Cost Of Interpretability - Blocked Value Iteration

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Abstract

In fields such as healthcare, law-making, or industrial design, there is a need to develop machine learning algorithms which are not black-box, and can be understood by experts in that field. Interpretability becomes a key requirement for machine learning algorithms to possess. In this paper, we define interpretable policies as being regionwise constant maps. Previous work such as in (Petrik and Luss 2016) has computed the optimal policy which takes actions on a partitioned state space. Our work is the first that aims to compute the regions as well as derive the optimal action to take on the partition. We compute upper bounds on the cost of interpretability as the error in summarizing the final policy by a region-wise constant map. Further we notice that this is given as a linear function of the summarization error for the final policy. Finally, we run experiments to check how our approach performs with the dimension and size of the state space. We end with the computation of the optimal interpretable policy for different final policies.

Introduction

Without a doubt, it is true that artificial intelligence has led to better outcomes for humanity. To further increase it's applicability in fields such as industrial design, robotics, or healthcare, the output of an algorithm must be interpretable. Towards this goal, dynamic decision-making algorithms whose output can be understood by humans is the need of the hour.

For humans to understand the output of a dynamic decision-making algorithm, the behaviour of the output policy must be understood by a person from that area of study. One way to do this is to ensure the output policy follows a data structure which we deem to be interpretable. Traditional ways of defining interpretability involve tree-based final policies (Zhu, Zeng, and Kosorok 2015), or policies based on lists as given in (Zhang et al. 2016). An information theoretic way to think of interpretability is to have the final policy be a region-wise constant map and let the number of regions be inversely proportional to the interpretability of the map. As from traditional ML, interpretable and accuracy are at odds with another. Interpretability increases when the number of allowable regions decreases, whereas accuracy of the final policy increases when we have more regions. When visualizing the policy, having coarser regions would make the policy easier to visualize and thus more 'interpretable'.

Copyright © 2027, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Computing piecewise constant approximations of functions has been studied in the signal processing literature. Algorithms exist to find the piecewise constant approximator for a given function in $1D$ space such as in (Bergerhoff, Weickert, and Dar 2019a) or (Dar and Bruckstein 2019). In our paper, we define the cost of interpretability as the expected difference in the optimal and interpretable value functions when averaged over the different timesteps. In our paper, we show that the cost of interpretability is bounded by a region-wise constant approximation to the optimal final policy. Further based on the geometry of the splitting procedure, we can derive an approximation to the optimal interpretable policy.

In (Bergerhoff, Weickert, and Dar 2019a) they assume that a partition of the state space is given through an interpretability map from states to a new space which they define to be observations. They obtain policies on the quotient of the state space by defining interpretable policies to be those deterministic policies that take the same value on the pre-image of the interpretability map. One can hope to define any partition of the state space by using such a map, and then use their method to compute the exact optimal policy. The problem we wish to solve involves computation of the interpretability map.

Contributions

Our contributions can be summarized as follows:

1. We define a notion of interpretability that is given by regionwise constant maps. Further, we define the cost of interpretability as the average sum over the differences between the expected values of the optimal and interpretable value functions.
2. We show that the cost of interpretability is bounded by the regionwise constant approximation to the final obtained policy.
3. We run simulations where we compute upper bounds for the cost of interpretability when we have a $2-D$ grid. We

check how the cost changes when we vary the dimension and size of the state space.

Our paper is organized in the following manner. In the next section we discuss related work in interpretable reinforcement learning. In Sec. 3 we set up the background and related definitions on interpretability. In Sec. 4 we list the assumptions for the bounds. This is followed by the blocked value iteration procedure in Sec. 5. In Sec. 6 we discuss the computational experiments done where we compute the cost of interpretability for different environmental settings. We end with the conclusion section where we discuss the main insights that can be drawn from this paper.

