

Multi-objective reinforcement learning for provably incentivising alignment with value systems

Manel Rodriguez-Soto ^{a,*}, Roxana Rădulescu ^b, Filippo Bistaffa ^a,
Oriol Ricart ^a, Arnau Mayoral-Macau ^a, Maite Lopez-Sanchez ^c,
Juan A. Rodriguez-Aguilar ^a, Ann Nowé ^d

^a Artificial intelligence research institute (IIIA-CSIC), Carrer de Can Planas, Campus de la UAB, Bellaterra, 08193, Spain

^b Utrecht University (UU), Netherlands

^c Universitat de Barcelona (UB), Spain

^d Vrije Universiteit Brussel (VUB), Belgium

ARTICLE INFO

Keywords:

Value alignment

Multi-objective reinforcement learning

Ethics

ABSTRACT

This paper addresses the problem of ensuring that autonomous learning agents align with multiple moral values. Specifically, we present the theoretical principles and algorithmic tools necessary for creating an environment where we ensure that the agent learns a behaviour aligned with multiple moral values while striving to achieve its individual objective. To address this value alignment problem, we adopt the Multi-Objective Reinforcement Learning framework and propose a novel algorithm that combines techniques from Multi-Objective Reinforcement Learning and Linear Programming. In addition, we illustrate our value alignment process with an example involving an autonomous vehicle. Here, we demonstrate that the agent learns to behave in alignment with the ethical values of safety, achievement, and comfort, with achievement representing the agent's individual objective. Such ethical behaviour differs depending on the ordering between values. We also use a synthetic multi-objective environment to evaluate the computational costs of guaranteeing ethical learning as the number of values increases.

1. Introduction

The problem of guaranteeing that autonomous agents act *value-aligned* (in alignment with human values) [1,2], is becoming of paramount importance as agents become more prevalent in our society. Consequently, there is a pressing need to develop ethically-aligned trustworthy AI [3] capable of respecting human values [4,5] across a variety of emerging application domains (e.g., social assistive robotics [6], self-driving cars [7], conversational agents [8]).

Indeed, there has been a rising interest, within both the fields of Machine Ethics [9,10] and AI Safety [11,12], in applying Reinforcement Learning (RL) [13] to address the challenging problem of *value alignment*. A prevailing approach in both of these communities for addressing this problem involves the creation of environments with incentives to behave ethically. These incentives are typically introduced through exogenous reward functions (e.g., [14–19]). The specification of such reward functions is based on some ethical insights. Afterwards, the rewards are integrated into an agent's learning environment employing an *ethical embedding*

* Corresponding author.

E-mail addresses: manel.rodriguez@iiia.csic.es (M. Rodriguez-Soto), r.t.radulescu@uu.nl (R. Rădulescu), filippo.bistaffa@iiia.csic.es (F. Bistaffa), oriolrv689@gmail.com (O. Ricart), arnau.mayoral@iiia.csic.es (A. Mayoral-Macau), maite_lopez@ub.edu (M. Lopez-Sanchez), jar@iiia.csic.es (J.A. Rodriguez-Aguilar), ann.nowe@vub.be (A. Nowé).

<https://doi.org/10.1016/j.artint.2025.104460>

Received 8 November 2024; Received in revised form 30 July 2025; Accepted 2 December 2025

Available online 7 December 2025

0004-3702/© 2025 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

process. However, in all these learning approaches, the ethical insights originate from a single moral value. It is widely recognised in the field of Ethics that human societies uphold multiple moral values, which are ordered according to how preferred they are (i.e., by considering first what they value most) [20–23]. This ordered collection of values is often called a *value system*. To the best of our knowledge, guaranteeing that an agent learns to behave aligned with a value system remains an open problem despite being the most common case in our society [20].

Against this background, in this work, we aim at automating the design of *ethical environments* that incentivise an agent to learn to behave in alignment with a value system while trying to pursue its individual objective. We do so by assimilating the agent's individual objective to the moral value of *achievement*¹, which is included in the value system at hand and prioritised with respect to the rest of moral values. Moreover, we adhere to the stance that “the end doesn't justify the means” and claim that achievement should always be ranked below the value system's higher ethical standards (such as, for instance, non-maleficence).

This paper tackles the value alignment problem by proposing a novel ethical embedding process in a reinforcement learning context. From a given (initial) social value system, we first enrich it by including the moral value of achievement, which encapsulates the agent's individual objective, as shown in Fig. 1 (Top-Left). Next, our ethical embedding shapes the learning environment, guaranteeing that an agent will be incentivised to learn an ethical behaviour aligned with the (enriched) value system. Since we consider multiple values, we refer to the ethical embedding of a value system as the Multi-Valued Ethical Embedding (MVEE), or simply as ethical embedding to ease readability.

Our contributions are three-fold. First, we formalise the Multi-Valued Ethical Embedding problem within the framework of Multi-Objective Markov Decision Processes (MOMDP) [25,26]. This formalisation models moral values as ethical objectives within a so-called *Multi-Valued* Markov Decision Process, an instance of a Multi-Objective MDP. Moreover, our formalisation paves the way for our definition of *ethical* policies, which characterise the behaviour of an agent aligned with a value system (i.e., in alignment with the moral values and respecting the preferences over those values). Finally, considering multiple objectives does require specific (more complex) algorithms for a learning agent. To ease the agent's learning, we reformulate the ethical embedding problem as finding the single-objective MDP that embeds all ethical objectives (so that the optimal policies in this MDP are ethical). To this end, we follow the prevailing approach (e.g. [16,18]) of applying a linear scalarisation function that *weighs* the rewards related to each ethical objective.

Second, we propose a novel algorithm to solve the ethical embedding problem that generalises the single-value ethical embedding process in [19]. Our novel algorithm combines recent developments in the Multi-Objective Reinforcement Learning literature (to compute ethical policies) with linear programming (to compute how to weigh ethical objectives). Fig. 1 outlines this algorithm, which transforms an input multi-objective environment $\mathcal{M}$ into a single-objective ethical environment $\mathcal{M}_*$. In this single-objective environment, the agent can thus apply a standard reinforcement learning method, and it is guaranteed that it will be incentivised to learn a policy aligned with the value system at hand.

Third, we illustrate our ethical embedding process by applying it to a novel (and simple) autonomous driving scenario, including three moral values (safety, achievement, and comfort), wherein the value of achievement represents the agents' individual objective. We chose those values inspired by those described by Caballero et al. in [27,28]. We show that an agent learns to behave according to a value system with a Q-learning algorithm [29]. Moreover, we also study the differences between the value-aligned behaviours that an agent would learn depending on the preferences over the considered values. Our study confirms that significantly different behaviours arise depending on the preferences on values in a value system. However, since the values mentioned above only represent a subset of those proposed by Caballero et al., we empirically analyse the computational cost to pay to guarantee ethical learning when considering an increasing number of values in a value system. We do so using the synthetic multi-objective environment generator from [30]. Our analysis indicates that our ethical embedding algorithm manages to do the ethical embedding of environments with up to eight objectives and almost 10^6 states in less than five hours. However, we also observe that its computational cost exponentially grows with the number of values considered.

Finally, notice that the work presented in this paper generalises our initial work in [19] and [31], which focused on developing theoretical foundations and algorithmic tools for designing a learning environment with guarantees for a single moral value. Therefore, the contributions in [19,31] must now be regarded as a particular case of the approach to design ethical environments presented in this paper.

The remainder of this paper is organised as follows. Section 2 provides the necessary background on Multi-Objective Reinforcement Learning and Ethics. Section 3 formally introduces our ethical embedding problem and the type of environments we target. Section 4 details our algorithm for building ethical environments. Section 5 details our empirical analysis of our ethical embedding algorithm. Finally, Section 6 discusses the related work, and Section 7 concludes and sets paths to future work.

2. Background

This section is dedicated to providing the necessary background for our approach to designing ethical environments with reinforcement learning. Thus, Section 2.1 introduces single-objective reinforcement learning. Subsequently, Section 2.2 introduces multi-objective reinforcement learning. Finally, Section 2.3 introduces the necessary concepts of the Ethics literature for understanding the role of moral values in our approach.

¹ Although achieving an individual goal can naturally be related to the moral value of diligence, we advocate for achievement because Schwartz defines it as based on competence and personal success when including it in his list of ten Basic Human Values [24].

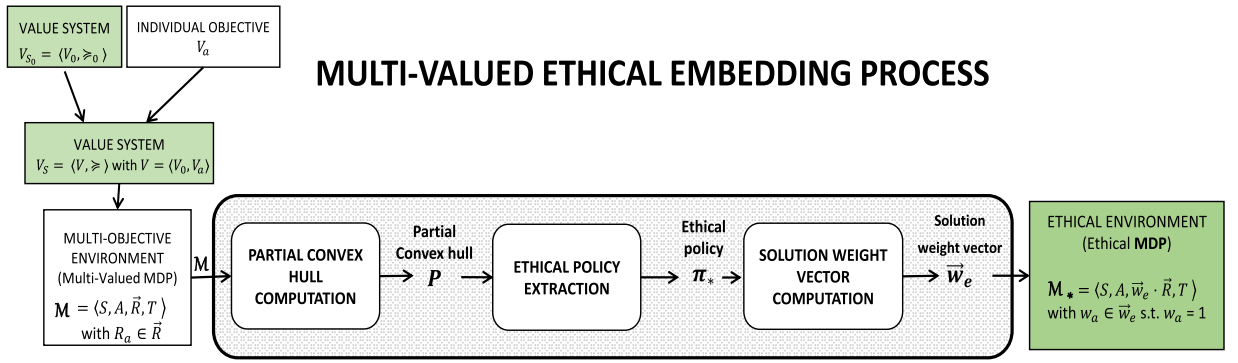


Fig. 1. Multi-Valued Ethical Embedding (MVEE) process for environment design (from left to right): partial convex hull computation, ethical policy extraction, and solution weight vector computation. The process takes as input a multi-objective environment $\mathcal{M}$ associated with an enriched value system $\mathcal{V}_S$, and returns as output an ethical (single-objective) environment $\mathcal{M}_*$. Rectangles stand for objects, whereas rounded rectangles correspond to processes. Colour green indicates that a given object is either a value system or an environment where aligning with the value system is incentivised. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

2.1. Single-objective reinforcement learning

In single-objective reinforcement learning (RL), the agent's learning environment is formalised as a *Markov decision process* (MDP) [13,32,33]. An MDP defines an environment in which an agent can consistently take actions to modify it and promptly receive a reward signal after each action. Formally:

Definition 1 (Markov Decision Process). A (finite single-objective)² Markov Decision Process (MDP) is defined as a tuple $\langle S, \mathcal{A}, R, T \rangle$ where 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 an MDP is characterised through a *policy* π that, for each state-action pair $\langle s, a \rangle$, specifies the probability of performing action a upon state s .

The agent begins at a given initial state $s_0 \in S$ of the MDP. In general, it is assumed that there is a probability function $\mu : S \rightarrow [0, 1]$ assigning the probability that each state will be the initial state of the agent [13].

The conventional approach for assessing the value of a policy is to compute the (expected) discounted accumulation of rewards an agent obtains when following it. The function that performs such computation is the so-called *value function* V , defined as:

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

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 $\pi_* \doteq \arg \max_{\pi} V^\pi$. Learning this policy π_* is the agent's 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, they all share the same optimal value function [13].

Notice that these optimal policies and this optimal value function exist because there is a total order between policies. In other words, we can always determine 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]. The main advantages of Q-learning are that it is a model-free algorithm (which means it is oblivious to the transition function T and the reward function R) and counts on theoretical properties that guarantee its convergence to a deterministic optimal policy.

2.2. Multi-objective reinforcement learning

Multi-objective reinforcement learning (MORL) deals with environments where an agent possesses various objectives. The added difficulty in MORL with respect to single-objective RL arises from the fact that, in many cases, the objectives are not fully compatible. Thus, the agent must choose between them, with each objective represented as a distinct reward function [34]. In MORL, the environment is formalised as a *Multi-Objective Markov Decision Process* (MOMDP), an MDP composed of a vectorial reward function. Formally:

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

² Through the paper, we refer to a finite single-objective MDP simply as an MDP.

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 ($V_i^\pi > V_i^{\pi'}$), whereas at the same time being worse for another objective ($V_j^\pi < V_j^{\pi'}$).

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 in an MOMDP. Otherwise, we might only be capable of determining a subset of policies that are incomparable 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* [35]. Now, we proceed to explain both problems.

2.2.1. Single-policy MORL

The majority of approaches in MORL assume the existence of a *scalarisation function* f , which condenses all the dimensions of a MOMDP into a single one (e.g., [17,36,37]). Formally, the 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 $f(\vec{V})$, thereby simplifying the multiple objectives into a single-objective problem.

It is particularly noteworthy in the case in which f is a linear function. When f is linear, the scalarised problem can be solved with single-objective reinforcement learning algorithms.³ Any linear scalarisation function f can be expressed as 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. Consequently, any $\vec{w}$ -optimal policy is 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 nor to be even *a priori* known, it is impossible to define an optimality criterion. In such cases, what we can do is compute the set of policies that *could* be optimal for some hypothetical scalarisation function. The literature refers to such policies as *undominated* policies. The set of all undominated policies is defined in [25]:

Definition 3 (Undominated set). Given an MOMDP $\mathcal{M}$, its undominated set U is the subset of policies π_* and their associated value vectors $\vec{V}^{\pi_*}$ that are maximal for some scalarisation function f :

$$U(\mathcal{M}) \doteq \{\vec{V}^{\pi_*} \mid \pi_* \in \Pi^{\mathcal{M}} \wedge \exists f \in \mathcal{F}: f(\vec{V}^{\pi_*}) = \max_{\pi \in \Pi^{\mathcal{M}}} f(\vec{V}^\pi)\} \quad (3)$$

where $\Pi^{\mathcal{M}}$ is the set of policies of $\mathcal{M}$, and $\mathcal{F}$ is the set of all possible scalarisation functions.

In this paper, we are interested in the particular case for which we consider linear scalarisation functions. In such case, the undominated set is called the *convex hull* [25]:

Definition 4 (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 $\vec{w}$:

$$CH(\mathcal{M}) \doteq \{\vec{V}^{\pi_*} \mid \pi_* \in \Pi^{\mathcal{M}} \wedge \exists \vec{w} \in \mathbb{R}^n: \vec{w} \cdot \vec{V}^{\pi_*} = \max_{\pi \in \Pi^{\mathcal{M}}} \vec{w} \cdot \vec{V}^\pi\}, \quad (4)$$

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) [38] is a classical algorithm for obtaining the convex hull of an MOMDP. CHVI is a model-based algorithm that is always guaranteed to converge. CHVI can be applied when only considering the linear scalarisation function f of an MOMDP. If it is unnecessary to compute the whole convex hull of an MOMDP, CHVI can be adapted to compute only a partial region (for instance, the region of positive weights).

There also exists model-free algorithms such as Pareto Q-Learning [39], which compute a superset of the convex hull, the *Pareto front*, instead. The Pareto front contains the set of policies that are maximal for some monotonically increasing scalarisation function, a notion weaker than linear. The algorithm selection will depend on the available knowledge of f .

2.3. Values: Value pluralism, value systems, and value alignment

To understand the value alignment problem and why it is a pressing challenge, we need to define the concept of *values* first. In this paper, we follow the philosophical views of Arnold et al. [40–42] by defining *values* as: “*natural or non-natural facts about what is good or bad, and about what kinds of things ought to be promoted, from an ethical point of view*”. In other words, moral values constitute ethical criteria that allow us to discern good from wrong daily. For instance, in the context of a driving scenario, the value of safety considers that driving aggressively is wrong, and the value of achievement regards pursuing one's own goals as good.

³ This is 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 not true in general for the non-linear case.

It is important to emphasise that, as can be observed from the previous example about driving, in this work, we assume a *value pluralistic* view [20–23,43], that is, that there exist many different moral values. This pluralistic view is typically assumed to be true in the Ethics and AI literature (for example, in [2,44,45]). It is also worth remarking here that value pluralism does not imply moral subjectivity [20]: values are not subjective preferences, which would imply that any ethical judgement is correct. Quoting Cooper [20]: “*Tastes and ethical values belong in different bags. [...] I may not be able to give good public reasons to justify my dislike of chocolate, but I can give good public reasons to support my judgment that sadistic behaviour is wrong.*”

The natural conclusion from adopting a pluralistic view of values is that these values might conflict in many situations. Back to the previous example, if someone’s goal is to get home as fast as possible, they will want to drive their car as fast as possible, aligning with the value of achievement. However, this action will conflict with the value of safety. When conflicts exist between values, a common solution is to compare them and observe which outcome would be preferred. The literature provides a formal name to a set of moral values with preferences among them: a *value system* [46–48]. We will use the concept of value system extensively in this work, for which we provide a formal definition in Section 3. For example, a value system prioritising safety over achievement tells us that it is morally wrong to drive dangerously even if we desire to get home early today.

While there are many possible ways of defining value systems, in this work we follow the specific definition of [48], which considers that value systems establish a total order between values. We consider that this formalisation can be a relevant model of a value system in multiple scenarios. Let us illustrate it with three examples:

- **Healthcare:** Clinicians typically consider four main bioethical values when making decisions: beneficence, non-maleficence, patient autonomy, and justice [22]. Although all four values are of the utmost importance for them, the patient’s autonomy is always the most prioritised value for them. This can lead to extreme cases in which a clinician cannot perform a life-saving surgery if the patient has explicitly rejected it (aligning with their autonomy). There would not be any acceptable (and probably neither legal) trade-off between the patient’s autonomy and beneficence in such an example.
- **Childcare:** We can also consider a parent who decides to stay at home nurturing their newborn child (prioritising alignment with the value of benevolence) even though they are craving to go to a dinner with friends (which would align with their own achievement). There is no intermediate option here.
- **Engineering:** Consider a car engineer who must destroy a new car prototype because it fails some security tests by a very small margin (prioritising alignment with safety over any other value). There would not be any option for a trade-off between producing and selling this car (aligning with achievement) and destroying it (aligning with safety). ”.

Having defined moral values and value system, we can finally introduce the concepts of value alignment and the value alignment problem. The *value alignment* problem is the problem of guaranteeing that agents act in alignment with human values [1,2,41]. An agent exhibiting a value-aligned behaviour should always pursue objectives beneficial to humans [1,2,40,42]. As mentioned in Section 1, as autonomous agents become more autonomous and intelligent, they are starting to play an increasingly prominent role in our daily lives. Thus, we need to guarantee that agents will always behave in alignment with *our value systems*. Not being able to guarantee value-aligned would lead us to a scenario with catastrophic consequences [2].

