Designing ethical environments using multi-agent reinforcement learning

Arnau Mayoral-Macau[*]
Artificial Intelligence
Research Institute (IIIA-CSIC)
Bellaterra, Spain
arnau.mayoral@iiia.csic.es

Manel Rodriguez-Soto
Artificial Intelligence
Research Institute (IIIA-CSIC)
Bellaterra, Spain
manel.rodriguez@iiia.csic.es

Enrico Marchesini
Massachusetts Institute of Technology
(MIT)
Massachusetts, US
emarche@mit.edu

Marti Sanchez-Fibla
AI Research Institute (IIIA-CSIC)
Bellaterra, Spain
marti.sanchez@iiia.csic.es

Maite Lopez-Sanchez[†]
Universitat de Barcelona (UB)
Barcelona, Spain
maite_lopez@ub.edu

Juan A. Rodriguez-Aguilar
AI Research Institute (IIIA-CSIC)
Bellaterra, Spain
jar@iiia.csic.es

Alessandro Farinelli
Università degli Studi di Verona
Verona, Italy
alessandro.farinelli@univr.it

ABSTRACT

This paper introduces the Approximate Ethical Embedding Process, an algorithm for automating the design of ethical environments for learning agents. Our algorithm helps build environments wherein multiple agents learn policies that align with an ethical (moral) value while simultaneously pursuing their individual objectives. Therefore, we contribute to endowing *environment designers* with algorithmic tools for building ethical environments. Moreover, we demonstrate the ethical design process for two different settings of a gathering environment, where agents have to adhere to beneficence to promote the collective survival of the population. Our experiments show that our approximate embedding process successfully generates environments that incentivise the learning of value-aligned policies.

KEYWORDS

Ethical values, learning for value alignment, multi-agent reinforcement learning.

1 INTRODUCTION

As autonomous agents gain more prevalence in daily tasks [5, 12, 35, 38], their risks also become more apparent. Thus, various international initiatives, such as the Artificial Intelligence (AI) Act [8], mandate the systems to behave aligned with human values [9, 27, 30]. Thereby, as Multi-Agent Reinforcement Learning (MARL) algorithms have found application in diverse domains, they have also been used to instil *value-alignment* in the context of Machine Ethics [26, 40].

Machine Ethics pursues that (ethical) value-aligned behaviour involves proactivity in performing good (praiseworthy) actions (e.g., [9]). To achieve such an objective, the literature on Machine Ethics has extensively used Reinforcement Learning (RL) to help agents learn to behave ethically. Specifically, it is common in the literature [1, 3, 16, 20, 32, 37] to adopt an *agent-centered* approach to value alignment: an agent is guided towards a value-aligned behaviour by providing it with extrinsic, manually-tuned ethical rewards incorporated into its learning environment.

Alternatively to the agent-centered approach, we find in the literature recent *environment-centered* approaches to value alignment [22, 23]. This strand of research takes an *environment designer* perspective. This perspective focuses on automating the design of an ethical environment — the so-called ethical embedding process — for either a single agent [22] or multiple agents [23]. In such ethical environments, agents are *guaranteed* to learn ethical policies. The environment-centered approaches to value alignment are particularly appealing with respect to agent-centered approaches because they provide theoretical guarantees regarding the learning of ethical policies and the automation of the reward design. Nonetheless, these approaches are based on strict theoretical assumptions (e.g., full observability, convergence to optimality of agents' learning), severely compromising their scalability.

Against this background, our goal is to contribute to the applicability of environment-centered approaches to value alignment. Thus, our primary contribution is a new ethical embedding algorithm (henceforth *approximate embedding*) that goes beyond the scalability of the embedding algorithm in [23] (henceforth *optimal embedding*). There are major differences between our approximate embedding and previous optimal embedding. First, our approximate embedding adopts more realistic assumptions (e.g., partial observability, and no need for MARL algorithms with convergence guarantees). Second,

[*] The research presented in this paper was supported by the EU-funded VALAWAI (# 101070930) project, the Spanish-funded VAE (# TED2021-131295B-C31) projects. This work was supported by grant PID2022-136787NB-I00 funded by MCIN/AEI/10.13039/501100011033. It was also financed by Fairtrans (PID2021-124361OB-C33), and grants 2021 SGR 00313 and 2021 SGR 00754.

[†] Maite Lopez-Sanchez belongs to the WAI research group (University of Barcelona) associated unit to CSIC by IIIA.

Proc. of the Adaptive and Learning Agents Workshop (ALA 2025), Avalos, Aydeniz, Müller, Mohammedalamen (eds.), May 19 – 20, 2025, Detroit, Michigan, USA, ala-workshop.github.io. 2025.

the approximate embedding builds on a completely different technical methodology compared with the optimal embedding, as we now detail.

Like optimal embedding, our approximate embedding takes as input a multi-objective environment, with *ethical* and *individual* objectives (rewards), to produce a single-objective ethical environment wherein agents learn. This approach prevents agents from learning unethical policies by ensuring that it is in their best interest to behave ethically. First, by considering more realistic assumptions, our approximate embedding requires introducing a new type of environment: an *ethical* Multi-objective Partially Observable Markov Game (MOPOMG), which extends MOPOMGs defined in [28]. Moreover, it implements two key computations differently from optimal embedding: (1) computing the *reference (ethical) joint policy* for agents to learn in an ethical environment; (2) computing the *ethical weight* to combine ethical and individual rewards into a single reward so that the ethical joint policy is the optimal policy to learn in the ethical environment.

In more detail, our approximate embedding computes a reference ethical joint policy as a Nash equilibrium of an ethical MOPOMG using Deep Reinforcement Learning (DRL). DRL has impressive results in approximating Nash equilibria despite its lack of theoretical guarantees [2, 13, 39]. Computing the ethical weight calls for introducing the Ethical Weight Finder, a novel algorithm based on binary search. In short, we make the following key contributions:

- We present a novel ethical embedding algorithm for building, under realistic assumptions, multi-agent ethical environments that incentivise the learning of ethical policies.
- We demonstrate how a novel multi-agent extension of a state-of-the-art multi-objective RL algorithm, lexicographic proximal policy optimization (LPPO) [29], can be applied to approximate ethical equilibria in Multi-objective Partially Observable Markov games. Our novel algorithm, Multi-Agent LPPO (MALPPO), utilises Lagrange multipliers to learn ethical policies, enabling agents to effectively adopt ethical policies within an ethical MOPOMG.
- We employ the approximate embedding to build ethical environments for an ethical version of the Gathering Game [11, 14]. We empirically demonstrate that approximate embedding produces ethical environments where agents do learn to adhere to the moral value of beneficence to guarantee the survival of the whole agent population.