Related Work

In this section, we recall different notions of interpretability for Markov Decision Processes and the reinforcement learning problem from the literature. In (Grand-Clement et al. 2021) model-based approaches for interpretable reinforcement learning is presented. In this paper, they optimize for the optimal decision tree and rule for a given tree structure in the final optimization problem. In our excerpt, we show that we can upper bound the cost of interpretability using regionwise constant maps and we get that this is given by approximating the final policy using such a map. We now compute the optimal partition using a tree-based splitting mechanism where we decide the splits based on which split minimizes the total distance from the median. We can think of it as an L^1 version of the CART algorithm from (Steinberg 2009).
The paper (Givan, Dean, and Greig 2003) gives another way to obtain this partitioning of a Markov Decision Process. In this paper, they define a factored MDP- a Markov Decision Process obtained over a smaller state space after combining equivalent states. For MDPs with a massive state space, this would result in exponentially fewer states and we would be able to use algorithms running in polynomial time over the state space on this reduced Markov Decision Process. They group together states to form quotient MDPs. They also come up with approaches to perform model minimization for MDPs.
In (Hein, Udluft, and Runkler 2018) we perform reinforcement learning on existing trajectory data to obtain policies which are represented by basic algebraic equations that are restricted to an adequate complexity. Genetic Programming for Reinforcement Learning approaches work as follows; we find the optimal policy amongst the set of all possible equations which can be built upto a certain complexity. This can be thought of as a version of solving the cost of interpretability problem from the introduction since we are choosing those policies to solve the problem that are below a certain degree of complexity. Here policies that are represented as simple algebraic equations are preferred.
In (Petrik and Luss 2016) they assume a partition of the space is known through an interpretability map. This maps states to a new space which they define to be the observation space. They obtain policies on the quotient of the state space by defining interpretable policies to be deterministic maps that take the same value on the pre-image of the interpretability map. One can hope to define any partition of the state space by using such a map, and then use their method to compute the exact optimal policy. The problem we wish to solve involves the computation of the interpretability map.

The article (Givan, Dean, and Greig 2003) is a survey on interpretable reinforcement learning that speaks about interpretability in different contexts. They start by introducing the notion of interpretable inputs, where the agent performs symbol extraction to convert a high-dimensional input from the world into a lower-dimensional perception model. Next, they speak about interpretable transition models where the probability transitions are made to be interpretable by learning decision trees or other graphical models to represent the probability transitions. Another area where interpretability can be introduced is when we have to model the transitions in the rewards. The ability to understand how the reward dynamics work in a given environment proves to be useful in learning which actions an agent would take and at what stage. The last area they speak about, which is related to our line of work is interpretability in the policies. Here they list literature where the agent learns tree-based interpretable policies either directly while solving for the optimal value function or as a post-hoc method after obtaining an approximation of the optimal interpretable policy.

In our paper, we show that a way to upper bound the cost of interpretability is by performing an interpretable approximation of the obtained final policy. We provide theoretical justification to the claim that a region-wise constant approximation of the policy would help us give upper bounds on the cost of interpretability.

Background

In this section, we give the necessary background to understand reinforcement learning, and provide the definitions related to interpretability that is neccesary to understand the next sections of the manuscript.

Reinforcement Learning Notation

The central object of study in reinforcement learning is the Markov Decision Process (MDP). The MDP is given by the following tuple $(T, \gamma, \rho_0, (S_t)_{t=0}^{T}, (A_t)_{t=0}^{T-1}, (r_t)_{t=0}^{T-1}, (P_t)_{t=0}^{T-1})$. In the above we have $T \in \mathbb{N}$ to be the time-horizon of the decision-making process, ρ_0 to be the initial state distribution on space S_0, $\{S_t\}_{t=1}^{T-1}$ are the state spaces for the different timesteps $t \in \{0, 1, 2, ..., T-1\}$, A_t the action spaces for the different timesteps $t \in \{0, 1, 2, ..., T-1\}$, $r_t : S_t \times A_t \to \mathbb{R}$ denotes the reward maps, and $P_t : S_t \times A_t \to \Delta(S_{t+1})$ are the transition kernels for timesteps $t \in \{0, 1, 2, ..., T-1\}$, where $\Delta(S_t)$ is the space of probability distributions over S_t. We provide the following notation $[T]$ which would look like a set of the form $\{0, 1, 2, ..., T-1\}$. With the above setup, we say that a Markov Decision Process $(T, \gamma, \rho_0, (S_t)_{t=1}^{T}, (A_t)_{t=0}^{T-1}, (r_t)_{t=0}^{T-1}, (P_t)_{t=0}^{T-1})$ is **time-**

homogeneous if $S_t = S_0, \forall t \in [T]$, $A_t = A_0, \forall t \in [T-1]$, and $P_t = P_0 \forall t \in [T-1]$. The equalities for the sets in the following are set-theoretic, and for the functions we have equivalence in the functional sense; i.e if f, g are functions over a domain S, we say that $f = g$, if $f(s) = g(s) \forall s \in S$.

