Deep Multi-Agent Reinforcement Learning for Dynamic and Sequential Watershed Management

Shresth Verma
ABV-Indian Institute of Information Technology and Management
Gwalior, India
vermashresth@gmail.com

Joydip Dhar
ABV-Indian Institute of Information Technology and Management
Gwalior, India
jdhar@iiitm.ac.in

ABSTRACT

Watershed management is a common pool resource appropriation problem that shows unique complexities due to the underlying downstream flow variable. It also involves multiple self-interested and often conflicting entities, making it a challenging decision-making problem. While water scarcity is increasingly becoming a global issue, it is of utmost importance to improve the water resource allocation in terms of efficiency and equality.

Multiple previous works have used Multi-Agent System (MAS) based methods for solving spatial water resource allocation as a static optimisation problem. However, in real-world, watershed management is both a spatial and temporal problem with uncertain system dynamics involving long term sequential decision making. Thus, we propose a deep multi-agent reinforcement learning (MARL) framework for watershed management. Further, we demonstrate how inter-agent communication is essential in reaching coordination in practical scenarios. We also use model-of-agent approach to incorporate influencing among agents, thus capturing complex societal dynamics that revolve around the watershed problem. To the best of our knowledge, this is the first work studying watershed management using Deep Multi-Agent Reinforcement Learning and emergent communicative behaviours.

KEYWORDS

Reinforcement Learning, Multi-agent systems, Watershed Management, Common-pool resources

ACM Reference Format:
Shresth Verma and Joydip Dhar. 2023. Deep Multi-Agent Reinforcement Learning for Dynamic and Sequential Watershed Management. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 9 pages.

1 INTRODUCTION

With advances in economies and living standard across the globe, there has been substantial growth of dependence on water resources. As a result, many communities are facing an acute shortage of water resources. This shortage has been attributed to numerous factors such as dynamic demographic patterns, changing energy mix, urbanisation, migration and industrialisation [14, 20, 28]. Human activities and climate change are also worsening these problems all across the globe and more prominently in developing countries [1]. This increasing imbalance between demand and supply of water resources makes it crucial for decision-makers to solve this conflict by improving the fairness and effectiveness of watershed management. While using socially optimal solutions and imposing them on the population is one way out, it can often cause unwanted outcomes pertaining to fairness in the process of decision making [4]. Such solutions may also prompt agents to behave selfishly because of displaced moral sentiments. It has been observed that promoting collective actions through community organisations can be expensive and creates a dependence on external schemes. Often, such cooperation doesn't sustain after the program ends [3].

Thus, it is crucial to allow the agents participating in watershed management to realise their interdependencies and internalise the value of cooperation. In this regard, extensive work has been dedicated to solving the problem of watershed management using agent-based modelling [10, 12, 14, 18, 28]. More importantly, these modelling approaches offer tools and techniques to decision-makers for observing water resources, their quality and potential hazard issues that can emerge from specific policies [29].

Traditionally, water resource management has been approached in a centralised manner, assuming full information exchange between participating agents and perfect economic efficiency. However, such top-down approaches are not practical for real-world contexts because the centralised management often fails to represent politically and socially feasible solutions. Decentralised management, on the other hand, can easily incorporate multiple and independent decision-makers from its bottom-up approach [31].

To implement such a system, we simulate watershed management as a multi-agent system (MAS) having spatially distributed water users (agents) in a predefined environment. These autonomous agents can act independently in the environment where they can either be working towards achieving a system-wide goal or have individual objectives. Watershed management in the context of MAS can be posed as a resource management problem. It consists of several self-interested parties that seek to gain benefit from water in the system. This water can be used for various purposes such as sustaining a city, generating hydroelectric power, irrigating farm or growing a natural ecosystem.

A multitude of literature has been dedicated to optimising watershed management in MAS setting. However, works using Multi-Agent RL are limited [8, 12, 18, 19, 31]. Moreover, these works assume watershed management to be a static problem with a single step decision making. To lift these restrictions and model complex decision making, we tackle watershed management using Deep Multi-Agent Reinforcement Learning. Reinforcement learning (RL) is a sub-area of machine learning that allows learning through trial

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

and error while interacting with the environment. Recently, RL has shown tremendous success in various domains such as robotic control, game playing, networking and routing, data centre cooling and many more [13, 16, 17, 26]. This success has partially been due to the combination of reinforcement learning paradigm with deep learning, to handle complex decision making and high dimensional data. Along the same lines, deep multi-agent reinforcement learning, which models multiple agents in an environment has also proven to be successful in solving several cooperative and competitive Multi-Agent tasks. Some of the applications include air traffic control, power grid management, traffic signal control, data routing in networks [2, 5, 27, 30].

Despite these successes, it is still a difficult problem to achieve coordination in MARL. In this regard, several previous works have proposed using centralised training to ensure coordination among learning agents. [6, 7]. A large number of works are also dedicated to the emergence of communication in multi-agent RL, where the agents learn to communicate from scratch, determining what information is essential to share in order to accomplish a task. However, using centralised training for self-interested agents can be impractical for real-world use cases.