2 BACKGROUND

The multi-agent reinforcement learning literature formally defines a multi-agent environment as a *Markov game* (MG) [2]. MGs are sequential decision-making settings where agents simultaneously act to modify the environment state and accumulate individual rewards. When agents have limited sensing capabilities over states, it is characterised as a *Partially Observable Markov Game* (POMG) [2]:

Definition 2.1 (POMG: Partially Observable Markov Game). A partially observable Markov Game (POMG) is defined as a tuple $\langle S, A^{i=1,\cdots,n}, R^{i=1,\cdots,n}, T, O^{i=1,\cdots,n}, \mathcal{O}^{i=1,\cdots,n}, \gamma \rangle$. Here, S is a finite set of states, and each A^i represents the set of actions available to agent i. The transition function $T : S \times A^{i=1,\cdots,n} \times S \to [0, 1]$ defines the probability of moving from state s to the next state s', given the joint action $a = \langle a^1, \dots, a^n \rangle$ of all agents. For each agent i, the reward function $R^i : S \times A^{i=1,\cdots,n} \times S \to \mathbb{R}$ specifies the reward r^i after applying joint action a to state s and transitioning to state s'. $O^{i=1,\cdots,n}$ is a finite set of observations and the function $\mathcal{O}^{i=1,\cdots,n} : A^{i=1,\cdots,n} \times S \times O^{i=1,\cdots,n} \to [0, 1]$ represents the probabilities over the agent's possible observations o^i given the state s and a joint action a. Finally, $\gamma \in (0, 1]$ is the discount factor which indicates how important future rewards are on the current state.

In MARL, each agent i aims to learn a policy (i.e., a behaviour) π^i that maximises its expected discounted accumulation of rewards $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r^t]$.

Similarly, RL defines the value[1] $V^{\pi}(s)$ as the expected discounted return a policy π gets from state s onwards. Knowing such value function enables an agent to choose the actions that lead to the most valuable states. However, computing an exact value function is computationally challenging for environments with large state and action spaces.

Typically, a joint policy $\pi = \langle \pi^1, \dots, \pi^n \rangle$ that maximises the return for all agents does not exist. Consequently, following the game theory literature, the main solution concept in MARL is reaching a *Nash equilibrium* (NE) [2], defined as a joint policy in which no agent can unilaterally improve its current accumulation of rewards:

Definition 2.2 (Nash equilibrium). Given a Partially-Observable Markov Game $\mathcal{M}$, a Nash equilibrium is a joint policy $\langle \pi^i_*, \pi^{-i}_* \rangle$ satisfying that for every agent i and every state observation $\vec{o} = (o^1, \dots, o^n)$, with each o^i in O^i, the policy π^i_* of agent i is a best-response against $\pi^{-i}_*(s)$, that is, it maximises the return against the joint policy of the rest of the agents π^{-i}_*:

$$V^i_{\langle \pi^i_*, \pi^{-i}_* \rangle}(\vec{o}) \geq V^i_{\langle \pi^i, \pi^{-i}_* \rangle}(\vec{o}), \text{ for every } \pi^i \text{and } \forall \vec{o} \in \vec{O}, \quad (1)$$

where $V^i_{\pi}(\vec{o})$ is the expected discounted accumulation of rewards $\mathbb{E}[\sum_{i=1}^{\infty} \gamma^i r^i \mid \pi, \vec{o}]$ of agent i if all agents follow the joint policy $\pi = \langle \pi^i, \pi^{-i} \rangle$ after observing $\vec{o}$.

Computing Nash equilibria in Markov games is a complex but well-studied problem [4, 17, 36]. The exact computation of equilibria is computationally intensive, with complexity increasing rapidly in high-dimensional environments. Recently, state-of-the-art single-agent algorithms leveraging deep neural networks —extended to multi-agent scenarios— have been employed to approximate equilibria under paradigms such as independent learning and centralised training with decentralised execution [6, 18, 39].

Multi-Objective MARL. When considering multiple ($m > 1$) learning goals in a POMG, agents aim to maximise the accumulation of rewards obtained with respect to each of the objectives in the so-called Partially Observable Multi-Objective Markov Game [28]:

Definition 2.3 (MOPOMG: Multi-Objective Partially Observable Markov game). A (finite) partially observable m-objective Markov game of n agents is defined as a tuple $\mathcal{M} = \langle S, A^{i=1,\cdots,n}, \vec{R}^{i=1,\cdots,n}, T, O^{i=1,\cdots,n}, \mathcal{O}^{i=1,\cdots,n}, \gamma \rangle$ whose elements $S, A^{i=1,\cdots,n}, T, O^i, \mathcal{O}^i$, and γ are defined exactly like those of an POMG. On a

[1] Here, “*value*" refers to a metric used in RL to evaluate the utility of states or actions. This term is not related to moral or ethical values.

MOPOMG, the reward function $\vec{R}^{i=1,\cdots,n} = (R^i_1, \cdots, R^i_m)$ is vectorial, where each scalar reward function $R^i_j \in \vec{R}^{i=1,\cdots,n}$ is the reward function regarding the j-th objective. Accordingly, at each simulation time step, the agent i gets a vectorial reward signal $\vec{r}^i = (r^i_1, \cdots, r^i_m)$.

In a MOPOMG $\mathcal{M}$, the value $\vec{V}^\pi$ of a policy π is a vector. Comparing vectorial returns requires additional information about objective priorities. This information, such as a user's prioritisation among objectives, is crucial for determining an optimal solution. A weight vector $\vec{w} \in \mathbb{R}^m$ can represent this prioritisation, allowing the scalarisation of value vectors as $\vec{w} \cdot \vec{V}^\pi$ into comparable scalar values. The single-agent MORL literature [25] defines the *convex hull* (*CH*) as the set of policies that are optimal for some scalarisation weight vector $\vec{w}$. Therefore, in the single-agent literature, the *CH* is a solution set containing the optimal policies for any linear prioritisations a final user might prefer. However, the *CH* has been extended to multi-agent scenarios *only* when agents fully cooperate and share exactly the same objective [24].

Besides linear scalarisation with a weight vector, multi-objective literature, also considers non-linear prioritisations. Lexicographic orders (LO) [33] are explicit orderings of the objectives where objectives are prioritised over those that follow them on the ordering. Thus, the two policies are compared objective-wise, following the LO. Imagine a policy π_1 that receives a return $\vec{V}^{\pi_1} = (3, 4, 5)$ and π_2 with $\vec{V}^{\pi_2} = (3, 5, 3)$. Then, for an LO $\ell_1 = \{V_1 \succeq V_2 \succeq V_3\}$, policy π_2 is better, as it has equal V_1 and has greater V_2; under another LO $\ell_2 = \{V_3 \succeq V_1 \succeq V_2\}$, π_1 is optimal over π_2 as it gets more return for V_3.

3 THE EMBEDDING PROBLEM

We aim to set up environments where agents learn to perform their tasks while being value-aligned. Crucially, we aim to design an environment in which the *only* optimal behaviour is the intended ethical behaviour within the environment.

Hence, our problem amounts to creating environments where agents are incentivised to learn behaviours aligned with an extrinsic ethical objective, independently of the learning algorithm they use.

Rather than developing a learning algorithm, we focus on transforming the agents' learning environment. This *environment-centered* perspective is motivated by the fact that such environments may be utilised by third-party entities whose choice of learning algorithms is unknown.

To design an *ethical* environment, our algorithm takes as input an initial (henceforth *source*) environment with two objectives: an individual task representing the agents' primary goal and an *ethical objective* that evaluates and quantifies the agents' alignment with a specific moral value. A moral value or ethical principle, in Ethics, represents a moral goal worth pursuing [34]. The environment is then transformed into a single-objective *ethical environment*, where the two objectives are embedded together in one reward function that prioritises the ethical objective. Therefore, maximising this single reward can only lead to value-aligned behaviour. Henceforth, we formalise the Ethical Embedding problem by defining: first, the agents' source and ethical environments; and then the problem itself of how to transform an original environment into an ethical environment wherein agents learn to behave ethically.