In the above setting, we see that the transition kernels $P_t^a : \mathcal{F}(S_t) \to \mathcal{F}(S_t)$ acts as a map from the space of functions over S_t where we have:

$$(P_t^a f)(s) := \sum_{s' \in S} P_t(s'|s,a) f(s') \tag{1}$$

We have the metric $d_{A_t}(.,.)$ on the action spaces for the different timesteps.

Interpretability definitions

We recall the notation and define the necessary preliminaries to setup the theory on the cost of interpretability.

1. For each timestep t, we are given a constant h_t which denotes the depth of the interpretable tree based policy. We would then want to find a partition of size d_t of the state space where $d_t = 2^{h_t}$, we then have that $S_t = \sqcup_{i=1}^{d_t} K_t^i$ and actions $a_t^1, a_t^2, ..., a_t^{d_t}$ such that we can define a region-wise constant map $\pi_t^I : \sqcup_{i=1}^{d_t} K_t^i \to A_t$ where $\pi_t^I|_{K_t^i} = a_t^i \forall i \in [d_t]$.

$$\pi_t^I(x) = \begin{cases} a_t^1, & \text{if } x \in K_t^1, \\ a_t^2, & \text{if } x \in K_t^2, \\ \vdots & \vdots \\ a_t^{d_t}, & \text{if } x \in K_t^{d_t}. \end{cases}$$

   In our setting, regionwise-constant maps from the state space to the action space can be regarded as **interpretable deterministic policies**.
2. Given heights $(h_t)_{t=0}^{T-1}$, we can define $\mathbf{P}((h_t)_{t=0}^{T-1})$ to be the space of interpretable policies over the T different timesteps such that π_t^I is a decision tree policy over a partition of utmost height $h_t \in \mathbb{R}^+$
3. We measure the cost of interpretability for the final polices $(\pi_t^*)_{t=0}^{T-1}, (\pi_t^I)_{t=0}^{T-1}$ with $t \in [0, 1, 2, ..., T-1]$. The cost of interpretability at timestep t for the final policy π_t^*, and interpretable policy π_t^I is given by:

$$\mathbf{C}_t(\pi_t^*, \pi_t^I) := \frac{1}{|S_t|} \int_{S_t} |V_t^{\pi^*}(s) - V_t^{\pi^I}(s)| ds \tag{2}$$

   The average cost of interpretability for a set of final policies π_t^*, π_t^I for different timesteps $t \in [T]$ is given by:

$$\mathbf{C}((\pi_t^*)_{t=0}^{T-1}, (\pi_t^I)_{t=0}^{T-1}) := \frac{1}{T|S_0|} \sum_{t=0}^{T-1} \int_{S_t} |V_t^{\pi^*}(s) - V_t^{\pi^I}(s)| ds \tag{3}$$

4. Here we introduce a special class of regionwise constant maps defined by regions and a leader set called **Leader Optimal Regionwise Constant (LORC)** maps. For a given timestep $t \in [T]$, we can define a LORC map given by the partition of the state space S_t, i.e $\mathcal{P}_t := \sqcup_{l=1}^{d_t} K_t^l$ and a leader set $L_t^* := \{k_t^{*l} | l \in [d_t], t \in [T], k_t^{*l} \in K_t^l\}$. Then the LORC characterized by the partition and the leader-set is given by:

$$\pi_t^I(x) = \begin{cases} \pi_t^*(k_t^{*1}), & \text{if } x \in K_t^1, \\ \pi_t^*(k_t^{*2}), & \text{if } x \in K_t^2, \\ \vdots & \vdots \\ \pi_t^*(k_t^{*d_t}), & \text{if } x \in K_t^{d_t}. \end{cases}$$

   The value function for the above policy (which we call as the interpretable value function) is denoted by V_t^I.
5. Note that each LORC is just characterized by the optimal regions and leader sets. We also introduce a notation for the average cost of interpretability based on the leader sets and the partitions used which is given by:

$$\mathbf{C}(\{\mathcal{P}_t\}_{t=0}^{T-1}, \{L_t^*\}_{t=0}^{T-1}) := \frac{1}{T|S_0|} \sum_{t=0}^{T-1} \int_{S_t} \left| V_t^{\pi^*}(s) - V_t^{\pi^I}(s) \right| ds \tag{4}$$

Assumptions for the bounds

We now list various assumptions that are necessary to get bounds on the cost of using an interpretable policy.