Moreover, for an agent’s behaviour to be considered ethical in a specific domain, we require that the value system actually prioritises the highest ethical standards in that domain over values without ethical considerations, such as the value of achievement, which only aims to maximise one own’s desires [24]. This requirement is inspired by biomedical ethics, where non-maleficence, beneficence, autonomy, and justice [49] are considered to be the highest ethical standards in the field. Hence, a medical environment must prioritise all four biomedical values higher than, for instance, the value of achievement when incentivising the agent to learn to behave ethically. This is so because a medical agent that, for instance, willingly hurts a patient cannot be considered ethical. As an alternative example, consider a driving scenario where safety is the highest ethical standard that must be prioritised over achievement. Thus, a car running into a pedestrian in a rush to reach its destination cannot be considered to behave ethically.

3. Formalisation of the multi-valued ethical embedding problem

In this section, we formalise the *ethical embedding* problem considering multiple moral values. As previously introduced, our main goal is to design an environment that guarantees that an agent is incentivised to learn to behave ethically, that is, in alignment with a system of multiple moral values. As mentioned in Section 2, the Ethics literature considers that moral values (also called ethical principles) express the moral objectives worth striving for [23]. Furthermore, it is common to consider the set of moral values together with preferences among them [46–48]. Here, we formalise a *Value System* in the vein of [48] as follows:

Definition 5. A value system $\mathcal{V}_S$ is a tuple $\mathcal{V}_S = \langle \mathcal{V}, \succeq \rangle$, where: $\mathcal{V} = \{v_1, \dots, v_n\}$ stands for a non-empty set (i.e. $n > 0$) of moral values; and $\succeq$ is a total order over the moral values of $\mathcal{V}$. If $v \succeq v'$ we say that v is more preferred than v' .

Notice that we formalise a value system as a totally ordered set. This total order between values of a value system helps the agent to behave when facing solve value conflicts. Thus, this total order between values will have a direct impact on our definition of value-aligned behaviour, which we formalise in the following Definition 7. “Notice that we formalise a value system as a totally ordered set. This total order between values of a value system helps the agent to behave when facing solve value conflicts. Thus, this total order between values will have a direct impact on our definition of value-aligned behaviour, which we formalise in the following Definition 7. In this work, we will prove that when considering this specific definition of value system, it is always possible to incentivise value-aligned behaviour. This theoretical property is one of the main reasons why we consider this value system definition: we focus on providing algorithms that guarantee a value-aligned behaviour. In more detail, we deem it important to remark on the

need for offering value alignment algorithms with formal guarantees. In many scenarios, we need to have absolute certainty that the agents will behave in strict alignment with a value system. For this reason, in this work we have decided to focus on such moral scenarios in which we could guarantee value alignment: those in which the value system could be formalised as a total order between values.

As mentioned in Section 1 and shown in Fig. 1, we consider that we are given $\mathcal{V}_{S_0}$ —the value system shared by a human society to align with— and the agent's objective. Then, we represent this objective as the *achievement* moral value v_a and embed it within $\mathcal{V}_{S_0}$ to produce $\mathcal{V}_S$. Afterwards, we can transform the ethical knowledge in the value system $\mathcal{V}_S$ into ethical rewards of a particular type of Multi-Objective Markov Decision Process (MOMDP) [34]. In this MOMDP $\mathcal{M}$, each reward function is associated with a value of the value system $\mathcal{V}_S$, generalising the approach of [31]. Each moral value evaluates actions as being aligned or misaligned with respect to it. We can translate this dichotomy in terms of rewards as: receiving a positive reward for value-aligned actions, and a negative reward for value-misaligned actions.⁴ Since the performance of an action cannot simultaneously be aligned and misaligned with the same moral value, the corresponding reward component will be simply positive if the action is value-aligned, negative if it is misaligned, and 0 if it is neutral to the value. In more detail, the numerical value of the reward indicates to us how aligned / misaligned it is with respect to the moral value. Thus, near-zero rewards are almost neutral to the value, while large rewards indicate a strong degree of aligned (if positive) or misalignment (if negative). Moreover, the moral value of achievement v_a —which corresponds to the agent's objective— is mapped to one reward R_a .

To simplify notation, in the vein of [50], we define a family of MOMDPs called *Multi-Valued MDPs* that encapsulates all previously mentioned conditions. In particular, Multi-Valued MDPs are a family of MOMDPs in which (i) maximising each of its reward function is equivalent to aligning with some value of a value system, (ii) one of its reward functions (R_a) is always associated with the value of achievement, but (iii) the value of achievement is never the most prioritised value of the value system. Formally:

Definition 6 (Multi-Valued MDP). Given: i) $\mathcal{V}_S = \langle \mathcal{V}, \succeq \rangle$ a value system $\mathcal{V}_S = \langle \mathcal{V}, \succeq \rangle$ with n values including the achievement value $v_n = v_a$ (which cannot be prioritised first in $\succeq$); and ii) an MOMDP

$$\mathcal{M} = \langle S, \mathcal{A}, (R_1, \dots, R_{n-1}, R_a), T \rangle, \quad (5)$$

we say that $\mathcal{M}$ is a *Multi-Valued MDP* if and only if each $R_{i:1,\dots,n} : S \times \mathcal{A} \rightarrow \mathbb{R}$ is an *ethical* reward function that rewards positively the performance of actions evaluated as value-aligned ($R_i > 0$), and rewards negatively the performance of misaligned actions ($R_i < 0$) for moral value $v_i \in \mathcal{V}$. In particular, we denote as $R_a = R_n$ the reward aligned with the agent's individual objective (i.e., with value v_a).

Notice that Definition 6 explicitly forbids Multi-Valued MDPs such that their associated value system $\mathcal{V}$ prioritise the value of achievement v_a over all other values. As argued in Section 1, this design choice aims to avoid value systems that promote highly individualistic behaviours over higher ethical standards.

Example 1. Let $\mathcal{V}_S = \langle \mathcal{V}, \succeq \rangle$ be a value system with three values ordered $v_1 \geq v_2 \geq v_3$, where v_3 corresponds to the achievement value. Then, we consider the associated Multi-Valued MDP $\mathcal{M} = \langle S, \mathcal{A}, (R_1, R_2, R_3), T \rangle$ specified by $\mathcal{V}_S$ such that R_1 is defined according to v_1 , R_2 to v_2 and R_3 to v_3 .

Having the Multi-Valued MDP $\mathcal{M}$ (see Fig. 1), we can now move forward with the value alignment process by addressing the *ethical embedding* problem: that of setting the rewards of the environment so that we guarantee that an agent is incentivised to learn to behave ethically. As ethical embedding constitutes the main focus of this work, the rest of the paper is dedicated to formalising and solving it.

To formalise the ethical embedding problem, we must first define the notion of ethical behaviour. We do so by first defining a value-aligned policy in the context of a Multi-Valued MDP $\mathcal{M}$ and its associated value system $\mathcal{V}_S$. In a Multi-Valued MDP associated with $\mathcal{V}_S$, its rewards are defined according to the values in $\mathcal{V}_S$. Moreover, the ordering between values also defines a lexicographic ordering [51] between rewards. This lexicographic ordering gives more importance to rewards from the most preferred values. In this manner, we define a *value-aligned policy* as the policy that is most aligned with the value system. Example 2 below illustrates how lexicographic ordering handles value preferences to order alternative policies, allowing us to determine the value-aligned policy as the policy performing aligned actions with respect to the most preferred values. Formally, a value-aligned policy maximises rewards following the lexicographic ordering induced by $\mathcal{V}_S$ over the rewards of a Multi-Valued MDP:

Definition 7 (Value-aligned policy). Let $\mathcal{M}$ be a Multi-Valued MDP with a value system $\mathcal{V}_S$. Let $l_{\mathcal{V}_S}$ be the lexicographic ordering of objectives induced by ordering $\succeq$ of the value system $\mathcal{V}_S$. We say that a policy π_* is a value-aligned policy in $\mathcal{M}$ if and only if its value vector $\vec{V}^{\pi_*}$ is optimal with respect to the lexicographic ordering $l_{\mathcal{V}_S}$ (in short, it is $l_{\mathcal{V}_S}$ -optimal).

Example 2. Considering the value system $\mathcal{V}_S$ and the Multi-Valued MDP $\mathcal{M}$ from Example 1, the order $v_1 \geq v_2 \geq v_3$ induces the lexicographic order $l_{\mathcal{V}_S} = \langle 1, 2, 3 \rangle$. Now, let us assume that $\mathcal{M}$ only has two states, s_0 , and s_1 . There are five possible actions, and the following rewards:

$$\vec{R}(s, a_1) = (-1, 5, 4),$$

$$\vec{R}(s, a_2) = (8, 1, -2),$$

⁴ For example, considering the value of safety, to yield is an aligned action, while producing an accident is misaligned with the value of safety. Regarding ethical rewards, yielding deserves a positive reward, while producing an accident causes a negative reward.

$$\begin{aligned}\bar{R}(s, a_3) &= (8, 4, 3), \\ \bar{R}(s, a_4) &= (2, 5, 3), \\ \bar{R}(s, a_5) &= (7, -1, 2).\end{aligned}$$

All actions transition to the terminal state s_1 . Then, the lexicographic ordering l_{V_S} prioritises the values of each action as: $(8, 4, 3) \geq (8, 1, -2) \geq (7, 1, -2) \geq (2, 5, 3) \geq (-1, 5, 4)$ because a_3 and a_2 accumulate more rewards for v_1 (8) than a_5 (2) (and a_5 more than a_4 , and a_4 more than a_1) and although both a_3 and a_2 accumulate the same rewards for v_1 , a_3 accumulates more rewards for v_2 (the second most preferred value) than a_2 . Hence, $a_3 \geq a_2 \geq a_5 \geq a_4 \geq a_1$ and $\pi(s) = a_3$ is the *value-aligned policy* (i.e., optimal with respect to the lexicographic ordering l_{V_S}).

In this manner, we can always compute its Value-aligned policy for any given value system. However, recall from the discussion in Section 2.3 that, for the policy to be considered ethical in a specific domain, we require that the value system actually prioritises the highest ethical standards in that domain over the value of achievement. Therefore, we only consider that a value-aligned policy is *ethical* if and only if it is derived from a value system V_S where v_a is ranked below values representing higher ethical standards.

In this work we take an environment-designer approach for incentivising value-aligned policies [19,31]. This approach has several benefits to the learning agent, which we detail in the coming Section 3.2. First, we explain how such an environment designer approach works. In short, this approach consists of designing a simpler learning environment for the agent where it can learn with single-objective reinforcement learning algorithms. Thus, we tackle the multi-valued ethical embedding problem by transforming a Multi-Valued MDP $\mathcal{M}$ into a single-objective MDP $\mathcal{M}_*$ (see Fig. 1). We do it by scalarising the vectorial value function $\vec{V}$ of $\mathcal{M}$ employing a scalarisation function f_e , which we call the *embedding function*.

In more detail, in this work, we assume that this embedding function f_e is linear.⁵ Thus, the embedding function will be of the form $f_e(\vec{V}) = \vec{w} \cdot \vec{V}$ for some given weight vector $\vec{w} \in \mathbb{R}^n$. Such an embedding function will transform the original environment $\mathcal{M} = \langle S, \mathcal{A}, \vec{R}, T \rangle$ into the single-objective environment $\mathcal{M}_* = \mathcal{M} = \langle S, \mathcal{A}, \vec{w} \cdot \vec{R}, T \rangle$.

Before we continue, we introduce the following notation: we will refer to the optimal policies in the single-objective environment $\mathcal{M}_*$ scalarised by some weight vector $\vec{w}$ as the $\vec{w}$ -optimal policies.

Consequently, given a Multi-Valued MDP $\mathcal{M}$, we aim to find a scalarisation function that guarantees that the only possible optimal policies for an agent in the scalarised environment ($\mathcal{M}_*$, a single-objective MDP) are the value-aligned policies of the original Multi-Valued MDP $\mathcal{M}$. Therefore, the ethical embedding problem consists of computing a weight vector $\vec{w}$ that transforms an initial environment into another one wherein an agent is incentivised to behave ethically. Formally:

Problem 1 (Ethical embedding problem). Let $\mathcal{M} = \langle S, \mathcal{A}, (R_1, \dots, R_{n-1}, R_n), T \rangle$ be a Multi-Valued MDP with a value system V_S . The multi-valued ethical embedding problem is that of finding the weight vector $\vec{w}$ such that all optimal policies in the scalarised MDP $\mathcal{M}_* = \langle S, \mathcal{A}, \sum_{i=1}^n w_i R_i, T \rangle$ are also ethical in $\mathcal{M}$ (as defined in Definition 7).

Any weight vector $\vec{w}$ that guarantees that all optimal policies (concerning $\vec{w}$) are also ethical is a solution of Problem 1. However, as aforementioned, in this work we take an environment-designer approach which requires us to consider the following assumptions:

1. **We fix the achievement weight:** First, we consider that the learning agent always has its own individual objective (represented as the value of achievement in the value system, and as the reward function R_a in the Multi-Valued MDP). As a point of reference, when designing an ethical environment, we will always consider that the scale of its individual/achievement rewards is R_a , the standard one (i.e., the weight vector $\vec{w}$ solution of Problem 1 will always have $w_a = 1$). This assumption has no impact on the solvability of Problem 1, as we will observe in the next Section 4. Moreover, forcing $w_a = 1$ allows us to know, when computing the solution weight vector $\vec{w}$, how much we need to scale the rest of the objectives of the Multi-Valued MDP so that the agent prefers to behave value-aligned rather than only maximising its own objective.
2. **We look for the minimum weight vector:** Second, we assume that incentivising the agent with ethical rewards has a cost to the designer or the agent. Take as a motivating example the problem of setting traffic sanctions. A sanction of 1 million euros if someone surpasses the speed limit will probably guarantee that people will respect speed limits. However, it is probably better to search for the minimum fee that will guarantee alignment with traffic rules. Thus, we aim at finding the solution weight vector $\vec{w}$ that has the smallest possible weights (i.e., the weight vector $\vec{w}$ with the minimal scalarised accumulated rewards for the agent).

Next, we provide an example of weight vector that would solve Problem 1 for the Multi-Valued MDP of previous Example 2:

Example 3. Considering the lexicographic ordering l_{V_S} and the Multi-Valued MDP $\mathcal{M}$ from Example 2, the weight vector $\vec{w} = (100, 10, 1)$ guarantees that the value-aligned policy is optimal. Indeed:

$$\vec{w} \cdot (8, 4, 3) = 843 > \vec{w} \cdot (8, 1, -2) = 808 > \vec{w} \cdot (2, 5, 3) = 253 > \vec{w} \cdot (-1, 5, 4) = -46.$$

Yet, there exist other solutions with smaller weights, such as $\vec{w}' = (4, 3, 1)$.

⁵ Linear scalarisation functions are widely used in the MORL literature because they guarantee that the Bellman equation will be preserved in the scalarised MDP [25]. It is important to remark that they pose some limitations on the learning of the agents. The most severe being that an agent with a linear scalarisation function will not be able to learn any policy from the concave region of the Pareto front solution set [25]. This will not be a problem in our case because we will prove that our desired solution lies in the convex region of the Pareto front: the convex hull.

3.1. Solvability of MVEE problems

In this section, we prove that [Problem 1](#) is solvable for any finite Multi-Valued MDP. First, we must prove an intermediate result: in finite MOMDPs, any lexicographic ordering can be expressed as a linear scalarisation function. Formally:

Theorem 1. *Given a finite n -objective Markov Decision Process (MOMDP) $\mathcal{M} = \langle S, \mathcal{A}, \vec{R}, T \rangle$ and a lexicographic ordering l , there exists some weight vector $\vec{w}$ with strictly positive weights $w_i > 0 \forall i$ for which every policy π that is $\vec{w}$ -optimal also maximises the lexicographic ordering l and vice versa.*

Proof. We prove it by induction. First, we consider a 2-objective MDP in which the lexicographic ordering is $l_2 = \langle 2, 1 \rangle$. We consider a weight vector of the form $\vec{w}_* = (1, w_2)$. Let π be a policy that is $(0, 1)$ -optimal (i.e., policies that are optimal for the weight vector $(0, 1)$). Then, let us define the weight w_2 as follows:

$$w_2 = \max_s \max_{\rho \notin \Pi_2} \frac{V_1^\rho(s) - V_1^\pi(s)}{V_2^\pi(s) - V_2^\rho(s)} + \epsilon,$$

where $\epsilon > 0$ and Π_2 is the set of $(0, 1)$ -optimal policies. For such weight vector $\vec{w}_* = (1, w_2)$, any $\vec{w}_*$ -optimal policy is necessarily $(0, 1)$ -optimal. Furthermore, since $w_1 = 1 > 0$, such a weight vector guarantees that among two $(0, 1)$ -optimal policies, the one with more value in objective 1 will be preferred. Therefore, any $\vec{w}_*$ -optimal policy is also a policy that maximises the lexicographic ordering l_2 , as desired.

By induction, let us assume we can create a linear scalarisation function for any lexicographic ordering of up to $n - 1$ objectives. Now we consider an MOMDP $\mathcal{M}$ with n objectives and a lexicographic ordering l_n of the n objectives. Without loss of generality, we consider that objective n is the most preferred one in l_n . Consider now the lexicographic l'_n ordering without objective 1, which orders the other $n - 1$ objectives. By the induction hypothesis, a weight vector $\vec{w}'$ exists for which any $\vec{w}'$ -optimal policy is l'_n -optimal. Hence, we can re-express the vectorial reward function of $\mathcal{M}$ as:

$$(R_1, \sum_{i=2}^n w'_i R_i).$$

In other words, we can re-express the MOMDP $\mathcal{M}$ as a 2-objective MDP $\mathcal{M}'$ in which the lexicographic ordering is preserved: Any policy maximising the lexicographic order l_n will maximise the second objective of $\mathcal{M}'$ as much as possible (i.e. the last objective of $\mathcal{M}$) subject to maximising the first objective of $\mathcal{M}'$ (i.e., the lexicographic order l'_n of $\mathcal{M}$). Since we have already proven that we can find a weight vector that preserves the lexicographic ordering for a 2-objective MDP, we can also find it for $\mathcal{M}$, as desired.

