Guaranteeing the Learning of Ethical Behaviour through Multi-Objective Reinforcement Learning*

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

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

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

ABSTRACT

AI research is being challenged with ensuring that autonomous agents behave ethically, namely in alignment with moral values. A common approach, founded on the exploitation of Reinforcement Learning techniques, is to design environments that incentivise agents to learn an ethical behaviour. However, to the best of our knowledge, current approaches do not offer theoretical guarantees that an agent will learn an ethical behaviour. Here, we advance along this direction by proposing a novel way of designing environments wherein it is formally guaranteed that an agent learns to behave ethically while pursuing its individual objective. Our theoretical results develop within the formal framework of Multi-Objective Reinforcement Learning to ease the handling of an agent's individual and ethical objectives. As a further contribution, we leverage on our theoretical results to introduce an algorithm that automates the design of ethical environments.

KEYWORDS

Value Alignment, Moral Decision Making, Multi-Objective Reinforcement Learning

1 INTRODUCTION

As artificial agents become more intelligent and pervade our societies, it is key to guarantee that situated agents act *value-aligned*, that is, in alignment with human values [23, 24]. Otherwise, we are prone to potential ethical risk in critical areas as diverse as elder caring [5], personal services [31], and automated driving [16]. As a consequence, there has been a growing interest in the Machine Ethics [22, 32] and AI Safety [2, 15] communities in the use of Reinforcement Learning (RL) [25] to deal with the urging problem of *value alignment*.

Among these two communities, it is common to find proposals to tackle the value alignment problem by designing an environment that incentivises ethical behaviours (or penalises unethical ones) by means of some exogenous reward function (e.g., [1, 4, 17–19, 30]). We observe that this approach consists in a two-step process: first, the ethical knowledge is encoded as rewards (*reward specification*); and then, these rewards are incorporated into the agent's learning environment (*ethical embedding*).

The literature is populated with embedding solutions that use a linear scalarisation function for *weighting* the agent's individual reward with the ethical reward (e.g. [19, 30]). However, to the best of our knowledge, there are no studies following the linear scalarisation approach that offer theoretical guarantees regarding the learning of ethical behaviours. Furthermore, [27] point out some shortages regarding the adoption of a linear ethical embedding: the agent's learnt behaviour will be heavily influenced by the relative scale of the individual rewards. This issue is specially relevant when the ethical objective must be wholly fulfilled (e.g., a robot in charge of buying an object should never decide to steal it [3]). For those cases, we argue that the embedding must be done in such a way that ethical behaviour is prioritised, providing theoretical guarantees for the learning of ethical policies.

Against this background, the objective of this work is twofold: (1) to offer theoretical guarantees for the linear embedding approach so that we can create an *ethical environment*, that is, an environment wherein it is ensured that an agent learns to behave ethically while pursuing its individual objective; and (2) to automate the design of such ethical environment. We address such goals within our view of ethical environment design process, as outlined in Figure 1. According to such view, a reward specification task combines the individual and ethical objectives to yield a multi-objective environment. Thereafter, an ethical embedding task transforms the multi-objective environment into an ethical environment, which is the one wherein an agent learns. Within the framework of such ethical environment design process, we address the goals above, focusing on the ethical embedding task, to make the following novel contributions.

Firstly, we characterise the policies that we want an agent to learn, the so-called *ethical policies*: those that prioritise ethical objectives over individual objectives. Thereafter, we propose a particular ethical embedding approach, and formally prove that the resulting learning environment that it yields is ethical. This means that we guarantee that an agent will always learn ethical policies when interacting in such environment. Our theoretical results are based on the formalisation of the ethical embedding process within the framework of Multi-Objective Reinforcement Learning (MORL)[20], which provides Multi-objective MDPs (MOMDPs) to handle both individual and ethical objectives. Thus, MOMDPs provide the model for the multi-objective environment that results from reward specification (Figure 1).

Secondly, based on our theoretical results, we propose an algorithm to implement our ethical embedding. This novel algorithm tailors current developments in the MORL literature to build an ethical environment as a single-objective MDP from the multi-objective MDP that stems from the reward specification process. Since the resulting single-objective MDP encapsulates the ethical rewards,

*Research supported by projects AI4EU (H2020-825619), LOGISTAR (H2020-769142), COREDEM (H2020-785907), Crowd4SDG (H2020-872944), CI-SUSTAIN (PID2019-104156GB-I00), COMRIDI18-1-0010-02, MISMIS PGC2018-096212B-C33, TAILOR (H2020-952215), 2017 SGR 172 and 2017 SGR 341. Manel Rodriguez-Soto was funded by the Spanish Government with an FPU grant (ref. FPU18/03387).

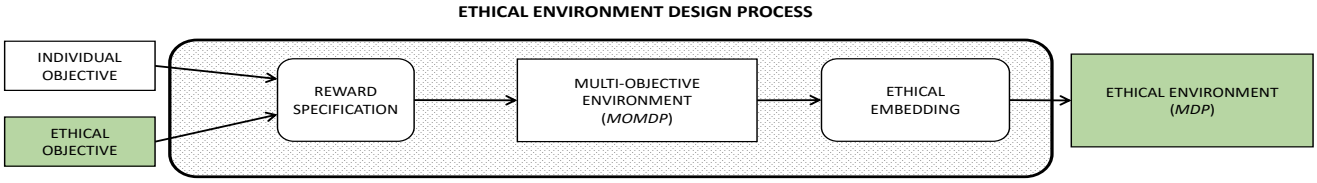

Figure 1: The process of designing an ethical environment is performed in two steps: a reward specification and an ethical embedding. Our algorithm computes the latter. Rectangles stand for objects whereas rounded rectangles correspond to processes.

the agent can thus apply a basic RL method to learn its optimal policy there. Specifically, we ground ethical embedding algorithm on the computation of convex hulls (as described in [6]) as the means to find ethical policies.

To summarise, in this paper we make headway in building ethical environments by providing two main novel contributions: (i) the theoretical means to design the learning environment so that an agent's ethical learning is guaranteed; and (ii) algorithmic tools for automating the configuration of the learning environment.

In what follows, Section 2 presents some necessary background on MORL. Then, Section 3 presents our formalisation of the ethical embedding problem that we must solve to create an ethical environment. Next, Section 4 studies how to guarantee the learning of ethical policies in ethical environments, and Section 5 introduces our algorithm to build ethical environments. Subsequently, Section 6 illustrates our proposal by means of a simple example, the public civility game. Finally, Section 7 concludes and sets paths to future work.

2 BACKGROUND

This section is devoted to present the necessary background and related work in both single-objective reinforcement learning and multi-objective reinforcement learning.