Recently, there has been a growing interest in modelling problems of common-pool resource appropriation through deep multi-agent reinforcement learning [9, 11, 15, 23]. But the common pool resources studied in these are spatially static. Watershed is also a common pool resource which has flow characteristic and involves complex interdependencies bounded by physical constraints as well as agents' behaviours. Such a resource can suffer from congestion without careful coordination. This is because, without an effective communication strategy, the use of the resource by one agent makes it difficult or impossible for others to access it. Furthermore, in [22], the authors have shown that most common pool resource experiments follow a structure of non-linear social dilemma with a non-excludable resource demanded by multiple players. Watershed management, as a common pool resource problem, shows spatial effects and vertical downstream externalities. In our work, we add on the dimension of temporal effect such that decision made by multiple parties have long term consequences. Doing this makes watershed management similar to the problem of sequential social dilemmas. Moreover, inspired by the work in [11], we study the effect of social influence on achieving coordination among agents.

The contributions of this paper are thus as follows:

(1) To propose watershed management as a Multi-Agent System with dynamic water flows, changing water requirements of agents and sequential decision making across multiple time steps.
(2) To apply recent advances in communication and social influence in Deep Multi-Agent Reinforcement Learning setting for solving the Watershed Management problem.

2 BACKGROUND

We consider a decentralized multi-agent reinforcement learning scenario and formulate it as an N-player partially observable Markov game M. The formulation includes a set S containing all possible states of the environments, action sets for each of the agents $\{\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_N\}$ where A_i is the set of actions for i^{th} agent, observation sets for each of the agents $\{O_1, O_2, ..., O_N\}$ where O_i is the set of observations available to the i^{th} agent. $o_i \in O_i^t$ is the observable state of i^{th} agent at t^{th} timestep. The joint actions are given by $a_1^t, a_2^t, ..., a_n^t \in \mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_N$ where a_i^t is the action of i^{th} agent at t^{th} timestep . This causes a change of state according to the stochastic transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \times ... \times \mathcal{A}_n \rightarrow S'$. Each player receives a reward according to the function $r_i : \mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \times ... \times \mathcal{A}_n \rightarrow \mathbb{R}$. The reward thus received by an agent may depend on other agents' actions. A trajectory is defined as history of these variables.

Each agents learns an independent behaviour policy parametrized by θ, which maps from observations to action distribution $\pi_i : O_i \rightarrow \mathcal{A}_i$. An action $a_i^t \in A_i$ is then deterministically, or stochastically sampled from this distribution. The agent's goal is to maximize long term expected $\gamma-$discounted reward, over a period of T timesteps. This is given by $J(\theta_i) = \mathbb{E}[R_i]$ where $R_i = \sum_{t=0}^{T} \gamma^t r_i(o_i^t, a_i^t)$

3 WATERSHED MANAGEMENT PROBLEM

Watershed Management is a problem of resource allocation consisting of several self-interested agents. These agents can withdraw water from a finite but common supply of water for individual purposes. The problem involves several constraints, multiple objectives, and optimisation involves continuous variables. For modelling watershed management as a multi-agent system, we use the hypothetical scenario first proposed in [31] and later used in [8, 19]. Fig. 1, shows a schematic diagram of the hypothetical watershed basin. The scenario consists of a watershed basin with one mainstream and one tributary. Three off-stream human agents are considered, namely, one city (OHA_1) and two farms (OHA_2, OHA_3). One Dam (IHA_1) in the system is considered as an in-stream human agent. Further, two ecosystem agents (EA_1, EA_2) are considered, one on the tributary and the other on the mainstream. The off-stream human agents can control the amount of water to withdraw from the stream they are located at. A dam can control the release of water from storage and in-flow stream. Ecosystems thrive upon the water from the stream they are based on. To simulate non-linear characteristics of realistic benefits from water, quadratic objective functions are assigned to each agent.

Two kinds of constraints are imposed on control actions of each agent in this scenario. First is a soft constraint, which relates to rules pertaining to policies such as minimum water requirement. Second is a hard constraint, which represents the physical constraints on the flow of water. In the real world, soft constraints can be violated at some cost, however, hard constraints are impossible to violate. In [8, 19, 31], constraints violations have been handled by incorporating penalties into the objective function.

In our work, we modify the watershed management problem so as to make it more practical for real-world implementation. In particular, we address the following limitations in the hypothetical scenario in question:

(1) Hard constraints are allowed to be violated, which is impossible in the real world.
(2) The water flow rates are considered static and limited to very few values. However, in the real world, they can constantly change over time.

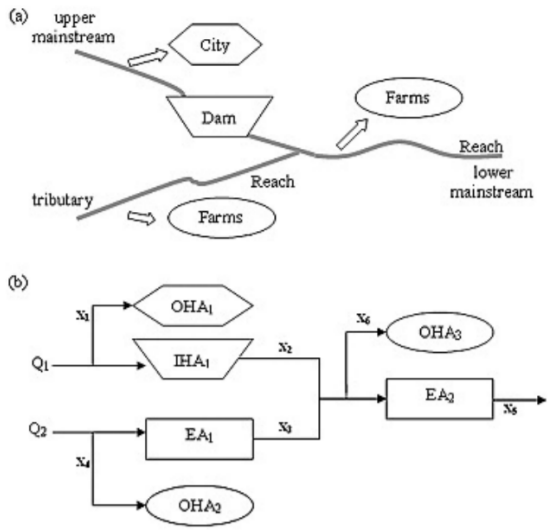

Figure 1: Watershed management, a schematic diagram [31]

Scenario	Q1		Q2		S	
	μ	σ	μ	σ	μ	σ
1	160	30	65	10	15	0
2	115	20	50	5	15	0
3	80	20	35	5	15	0

Table 1: Environmental flow variables in watershed management

Variables	Minimum Value	Maximum Value
α_1	8	24
α_2	8	30
α_3	8	20
α_4	8	24
α_5	8	20
α_6	8	30

Table 2: Range of agent requirements in watershed management

(3) The water requirements for each agent are considered to be static. Again, they can be variable over a long duration.
(4) The hypothetical scenario is only a one-step decision-making problem. Watershed management, however, can be a long term sequential decision-making task which can involve temporally stretched inter-agent interactions.