1. **Lipschitzness** in the reward map:
   We assume that
$$\mathbb{E}\Big[|r_t(s,a) - r_t(s,b)| \Big] \leq C_r d_A(a,b)$$
   . for $t \in [T]$.
2. **Bounds** on the reward maps : $|r_t(s,a)| \leq A$ for $t \in [T]$
3. **Lipschitz** contractivity in the action coefficient:
$$|P_t(s'|s,a) - P_t(s'|s,b)| \leq C_p d_A(a,b)$$
   for $t \in [T]$
4. Small γ values:
   We have that $(1 - \gamma|S|) > 0$ which implies $\gamma|S| < 1$.
5. **Bounds on action distances**: $\max_{a \in A, b \in A} d_A(a,b) \leq D_A$

With the assumptions as in , we see that we can bound the value functions for the different timesteps. We see that we have the following theorem related to bounds on the value functions, it works as:

Lemma 1. *Under the assumption as in , the bounds for the value function are given by :*

$$||V_{T-t}^{\pi^*}|| \leq A \frac{1 - \gamma^{t+1}}{1 - \gamma}$$

If V_b is the bound for the value functions for all the timesteps $t \in [T]$, then:

$$V_b := A \frac{1}{1 - \gamma} \tag{5}$$

Proof. We recall that the rewards are bounded;
To expand on this, assume that the rewards $||r_t|| \leq A$ for all

$t \in [T]$.
Also note that we manually set:

$$||V_T^{\pi^*}|| = 0 \tag{6}$$

In that case, we have that the value function at time $t-1$ is bounded by A. We would then have that:

$$||V_{T-1}^{\pi^*}|| := |\mathbb{E}[\max_a r_{T-1}(s,a)]| \leq A \tag{7}$$

For the penultimate timestep, we see that the reward

$$||V_{T-2}^{\pi^*}|| \leq \mathbb{E}[r_{T-2} + \gamma P_{T-2}^a V_{T-1}] \leq A + \gamma A$$

If we have the claim that:

$$||V_{T-t-1}^{\pi^*}|| \leq A(1 + \gamma + \gamma^2 + ... + \gamma^t) \leq A\frac{1-\gamma^{t+1}}{1-\gamma} \tag{8}$$

In this case, we can see that if we set:

$$V_b := A\frac{1}{1-\gamma} \tag{9}$$

Then we would be able to derive bounds on the value function for the different timesteps $t \in [T]$. □

Blocked Value Iteration

We develop a procedure called blocked value iteration which enables us to come up with bounds on the cost of interpretability for a **Leader Optimal Regionwise Constant** (LORC) map which is defined above. The theorem below provides bounds on how the cost of interpretability gets modified with the different constants related to the MDP. We see that the bound is a function of the average cost of interpretability over several steps:

Theorem 2. *The cost of interpretability for a leader-optimal regionwise constant policy which belongs to the space* $\mathbf{IP}((h_t)_{t=0}^{T-1})$*, and has the partition* $S_t := \sqcup_{i=1}^{d_t} K_t^i$ *with leader set* $L_t^* := \{k_t^{*l}\}_{l=1}^{d_t}$ *is bounded above by:*

$$\frac{C_r + \gamma V_b |S|^2 C_p}{T} \times \tag{10}$$

$$\sum_{t=0}^{T-1} f_t(h_t, (\pi_i^*)_{i=0}^{T-1}, \{K_t^i\}_{i=1}^{d_t}, \{k_t^{*i}\}_{i=1}^{d_t}) \frac{1}{1-\gamma|S|} \tag{11}$$