We have proven that, for any lexicographic order l , there exists a weight vector $\vec{w}$ for which all its $\vec{w}$ -optimal policies maximise l . Notice that all policies π that maximise a lexicographic ordering l share the same value vector $\vec{V}^\pi$ by definition. Hence, for such weight vector $\vec{w}$, all policies that maximise l will also be $\vec{w}$ -optimal. $\square$

Now, we are ready to prove that [Problem 1](#) is solvable.

Theorem 2. *Let $\mathcal{M} = \langle S, \mathcal{A}, (R_1, \dots, R_{n-1}, R_a), T \rangle$ be a Multi-Valued MDP with a value system $\mathcal{V}_S$. There exists a vector $\vec{w}_E$ of positive weights $w_a > 0, w_i > 0$ for every i , for which every optimal policy in the MDP*

$$\mathcal{M}' = \langle S, \mathcal{A}, w_a \cdot R_a + \sum_{i=1}^{n-1} w_i R_i, T \rangle$$

is also a value-aligned policy in $\mathcal{M}$.

Proof. Direct from [Theorem 1](#), since any value-aligned policy maximises some lexicographic ordering l . $\square$

To finish this section, there is an essential remark about [Theorem 1](#) concerning the relationship between lexicographic orders and linear scalarisation functions. Given a MOMDP $\mathcal{M}$, its *convex hull* $CH(\mathcal{M})$ [52] is defined as the set of policies that are strictly better than any other policy for some linear scalarisation function (i.e., some weights). A natural conclusion of [Theorem 1](#) is that the *convex hull* of any finite MOMDP contains all policies optimal for some lexicographic ordering. Formally:

Corollary 1. *Given a finite n -objective Markov Decision Process (MOMDP) $\mathcal{M} = \langle S, \mathcal{A}, \vec{R}, T \rangle$, then its set of policies optimal for any lexicographic ordering is a subset of its convex hull $CH(\mathcal{M})$.*

3.2. The environment-designer approach

As previously discussed, in this work, we follow an environment-designer approach for incentivising the agent to learn a value-aligned policy. Such an approach consists of designing a single-objective environment in which the learning agent will find that it is optimal to behave following a value-aligned policy. Moreover, in the previous section we have proven that we can always design such an environment for any Multi-Valued MDP thanks to [Theorem 2](#). However, one fundamental question remains unanswered: why we should design such a single-objective environment in the first place? Why not let the agents learn using a multi-objective reinforcement learning algorithm directly? We will refer to this alternative approach as the *agent-centric* approach.

There are three main reasons why we advocate for the environment-designer approach:

1. First, in this work, we follow the mechanism design literature [53], where it is assumed that we cannot modify the agent's preferences. This constraint might occur for a variety of reasons, including that the agent might belong to a third party unrelated to us, or that the agent is autonomous and has its own goals and motivations. Thus, in order for the agent to behave value-aligned, we will need to incentivise it to do so. Hence, our work focuses on creating the incentives that guarantee that the agent learns to behave value-aligned even if we have no control over its goals.
2. Second, in a similar vein, we assume that we cannot control the learning algorithm of the agent (for the same reasons as previously mentioned). Ideally, an agent could directly learn an ethical policy applying a lexicographic reinforcement learning algorithm (such as TLO [54]). However, we assume that we have no guarantee that they will choose such an algorithm during training (or even the algorithm with the correct lexicographic order). In the worst case, the agent's learning algorithm could have been designed to be as misaligned as possible to the value system. In our ethical embedding approach, by transforming the environment into a single-objective one, we are ensuring that the agent's learning algorithm will be irrelevant. In more detail, by designing an environment in which the only optimal policy is to behave value-aligned, all single-objective reinforcement learning algorithms will lead to the value-aligned policy, as they always compute the optimal policy.
3. Finally, transforming the multi-objective environment into a single-objective one also simplifies the learning problem for the agents. For any multi-objective reinforcement learning algorithm, its single-objective counterpart will always be less computationally complex, even if minimally. This is also one of the reasons why linear scalarisation functions are so popular in the multi-objective reinforcement learning literature: they allow for the direct use of any single-objective RL algorithm [25].

For these three reasons, in this work, we advocate for the need to develop an environment-designer approach, which involves designing an ethical single-objective environment that incentivises agents' ethical behaviour. These three reasons show how the value alignment problem is more complex than simply finding the ethical policy the agent should follow. To guarantee value alignment, we need to also be resilient against agents equipped with their own learning algorithms, and with their own preferences beyond our control.

4. Solving the MVEE problem

This section explains how to compute a solution weight vector for the multi-valued ethical embedding problem (Problem 1). Such weight vector will allow us to transform our Multi-Value MDP $\mathcal{M}$ into a (single objective) MDP $\mathcal{M}_*$ by combining the ethical rewards derived from the value system $\mathcal{V}_S$ into a single reward in $\mathcal{M}_*$, the environment in which the agent learns a value-aligned policy.

In short, our algorithm to solve the MVEE problem, the so-called *Ethical Embedding* algorithm, performs the following three steps (see Fig. 1):

1. *Computation of the partial convex hull* of a Multi-Valued MDP $\mathcal{M}$ containing the subset P of policies optimal for some weight vector with positive weights.
2. *Extraction of one value-aligned policy* π_* from the partial convex hull P .
3. *Computation of the solution weight vector*: use the extracted value-aligned policy π_* to find a weighting $\vec{w}$ of the rewards in $\mathcal{M}$ to yield a single-objective, ethical environment $\mathcal{M}_*$ wherein all optimal policies are guaranteed to be value-aligned.

The following three subsections provide the theoretical ground for computing each step of our algorithm. After that, we present in Section 4.4 the algorithm as a whole.

4.1. Computation of the partial convex hull

To compute the specific weight vector $\vec{w}$ that solves Problem 1, we resort to the multi-objective RL concept of *convex hull*. Recall that, given a MOMDP $\mathcal{M}$, its *convex hull* $CH(\mathcal{M})$ [52] contains those policies that are strictly better than any other policy for some linear weights. By Theorem 2 and Corollary 1, we know that the convex hull of a Multi-Valued MDP contains all value-aligned policies since we can express them as optimal for some linear combination of weights. Therefore, the convex hull allows us to compute the weight vector necessary to guarantee that all optimal policies are ethical. Furthermore, again from Theorem 2, we only need to compute the *partial convex hull* P of policies that are optimal for some positive weights (i.e., $\vec{w}$ such that $w_i > 0$ for all i).

Significantly, we can benefit from state-of-the-art algorithms (such as Convex Hull Value Iteration [38], which compute the whole convex hull of a MOMDP) to compute only the region of interest from the convex hull.

Example 4. Consider the Multi-Valued MDP $\mathcal{M}$ from Example 3, that only has two states, s_0 , and s_1 . There are five possible actions, and the following rewards:

$$\vec{R}(s, a_1) = (-1, 5, 4),$$

$$\vec{R}(s, a_2) = (8, 1, -2),$$

$$\vec{R}(s, a_3) = (8, 4, 3),$$

$$\vec{R}(s, a_4) = (2, 5, 3),$$

$$\vec{R}(s, a_5) = (7, -1, 2).$$

All actions transition to the terminal state s_1 . Then, define for each action a_i the deterministic policy π_i such that it always selects the action $\pi(s_i) = a_i$. It can be proven that the partial convex hull of deterministic policies that are optimal for some positive weights is:

$$CH(\mathcal{M}) = \{\vec{V}^{\pi_1}, \vec{V}^{\pi_2}, \vec{V}^{\pi_3}, \vec{V}^{\pi_4}\} = \{(-1, 5, 4), (8, 1, -2), (8, 4, 3), (2, 5, 3)\}.$$

In particular, notice how π_3 achieves a vectorial reward strictly greater than π_5 for all objectives (i.e., π_5 is strictly dominated by π_3). Thus, π_5 cannot belong to the convex hull. Meanwhile, notice how each of the four remaining policies achieves maximum value for at least one objective and, thus, they are guaranteed to belong to the partial convex hull.

4.2. Extraction of a value-aligned policy

After computing the partial convex hull $P \subseteq CH(\mathcal{M})$, we are ready to perform the second step of our algorithm, which is the extraction of the value-aligned policy (together with its value vector) from P .

To find the value-aligned policy among the policies in P , we must order P lexicographically. We need to follow the total ordering established by the value system $\mathcal{V}_S$ to order the policies, which allows us to select the value-aligned policy. Formally, let P_l be the sequence of policies of the partial convex hull ordered by the lexicographic ordering $l_{\mathcal{V}_S}$ induced by $\mathcal{V}_S$:

$$P_l \doteq (\pi_k)_{k=1}^K \text{ such that } \vec{V}^{\pi_i} \geq \vec{V}^{\pi_{i+1}}, \quad (6)$$

where K is the number of policies in P . Let P_{l_k} denote the k^{th} element of P_l . Then, we can extract a value-aligned policy π_* from P by computing P_{l_1} , the first element of P_l .

Notice that for any policy π in the positive partial convex hull P , we know its value $\vec{V}^\pi$ because we obtained it when computing the partial convex hull. Thus, computing P_l requires only a sorting operation.

Example 5. Consider the Multi-Valued MDP $\mathcal{M}$ from [Example 4](#), together with the value system $\mathcal{V}_S$ with order $v_1 \geq v_2 \geq v_3$ from [Example 3](#). Recall from [Example 4](#) that the partial convex hull of $\mathcal{M}$ is:

$$CH(\mathcal{M}) = \{\vec{V}^{\pi_1}, \vec{V}^{\pi_2}, \vec{V}^{\pi_3}, \vec{V}^{\pi_4}\} = \{(-1, 5, 4), (8, 1, -2), (8, 4, 3), (2, 5, 3)\}.$$

Then, we can order lexicographically the values of these policies according to the order of the value system as follows: $(8, 4, 3) \geq (8, 1, -2) \geq (2, 5, 3) \geq (-1, 5, 4)$. Thus, we extract the policy $\pi_3(s_0) = a_3$ with vectorial value $\vec{V}^{\pi_3}(s_0) = (8, 4, 3)$ as it is the ethical policy (i.e., the policy that maximises the lexicographic order induced by $\mathcal{V}_S$).

4.3. Computation of the solution weight vector

To compute the solution weight vector (the scalarisation function), we use the computed partial convex hull (step 1) and the reference value-aligned policy π_* (step 2). The solution weight vector ($\vec{w}_e$) will guarantee that the value-aligned policies in the initial environment $\mathcal{M}$ are the optimal policies in the ethical (single-objective) environment $\mathcal{M}_*$ (see [Fig. 1](#)). In other words, the scalarisation function with weight vector $\vec{w}_e$ will help us create an ethical environment as a single-objective MDP, wherein the agent will learn a value-aligned policy.

Next, we show how to cast the problem of finding the solution weight vector ($\vec{w}_e \in \mathbb{R}^n$, where $n > 0$ is the number of moral values) as an optimisation problem that we solve with linear programming. Consider a Multi-Valued MDP $\mathcal{M}_1$ with a single initial state s_0 , with value-aligned policy π_* and partial convex hull P . Our goal is to find the smallest (non-negative) values for the weight vector $\vec{w}_e$ of the scalarised function $\vec{w}_e \cdot \vec{V}^\pi$ so that the value-aligned policy π_* of $\mathcal{M}_1$ is optimal in $\mathcal{M}_*$. This computation amounts to solving the following LP:

$$\text{Min. } \vec{w}_e \cdot \vec{V}^{\pi_*}(s_0) \quad (7)$$

$$\text{s. t. } \vec{w}_e \cdot V^{\pi_*}(s_0) \geq \vec{w}_e \cdot V^\pi(s_0) + \epsilon \forall \pi \in P_{-\pi_*}, \quad (8)$$

$$w_i > 0 \forall i, \quad (9)$$

$$w_a = 1, \quad (10)$$

where $w_i \in \mathbb{R}^+$ are the decision variables, $\epsilon > 0$ is a small positive number, $P_{-\pi_*}$ is the subset of the partial convex hull P without π_* , and w_a is the weight corresponding to the agent's individual objective. The objective function of [Eq. \(7\)](#) indicates that we aim at minimising the scalarised value of the value-aligned policy (by minimising the values in $\vec{w}_e$). The constraint in [Eq. \(8\)](#) ensures that the value-aligned policy π_* is an optimal policy for the scalarised function $\vec{w}_e \cdot \vec{V}^\pi$ of $\mathcal{M}_*$. Constraint [Eq. \(9\)](#) ensures that the weights in $\vec{w}_e$ are positive, and constraint [Eq. \(10\)](#) ensures that the individual objective's rewards do not change. Moreover, notice that the linear program described by [Eqs. \(7\)–\(10\)](#) always has a feasible solution for an $\epsilon > 0$ small enough: We already know by [Theorem 1](#) that there is a weight vector $\vec{w}_e$ with all $w_i > 0$ (the constraint of [Eq. \(9\)](#)) such that only the policy π_* maximising a given lexicographic order will be optimal (i.e., fulfil the constraint of [Eq. \(8\)](#)). Finally, recall that for any weight vector $\vec{w}_e$, multiplying all its weights w_i by the same constant $\lambda > 0$ does not modify its set of optimal policies. Hence, we can always satisfy the constraint of [Eq. \(10\)](#).

Consider now that the Multi-Valued MDP has a set of initial states S_0 . Then, we must change our LP above as follows. First, we must change the objective function in [Eq. \(7\)](#). For that, we must consider that each initial state $s_i \in S_0$ has a probability p_i of occurring, and therefore minimise the *expectation* of the scalarised value of the initial states: $\mathbb{E}[\vec{w}_e \cdot \vec{V}^{\pi_*}(s_0)]$. Second, we must expand constraint [Eq. \(8\)](#) to ensure that the value-aligned policy π_* will be optimal for each initial state in S_0 , and not only for a specific s_0 .

The LP above contains n decision variables (the n ethical weights of $\vec{w}_e$); and $|S_0| \cdot (|P| - 1)$ constraints.

Example 6. Consider the Multi-Valued MDP $\mathcal{M}$ and the value system $\mathcal{V}_S$ from [Example 5](#). Recall from [Example 4](#) that the partial convex hull of $\mathcal{M}$ is:

$$CH(\mathcal{M}) = \{\vec{V}^{\pi_1}, \vec{V}^{\pi_2}, \vec{V}^{\pi_3}, \vec{V}^{\pi_4}\} = \{(-1, 5, 4), (8, 1, -2), (8, 4, 3), (2, 5, 3)\}.$$

Recall also from [Example 5](#) that its ethical policy is π_3 such that it achieves vectorial value $\vec{V}^{\pi_3}(s_0) = (8, 4, 3)$. Then, we can solve the Linear Problem of [Eqs. \(7\)–\(10\)](#) to find a solution weight vector. Instanced for our case, we need to solve the following linear program (LP):

$$\begin{aligned} \text{Min. } & w_1 \cdot 8 + w_2 \cdot 4 + w_3 \cdot 3 \\ \text{s.t. } & w_1 \cdot 8 + w_2 \cdot 4 + w_3 \cdot 3 \geq w_1 \cdot (-1) + w_2 \cdot 5 + w_3 \cdot 4 + 0.1 \\ & w_1 \cdot 8 + w_2 \cdot 4 + w_3 \cdot 3 \geq w_1 \cdot 8 + w_2 \cdot 1 + w_3 \cdot (-1) + 0.1 \\ & w_1 \cdot 8 + w_2 \cdot 4 + w_3 \cdot 3 \geq w_1 \cdot 2 + w_2 \cdot 5 + w_3 \cdot 3 + 0.1 \\ & w_i \geq 0.1 \forall i, \\ & w_1 = 1. \end{aligned}$$

This LP has as its exact solution the weight vector $\vec{w}_e = (1, 0.1, 0.1)$. We can easily check that indeed with such weight vector the only optimal policy is π_3 :

$$\begin{aligned} \vec{w}_e \cdot \vec{V}^{\pi_1}(s_0) &= 1 \cdot (-1) + 0.1 \cdot 5 + 0.1 \cdot 4 = -0.1, \\ \vec{w}_e \cdot \vec{V}^{\pi_2}(s_0) &= 1 \cdot 8 + 0.1 \cdot 1 + 0.1 \cdot (-2) = 7.9, \\ \vec{w}_e \cdot \vec{V}^{\pi_3}(s_0) &= 1 \cdot 8 + 0.1 \cdot 4 + 0.1 \cdot 3 = 8.7, \\ \vec{w}_e \cdot \vec{V}^{\pi_4}(s_0) &= 1 \cdot 2 + 0.1 \cdot 5 + 0.1 \cdot 3 = 2.8. \end{aligned}$$

Thus, such $\vec{w}_e$ is the solution weight vector with which we can design an ethical environment wherein π_3 will be the optimal policy.

4.4. An algorithm for designing ethical environments

We now have all the tools required to solve [Problem 1](#), hence building an environment wherein an agent will be incentivised to learn a value-aligned policy. [Algorithm 1](#) implements the ethical embedding process and receives as an input both a Multi-Valued MDP $\mathcal{M}$ and its corresponding value system $\mathcal{V}_S$. Then, it starts in line 1 by computing the partial convex hull $P \subseteq CH(\mathcal{M})$ of the input $\mathcal{M}$; and then in line 2 it obtains the value-aligned policy π^* out of those in the partial convex hull P . After that, in line 3, our weighting process searches, within P , for an ethical weight vector $\vec{w}_e$ that solves the Linear Program in [Eqs. \(7\)–\(10\)](#) (see [Section 4.3](#)). For the obtained weight vector $\vec{w}_e$, all optimal policies of the single-objective MDP $\mathcal{M}_* = \langle S, \mathcal{A}, \vec{w}_e \cdot \vec{R}, T \rangle$ are ethical. In other words, such a weight vector solves the ethical embedding problem ([Problem 1](#)). Finally, it returns the MDP $\mathcal{M}_*$ in line 4.

Algorithm 1 Ethical embedding.