Thus, we model watershed management with dynamic water flows, changing minimum water requirements and temporally extended decision making across multiple timesteps. Furthermore, we introduce a fixed ordering in agents' decision making to simulate downstream water flow, thus preventing hard constraint violations.

3.1 Watershed Management as Sequential Decision Making

We consider timesteps in this model as one unit for which water flows and requirements are constant. An episode is simulated with T timesteps representing one annual cycle. Four agents are considered in the system, which are optimising six variables at every timestep, four of which are directly controlled and the remaining two are controlled indirectly (see Fig. 1). The four directly optimised variables are as follows:

(1) Water withdrawn for industrial and municipal use in the city represented by x_1.
(2) Water withdrawn for irrigating farms. In this system, two farms are considered. These variables are represented by x_4 and x_6.
(3) The water released from a dam that can be used for hydroelectric power generation. It is given by x_2.

We refer to agents controlling these variables as active agents. The two indirectly controlled variables represent water flowing through the two ecosystems and are given by the variables x_3 and x_5. These are referred to as reactive agents. We represent water withdrawn (for city and farm agents), water released (for dam agent) or water flowing through (for ecosystem agents) for i_{th} agent at t^{th} timestep as x_i^t.

Dynamic flow rates and water requirements
There are two streams in the system whose flow rates can be variable. These are represented by Q1 and Q2 cubic units. Furthermore, there is a dam in the system having a capacity of S cubic units. In [19], only three scenarios are considered for simulating different flow rates and dam capacities. In our formulation, for simulating changing environmental conditions, we allow the flow rates to be dynamic. However, to ensure that a feasible situation still exists for all agents, we sample the flow rates from a Gaussian distribution having mean value as one of the flow rates in the three scenarios proposed. This effectively allows the flow rates to change at every timestep as the scenario is also changing, and they can take on a wide range of values. Furthermore, since dam capacity is an infrastructure based value, we do not change it at every timestep. Table 1 shows the μ and σ values of Gaussian distribution from which flows are randomly sampled.

Similarly, for having dynamic water requirements for every agent, we take a uniform random sample between a minimum and maximum requirement range. The range values for α_i for i_{th} agent are given in Table 2.

Action Space: For each of the active agents, we consider action u_i^t as proportion of incoming water flow inf_i^t which the agent can access. Thus, an agent i can withdraw (or, for the dam agent, release) $x_i^t = u_i^t * inf_i^t$ amount of water where $u_i^t \in [0, 1]$. This makes sure that hard constraints are never violated. However, such a scheme requires that all reactive agents in a single timestep must act in a fixed order. Particularly, this order is starting from agents near waterbody source upstream towards agents that are downstream. Note that for reactive agents, $x_i^t = inf_i^t$.

Constraints: Only soft constraints can be violated in our setup. Moreover, these constraints are simplified as

$$x_i^t \geq \alpha_i^t \tag{1}$$

where α_i^t is water requirement for the i_{th} agent at t^{th} timestep

Agent Ordering: The ordering of agents for decision making is based on their spatial arrangement in the watershed basin. For the given hypothetical scenario, a valid ordering must satisfy the following conditions

(1) Agent 2 acts after agent 1
(2) Agent 6 acts after agent 1, agent 2, agent 4

In our experiments, we have used the ordering
Agent 1 $\longrightarrow$ $Agent2 \longrightarrow Agent4 \longrightarrow Agent6$

Further, the inflows for each agent can also be determined by this arrangement and defined recursively based on inflows of other agents. These are given in the following equations

$$inf_1^t = Q1^t \tag{2}$$

$$inf_2^t = inf_1^t(1 - u_1^t) \tag{3}$$

$$inf_3^t = inf_4^t(1 - u_4^t) \tag{4}$$

$$inf_4^t = Q2^t \tag{5}$$

$$inf_5^t = inf_6^t(1 - u_6^t) \tag{6}$$

$$inf_6^t = inf_3^t + (inf_2^t + S)u_2^t \tag{7}$$

Observation Space: The observation space for each agent consists of its own incoming flow as well as its personal water requirement at every timestep. Since we have defined some inflows in the observation vector, the current observation $o_i \in O_i$ thus becomes a function of other agents' actions. In our experiments, we further explore other scenarios where global information is available and when communication is used. These are described in the following sections.

Rewards: We define two kinds of reward function for each agent and one penalty function. One of the rewards is provided immediately at every timestep and is represented by f_i for the i^{th} agent. This immediate reward is a quadratic function of water withdrawn at the current timestep and is given by

$$f_i(x_i^t) = a_i(x_i^t)^2 + b_i x_i^t + c_i \tag{8}$$

where, for i^{th} agent, x_i^t is water withdrawn at t^{th} timestep and a_i, b_i, c_i are dimensionless constants given in Table 3. This reward conceptualizes the benefits gained from water for various purposes for different agents. Note that we will use a negative value for a_i which means that this quadratic function will have positive reward only for a particular range of x_i^t. This makes sure that both the scenarios where an agent withdraws too much or too little water will result in negative reward.