2.1 Single-objective reinforcement learning

In single-objective reinforcement learning (RL), the environment is characterised as a *Markov decision process* (MDP) [7, 14, 25]. An MDP characterises an environment in which an agent is capable of repeatedly acting upon it to modify it, and immediately receive a reward signal after each action. Formally:

Definition 1 (Markov Decision Process). *A (finite single-objective)*[1] *Markov Decision Process (MDP) is defined as a tuple* $\langle \mathcal{S}, \mathcal{A}, R, T \rangle$ *where* $\mathcal{S}$ *is a (finite) set of states,* $\mathcal{A}(s)$ *is the set of actions available at state* s*,* $R(s, a, s')$ *is a reward function specifying the expected reward for each tuple of state* s*, action* a *and future state* s'*, and* $T(s, a, s')$ *is a transition function specifying the probability that the next state is* s' *if an action* a *is performed upon the state* s*.*

An agent's behaviour in a MDP is characterised by means of a *policy* π which, for each state-action pair $\langle s, a \rangle$, specifies the probability of performing action a upon state s.

The classical method to evaluate a policy is to compute the (expected) discounted sum of rewards that an agent obtains by following it. This operation is formalised by means the so-called *value function* V, defined as:

$$V^{\pi}(s) \doteq \mathbb{E}[\sum_{t=0}^{\infty} \gamma^k r_{t+k+1} \mid S_t = s, \pi] \text{ for every state } s \in \mathcal{S}, \quad (1)$$

where $\gamma \in [0, 1)$ is referred to as the discount factor.

The solution of an MDP is a policy that maximises the value function $\pi_* \doteq \arg\max_{\pi} V^{\pi}$. Such policy is the agent's learning goal in an MDP. We refer to π_* as an *optimal* policy. We call the value function V^{π_*} of an optimal policy π_* simply as the *optimal value function* V^*. While there might be several optimal policies in an MDP, all of them share the same optimal value function [25].

Notice that these optimal policies and optimal value function exist because there is a total order between policies. In other words, we can always determinate whether $V^{\pi} > V^{\pi'}$ or $V^{\pi} < V^{\pi'}$ or $V^{\pi} = V^{\pi'}$ for any pair of policies π, π'.

In the reinforcement learning literature, Q-learning is a classical algorithm for learning an optimal policy [29].

2.2 Multi-objective reinforcement learning

Multi-objective reinforcement learning (MORL) formalises problems in which an agent has to ponder between several objectives, each represented as an independent reward function [20]. Hence, in MORL, the environment is characterised as a *Multi-Objective Markov Decision Process* (MOMDP), an MDP composed of a vectorial reward functions. Formally:

Definition 2. *An* n*-objective Markov Decision Process (MOMDP) is defined as a tuple* $\langle \mathcal{S}, \mathcal{A}, \vec{R}, T \rangle$ *where* $\mathcal{S}$*,* $\mathcal{A}$ *and* $\mathcal{T}$ *are the same as in an MDP, and* $\vec{R} = (R_1, \ldots, R_n)$ *is a* vectorial *reward function with each* R_i *as the associated scalar reward function to objective* $i \in \{1, \ldots, n\}$*.*

Policies in an MOMDP are evaluated by means of a *vectorial* value function $\vec{V}$ (or simply *value vector*), defined as:

$$\vec{V}^{\pi}(s) \doteq \mathbb{E}[\sum_{t=0}^{\infty} \gamma^k \vec{r}_{t+k+1} \mid S_t = s, \pi] \text{ for every state } s \in \mathcal{S}. \quad (2)$$

In an MOMDP, it is not straightforward to define its solution, unlike in an MDP, because the value vector $\vec{V}$ only offers a partial order between policies and not a total order. For example, a policy π can be better than another policy π' for some objective ($V_i^{\pi} > V_i^{\pi'}$) while at the same time being worse for another objective ($V_j^{\pi} < V_j^{\pi}$).

[1] Thorough the paper we refer to a finite single-objective MDP simply as an MDP.

Thus, it is not possible to determine optimal policies in an MOMDP without additional environment knowledge. If this additional knowledge is specific enough, we can create an alternative definition of optimality for the policies an MOMDP. Otherwise, we might be only capable of determining a subset of policies uncomparable between them but better than the rest.

Therefore, depending on this additional knowledge, an MOMDP can have two different kinds of solution: those for which the goal is to learn a *single policy*, and those in which the goal is to learn *multiple policies* [26]. Now we proceed to explain both problems.

2.2.1 Single-policy MORL. Most approaches in MORL assume the existence of a *scalarisation function* f capable of reducing the dimensionality of an MOMDP into a single one. Such scalarisation function transforms the vectorial value function $\vec{V}$ into a scalar value function $f(\vec{V})$. With f, the agent's goal becomes to learn a policy that maximises $f(\vec{V})$, a single-objective problem.

It is specially notable the particular case in which f is linear, because in such case the scalarised problem can be solved with single-objective reinforcement learning algorithms[2]. Any linear scalarisation function f is a weighted combination of rewards, and henceforth we will refer to such function by the weight vector $\vec{w} \in \mathbb{R}^n$ that it employs. We refer to any policy that maximises $f(\vec{V}) = \vec{w} \cdot \vec{V}$ as $\vec{w}$-optimal. Any $\vec{w}$-optimal policy is thus optimal in the associated MDP $\langle \mathcal{S}, \mathcal{A}, \vec{w} \cdot \vec{R}, T \rangle$.

2.2.2 Multiple-policy MORL. If the scalarisation function is not assumed to exist or to be even *a priori* known, it is not possible to define an optimality criterion. In such cases, what we can do is to compute the set of policies that *could* be optimal for some hypothetical scalarisation function. We refer to such policies as *undominated* policies.

In this paper, we are interested in the particular case were we just consider linear scalarisation functions. In such case, the undominated set is called the *convex hull* [21]:

Definition 3 (Convex hull). *Given an MOMDP $\mathcal{M}$, its convex hull CH is the subset of policies π_* and their associated value vectors $\vec{V}^{\pi_*}$ that are maximal for some weight vector $\vec{w}$:*

$$CH(\mathcal{M}) \doteq \{\vec{V}^{\pi_*} \mid \pi_* \in \Pi^{\mathcal{M}} \wedge \exists \vec{w} \in \mathbb{R}^n : \vec{w} \cdot \vec{V}^{\pi_*} = \max_{\pi \in \Pi^{\mathcal{M}}} \vec{w} \cdot \vec{V}^{\pi}\}, \tag{3}$$

where $\Pi^{\mathcal{M}}$ is the set of policies of $\mathcal{M}$, and n is the number of objectives of $\mathcal{M}$.

In the multi-objective reinforcement learning literature, Convex Hull Value Iteration (CHVI) [6] is a classical algorithm for obtaining the convex hull of an MOMDP. CHVI can be applied when considering only the linear scalarisation functions f of an MOMDP.

3 FORMALISING THE ETHICAL EMBEDDING PROBLEM

In this section we propose a formalisation of the *ethical embedding* of value alignment problems in which an ethical objective must be fulfilled and an individual objective is pursued. Our main goal is to guarantee that an agent will learn to behave ethically, that is, to behave in alignment with a moral value. In the Ethics literature, moral values (also called ethical principles) express the moral objectives worth striving for [28].