Input: Multi-Valued MDP $\mathcal{M} = \langle S, \mathcal{A}, \vec{R}, T \rangle$, $\mathcal{V}_S = \langle \mathcal{V}, \succeq \rangle$

- 1: Compute $P \subseteq CH(\mathcal{M})$ the positive partial convex hull of $\mathcal{M}$.
 - 2: Extract π_* the value-aligned policy within P by computing Equation (6) according to the ordering $\succeq$ in $\mathcal{V}_S$.
 - 3: Find a value for $\vec{w}_e$ that solves the Linear Problem of Equations (7)–(10).
 - 4: **return** ethical MDP $\mathcal{M}_* = \langle S, \mathcal{A}, \vec{w}_e \cdot \vec{R}, T \rangle$.
-

The computational cost of the algorithm mainly resides in computing the partial convex hull of an MOMDP. The Convex Hull Value Iteration algorithm requires $O(n \cdot \log K)$ times what its single-objective counterpart [\[38,55\]](#) requires, where K is the number of policies in the convex hull. In our case, this number will be $k' \leq K$ since we are only computing the positive half of the convex hull. Notice that the second step of our algorithm, solving [Eq. \(6\)](#), is a sorting operation because we already have calculated $\vec{V}^\pi$ for every $\pi \in P$. Finally, the third step involves solving an LP with $|S_0| \cdot (k' - 1)$ constraints.

4.4.1. Algorithm properties

We conclude by offering some relevant properties of our algorithm. First and foremost, notice that [Algorithm 1](#) is always guaranteed to design an ethical environment where value-aligned policies are incentivised. We provide the formal proof in next [Theorem 3](#):

Theorem 3. *Let $\mathcal{M}$ be a finite multi-valued MDP, and $\mathcal{V}_S$ a value system. Consider the single-objective MDP $\mathcal{M}_*$ resulting from applying [Algorithm 1](#) to $\mathcal{M}$ and $\mathcal{V}_S$. Assume that step 1 of [Algorithm 1](#) has been computed with an algorithm guaranteed to converge to the convex hull of $\mathcal{M}$. Then, all the optimal policies π_* of $\mathcal{M}_*$ are value-aligned policies of $\mathcal{M}$ and vice versa.*

Proof. First, if we have computed the correct partial convex hull of $\mathcal{M}$, we will obtain from it the exact value vector of value-aligned policies of $\mathcal{M}$ (step 2 of [Algorithm 1](#)).

Thus, step 3 will compute the weight vector $\vec{w}_e$ for which only the ethical policies of $\mathcal{M}$ achieve the maximum scalarised value by solving the Linear Problem of [Eqs. \(7\)–\(10\)](#). In other words, for weight vector $\vec{w}_e$ the only policies that will maximise its value will be the value-aligned policies (all of them since they share the same value vector). $\square$

Second, it is essential to remark that the convex hull contains all optimal policies for any given lexicographic ordering ([Theorem 1](#)). Thus, if there were changes in the order of values of a given value system $\mathcal{V}_S$, our algorithm would only need to re-compute steps 2 and 3.

A third and final important property of our algorithm is that it works no matter the disparities between the different ranges of the multiple reward functions. In fact, if our [Algorithm 1](#) has computed the ethical weight vector $\vec{w}_e$ for a given Multi-Valued MDP $\mathcal{M}$, we can immediately obtain the ethical weight vector for any linear transformation of any of its reward functions. Formally:

Theorem 4. *Let $\mathcal{M} = \langle S, \mathcal{A}, (R_1, \dots, R_{n-1}, R_a), T \rangle$ be a Multi-Valued MDP and let $\mathcal{V}_S = \langle \mathcal{V}, \succeq \rangle$ be a value system. Consider the weight vector $\vec{w}_e = (w_1, \dots, w_{n-1}, 1)$ resulting from applying [Algorithm 1](#) to $\mathcal{M}$ and $\mathcal{V}_S$.*

Now consider an alternative MOMDP $\mathcal{M}'$ in which the only difference is that either:

- *We have $R'_i = \alpha \cdot R_i$ with $\alpha > 0$ being any positive real number, and some $R_i \neq R_a$. Then, for input environment $\mathcal{M}'$, and the same value system $\mathcal{V}_S$ from before, [Algorithm 1](#) would compute the weight vector*

$$\vec{w}'_e = (w_1, \dots, \frac{1}{\alpha} w_i, \dots, w_{n-1}, 1). \quad (11)$$

- *We have $R'_a = \alpha \cdot R_a$ with $\alpha > 0$ being any positive real number. Then, for input environment $\mathcal{M}'$, and the same value system $\mathcal{V}_S$ from before, [Algorithm 1](#) would compute the weight vector*

$$\vec{w}''_e = (\alpha \cdot w_1, \dots, \alpha \cdot w_{n-1}, 1). \quad (12)$$

Proof. For any policy π in the original environment $\mathcal{M}$, if it achieves value V^π , in the modified environment $\mathcal{M}'$ it will achieve exactly the value $\vec{V}'$ such that $V'_i = \alpha V_i$ for some objective i , and $V'_j = V_j$ for all the rest of objectives.

Moreover, it is clear that all possible environments $\mathcal{M}'$ would have the same set of value-aligned policies π_* as the original environment $\mathcal{M}$, since the lexicographic order is insensitive to reward ranges.

Similarly, notice that any policy belonging to the convex hull of $\mathcal{M}$ directly belongs to the convex hull of $\mathcal{M}'$ and vice-versa.

Then, recall that the weight vector $\vec{w}_e$ was sufficient to guarantee that all value-aligned policies π_* become optimal when scalarising the original environment $\mathcal{M}$ with $\vec{w}_e$. Thus, for the modified environment $\mathcal{M}'$, the alternative weight vector $\vec{w}'_e$ in which we transform the scales of the modified objective back to the original scale would suffice: $\vec{w}'_e = (w_1, \dots, \frac{1}{\alpha} w_i, \dots, w_{n-1}, 1)$.

If the modification was performed in the achievement reward function R_a , recall that Constraint [Eq. \(10\)](#) of our [Algorithm 1](#) requires $w_a = 1$. Thus, instead, we multiply all other objectives by the same α to obtain the desired weight vector $\vec{w}''_e = (\alpha \cdot w_1, \dots, \alpha \cdot w_{n-1}, 1)$. $\square$

5. Experimental analysis

The purpose of this section is three-fold: (i) to illustrate our process for designing an ethical environment (see [Fig. 1](#)); (ii) to illustrate the sensitiveness of the learnt policy to the value system of choice; and (iii) to perform an empirical analysis of the computational cost to pay to design ethical environments. Due to the lack of benchmark multi-objective reinforcement learning environments that consider multiple ethical objectives, we propose a novel (and simple) autonomous car environment. This environment is inspired by the work by Caballero et al. [[27,28](#)], which offers a complete taxonomy of ethical objectives in the autonomous car domain and study the problem from a non-sequential decision-making perspective without learning. For our environment, we consider the moral values of safety, achievement, and comfort to illustrate our approach and the sensitivity of the learnt policy to different value priorities. However, the proposed taxonomy in Caballero et al. tells us that a value-aligned agent must potentially deal with multi-objective environments considering more than three values. Thus, to analyse the cost of applying our ethical embedding process to environments with large value systems, we resort to the WalkRoom synthetic environment [[30](#)], one of the few many-objective⁶ environments in the reinforcement learning literature [[52](#)].

The remainder of this section is organised as follows. [Section 5.1](#) illustrates the application of our Ethical Embedding algorithm to the autonomous car environment. Then, [Section 5.2](#) analyses the computational cost of guaranteeing ethical learning in a synthetic many-objective environment, where each objective corresponds to a value.

5.1. Multi-valued autonomous car environment

The Multi-Valued Autonomous Car environment is broadly characterised as a two-lane road wherein an autonomous car (the learning agent) drives towards its destination. [Fig. 2\(a\)](#) depicts the initial state of the environment, where the autonomous car is represented as the black circle with a C, and the goal is shown as the two cells with an X. The road features bumps (depicted as purple squares) and pedestrians (depicted as black circles with a P) that the car may encounter. Pedestrians can only walk over red cells (representing walkable streets that the car cannot occupy) or blue cells (representing crosswalks) on the map. The autonomous car can drive over grey and blue cells.

In this environment, we consider a value system with three values:

⁶ Many-objective environments refer to environments with more than three objectives.

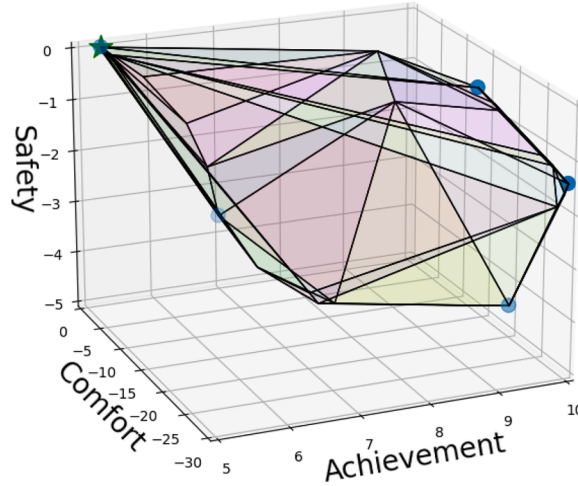


Fig. 3. Convex hull of the Multi-Valued Autonomous Car Environment. All optimal policies for at least one lexicographic ordering are highlighted with a blue dot or a green star. Notice that the convex hull is zoomed in: achievement axis starts at 5. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

systems arising from the six different orders $\succeq^i$, with $i = 1, \dots, 6$, plus a value system that only considers achievement v_a . Notice, however, that Definition 6 explicitly forbids aligning agents with value systems where the value of achievement is prioritised first. However, for the sake of completeness and to show the ethical problems of prioritizing achievement first, we have also evaluated these value systems (which are $\mathcal{V}_S^5$, $\mathcal{V}_S^6$, and $\mathcal{V}_S^7$). We now make explicit each value system $\mathcal{V}_S^i = \{v_a, v_c, v_s\}, \succeq^i$:

1. $\mathcal{V}_S^1$, with ordering $v_s \geq v_c \geq v_a$: safety prioritised over comfort, and comfort over achievement. An agent aligned with this value system will first avoid running over pedestrians and, second, try to avoid as many bumps as possible (speed is third).
2. $\mathcal{V}_S^2$, with ordering $v_s \geq v_a \geq v_c$: safety prioritised over achievement, and achievement over comfort. An agent aligned with this value system will try to follow the route that avoids accidents and dangerous driving while reaching its destination as fast as possible (comfort is last considered).
3. $\mathcal{V}_S^3$, with ordering $v_c \geq v_s \geq v_a$: comfort prioritised over safety, and safety over achievement. An agent aligned with this value system will prioritise avoiding bumps in the first place (and safety second and speed third).
4. $\mathcal{V}_S^4$, with ordering $v_c \geq v_a \geq v_s$: comfort prioritised over achievement, and achievement over safety. An agent aligned with this value system will try to reach its goal following the route that avoids all bumps as fast as possible (and safety is last considered).
5. $\mathcal{V}_S^5$, with ordering $v_a \geq v_s \geq v_c$ prioritises achievement over safety, and safety over comfort. A car aligned with this value system will drive as fast as possible to its destination. From all the possible journeys that minimise the time spent to reach its goal, it will select the safest one (that is, as comfortable as possible).
6. $\mathcal{V}_S^6$, with ordering $v_a \geq v_c \geq v_s$ prioritises achievement over comfort, and comfort over safety. This value system encourages the car to run as fast as possible to its destination. From all the possible journeys that minimise the time spent to reach its goal, it will select the one with the maximum comfort that achieves as much safety as possible.
7. $\mathcal{V}_S^7$, with ordering v_a : the only value in this case is achievement, which is the agent's own objective. An agent aligned with this value system will run as fast as possible to its destination regardless of any other consideration.

5.1.2. Building an ethical environment for a value system

This section shows how our ethical embedding algorithm designs an ethical environment for $\mathcal{M}$. The procedure is the same regardless of which of the seven value systems we select. In particular, to illustrate the steps of the ethical embedding process, we select the value system $\mathcal{V}_S^1$ with ordering $v_s \geq v_c \geq v_a$ because it prioritises safety –its higher ethical standard– over achievement. Thus, as previously mentioned, it incentivises learning an ethical behaviour. We now apply Algorithm 1 to design an ethical environment $\mathcal{M}_*$ for the Multi-Valued Autonomous Car Environment. Next, we detail the three steps to obtain an ethical environment.

1. **Partial convex hull computation:** Considering the Multi-Valued MDP $\mathcal{M}$, we compute its partial convex hull $P \subseteq CH(\mathcal{M})$ by applying Convex Hull Value Iteration. Recall from Algorithm 1 that for this step we can apply any algorithm that computes the convex hull of an MOMDP (for instance, Convex Hull Value Iteration (CHVI) or Optimistic Linear Support) [25]. Fig. 3 depicts the resulting P for the initial state s_0 .
2. **Extraction of the ethical policy:** We order the policies of P by following the value system $\mathcal{V}_S^1$ with ordering $v_s \geq v_c \geq v_a$ and pick the first policy as the ethical policy π_* , highlighted with a green star in Fig. 3. Notice that π_* is the only policy that maximises both safety and comfort $\vec{V}^{\pi_*} = (V_a, V_c, V_s) = (5.283, 0, 0)$ because when the agent follows π_* it is capable of reaching its destination without running over any pedestrian nor over any bump, and thus, it avoids receiving any negative comfort reward (R_c) or safety reward (R_s).

Table 1

Table of the different ethical weightings computed by the Multi-Valued Ethical Embedding algorithm depending on which value system is selected for the Autonomous Car environment. The fifth column presents the vectorial value of the policy that maximises each respective weight vector. Each policy is also the most aligned with each respective value system. All values are rounded to the nearest thousandth. For all vectors in columns 4 and 5, recall that their dimensions follow this order: first dimension is achievement, second dimension is comfort, and third dimension is safety.

Value system $\mathcal{V}_S^i$	Ordering	Ethical	Weight vector $\vec{w}_e = (w_a^i, w_c^i, w_s^i)$	Value-aligned $\vec{V}^{\pi_i} = (V_a^{\pi_i}, V_c^{\pi_i}, V_s^{\pi_i})$
$\mathcal{V}_S^1$	$v_s \geq v_c \geq v_a$	Yes	(1, 1.370, 5.540)	(5.283, 0.0, 0.0)
$\mathcal{V}_S^2$	$v_s \geq v_a \geq v_c$	Yes	(1, 0.003, 1.240)	(9.27, -22.5, 0.0)
$\mathcal{V}_S^3$	$v_c \geq v_s \geq v_a$	Yes	(1, 1.370, 5.540)	(5.283, 0.0, 0.0)
$\mathcal{V}_S^4$	$v_c \geq v_a \geq v_s$	No	(1, 0.505, 0.001)	(7.0, 0.0, -3.62)
$\mathcal{V}_S^5$	$v_a \geq v_s \geq v_c$	No	(1, 0.027, 0.093)	(10.0, -30.0, -1.41)
$\mathcal{V}_S^6$	$v_a \geq v_c \geq v_s$	No	(1, 0.097, 0.001)	(10.0, -20.0, -4.48)
$\mathcal{V}_S^7$	v_a	No	(1, 0, 0)	(10.0, -25.0, -2.95)

3. **Computation of the scalarisation function:** Line 3 in Algorithm 1 computes the weight vector $\vec{w}_e$ for which π_* is the only optimal policy of P , by solving the linear program in Eqs. (7)–(10). This amounts to solving a linear program in which the objective function is to minimise the scalarised value of the ethical policy. By solving it, we find that if $\vec{w}_e = (1, 1.37, 5.54)$, then π_* becomes the only optimal policy in $\mathcal{M}_*$. Hence, Algorithm 1 returns a single objective MDP $\mathcal{M}_*$ whose scalarised reward $\vec{w}_e \cdot \vec{R}$ comes from scalarising the Multi-Valued MDP by $\vec{w}_e$. Such weight vector $\vec{w}_e$ solves the Autonomous Car environment. More specifically, an MDP created from a scalarisation function with such ethical weight vector $\vec{w}_e$ incentivises an agent to learn the ethical policy π_* , aligned with $\mathcal{V}_S^1$.

Repeating the ethical embedding process for alternative value systems leads to alternative value-aligned policies within the convex hull that require a different weight vector $\vec{w}_e$ in the resulting $\mathcal{M}_*$. Table 1 shows the results for the value systems $\mathcal{V}_S^1, \dots, \mathcal{V}_S^7$ previously defined. In particular, it details the $\vec{w}_e$ used to incentivise the agent to behave value-aligned and the vectorial value accumulating the optimal policy. Notice that we obtain the same ethical weight vector and ethical policy for value systems $\mathcal{V}_S^3$ and $\mathcal{V}_S^1$. This is so because, for this specific setting, the policy can be maximally aligned with safety and comfort simultaneously (i.e., the agent can avoid all pedestrians and bumps).⁷

5.1.3. Learning in an ethical environment

We guarantee that for each environment designed with our ethical embedding process, the optimal policy is the most aligned with its respective value system. Considering again the value system $\mathcal{V}_S^1$ with ordering $v_s \geq v_c \geq v_a$, we designed for it an ethical environment $\mathcal{M}_*$ by applying the ethical weight vector (1, 1.37, 5.54) (see Table 1). As expected, an agent converges to the optimal policy π_* when learning within the ethical environment $\mathcal{M}_*$, even if it applies a basic single-objective reinforcement learning algorithm such as tabular Q-learning.

Fig. 4 plots the rewards per episode accumulated by a Q-learning agent, which stabilises at $\vec{V}^{\pi_*} = (5.283, 0, 0)$, the values of ethical policy π_* associated with $\mathcal{V}_S^1$ (first row in Table 1). For reference, the policy that only maximises the individual objective v_a obtains a value of (10, -25, -2.95), as shown in Table 1. Thus, by behaving ethically, the agent completely eliminates the accumulation of negative rewards in terms of safety and comfort (from -25 to 0 and from -2.95 to 0) at the cost of decreasing its individual rewards (from 10 to 5.283).

5.1.4. Comparing value-aligned policies

This subsection aims to analyse the consequences of each value system in Table 1 to determine which would be the most appropriate for a real-world autonomous car. Thus, we compare the behaviour of the seven policies $\pi_1, \dots, \pi_7$ in Table 1 in terms of four ethical metrics. These four ethical metrics empirically quantify the degree of alignment of each policy with the three values in our value system. In particular, for each of the seven environments that result from applying the embedding process considering the value systems in Table 1, we trained an agent with Q-learning. We then evaluated the learnt policies according to the following four ethical metrics. Each ethical metric is associated with a moral value of the value system, so the lower the value of the metric, the higher the value alignment. That way, we can empirically measure the degree of alignment of each policy to each value. The four ethical metrics are:

- **Time steps to destination:** measures the number of time steps the agent requires to reach its destination, fulfilling its individual objective. This metric is associated with the value of achievement v_a .