The other reward g_i is given at the end of episode. We define it as a linear function of cumulative water withdrawn by the i^{th} agent across T timesteps, normalised by total water requirement by that agent. This reward incentivises agents to extract more water in the long run and is larger for agents who have fulfilled their water requirements for most timesteps. The episode end reward function for the i^{th} agent is given by

$$g_i(t) = \begin{cases} 0 & t < T \\ \frac{X_i}{A_i} d_i & t = T \end{cases} \tag{9}$$

where X_i is the cumulative water withdrawn (or released for dam agent) and A_t is cumulative water requirements for i^{th} agent across T timesteps. d_i is a dimensionless constant which controls the amount of weight to be given for cumulative reward. We keep it same for all the agents, as is given in 3.

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
a_1	-0.20	b_1	6	c_1	-5	d_1	100
a_2	-0.06	b_2	2.5	c_2	0	d_2	100
a_3	-0.29	b_3	6.28	c_3	-3	d_3	100
a_4	-0.13	b_4	6	c_4	-6	d_4	100
a_5	-0.056	b_5	3.74	c_5	-23	d_5	100
a_6	-0.15	b_6	7.6	c_6	-15	d_6	100

Table 3: Watershed Constants [19]

$$X_i = \sum_{t=1}^{T} x_i^t \tag{10}$$

$$A_i = \sum_{t=1}^{T} \alpha_i^t \tag{11}$$

The penalty function incorporates soft violations, and we use the same penalty function as described in [19]. It is given by

$$f_p^t = \sum_{i=1}^{N} C(h_i^t + 1)\delta_i \tag{12}$$

where δ is 0 when no violation is made and 1 if there is constraint violation. C is 100 and h_i^t represents amount of violation made by i_{th} agent at the t^{th} timestep. N is total number of soft constraints, which in our case is equal to number of active agents. The reward function at every timsetep thus becomes

$$r_i(x_i^t) = f_i(x_i^t) + g_i(t) - f_p^t \tag{13}$$

And the optimization function that each agent has to maximize in an episode is thus given by

$$F_i(x_i) = max \sum_{t=1}^{T} [r_i(x_i^t)] \tag{14}$$

where x_i is the trajectory of water-actions taken by agent i.

3.2 Inter-Agent Communication as a Bargaining Process

In [31], the authors propose a bargaining scheme to allow agents to observe local information and send the solution to a centralised processor. This processor then computes violations and system costs, which is then available to all the agents as information in the next round. In our scenario, since we are simulating a completely decentralised system, we model bargaining as a communication process where messages are broadcasted to all agents prior to taking individual actions. Thus, for every agent, we define two policies. The first policy is for generating communication messages for every other agent, which are then broadcasted. The other policy is for taking water-withdrawl decisions after observing the communication messages from other agents as well as local information on water flow and requirements. We call them Communication Policy and Water-Action Policy, respectively. Thus, we define the watershed management game in two phases, first for bargaining using the communication policies. And the second phase for taking water-based actions. The communication policies for all agents take actions simultaneously while the water-action policies take actions

according to the ordering described in the previous section, after the communication phase.

Furthermore, we can model a complex bargaining process using multiple rounds of communication in the communication phase. The output of the previous communication round is given as input to the next round of communication. In this regard, we define the observation space of the i^{th} agent's communication policy to include (i) personal water requirement, (ii) flow rates of streams in the system (this is subject to global or local observation explained in section 5.1), (iii) communication messages from the previous round. Fig. 2 shows a schematic representation of using communication policy along with the water-action policy.

The action space of communication policy is a discrete message token for every other agent. This results in a vector of length $n-1$, where n are the number of agents in the system. We also fix the vocabulary size of tokens to V. The communication policies are given the same rewards as water-action policies. This incentivises communication policies to produce messages that benefit water-action policies. For multistep communication, the reward is 0 for all intermediary communication steps except the last one, before which the second phase begins.

3.3 Intrinsic Reward through Social Influence

The work proposed by [11] consider using an intrinsic reward for agents to improve coordination among self-interested agents. This intrinsic reward is determined by the agents' ability to causally influence other agents' actions. For implementing this, a separate 'model of (opponent) agent' is developed that predicts the actions of other agents. The authors argue that actions which cause a greater change in the behaviour of other agents are highly influential and must be rewarded. This causal influence is estimated by simulating counterfactual actions, that is actions an agent could have taken, and then their effect on other agents is determined.

The causal influence of agent j on k is computed as follows. First the probability of agent j's next action is computed conditioned on the current state s^t and action a_k^t of agent k at t_{th} time step. This is given as $p(a_j^t|a_k^t, s_j^t)$. Then the counterfactual action $\tilde{a}_k^t$ is replaced in place of actual action a_k^t. Several actions are then sampled, and the resulting distribution of j in each case is averaged, to obtained marginal policy of j. In the original paper, discrete actions are considered. However, in our work, watershed management has continuous actions. Hence, we use Monte-Carlo sampling to obtain the estimated marginal distribution of j.

The causal influence of agent k on agent j is computed as the discrepancy between the marginal policy of j and the conditional policy of j given k's action. This discrepancy is quantified using KL divergence between the two measures. The influence reward for agent k is thus given by

$$e_i^t = \sum_{j=0, j\neq k}^{N} \left[D_{KL} \left[p(a_j^t|a_k^t, s_j^t) || \sum_{\tilde{a}_k^t} p(a_j^t|\tilde{a}_k^t, s_j^t) p(\tilde{a}_k^t|s_j^t) \right] \right] \quad (15)$$

$$= \sum_{j=0, j\neq k}^{N} \left[D_{KL} \left[p(a_j^t|a_k^t, s_j^t) || p(a_j^t|s_j^t) \right] \right] \quad (16)$$

Thus the immediate reward for an agent becomes $r_i^t + \beta e_i^t$ where beta is the influence weight.