As mentioned above, the value alignment problem can be divided in two steps: the *reward specification* (to transform ethical knowledge into ethical rewards) and the *ethical embedding* (to ensure that these rewards incentivise the agent to be ethical). Although both are critical problems in the Machine Ethics and AI Safety community, in this paper we focus on the ethical embedding problem, and likewise we assume that we already have a reward specification in the form of a Multi-Objective Markov Decision Processes (MOMDP) [20]. This way we can handle an ethical objective and an agent's individual objective within the same learning framework. Precisely, MOMDPs formalise sequential decision making problems in which we need to ponder several objectives.

Thus, we define an *ethical MOMDP* as an MOMDP encoding the reward specification of a value alignment problem in which the agent must consider both its individual objective and an ethical objective. The first component in the corresponding vectorial reward function characterises the individual agent's objective (as usually done in RL), whereas the subsequent components represent the ethical objective [13]. Following the Ethics literature [8, 11, 12, 28], we define an ethical objective through two equally-important dimensions: (i) a *normative dimension*, which punishes the violation of normative requirements; and (ii) an *evaluative dimension*, which rewards morally praiseworthy actions. Formally:

Definition 4 (Ethical MOMDP). *Given a MOMDP*

$$\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_{\mathcal{N}} + R_E), T \rangle, \tag{4}$$

where R_0 corresponds to the reward associated to the individual objective, we say that $\mathcal{M}$ is an ethical MOMDP *if and only if:*

- *$R_{\mathcal{N}} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^-$ is a* normative *reward function penalising the violation of normative requirements; and*
- *$R_E : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^+$ is an* evaluative *reward function that (positively) rewards the performance of actions evaluated as praiseworthy.*

Dividing the ethical reward function in two parts allows us to avoid the ethical problem of an agent learning to maximise its accumulation of praiseworthy actions while disregarding some of its normative requirements.

In the ethical embedding, we transform an ethical MOMDP into a single-objective MDP (in which the agent will learn its policy) by means of scalarisation function f_e, which we call the *embedding function*. In the particular case that f_e is linear, we say that we are applying a linear embedding or a *weighting*.

Ethical MOMDPs pave the way to characterise our notion of ethical policy: an *ethical policy* is a policy that abides to all the norms while also behaving as praiseworthy as possible. In other words, it is a policy that adheres to the specification of the ethical objective. We capture this notion by means of the normative and evaluative components of the value function in an ethical MOMDP:

Definition 5 (Ethical policy). *Let $\mathcal{M}$ be an ethical MOMDP. We say that a policy π_* is an ethical policy in $\mathcal{M}$ if and only if its*

[2]Because the linear scalarisation function for $\vec{V}$ also induces a scalarisation function for $\vec{R}$, by setting $\vec{w} \cdot \vec{V} = \vec{w} \cdot \mathbb{E}[\sum_{t=0}^{\infty} \gamma^k \vec{r}_{t+k+1}] = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^k \vec{w} \cdot \vec{r}_{t+k+1}]$, which is usually not true in the non-linear case.

value vector $\vec{V}^{\pi_*} = (V_0^{\pi_*}, V_{\mathcal{N}}^{\pi_*}, V_E^{\pi_*})$ *is optimal for its ethical objective (i.e., both its normative* $V_{\mathcal{N}}$ *and evaluative* V_E *components):*

$$V_{\mathcal{N}}^{\pi_*} = \max_{\pi} V_{\mathcal{N}}^{\pi},$$
$$V_E^{\pi_*} = \max_{\pi} V_E^{\pi}.$$

For the sake of simplicity, we refer to a policy that is not ethical in the sense of Definition 5 as an *unethical* policy.

With ethical policies, we can now define formally *ethical-optimal* policies: the policies that we want an agent to learn. Ethical-optimal policies correspond to those policies in which the individual objective is pursued subject to the ethical objective being fulfilled. Specifically, we say that a policy is *ethical-optimal* if and only if it is ethical and it also maximises the individual objective V_0 (i.e., the accumulation of rewards R_0). Formally:

Definition 6 (Ethical-optimal policy). *Given an MOMDP* $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_{\mathcal{N}} + R_E), T \rangle$*, a policy* π_* *is* ethical-optimal *in M if and only if it is maximal among the set* Π_e *of ethical policies:*

$$V_0^{\pi_*} = \max_{\pi \in \Pi_e} V_0^{\pi}.$$

Notice that while there can be several ethical-optimal policies in an ethical MOMDP, all of them will share the same value vector. We refer to such value vector as the *ethical-optimal* value vector $\vec{V}^*$.

Given an MOMDP encoding individual and ethical rewards, our aim is to find an embedding function that guarantees that it is only possible for an agent to learn ethical-optimal policies over the scalarised MOMDP (as a single-objective MDP). Thus, we must design an embedding function that scalarises the rewards received by the agent in such a way that ensures that ethical-optimal policies are optimal for the agent. In its simplest form, this embedding function will have the form of a linear combination of individual and ethical objectives

$$f(\vec{V}^{\pi}) = \vec{w} \cdot \vec{V}^{\pi} = w_0 V_0^{\pi} + w_e (V_{\mathcal{N}}^{\pi} + V_E^{\pi}) \tag{5}$$

where $\vec{w} = (w_0, w_e)$ is a weight vector with weights $w_0, w_e > 0$ to guarantee that the agent is taking into account all rewards (i.e., both objectives). We will be referring thus to w_0 as the individual weight and w_e as the *ethical weight*. Without loss of generality, we fix the individual weight to $w_0 = 1$.

Therefore, we can formalise the ethical embedding problem as that of computing a weight vector $\vec{w}$ that incentivises an agent to behave ethically while still pursuing its individual objective.Formally:

Problem 1 (Ethical embedding). *Let* $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_{\mathcal{N}} + R_E), T \rangle$ *be an ethical MOMDP. Compute the weight vector* $\vec{w}$ *with positive weights such that all optimal policies in the MDP* $\mathcal{M}' = \langle \mathcal{S}, \mathcal{A}, w_0 R_0 + w_e (R_{\mathcal{N}} + R_E), T \rangle$ *are also ethical-optimal in* $\mathcal{M}$ *(as defined in Def. 6).*

A weight vector $\vec{w}$ with positive weights that guarantees that all optimal policies (with respect to $\vec{w}$) are also ethical-optimal is a solution of Problem 1. Moreover, we aim at finding solutions $\vec{w}$ that are as little intrusive with the agent's learning process as possible (i.e., the $\vec{w}$ that guarantees the learning of an ethical policy with the minimal ethical weight w_e). The next section proves that there always exist a solution to the ethical embedding problem for any ethical MOMDP.