In the above, we have the constants being defined as in , which we recall as C_r *is the Lipschitz continuity constant for the reward maps,* $|S|$ *is the size of the state space,* V_b *is an upper bound on the value functions for the different timesteps; recall from later that* $V_b = A\frac{1}{1-\gamma}$*;* C_p *is the Lipschitz bound on the probability transition kernel. For the above we have* $f_t(h_t, (\pi_i^*)_{i=0}^{T-1}, \{K_t^l\}_{l=1}^{d_t}, \{k_t^{*l}\}_{l=1}^{d_t})$ *to be given by:*

$$f_t(h_t, (\pi_i^*)_{i=0}^{T-1}, \{K_t^i\}_{i=1}^{d_t}, \{k_t^{*i}\}_{i=1}^{d_t}) := \tag{12}$$

$$\frac{1}{|S|}\sum_{i=1}^{d_t}\sum_{s\in K_t^l} d_A(\pi_t^*(s), \pi_t^*(k_t^{*i})) \tag{13}$$

The term $f_t(h_t, (\pi_i^*)_{i=0}^{T-1}, \{K_t^i\}_{i=1}^{d_t}, \{k_t^{*i}\}_{i=1}^{d_t})$ *can be bounded in general as the* h_t *function summarization cost for* π^* *at the timestep t; this would be given by* $f_t(\pi^*, h_t)$*; which would be formally defined as:*

$$f_t(\pi^*, h_t) := \min_{\{\{K_t^i\}_{i=1}^{d_t}, \{k_t^{*i}\}_{i=1}^{d_t}\}} f_t(h_t, (\pi_t^*)_{t=0}^{T-1}, \{K_t^i\}_{i=1}^{d_t}, \{k_t^{*i}\}_{i=1}^{d_t}) \tag{14}$$

Proof. Let us assume that $t \in [T]$. Further we assume to start with a partition $\{K_t^i\}_{i=1}^{d_t}$ and a leader set $\{k_t^{*i}\}_{i=1}^{d_t}$. We recall the LORC map characterized by $\{K_t^l\}_{l=1}^{d_t}$ and $\{k_t^{*l}\}_{l=1}^{d_t}$. If we let π_t^I be the interpretable LORC maps for the different timesteps we then have that the LORC map is given by:

$$\pi_t^I(s) := \pi_t^*(k_t^{*i}) \forall s \in K_t^i \tag{15}$$

After doing this, we obtain the cost of interpretability for a timestep t, for the given partition $\mathcal{K}_t$, and a set of leader elements $L_t^* = [k_t^{*1}, k_t^{*2}, ..., k_t^{*d_t}]$ in the partition which we define as $\mathcal{C}_t(\mathcal{K}_t, L_t^*)$. As in the previous definitions we have a measure ds on our space (which in the finite state space setting is the counting measure). We then define the cost of interpretability for the partition and the leader set as:

$$C(\{\mathcal{K}_t\}_{t=0}^{T-1}, \{L_t^*\}_{t=0}^{T-1}) := \frac{1}{T|S|}\sum_{t=0}^{T-1}\sum_{s\in S_t} |V_t^{\pi^*}(s) - V_t^{\pi^I}(s)| \tag{16}$$

The above theorem on the cost of interpretability shows that for a regionwise constant policy, we can bound the cost of using an interpretable policy as a function of Lipschitz parameters describing the MDP and the function summarization cost over the different timesteps.

In the next lemma we relate the cost of interpretability for the different timesteps in a recursive manner.

Lemma 3. *The cost of interpretability for the different timesteps are related to each other in the following manner:*

$$C_t(\mathcal{K}_t, L_t^*) \leq (C_r + \gamma V_b |S|^2 C_p)\frac{\sum_{l=1}^{d_t}\sum_{s\in K_t^l} d_A(\pi_t^*(s), \pi_t^*(k_t^{*l}))}{|S|} \tag{17}$$

$$+\gamma|S| C_{t+1}(\mathcal{K}_{t+1}, L_{t+1}^*) \tag{18}$$

We highlight the proof of the above lemma in the appendix. Next, we have the following lemma which enables us to bound recursive equations such as the one that relates the costs for the different timesteps.