We model the agents' source environment as a 2-objective *Partially Observable* Markov game with: an individual reward function R^i_0 (the reward function that rewards each agent i for fulfilling its individual objective), and an ethical reward function R^i_e that rewards each agent i when behaving ethically. In our work, we employ the ethical framework presented in [21] to construct ethical reward functions grounded in the literature on ethical theory. However, any other ethical framework for properly designing R^i_e could be used. We refer to such family of Markov games as *Ethical Multi-Objective Partially Observable* Markov Games:

Definition 3.1 (Ethical MOPOMG). An Ethical Multi-Objective Partially-Observable Markov Game (EMOPOMG) $\mathcal{M}$ is defined as a tuple $\langle S, A^{i=1,\ldots,n}, R_0^{i=1,\ldots,n}, R_N^{i=1,\ldots,n}, R_E^{i=1,\ldots,n}, T, O^{i=1,\ldots,n}, O^{i=1,\ldots,n}, \gamma\rangle$ such that for each agent i: First, R^i_0 is the individual reward function of each agent i, representing their individual objective. Then, $R^i_N : \mathcal{S} \times \mathcal{A}^i \to \mathbb{R}^-$ is the *normative* reward function of each agent i, penalising blameworthy actions. Finally, $R^i_E : \mathcal{S} \times \mathcal{A}^i \to \mathbb{R}^+$ is the *evaluative* reward function of each agent i, rewarding praiseworthy actions.

Tuple elements $S, A^{i=1,\ldots,n}, T, O^{i=1,\ldots,n}, O^{i=1,\ldots,n}$, and γ of $\mathcal{M}$ are defined identically to Partially Observable Markov games (POMG).

We define an *ethical* equilibrium of an Ethical MOPOMG as a Nash equilibrium with respect to the ethical reward function $\vec{R}_e = \vec{R}_N + \vec{R}_E$, where $\vec{R}_e$ denotes $R_e^{i=1,\ldots,n}$. Among ethical equilibria, we highlight *best-ethical* equilibria. A best-ethical equilibrium π_* is a Nash equilibrium with respect to the individual reward function of agents, subject to also being an ethical equilibrium.

Notice that the notion of Ethical MOPOMG and ethical equilibrium of an Ethical MOPOMG generalise analogous concepts in [23], which considers full observability.

Thus, the *Ethical embedding problem* is: how to design, from a given Ethical MOPOMG $\mathcal{M}$, a (single-objective) POMG $\mathcal{M}_e$ that provides enough incentives to the agents to learn to behave ethically (a best-ethical equilibrium). These incentives are provided to agents by weighting ethical rewards with a large enough ethical weight $w_e > 0$ such that the (scalar) reward that each agent receives $R^i_0 + w_e \cdot R^i_e$ promotes the agents to behave ethically. Formally, we refer to our *target* environment as an *Ethical Partially Observable Markov Game*:

Definition 3.2 (Ethical Partially Observable Markov Game). Let $\mathcal{M}$ be an Ethical MOPOMG with reward functions R^i_0, R^i_e for each agent i. We refer to the *Ethical Partially Observable Markov Game* $\mathcal{M}_e$ associated with $\mathcal{M}$ to a (single-objective) POMG with reward function $R^i_0 + w_e \cdot R^i_e$ with $w_e > 0$, such that at least one Nash equilibrium of $\mathcal{M}_e$ is a best-ethical equilibrium in $\mathcal{M}$.

To ease notation, we mark $\mathcal{M}_{\langle w \rangle}$ to denote a single-objective POMG with rewards $R^i_0 + w \cdot R^i_e$ resulting of scalarising $\mathcal{M}$'s rewards (R^i_0, R^i_e), with ethical weight w, for each agent i.

Moreover, henceforth, we refer to an Ethical MOPOMG as a *source* environment and to its respective Ethical POMG as the *target* environment.

Next, Section 4 outlines our algorithm for designing ethical environments.

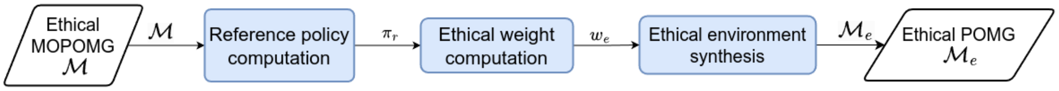

Ethical MOPOMG $\mathcal{M}$
$\mathcal{M}$
Reference policy computation
π_r
Ethical weight computation
w_e
Ethical environment synthesis
$\mathcal{M}_e$
Ethical POMG $\mathcal{M}_e$

Figure 1: Approximate Ethical Embedding Process.

4 SOLVING THE EMBEDDING PROBLEM

This section introduces *approximate embedding* as a process to design an ethical environment where all agents are incentivised to behave ethically. Figure 1 outlines the approximate embedding process. Following the environment-centered perspective previously mentioned, this process aims to design an ethical environment by transforming a multi-objective environment (a source environment $\mathcal{M}$) into an *ethical single-objective* environment (the target environment $\mathcal{M}_e$). Our purpose is that in the ethical environment $\mathcal{M}_e$, any third-party agent, independently of its learning algorithm, will learn a value-aligned behaviour. The approximate embedding process follows three steps:

(1) **Reference policy computation.** We compute a so-called *reference joint policy* π_r in the source environment. This is the ethical policy we want agents to, ultimately, jointly learn in the ethical single-objective, target environment.
(2) **Ethical weight computation.** We compute the ethical weight w_e to transform the source environment into our target environment.
(3) **Ethical environment synthesis.** We build the target environment as a POMG ($\mathcal{M}_e = \mathcal{M}_{\langle w_e \rangle}$) by scalarising the ethical rewards in the original environment using the ethical weight w_e computed at the previous step.

Subsections 4.1 and 4.2 detail the main steps of approximate embedding.

4.1 Reference policy computation

The first thing we must compute to obtain our target environment is an ethical policy that serves as a reference. Using the value vectors (multi-objective returns) of this policy, we will be able to properly design an ethical environment where ethical behaviour is preferred over individual behaviours. This reference policy is formally a best-ethical equilibrium in the source environment, the MOPOMG. As defined in Section 3, this best-ethical equilibrium corresponds to the equilibrium agents reach when prioritising ethical rewards over individual rewards. This reference policy π_r is important for two reasons. First, it is the joint policy we want to incentivise the agents to learn in the target environment. Second, we will need it to find a scalarisation weight, the so-called ethical weight w_e, to combine ethical and individual rewards in the target environment.

To obtain such an ethical reference policy, we need a learning algorithm that always prioritises ethical returns over individual returns. In the MORL literature, this type of non-linear prioritisation is studied under the name of lexicographic RL [10, 29, 33]. Lexicographic RL (LRL) prioritises objectives according to an explicit ordering.

Using a lexicographic learning algorithm in an ethical MOPOMG $\mathcal{M}$, which prioritises the ethical objective over the individual objective ($R_e \succeq R_0$), ensures the learned joint policy first optimises the ethical objective and then, without altering their obtained ethical return, optimise the individual objective. Notice that, when learning a joint policy in an ethical MOPOMG with a lexicographic algorithm, any Nash equilibrium will be a *best-ethical* equilibrium by definition.