4 SOLVABILITY OF THE ETHICAL EMBEDDING PROBLEM

This section is devoted to describe the minimal conditions under which there always exists a solution to Problem 1, and to prove that such solution actually exists. This solution (a weight vector) will allow us to apply the ethical embedding process to produce an ethical environment (a single-objective MDP) wherein an agent learns to behave ethically (i.e., an ethical-optimal policy).

For all the following theoretical results, we assume the following condition for any ethical MOMDP: if we want the agent to behave ethically, it must be actually possible for it to behave ethically[3]. Formally:

Condition 1 (Ethical policy existence). *Given an ethical MOMDP, there is at least one ethical policy (as defined by Def. 5).*

If Condition 1 holds, next Theorem guarantees that Problem 1 is always solvable, or in other words, that it is always possible to guarantee that the learnt behaviour of an agent will be ethical if we give a reward incentive that is large enough.

Theorem 1 (Solution existence). *Given an ethical MOMDP* $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_{\mathcal{N}} + R_E), T \rangle$ *for which Condition 1 is satisfied, there exists a weight vector* $\vec{w} = (1, w_e)$ *with* $w_e > 0$ *for which every optimal policy in the MDP* $\mathcal{M}' = \langle \mathcal{S}, \mathcal{A}, w_0 R_0 + w_e (R_{\mathcal{N}} + R_E), T \rangle$ *is also ethical-optimal in* $\mathcal{M}$*.*

Proof. Without loss of generality we only consider deterministic policies, by the Policy Improvement Theorem (see [25] for more details).

Consider a weight vector $\vec{w} = (1, w_e)$ with $w_e \geq 0$. Suppose that for that weight vector, the only deterministic $\vec{w}$-optimal policies are ethical policies. Then we have finished.

Suppose that it is not the case, and there is some $\vec{w}$-optimal policy ρ that is not ethical. This implies that for some state s':

$$V_{\mathcal{N}}^{\rho}(s') + V_E^{\rho}(s') < V_{\mathcal{N}}^{*}(s') + V_E^{*}(s').$$

For an $\epsilon > 0$ large enough and for the weight vector $\vec{w}' = (1, w_e + \epsilon)$, any ethical policy π will have a better value vector at that state s' than ρ:

$$\vec{w}' \cdot \vec{V}^{\rho}(s') < \vec{w}' \cdot \vec{V}^{\pi}(s').$$

Therefore, ρ will not be an $\vec{w}'$-optimal policy. Notice that ρ will remain being $\vec{w}'$-suboptimal even if we increase again the value of w_e by defining $\vec{w}'' = (1, w_e + \epsilon + \delta)$ with $\delta > 0$ as large as we wish.

It follows that if we choose ρ to be the unethical policy that requires the maximum increase of ϵ (we know that this maximum exists since there is a finite number of deterministic policies in a finite MOMDP), after increasing w_e so it is not an optimal policy, then no unethical policy can be $\vec{w}$-optimal for the new w_e. Therefore, by elimination every $\vec{w}$-optimal policy is also ethical for this new weight vector.

To finish, notice now that if ethical policies π exist (due to Condition 1), so does at least one ethical-optimal policy π_* that maximises

[3] In the Ethics literature this condition is summarised with the expression *Ought implies can* [10].

V_0 among them. And for any weight vector $\vec{w} = (1, w_e)$ with $w_e > 0$, the scalarised value of π^* will be greater or equal than the one of any other ethical policy π. Therefore, only the ethical-optimal policies π_* (among the ethical policies) will be $\vec{w}$-optimal policies and thus optimal policies in $\mathcal{M}'$. □

In particular, we aim at knowing the *minimal* w_e for which $(1, w_e)$ is a solution of Problem 1. In other words, the minimal ethical weight w_e for which $\vec{V}^*$ is the only optimal policy for $(1, w_e)$. Every time we refer to the *minimal* ethical weight we do it in such sense.

5 SOLVING THE ETHICAL EMBEDDING PROBLEM

This section explains how to compute a solution weight vector $\vec{w}$ for the ethical embedding problem (Problem 1). Such weight vector $\vec{w}$ allows us to combine individual and ethical rewards into a single reward to create an ethical environment in which the agent learns an ethical behaviour, that is, an ethical-optimal policy. For that, next we detail an algorithm to solve the ethical embedding problem, the so-called *Ethical Embedding* algorithm.

Before delving into details, we outline and illustrate, with the aid of the Figure 2, the steps involved in computing a solution for the embedding problem. First step focuses on obtaining a particular subset P of the convex hull CH of the ethical MOMDP. This subset P must contain the ethical-optimal value vector $\vec{V}^*$. Figure 2 (Left) shows an example of $P \subseteq CH$ where black-rounded points constitute the partial convex hull (P) while grey points are values of policies never maximal for any weight.

Figure 2 (Centre) highlights in green $\vec{V}^*$, which accumulates the greatest ethical value (Y axis). This ethical-optimal value vector $\vec{V}^*$ will serve as a reference value vector to find the minimal weight vector $\vec{w} = (1, w_e)$ that solves Problem 1. For such weight vector, $\vec{w} \cdot \vec{V}^*$ is maximal (and the only maximal one) among all value vectors of P.

Figure 2 (Right) plots how the scalarised values of the points in the partial convex hull P (Figure 2 (Left)) change as the ethical weight increases. Notice that the ethical-optimal value vector becomes the only maximal value vector for any $w_e > 0.7$, indicated by the green vertical line.

Computing the minimal ethical weight does not require to consider all value vectors on the partial convex hull. In fact, it suffices to consider the so-called *second-best* value vector (highlighted in yellow in Figure 2 (Centre)) to compute it. The second-best value vector accumulates the greatest amount of ethical value after the ethical-optimal one. As shown in Figure 2 (Right): immediately after the line representing the ethical-optimal value vector $\vec{V}^*$ intersects the second-best value vector, $\vec{V}^*$ becomes maximal. Such intersection point is the value of the minimal ethical weight w_e (see the green vertical line in Figure 2 (Right)).

To summarise, our algorithm computes the ethical embedding function $\vec{w} = (1, w_E)$ with the minimal ethical weight w_E in the following three steps :

(1) *Computation of the partial convex hull* (Figure 2 (Left)).
(2) *Extraction of the two value vectors with the greatest ethical values* (Figure 2 (Centre)).
(3) *Computation of the ethical embedding function* $(1, w_E)$ *with minimal* w_E (Figure 2 (Right)).

Subsequent subsections provide the theoretical grounds for computing each step of our algorithm. Then, Subsection 5.4 presents the algorithm as a whole.

