data efficiency, hence reducing computational loads. Finally, off-policy methods free our agent from target policy to explore optimality. Separately, each of these methods help RL researchers reach their desired benefits and design a better optimisation algorithms, however when combined the same methods induce instability and divergence – the deadly triad issue (Tsitsiklis and Van Roy, 1997; Sutton and Barto, 2018). This instability can be detrimental when controlling urban water management system and could result in undesirable states. Issues rising from instability often spill into sub-optimal policy development which leads to low performing algorithms. In addition, this could lead to weak dependencies between the observation data and the action space forming unresponsive algorithms. In UWS, this would echo as low performing water systems affecting their resilience and ability to handle change. Further implications depend mostly on the system being managed for example in water distribution, which could mean supply interruptions or pressure limit violations. Ensuring stability and resilience should be a primary goal of DRL design.

Another common challenge is the 'credit assignment problem'. This refers to the notable phenomena of incorrectly evaluating the credit of the action due to unclear or unforeseeable consequences manifesting later (Arulkumaran *et al.*, 2017). These long-term dependencies are necessary to allow the agent to better comprehend the value of its action. Hence, value functions have been modified to incorporate the estimated subsequent rewards and they have been discounted to signify the dwindling nature of consequence (Eq. 2-5 & 2-6). UWS applications tend to be connected through both short-term and long-term dependencies therefore it is importance to include these consequences in the DRL algorithm's learning strategy. UWSs are complex and interconnected systems, and the consequences of specific actions may not be immediately apparent. Unforeseeable impacts on water quality, pipeline integrity, or energy consumption may manifest over time. In addition, UWS are often dynamic with changing environments which will further emphasise the effect of the credit assignment problem when attempting to navigate the evolving nature of UWS.

Finally, the exploration versus exploitation dilemma. This problem riddles most RL (and DRL) algorithms as agents tend to behave in a reward greedy manner. Since the agent's observation depends on its actions and its actions depend on the reward generated; RL agents can find themselves in a loop around a local optimum rather than finding the global optima - exploitation. Ultimately, the only way to solve this is to introduce randomness to the agent's behaviour hence allowing the agent to receive new observations and possibly lead it to the global optima – exploration. This trade-off in agent behaviour has been navigated in many ways and the simplest is the use of ε-greedy exploration policy where the agent acts randomly with probability $\varepsilon \in [0,1]$. The value of ε decreases as time passes leading the agent to a more exploitative nature as it learns. For continuous control, more complex methods have been used to introduce randomness over time to preserve momentum (Lillicrap *et al.*, 2016; Arulkumaran *et al.*, 2017). Other methods to tackle the exploration-exploitation dilemma include Osband et al.'s bootstrapped DQN using experience replay memory (Osband *et al.*, 2016), Usuneier et al.'s exploration in policy space (Usunier *et al.*, 2017) and upper confidence bounds (UCB) (Lai and Robbins, 1985; Arulkumaran *et al.*, 2017; Pathak *et al.*, 2017). Managing the exploration-exploitation trade-off should be bespoke to each UWS application to ensure that agents don't converge at sub-optimal policies. If not managed properly, the exploration-exploitation dilemma could affect UWSs manifest in operational inefficiencies. This is particularly critical in regions where water resources are scarce, and efficient use is imperative.

These challenges are inherent in most RL problems and navigating them is a skill necessary to develop an effective DRL algorithm. The application of DRL in UWS include specific limitations such as its reliance on clear data. Data-driven optimisation tends to be insightful nevertheless it requires sensor data across the entire network. UWSs vary in their data availability and data quality which could limit the usability of DRL algorithms. Therefore, this study is best applied to UWSs that have established a coherent data pipeline and are looking to expand their facilities. Consequently, it is important to build accurate models/data pipelines that can be used to build the DRL agents. Well-developed DRL models also tend to be quite sensitive to erroneous observation data which could falsely trigger harmful actions by the pressure valves. The DRL input data must be cleaned and tested for accuracy to ensure that it represents the current state of the system.

Furthermore, the application of DRL requires reliability evaluations before being deployed on UWSs. It is necessary to ensure that the optimisation algorithm won't endanger the customers/water system. For example, in WDN, agents need to ensure that water supply remains uninterrupted without affecting asset life or risking future bursts. These concerns were covered by (Tian, Liao, Zhi, *et al.*, 2022) where the authors devised a 'voting' method to improve reliability. Most UWSs are subject to daily and seasonal changes that will undoubtedly influence the performance of the DRL models. While the DRL algorithms were proven to deal with randomness in the observation data, seasonal changes might require re-training of the models and further policy development. This could be achieved through a continuous integration/deployment (CI/CD) pipeline for the DRL models which automates the deployment of newer, more suitable models.

Limitations also include the effect of the DRL algorithm on designing a reward function that incorporates multiple objectives. Most UWSs control tasks require the optimisation of multiple objectives as they influence each other hence why any relevant objectives should be included in the reward formulation design to ensure that the agents are trained with a complete picture of the desired behaviour. Complex model design is not limited to the selection of the reward function but includes DRL sensitivity to hyperparameters and neural network architecture. The design of DRL algorithms involve many decisions including various options for neural network architectures, optimisers, activation functions, pre-training techniques, and hyperparameters. The complexity of making these design choices require careful consideration and experimentation. Furthermore, generalisation of the DRL models is limited since the policy developed for one network may not necessarily work for another therefore it is important to develop a separate model for each network. On another hand, the option for transfer learning between the neural networks is valid as that could help train models from different networks.

The risks associated with DRL issues stem from unreliable sub-optimal control. This could appear as concerns with water quality. Unanticipated consequences, such as changes in flow patterns or variations in water treatment processes, may lead to water quality issues that pose risks to public health. Other issues could arise from adjustments in water flow and pressure affecting the integrity of the pipeline infrastructure. Over time, actions that seem reasonable in the short term may contribute to pipeline degradation or leaks. The challenge lies in identifying the causal relationships between management decisions and the gradual deterioration of the infrastructure. UWSs often require energy for pumping, treatment, and distribution processes. Management decisions that impact system dynamics can influence energy consumption. Unforeseen consequences may lead to suboptimal energy use or inefficiencies in the system, affecting both operational costs and environmental sustainability. Further implications are bespoken to the application of DRL and would appear with testing.

3.2. DRL Research in UWS

In essence, there are many parameters to consider when selecting a DRL algorithm but through careful consideration of selecting the correct DRL components and algorithms. Depending on the optimisation objective, the agent's nature (pump, valve, etc.) and requirements (nodal pressures, head measurements, pump speed, etc.) would vary. In a critical review of deep learning in the water industry Fu et al. mentioned the applicability of DRL in water distribution networks (WDN) and urban wastewater systems (Fu *et al.*, 2022). In (Croll *et al.*, 2023), the applications of reinforcement learning techniques in wastewater treatment were reviewed with a few studies utilising DRL methods. Otherwise, there are no mentions or reviews published on DRL algorithms in UWS research. There is limited literature on the application of DRL in UWS where most research relate to stormwater systems, water distribution networks and a few publications in wastewater systems. This shows a massive gap in the research field and an exciting journey for researchers in UWS at the cusp of realisation. In this section we will review the available literature on deep reinforcement learning in urban water systems.