4 EXPERIMENTAL SETTINGS

4.1 Reward-Schemes, Observation Spaces, Communication

In our experiments, we compare the performance of Deep Multi-Agent RL Framework for watershed management problem across different reward schemes, observation spaces and the choice of using communication or not.

For rewarding the agent, we consider 2 scenarios:

(1) Global Reward : This is akin to the the setup in [19], where all agents are given the same reward. Here, the watershed management problem becomes a purely cooperative game. The global reward is equal to sum of individual reward functions for each agent as well as penalty on the system due to constraint violation. At t^{th} timestep, the reward given to i^{th} agent is given by

$$r_i(x_i^t) = \sum_{i=1}^{6} \left[f_i(x_i^t) + g_i(t) - f_p^t \right] \quad (17)$$

(2) Local Reward but with a global penalty: This rewarding scheme is more realistic as it makes the agents self-interested. However, when constraints are violated, the water deficit agents affect the whole system. This adds repercussion on actions taken by upstream agents which would otherwise never get affected by the condition of agents downstream. Thus, cooperative behaviour would be incentivised. At t^{th} timestep, the reward given to i^{th} agent is given by

$$r_i(x_i^t) = f_i(x_i^t) + g_i(t) - f_p^t \quad (18)$$

For observation space, we again consider two scenarios:

(1) Global Observation: In this scenario, at every timestep, all agents have full visibility of flow rates of mainstream and tributary as well as minimum water requirement of all the agents.
(2) Local Observation: In this scenario, at every timestep, an agent only has visibility of its own water inflow and minimum water requirement.

We also consider whether to use inter-agent communication for simulating a bargaining process. In addition to this, we also evaluate results on multistep communication with varying number of communication phases. Lastly, we introduce an experiment with additional intrinsic reward in the local observation, local reward setting and show its results.

4.2 Implementation Details

The neural networks used in all the policies, except for Intrinsic Reward Experiments, have the same architecture. All policies are recurrent policies, that is they are keeping the state from previous timesteps. First, the input is preprocessed by two fully connected layers with 16 hidden units each. This is followed by LSTM cell with 128 cell size. The results are again processed by a fully connected layer, and outputs are given. All water-action policies are Gaussian policies which output mean and variance of a gaussian distribution.

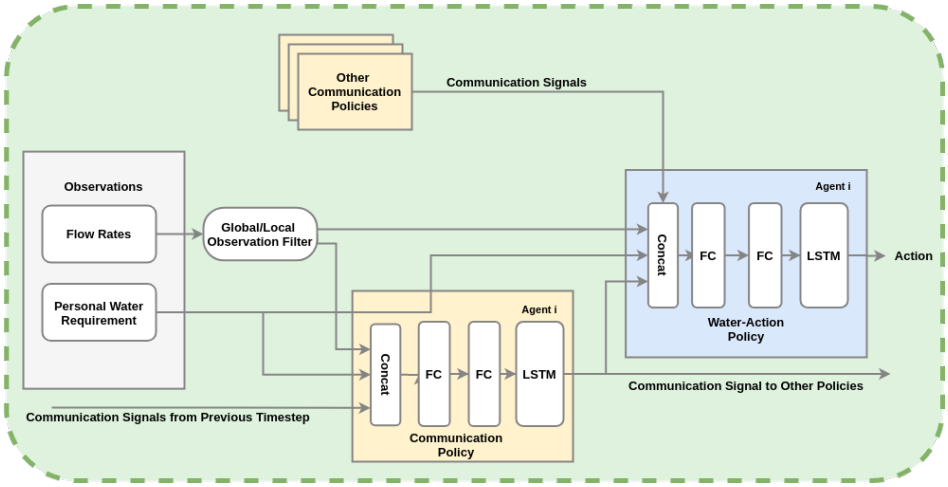

Figure 2: Schematic diagram of using communication policy along with water-action policy

The output is scaled between 0 and 1 using a hyperbolic tan function. For communication policies, V outputs are returned, which are probabilities corresponding to each of the communication symbols. Here, V is the communication message vocabulary size.

In the water-action policy for intrinsic reward experiment, we use the same hyperparameters and architecture as in [11], unless otherwise mentioned. An additional output head is used to model other agent's policies. Base layers are shared between this policy and the actual policy that takes water-actions. The shared layers consist of two fully connected layers with 16 hidden units each followed by LSTM cell with 128 cell size. The output from LSTM is post-processed with another fully connected layer. This output is then passed to the two output heads which process it with another fully connected layers based on their output sizes.

We optimize our models through Proximal Policy Optimization [25]. We use a learning rate schedule that decays linearly from 1e-4 to 1e-5 at 1 million timesteps and then stays constant. Generalized Advantage Estimation (GAE)[24] is used with GAE lambda value is taken as 1.0, and KL coefficient is fixed at 0.2. As proposed in [21], we use entropy regularization with entropy coefficient 0.001. The value function loss coefficient is taken as 1e-4. All experiments are ran for 200 iterations, each with 26000 training batches to stabilise training. The vocabulary size of communicating policies is fixed to 3, and all experiments are run with time horizon T as 10.