5.1 Computation of the partial convex hull

Importantly, in order to obtain the embedding function $\vec{w} = (1, w_e)$ that solves our problem, we do not need to compute the whole convex hull CH. We know that CH contains the ethical-optimal value vector $\vec{V}^*$, necessary for obtaining $\vec{w}$. However, since the ethical-optimal value vector $\vec{V}^*$ is the same for all ethical-optimal policies, any subset of the convex hull, $P \subseteq CH$, containing at least one ethical-optimal policy will suffice. Theorem 2 below naturally characterises the minimal subset P of the convex hull that we must compute to find the ethical-optimal value vector. Formally:

Theorem 2. *Given an ethical MOMDP* $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_\mathcal{N} + R_E), T \rangle$ *in which Condition 1 is satisfied, let* $P \subseteq CH(\mathcal{M})$ *be the subset of the convex hull of* $\mathcal{M}$ *limited to weight vectors of the form* $\vec{w} = (1, w_e)$ *with* $w_e > 0$. *Then, there is a policy* $\pi_* \in P$ *such that its value is the ethical-optimal value vector* ($V^{\pi_*} = V^*$).

Proof. From Theorem 1, we know that at least one ethical-optimal policy is optimal for a weight vector $\vec{w}$ of the form $\vec{w} = (1, w_e)$ with $w_e > 0$, and thus such policy belongs to this partial region P of the convex hull $CH(\mathcal{M})$. Therefore, its associated value vector, $\vec{V}^*$, also belongs to P. □

Henceforth, when referring to the *partial convex hull*, we are referring to this particular subset P defined in Theorem 2. We compute this subset of the convex hull by adapting the Convex Hull Value Iteration algorithm in [6] to constrain its search space of weight vectors to be of the form $\vec{w} = (1, w_e)$ with $w_e > 0$ (thus reducing its computational cost).

5.2 Extraction of two value vectors

In order to know which value vector in P corresponds to an ethical-optimal policy, we have to find the one that maximises the ethical reward function $(V_\mathcal{N} + V_E)$ of the ethical MOMDP. Formally, to obtain the ethical-optimal value vector within P, we must compute:

$$\vec{V}^*(s) = \underset{(V_0, V_\mathcal{N}+V_E) \in P}{\arg\max} [V_\mathcal{N}(s) + V_E(s)] \text{ for every state } s. \quad (6)$$

The ethical-optimal value vector $\vec{V}^*$ is the only maximal one in P. By maximal, we mean that its scalarised value is strictly greater than any other scalarised value vector of P. In particular, in this subsection we will see that $\vec{V}^*$ is the only maximal value vector when its scalarised value is strictly greater than the second most ethical value vector $\vec{V'}^* \in P$. We refer to $\vec{V'}^*$ as the *second-best* value vector, which we define as follows:

$$\vec{V'}^* \doteq \underset{(V_0, V_\mathcal{N}+V_E) \in P \setminus \{V^*\}}{\arg\max} [V_\mathcal{N}(s) + V_E(s)] \text{ for every state } s. \quad (7)$$

Thus, the second-best value vector accumulates the greatest amount of ethical rewards in P if we disregard $\vec{V}^*$ (i.e., when considering $P \setminus \{V^*\}$).

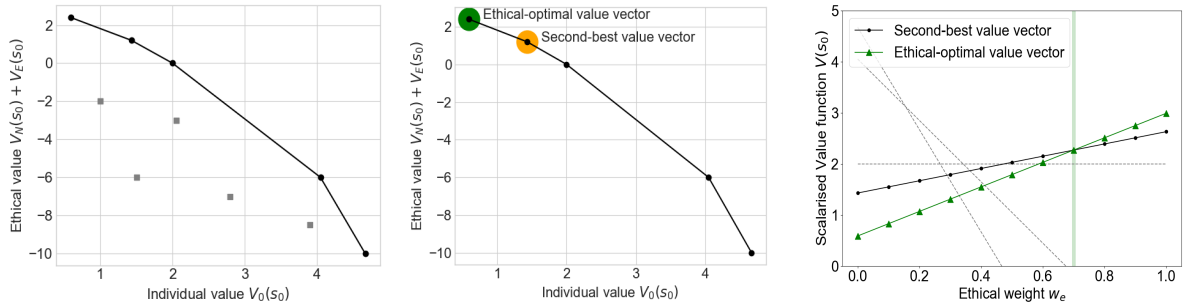

Figure 2: Left: Example of partial convex hull P, represented in objective space. Centre: Identification of the points of P corresponding with the ethical-optimal value vector $\vec{V}^*$ and the second-best value vector $\vec{V'}^*$. Right: Representation in weight space of P. The minimal weight value w_e for which $\vec{V}^*$ is optimal is identified with a green vertical line.

The following Theorem 3 proves that, indeed, we only need to compare $\vec{V}^*$ and $\vec{V'}^*$, and hence disregard the rest of value vectors in the convex hull, in order to find the minimal ethical weight w_e for which $\vec{V}^*$ is the only maximal value vector:

Theorem 3. *Given an ethical MOMDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_\mathcal{N} + R_E), T \rangle$ in which Condition 1 is satisfied, let $P \subseteq CH(\mathcal{M})$ be the subset of the convex hull of $\mathcal{M}$, limited to weight vectors of the form $\vec{w} = (1, w_e)$ with $w_e > 0$. Consider $\vec{V}^*$ the ethical-optimal policy, and $\vec{V'}^*$ the second-best value vector. If for a given weight vector $\vec{w} = (1, w_e)$ we have that $\vec{w} \cdot \vec{V}^* > \vec{w} \cdot \vec{V'}^*$, then $\vec{V}^*$ is $\vec{w}$-optimal, and the only $\vec{w}$-optimal policy of P.*

Proof. If $\vec{w} \cdot \vec{V}^* > \vec{w} \cdot \vec{V'}^*$, then $\vec{V'}^*$ is not maximal for $\vec{w}$ within P, which implies that some other value vector $\vec{V} \in P$ is. This value vector $\vec{V}$ needs to have more accumulation of ethical rewards than $\vec{V'}^*$, so the only possible candidate is the ethical-optimal policy $\vec{V}^*$. Hence, $\vec{V}^*$ is the only $\vec{w}$-optimal policy of P. □

Thus, these two value vectors $\vec{V}^*$ and $\vec{V'}^*$ are all we need to compute the embedding function $\vec{w} = (1, w_e)$ with minimal ethical weight w_e. Notice that *the two value vectors can be found simultaneously while sorting the value vectors of P*. Furthermore, $\vec{V}_\mathcal{N}$ and $\vec{V}_E$ are already available for these two value vectors because they are both part of the partial convex hull P, which we computed in Subsection 5.1 through our adapted CHVI.

5.3 Computation of the embedding function with minimal ethical weight

In the last step of our algorithm, the computation of the *embedding function* (the weight vector), we use the two extracted value vectors $\vec{V}^*$ and $\vec{V'}^*$ to find the minimal solution weight vector $\vec{w} = (1, w_e)$ that guarantees that optimal policies are ethical-optimal. In other words, such weight vector $\vec{w}$ will create an ethical environment (a single-objective MDP) in which the agent will learn an ethical-optimal policy. As anticipated by Theorem 3, we need to find the minimal value for $w_e \in \vec{w}$ such that:

$$V_0^*(s) + w_e[V_\mathcal{N}^*(s) + V_E^*(s)] > V_0'(s) + w_e[V_\mathcal{N}'(s) + V_E'(s)], \quad (8)$$

for every state $s \in \mathcal{S}$, where $\vec{V}^* = (V_0^*, V_N^* + V_E^*)$ and $\vec{V'}^* = (V_0', V_N' + V_E')$. This process is illustrated in Figure 2 (Right).