3.2.1. DRL in Water Distribution

In article (Hajgató, Paál and Gyires-Tóth, 2020), the authors use a Duelling Deep Q Network (D-DQN) to find the optimal pump speeds for hydraulic efficiency in randomly generated demands. The algorithm minimises the inflow and outflow of tanks whilst keeping heads within an acceptable range in all the nodes. The reward is calculated by evaluating the consumer satisfaction as the number of problematic nodes divided by the number of all nodes; the efficiency of the pumps as the product of standalone pumps divided by the product of theoretical peak efficiencies; the feed ratio by comparing the ratio of pumps supplying the water to the tanks and reservoirs supply. When compared to a test set of Nelder-Mead, Differential Evolution (DE), Particle Swarm Optimisation (PSO), Fixed-Step Size Random Search (FSSRS) and One-shot Random Trial; the agent performed at a comparable level to the differential evolution algorithm and much better than the rest of the test set. All the algorithms were tested on a small (Anytown) and large (D-town) WDN model. When using the one-shot random trial as a reference solution as a sub optimal policy; the agent reaches a better solution and moves off policy to overperform the DE algorithm. This technique relies entirely on live measurement data and can predict the best action in real-time making it the most suitable controller for real life application.

Hu et al. conducted a thorough experiment where they optimised the scheduling of fixed speed pumps to minimise the electric cost of the pumps and tank level variations whilst adhering to sensible hydraulic constraints using Proximal Policy Optimisation (PPO) and Exploration enhanced Proximal Policy Optimisation (E-PPO) (Hu *et al.*, 2023). Both DRL algorithms are policy-driven methods set out to find the best policy to achieve the highest rewards. They conducted three experiments that introduced three increasing levels of uncertainty to the consumer demand patterns using 0.3, 0.6 and 0.9 multiplier respectively on the Net3 test networks model. The results were compared with metaheuristics including genetic algorithms (GA), PSO and DE. GA converged after 100 epochs and were considered the optimal solutions (Hu *et al.*, 2023). They were followed in performance E-PPO followed by PPO, DE and PSO. The exploration enhanced policy saves approximately 6.10% of the energy cost with respect to PPO. Unlike the rest of the metaheuristic methods that require to be trained before each scheduling case; the DRL methods (PPO, E-PPO) can just call their trained models to act in a fraction of a second (0.4s) (Hu *et al.*, 2023).

(Xu *et al.*, 2021) tackles the pump scheduling optimisation problem in WDNs through combining knowledge learning and deep reinforcement learning in a knowledge assisted proximal policy optimisation learning (KA-PPO) (Xu *et al.*, 2021). KA-RL evaluates the state using historical nodal pressure data and a reward function. Pressure management objectives were placed to maintain junction heads within a specific range, minimise water age, and increase pump efficiency. The proposed algorithm was tested on the benchmark Anytown network to manage the performance of two pumps in the pump station. The results show that the algorithm performs favourably in comparison to the Nelder-Mead method and the DDQN algorithm used in (Hajgató, Paál and Gyires-Tóth, 2020; Xu *et al.*, 2021). Future work can improve the reward formulation process by including energy prices. The problem setup can also be modified to consider a continuous action space and long period accumulated return. The use of emulators and parallel computing can also minimise the training time.

In (Hasan *et al.*, 2019), the authors offer four novel contributions to the fields of dynamic multiple-objective deep reinforcement learning and water quality resilience applications. Based on the deep-sea treasure (DST) test bed, the authors develop a new test bed to fit the RL settings hence creating the first test bed accommodating for dynamic multi-objective DRL (DMODRL). They also devise a new for multi-objective optimisation using DRL and the first deployment of objective relation mapping (ORM) to construct the govern policy (Hasan *et al.*, 2019). The last contribution is an expert system to evaluate the water quality resilience (WQR) in Sao Paulo, Brazil. The proposed parity-Q deep Q network (PQDQN) algorithm proposed was tested in the two DST environments and the WQR model. In all three test beds, the PQDQN algorithm has outperformed the state-of-the-art multi-policy DRL algorithms which were multi-policy DQN (MP-DQN), multi-objective monte carlo tree search (MO-MCTS) and multi-pareto Q learning (MPQ). In all three test beds, the performance of the algorithms were assessed using the evaluation matrices generational distance measure (GD), inverted generational distance (IGD) and hypervolume (HV) (Hasan *et al.*, 2019). PQDQN managed priorities best using the ORM aiding its impressive performance and defeating the other multi-policy algorithms (MP-DQN, MO-MCTS, MPQ) (Hasan *et al.*, 2019). This work can benefit by experimenting with multi-agent DRL and integrating real-world scenarios to the WQR model. Parallel computing and GPU processors can also reduce training time. Hyperparameter optimisation may even improve the performance of the PQDQN algorithm further.

In a broader look on water systems, (Fan, Zhang and Yu, 2022) tackles asset management of water distribution networks post-earthquake. The problem setup involves four models that assess damages incurred by the earthquake, recover the water distribution network (WDN) using the optimisation algorithms, measure the WDN hydraulic performance using the performance degree (PDW) at each timestep, quantify the overall WDN resilience using the system resilience index (SRI). The chronological and iterative process between these models is clearly displayed in (Fan, Zhang and Yu, 2022, fig. 2). A graph convolutional network (GCN) was deployed as the function approximator for a DQN algorithm hence creating GCN-DQN. This selection was a great step towards better representation for water distribution networks since the graphical nature of the data requires a similar deep neural network architecture. Other strategies used for comparison included two greed search algorithms (static importance based and dynamic importance based), genetic algorithm (GA) and diameter-based prioritisation method. All five strategies were tested under three identical earthquake scenarios with different magnitudes. In all three scenarios the GCN-DRL model outperforms the other strategies by following repairing sequences that lead to higher SRI scores (Fan, Zhang and Yu, 2022). The importance-based methods cam second and third whilst the diameter-based prioritisation came last. In order to minimise the training computation time, the authors have used transfer learning to use the previous GCN weights on an old damage scenario to initialise the GCN weights for the new scenario. This reduced the computational load significantly and proved the scalability of the GCN-DRL model across all scenarios. Accommodating more sophisticated assumptions can be easily implemented to improve the GCN-DQN model's reliability and improve the problem setup. Applying this work on different test networks can further prove its generality and encourage more development of asset management through deep reinforcement learning.

3.2.2. DRL in Stormwater Systems

Mullapudi et al. provide a first look on the application of deep reinforcement learning for real time control in storm water systems (Mullapudi *et al.*, 2020). The authors test a simple DQN algorithm on the urban watershed in Ann Arbor as a benchmark test network. The problem setup involved agents taking actions to control valves status; water levels and outflows as states and an assumption of uniform rainfall and negligible base flow (Mullapudi *et al.*, 2020). The authors set out to test the stability of DRL algorithms in controlling storm water management models (SWMM) through controlling a singular basin and controlling multiple basins. Their research highlighted DRL algorithms' known sensitivity to reward formulation and deep neural network architecture. Even though the agent could have benefitted from a longer learning phase, the DRL proved useful in managing the single-basin SWMM scenario. Due to the increase in state and action space, controlling multiple basins was more challenging. The agent behaved favourably in comparison to uncontrolled SWMMs in both scenarios but were outperformed by the equal-filling algorithm. The authors remain determined that RL-based controllers need to be explored further and applied to SWMM in hopes of reaching a stable real-time controller. The results provided in this paper could be used as a starting point to compare more capable DRL algorithms A3C and advanced variations of DQN. Also, a more systematic method for reward formulation and neural network hyperparameter optimisation would greatly improve the scalability and stability of the model.