⁷ Notice also that the value of the policy aligned with $\mathcal{V}_S^7$ is an average of the two policies that prioritise v_a as its most important value (i.e., those aligned with $\mathcal{V}_S^6$ and $\mathcal{V}_S^5$, respectively).

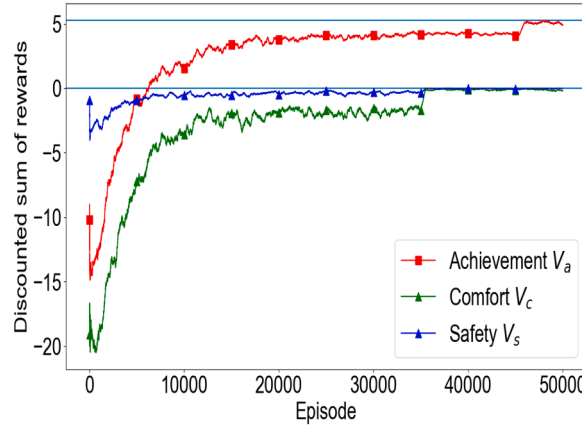


Fig. 4. Evolution of the accumulated rewards per episode that the learning agent obtains in the ethical environment with value system $\mathcal{V}_S^1$ with ordering $v_s \geq v_c \geq v_a$. Horizontal straight lines mark the corresponding convergence values in the first row of Table 1.

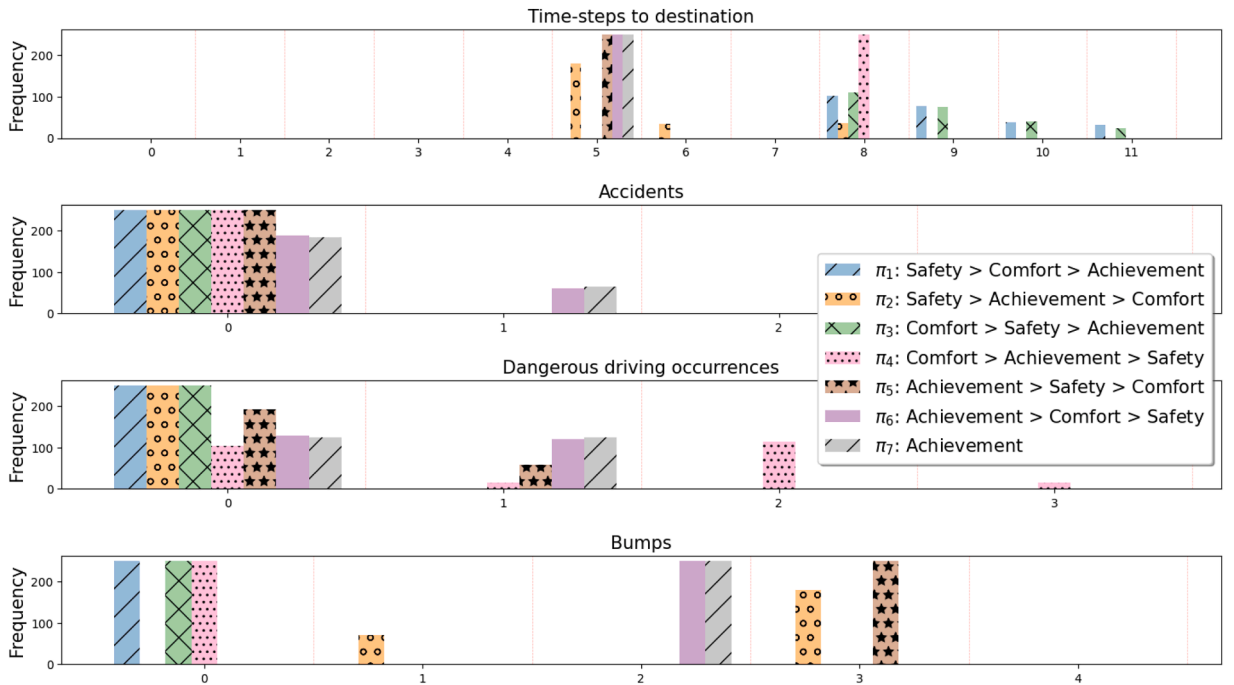


Fig. 5. Bar graphs evaluating the seven different value-aligned policies for the Multi-Valued Autonomous Car environment. Each policy π_i is learned to be aligned with each value system in Table 1: from $\mathcal{V}_S^1$ ($v_s \geq v_c \geq v_a$, displayed as “Safety > Comfort > Achievement”) to $\mathcal{V}_S^7$ (v_a , displayed as “Achievement”). Each row presents the results for the four ethical metrics: Time to destination, Accidents, Dangerous driving, and Bumps. The x-axis is the possible discrete values of each metric, and the y-axis is the number of times, from the 250 executions, that each value is registered.

- **Accidents:** the number of pedestrians the agent runs over. This metric is associated with the safety value and amounts to how often the agent receives a negative safety reward of $R_s = -10$.
- **Dangerous driving occurrences:** number of times the agent drives too close to pedestrians. This metric is also associated with the value of safety and amounts to how often the agent receives a negative safety reward of $R_s = -3$.
- **Bumps:** number of times the agent crossed any of the three bumpy areas. This metric is associated with the comfort value and amounts to how often the agent receives a negative comfort reward of $R_c = -10$.

We evaluated the learnt value-aligned policies for each of the seven value systems in 250 episodes. We measured the four metrics per episode. Since the Autonomous Car Environment is stochastic (in the behaviour of pedestrians), we simulated 250 episodes per policy for more robust conclusions. Fig. 5 presents the bar graph of all seven evaluated policies. Notice that since Fig. 5 is a bar graph, there are more than seven columns per graph, even though we evaluated seven policies. For instance, policy π_6 with ordering

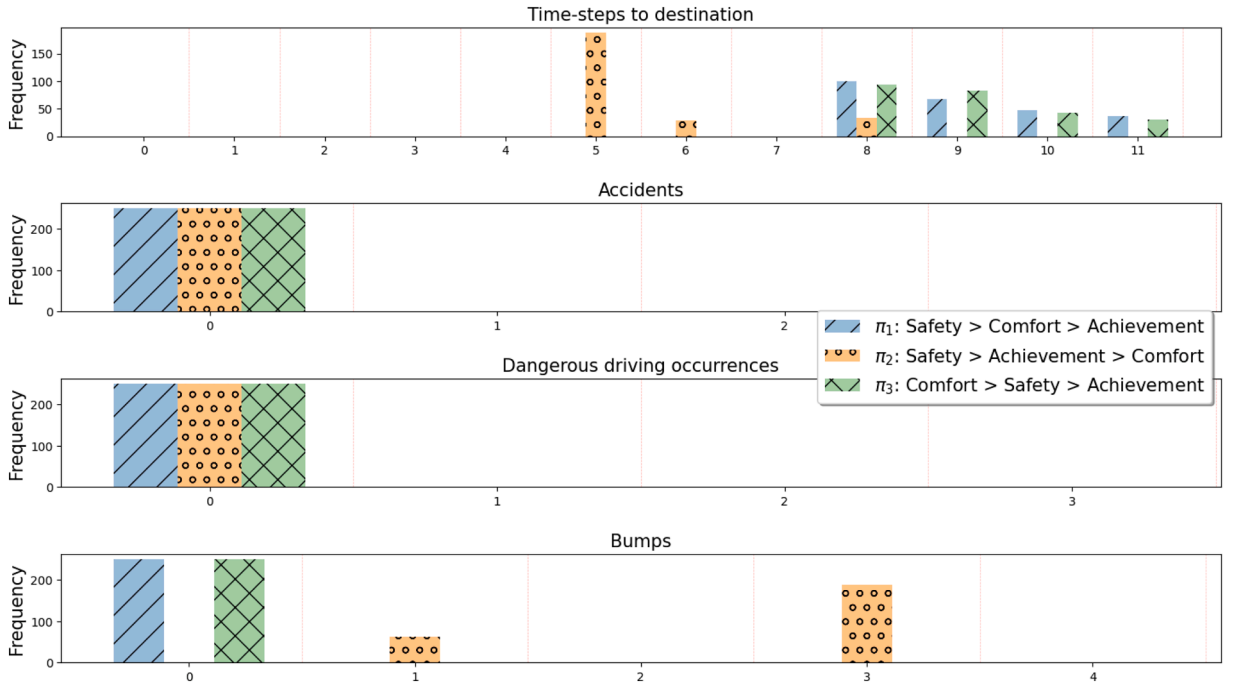


Fig. 6. Bar graphs of π_1, π_2 , and π_3 , the three different ethical policies in Fig. 5: they are aligned with value systems $\mathcal{V}_S^1 (v_s \geq v_c \geq v_a)$, displayed as “Safety > Comfort > Achievement”, $\mathcal{V}_S^2 (v_s \geq v_a \geq v_c)$, and $\mathcal{V}_S^3 (v_c \geq v_s \geq v_a)$. Each subsequent bar graph presents the results for each of the four ethical metrics: Time to destination, Accidents, Dangerous driving, and Bumps.

$v_a \geq v_c \geq v_s$ there were no accidents in 195 episodes, but for 55 episodes there is at least 1 accident (hence, its column appears twice in second row of Fig. 5). Moreover, to discuss the results presented in Fig. 5, we divide the seven policies into two categories:

- **Ethical policies:** the three policies associated with a value system that prioritises safety over achievement. They satisfy our definition of being ethical because safety is a higher ethical standard than achievement.
- **Unethical policies:** the four remaining policies, where achievement is prioritised over safety.

For the sake of simplicity, we refer to these policies as being ethical or unethical, but we may also refer to them as *virtuous behaviour* (if they are aligned with value systems that prioritise higher ethical standards over achievement) or as *achievement-driven behaviour* (if achievement is prioritised instead). This richer terminology would emphasise the fact that value-aligned behaviours can exhibit different degrees of ethicality.⁸ However, it may also hinder the ethical superiority of virtuous behaviour, which may fatally lead us to use our value-alignment process for ethics-washing purposes. Therefore, we opt for the ethical and unethical terminology, as in our autonomous vehicle case study, any autonomous car not prioritising human life should be considered unethical even if it is value-aligned. Indeed, the work in [58] goes a step forward in this direction and explicitly distinguishes good (such as prudence or generosity) from bad values (such as greed or selfishness).

Now, we proceed to discuss the first category, the ethical policies. Afterwards, we analyse the unethical policies.

Ethical policies. Three ethical policies prioritise safety over achievement: π_1, π_2 , and π_3 , the ones associated with value systems $\mathcal{V}_S^1$ (with $v_s \geq v_c \geq v_a$), $\mathcal{V}_S^2$ (with $v_s \geq v_a \geq v_c$), and $\mathcal{V}_S^3$ (with $v_c \geq v_s \geq v_a$), respectively. Ethical policies occupy the first three rows of Table 1.

Empirically, Fig. 6 shows that none of the three ethical policies accumulates accidents or dangerous driving occurrences. In other words, all three ethical policies obtain the optimal value (0) regarding the two safety-related metrics. Moreover, as π_1 and π_3 prioritise achievement the least, they obtain the optimal value (0) for both the safety-related and comfort metrics, meaning they are completely aligned with both values. Only π_2 traverses bumps because it prioritises the comfort value the least.

Of course, prioritising safety over achievement has a cost in terms of the time steps needed to reach the destination. As expected, Fig. 6 shows that two of the three ethical policies require, on average, more than eight time steps to reach their destination. Interestingly, the ethical policy π_2 giving the value of comfort ($v_s \geq v_a \geq v_c$) the lowest priority, reaches the destination significantly faster than all the rest of the ethical policies, as the first bar graph of Fig. 6 illustrates. Indeed, the agent does not care about following the

⁸ The Ethics literature defines the idea of bounded ethicality [56] to highlight the restrictions to make ethical choices. For example, self-serving bias [57] often causes us to subconsciously favour ourselves at the expense of others.

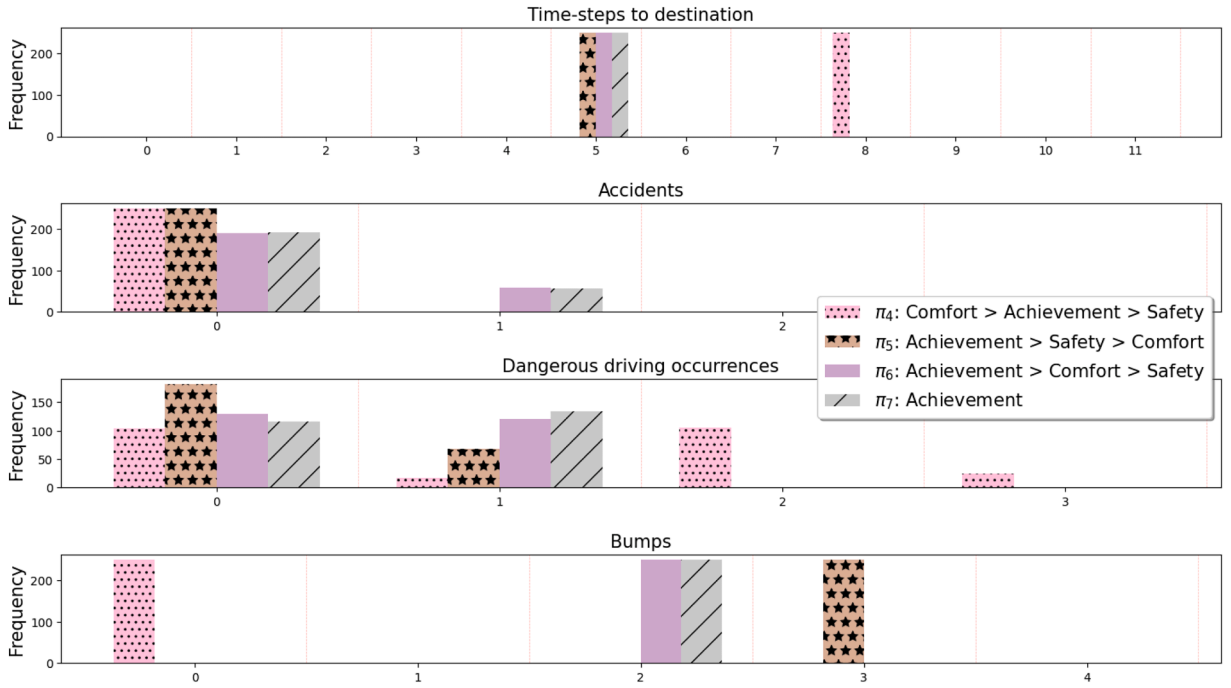


Fig. 7. Bar graphs of $\pi_4, \dots, \pi_7$, the four unethical policies in Fig. 5 that prioritise the moral value of achievement over safety they are aligned with value systems $\mathcal{V}_S^4$ ($v_c \geq v_a \geq v_s$, displayed as “Comfort > Achievement > Safety”), $\mathcal{V}_S^5$ ($v_a \geq v_s \geq v_c$), $\mathcal{V}_S^6$ ($v_a \geq v_c \geq v_s$), and $\mathcal{V}_S^7$ (v_a , simply displayed as “Achievement”). Each subsequent bar graph presents the results for the four metrics: Time to destination, Accidents, Dangerous driving, and Bumps.

most bumpy trajectory, shown in the fourth bar graph of Fig. 6, as long as it reaches its destination as fast as possible while avoiding dangerous driving situations and accidents. However, compared to unethical policies, π_2 can only surpass the slowest unethical policy regarding speed.

Unethical policies. Four unethical policies $\pi_4, \dots, \pi_7$ correspond to rows 4 to 7 of Table 1. All of them prioritise achievement over safety. Empirically, Fig. 7 shows that prioritising achievement has a cost: in all cases, none of the four unethical policies manages to avoid dangerous driving situations, unlike the policies shown in Fig. 6.

Regarding their results in terms of achievement, as expected, Fig. 7 shows that amongst the four policies, π_5, π_6 , and π_7 always allow the agent to reach its destination in five time-steps, the fastest possible route. No ethical policy can reach the destination as consistently fast, as shown in Fig. 5. Of course, this difference occurs because achievement is the most prioritised value for these three policies. Only the unethical policy π_4 prioritising comfort exhibits a slower behaviour, requiring eight time steps. Still, π_4 is faster than all ethical policies but π_2 .

Regarding comfort, we also appreciate the differences between the four unethical policies. π_3 prioritises comfort over the rest of values and completely avoids bumps. Instead, π_6 traverses some bumps, though less than the other two unethical policies.

Finally, as to safety, the unethical policies π_4 ($v_c \geq v_a \geq v_s$) and π_5 ($v_a \geq v_s \geq v_c$) follow journeys that avoid accidents. However, π_4 is the one accumulating more dangerous driving occurrences. Finally, the other two unethical policies (i.e., π_6 and π_7) accumulate accidents and dangerous driving occurrences.

Summary. Our analysis above allows us to conclude that the ordering of the values in a value system can drastically affect the policy learnt by an agent. Fig. 5 shows that six of the seven policies present a unique behaviour, with two (π_6 and π_7) having fatal consequences. Thus, it is of uttermost importance to carefully decide on the appropriate value system to enact in real-world applications and not only on the values to consider.

Moreover, we empirically observed that the most prioritised value in the value system plays a determining role in the learnt behaviour of the agent. In general, if two policies prioritise the most the very same moral values, they exhibit the same behaviour in terms of alignment with that value. Here, we observed that all policies prioritising the most achievement (v_a) followed the fastest route to the destination, all policies prioritising the most comfort (v_c) avoided bumps, and policies that prioritise safety (v_s) safely drove away from pedestrians. Ultimately, we must always incentivise to learn ethical policies.

5.2. Computational cost of the ethical embedding process

As previously mentioned, *many*-objective environments (i.e., multi-objective environments with more than three objectives) hardly exist in the literature. However, in the autonomous car scenario from Caballero et al. [27], many more values are at hand. Thus, to analyse the cost of applying our ethical embedding process to environments with an increasing number of values, we resort to a synthetic many-objective environment from [30] called WalkRoom.

Walkroom models an n -dimensional grid world where an agent can move in any of the n dimensions towards several goal positions. It can be instantiated with an arbitrary number of dimensions and an arbitrary grid size per dimension. Any time the agent moves along dimension i , it receives a penalty for objective i . Thus, the dimensions correspond to the number of objectives in the environment.