Notice that in Eq. 8 the only unknown variable is w_e. This amounts to solving a system of $|\mathcal{S}_0|$ linear inequalities (here $\mathcal{S}_0 \subseteq \mathcal{S}$ is the set of initial states) with a single unknown variable.

5.4 An algorithm for designing ethical environments

At this point we now count on all the tools for solving Problem 1, and hence build an ethical environment where the learning of ethical policies is guaranteed. Algorithm 1 implements the ethical embedding outlined in Figure 1. The algorithm starts in line 2 by computing the partial convex hull $P \subseteq CH(\mathcal{M})$ of the input ethical MOMDP $\mathcal{M}$ (see Subsection 5.1); and then in line 3 it obtains the ethical-optimal value vector $\vec{V}^*$ and the second-best value vector $\vec{V'}^*$ out of those in the partial convex hull P (see Subsection 5.2). Thereafter, in line 4 our weighting process searches, comparing $\vec{V}^*$ and $\vec{V'}^*$, for an ethical weight w_e that satisfies Equation 8 (see Subsection 5.3). For the obtained weight vector $\vec{w} = (1, w_e)$, all optimal policies of the single-objective MDP $\mathcal{M}' = \langle \mathcal{S}, \mathcal{A}, w_0 R_0 + w_e(R_\mathcal{N} + R_E), T \rangle$ will be ethical. In other words, such weight vector will solve the ethical embedding problem (Problem 1). Finally, the algorithm returns the MDP $\mathcal{M}'$ in line 5.

Algorithm 1 Ethical Embedding

1: **function** Embedding(Ethical MOMDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_\mathcal{N} + R_E), T \rangle$)
2: Compute $P \subseteq CH(\mathcal{M})$ the partial convex hull of $\mathcal{M}$ for weight vectors $\vec{w} = (1, w_e)$ with $w_e > 0$.
3: Find $\vec{V}^*$ the ethical-optimal value vector, and $\vec{V'}^*$ the second-best value vector, within P by solving Eq. 6.
4: Find the minimal value for w_e that satisfies Eq. 8.
5: Return MDP $\mathcal{M}' = \langle \mathcal{S}, \mathcal{A}, R_0 + w_e(R_\mathcal{N} + R_E), T \rangle$.
6: **end function**

The computational cost of the algorithm mainly resides in computing the partial convex hull of an MOMDP. The Convex Hull Value Iteration algorithm requires $O(n \cdot log\, n)$ times what its single-objective Value Iteration counterpart [6, 9] requires, where n is the number of policies in the convex hull. In our case this number will be $n' \leq n$ since we are just allowing a particular form of weights,

as explained in previous subsections. Notice that after computing $P \subseteq CH$, solving Eq. 6 is a sorting operation because we already have calculated $\vec{V}$ for every $\vec{V} \in P$. Similarly, solving Eq. 8 requires to solve $|\mathcal{S}_0|$ inequalities and then sort them to find the ethical weight w_e.

6 EXAMPLE: THE PUBLIC CIVILITY GAME

This section illustrates our process of designing an ethical environment (Algorithm 1) with an example. We use a single-agent version of the *Public Civility Game* [19], a value alignment problem where an agent learns to behave according to the moral value of civility. Far from being realistic, the example at hand is simple enough to serve our illustrative purposes. Furthermore, it can be seen as an ethical adaptation of the *irreversible side effects* environment [15].

Figure 3 (Left) depicts the environment, wherein two agents (L and R) move from their initial positions to their respective goal destinations (GL and GR). Since the L agent finds garbage (small red square) blocking its way, it needs to learn how to handle the garbage civically while moving towards its goal GL. The civic (ethical) behaviour we expect agent L to learn is to push the garbage to any wastebasket (WL and WR) without throwing it to agent R. The R agent is endowed with a fixed behaviour for reaching its goal: R always moves forward except for the first time-step, when it may not advance with a 50% chance to induce some randomness.

6.1 Reward specification

The Public Civility Game represents an ethical embedding problem where civility is the moral value to embed in the environment. As such, we encode it as an ethical MOMDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, (R_0, R_\mathcal{N} + R_E), T \rangle$. Next, we describe the states and actions of the environment together with its reward functions.

The environment is represented as a grid of cells as Figure 3 (Left) shows. Thus, a **state** $s \in S$ is defined as a tuple $s = \langle cell^L, cell^R, cell^G \rangle$ where $cell^L$ and $cell^R$ correspond to the position (cell) of agents L and R respectively, and $cell^G$ corresponds to the position of the garbage obstacle. The *initial state* s_0 of the MOMDP is illustrated in Figure 3 (Left): both agents are in adjacent cells at the bottom, and the garbage is located immediately in front of the left agent.

The set of **actions** is $\mathcal{A} = \{mF, mR, mL, pF, pR, pL\}$, where m stands for **m**ovement, p for **p**ush, F=Forward, R=Right, and L=Left. Actions $m*$ (mF, mR, and mL) change the agent's position accordingly, and actions $p*$ (pF, pR, and pL) change the garbage's position ($s.cell^G$) whenever the garbage is in front of the agent.

The agent's individual objective and the ethical objective have been specified as follows.

On the one hand, the **agent's individual objective** is to reach its destination (GL) as fast as possible, thus

$$R_0(s, a, s') \doteq \begin{cases} 20 & \text{if } s'.cell^L \in \text{GL}, \\ -1 & \text{otherwise.} \end{cases} \tag{9}$$

where $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$. In this manner, R_0 encourages the agent to never stop until it reaches GL.

On the other hand, the **ethical objective** is to promote civility by means of:

Policy π	Value $\vec{V}^\pi(s_0)$	w_e ranges
Unethical	(4.67, -5+0)	[0.0, 0.52]
Regimented	(1.43, 0+1.2)	[.52, 0.7]
Ethical	(0.59, 0+2.4)	[0.7, ∞)

Table 1: Policies π within the partial convex hull of the Public Civility Game and their associated values $\vec{V}^\pi = (V_0^\pi, V_\mathcal{N}^\pi + V_E^\pi)$. Weight w_e ranges indicate the values of ethical weights for which each policy is optimal.