Lemma 4. *For* $t \in [T-1]$*, we solve the recursive set of equations given by:*

$$c_t \leq a_t + \alpha c_{t+1} \tag{19}$$

$$c_{T-1} \leq D \tag{20}$$

We claim that this implies

$$c_{T-k} \leq \sum_{i=0}^{k-1} \alpha^i a_{T-k+i} + \alpha^k D \tag{21}$$

Summing up of the c_{T-k} *values where* $k \in [1, 2.., T-1]$ *gives us the following bound:*

$$\sum_{k=1}^{T-1} c_{T-k} \leq \sum_{n=0}^{T-1} a_n \frac{\alpha^{T-n+1}-1}{\alpha-1} + \frac{1-\alpha^{T-1}}{1-\alpha}\alpha D \tag{22}$$

Proof. The equation [19] gives rise to a recursive set of equations given by:

$$c_t \leq a_t + \alpha c_{t+1} \tag{23}$$

To the above, we know that:

$$c_{T-1} \leq D \tag{24}$$

The final set of equations for a general c_{T-k} is given by:

$$c_{T-k} \leq \sum_{i=0}^{k-1} \alpha^i a_{T-k+i} + \alpha^k D \tag{25}$$

The above equation can be shown inductively. Assume the above holds for a general k, we would seek to extend this to $(k+1)$ using the following logic:

$$c_{T-k-1} \leq a_{T-k-1} + \alpha c_{T-k} \tag{26}$$

The above can then be extended as follows:

$$c_{T-k-1} \leq a_{T-k-1} + \alpha[\sum_{i=0}^{k-1} \alpha^i a_{T-k+i} + \alpha^k D] \tag{27}$$

This can further be extended as:

$$c_{T-k-1} \leq a_{T-k-1} + \alpha[\sum_{i=0}^{k-1} \alpha^i a_{T-k+i} + \alpha^k D] \tag{28}$$

The final form of 25 reads as follows:

$$c_{T-(k+1)} \leq \sum_{i=0}^{k} \alpha^i a_{T-(k+1)+i} + \alpha^{k+1} D \tag{29}$$

This would verify our inductive claim.

Summing up the entries over $k = 1$ to $k = T-1$, we get:

$$\sum_{k=1}^{T-1} c_{T-k} \leq \sum_{k=1}^{T-1} \sum_{i=0}^{k-1} \alpha^i a_{T-k+i} + \sum_{k=1}^{T-1} \alpha^k D \tag{30}$$

We simplify 30 as:

$$\sum_{k=1}^{T-1} c_{T-k} \leq \sum_{i=0}^{k-1} \alpha^i \sum_{k=1}^{T-1} a_{T-k+i} + \frac{1-\alpha^{T-1}}{1-\alpha} \alpha D \tag{31}$$

We now provide an alternate form of writing:

$$\sum_{i=0}^{k-1} \alpha^i \sum_{k=1}^{T-1} a_{T-k+i}$$

.

To do so, we reorder the summation to give us

$$\sum_{k=1}^{T-1} \sum_{i=0}^{k-1} \alpha^i a_{T-k+i} \tag{32}$$

To the above, we provide alternate indices; namely we have $n = T-k+i$ which implies $i = n-(T-k)$;
Here we would have $n \in [1, 2, ..., T-1]$ and $i \in [0, ..., T-n]$. We rewrite the notation to 32 as follows:

$$\sum_{n=1}^{T-1} \sum_{i=0}^{T-n} \alpha^i a_n \tag{33}$$

We bring the n summation outside which gives us the following:

$$\sum_{n=1}^{T-1} a_n \sum_{i=0}^{T-n} \alpha^i \tag{34}$$

Substituting the geometric sum into the expression gives us the following:

$$\sum_{n=0}^{T-1} a_n \frac{\alpha^{T-n+1}-1}{\alpha-1} + \frac{1-\alpha^{T-1}}{1-\alpha} \alpha D \tag{35}$$

□

□

In our setting, we have $c_{T-1} = \frac{1}{|S|} \int_S |V_{T-1}^{\pi^*}(s) - V_{T-1}^{\pi_I^I}(s)| ds$, this implies:

$$c_{T-1} = \frac{1}{|S|} \int_S \left| r_{T-1}(s, \pi_{T-1}^*(s)) - r_{T-1}(s, \pi_{T-1}^I(s)) \right| \, ds \leq \frac{1}{|S|} \int_S C_r \, d_A\big(\pi_{T-1}^*(s), \pi_{T-1}^I(s)\big) \, ds \leq C_r D_A. \tag{36}$$