We adapt this environment by adding a value system $\mathcal{V}_S$ with as many values as dimensions. Thus, the agent's objective is to find the goal position which requires minimal movement alongside the most prioritised objectives according to $\mathcal{V}_S$. The agent's individual objective in this environment, associated with the achievement value v_a , is arbitrarily selected as dimension 1. Notice, however, that all results in this section would be the same no matter which objective was chosen as the individual objective of the agent due to the environment specifications that we further detail in Section 5.2.1.

5.2.1. Multi-valued MDP and algorithmic specification

Given a value system $\mathcal{V}_S = \{v_1, \dots, v_n\}$, $\geq$ (see Definition 5 with n values, we define a finite Multi-Valued MDP $\mathcal{M}$ (see Definition 6) for the Walkroom environment with the corresponding vectorial reward function $\vec{R} = (R_1, \dots, R_n)$, with each reward corresponding to one of the values of the value system. Now we specify each element of the Multi-Valued MDP tuple $\langle S, \mathcal{A}, \vec{R}, T \rangle$:

- **States:** States are fully observable by the agent and contain its own position. Assuming a grid size of k and n moral values, then the environment has the shape of a hypercube of n dimensions divided into k^n states.
- **Actions:** There are twice as many actions as values. Each value v_i has two associated actions i and $-i$ that move the agent along the i dimension forward or backwards, respectively. For example, an agent in position $(5, 4, 3, 1)$ that selects action -2 associated with v^2 then moves in dimension 2 by $(0, -1, 0, 0)$ to position $(5, 3, 3, 1)$.
- **Rewards:** Any time the agent moves along dimension i , it receives a penalty $R_i = -1$ associated with value v_i .
- **Transitions:** the environment is fully deterministic. There are several goal positions randomly generated within the environment. The agent reaching any of them is in a terminal state.

For computing the ethical embedding process to the WalkRoom environment, we applied a parallelised version of CHVI [38] to compute its convex hull, which is necessary for our embedding. Our parallelised version exploits the fact that in CHVI, at each iteration, we can update the value function of each state independently (and this fact remains valid for any possible MOMDP).

5.2.2. Experimental setup and results

We evaluated the ethical embedding algorithm in a set of randomly generated Walkroom environments, from 2 to 9 objectives, and with a varying grid size per dimension from 2 to 9. Recall that the number of states in the environment is determined as $(s, n) = s^n$ with s being the grid size and n the number of values (hence, the number of states is $\text{grid_size}^{\text{values}}$), so in the case $(9, 9)$ we are dealing with an environment with more than 380 million states, a very challenging environment for any tabular reinforcement learning algorithm. We tested each configuration (s, n) for 20 different seeds (i.e. for a total $8 \cdot 8 \cdot 20 = 1280$ experiments). We performed all our experiments on a cluster whose computing nodes have 64-core 2.20GHz CPUs and 32GB RAM, allowing us to fully exploit the parallelism of our CHVI implementation.⁹ The heat maps in Fig. 8 contain the average times required to compute the ethical embedding for each of the 64 possible configurations of the synthetic environment.

The heat maps show that the computational cost of our algorithm exponentially increases with the grid size per objective, and such increase is amplified with the number of objectives in the environment. This is expected due to the computational cost of CHVI. In more detail, we find in Fig. 8 (Left) how, for a third of the settings, the required amount of time is at most 10 s (e.g., any configuration with less than four values, and any configuration with grid size 2). However, Fig. 8 (Right) illustrates how almost all configurations with more than 8 values exceeded the 10-h threshold of computation time.

Table 2 displays some significant cases' average computation times. Those cases exemplify a general pattern: the computational costs of finding an ethical policy (step 2) and computing the solution weight vector (step 3) are negligible compared with computing the convex hull (step 1). For example, see the case with $(s, n) = (6, 7)$, where step 1 takes 705 s, but step 2 takes less than a millisecond, and step 3 takes 1.07 s. In this specific case, 99.85 % of the time is spent in step 1.

As mentioned, we have discarded the environments in the literature for our empirical analysis because they do not go beyond three objectives. Nevertheless, thanks to the obtained results, we can provide rough estimates of how much time our algorithm would need to do ethical embeddings depending on the number of values and their size. To illustrate this point, consider the discrete environments of MO-Gymnasium [59,60], the main multi-objective library. We assume that if our algorithm needs t seconds for WalkRoom with n objectives and a state space S , it needs at most t seconds for an environment with n objectives and less than S states. Thus: For a small state space (less than 500 states) such as *Fruit-tree* (6 objectives) or *Deep Sea Treasure* (2 objectives), our algorithm would need less than 2 s in total (the result from $(s, n) = (3, 6)$, which has $3^6 = 729$ states). For a state space of almost 700,000 states and 3 objectives such as *Four-Room*, our algorithm would need 5 h (result from $(s, n) = (7, 7)$).

⁹ Our CHVI implementation is available at <https://github.com/filippobistaffa/chvi>.

Table 2

Average computational cost (mean $\pm$ 3std in seconds) of each step of the ethical embedding algorithm for particular Walkroom environment configurations (grid size s and n objectives). Cells coloured in green, yellow, orange, and red indicate $10^0 \leq \text{mean} < 10^1$, $10^1 \leq \text{mean} < 10^2$, $10^2 \leq \text{mean} < 10^3$, and $\text{mean} \geq 10^3$, respectively.

(s, n)	Step 1	Step 2	Step 3
(2, 2)	$3.21 \cdot 10^{-2} \pm 9.13 \cdot 10^{-2}$	$1.22 \cdot 10^{-4} \pm 5.97 \cdot 10^{-5}$	$1.16 \cdot 10^0 \pm 5.48 \cdot 10^{-1}$
(2, 3)	$1.13 \cdot 10^{-2} \pm 1.51 \cdot 10^{-3}$	$1.12 \cdot 10^{-4} \pm 1.57 \cdot 10^{-5}$	$9.72 \cdot 10^{-1} \pm 4.75 \cdot 10^{-2}$
(2, 4)	$1.34 \cdot 10^{-2} \pm 2.54 \cdot 10^{-2}$	$1.24 \cdot 10^{-4} \pm 2.19 \cdot 10^{-5}$	$9.68 \cdot 10^{-1} \pm 6.81 \cdot 10^{-2}$
(2, 5)	$4.79 \cdot 10^{-2} \pm 5.02 \cdot 10^{-2}$	$1.46 \cdot 10^{-4} \pm 5.37 \cdot 10^{-5}$	$1.01 \cdot 10^0 \pm 1.86 \cdot 10^{-1}$
(2, 6)	$5.66 \cdot 10^{-2} \pm 3.77 \cdot 10^{-2}$	$1.70 \cdot 10^{-4} \pm 4.63 \cdot 10^{-5}$	$1.06 \cdot 10^0 \pm 1.48 \cdot 10^{-1}$
(2, 7)	$9.64 \cdot 10^{-2} \pm 2.03 \cdot 10^{-1}$	$2.09 \cdot 10^{-4} \pm 9.58 \cdot 10^{-5}$	$1.19 \cdot 10^0 \pm 8.86 \cdot 10^{-1}$
(2, 8)	$9.41 \cdot 10^{-2} \pm 7.96 \cdot 10^{-2}$	$2.22 \cdot 10^{-4} \pm 9.14 \cdot 10^{-5}$	$1.04 \cdot 10^0 \pm 5.08 \cdot 10^{-1}$
(2, 9)	$1.90 \cdot 10^{-1} \pm 1.51 \cdot 10^{-1}$	$2.67 \cdot 10^{-4} \pm 1.60 \cdot 10^{-4}$	$1.00 \cdot 10^0 \pm 2.05 \cdot 10^{-1}$
(3, 2)	$2.71 \cdot 10^{-2} \pm 7.05 \cdot 10^{-2}$	$1.16 \cdot 10^{-4} \pm 5.29 \cdot 10^{-5}$	$1.05 \cdot 10^0 \pm 3.36 \cdot 10^{-1}$
(3, 3)	$4.26 \cdot 10^{-2} \pm 6.82 \cdot 10^{-2}$	$1.21 \cdot 10^{-4} \pm 5.14 \cdot 10^{-5}$	$9.94 \cdot 10^{-1} \pm 2.22 \cdot 10^{-1}$
(3, 4)	$6.66 \cdot 10^{-2} \pm 6.20 \cdot 10^{-2}$	$1.44 \cdot 10^{-4} \pm 5.88 \cdot 10^{-5}$	$9.92 \cdot 10^{-1} \pm 1.95 \cdot 10^{-1}$
(3, 5)	$1.01 \cdot 10^{-1} \pm 8.98 \cdot 10^{-2}$	$1.80 \cdot 10^{-4} \pm 5.52 \cdot 10^{-5}$	$1.02 \cdot 10^0 \pm 3.38 \cdot 10^{-1}$
(3, 6)	$1.13 \cdot 10^{-1} \pm 9.07 \cdot 10^{-2}$	$2.01 \cdot 10^{-4} \pm 7.15 \cdot 10^{-5}$	$1.02 \cdot 10^0 \pm 2.11 \cdot 10^{-1}$
(3, 7)	$5.10 \cdot 10^{-1} \pm 2.00 \cdot 10^{-1}$	$2.52 \cdot 10^{-4} \pm 1.17 \cdot 10^{-4}$	$1.11 \cdot 10^0 \pm 5.09 \cdot 10^{-1}$
(3, 8)	$1.08 \cdot 10^1 \pm 4.00 \cdot 10^0$	$3.78 \cdot 10^{-4} \pm 3.71 \cdot 10^{-4}$	$9.81 \cdot 10^{-1} \pm 1.97 \cdot 10^{-1}$
(3, 9)	$1.78 \cdot 10^2 \pm 9.39 \cdot 10^2$	$2.38 \cdot 10^{-4} \pm 1.80 \cdot 10^{-4}$	$1.06 \cdot 10^0 \pm 2.64 \cdot 10^{-1}$
(4, 2)	$3.38 \cdot 10^{-2} \pm 1.37 \cdot 10^{-1}$	$1.10 \cdot 10^{-4} \pm 4.98 \cdot 10^{-5}$	$1.07 \cdot 10^0 \pm 4.77 \cdot 10^{-1}$
(4, 3)	$6.82 \cdot 10^{-2} \pm 5.06 \cdot 10^{-2}$	$1.31 \cdot 10^{-4} \pm 3.99 \cdot 10^{-5}$	$9.81 \cdot 10^{-1} \pm 1.35 \cdot 10^{-1}$
(4, 4)	$9.08 \cdot 10^{-2} \pm 6.12 \cdot 10^{-2}$	$1.48 \cdot 10^{-4} \pm 4.74 \cdot 10^{-5}$	$9.79 \cdot 10^{-1} \pm 1.57 \cdot 10^{-1}$
(4, 5)	$1.52 \cdot 10^{-1} \pm 1.12 \cdot 10^{-1}$	$1.89 \cdot 10^{-4} \pm 7.43 \cdot 10^{-5}$	$1.02 \cdot 10^0 \pm 2.29 \cdot 10^{-1}$
(4, 6)	$5.92 \cdot 10^{-1} \pm 3.58 \cdot 10^{-1}$	$2.22 \cdot 10^{-4} \pm 1.39 \cdot 10^{-4}$	$1.00 \cdot 10^0 \pm 1.11 \cdot 10^{-1}$
(4, 7)	$1.25 \cdot 10^1 \pm 3.76 \cdot 10^0$	$2.93 \cdot 10^{-4} \pm 1.79 \cdot 10^{-4}$	$1.03 \cdot 10^0 \pm 3.87 \cdot 10^{-2}$
(4, 8)	$5.82 \cdot 10^2 \pm 1.10 \cdot 10^2$	$3.41 \cdot 10^{-4} \pm 2.24 \cdot 10^{-4}$	$1.12 \cdot 10^0 \pm 4.63 \cdot 10^{-1}$
(4, 9)	> 10h	—	—
(5, 2)	$6.42 \cdot 10^{-2} \pm 1.31 \cdot 10^{-1}$	$1.21 \cdot 10^{-4} \pm 6.45 \cdot 10^{-5}$	$1.07 \cdot 10^0 \pm 4.05 \cdot 10^{-1}$
(5, 3)	$8.36 \cdot 10^{-2} \pm 5.09 \cdot 10^{-2}$	$1.32 \cdot 10^{-4} \pm 3.71 \cdot 10^{-5}$	$9.75 \cdot 10^{-1} \pm 7.34 \cdot 10^{-2}$
(5, 4)	$1.21 \cdot 10^{-1} \pm 8.20 \cdot 10^{-2}$	$1.61 \cdot 10^{-4} \pm 6.87 \cdot 10^{-5}$	$1.06 \cdot 10^0 \pm 3.01 \cdot 10^{-1}$
(5, 5)	$2.80 \cdot 10^{-1} \pm 8.44 \cdot 10^{-2}$	$1.87 \cdot 10^{-4} \pm 7.97 \cdot 10^{-5}$	$9.88 \cdot 10^{-1} \pm 2.05 \cdot 10^{-1}$
(5, 6)	$3.16 \cdot 10^0 \pm 3.62 \cdot 10^0$	$2.35 \cdot 10^{-4} \pm 2.01 \cdot 10^{-4}$	$9.82 \cdot 10^{-1} \pm 9.80 \cdot 10^{-2}$
(5, 7)	$1.20 \cdot 10^2 \pm 1.75 \cdot 10^1$	$3.12 \cdot 10^{-4} \pm 1.81 \cdot 10^{-4}$	$1.06 \cdot 10^0 \pm 3.71 \cdot 10^{-1}$
(5, 8)	$7.90 \cdot 10^3 \pm 7.40 \cdot 10^2$	$4.83 \cdot 10^{-4} \pm 4.48 \cdot 10^{-4}$	$1.15 \cdot 10^0 \pm 3.99 \cdot 10^{-1}$
(5, 9)	> 10h	—	—
(6, 2)	$8.18 \cdot 10^{-2} \pm 1.04 \cdot 10^{-1}$	$1.44 \cdot 10^{-4} \pm 3.85 \cdot 10^{-5}$	$1.01 \cdot 10^0 \pm 3.00 \cdot 10^{-1}$
(6, 3)	$9.50 \cdot 10^{-2} \pm 7.01 \cdot 10^{-2}$	$1.28 \cdot 10^{-4} \pm 1.92 \cdot 10^{-5}$	$9.76 \cdot 10^{-1} \pm 6.17 \cdot 10^{-2}$
(6, 4)	$1.41 \cdot 10^{-1} \pm 1.07 \cdot 10^{-1}$	$1.53 \cdot 10^{-4} \pm 6.47 \cdot 10^{-5}$	$9.79 \cdot 10^{-1} \pm 1.36 \cdot 10^{-1}$
(6, 5)	$7.12 \cdot 10^{-1} \pm 1.54 \cdot 10^{-1}$	$2.45 \cdot 10^{-4} \pm 3.36 \cdot 10^{-4}$	$1.14 \cdot 10^0 \pm 3.94 \cdot 10^{-1}$
(6, 6)	$1.45 \cdot 10^1 \pm 1.72 \cdot 10^1$	$2.52 \cdot 10^{-4} \pm 1.88 \cdot 10^{-4}$	$9.80 \cdot 10^{-1} \pm 1.06 \cdot 10^{-1}$
(6, 7)	$7.05 \cdot 10^2 \pm 5.97 \cdot 10^1$	$3.33 \cdot 10^{-4} \pm 2.91 \cdot 10^{-4}$	$1.07 \cdot 10^0 \pm 3.75 \cdot 10^{-1}$
(6, 8)	> 10h	—	—
(6, 9)	> 10h	—	—
(7, 2)	$1.19 \cdot 10^{-1} \pm 9.51 \cdot 10^{-2}$	$1.21 \cdot 10^{-4} \pm 6.54 \cdot 10^{-5}$	$1.00 \cdot 10^0 \pm 2.32 \cdot 10^{-1}$
(7, 3)	$1.07 \cdot 10^{-1} \pm 8.21 \cdot 10^{-2}$	$1.35 \cdot 10^{-4} \pm 4.56 \cdot 10^{-5}$	$9.81 \cdot 10^{-1} \pm 6.97 \cdot 10^{-2}$
(7, 4)	$2.01 \cdot 10^{-1} \pm 1.69 \cdot 10^{-1}$	$1.59 \cdot 10^{-4} \pm 5.39 \cdot 10^{-5}$	$1.06 \cdot 10^0 \pm 3.05 \cdot 10^{-1}$
(7, 5)	$1.74 \cdot 10^0 \pm 1.96 \cdot 10^{-1}$	$2.14 \cdot 10^{-4} \pm 8.74 \cdot 10^{-5}$	$1.03 \cdot 10^0 \pm 1.97 \cdot 10^{-1}$
(7, 6)	$4.63 \cdot 10^1 \pm 7.47 \cdot 10^1$	$2.53 \cdot 10^{-4} \pm 1.96 \cdot 10^{-4}$	$9.93 \cdot 10^{-1} \pm 9.99 \cdot 10^{-2}$
(7, 7)	$3.07 \cdot 10^3 \pm 2.43 \cdot 10^2$	$4.42 \cdot 10^{-4} \pm 4.66 \cdot 10^{-4}$	$1.30 \cdot 10^0 \pm 5.58 \cdot 10^{-1}$
(7, 8)	> 10h	—	—
(7, 9)	> 10h	—	—
(8, 2)	$5.31 \cdot 10^{-2} \pm 7.62 \cdot 10^{-2}$	$1.65 \cdot 10^{-4} \pm 6.12 \cdot 10^{-5}$	$9.72 \cdot 10^{-1} \pm 9.59 \cdot 10^{-2}$
(8, 3)	$1.26 \cdot 10^{-1} \pm 8.48 \cdot 10^{-2}$	$1.35 \cdot 10^{-4} \pm 3.27 \cdot 10^{-5}$	$9.78 \cdot 10^{-1} \pm 1.01 \cdot 10^{-1}$
(8, 4)	$3.28 \cdot 10^{-1} \pm 9.12 \cdot 10^{-2}$	$1.66 \cdot 10^{-4} \pm 4.85 \cdot 10^{-5}$	$1.04 \cdot 10^0 \pm 2.30 \cdot 10^{-1}$
(8, 5)	$4.23 \cdot 10^0 \pm 2.64 \cdot 10^{-1}$	$2.29 \cdot 10^{-4} \pm 8.65 \cdot 10^{-5}$	$1.05 \cdot 10^0 \pm 2.42 \cdot 10^{-1}$
(8, 6)	$1.89 \cdot 10^2 \pm 1.66 \cdot 10^1$	$3.19 \cdot 10^{-4} \pm 2.51 \cdot 10^{-4}$	$1.03 \cdot 10^0 \pm 2.43 \cdot 10^{-1}$
(8, 7)	$1.15 \cdot 10^4 \pm 6.84 \cdot 10^2$	$4.38 \cdot 10^{-4} \pm 4.86 \cdot 10^{-4}$	$1.50 \cdot 10^0 \pm 2.13 \cdot 10^0$
(8, 8)	> 10h	—	—
(8, 9)	> 10h	—	—
(9, 2)	$9.57 \cdot 10^{-2} \pm 1.08 \cdot 10^{-1}$	$1.24 \cdot 10^{-4} \pm 4.86 \cdot 10^{-5}$	$9.91 \cdot 10^{-1} \pm 2.27 \cdot 10^{-1}$
(9, 3)	$1.37 \cdot 10^{-1} \pm 1.14 \cdot 10^{-1}$	$1.34 \cdot 10^{-4} \pm 3.55 \cdot 10^{-5}$	$9.73 \cdot 10^{-1} \pm 6.19 \cdot 10^{-2}$
(9, 4)	$4.43 \cdot 10^{-1} \pm 6.03 \cdot 10^{-2}$	$1.68 \cdot 10^{-4} \pm 4.52 \cdot 10^{-5}$	$9.94 \cdot 10^{-1} \pm 6.11 \cdot 10^{-2}$
(9, 5)	$9.19 \cdot 10^0 \pm 7.88 \cdot 10^{-1}$	$2.31 \cdot 10^{-4} \pm 1.34 \cdot 10^{-4}$	$1.05 \cdot 10^0 \pm 2.16 \cdot 10^{-1}$
(9, 6)	$4.86 \cdot 10^2 \pm 5.10 \cdot 10^1$	$2.94 \cdot 10^{-4} \pm 1.75 \cdot 10^{-4}$	$1.09 \cdot 10^0 \pm 2.95 \cdot 10^{-1}$
(9, 7)	> 10h	—	—
(9, 8)	> 10h	—	—
(9, 9)	> 10h	—	—