- An evaluative reward function R_E that rewards the agent positively when performing the praiseworthy action of pushing the garbage inside the wastebasket. Thus,

$$R_E(s, a, s') \doteq \begin{cases} 10 & \text{if } s'.cell^G \in \{\text{WL,WR}\} \text{ and } a \in p*, \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

- A normative reward function $R_\mathcal{N}$ that punishes the agent for not complying with the moral requirement of being respectful with other agents. Thus, agent L will be punished with a negative reward if it throws the garbage to agent R:

$$R_\mathcal{N}(s, a, s') \doteq \begin{cases} -10 & \text{if } s'.cell^G = s'.cell^R \text{ and } a \in p*, \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

6.2 Ethical embedding

We now apply Algorithm 1 to design an ethical environment for the Public Civility Game. In what follows, we detail the three processes involved in obtaining this new environment.

Partial convex hull computation: Considering the ethical MOMDP $\mathcal{M}$, we compute its partial convex hull $P \subseteq CH(\mathcal{M})$. Figure 3 (centre) depicts the resulting P for the initial state s_0. It is composed of 3 different policies named after the behaviour they encapsulate:

(1) An **U**nethical (uncivil) policy, in which the agent moves towards the goal and throws away the garbage without caring about any ethical implication.
(2) A **R**egimented policy, in which the agent complies with the norm of not throwing the garbage to the other agent.
(3) An **E**thical policy, in which the agent behaves civically as desired.

Table 1 provides the specific vectorial value $\vec{V}^\pi = (V_0^\pi, V_\mathcal{N}^\pi + V_E^\pi)$ of each policy π and the range of values of the ethical weight w_e for which each policy is optimal.

Extraction of the two value vectors with the greatest ethical value: In our case, the **E**thical policy π_e has associated the ethical-optimal value vector since it is the policy with greatest ethical value within the partial convex hull P. Indeed, π_e is the only policy that maximises both the normative and the evaluative components ($V_\mathcal{N}$ and V_E respectively). Thus, the ethical-optimal value vector is its value vector $\vec{V}^{\pi_e}$. Last row in Table 1 shows the value of π_e for the initial state s_0: $\vec{V}^{\pi_e}(s_0) = (V_0^{\pi_e}, V_\mathcal{N}^{\pi_e} + V_E^{\pi_e}) = (0.59, 0 + 2.4)$. Similarly, the second most ethical value vector in P corresponds to the value of the **R**egimented policy π_R, which has the value $\vec{V}^{\pi_R}(s_0) = (V_0^{\pi_R}, V_\mathcal{N}^{\pi_R} + V_E^{\pi_R}) = (1.43, 0 + 1.2)$ for the initial state s_0.

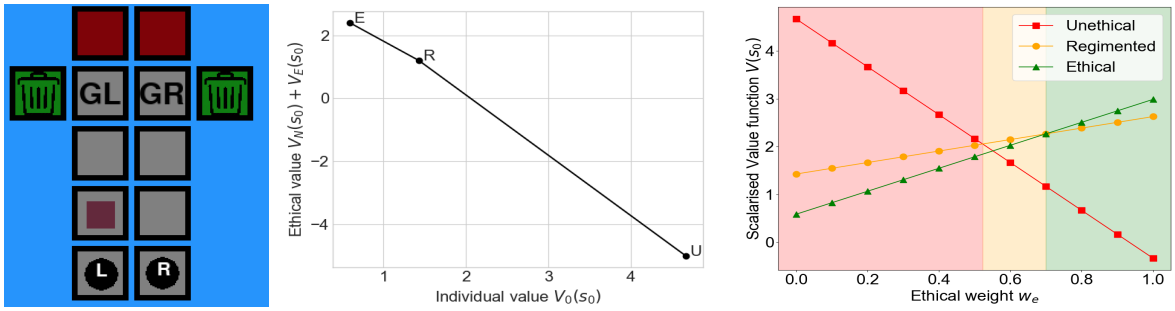

Figure 3: Left: Initial state of the public civility game. The agent on the left has to deal with the garbage obstacle, which has been located in front of it. Centre: Visualisation in Objective Space of the partial convex hull of $\mathcal{M}$ composed by 3 policies: E (Ethical), R (Regimented) and U (Unethical). Right: Visualisation in Weight Space of the partial convex hull of $\mathcal{M}$. Painted areas indicate which policy is optimal for the varying values of the ethical weight w_e.

Computation of the embedding function: Line 4 in Algorithm 1 computes the weight w_e in $\vec{w} = (1, w_e)$ for which π_e is the only optimal policy of P, by solving Eq. 8. This amounts to solve:

$$V_0^{\pi_e}(s_0) + w_e[V_{\mathcal{N}}^{\pi_e}(s_0) + V_E^{\pi_e}(s_0)] > V_0^{\pi_R}(s_0) + w_e[V_{\mathcal{N}}^{\pi_R}(s_0) + V_E^{\pi_R}(s_0)].$$

By solving it, we find that if $w_e > 0.7$, then the **E**thical policy becomes the only optimal one. We can check it (set $\epsilon > 0$):

$$0.59 + (0.7 + \epsilon) \cdot (0 + 2.4) = 2.27 + 2.4\epsilon > 1.43 + 0.7 \cdot (0 + 1.2).$$

Figure 3 (right) illustrates the scalarised value of the 3 policies for varying values of w_e in [0,1] (for w_e>1 tendencies do not change). Painted areas in the plot help to identify the optimal policies for specific intervals of w_e. Focusing on the green area, we can observe that the **E**thical policy becomes the only optimal one for $w_e > 0.7$.

Therefore, the last step in our algorithm returns an MDP $\mathcal{M}'$ whose reward comes from scalarising the MOMDP by $\vec{w} = (1, w_e)$, being w_e strictly greater than 0.7. Thus, adding any $\epsilon > 0$ will suffice. If, for instance, we set $\epsilon = 0.01$ then, the weight vector $(1, 0.7 + 0.01) = (1, 0.71)$ solves the Public Civility Game. More specifically, an MDP created from an embedding function with such ethical weight w_e incentivises the agent to learn the **E**thical (civic) policy.

6.3 Learning

After creating our ethical environment $\mathcal{M}' = \langle \mathcal{S}, \mathcal{A}, R_0 + w_e(R_{\mathcal{N}} + R_E), T \rangle$ (in our case with $w_e = 0.71$) we can confirm our theoretical results by letting the agent learn an optimal policy in $\mathcal{M}'$.

We provide the L agent with Q-learning [29] as its learning algorithm. In Q-learning, we need to specify two hyperparameters: the learning rate α and the discount factor γ. In our case, we set them to $\alpha = 0.8$ and $\gamma = 0.7$. Furthermore, we set the learning policy to be ϵ-greedy [25].

After letting the agent learn for 5000 iterations it actually learns to bring the garbage to the wastebasket while moving towards its goal, as it could not be otherwise. Figure 4 shows how the agent's value vector $\vec{V}(s_0)$ stabilises, with less than 1500 episodes, at 0.59 (V_0 line) and 2.4 ($V_{\mathcal{N}} + V_E$ line), which is precisely the value of the **E**thical policy.