Consequently, by using a lexicographic learning algorithm in the source environment $\mathcal{M}$, we can obtain ethical joint policies that abide by the ethical objective. This reference policy, π_r, represents the behaviour we aim to establish as the optimal policy in the resulting scalarised ethical environment $\mathcal{M}_e$, so that any agent learning in $\mathcal{M}_e$ finds it optimal to align with the moral value encoded in R_e.

However, no such algorithm exists for multi-agent environments. Subsequently, to learn a best-ethical equilibrium, we implemented a new lexicographic multi-agent algorithm. We provide more details on this contribution in Sections 5 and 6.2. The next section shows how to exploit a reference policy π_r to compute the ethical weight.

4.2 Ethical weight computation

Recall that our goal is to build a target environment $\mathcal{M}_e$ whose optimal policy for any learning algorithm is the (ethical) reference policy π_r. Since the target environment $\mathcal{M}_e$ is intended to be single-objective, we must find an ethical weight w_e that scalarises the ethical reward in the original environment $\mathcal{M}$ and sets π_r as the equilibrium in target environment $\mathcal{M}_e$.

We can exploit the multi-objective returns of the reference policy $\vec{V}^{\pi_r}$ to compute the ethical weight. Since we know that the reference policy is the one that attains the largest ethical return, the ethical weight w_e that we choose to scalarise the source environment $\mathcal{M}$ must be large enough to make π_r the optimal policy in the scalarised environment $\mathcal{M}_e$. More technically, we must compute an ethical weight w_e such that the Nash equilibrium π_r attains more scalarised return than any other equilibrium in $\mathcal{M}_e$. Thereafter, we will be able to build our target environment $\mathcal{M}_e$ using w_e wherein the (ethical) reference policy stands as the optimal policy to learn.

When computing the ethical weight w_e, we target an ethical weight w_e *as low as possible* while still capable of *incentivising* the learning of ethical equilibria. There are two reasons for searching for such an ethical weight. First, a large ethical weight could hinder or even prevent agents from learning their individual objectives, as demonstrated in the experiments section. Second, we consider that a reward function might have an associated cost when deploying the agents. Thus, an excessive weight w_e would involve a higher cost.

Since we know that an ethical equilibrium π_r prioritising ethical reward exists, we assume there must be an ethical weight w_e sufficiently large such that when learning with scalarised objectives $R = R_0 + w_e \cdot R_e$, the ethical objective is completely prioritised over the individual objective.

With that assumption in mind, we argue that, in the space of possible weights, we can differentiate two intervals: (i) $[0, w_e)$ with weights that do not incentivise the learning of an ethical equilibrium enough, and (ii) $[w_e, \infty)$ with ethical weights that effectively incentivise ethical behaviour. Note that any weight w in the interval $[w_e, \infty)$ allows the building of a scalarised, target environment whose ethical equilibrium is the reference policy. However, for the reasons mentioned above, we pursue the lowest possible ethical weight w_e.

To identify w_e, we need a method for testing whether a given weight w qualifies as an ethical weight. Once computed a reference policy π_r, testing whether a weight w is ethical is straightforward: (1) build a scalarised environment $\mathcal{M}_{\langle w \rangle}$; (2) learn the optimal joint policy π in $\mathcal{M}_{\langle w \rangle}$ (the so-called *approximate reference policy*); (3) compare the ethical returns of π and π_r. If π achieves *close enough* ethical return to the reference policy's ethical return (i.e., $|V_e^{\pi} - V_e^{\pi_r}| < \tau$ for some *policy approximation error* $\tau > 0$), then we consider w as an ethical weight. We do so because it sufficiently incentivises the learning of a policy that is as ethical as the reference policy. For simplicity, henceforth, we will refer to testing whether a weight w is ethical as function $test(\mathcal{M}_{\langle w \rangle}, \tau)$.

This definition for a single-agent $test(\mathcal{M}_{\langle w \rangle}, \tau)$ function can be extended to multi-agent setups simply by adding a sum over the agents returns such that $\left| \sum_{i=1}^{n} \left[V_e^{\pi^i} - V_e^{\pi_r^i} \right] \right| < \tau$.

By repeatedly using the $test$ function, we get valuable information on the value of w_e needed to create an environment whose equilibrium is π_r. Therefore, the set of joint policies obtained after testing different weights on a MOPOMG is relevant to inform our search for the minimum ethical weight. To characterise this set more rigorously, we introduce the *Nash Convex Hull* (*NCH*) for a MOPOMG, which we define as the set of joint policies that include a Nash equilibrium for any possible linear scalarisation weight vector. Formally:

Definition 4.1 (Nash Convex Hull). The Nash Convex Hull (*NCH*) of a MOPOMG $\mathcal{M}$ is the set of policies that contains the Nash equilibrium of all possible scalarisations $\mathcal{M}_{\langle \vec{w} \rangle}$ performed with any linear scalarisation weights $\vec{w}$.

$NCH(\mathcal{M}) = \{\pi \in \Pi^{\mathcal{M}} | \exists \vec{w} : \pi \in NE(\mathcal{M}_{\langle \vec{w} \rangle})\}$,

where $\Pi^{\mathcal{M}}$ is the set of possible policies in $\mathcal{M}$ and $NE(\mathcal{M}_{\langle \vec{w} \rangle})$ is the set of equilibria that exist in the scalarised POMG $\mathcal{M}_{\langle \vec{w} \rangle}$.

To obtain the ethical weight, there is no need to compute the whole NCH. Consequently, the resulting set of our search is a subset of the NCH that contains the Nash equilibria associated with the candidate ethical weights considered throughout the search for the minimal ethical weight. Additionally, considering that the learning algorithm might lack guarantees to find exact equilibria, we frame the set of policies we compute as an *approximate NCH*. Later in this section, we detail the construction of this NCH.

With $test(\mathcal{M}_{\langle w \rangle}, \tau)$ and considering a finite search space between 0 and an upper bound set by the environment designer based on domain knowledge, we can use any search algorithm to find our desired ethical weight. We propose using a general search paradigm like binary search [15] to automate the search of an *ethical weight* inside a search interval $w_e \in I$.

In particular, we propose to follow the principles of binary search. Then, our weight computation would work as follows. Let $I = [w_l, w_r]$ represent the unexplored search space, where w_l is not an ethical weight and w_r is an ethical weight. We can set w_l to 0 because it produces an environment without ethical rewards, and hence no ethical policies. Regarding w_r, the environment designer can set it to any large upper-bound number based on their expert knowledge. Thus, the search would begin with an initial, non-negative solution $w_e = w_r/2$, along with a precision parameter ϵ, which controls the depth of the search, and a policy approximation parameter τ. We test with $Test(\mathcal{M}_{\langle w_r/2 \rangle}, \tau)$ whether $w_r/2$ is an ethical weight. That is, we test whether the optimal policy in $\mathcal{M}_{\langle w_r/2 \rangle}$ is as ethical as the reference policy. If it is, we update the interval to $I = [w_l, w_r/2]$. If not, we update it to $I = [w_r/2, w_r]$. The procedure continues iteratively narrowing interval I until the distance between its endpoints is less than or equal to ϵ, while ensuring that the left endpoint remains a non-ethical weight and the right endpoint remains an ethical weight. The right endpoint is the approximation to the minimum ethical weight we are looking for. We refer to the whole process as the *Ethical Weight Finder* (EWF). The full pseudocode of the EWF is shown in 1.