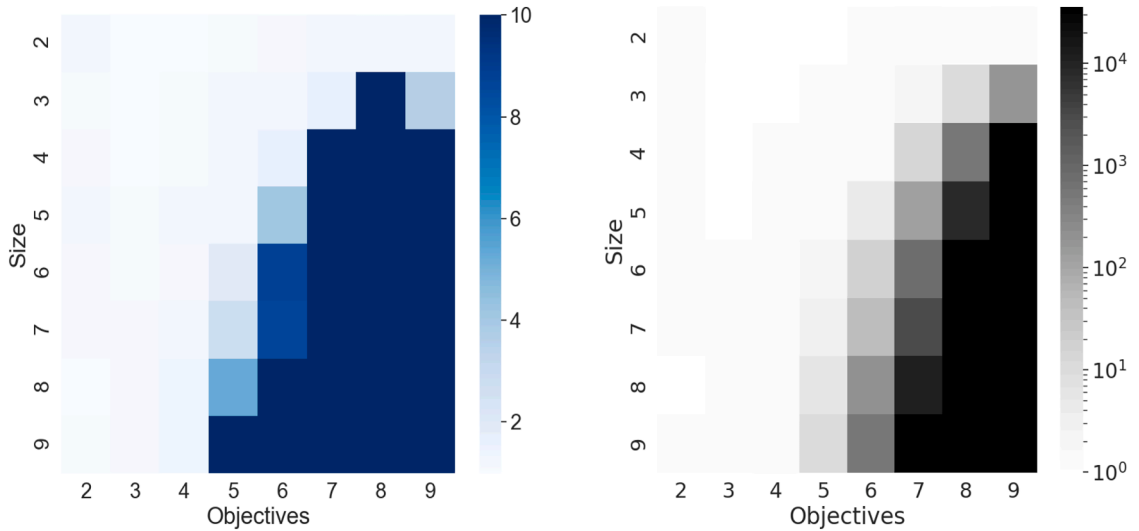


Fig. 8. Heat maps reporting computational costs in seconds of our ethical embedding algorithm for 64 possible configurations of Walkroom. Both left and right heat maps show the same results in different scales. Left: configurations that exceed a threshold of 10 s appear in dark blue (20 out of 64). Right: configurations that exceed a threshold of 10 h appear in black (11 out of 64). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

6. Related work

The literature on AI Value Alignment is broadly divided in two separate communities: the AI Safety community [11,12] and the Machine Ethics community [9,10]. Whilst these two communities typically do not communicate with each other, our work here aims to help bridge the gap between them. We provide first a summary of the focus of each research community, and then explain how this paper interconnects both communities.

6.1. AI safety

On the one hand, the focus of AI Safety is to prevent any possible harm that the agent may inflict upon itself or others [11] by guaranteeing a safe behaviour of the agent. The general approach to achieve a safe behaviour is to constrain [61,62] the agent so that it performs its intended objective while avoiding all *unsafe* outcomes. Unsafe outcomes are defined in different ways, such as minimising the side effects of the agent [63,64], or reducing its impact on the environment [12,54]. The AI Safety literature also differentiates between those situations in which the agent must be protected from unsafe outcomes once it has been deployed, or if the agent also needs to be protected while training [65,66]. While these works guarantee that the agent will behave safely, they cannot generalise to guarantee alignment to a different value, or alignment to a value system composed by several values, focusing exclusively on safety. In our work, the multi-valued ethical embedding process allows to model safety as one of the several values included in the value system that we expect the agent to align with. If in a given value system, the value of safety is the most prioritised one, our ethical embedding process will guarantee that it will be optimal for the agent to behave safely. If a different value is the most prioritised one of the value system, the same ethical embedding algorithm will design a different environment, adapting to the priorities of the value system.

6.2. Machine ethics

On the other hand, the focus of Machine Ethics is to develop beneficial agents to humans [2]. Again, different works define *beneficial* in different ways (see an extensive summary of the different possible definitions of what is beneficial from a computational point of view in [41,42]). There is an ongoing effort on defining and computationally formalising the moral values that an agent should align with [45,67] to be beneficial. Another line of research aims to directly incorporate the different moral theories into the decision making process of the agent [68,69]. Because the literature is still far from reaching an agreement on which is the correct definition of ethically beneficial behaviour, an alternative agnostic approach has emerged that can work under moral uncertainty [70]. Similarly, our work presents a model that can work for any value system, independently of which values the agent ought to align with. We provide a formal definition of the structure of a value system compatible with reinforcement learning. In particular, our definition of value system allows us to computationally design environments guaranteed to incentivise the agent to behave in alignment with a given value system.

There are alternative ways of classifying the literature on Machine Ethics. One relates to the different methods of instilling ethical behaviours: top-down, bottom-up, and hybrid, as outlined in surveys such as [71,72]. To summarise, top-down methodologies focus

on formalising ethical knowledge for direct integration into the agent's behaviour, while bottom-up strategies involve agents learning the ethical knowledge by themselves. As for hybrid approaches, they combine both bottom-up and top-down methods.

Several top-down approaches for formalising ethical knowledge have been explored in the literature. For instance, Sierra et al. [45] formalised values as preferences, while Mercuur et al. [73] introduced a conceptual division between values and norms, with values acting as a static component within agent behaviour and norms serving as a dynamic element. Additionally, Hansson et al. conducted studies on the formal relationship between norms and values [74]. In many approaches, such as the work of Liscio et al. [67], actions are considered to be either morally good or wrong depending on their context. Similarly, our work also defines how aligned an action is to a given value depending on the current environmental state. An alternative top-down way of formalising ethical knowledge is to directly conceptualise moral theories in terms of Markov Decision Process, like in the work of Nashed et al. [68,69]. Nashed et al. add a constraint to the decision processes of the agents that enforces them to follow the principles of a given ethical theory. This way, the agents learn to behave in compliance with the constraints of the chosen ethical theory (act utilitarianism, Kantianism, etc.¹⁰). Furthermore, other approaches consider environments with several coexisting moral values and propose the definition of a value system as a set of moral values together with preferences among them [27,46–48,76]. Here, we apply a definition of value system in the vein of [48], and extend it to the concept of *Multi-Valued Markov Decision Process* so that it can deal with sequential decision making problems. In general, the efforts made in formalising moral values and moral theories represent a significant contribution to the field of value alignment. This is because top-down approaches generally ensure that an agent adhering to them will exhibit ethical behaviour. Nonetheless, it is important to note that these approaches often involve significant computational costs and, in the most challenging scenarios, may become intractable. Hence, it is widely accepted that pure top-down methods are insufficient to address the entirety of the value alignment problem, as explained by Arnold et al. [40].

6.3. Reinforcement learning for alignment

Moving to bottom-up approaches, their focus is on the agent to learn to behave value-aligned without giving explicit definitions of what is ethical. Currently, most approaches deal with this ethical learning problem with reinforcement learning, following the research started by Russell, Soares and Fallenstein, among others [1,2]. In more detail, *inverse* reinforcement learning (IRL)[77] has been proposed as a potential solution for the value alignment problem. In Inverse Reinforcement Learning, the agent does not know its reward function. Instead, it has to learn its rewards from studying the optimal policy. This methodology, applied to the value alignment problem, would allow an agent to discover the moral values of humans (the hidden reward function) by observing their behaviour (the optimal policy) [14,15,17,78]. However, Inverse Reinforcement Learning approaches, and bottom-up approaches in general, also have several flaws. First of all, many of them consider the problem of reward specification as equivalent to the whole value alignment problem by considering that the agent's only objective is to behave ethically while ignoring that it may have its own goals. This issue has only started to be addressed recently by conceptualising the value alignment problem as a two-step process (reward specification and ethical embedding) that must take into account that the agent will have its own objectives (for instance, in [16–19,31]). By separating both steps, the learning environment of the agent can be adequately modelled as a multi-objective (reinforcement) learning environment.

All of these approaches consider a linear combination of rewards for the ethical embedding, similar to our approach. However, none of them considers the case in which the agent might need to be aligned with more than one moral value and, only the algorithms in [19,31] tackle the problem of how to guarantee that a linear combination of the individual objective and a single moral value actually incentivises the agent to behave ethically (i.e., in alignment with this single moral value). This last strand of work by Rodríguez-Soto et al., was originally only compatible with tabular reinforcement learning to offer theoretical guarantees. Interestingly, such a strand has been recently adapted to work with deep reinforcement learning algorithms in [79]. The deep RL methodology of Mayoral-Macau et al. can automate the weight of ethical rewards for value systems with a single value in large environments with millions of states at the expense of losing all the theoretical guarantees of the methodology of Rodríguez-Soto et al. In all other works, it is left to the environment designer to hand-tune rewards.

In parallel, there have started to appear non-linear approaches for tackling the problem of learning to align with multiple objectives [54,80,81]. Probably one of the earliest approaches was presented by Vamplew et al. in [80]. In that work, they formalised the problem of human alignment as a multi-objective reinforcement learning problem, as we do. Thereafter, they show some of the potential limitations of performing a linear combination of rewards for achieving a human-aligned behaviour: importantly that depending on the definition of aligned behaviour it might be unlearnable for the agent with a linear combination of rewards. Such limitation is not a problem in our case as proven by our theorems, which ensures that our current definition of value-aligned behaviour is guaranteed to be learnable with our embedding process.

Then, in the AI Safety community, alternatively to our approach of linearly combining the multiple reward functions, in both the works of Saisubramanian et al. and Vamplew et al., they incorporate a lexicographic ordering between the objectives of the agents with slack.¹¹ A lexicographic ordering guarantees that the agent will learn an ethical behaviour. This *agent-centred* methodology is perfectly valid if we have full control over the learning agent, but has several limitations if the agent is external to us, as we argued in Section 3.2. In particular, such an approach assumes that the agent's preference will always coincide with the value system, which may not be always the case. Instead, our ethical embedding process ensures that the agent's goals will always be aligned with the

¹⁰ For a brief explanation of each ethical theory, see [75].

¹¹ The slack denotes the maximum deviation from an optimal policy with respect to the agent's primary objective that is allowed.

value system by means of creating a simplified single-objective learning process for the agent. This problem simplification is achieved by automating the design of the weighting of the different reward functions.

Saisubramanian et al. also seek to address the limitations of non-linear approaches with an alternative approach in [81]. Their strategy involves reconfiguring the agent's states to enhance its ability to prevent adverse side effects. Much like our own approach, they acknowledge the cost associated with the designer's efforts in shaping the environment and, therefore, aim to identify the minimal modifications required to guarantee safe behaviour. While their approach shows promise, it becomes increasingly challenging to modify an environment's state in more complex scenarios. Additionally, their AI Safety focus is solely on preventing the agent from engaging in harmful or unsafe actions. Our approach generalises that problem into the problem of guaranteeing alignment to the multiple values of a value system.

Finally, from a formal point of view, this work further advances the theoretical results on lexicographic orderings in Multi-Objective MDPs. Previous work formalised lexicographic orders as preference relations, a different concept from scalarisation/utility functions in MOMDPs [82]. Furthermore, Rodríguez-Soto et al. showed how, in some example MOMDPs, it was possible to design a scalarisation function for which all lexicographic-order-maximising policies also maximised it and vice versa. In this work, we generalised this result with our [Theorem 1](#) for any finite MOMDP.

7. Conclusion and future work

The literature on value alignment has focused on aligning an agent with a single moral value, and except for [31], disregarding guarantees on an agent's ethical learning. Here, we tackled the problem of building an *ethical* environment that guarantees that an agent learns a policy aligned with multiple moral values.

Our novel contributions are founded in the framework of Multi-Objective MDPs (MOMDPs). With MOMDPs, we can formalise what it means for an agent to behave *value-aligned*, that is, following a value system of multiple values. Furthermore, we specify an algorithm to build an ethical environment with a so-called *multi-valued ethical embedding* process. An agent is guaranteed to learn a value-aligned policy in an ethical environment, provided that the agent employs a reinforcement learning algorithm that converges to the optimal policy.

Moreover, we provide a distinction between value-aligned and ethical policies to emphasise the requirement of the prioritisation of higher ethical standards over individualistic interests. Therefore, the problem of guaranteeing ethical behaviour learning also implies selecting the appropriate (ethical) value system. As further proof, our empirical analysis has shown that the policy learned in an ethical environment significantly varies depending on the value system of choice. Thus, ethical designers must carefully decide how to prioritise values in the value system to embed in an ethical environment. Ethical designers should correspond to a team of ethicists and domain experts that assess the ethical standards by a careful analysis of the domain and previous studies. From a pragmatic perspective, the designer questions how to analyse and compare different value-aligned policies. In this paper, we exemplify a methodology to tackle that challenge. First, a designer must define the candidate value systems to study. Moreover, they must define a collection of metrics, the so-called *ethical metrics*, to help them *quantify* the degree of value alignment of a policy for each moral value of a value system (e.g., time-steps to destination, accidents, dangerous driving, and bumps in our case study). Second, the designer must create one ethical environment per value system using our ethical embedding algorithm. After that, they must conduct the learning of policies for each ethical environment. Finally, the designer can roll out the learned policies to measure the individual value alignment of each policy per moral value. The result of those simulations will allow the designer to compare value-aligned policies, filter out those that are not admissible (e.g., policies resulting in accidents and dangerous driving situations), and ultimately select the value system to deploy. However, it is worth recalling that for the agent to learn an ethical policy, the selected value system must necessarily prioritise the highest ethical standards (e.g., safety) above the agent's individual goal.

Finally, as our empirical analysis shows, providing theoretical guarantees comes at a computational cost, which mainly resides in Convex Hull Value Iteration. The more values we want an agent to align with, the more the computational cost increases. Hence, future research should focus on testing and developing efficient algorithms to compute convex hulls to cope with large environments. We expect that such research will also allow us to extend our work to environments with continuous or high-dimensional states. Moreover, another interesting future line of research would be the use of symbolic regression models to extract directly the desired ethical weight vector that designs an ethical environment without the need of computing the convex hull altogether. In more detail, our empirical findings show that the computational cost explodes when considering 8 or more moral values. Many-objective environments with 8 or more objectives are still scarce, so future research should also focus on developing new many-objective environments as a test bed for new algorithms.

CRediT authorship contribution statement

Manel Rodríguez-Soto: Writing – review & editing, Writing – original draft, Formal analysis, Data curation, Conceptualization; **Roxana Rădulescu:** Writing – review & editing, Formal analysis, Data curation, Conceptualization; **Filippo Bistaffa:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Data curation; **Oriol Ricart:** Writing – original draft, Visualization, Software, Data curation; **Arnau Mayoral-Macau:** Writing – review & editing, Writing – original draft, Formal analysis; **Maite Lopez-Sanchez:** Writing – review & editing, Writing – original draft, Formal analysis, Conceptualization; **Juan A. Rodríguez-Aguilar:** Writing – review & editing, Writing – original draft, Formal analysis; **Ann Nowé:** Writing – review & editing, Writing – original draft, Formal analysis.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research project was made possible through the access granted by the Galician Supercomputing Center (CESGA) to its supercomputing infrastructure. The supercomputer FinisTerae III and its permanent data storage system have been funded by the Spanish Ministry of Science and Innovation, the Galician Government, and the [European Regional Development Fund](#) (ERDF). **Funding sources** The research presented in this paper was supported by the EU-funded VALAWAI (# 101070930) project, the Spanish-funded EVASAI (# PID2024-158227NB-C31), ANNOTATE-SenSIT (# PID2024-156022OB-C33), VAE (# TED2021-131295B-C31), EMOROBCARE (# IASOMMA2024) and Rhymas (# PID2020-113594RB-100) projects. This work was supported by grant PID2022-136787NB-I00 funded by MCIN/AEI/10.13039/501100011033. It was also funded by GUARDEN (101060693), Fairtrans (PID2021-124361OB-C33), AUTODEMO (SR21-00329), TAILOR (H2020-952215), PrepParticip2.0 (24S03545-001), 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. Roxana Rădulescu was partly supported by the Research Foundation Flanders (FWO), grant number 1286223N. This research was partially supported by funding from the Flemish Government under the “Onderzoekprogramma Artificiële Intelligentie (AI) Vlaanderen” program and by the FWO, grant number G062819N.