4.3 Evaluation Metrics

While in a single agent Reinforcement Learning, value function can be used as a measure of performance, the mixed incentives in multi-agent games don't allow for a straightforward metric for gauging performance. Hence, we use some social metrics proposed in [23] with some modifications, along with individual rewards for each agent. Particularly, we use Utilitarian and Equality metrics.
Utilitarian Metric: Measures the total sum of rewards for all the agents. It is given by

$$U = E\left[\frac{\sum_{i=1}^{N} r_i}{T}\right] \tag{19}$$

Equality Metric: Since rewards in our work can be negative, hence Gini coefficient can't be used. We thus use reciprocal of dispersion index to quantify equality. It is given by

$$E = \mu_r / \sigma_r^2 \tag{20}$$

where μ_r is the mean of rewards r_i's and σ_r is the standard deviation of r_i's for i from 1 to N and N is the number of agents.

5 RESULTS

We analyse the performance of different reward schemes, observation spaces, social influence and communication setups for the watershed problem. Table 4 summarises the utilitarian metric, equality metric and the average number of violations per step for each of the experiment. Since environments with global rewards form a fully cooperative system, we study them separately from local reward scenario. Also, equality metric is not reported in global reward scenarios. Fig. 3 and Fig. 4 show utility over training iterations for global reward and local reward scenarios, respectively.

It is seen that the highest utility is achieved in the experiment with global observation, global reward, without any inter-agent communication or social influence. The communication-based system, in the same scenario, has slightly lesser utility. We owe this to non-stationarity introduced by communication channels which leads to a sub-optimal solution. However, in local observation, local reward scenario, communication plays a crucial role and achieves a utility which is even better than global observation, local reward scenario utility without communication. This shows that inter-agent communication can provide necessary information that was otherwise included in global observation. However, in local observation, global reward, the communication-based system fails to find an optimal policy. We argue that having a global reward creates a credit assignment problem for communicative policy, thus making it hard to attribute reward for good or bad communication.

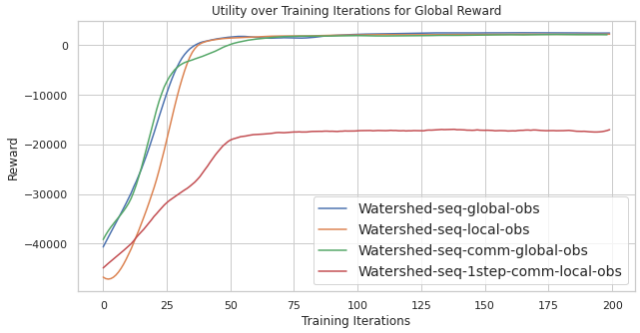

Figure 3: Utility across training iterations in global reward scenarios

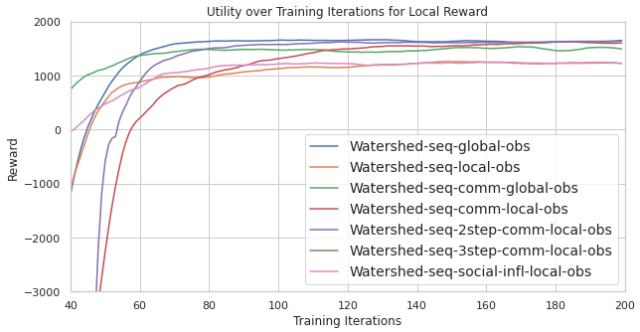

Figure 4: Utility across training iterations in local reward scenarios

Figure 5: Equality across training iterations

We also report results on 2-step and 3-step communication. It can be seen that 2-step communication brings slightly better utility as compared to single step communication. However, 3-step communication fails to converge in our case.

The last experiment of using social influence reward in local observation, local reward scenario has a very similar utility as for the experiment without social influence. This shows that communication is crucial for the watershed management task, and it can't completely be replaced by social influence.

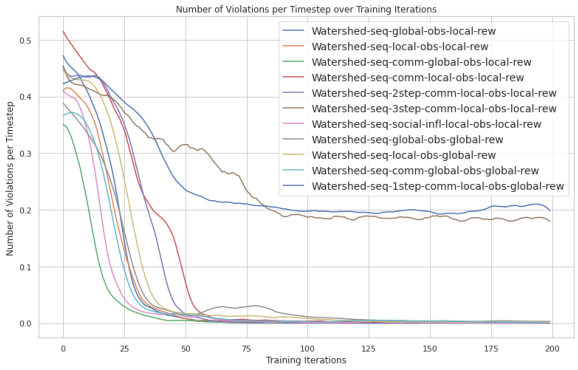

Figure 6: Average number of violations across training iterations

We also plot the average violations and equality metric over training iterations for all experiments (see Fig. 6 and Fig. 5). Since violations are heavily penalised, a general downward trend is seen in it in all the experiments. In Table 4, it can be seen that violations are close to zero at convergence. The equality metric, however, shows an increasing trend, even though explicitly equality isn't rewarded. We only plot equality over training iteration for local reward scenarios. We see that the highest equality is achieved in the scenario with global observation. However, for local observation, inter-agent communication achieves better equality than experiment without communication. Further, we again see that social influence experiment achieves equality very similar to experiment without social influence.

5.1 Communication Analysis

We further explore if communication-based policy shows any pattern in water-actions at different time steps in an episode. For this, we sample one trajectory each from with-communication and without-communication scenarios (with local observation, local reward). Fig. 7 shows water-actions in both polices for different agents. It can be observed that while the policy without communication takes safe actions, they do not show much variation. Communication-based policy, however, shows that at different timesteps, different agents change their action. This behaviour is desirable as agents can synchronise water-actions in such a way that congestion doesn't happen at a single timestep and greedy actions are distributed across the episode.