Algorithm 1 Ethical Weight Finder

```
procedure EWF($\mathcal{M}$, $w_l$, $w_r$, $\epsilon$, $\tau$)
    $d \leftarrow w_r - w_l$
    while $d > \epsilon$ do
        $w \leftarrow w_l + \frac{d}{2}$
        if TEST($\mathcal{M}_{\langle w \rangle}$, $\tau$) then
            $w_r \leftarrow w$
        else
            $w_l \leftarrow w$
        end if
        $d \leftarrow w_r - w_l$
    end while
    return $w_r$
end procedure
```

It is important to note that this algorithm does not compute the exact w_e that makes π_r the optimal policy. Instead, it constructs an approximate subset NCH around such theoretical w_e. Since we progressively narrow the interval containing the exact w_e to a width of ϵ, by the end of the search, we obtain a set of policies that are Nash equilibria for weights close to w_e. As environment designers, we then select the weight that best incentivises ethical behaviour.

In this section, so far, we have demonstrated how to perform a binary search to identify the desired ethical weight w_e, minimising the number of times we run a learning algorithm, as this is the most computationally expensive part of the search. The approach described so far is only informed by the value vectors. We tried to develop a method as generic as possible following the lines of [31]. However, we recognise that the search can be optimised in different ways that may result in a more optimal search for specific environments, i.e. any environment-specific heuristic.

5 IMPLEMENTING APPROXIMATE EMBEDDING

In this section, we detail how to compute the two main elements of our approximate embedding process: the reference joint policy and the approximate reference policies, both of which are necessary for finding our desired ethical weight.

Recall that the first element, the reference policy, is a best-ethical equilibrium in a source *multi-objective* POMG. Thus, its computation requires a lexicographic multi-objective RL algorithm. Recently, the work in [29] introduced value-based and policy-based lexicographic RL algorithms (Lexicographic DQN, A2C, and PPO). Nonetheless, those algorithms only exist and were tested for the single-agent case. Therefore, to compute a reference policy, we are forced to develop a multi-agent version of a state-of-the-art lexicographic RL algorithm, as we do and detail in Section 6.

The second type of elements we need to compute in our embedding, approximate reference policies, are formally Nash equilibria in a *single-objective* POMG. Thus, we can find them using single-objective multi-agent deep reinforcement learning (MADRL) techniques. Indeed, MADRL techniques have shown great results in approximating equilibria for complex cooperative games [6, 7, 39] by extending powerful single-agent algorithms to MARL scenarios. Examples include IPPO and IA2C [2], as well as their centralised training and decentralised execution (CTDE) counterparts, MAPPO and MAA2C [2].

The next section presents a practical example using our process from Section 4 to design an ethical environment. There, we detail the lexicographic RL algorithm employed to compute a reference policy together with the MADRL algorithm employed to approximate Nash equilibria. Moreover, we examine the learning cost of the whole design process.

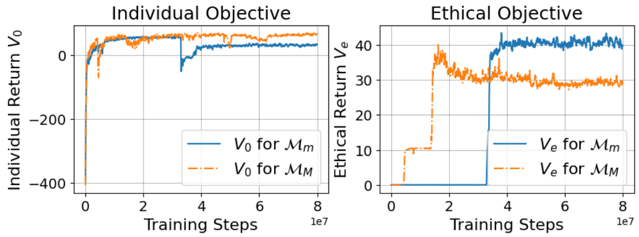


Figure 2: Sum of the individual and ethical returns of all agents for the learned reference policies in environments $\mathcal{M}_m$ (solid line) and $\mathcal{M}_M$ (dash-dot line).

6 EMPIRICAL ANALYSIS

In this section, we employ our approximate embedding to build ethical environments for the *Ethical Gathering Game* (EGG), an ethical extension of the Gathering Game originally introduced in [14] and outlined in Section 6.1. Our purpose is twofold: (1) to quantify the *price to pay*, the cost, to fabricate ethical environments through approximate embedding (Section 6.2); and (2) to empirically verify that agents do learn best-ethical policies in the designed ethical environments (Section 6.3).

6.1 The Ethical Gathering Game

The *Ethical Gathering Game* (EGG) is a grid-world environment where agents with different capabilities gather apples to survive [23]. An agent survives if it manages to accumulate apples beyond some survival threshold. The EGG is an ethical extension of the well-known Gathering Game [14] in the MARL literature. The EGG simulates an unequal environment where only efficient agents can survive by themselves while inefficient agents require the efficient agents' help.[2] The EGG also introduces a *public* donation box: agents can either donate apples to the box or retrieve apples from it. Then, two objectives drive agents in the EGG: an *individual* objective to collect apples for personal survival, and an *ethical* objective to contribute to collective survival. Therefore, our goal as environment designers will be to create ethical environments whose agents learn the moral value of *beneficence* to help the whole agent population survive. For that, we will instil alignment with beneficence through approximate embedding.

Importantly, since the EGG uses the same grid size as the original Gathering game, any EGG environment goes far beyond the capabilities of *optimal embedding* approaches (e.g., [22, 23]) to build ethical environments. Therefore, discard using optimal embedding to design EGG ethical environments.

Source environments. We considered two source environments with five agents, a survival threshold set to 40 and a 15-apple donation box capacity each: $\mathcal{M}_m$ contains a minority of efficient agents (40%), and $\mathcal{M}_M$ contains a majority of efficient agents (80%).

MOPOMG formalisation. The EGG is a MOPOMG whose agents act during a 500-step gathering season. To specify a *source environment*, following Def. 3.1 of Ethical MOPOMG, we define an individual reward function R_0^i that gives a reward of +1 to an agent for each collected apple. There are two types of agents: an inefficient agent has a 15 % chance of gathering an apple from the ground, while an efficient agent has an 85 % chance. An agent receives a negative reward $R_0^i = -1$ when: (i) being below the survival threshold at a given time step, or (ii) donating an apple to the public donation box. As to ethical reward, $R_e^i = R_N^i + R_E^i$, we penalise *unethical actions* ($R_N^i = -1$ when taking an apple from the donation box despite having enough to survive), and positively reward *ethical actions* ($R_E^i = 0.7$ when donating an apple when having enough to survive, namely accumulated apples beyond the survival threshold).

Metrics. Once designed an ethical environment, we will have the agents independently learn their policies. We will then employ two metrics to analyse whether they have learned to behave ethically: (i) the **survival rate** (how many times *all* agents are able to reach the survival threshold on average); (ii) the **ethical returns** (the accumulation of discounted ethical rewards that all agents obtain on average).

6.2 Designing ethical environments

Approximate embedding implementation. As discussed in Section 5, we must select the MADRL algorithms that implement our approximate embedding process. From Section 4.1, the first step must compute a reference policy using a lexicographic algorithm. Because of the action-space size and number of agents, we would need a lexicographic MADRL algorithm. Due to a lack of such multi-agent algorithms in the literature, we developed our so-called Multi-agent Lexicographic PPO—a novel algorithm extending MAPPO [39] with a lexicographic order along the lines of [29].