A common issue with real-time control using DRL is concerns of the reliability and uncertainty of its fluctuating actions in high-risk real-world cases. Tian et al.'s paper tackles this issue through a novel methodology called 'voting' (Tian, Liao, Zhi, *et al.*, 2022). Voting compares actions from five different DRL algorithms to select the safest and most rewardable action hence minimising the risk associated with DRL control. If none of the DRL agents provide a viable action, a backup user-defined rule-based action is executed. The methodology is used to minimise combined sewer overflow (CSO) and flooding in urban drainage system. The DRL algorithms used in this study are DQN, DDQN, PPO1, PPO2 and A2C. Voting uses a novel independent security system to evaluate whether the actions meet the user-defined safety requirements. All five DRL algorithms and voting algorithms are compared to a GA algorithm that was used as an upper bound performance reference by subjecting them to eight scenarios under different rainfall patterns. The results prove that voting avoids harmful actions to minimise risk hence improving the reliability of the real-time control. Figure 16 highlights that voting often draws its actions from PPO1 and never needed to use the backup action in all eight scenarios (Tian, Liao, Zhi, *et al.*, 2022, fig. 16). All DRL algorithms have performed well in this sequential problem and are therefore suitable candidates for CSO and flooding mitigation. Concerns of long training times and computational loads can be mitigated with parallel computing and an emulator for the stormwater model. The DRL algorithms can benefit from hyperparameter optimisation to improve the results further. Future work can also attempt deploying the voting algorithm on a SCADA system or online monitoring system to uncover uncertainties from real world applications.

It is worth mentioning that the authors published a different paper where they developed an emulator for the stormwater model to relieve the high computational load associated with training the DRL agents (Tian, Liao, Zhang, *et al.*, 2022). This emulator succeeded in decreasing the training time by 9 hours and 57 minutes hence improving data efficiency when compared to the regular RL-stormwater model approach.

Like the previous article, (Bowes *et al.*, 2021) leverages the power of DRL for flood mitigation. In this experiment, the authors developed a DDPG algorithm to create control policies that mitigate flood risks in the coastal city of Norfolk, Virginia. The DRL agent manages to balance flooding throughout the system and follow the control objectives of maintaining target pond levels and mitigating flood through controlling valves in the stormwater management model. The performance of DDPG as a DRL method was compared to rule-based control strategy, model predictive control and a passive system. In summary, the DDPG algorithm boasted a 32% reduction in flooding in comparison to the passive system and a 19% reduction with respect to rule-based control. The model predictive control strategy deployed an online genetic algorithm optimisation as in (Sadler *et al.*, 2020) to produce similar results to the DDPG algorithms (3% reduction in flood compared to DDPG). The model predictive control was too computationally expensive to run on the complete dataset whilst RL provided an 88x speed up in the creation of control policy (Bowes *et al.*, 2021). This research highlights the power of DRL in real-time control of stormwater systems and its ability to produce impressive results with a lower computational load. Further research should aim to recreate these results on real-world systems through RL controllers. Combining the different real-time control methods as decision support tools should be investigated to enhance stormwater systems.

3.2.3. DRL in Wastewater Treatment

Wastewater treatment has initially experimented with RL methods to manage the oxidation-reduction potential and pH levels of wastewater using Model Free Linear Control (MFLC-MSA) (Syafiie *et al.*, 2011), improve the cost of N-ammonia removal using tabular Q-learning (Hernández-Del-olmo *et al.*, 2016), improving energy and environmental efficiency of N-ammonia removal using policy iteration (Hernández-del-Olmo *et al.*, 2018), and optimising hydraulic retention through aerobic and anaerobic processes for biological phosphorous removal using Q-learning (Pang *et al.*, 2019). In addition, actor critic RL methods are utilised for pH adjustment for electroplating industry wastewater in a continuous action space (Alves Goulart and Dutra Pereira, 2020). This RL method was mimicked in (Yang *et al.*, 2022) where the authors utilise an actor critic RL method to track the desired dissolved oxygen set points in a wastewater treatment plant (WWTP). A more detailed review of RL application in WWTP can be found at (Croll *et al.*, 2023). Following the successes of DRL algorithms and its growing popularity, more research has deployed DRL methods to solve issues in WWTPs.

The only use of value-based DRL algorithm in wastewater treatment is present in (Nam *et al.*, 2020). The article carries out an experiment involving both RL (Q, SARSA) algorithms and DRL (DQN, deep-SARSA) to reduce the aeration energy consumption without decreasing the effluent quality index. These factors were estimated using the activated sludge model soluble product (ASM-SMP) named benchmark simulation model 1 (BSM 1) developed by (Alex *et al.*, 2018). The DQN model largely outperformed the other methods as it develops a trajectory that simultaneously improves the economic benefits by 36.53% and the environmental efficiency by 0.23%. The RL methods deployed fail to handle the complexity and caused decreases in energy savings and environmental efficiency. Further work recommended includes the experimentation with multi-agent systems to control environmental and economic benefits whilst minimising risks from membrane fouling (Nam *et al.*, 2020). The authors did not discuss hyperparameter optimisation which could further improve their current results. In addition, the use of policy gradient methods can provide insights on the difference in policy gradient and value driven DRL in performance.

In (Panjapornpon *et al.*, 2022), the author leverage the hybrid properties of multiple DDPG agents as an actor critic method. This study is more focused on developing a MADRL for pH control and tank level control by simultaneously managing the flow rates of the influent stream and neutralisation stream (Panjapornpon *et al.*, 2022) in a continuous stirred tank reactor. The authors use the grid search methods for hyperparameter tuning of three performance indexes. The DDPG uses a gated recurrent unit and rectified linear units for the actor and critic networks as shown in figures 6 & 7 (Panjapornpon *et al.*, 2022, figs 6 & 7). The multi agent DDPG algorithm performed favourably in comparison to the proportional-integral controller with controlling efficiency with better performance indexes and less oscillations (Panjapornpon *et al.*, 2022). This paper highlights the benefits of using DRL to optimise control performance. Deploying the RL controllers using programmable logical controllers on real WWTPs can provide social proof.

MADRL is utilised in (Chen *et al.*, 2021) to control dissolved oxygen set points and chemical dosage in WWTP. In this article, the authors use a multiple agent DDPG algorithm to lower environmental impacts, cost and energy consumption using a life cycle driven reward function. The life cycle assessment driven strategy has outperformed cost oriented and effluent quality optimisation in eliminating environment impacts. The use of multiple agent DDPG has provided good results however the study lacks comparisons with other optimisation algorithms which should be investigated in the future. MADRL should enable better navigation in highly complex environments therefore it would be great to validate this novel algorithm with field data.

A statistical learning based PPO algorithm is used to develop a predictive control strategy that minimises energy consumption in a wastewater pumping station in (Filipe *et al.*, 2019). The model free method decreases electrical consumption by 16.7% and tank level violations by 97% in comparison to the current operating conditions of the pumping station based in a WWTP in Fábrica da Água de Alcântara, Portugal. The authors also compare the results of using wastewater intake rate forecasts to improve the PPO algorithm's results. Indeed the forecasts help improve the results of the algorithm with cumulative energy consumption dropping from 459MWh-469MWh to 340MWh-348MWh (Filipe *et al.*, 2019). Bayesian optimisation was also utilised to optimise the forecasting hyperparameters. It is important to compare these results to other model predictive control methods used in WWTP pumping stations and other optimisation approaches to highlight the DRL algorithm's performance with respect to known benchmarks. It will be beneficial to recreate the results using WWTP benchmark models and validate the results in real-world applications.

3.2.4. DRL in Raw Water Treatment

The authors haven't found many papers to review relating to the application of DRL to the supply and treatment of raw water. A related paper discusses the use of DRL as a smart planning agent for off-grid camp water infrastructure (Makropoulos and Bouziotas, 2023) therefore it is not an urban water system. DQN, PPO and multi-armed bandits were tested using an urban water optioneering tool (UWOT). The DRL agents are tasked with using an array of different supply technologies with relevant costs and a set of demand pattern for potable and non-potable water to explore conditions of deployment in the off-grid system. This paper's ability to train and test DRL agents in strategic planning paves the way for strategic planning opportunities in UWS as well.