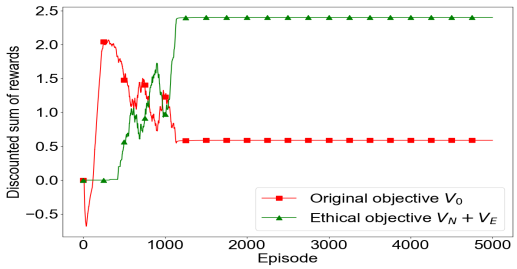

Figure 4: Evolution of the accumulated rewards per episode that the agent obtains in the ethical environment.

7 CONCLUSIONS AND FUTURE WORK

Designing ethical environments for learning agents is a challenging problem. We make headway in tackling this problem by providing novel formal and algorithmic tools that build upon Multi-Objective Reinforcement Learning. In particular, our problem consists in ensuring that the agent wholly fulfils its ethical objective while pursuing its individual objective.

MORL is a valuable framework to handle multiple objectives. In order to ensure ethical learning (value-alignment), we formalise –within the MORL framework– *ethical-optimal* policies as those that prioritise their ethical objective. Overall, we design an ethical environment by considering a two-step process that first specifies rewards and second performs an ethical embedding. We formalise this last step as the ethical embedding problem and theoretically prove that it is always solvable. Our findings lead to an algorithm for automating the design of an ethical environment. Our algorithm ensures that, in this ethical environment, it will be in the best interest of the agent to behave ethically while still pursuing its individual objectives. We illustrate it with a simple example that embeds the moral value of civility.

As to future work, we would like to further examine empirically our algorithm in more complex environments.

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] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Francis Christiano, John Schulman, and Dan Mané. 2016. Concrete Problems in AI Safety. *CoRR* abs/1606.06565 (2016).

[3] T. Arnold, Daniel Kasenberg, and Matthias Scheutz. 2017. Value Alignment or Misalignment - What Will Keep Systems Accountable?. In *AAAI Workshops*.

[4] 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

[5] Rosangela Barcaro, M. Mazzoleni, and P. Virgili. 2018. Ethics of care and robot caregivers. *Prolegomena* 17 (06 2018), 71–80. https://doi.org/10.26362/20180204

[6] Leon Barrett and Srini Narayanan. 2008. Learning all optimal policies with multiple criteria. *Proceedings of the 25th International Conference on Machine Learning* (01 2008), 41–47. https://doi.org/10.1145/1390156.1390162

[7] RICHARD BELLMAN. 1957. A Markovian Decision Process. *Journal of Mathematics and Mechanics* 6, 5 (1957), 679–684.

[8] R. M. Chisholm. 1963. Supererogation and Offence: A Conceptual Scheme for Ethics. *Ratio (Misc.)* 5, 1 (1963), 1.

[9] K. L. Clarkson. 1988. Applications of Random Sampling in Computational Geometry, II. In *Proceedings of the Fourth Annual Symposium on Computational Geometry* (Urbana-Champaign, Illinois, USA) *(SCG '88)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/73393.73394

[10] Brian Duignan. 2018. Ought implies can. https://www.britannica.com/topic/ought-implies-can. Accessed: 2021-01-15.

[11] Amitai Etzioni and Oren Etzioni. 2016. Designing AI Systems That Obey Our Laws and Values. *Commun. ACM* 59, 9 (Aug. 2016), 29–31. https://doi.org/10.1145/2955091

[12] William K. Frankena. 1973. *Ethics, 2nd edition.* Englewood Cliffs, N.J. : Prentice-Hall,.

[13] Terry Horgan and Mark Timmons. 2010. Untying a knot from the inside out: Reflections on the "paradox" of supererogation. *Social Philosophy and Policy* 27 (07 2010), 29 – 63. https://doi.org/10.1017/S026505250999015X

[14] Leslie Pack Kaelbling, Michael L. Littman, and Andrew W. Moore. 1996. Reinforcement Learning: A Survey. *J. Artif. Int. Res.* 4, 1 (May 1996), 237–285.

[15] Jan Leike, Miljan Martic, Viktoriya Krakovna, Pedro Ortega, Tom Everitt, Andrew Lefrancq, Laurent Orseau, and Shane Legg. 2017. AI Safety Gridworlds. *aRXiV 1711.09883* (11 2017).

[16] Patrick Lin. 2015. *Why Ethics Matters for Autonomous Cars.* Springer Berlin Heidelberg, Berlin, Heidelberg, 69–85. https://doi.org/10.1007/978-3-662-45854-9_4

[17] 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

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

[19] Manel Rodriguez-Soto, Maite Lopez-Sanchez, and Juan A. Rodríguez-Aguilar. 2020. A Structural Solution to Sequential Moral Dilemmas. In *Proceedings of the 19th International Conference on Autonomous Agents and Multi-Agent Aystems (AAMAS 2020)*.

[20] Diederik M. Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. 2013. A Survey of Multi-Objective Sequential Decision-Making. *J. Artif. Int. Res.* 48, 1 (Oct. 2013), 67–113.

[21] D. M. Roijers, S. Whiteson, R. Brachman, and P. Stone. 2017. . https://doi.org/10.2200/S00765ED1V01Y201704AIM034

[22] 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

[23] Stuart Russell, Daniel Dewey, and Max Tegmark. 2015. Research Priorities for Robust and Beneficial Artificial Intelligence. *Ai Magazine* 36 (12 2015), 105–114. https://doi.org/10.1609/aimag.v36i4.2577

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

[25] Richard S. Sutton and Andrew G. Barto. 1998. *Reinforcement learning - an introduction.* MIT Press. http://www.worldcat.org/oclc/37293240

[26] Peter Vamplew, Richard Dazeley, Adam Berry, Rustam Issabekov, and Evan Dekker. 2011. Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Machine Learning* 84 (07 2011), 51–80. https://doi.org/10.1007/s10994-010-5232-5

[27] Peter Vamplew, Richard Dazeley, Cameron Foale, Sally Firmin, and Jane Mummery. 2018. Human-Aligned Artificial Intelligence is a Multiobjective Problem. *Ethics and Information Technology* 20 (03 2018). https://doi.org/10.1007/s10676-017-9440-6

[28] Ibo van de Poel and Lambèr Royakkers. 2011. *Ethics, Technology, and Engineering: An Introduction.* Wiley-Blackwell.

[29] Christopher J. C. H. Watkins and Peter Dayan. 1992. Technical Note Q-Learning. *Machine Learning* 8 (1992), 279–292. https://doi.org/10.1007/BF00992698

[30] Yueh-Hua Wu and Shou-De Lin. 2017. A Low-Cost Ethics Shaping Approach for Designing Reinforcement Learning Agents. *arXiv* (12 2017).

[31] A. Wynsberghe. 2016. Service Robots, Care Ethics, and Design. *Ethics and Inf. Technol.* 18, 4 (Dec. 2016), 311–321. https://doi.org/10.1007/s10676-016-9409-x

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