As results in Table 4 suggest that communication-based policy in local observation can achieve similar reward as communication-less policy in global observation, we try to find what kind of information these messages provide. In the local observation setting, the communication policy can only see personal water requirements. Hence, for every communication message, we plot the distribution of personal water requirements, binned into three categories (see Fig. 8. It can be seen that message-1 clearly favours first bin, message-2 third bin and message-3 has no clear winner. This suggests that there is some kind of one-to-one association between the communication message and water requirement, that is, agents communicate their water requirements through message symbols.

Reward Scheme	Observation Scheme	Communication	Social Influence	Utility	Equality	Average Violations
Global	Global	No	No	2430.048 ± 29.29	-	0.003 ± 0.0
Local	Global	No	No	1532.312 ± 16.06	0.043 ± 0.02	0.001 ± 0.0
Global	Local	No	No	2169.569 ± 27.92	-	0.002 ± 0.0
Local	Local	No	No	1228.566 ± 25.52	0.003 ± 0.002	0.002 ± 0.0
Global	Global	Yes, 1 step	No	2089.56 ± 32.13	-	0.001 ± 0.0
Local	Global	Yes, 1 step	No	1408.306 ± 45.54	0.012 ± 0.0035	0.001 ± 0.0
Global	Local	Yes, 1 step	No	-17321.083 ± 310.78	-	0.206 ± 0.01
Local	Local	Yes, 1 step	No	1660.134 ± 19.09	0.011 ± 0.003	0.001 ± 0.0
Local	Local	Yes, 2 step	No	1739.77 ± 21.43	0.002 ± 0.001	0.0 ± 0.0
Local	Local	Yes, 3 step	No	-54929.427 ± 1521.86	-0.02 ± 0.002	0.184 ± 0.0
Local	Local	No	Yes	1234.048 ± 30.94	0.0035 ± 0.001	0.001 ± 0.0
Local	Local	Yes, 1 step	Yes	1087.28 ± 125.45	0.0045 ± 0.001	0.008 ± 0.0

Table 4: Individual agent rewards, utilitarian metric, equality metric and the average number of violations per step for different experiments in the watershed management problem.

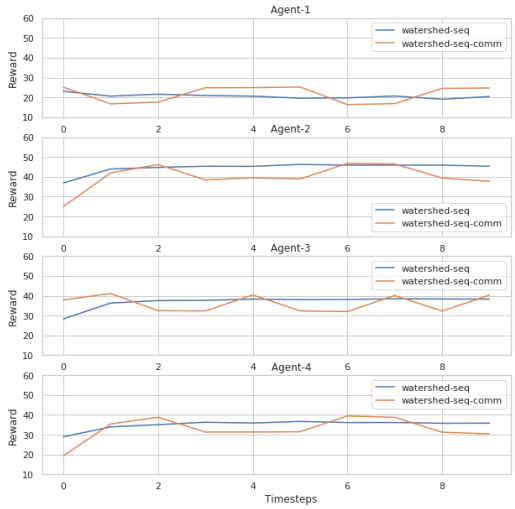

Figure 7: Water-actions by different agents across timesteps in an episode

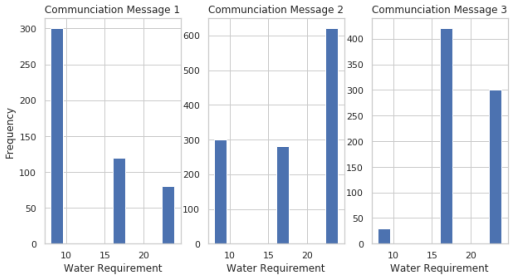

Figure 8: Water requirement distribution for different communication messages

However, it can't be ascertained without proper quantitative analysis. To stay within the scope of our work, we leave this as future work.

6 CONCLUSION AND FUTURE WORK

In this work, we have introduced a watershed management game with dynamic inputs and sequential decision making. Further, we have proposed a Deep Multi-Agent Framework for solving this watershed management problem. In this regard, several configurations are explored which correspond to varying rewarding schemes, observation paradigms, whether to include inter-agent communication and providing social motivation. It is shown that communication is meaningful, demonstrates a desirable behaviour in agents and helps improve the overall utility of the system. Social influence is also shown to be limited in this scenario without a proper communication channel. However, there is still scope of improvement as the best utility is achieved only by globally rewarding agents, which is a fully cooperative scenario. Thus we propose sequential watershed management as a testbed for research into the use of Deep MARL for real-world resource management problems.

For future work, we consider incorporating explicit communication channels which are given intrinsic reward using social motivation. There is also a scope of studying grounding of communication messages. Lastly, our study has been limited to a hypothetical watershed basin. We would like to apply Deep MARL for decision making in real-world watersheds and compare the results with approaches from other disciplines.

REFERENCES

[1] 2020. *World Water Development Report 2020 – Water and Climate Change.* Technical Report. United Nations Educational, Scientific, and Cultural Organization (UNESCO).

[2] Marc Brittain and Peng Wei. 2019. Autonomous air traffic controller: A deep multi-agent reinforcement learning approach. *arXiv preprint arXiv:1905.01303* (2019).

[3] Bryan Bruns and Pakping Chalad Bruns. 2004. *Strengthening collective action.* Technical Report.