The only raw water supply application can be found in (Li *et al.*, 2023) where the researchers apply proximal policy optimisation (PPO) algorithm to lower suspended sediment concentration (SSC) and energy consumption tested on data from the Yellow River pumping station in China. The DRL environment is made by combining data from the hydraulic model and the SSC predictive model which is formed of a multilayer perceptron model. The PPO algorithm is trained on the predicted SSC (predictive control) and real-world SSC data (perfect predictive control). Both strategies are compared to manual strategy developed by experienced operators. The SSC predictive model was not accurate as it deviates from the training and validation sets. In both the predictive and perfect predictive control, the DRL algorithm outperforms the manual strategy resulting in a smoother sediment profile, decreases the energy consumption by 8.33%, and average sand volume per unit water withdrawal by 37.01% and 40.575% respectively (Mullapudi *et al.*, 2020). Furthermore, the authors investigate the effects of reservoir water outflows and initial reservoir water volumes. There is a strong relationship between reservoir initial water volume. This paper can benefit by comparing the DRL algorithm to other heuristic optimisation algorithms such as iterations of genetic algorithm (GA) or differential evolution (DE). The researchers should attempt to optimise the reward function by experimenting with different weights and apply some form of hyperparameter optimisation to increase the accuracy of the SSC predictive model.

Table 3-1 *Summary of reviewed articles*

System	Application	Algorithms	Case Study	Remarks	Reference
Water Distribution	Pump control	DDQN	D-town, Anytown	DDQN controls pump speeds to minimise tank outflows and keep junction heads within an acceptable range.	(Hajgató, Paál and Gyires-Tóth, 2020)
		PPO, E-PPO	EPANET Net3	E-PPO achieves the better performance in minimising tank level fluctuations and pump energy consumption.	(Hu *et al.*, 2023)
		KA-PPO	Anytown	KA-PPO controls pump speed to keep junction heads in acceptable range, minimise water age and increase pump efficiency	(Xu *et al.*, 2021)
	Water quality	PQDQN	Sao Paolo, Brazil	A novel DST and WQR expert system for DMODRL. PPQN outperforms the other algorithms.	(Hasan *et al.*, 2019)
	Asset management	GCN-DQN	Rancho Solano Zone III	Novel problem setup to test resilience post-earthquake. Use of GCN as function approximator and transfer learning greatly improves results.	(Fan, Zhang and Yu, 2022)
Stormwater systems	Flood control	DQN, DDQN, PPO1, PPO2, A2C, Voting	Sewer system in eastern China	Novel method to improve the reliability of DRL algorithms (voting). Novel emulator that outperforms benchmarks in modelling storm water systems.	(Tian, Liao, Zhi, *et al.*, 2022)
		DDPG	Norfolk, Virginia, USA	DDPG used for flood mitigation in real-time. Better results than rule-based control and faster than model predictive control by 88x.	(Bowes *et al.*, 2021)

	Valve control	DQN	Ann Arbor	DQN algorithm successfully controls SWMM but raises issues of reliability for real-world application. Serves as a starting point for further research.	(Mullapudi et al., 2020)
Wastewater systems	Dissolved oxygen settings	Deep SARSA, DQN	BSM 1	DQN algorithm outperforms all RL and DRL methods used to simultaneously increase environmental efficiency and minimise energy consumption.	(Nam *et al.*, 2020)
		Multi agent DDPG	Jiangsu Province, China	Life cycle assessment proven as a superior reward function for a multi agent DDPG in minimising environmental impact.	(Chen *et al.*, 2021)
	Pump control	PPO	Fábrica da Água de Alcântara, Portugal	WWTP pump control using wastewater intake rate forecasting to improve energy efficient and tank level violations with respect to normal operating conditions.	(Filipe *et al.*, 2019)
	pH control, tank level control	Multi agent DDPG	Servo-regulatory MATLAB test	Multi agent DDPG used to improve real time control of pH and tank levels with respect to a proportional integral controller.	(Panjapornpon *et al.*, 2022)
Raw water supply	Sediment control	PPO	Yellow river pumping station	PPO outperforms experts' manual strategy and decreases energy consumption by 8.33%. Should be compared to other optimisation algorithms	(Li *et al.*, 2023)

4. Future Work

As repeatedly displayed throughout this review, the field of deep reinforcement learning is growing rapidly and expanding across various real-world applications; the most recent of which being the water industry. This field of application is relatively new and is brimming with new possibilities for the real-time control. Extending this technology to the operational management of water systems is a field of untapped potential with many avenues to explore. DRL provides a method to continuously train the model to react and adjust to the environment it is placed in. This ability for unsupervised learning makes DRL a great tool for the instantaneous optimisation of any foreign network hence possibly globalising it water networks across the country. Researchers are therefore encouraged to experiment with simple DRL algorithms in different aspects of water distribution networks, stormwater systems, water treatment and sanitation, wastewater management such as strategic planning and asset management. The link between leakage and greenhouse gas emissions has been repeatedly mentioned in water management literature (Negm, Ma and Aggidis, 2023a) due to its relevance in the research community. It will be interesting to extend DRL algorithms in water applications to minimize carbon emissions.

As this is the first review paper dedicated to deep reinforcement learning in UWS, the collation of this evolving field should be constant to act as a beacon to new researchers. More review papers will also help define the community's direction, evaluate recent findings and reveal possible novelties. Nevertheless, it is essential that researchers interested in this field spend a considerable amount of effort understanding the fundamentals of DRL. This will help clear any misconception on the applicability of the field and highlight any new advancements. Hopefully, this will steer academics away from repeating mistakes. More research articles with the purpose of formalising methods of DRL application would serve as a great bridge for aspiring researchers. Whilst researcher focus on testing DRL on models and software case studies, it is necessary to validate the use of DRL as controllers in real-world case studies. Finally, focusing on the application of DRL in graphical based distribution systems such as the electrical distribution networks will provide a clearer perspective on possible overlaps and trends that could benefit water distribution.

To fuel further research, the research community should focus its efforts on benchmarking scalable DRL environments for testing. Early efforts to benchmark environments can save upcoming researchers the need to repeatedly contextualise the optimisation problem in the scope of DRL. These environments should be able to communicate effectively with the most popular hydraulic simulators (e.g., EPANET, SWMM and so on) through wrappers such as PYSWMM (McDonnell *et al.*, 2020) and EPYNET (Vitens, 2017). They should also be written in the necessary syntax to include benchmarked DRL libraries such as Stable Baselines, PyTorch, TensorFlow and so on. As this is an engineering application, researchers should aim to develop models that focus on reliability and scalability. Demonstrations of these algorithms acting on live data and ground-truth models in real-time should be the objective from an engineering perspective.

5. Conclusions

In this new age of digitalisation, it is necessary that our physical systems do not lack too far behind. Hence the need to constantly explore new avenues to incorporate and test the state-of-the-art algorithms. After introducing the proposed field of DRL in the water industry, the field was contextualised in the realm of artificial intelligence and machine learning. The main advantages and properties of reinforcement learning were highlighted to explain the appeal behind the technology. This was followed with a gradual explanation of the formalism and mechanisms behind reinforcement learning and deep reinforcement learning supported with mathematical proof. Different computing fields were explained thoroughly to highlight the origins of commonly used computing methods in DRL. Furthermore, the milestones, trends and challenges of deep reinforcement learning were discussed to develop a better understanding of the current research area. The main research articles that have adapted deep reinforcement learning methods to solve problems in urban water systems were review thoroughly and summarised in **Table 1**. Finally, future works and recommendations were included to provide a clear view for the application of DRL in UWSs. Therefore, the conclusion of this review can be summarised below.