Thereafter, as explained in Section 4.2, we must compute several Nash equilibria in multiple scalarised POMGs during the search

[2] The Gathering Game [14] deals with resource depletion, a different problem from ours, for which agents must coordinate to survive.

for the ethical weight. Thus, we applied standard MAPPO for this second step. Finally, we run MALPPO and MAPPO using the EPyMARL library [19].

Learning reference policies. We applied approximate embedding to our two source environments: *minority* ($\mathcal{M}_m$) and *majority* ($\mathcal{M}_M$). We first computed the reference joint policy for each environment using MALPPO, our novel lexicographic algorithm. Figure 2 presents the expected (individual and ethical) returns that the reference policy obtained in each environment. We summed up the returns over all agents and smoothed them using an exponential moving average at 0.6 to reduce variance visually.

Finding ethical weights. We applied the EWF algorithm to search for the ethical weight for our two EGG environments. Recall that an ethical weight allows us to scalarise ethical rewards from a source environment so that the reference (ethical) policy becomes optimal in an ethical, target environment. For both environments, we used $[0, 10]$ ($w_l = 0, w_r = 10$) to search for ethical weights, $\tau = 4.0$ as policy approximation error, and $\epsilon = 0.2$.

Now, for each environment, we search for an as-low-as-possible ethical weight w_e that produces a scalarised environment $\mathcal{M}_{\langle w_e \rangle}$ whose Nash equilibrium is ethical, namely a good approximation of the reference policy. Figure 3 shows the evolution of the search for the ethical weight in each environment for six iterations. For iteration i and ethical weight w_e (different for each i), we show the multi-objective expected returns of policy i (blue circle), which we previously computed with MAPPO as a Nash equilibrium in the scalarised environment $\mathcal{M}_{\langle w_e \rangle}$. As iterations increase, w_e evolves as well as the (approximate) Nash equilibrium in each $\mathcal{M}_{\langle w_e \rangle}$. The Nash equilibrium for each w_e can be regarded as an approximation to the reference policy, and the set of policies that we obtain, as an approximate *NCH* as defined in Def.4.1. Notice that the *approximate* reference policies within each shaded grey area can be considered ethical because they are close enough, according to policy approximation error τ, to the reference policy ($\left| \sum_{i=1}^{n} \left[V_e^{\pi^i} - V_e^{\pi_r^i} \right] \right| < \tau$ considering all n agents). From the resulting set of policies, we finally select $w_e = 2.5$ and $w_e = 1.71875$ as ethical weights for $\mathcal{M}_m$ for $\mathcal{M}_M$ respectively because they are the lowest weights leading to good enough approximations of their reference policies. In detail, our EWF converged to these points because we obtained the final intervals $I_m = [2.34375, 2.5]$ for $\mathcal{M}_m$, and $I_M = [1.56, 1.71875]$ for $\mathcal{M}_M$, with both intervals with a length smaller than $\epsilon = 0.2$, and with the left extrema being unethical, and the right extrema being ethical. Notably, in Figure 3, we observe that using a weight larger than strictly necessary (i.e., policies learned with $w_e = 5$) is detrimental to individual return while providing only a marginal improvement in ethical return. This underscores the importance of identifying the approximate minimal weight as we motivated in Section 4.2.

Embedding cost. To measure the cost of the design process, we analyse the number of algorithm executions required to compute: (i) the reference policy; and (ii) the approximate reference policies when searching for the ethical weights. For each source environment, we run MALPPO once (to compute each reference policy), and we run MAPPO six times (during the ethical weight search). The total amount of steps needed to learn the necessary policies to design ethical environments is 420M for both $\mathcal{M}_m$ and $\mathcal{M}_M$. MALPPO needed 80M steps to learn both policies, while the different runs of MAPPO during both binary searches needed 70M on average.

We conclude that there is a price to pay to design an ethical environment wherein agents are incentivised to learn an ethical policy. In this particular experiment, designing each EGG ethical environment required running our MARL algorithms seven times. The next section shows that this investment paid off because agents did learn ethical policies in the EGG ethical environments.

Agents	**Ethical Return π_r**	**Ethical Return π_***	**Individual Return π_r**	**Individual Return π_***
1	17.54 ± 8.67	18.35 ± 5.83	26.15 ± 16.97	27.18 ± 9.53
2	17.61 ± 8.53	17.51 ± 5.79	26.4 ± 16.78	26.17 ± 9.68
3	1.69 ± 1.42	0.77 ± 1.09	-8.01 ± 10.87	-8.85 ± 8.24
4	1.7 ± 1.31	0.54 ± 0.91	-7.83 ± 10.64	-10.16 ± 8.70
5	1.64 ± 1.34	0.60 ± 0.85	-7.63 ± 10.52	-10.36 ± 8.62
Efficient	17.575 ± 0.035	17.93 ± 0.42	18.38 ± 0.13	26.67 ± 0.50
Inefficient	1.676 ± 0.026	0.63 ± 0.09	-7.82 ± 0.15	-9.79 ± 0.67

Table 1: Value vectors of each agent for the reference policy π_r and the ethical policy π_* learned in the designed $\mathcal{M}_m$. Results correspond to an average of 2000 rollouts. Recall that π_r is the same policy for all agents but it is not the case of π_*.

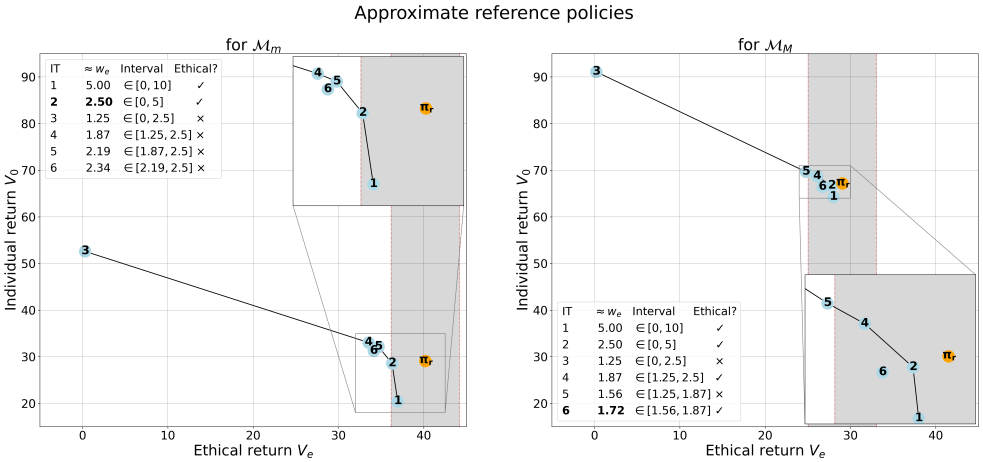
Agents	**Ethical Return π_r**	**Ethical Return π_***	**Individual Return π_r**	**Individual Return π_***
1	7.13 ± 3.97	7.26 ± 4.59	19.11 ± 9.49	19.70 ± 10.64
2	6.96 ± 4.13	7.09 ± 4.81	18.59 ± 9.63	18.10 ± 10.18
3	7.08 ± 3.89	7.49 ± 4.74	18.75 ± 10.01	20.30 ± 10.80
4	7.05 ± 3.96	7.41 ± 4.62	18.65 ± 9.77	20.40 ± 10.29
5	0.83 ± 0.66	0.00 ± 0.02	-7.93 ± 5.86	-18.10 ± 8.44
Efficient	7.05 ± 0.06	7.34 ± 0.26	18.775 ± 0.2	19.92 ± 1.16
Inefficient	0.83 ± 0.00	0.00 ± 0.00	-7.93 ± 0.00	-18.10 ± 0.00

Table 2: Value vectors of each agent for the reference policy π_r and the ethical policy π_* learned in the designed $\mathcal{M}_M$. Results correspond to an average of 2000 rollouts. Recall that π_r is the same policy for all agents but it is not the case of π_*.