[4] Daniel A DeCaro, Marco A Janssen, Allen Lee, et al. 2015. Synergistic effects of voting and enforcement on internalized motivation to cooperate in a resource dilemma. *Judgment and Decision Making* 10, 6 (2015), 511–537.

[5] Ruijin Ding, Yuwen Yang, Jun Liu, Hongyan Li, and Feifei Gao. 2020. Packet Routing Against Network Congestion: A Deep Multi-agent Reinforcement Learning Approach. In *2020 International Conference on Computing, Networking and Communications (ICNC).* IEEE, 932–937.

[6] Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems.* 2137–2145.

[7] Jakob Foerster, Richard Y Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. 2018. Learning with opponent-learning awareness. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems.* International Foundation for Autonomous Agents and Multiagent Systems, 122–130.

[8] Matteo Giuliani, A Castelletti, Francesco Amigoni, and X Cai. 2012. Multi-agent systems optimization for distributed watershed management. (2012).

[9] Edward Hughes, Joel Z. Leibo, Matthew G. Phillips, Karl Tuyls, Edgar A. Duéñez-Guzmán, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin R. McKee, Raphael Koster, Heather Roff, and Thore Graepel. 2018. Inequity aversion improves cooperation in intertemporal social dilemmas. arXiv:1803.08884 [cs.NE]

[10] Ivana Huskova and Julien J Harou. 2012. An agent model to simulate water markets. (2012).

[11] Natasha Jaques, Angeliki Lazaridou, Edward Hughes, Caglar Gulcehre, Pedro A. Ortega, DJ Strouse, Joel Z. Leibo, and Nando de Freitas. 2018. Social Influence as Intrinsic Motivation for Multi-Agent Deep Reinforcement Learning. arXiv:1810.08647 [cs.LG]

[12] Lufthansa Kanta and Emily Zechman. 2014. Complex adaptive systems framework to assess supply-side and demand-side management for urban water resources. *Journal of Water Resources Planning and Management* 140, 1 (2014), 75–85.

[13] Nevena Lazic, Craig Boutilier, Tyler Lu, Eehern Wong, Binz Roy, MK Ryu, and Greg Imwalle. 2018. Data center cooling using model-predictive control. In *Advances in Neural Information Processing Systems.* 3814–3823.

[14] Chih-Sheng Lee. 2012. Multi-objective game-theory models for conflict analysis in reservoir watershed management. *Chemosphere* 87, 6 (2012), 608–613.

[15] Joel Z Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. 2017. Multi-agent reinforcement learning in sequential social dilemmas. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems.* International Foundation for Autonomous Agents and Multiagent Systems, 464–473.

[16] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous control with deep reinforcement learning. arXiv:1509.02971 [cs.LG]

[17] Nguyen Cong Luong, Dinh Thai Hoang, Shimin Gong, Dusit Niyato, Ping Wang, Ying-Chang Liang, and Dong In Kim. 2019. Applications of deep reinforcement learning in communications and networking: A survey. *IEEE Communications Surveys & Tutorials* 21, 4 (2019), 3133–3174.

[18] Kaveh Madani. 2010. Game theory and water resources. *Journal of Hydrology* 381, 3-4 (2010), 225–238.

[19] Karl Mason, Patrick Mannion, Jim Duggan, and Enda Howley. 2016. Applying multi-agent reinforcement learning to watershed management. In *Proceedings of the Adaptive and Learning Agents workshop (at AAMAS 2016).*

[20] Bo Ming, Pan Liu, Shenglian Guo, Xiaoqi Zhang, Maoyuan Feng, and Xianxun Wang. 2017. Optimizing utility-scale photovoltaic power generation for integration into a hydropower reservoir by incorporating long-and short-term operational decisions. *Applied Energy* 204 (2017), 432–445.

[21] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning.* 1928–1937.

[22] Elinor Ostrom, Roy Gardner, James Walker, James M Walker, and Jimmy Walker. 1994. *Rules, games, and common-pool resources.* University of Michigan Press.

[23] Julien Perolat, Joel Z Leibo, Vinicius Zambaldi, Charles Beattie, Karl Tuyls, and Thore Graepel. 2017. A multi-agent reinforcement learning model of common-pool resource appropriation. In *Advances in Neural Information Processing Systems.* 3643–3652.

[24] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438* (2015).

[25] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[26] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. 2017. Mastering the game of go without human knowledge. *Nature* 550, 7676 (2017), 354–359.

[27] Elise Van der Pol and Frans A Oliehoek. 2016. Coordinated deep reinforcement learners for traffic light control. *Proceedings of Learning, Inference and Control of Multi-Agent Systems (at NIPS 2016)* (2016).

[28] Lizhong Wang, Liping Fang, and Keith W Hipel. 2008. Basin-wide cooperative water resources allocation. *European Journal of Operational Research* 190, 3 (2008), 798–817.

[29] Aaron T Wolf. 2007. Shared waters: Conflict and cooperation. *Annu. Rev. Environ. Resour.* 32 (2007), 241–269.

[30] Yaodong Yang, Jianye Hao, Mingyang Sun, Zan Wang, Changjie Fan, and Goran Strbac. 2018. Recurrent Deep Multiagent Q-Learning for Autonomous Brokers in Smart Grid.. In *IJCAI*, Vol. 18. 569–575.

[31] Yi-Chen E Yang, Ximing Cai, and Dušan M Stipanović. 2009. A decentralized optimization algorithm for multiagent system–based watershed management. *Water resources research* 45, 8 (2009).