- Deep reinforcement learning improves on reinforcement learning using deep neural networks for function approximation. This has improved scalability and resulted in many successes across simulated and real applications.
- Current DRL trends tackle high dimensional complexity by mimicking human psychology and natural hierarchy structures.
- The field of deep reinforcement learning can benefit from better classification to help new researchers navigate better.
- The application of DRL in the UWS is still developing yet it shows great promise to improve our current practices with water. Early efforts to benchmark DRL test beds and environments will aid the growth of this topic.

This paper aims to spark discussions and actions on future applications that harness the power of deep reinforcement learning's experience-based real-time learning in the UWS. Water is earth's most valuable resource hence the necessity to continuously improve our water practices.

6. Acknowledgments

This work is funded by Lancaster University through European Regional Development Fund, Centre of Global Eco-Innovation, and industrial partners DNS Ltd. The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

7. References

Achiam, J. (2020) 'Spinning Up Documentation Release'.

Adams, S., Cody, · Tyler and Beling, P. A. (2022) 'A survey of inverse reinforcement learning', *Artificial Intelligence Review*, 55(6), pp. 4307–4346. doi: 10.1007/s10462-021-10108-x.

Ahiablame, L. and Shakya, R. (2016) 'Modeling flood reduction effects of low impact development at a watershed scale', *Journal of Environmental Management*, 171, pp. 81–91. doi: 10.1016/J.JENVMAN.2016.01.036.

Alex, J *et al.* (2018) 'Benchmark Simulation Model no. 1 (BSM1)'.

Alves Goulart, D. and Dutra Pereira, R. (2020) 'Autonomous pH control by reinforcement learning for electroplating industry wastewater', *Computers & Chemical Engineering*, 140, p. 106909. doi: 10.1016/J.COMPCHEMENG.2020.106909.

Arulkumaran, K. *et al.* (2017) 'Deep reinforcement learning: A brief survey', *IEEE Signal Processing Magazine*, 34(6), pp. 26–38. doi: 10.1109/MSP.2017.2743240.

Aryal, S. K. *et al.* (2016) 'Assessing and Mitigating the Hydrological Impacts of Urbanisation in Semi-Urban Catchments Using the Storm Water Management Model', *Water Resources Management*, 30(14), pp. 5437–5454. doi: 10.1007/S11269-016-1499-Z.

Baird, L. (1995) 'Residual Algorithms: Reinforcement Learning with Function Approximation', *Machine Learning Proceedings 1995*, pp. 30–37. doi: 10.1016/B978-1-55860-377-6.50013-X.

Barton, N. A. *et al.* (2019) 'Improving pipe failure predictions: Factors effecting pipe failure in drinking water networks', *Water Research*, 164. doi: 10.1016/J.WATRES.2019.114926.

Beattie, C. *et al.* (2016) 'DeepMind Lab'. Available at: https://arxiv.org/abs/1612.03801v2 (Accessed: 4 May 2023).

Bellemare, M. G., Dabney, W. and Munos, R. (2017) 'A Distributional Perspective on Reinforcement Learning', *34th International Conference on Machine Learning, ICML 2017*, 1, pp. 693–711. Available at: https://arxiv.org/abs/1707.06887v1 (Accessed: 10 May 2023).

Bellman, R. (1952) 'On the Theory of Dynamic Programming', *Proceedings of the National Academy of Sciences*, 38(8), pp. 716–719. doi: 10.1073/PNAS.38.8.716/ASSET/BADDE5C3-CE28-4677-B095-95015576EEBC/ASSETS/PNAS.38.8.716.FP.PNG.

Benjamin, M. M. (2014) *Water chemistry*.

Berner, C. *et al.* (2019) 'Dota 2 with Large Scale Deep Reinforcement Learning'. Available at: https://www.facebook.com/OGDota2/ (Accessed: 14 February 2023).

Bertsekas, D. P., Tsitsiklis, J. N. and Τσιτσικλής, Γ. Ν. (1996) 'Neuro-dynamic programming', p. 491.

Bloembergen, D. *et al.* (2015) 'Evolutionary dynamics of multi-agent learning: A survey', *Journal of Artificial Intelligence Research*. doi: 10.1613/jair.4818.

Bowes, B. D. *et al.* (2021) 'Flood mitigation in coastal urban catchments using real-time stormwater infrastructure control and reinforcement learning', *Journal of Hydroinformatics*, 23(3), pp. 529–547. doi: 10.2166/HYDRO.2020.080.

Buşoniu, L., Babuška, R. and De Schutter, B. (2008) 'A comprehensive survey of multiagent reinforcement learning', *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 38(2), pp. 156–172. doi: 10.1109/TSMCC.2007.913919.

Chen, K. *et al.* (2021) 'Optimal control towards sustainable wastewater treatment plants based on multi-agent reinforcement learning', *Chemosphere*, 279, p. 130498. doi: 10.1016/J.CHEMOSPHERE.2021.130498.

Crini, G. and Lichtfouse, E. (2019) 'Advantages and disadvantages of techniques used for wastewater treatment', *Environmental Chemistry Letters*, 17, pp. 145–155. doi: 10.1007/s10311-018-0785-9.

Croll, H. C. *et al.* (2023) 'Reinforcement learning applied to wastewater treatment process control optimization: Approaches, challenges, and path forward', *Critical Reviews in Environmental Science and Technology*, 53(20), pp. 1775–1794. doi: 10.1080/10643389.2023.2183699.

Dabney, W. *et al.* (2017) 'Distributional Reinforcement Learning with Quantile Regression', *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pp. 2892–2901. doi: 10.1609/aaai.v32i1.11791.

Dabney, W. *et al.* (2018) 'Implicit Quantile Networks for Distributional Reinforcement Learning', *35th International Conference on Machine Learning, ICML 2018*, 3, pp. 1774–1787. Available at: https://arxiv.org/abs/1806.06923v1 (Accessed: 10 May 2023).

Dabney, W. *et al.* (2020) 'A distributional code for value in dopamine-based reinforcement learning', *Nature 2020 577:7792*, 577(7792), pp. 671–675. doi: 10.1038/s41586-019-1924-6.

Desharnais, J. *et al.* (2004) 'Metrics for labelled Markov processes', *Theoretical Computer Science*, 318(3), pp. 323–354. doi: 10.1016/J.TCS.2003.09.013.

Duan, Y. *et al.* (2016) 'Benchmarking Deep Reinforcement Learning for Continuous Control'. PMLR, pp. 1329–1338. Available at: https://proceedings.mlr.press/v48/duan16.html (Accessed: 4 May 2023).

Endo, A. *et al.* (2017) 'A review of the current state of research on the water, energy, and food nexus', *Journal of Hydrology: Regional Studies*, 11, pp. 20–30. doi: 10.1016/J.EJRH.2015.11.010.

Etikala, B., Madhav, S. and Somagouni, S. G. (2022) 'Urban water systems: An overview', 6, pp. 1–19. doi: 10.1016/B978-0-323-91838-1.00016-6.

Fan, X., Zhang, X. and Yu, X. (2022) 'A graph convolution network-deep reinforcement learning model for resilient water distribution network repair decisions', *Computer-Aided Civil and Infrastructure Engineering*, 37(12), pp. 1547–1565. doi: 10.1111/MICE.12813.

Filipe, J. *et al.* (2019) 'Data-driven predictive energy optimization in a wastewater pumping station'. doi: 10.1016/j.apenergy.2019.113423.

Finn, C., Levine, S. and Abbeel, P. (2016) 'Guided Cost Learning: Deep Inverse Optimal Control via Policy Optimization', *33rd International Conference on Machine Learning, ICML 2016*, 1, pp. 95–107. Available at: https://arxiv.org/abs/1603.00448v3 (Accessed: 10 May 2023).