6.3 Learning in the ethical environments

Finally, we built the ethical environments for the source environments $\mathcal{M}_m$ and $\mathcal{M}_M$ using the weights computed above. In these ethical environments, we let agents learn with a single-objective learning algorithm to test whether they do learn to behave ethically. Specifically, we employed independent PPO (IPPO) [7], where all agents learn independently within each environment.

The results obtained in the designed environments show that, indeed, agents were incentivised to learn a joint policy as ethical as the reference policy, as the metrics in Table 3 illustrate. First of all, the collective survival is achieved almost 100% of the times for both the reference policy π_r and the joint policy π_* that agents learn in both target ethical environments (column 3 of Table 3). For comparison, an unethical policy π_u, trained in an environment with ethical weight $w_e = 0$, reaches a much lower level of collective survival. Moreover, in terms of ethical returns, joint policies learnt



IT	$\approx w_e$	Interval	Ethical?
1	5.00	$\in [0, 10]$	✓
2	**2.50**	$\in [0, 5]$	✓
3	1.25	$\in [0, 2.5]$	×
4	1.87	$\in [1.25, 2.5]$	×
5	2.19	$\in [1.87, 2.5]$	×
6	2.34	$\in [2.19, 2.5]$	×

IT	$\approx w_e$	Interval	Ethical?
1	5.00	$\in [0, 10]$	✓
2	2.50	$\in [0, 5]$	✓
3	1.25	$\in [0, 2.5]$	×
4	1.87	$\in [1.25, 2.5]$	✓
5	1.56	$\in [1.25, 1.87]$	×
6	**1.72**	$\in [1.56, 1.87]$	✓

Figure 3: Approximate reference policies obtained in the search for the ethical weights of environments $\mathcal{M}_m$ (left) and $\mathcal{M}_m$ (right). Black lines show the approximate *NCH* around the exact ethical weight needed to build an ethical environment.

Policy	Survival rate in $\mathcal{M}_m$	Survival rate in $\mathcal{M}_M$
π_r	100%	99%
π_*	100%	98%
π_u	10%	66%

Table 3: EGG metrics for the reference (π_r), ethical (π_*), and unethical (π_u) policies. The expected ethical return of the policies is shown only for the efficient agents of the population.

in the target ethical environments obtain very close average returns to those obtained by the reference policies (column 4 of Table 3).

Considering all agents returns, not only efficient agents, Tables 1 and 2 show the value vectors of the ethical policy π_* and the reference policy π_r for all agents. Overall we can see how the value vectors are within the tolerance parameter $\tau = 4$ we established for the ethical return. Interestingly, the policies π_* learned in the designed environments achieve better individual returns than the reference policy. When looking at the statistics of the ethical gathering game rollouts, we find that agents in π_* learned to gather more apples than those in π_r in the same amount of time.

Note that both policies have been achieved through different algorithms. While in the reference policy, agents shared the same value and policy network, in the ethical policy, each agent learned its own value and policy networks. This can explain the difference between deviation for agents sharing efficiency groups in both policies.

Finally, Figure 4 helps us understand the policies that inefficient and efficient agents learn in the majority, $\mathcal{M}_M$, environment. Agents learn analogous policies in the minority, $\mathcal{M}_M$, environment The figure shows the median number of apples (over 2000 policy rollouts) efficient and inefficient agents have throughout the episode. We observe that, in all runs, efficient agents have learned to donate their surplus apples to aid inefficient agents in survival. Moreover, agents collect apples from the donation box only when they do not have enough apples to survive. This behaviour confirms again that, indeed, agents have been incentivised to learn to behave ethically.

In summary, all results from the experiments we conducted corroborate that the approximate embedding algorithm can design environments where agents learn best-ethical policies.

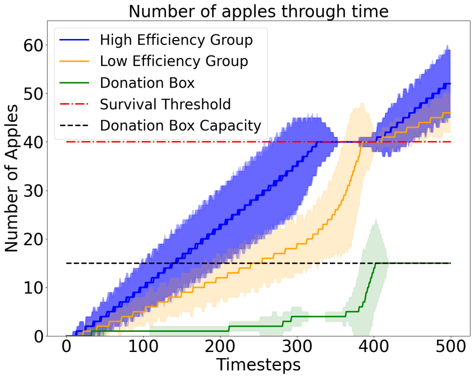


Figure 4: Median number of apples (and inter-quartile ranges) collected by agents throughout 2000 episodes of 500 steps in $\mathcal{M}_M$.

7 CONCLUSIONS AND FUTURE WORK

This work introduced the *Approximate Embedding*, an algorithm for designing environments where all agents are incentivised to learn to behave ethically, independently of the learning algorithm used. Our empirical analysis shows that by combining MORL and deep reinforcement learning tools, our *approximate embedding* successfully incentivises the learning of ethical policies in a large environment such as the *Ethical Gathering Game*. As future work, we plan on evaluating our approximate embedding in further environments. Nevertheless, this will first require the engineering of more MARL environments with ethical considerations because they do not currently exist.

REFERENCES

[1] David Abel, James MacGlashan, and Michael L Littman. 2016. Reinforcement Learning as a Framework for Ethical Decision Making.. In *AAAI Work.: AI, Ethics, and Society*, Vol. 92.

[2] Stefano V. Albrecht, Filippos Christianos, and Lukas Schäfer. 2024. *Multi-Agent Reinforcement Learning: Foundations and Modern Approaches*. MIT Press. https://www.marl-book.com

[3] Avinash Balakrishnan, Djallel Bouneffouf, Nicholas Mattei, and Francesca Rossi. 2019. Incorporating Behavioral Constraints in Online AI Systems. *Proceedings of the AAAI Conference on Artificial Intelligence* 33 (07 2019), 3–11. https://doi.org/10.1609/aaai.v33i01.33013

[4] Lucian Buşoniu, Robert Babuška, and Bart De Schutter. 2010. Multi-agent reinforcement learning: An overview. *Innovations in multi-agent systems and applications-1* (2010), 183–221.

[5] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Kaijie Zhu, Hao Chen, Linyi Yang, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2023. A survey on evaluation of large language models. *arXiv preprint arXiv:2307.03109* (2023).

[6] Filippos Christianos, Georgios Papoudakis, and Stefano V. Albrecht. 2023. Pareto Actor-Critic for Equilibrium Selection in Multi-Agent Reinforcement Learning. arXiv:2209.14344 [cs.LG] https://arxiv.org/abs/2209.14344

[7] Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip H. S. Torr, Mingfei Sun, and Shimon Whiteson. 2020. Is Independent Learning All You Need in the StarCraft Multi-Agent Challenge?