We apply the above lemma to the cost of interpretability calculation where the a_t and α values are given as follows:

Recall what we want the values of a_t, α, D to be:

$$a_t := (C_r + \gamma C_p V_b |S|^2) \frac{\sum_{l=1}^{d_t} \sum_{s \in K_t^l} d_A\big(\pi_t^*(s), \pi_t^*(k_t^{*l})\big)}{|S|}$$
$$\alpha := \gamma |S|$$
$$D := C_r D_A. \tag{37}$$

The cost of interpretability averaged over different timesteps T is:

$$\frac{C_r + \gamma V_b |S|^2 C_p}{T|S|} \sum_{n=0}^{T-1} \left(\sum_{l=1}^{d_t} \sum_{s \in K_t^l} d_A\left(\pi_n^*(s), \pi_n^*(k_n^{*l})\right) \right) \times \frac{(\gamma|S|)^{T-(n-1)} - 1}{\gamma|S| - 1} + \frac{(\gamma|S|)^{T-1} - 1}{\gamma|S| - 1} \gamma |S| C_r D_A.$$

In the above we can define:

$$f_n(d_n, \pi^*, \{K_n^l\}_{l=1}^{d_n}, \{k_n^{*l}\}_{l=1}^{d_n}) := \frac{1}{|S|} \sum_{l=1}^{d_n} \sum_{s \in K_n^l} d_A(\pi_n^*(s), \pi_n^*(k_n^{*l})) \tag{38}$$

Plugging this in, the optimization problem is now:

$$\frac{C_r + \gamma V_b |S|^2 C_p}{T} \sum_{n=0}^{T-1} \Big(f_n(d_n, \pi^*, \{K_n^l\}_{l=1}^{d_n}, \{k_n^{*l}\}) \Big) \times \frac{(\gamma|S|)^{T-(n-1)} - 1}{\gamma|S| - 1} + \frac{(\gamma|S|)^{T-1} - 1}{\gamma|S| - 1} \gamma |S| C_r D_A.$$

Since we are free to choose the partition $\{K_n^l\}_{l=1}^{d_n}$ and the leaders $\{k_n^{*l}\}_{l=1}^{d_n}$ we pick the partition and leader sets that give the tightest bounds on the LHS. In particular we would have $f_n(d_n, \pi^*) :=$

$\min_{\{K_n^l\}_{l=1}^{d_n},\{k_n^{*l}\}_{l=1}^{d_n}} f_n(d_n, \pi^*, \{K_n^l\}_{l=1}^{d_n}, \{k_n^{*l}\}_{l=1}^{d_n})$. Here $f_n(d_n, \pi^*)$ can be thought of as the minimal error if we had to approximate the policy in d_n regions where the actions are determined by what occurs on the 'leaders' in the above partition. Since the bounds from 39 was valid for any partition and leader set, this will continue to hold true even in the case where the bounds are tight. This gives us :

$$\leq \frac{C_r + \gamma V_b |S|^2 C_p}{T} \sum_{n=0}^{T-1} f_n(d_n, \pi^*) \frac{1}{1-\gamma|S|} + \frac{\gamma|S|C_r D_A}{1-\gamma|S|} \tag{39}$$

In the above, we see that if $\gamma|S| < 1$, we can pull out the $(1-(\gamma|S|)^{T-n})$ term which is less than 1 and we pull out the fraction from below to get the following upper bound:

$$\leq \frac{C_r + \gamma V_b |S|^2 C_p}{(1-\gamma|S|)} \sum_{n=0}^{T-1} \frac{f_n(d_n, \pi^*)}{T} + \frac{\gamma|S|C_r D_A}{1-\gamma|S|} \tag{40}$$

If we assume that f_n is the function summarization cost for timestep n, we can assume that $\sum_{n=0}^{T-1} \frac{f_n(d_n,\pi^*,\{K_l^n\}_{l=1}^{d_n},\{k_l^{*n}\}_{l=1}^{d_n})}{T}$ is the average function summarization cost for the optimal policies at the different timesteps. □