Fu, G. *et al.* (2022) 'The role of deep learning in urban water management: A critical review', *Water Research*, 223. doi: 10.1016/j.watres.2022.118973.

Gomez, F. and Schmidhuber, J. (2005) 'Evolving modular fast-weight networks for control', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3697 LNCS, pp. 383–389. doi: 10.1007/11550907_61/COVER.

Gordon, G. J. (1995) 'Stable Function Approximation in Dynamic Programming', *Machine Learning Proceedings 1995*, pp. 261–268. doi: 10.1016/B978-1-55860-377-6.50040-2.

Gu, S. *et al.* (2016) 'Continuous Deep Q-Learning with Model-based Acceleration', *33rd International Conference on Machine Learning, ICML 2016*, 6, pp. 4135–4148. Available at: https://arxiv.org/abs/1603.00748v1 (Accessed: 1 April 2023).

Hajgató, G., Paál, G. and Gyires-Tóth, B. (2020) 'Deep Reinforcement Learning for Real-Time Optimization of Pumps in Water Distribution Systems', *Journal of Water Resources Planning and Management*, 146(11). doi: 10.1061/(asce)wr.1943-5452.0001287.

Hasan, M. M. *et al.* (2019) 'Dynamic multi-objective optimisation using deep reinforcement learning: benchmark, algorithm and an application to identify vulnerable zones based on water quality', *Engineering Applications of Artificial Intelligence*, 86, pp. 107–135. doi: 10.1016/J.ENGAPPAI.2019.08.014.

Van Hasselt, H., Guez, A. and Silver, D. (2016) 'Deep Reinforcement Learning with Double Q-learning', in *30th AAAI Conference on Artificial Intelligence*, pp. 2094–2100. Available at: www.aaai.org (Accessed: 4 May 2023).

Heess, N. *et al.* (2015) 'Learning Continuous Control Policies by Stochastic Value Gradients', *Advances in Neural Information Processing Systems*, 2015-January, pp. 2944–2952. Available at: https://arxiv.org/abs/1510.09142v1 (Accessed: 9 May 2023).

Hernández-del-Olmo, F. *et al.* (2018) 'Tackling the start-up of a reinforcement learning agent for the control of wastewater treatment plants', *Knowledge-Based Systems*, 144, pp. 9–15. doi: 10.1016/J.KNOSYS.2017.12.019.

Hernández-Del-olmo, F. *et al.* (2016) 'Energy and Environmental Efficiency for the N-Ammonia Removal Process in Wastewater Treatment Plants by Means of Reinforcement Learning', *Energies 2016, Vol. 9, Page 755*, 9(9), p. 755. doi: 10.3390/EN9090755.

Hernandez-Leal, P., Kartal, B. and Taylor, M. E. (2018) 'A Survey and Critique of Multiagent Deep Reinforcement Learning'. doi: 10.1007/s10458-019-09421-1.

Hernandez-Leal, P., Kartal, B. and Taylor, M. E. (2019) 'Is multiagent deep reinforcement learning the answer or the question? A brief survey', *Autonomous Agents and Multi-Agent Systems*, 33(6).

Ho, J. and Ermon, S. (2016) 'Generative Adversarial Imitation Learning', *Advances in Neural Information Processing Systems*, pp. 4572–4580. Available at: https://arxiv.org/abs/1606.03476v1 (Accessed: 10 May 2023).

Hu, S. *et al.* (2023) 'Real-Time Scheduling of Pumps in Water Distribution Systems Based on Exploration-Enhanced Deep Reinforcement Learning', *Systems 2023, Vol. 11, Page 56*, 11(2), p. 56. doi: 10.3390/SYSTEMS11020056.

Hussain, A. *et al.* (2021) 'Biological wastewater treatment technology: Advancement and drawbacks', *Microbial Ecology of Wastewater Treatment Plants*, pp. 175–192. doi: 10.1016/B978-0-12-822503-5.00002-3.

Hutsebaut-Buysse, M., Mets, K. and Latré, S. (2022) 'Hierarchical Reinforcement Learning: A Survey and Open Research Challenges', *Machine Learning and Knowledge Extraction 2022, Vol. 4, Pages 172-221*, 4(1), pp. 172–221. doi: 10.3390/MAKE4010009.

Jefferson, A. J. *et al.* (2017) 'Stormwater management network effectiveness and implications for urban watershed function: A critical review', *Hydrological Processes*, 31(23), pp. 4056–4080. doi: 10.1002/HYP.11347.

Jiang, J.-Q. (2015) 'The role of coagulation in water treatment This review comes from a themed issue on Separation engineering', *Current Opinion in Chemical Engineering*, 8, pp. 36–44. doi: 10.1016/j.coche.2015.01.008.

Jotte, L., Raspati, G. and Azrague, K. (2017) *REVIEW OF STORMWATER MANAGEMENT PRACTICES*. Available at: www.klima2050.no (Accessed: 26 September 2023).

Kalashnikov, D. *et al.* (2018) 'QT-Opt: Scalable Deep Reinforcement Learning for Vision-Based Robotic Manipulation'. Available at: https://arxiv.org/abs/1806.10293v3 (Accessed: 26 March 2023).

Kentish, S. and Stevens, G. (2001) 'Innovations in separations technology for the recycling and re-use of liquid waste streams', *Chemical Engineering Journal*, 84(2). Available at: https://www.sciencedirect.com/science/article/pii/S1385894701001991 (Accessed: 25 September 2023).

Kingma, D. P. and Welling, M. (2013) 'Auto-Encoding Variational Bayes', *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. Available at: https://arxiv.org/abs/1312.6114v11 (Accessed: 9 May 2023).

Kılkış, Ş. *et al.* (2023) 'Sustainable development of energy, water and environment systems in the critical decade for climate action', *Energy Conversion and Management*, 296, p. 117644. doi: 10.1016/J.ENCONMAN.2023.117644.

Kohl, N. and Stone, P. (2004) 'Policy Gradient Reinforcement Learning for Fast Quadrupedal Locomotion', pp. 2619–2624. Available at: http://www.cs.utexas.edu/~%7Bnate,pstone%7D (Accessed: 6 February 2023).

Konda, V. R. and Tsitsiklis, J. N. (1999) 'ON ACTOR-CRITIC ALGORITHMS', *Advances in Neural Information Processing Systems*, 42(4), pp. 1143–1166. doi: 10.1137/S0363012901385691.

Kool, W., Van Hoof, H. and Welling, M. (2018) 'Attention, Learn to Solve Routing Problems!', *7th International Conference on Learning Representations, ICLR 2019*. doi: 10.48550/arxiv.1803.08475.

Koutník, J. *et al.* (2013) 'Evolving Large-Scale Neural Networks for Vision-Based Reinforcement Learning', in *Genetic and Evolutionary Computation Conference*. Available at: http://www.idsia.ch/~koutnik/images/octo (Accessed: 26 March 2023).

Lai, T. L. and Robbins, H. (1985) 'Asymptotically efficient adaptive allocation rules', *Advances in Applied Mathematics*, 6(1). doi: 10.1016/0196-8858(85)90002-8.

Lapan, M. (2019) ‘Deep Reinforcement Learning Learning Hands-on’, *Reinforcement Learning for Cyber-Physical Systems*, pp. 125–154. Available at: https://www.packtpub.com/product/deep-reinforcement-learning-hands-on-second-edition/9781838826994 (Accessed: 24 February 2023).

Levine, S. *et al.* (2016) ‘End-to-end training of deep visuomotor policies’, *Journal of Machine Learning Research*.

Levine, S. and Van De Panne, M. (2018) ‘DeepMimic: Example-Guided Deep Reinforcement Learning of Physics-Based Character Skills’, *ACM Trans. Graph*, 37(143), p. 18. doi: 10.1145/3197517.3201311.