[8] European Comission. 2021. Artificial Intelligence Act. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:52021PC0206. Accessed: 2024-01-22.

[9] Iason Gabriel. 2020. Artificial Intelligence, Values, and Alignment. *Minds and Machines* 30 (09 2020), 411–437. https://doi.org/10.1007/s11023-020-09539-2

[10] Zoltán Gábor, Zsolt Kalmár, and Csaba Szepesvári. 1998. Multi-criteria reinforcement learning.. In *ICML*, Vol. 98. 197–205. https://sites.ualberta.ca/~szepesva/papers/multi98.ps.pdf

[11] Edward Hughes, Joel Z Leibo, Matthew Phillips, Karl Tuyls, Edgar Dueñez-Guzman, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin McKee, Raphael Koster, et al. 2018. Inequity aversion improves cooperation in intertemporal social dilemmas. *Advances in neural information processing systems* 31 (2018).

[12] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. 2021. Highly accurate protein structure prediction with AlphaFold. *Nature* 596, 7873 (2021), 583–589.

[13] Maxim Lapan. 2018. *Deep Reinforcement Learning Hands-On: Apply modern RL methods, with deep Q-networks, value iteration, policy gradients, TRPO, AlphaGo Zero and more*. Packt Publishing Ltd.

[14] Joel Z. Leibo, Vinícius Flores Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. 2017. Multi-agent Reinforcement Learning in Sequential Social Dilemmas. *CoRR* abs/1702.03037 (2017). http://arxiv.org/abs/1702.03037

[15] Anthony Lin. 2019. Binary search algorithm. *WikiJournal of Science* 2, 1 (2019), 1–13.

[16] Ritesh Noothigattu, Djallel Bouneffouf, Nicholas Mattei, Rachita Chandra, Piyush Madan, Ramazon Kush, Murray Campbell, Moninder Singh, and Francesca Rossi. 2019. Teaching AI Agents Ethical Values Using Reinforcement Learning and Policy Orchestration. *IBM Journal of Research and Development* PP (09 2019), 6377–6381. https://doi.org/10.1147/JRD.2019.2940428

[17] Christos H Papadimitriou and Tim Roughgarden. 2005. Computing equilibria in multi-player games.. In *SODA*, Vol. 5. 82–91.

[18] Georgios Papoudakis, Filippos Christianos, Lukas Schäfer, and Stefano V. Albrecht. 2021. Benchmarking Multi-Agent Deep Reinforcement Learning Algorithms in Cooperative Tasks. http://arxiv.org/abs/2006.07869

[19] Georgios Papoudakis, Filippos Christianos, Lukas Schäfer, and Stefano V. Albrecht. 2021. Benchmarking Multi-Agent Deep Reinforcement Learning Algorithms in Cooperative Tasks. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS)*. http://arxiv.org/abs/2006.07869

[20] Mark O. Riedl and B. Harrison. 2016. Using Stories to Teach Human Values to Artificial Agents. In *AAAI Workshop: AI, Ethics, and Society*.

[21] Manel Rodriguez-Soto, Maite Lopez-Sanchez, and Juan A Rodriguez-Aguilar. 2020. A structural solution to sequential moral dilemmas. In *Proceedings of the 19th international conference on autonomous agents and multiagent systems*. 1152–1160.

[22] Manel Rodriguez-Soto, Maite Lopez-Sanchez, and Juan A Rodriguez-Aguilar. 2021. Multi-Objective Reinforcement Learning for Designing Ethical Environments.. In *IJCAI*. 545–551.

[23] Manel Rodriguez-Soto, Maite Lopez-Sanchez, and Juan A Rodriguez-Aguilar. 2023. Multi-objective reinforcement learning for designing ethical multi-agent environments. *Neural Computing and Applications* (2023), 1–26.

[24] Diederik M Roijers. 2016. Multi-objective decision-theoretic planning. *AI Matters* 2, 4 (2016), 11–12.

[25] Diederik M. Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. 2013. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research* 48 (2013), 67–113.

[26] Francesca Rossi and Nicholas Mattei. 2019. Building Ethically Bounded AI. *Proceedings of the AAAI Conference on Artificial Intelligence* 33 (07 2019), 9785–9789. https://doi.org/10.1609/aaai.v33i01.33019785

[27] Stuart Russell, Daniel Dewey, and Max Tegmark. 2015. Research priorities for robust and beneficial artificial intelligence. *Ai Magazine* 36, 4 (2015), 105–114.

[28] Roxana Rădulescu, Patrick Mannion, Diederik M. Roijers, and Ann Nowé. 2020. Multi-objective multi-agent decision making: a utility-based analysis and survey. *Autonomous Agents and Multi-Agent Systems* 34, 1 (April 2020). https://doi.org/10.1007/s10458-019-09433-x

[29] Joar Skalse, Lewis Hammond, Charlie Griffin, and Alessandro Abate. 2022. Lexicographic Multi-Objective Reinforcement Learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, Vienna, Austria, 3430–3436. https://doi.org/10.24963/ijcai.2022/476

[30] Nate Soares and Benja Fallenstein. 2014. Aligning superintelligence with human interests: A technical research agenda. *Machine Intelligence Research Institute (MIRI) technical report* 8 (2014).

[31] Richard Sutton. 2019. The bitter lesson. *Incomplete Ideas (blog)* 13, 1 (2019), 38.

[32] Elizaveta Tennant, Stephen Hailes, Mirco Musolesi, et al. 2023. Modeling moral choices in social dilemmas with multi-agent reinforcement learning. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*. 317–325.

[33] Alperen Tercan and Vinayak S. Prabhu. 2024. Thresholded Lexicographic Ordered Multiobjective Reinforcement Learning. http://arxiv.org/abs/2408.13493 arXiv:2408.13493 [cs].

[34] Ibo Van de Poel and Lambèr Royakkers. 2023. *Ethics, technology, and engineering: An introduction*. John Wiley & Sons. https://books.google.com/books?hl=en&lr=&id=SYq4EAAAQBAJ&oi=fnd&pg=PR10&dq=Ethics,+technology,+and+engineering:+an+introduction.&ots=O46iJck9zx&sig=ZDX7DwrsWkNNBWMe0YlSgq1v2hY

[35] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (2019), 350–354.

[36] Bernhard Von Stengel. 2002. Computing equilibria for two-person games. *Handbook of game theory with economic applications* 3 (2002), 1723–1759.

[37] Yueh-Hua Wu and Shou-De Lin. 2018. A low-cost ethics shaping approach for designing reinforcement learning agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[38] Peter R Wurman, Samuel Barrett, Kenta Kawamoto, James MacGlashan, Kaushik Subramanian, Thomas J Walsh, Roberto Capobianco, Alisa Devlic, Franziska Eckert, Florian Fuchs, et al. 2022. Outracing champion Gran Turismo drivers with deep reinforcement learning. *Nature* 602, 7896 (2022), 223–228.

[39] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. 2022. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in Neural Information Processing Systems* 35 (2022), 24611–24624.

[40] Han Yu, Zhiqi Shen, Chunyan Miao, Cyril Leung, Victor R. Lesser, and Qiang Yang. 2018. Building Ethics into Artificial Intelligence. In *IJCAI*. 5527–5533.