References

- [1] N. Soares, B. Fallenstein, Aligning Superintelligence with Human Interests: A Technical Research Agenda, Machine Intelligence Research Institute (MIRI) technical report 8, 2014.
- [2] S. Russell, D. Dewey, M. Tegmark, Research priorities for robust and beneficial artificial intelligence, *Ai Magazine* 36 (2015) 105–114. <https://doi.org/10.1609/aimag.v36i4.2577>
- [3] R. Chatila, V. Dignum, M. Fisher, F. Giannotti, K. Morik, S. Russell, K. Yeung, Trustworthy AI, in: *Reflections on Artificial Intelligence for Humanity*, Springer, 2021, pp. 13–39.
- [4] IEEE, IEEE global initiative on ethics of autonomous and intelligent systems, 2019, (<https://standards.ieee.org/industry-connections/ec/autonomous-systems.html>). Accessed: 2024-10-31.
- [5] E. Commission, Artificial intelligence act, 2021, (<https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1623335154975&uri=CELEX%3A52021PC0206>). Accessed: 2021-06-29.
- [6] J.P. Boada, B.R. Maestre, C.T. Genís, The ethical issues of social assistive robotics: a critical literature review, *Technol. Soc.* 67 (2021) 101726.
- [7] S.O. Hansson, *The Structure of Values and Norms*, Cambridge Studies in Probability, Induction and Decision Theory, Cambridge University Press, Cambridge, Cambridge, 2001. <https://doi.org/10.1017/CBO9780511498466>
- [8] J. Casas-Roma, J. Conesa, Towards the design of ethically-aware pedagogical conversational agents, in: *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, Springer, 2020, pp. 188–198.
- [9] H. Yu, Z. Shen, C. Miao, C. Leung, V.R. Lesser, Q. Yang, Building ethics into artificial intelligence, in: *IJCAI*, 2018, p. 5527–5533.
- [10] F. Rossi, N. Mattei, Building ethically bounded AI, *Proc. AAAI Confer. Artif. Intell.* 33 (2019) 9785–9789. <https://doi.org/10.1609/aaai.v33i01.33019785>
- [11] D. Amodei, C. Olah, J. Steinhardt, P.F. Christiano, J. Schulman, D. Mané, Concrete problems in AI safety, *arXiv:1606.06565* (2016).
- [12] J. Leike, M. Martic, V. Krakovna, P. Ortega, T. Everitt, A. Lefrancq, L. Orseau, S. Legg, AI safety gridworlds, *arXiv: 1711.09883* (2017).
- [13] R.S. Sutton, A.G. Barto, *Reinforcement Learning - An Introduction*, Adaptive computation and machine learning, MIT Press, 1998. <http://www.worldcat.org/oclc/37293240>.
- [14] M.O. Riedl, B. Harrison, Using stories to teach human values to artificial agents, in: *AAAI Workshop: AI, Ethics, and Society*, 2016.
- [15] D. Abel, J. MacGlashan, M.L. Littman, Reinforcement learning as a framework for ethical decision making, in: *AAAI Work.: AI, Ethics, and Society*, 92, 2016.
- [16] Y.-H. Wu, S.-D. Lin, A low-cost ethics shaping approach for designing reinforcement learning agents, in: *Proceedings of the AAAI Conference on Artificial Intelligence*, 32, 2018.
- [17] R. Noothigattu, D. Bouneffouf, N. Mattei, R. Chandra, P. Madan, R. Kush, M. Campbell, M. Singh, F. Rossi, Teaching AI agents ethical values using reinforcement learning and policy orchestration, *IBM J. Res. Dev. PP* (2019) 6377–6381. <https://doi.org/10.1147/JRD.2019.2940428>
- [18] A. Balakrishnan, D. Bouneffouf, N. Mattei, F. Rossi, Incorporating behavioral constraints in online AI systems, *Proc. AAAI Conf. Artif. Intell.* 33 (2019) 3–11. <https://doi.org/10.1609/aaai.v33i01.33013>
- [19] M. Rodríguez-Soto, M. Lopez-Sanchez, J.A. Rodríguez Aguilar, Multi-objective reinforcement learning for designing ethical environments, in: Z.-H. Zhou (Ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization*, 2021, pp. 545–551. Main Track.
- [20] D. Cooper, *Value Pluralism and Ethical Choice*, St. Martin Press, Inc., New York, New York, 1993.
- [21] W.D. Ross, *The Right and the Good Some Problems in Ethics*, Clarendon Press, 1930.
- [22] T.L. Beauchamp, J.F. Childress, *Principles of Biomedical Ethics* / Tom L. Beauchamp, James F. Childress, Oxford University Press New York, 1979.
- [23] I. van de Poel, L. Royakkers, *Ethics, Technology, and Engineering: An Introduction*, Wiley-Blackwell, 2011.
- [24] S. Schwartz, An Overview of the Schwartz Theory of Basic Values, <https://doi.org/10.9707/2307-0919.1116>, *Online Readings in Psychology and Culture*, 2, pp. 11–20 Jerusalem Hebrew University (2012).
- [25] D. Roijers, S. Whiteson, Multi-Objective Decision Making, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Morgan and Claypool, California, USA, California, USA, 2017. <https://doi.org/10.2200/S00765ED1V01Y201704AIM034>, <http://www.morganclaypool.com/doi/abs/10.2200/S00765ED1V01Y201704AIM034>.
- [26] R. Rădulescu, P. Mannion, D.M. Roijers, A. Nowé, Multi-objective multi-agent decision making: a utility-based analysis and survey, *Auton. Agent Multi. Agent Syst.* 34 (2019) 1–52.
- [27] W. Caballero, R. Naveiro, D. Rios, Modeling ethical and operational preferences in automated driving systems, *Decis. Anal.* 19 (2021). <https://doi.org/10.1287/deca.2021.0441>

- [28] W.N. Caballero, D. Rios Insua, R. Naveiro, Some statistical challenges in automated driving systems, *Appl. Stoch. Models Bus. Ind.* 39 (5) (2023) 629–652. <https://onlinelibrary.wiley.com/doi/abs/10.1002/asmb.2765>.
- [29] C.J. Watkins, P. Dayan, Q-learning, *Mach. Learn.* 8 (1992) 279–292.
- [30] M. Reymond, E. Bargiacchi, A. Nowé, Pareto conditioned networks, in: *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS '22*, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2022, p. 1110–1118.
- [31] M. Rodríguez-Soto, M. Serramia, M. López-Sánchez, J. Rodríguez-Aguilar, Instilling moral value alignment by means of multi-objective reinforcement learning, *Ethics Inf. Technol.* 24 (2022). <https://doi.org/10.1007/s10676-022-09635-0>
- [32] R. Bellman, A Markovian decision process, *J. Math. Mech.* 6 (5) (1957) 679–684.
- [33] L.P. Kaelbling, M.L. Littman, A.W. Moore, Reinforcement learning: a survey, *J. Artif. Int. Res.* 4 (1) (1996) 237–285.
- [34] D.M. Roijers, P. Vamplew, S. Whiteson, R. Dazeley, A survey of multi-objective sequential decision-making, *J. Artif. Int. Res.* 48 (1) (2013) 67–113.
- [35] P. Vamplew, R. Dazeley, A. Berry, R. Issabekov, E. Dekker, Empirical evaluation methods for multiobjective reinforcement learning algorithms, *Mach. Learn.* 84 (2011) 51–80. <https://doi.org/10.1007/s10994-010-5232-5>
- [36] A. Castelletti, G. Corani, A. Rizzoli, R.S. Sessa, E. Weber, Reinforcement learning in the operational management of a water system, in: *Modelling and Control in Environmental Issues*, 2002.
- [37] S. Natarajan, P. Tadepalli, Dynamic preferences in multi-criteria reinforcement learning, in: *Proceedings of the 22nd International Conference on Machine Learning, ICML '05*, Association for Computing Machinery, New York, NY, USA, 2005, p. 601–608. <https://doi.org/10.1145/1102351.1102427>
- [38] L. Barrett, S. Narayanan, Learning all optimal policies with multiple criteria, *Proc. 25th Int. Conf. Mach. Learn.* (2008) 41–47. <https://doi.org/10.1145/1390156.1390162>
- [39] K. Van Moffaert, A. Nowé, Multi-objective reinforcement learning using sets of pareto dominating policies, *J. Mach. Learn. Res.* 15 (1) (2014) 3483–3512.
- [40] T. Arnold, D. Kasenberg, M. Scheutz, Value alignment or misalignment - what will keep systems accountable?, in: *AAAI Workshops*, 2017.
- [41] I. Gabriel, Artificial intelligence, values, and alignment, *Minds Mach.* 30 (2020) 411–437. <https://doi.org/10.1007/s11023-020-09539-2>
- [42] M. Sutrop, Challenges of aligning artificial intelligence with human values, *Acta Baltica Historiae et Philosophiae Scientiarum* 8 (2020) 54–72. <https://doi.org/10.11590/abhs.2020.2.04>
- [43] E. Mason, Value pluralism, in: E.N. Zalta, U. Nodelman (Eds.), *The Stanford Encyclopedia of Philosophy*, Metaphysics Research Lab, Stanford University, Summer 2023 edition, 2023.
- [44] M. Serramia, M. López-Sánchez, J.A. Rodríguez-Aguilar, J. Morales, M. Wooldridge, C. Ansotegui, Exploiting moral values to choose the right norms, in: *Proceedings of the 1st Conference on Artificial Intelligence, Ethics and Society (AIIES'18)*, 2018, pp. 1–7. <https://doi.org/10.1145/3278721.3278735>
- [45] C. Sierra, N. Osman, P. Noriega, J. Sabater-Mir, A. Perello-Moragues, Value alignment: a formal approach, *Responsible Artif. Intell. Agents Workshop (RAIA) AAMAS 2019* (2019).
- [46] T.J.M. Bench-Capon, K. Atkinson, *Abstract argumentation and values*, in: *Argumentation in Artificial Intelligence*, Springer, 2009, pp. 45–64. https://doi.org/10.1007/978-0-387-98197-0_3
- [47] J. Luo, J.-J. Meyer, M. Knobbout, Reasoning about opportunistic propensity in multi-agent systems, in: *AAMAS 2017 Workshops, Best Papers.*, 2017, pp. 1–16.
- [48] M. Serramia, M. López-Sánchez, S. Moretti, J.A. Rodríguez-Aguilar, On the dominant set selection problem and its application to value alignment, *Auton. Agent Multi Agent Syst.* 35 (2) (2021) 1–38.
- [49] T.L. Beauchamp, J.F. Childress, *Principles of biomedical ethics*, Oxford University Press. 8th Edition., New York, 2019.
- [50] K. Wray, S. Zilberstein, A.-i. Mouaddib, Multi-objective MDPs with conditional lexicographic reward preferences, *Proc. AAAI Conf. Artif. Intell.* 29 (2015). <https://doi.org/10.1609/aaai.v29i1.9647>
- [51] E.W. Weisstein, *CRC Concise Encyclopedia of Mathematics*, Chapman & Hall : CRC Press, Boca Raton, FL, Boca Raton, FL, 2002.
- [52] C.F. Hayes, R. Rădulescu, E. Bargiacchi, J. Källström, M. Macfarlane, M. Reymond, T. Verstraeten, L.M. Zintgraf, R. Dazeley, F. Heintz, E. Howley, A.A. Irissappane, P. Mannion, A. Nowé, G. Ramos, M. Restelli, P. Vamplew, D.M. Roijers, A practical guide to multi-objective reinforcement learning and planning, *Auton. Agent Multi. Agent Syst.* 36 pp. 1–59 (2022).
- [53] R.K. Dash, N.R. Jennings, D.C. Parkes, Computational-Mechanism design: a call to arms, *IEEE Intell. Syst.* 18 (6) (2003) 40–47. <https://doi.org/10.1109/MIS.2003.1249168>
- [54] P. Vamplew, C. Foale, R. Dazeley, A. Bignold, Potential-based multiobjective reinforcement learning approaches to low-impact agents for AI safety, *Eng. Appl. Artif. Intell.* 100 (2021). <https://doi.org/10.1016/j.engappai.2021.104186>
- [55] K.L. Clarkson, Applications of random sampling in computational geometry, II, in: *Proceedings of the Fourth Annual Symposium on Computational Geometry, SCG '88*, Association for Computing Machinery, New York, NY, USA, 1988, p. 1–11. <https://doi.org/10.1145/73393.73394>
- [56] Ethics Unwrapped, Bounded ethicality, 2024, (<https://ethicsunwrapped.utexas.edu/glossary/bounded-ethicality>). Accessed: 2024-06-11.
- [57] Ethics Unwrapped, Self-serving bias, 2024, (<https://ethicsunwrapped.utexas.edu/glossary/self-serving-bias>). Accessed: 2024-06-11.
- [58] M. Stojanovski, N. Bourdache, G. Bonnet, A.-I. Mouaddib, Ethical Markov decision processes with moral worth as rewards, in: *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, 2024, pp. 2486–2488.
- [59] L.N. Alegre, F. Felten, E.-G. Talbi, G. Danoy, A. Nowé, A.L.C. Bazzan, B.C.D. Silva, MO-Gym: a library of multi-objective reinforcement learning environments, in: *Proceedings of the 34th Benelux Conference on Artificial Intelligence BNAIC/Benelearn 2022*, 2022.
- [60] F. Felten, L.N. Alegre, A. Nowé, A.L.C. Bazzan, G. Danoy, B.C. da Silva, et al., A toolkit for reliable benchmarking and research in multi-objective reinforcement learning, in: *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.
- [61] J. García, F. Fernández, A comprehensive survey on safe reinforcement learning, *J. Mach. Learn. Res.* 16 (1) (2015) 1437–1480.
- [62] Y. Chow, O. Nachum, E. Duenez-Guzman, M. Ghavamzadeh, A Lyapunov-based approach to safe reinforcement learning, in: *NeurIPS 2018*, 2018.
- [63] V. Krakovna, L. Orseau, M. Martic, S. Legg, Penalizing side effects using stepwise relative reachability, *arXiv:1806.01186* <https://doi.org/10.48550/arXiv.1806.01186> (2019).
- [64] S. Saisubramanian, E. Kamar, S. Zilberstein, A multi-objective approach to mitigate negative side effects, in: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2020, pp. 354–361. <https://doi.org/10.24963/ijcai.2020/50>
- [65] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, U. Topcu, Safe reinforcement learning via shielding, in: *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [66] I. Elsayed-Aly, S. Bharadwaj, C. Amato, R. Ehlers, U. Topcu, L. Feng, Safe multi-agent reinforcement learning via shielding, in: *Proceedings of the 20th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2021)*, 2021.
- [67] E. Liscio, M.V.D. Meer, L.C. Siebert, C. Jonker, N. Mouter, P.K. Murukannaiah, Axes: identifying and evaluating context-specific values, in: *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, 2021.
- [68] J. Svegliato, S.B. Nashed, S. Zilberstein, Ethically compliant sequential decision making, in: *Proceedings of the 35th AAAI International Conference on Artificial Intelligence*, 2021.
- [69] S.B. Nashed, J. Svegliato, S. Zilberstein, Ethically compliant sequential decision making, in: *Proceedings of the 4th Conference on AI, Ethics, and Society (AIIES)*, 2021.
- [70] A. Ecoffet, J. Lehman, Reinforcement learning under moral uncertainty, in: M. Meila, T. Zhang (Eds.), *Proceedings of the 38th International Conference on Machine Learning*, 139 of *Proceedings of Machine Learning Research*, PMLR, 2021, pp. 2926–2936. <https://proceedings.mlr.press/v139/ecoffet21a.html>
- [71] C. Allen, I. Smit, W. Wallach, Artificial morality: top-down, bottom-up, and hybrid approaches, *Ethics Inf. Technol.* 7 (2005) 149–155. <https://doi.org/10.1007/s10676-006-0004-4>
- [72] S. Tolmeijer, M. Kneer, C. Sarasua, M. Christen, A. Bernstein, Implementations in machine ethics: a survey, *ACM Comput. Surv.* 53 (6) (2021). <https://doi.org/10.1145/3419633>
- [73] R. Mercuur, V. Dignum, C. Jonker, et al., The value of values and norms in social simulation, *J. Artif. Soc. Social Simul.* 22 (1) (2019) 1–9.
- [74] S.O. Hansson, V. Hendricks, *Introduction to Formal Philosophy*, Springer, Berlin, Berlin, 2018.

- [75] R. Audi, *The Cambridge Dictionary of Philosophy*, Cambridge University Press, Cambridge, Cambridge, 1999.
- [76] L.C. Siebert, E. Liscio, P.K. Murukannaiah, L. Kaptein, S.L. Spruit, J. van den Hoven, C.M. Jonker, Estimating value preferences in a hybrid participatory system, in: *Proceedings of the First International Conference on Hybrid Human-Artificial Intelligence, HHAI 2022*, IOS Press, Amsterdam, the Netherlands, 2022, pp. 1–14.
- [77] P. Abbeel, A.Y. Ng, Apprenticeship learning via inverse reinforcement learning, in: *Proceedings of the Twenty-first International Conference on Machine Learning, ICML '04*, ACM, New York, NY, USA, 2004, pp. 1. <https://doi.org/10.1145/1015330.1015430>
- [78] D. Hadfield-Menell, S.J. Russell, P. Abbeel, A. Dragan, Cooperative inverse reinforcement learning, in: *Advances in Neural Information Processing Systems 29*, Berkeley, 2016, pp. 3909–3917.
- [79] A. Mayoral-Macau, E. Marchesini, M. Rodríguez-Soto, M. Lopez-Sanchez, M. Sanchez-Fibla, J.A. Rodríguez-Aguilar, A. Farinelli, Designing ethical environments using multi-agent reinforcement learning, in: *Adaptive and Learning Agents Workshop (AAMAS 2025)*, 2025.
- [80] P. Vamplew, R. Dazeley, C. Foale, S. Firmin, J. Mummery, Human-aligned artificial intelligence is a multiobjective problem, *Ethics Inf. Technol.* 20 (2018). <https://doi.org/10.1007/s10676-017-9440-6>
- [81] S. Saisubramanian, S. Zilberstein, Mitigating Negative Side Effects via Environment Shaping, *International Foundation for Autonomous Agents and Multiagent Systems*, 2021, p. 1640–1642.
- [82] M. Rodríguez-Soto, J.A. Rodríguez-Aguilar, M. Lopez-Sanchez, An analytical study of utility functions in multi-objective reinforcement learning, in: *Proceedings of the 38th International Conference on Neural Information Processing Systems, NIPS '24*, Curran Associates Inc., Red Hook, NY, USA, 2025.