Li, Y. (2017) ‘Deep Reinforcement Learning: An Overview’. doi: 10.48550/arxiv.1701.07274.

Li, Z. *et al.* (2023) ‘Online Control of the Raw Water System of a High-Sediment River Based on Deep Reinforcement Learning’, *Water 2023, Vol. 15, Page 1131*, 15(6), p. 1131. doi: 10.3390/W15061131.

Libbrecht, M. W. and Noble, W. S. (2015) ‘Machine learning applications in genetics and genomics’, *Nature Reviews Genetics 2015 16:6*, 16(6), pp. 321–332. doi: 10.1038/nrg3920.

Lillicrap, T. P. *et al.* (2016) ‘Continuous control with deep reinforcement learning’, in *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*.

Lipps, W. C., Braun-Howland, E. B. and Baxter, T. E. (2022) ‘Standard methods for the examination of water and wastewater’, p. 1536.

Loubet, P. *et al.* (2014) ‘Life cycle assessments of urban water systems: A comparative analysis of selected peer-reviewed literature’. doi: 10.1016/j.watres.2014.08.048.

Lowet, A. S. *et al.* (2020) ‘Distributional Reinforcement Learning in the Brain’, *Trends in Neurosciences*, 43(12), pp. 980–997. doi: 10.1016/j.tins.2020.09.004.

Mace, M. (2020) *Water Industry Launches First Sector Wide Innovation Strategy*, *Water.org*. Available at: https://www.water.org.uk/news-item/water-industry-launches-first-sector-wide-innovation-strategy (Accessed: 29 December 2020).

Maier, H. R. *et al.* (2014) ‘Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions’, *Environmental Modelling & Software*, 62, pp. 271–299. doi: 10.1016/J.ENVSOFT.2014.09.013.

Makropoulos, C. and Bouziotas, D. (2023) ‘Artificial intelligence for decentralized water systems: A smart planning agent based on reinforcement learning for off-grid camp water infrastructures’, *Journal of Hydroinformatics*, 25(3), pp. 912–926. doi: 10.2166/HYDRO.2023.168.

McDonnell, B. *et al.* (2020) ‘PySWMM: The Python Interface to Stormwater Management Model (SWMM)’, *Journal of Open Source Software*, 5(52), p. 2292. doi: 10.21105/JOSS.02292.

Mnih, V. *et al.* (2015a) ‘Human-level control through deep reinforcement learning’, *Nature*, 518(7540). doi: 10.1038/nature14236.

Mnih, V. *et al.* (2015b) ‘Human-level control through deep reinforcement learning’, *Nature*, 518(7540), pp. 529–533. doi: 10.1038/NATURE14236.

Mnih, V. *et al.* (2016) ‘Asynchronous Methods for Deep Reinforcement Learning’, *33rd International Conference on Machine Learning, ICML 2016*, 4, pp. 2850–2869. Available at: https://arxiv.org/abs/1602.01783v2 (Accessed: 1 April 2023).

Mosetlhe, T. C. *et al.* (2020) ‘A survey of pressure control approaches in water supply systems’, *Water (Switzerland)*. doi: 10.3390/W12061732.

Mullapudi, A. *et al.* (2020) ‘Deep reinforcement learning for the real time control of stormwater systems’, *Advances in Water Resources*, 140, p. 103600. doi: 10.1016/J.ADVWATRES.2020.103600.

Nair, S. *et al.* (2014) ‘Water–energy–greenhouse gas nexus of urban water systems: Review of concepts, state-of-art and methods’, *Resources, Conservation and Recycling*, 89, pp. 1–10. doi: 10.1016/J.RESCONREC.2014.05.007.

Nam, K. J. *et al.* (2020) ‘An autonomous operational trajectory searching system for an economic and environmental membrane bioreactor plant using deep reinforcement learning’, *Water Science and Technology*, 81(8), pp. 1578–1587. doi: 10.2166/WST.2020.053.

Nazari, M. *et al.* (2018) ‘Reinforcement Learning for Solving the Vehicle Routing Problem’, *Advances in Neural Information Processing Systems*, 2018-December, pp. 9839–9849. doi: 10.48550/arxiv.1802.04240.

Negm, A., Ma, X. and Aggidis, G. (2023a) 'Review of leakage detection in water distribution networks', *IOP Conference Series: Earth and Environmental Science*, 1136(1), p. 012052. doi: 10.1088/1755-1315/1136/1/012052.

Negm, A., Ma, X. and Aggidis, G. (2023b) 'Water Pressure Optimisation for Leakage Management Using Q Learning', *2023 IEEE Conference on Artificial Intelligence (CAI)*, pp. 270–271. doi: 10.1109/CAI54212.2023.00120.

Ng, A. Y. and Russell, S. (2000) 'Algorithms for Inverse Reinforcement Learning', in *International Conference of Machine learning*, pp. 663–670. Available at: http://www.eecs.harvard.edu/cs286r/courses/spring06/papers/ngruss_irl00.pdf (Accessed: 10 May 2023).

Nguyen, H. and La, H. (2019) 'Review of Deep Reinforcement Learning for Robot Manipulation', *Proceedings - 3rd IEEE International Conference on Robotic Computing, IRC 2019*, pp. 590–595. doi: 10.1109/IRC.2019.00120.

Nguyen, T. T., Nguyen, N. D. and Nahavandi, S. (2020) 'Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications', *IEEE Transactions on Cybernetics*, 50(9), pp. 3826–3839. doi: 10.1109/TCYB.2020.2977374.

Nichols, J. A., Herbert Chan, H. W. and Baker, M. A. B. (2019) 'Machine learning: applications of artificial intelligence to imaging and diagnosis', *Biophysical Reviews*, 11(1), pp. 111–118. doi: 10.1007/S12551-018-0449-9/METRICS.

Olsson, G. (2012) 'Water and Energy Nexus', *Life cycle assessment and water management-related issues. - (Quaderns de medi ambient ; 4)*, pp. 137–164. doi: 10.1400/241100.

Osband, I. *et al.* (2016) 'Deep exploration via bootstrapped DQN', in *Advances in Neural Information Processing Systems*.

Paine, T. Le *et al.* (2018) 'One-Shot High-Fidelity Imitation: Training Large-Scale Deep Nets with RL'. Available at: https://arxiv.org/abs/1810.05017v1 (Accessed: 10 May 2023).

Pang, J. W. *et al.* (2019) 'An influent responsive control strategy with machine learning: Q-learning based optimization method for a biological phosphorus removal system', *Chemosphere*, 234, pp. 893–901. doi: 10.1016/J.CHEMOSPHERE.2019.06.103.

Panjapornpon, C. *et al.* (2022) 'Reinforcement Learning Control with Deep Deterministic Policy Gradient Algorithm for Multivariable pH Process', *Processes 2022, Vol. 10, Page 2514*, 10(12), p. 2514. doi: 10.3390/PR10122514.

Pathak, D. *et al.* (2017) 'Curiosity-Driven Exploration by Self-Supervised Prediction', in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. doi: 10.1109/CVPRW.2017.70.

Peng, X. Bin *et al.* (2018) 'Variational Discriminator Bottleneck: Improving Imitation Learning, Inverse RL, and GANs by Constraining Information Flow', *7th International Conference on Learning Representations, ICLR 2019*. Available at: https://arxiv.org/abs/1810.00821v4 (Accessed: 10 May 2023).

Pomerleau, D. A. (1989) 'ALVINN: AN AUTONOMOUS LAND VEHICLE IN A NEURAL NETWORK', *Advances in Neural Information Processing Systems 1*, 1, pp. 305–315.

Prudencio, R. F., Maximo, M. R. O. A. and Colombini, E. L. (2022) 'A Survey on Offline Reinforcement Learning: Taxonomy, Review, and Open Problems'. doi: 10.1109/TNNLS.2023.3250269.

Puterman, M. L. (1990) 'Chapter 8 Markov decision processes', *Handbooks in Operations Research and Management Science*, 2(C), pp. 331–434. doi: 10.1016/S0927-0507(05)80172-0.

Rezende, D. J., Mohamed, S. and Wierstra, D. (2014) 'Stochastic Backpropagation and Approximate Inference in Deep Generative Models', in. PMLR, pp. 1278–1286. Available at: https://proceedings.mlr.press/v32/rezende14.html (Accessed: 9 May 2023).

Sadler, J. M. *et al.* (2020) 'Exploring real-time control of stormwater systems for mitigating flood risk due to sea level rise', *Journal of Hydrology*, 583, p. 124571. doi: 10.1016/J.JHYDROL.2020.124571.

Salimans, T. *et al.* (2017) 'Evolution Strategies as a Scalable Alternative to Reinforcement Learning'.

Schulman, J. *et al.* (2015) 'High-Dimensional Continuous Control Using Generalized Advantage Estimation', *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*. Available at: https://arxiv.org/abs/1506.02438v6 (Accessed: 27 March 2023).

Sharma, A. *et al.* (2010) 'Role of decentralised systems in the transition of urban water systems', *Water Supply*, 10(4), pp. 577–583. doi: 10.2166/WS.2010.187.

Shinde, P. P. and Shah, S. (2018) 'A Review of Machine Learning and Deep Learning Applications', *Proceedings - 2018 4th International Conference on Computing, Communication Control and Automation, ICCUBEA 2018*. doi: 10.1109/ICCUBEA.2018.8697857.

Da Silva, F. L., Taylor, M. E. and Costa, A. H. R. (2018) 'Autonomously reusing knowledge in multiagent reinforcement learning', in *IJCAI International Joint Conference on Artificial Intelligence*. doi: 10.24963/ijcai.2018/774.

Silver, D. *et al.* (2014) 'Deterministic Policy Gradient Algorithms'.

Silver, D. *et al.* (2016) 'Mastering the game of Go with deep neural networks and tree search', *Nature*, 529(7587). doi: 10.1038/nature16961.

Singh, S. *et al.* (2002) 'Optimizing dialogue management with reinforcement learning: Experiments with the NJFun system," ', *Journal of Artificial Intelligence*, 16, pp. 105–133.

Strehl, A. L. *et al.* (2006) 'PAC Model-Free Reinforcement Learning', in *23rd International Conference on Machine learning*, pp. 881–888.

Sutton, R. S. *et al.* (2000) 'Policy gradient methods for reinforcement learning with function approximation', in *Advances in Neural Information Processing Systems*.

Sutton, R. S. and Barto, A. G. (2018) *Reinforcement Learning: An Introduction*. Second. MIT Press.

Syafiie, S. *et al.* (2011) 'Model-free control based on reinforcement learning for a wastewater treatment problem', *Applied Soft Computing*, 11(1), pp. 73–82. doi: 10.1016/J.ASOC.2009.10.018.

Teodosiu, C. *et al.* (2018) 'Emerging pollutants removal through advanced drinking water treatment: A review on processes and environmental performances assessment'. doi: 10.1016/j.jclepro.2018.06.247.

Tesau, C. and Tesau, G. (1995) 'Temporal difference learning and TD-Gammon', *Communications of the ACM*, 38(3), pp. 58–68. doi: 10.1145/203330.203343.

Tessler, C. *et al.* (2017) 'A Deep Hierarchical Approach to Lifelong Learning in Minecraft', *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), pp. 1553–1561. doi: 10.1609/AAAI.V31I1.10744.

Tian, W., Liao, Z., Zhi, G., *et al.* (2022) 'Combined Sewer Overflow and Flooding Mitigation Through a Reliable Real-Time Control Based on Multi-Reinforcement Learning and Model Predictive Control', *Water Resources Research*, 58(7), p. e2021WR030703. doi: 10.1029/2021WR030703.

Tian, W., Liao, Z., Zhang, Z., *et al.* (2022) 'Flooding and Overflow Mitigation Using Deep Reinforcement Learning Based on Koopman Operator of Urban Drainage Systems', *Water Resources Research*, 58(7), p. e2021WR030939. doi: 10.1029/2021WR030939.

Tsitsiklis, J. N. and Van Roy, B. (1997) 'An analysis of temporal-difference learning with function approximation', *IEEE Transactions on Automatic Control*, 42(5). doi: 10.1109/9.580874.

U.K.W.I.R. (2020) *UK Water Innovation Strategy*. Available at: http://brilliantnoise.com/wp-content/uploads/2020/09/UK-2050-Water-Innovation-Strategy.pdf.

UN-Water (2012) *UN World Water Development Report*. Available at: https://www.unwater.org/publications/un-world-water-development-report-2012 (Accessed: 26 September 2023).

Usunier, N. *et al.* (2017) 'Episodic exploration for deep deterministic policies for starcraft micromanagement', in *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*.

Vezhnevets, A. S. *et al.* (2017) 'FeUdal Networks for Hierarchical Reinforcement Learning', *34th International Conference on Machine Learning, ICML 2017*, 7, pp. 5409–5418. Available at: https://arxiv.org/abs/1703.01161v2 (Accessed: 9 May 2023).

Vitens (2017) *Epynet: Object-oriented wrapper for EPANET 2.1*. Available at: https://github.com/Vitens/epynet (Accessed: 6 May 2022).

Wang, Z. *et al.* (2015) 'Dueling Network Architectures for Deep Reinforcement Learning', *33rd International Conference on Machine Learning, ICML 2016*, 4, pp. 2939–2947. Available at: https://arxiv.org/abs/1511.06581v3 (Accessed: 1 April 2023).

Williams, R. J. (1988) 'On the use of backpropagation in associative reinforcement learning', pp. 263–270. doi: 10.1109/ICNN.1988.23856.

Williams, R. J. (1992) 'Simple statistical gradient-following algorithms for connectionist reinforcement learning', *Machine Learning 1992 8:3*, 8(3), pp. 229–256. doi: 10.1007/BF00992696.

Xu, J. *et al.* (2021) 'Zone scheduling optimization of pumps in water distribution networks with deep reinforcement learning and knowledge-assisted learning', *Soft Computing*, 25(23), pp. 14757–14767. doi: 10.1007/S00500-021-06177-3/FIGURES/7.

Xu, Q. *et al.* (2014) 'Review on water leakage control in distribution networks and the associated environmental benefits', *Journal of Environmental Sciences (China)*, 26(5), pp. 955–961. doi: 10.1016/S1001-0742(13)60569-0.

Yang, D. *et al.* (2019) 'Fully Parameterized Quantile Function for Distributional Reinforcement Learning', *Advances in Neural Information Processing Systems*, 32.

Yang, Q. *et al.* (2022) 'Reinforcement-Learning-Based Tracking Control of Waste Water Treatment Process under Realistic System Conditions and Control Performance Requirements', *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(8), pp. 5284–5294. doi: 10.1109/TSMC.2021.3122802.

Zhao, W., Queralta, J. P. and Westerlund, T. (2020) 'Sim-to-Real Transfer in Deep Reinforcement Learning for Robotics: A Survey', *2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020*, pp. 737–744. doi: 10.1109/SSCI47803.2020.9308468.

Zhu, Y. *et al.* (2016) 'Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning', *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3357–3364. doi: 10.1109/ICRA.2017.7989381.

Ziebart, B. D. and Fox, D. (2010) 'Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy'.