Computational Experiments

In this section, we design computational experiments to compute the cost of interpretability for the 'Taxi' reinforcement learning environment. We compute the necessary hyperparameters to make this computation and we employ the function summarization technique from (Bergerhoff, Weickert, and Dar 2019b). We perform experiments to find the optimal environmental parameters that minimizes the function summarization cost. Further we check how the cost of interpretability behaves with the number of allowable regions in the partition k. The steps we employ in this procedure is given by:

1. Compute the optimal policy using a model-based reinforcement learning method such as value iteration.
2. Compute the Lipschitz constant, value bounds, action space bounds C_r, C_p, V_b, D_A.
3. Compute the function summarization cost of the optimal policy. In $1-D$ environments we do this using Bergerhoff's method.
4. We wish to measure the effectiveness of the obtained bounds on the cost of interpretability. We do this by coming up with other regionwise constant maps and measuring how far they are from the optimal policy.

In the next part, we list the experimental setup, procedure, and results for the *Taxi* environment from Gym. In the appendix section, we present the cost of interpretability analysis for the $2-D$ grid setup. We also present the analysis in the appendix for the *Cliffwalking* environment.

Taxi Environment

We begin this subsection by giving a description of the environment, a description of the experiments, and the results obtained.

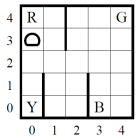
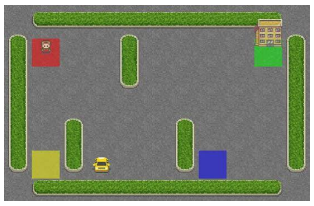


Figure 1: The Taxi Domain



(a) Taxi domain. (b) Taxi environment.

Figure 1: Taxi domain and reinforcement learning environment.

Environment Description The Taxi Problem involves navigating to passengers in a grid world, picking them up and dropping them off at one of four locations as shown in 1. There are four designated pick-up and drop-off locations (Red, Green, Yellow and Blue) in the grid world. The taxi starts off at one of the 25 squares in the 5×5 grid, and the passenger starts off at one of the 4 designated locations.

The goal is move the taxi to the passenger's location, pick up the passenger, move to the passenger's desired destination, and drop off the passenger. Once the passenger is dropped off, the episode ends.

The player receives positive rewards for successfully dropping-off the passenger at the correct location. Negative rewards for incorrect attempts to pick-up/drop-off passenger and for each step where another reward is not received. The environmental parameters are given as follows:

1. S : In the reinforcement learning environment, we have a 5×5 grid present. A passenger can be located at 5 different locations. We have 4 possible choices for the destination. Thus the state space has a size of $25 \times 5 \times 4 = 500$.
2. ρ_0 : The distribution of the initial state. We sample the initial state uniformly from 25 taxi positions, 4 passenger locations (excluding inside the taxi) and 3 destinations. This means the probability of sampling a certain state is $1/300$.
3. The reward maps $r_t \forall t \in [T]$ are as follows:
   (a) $+20$ if we deliver the passenger to the destination.
   (b) -10 executing pickup and dropoff actions illegally.
   (c) -1 if we don't execute the first 2 actions.
4. The action space is enumerated with variables $0-5$, where $0-3$ denotes the up, down, left, right coordinates. The actions 4 and 5 denote pickup and dropoff of the passenger. The action space is endowed with a $0-1$ metric d_A, such that :
$$d_A(a,b) = \begin{cases} 1, & \text{if } a \neq b, \\ 0, & \text{if } a = b. \end{cases} \tag{41}$$
5. The episode ends when the taxi drops off the passenger or when the terminal time reaches $T = 200$.

We proceed to calculate the environmental parameters which is necessary in the calculation of the cost of interpretability.

Parameter calculation In this subsubsection we present calculations on various parameters needed to compute the cost of interpretability.

1. We now provide the computation of the reward Lipschitz constant C_r, probability Lipschitz C_p, value upper bound V_b, and the reward upper bound A, and action Lipschitz bound D_A.

   The C_r coefficient calculation for the above works as follows :

   $$C_r := \max_{s,a,b} \frac{|r_t(s,a) - r_t(s,b)|}{d_A(a,b)} \tag{42}$$

   For the taxi-environment, we see that $\max_a |r_t(s,a) - r_t(s,b)|$ is attained for $+20 - (-10) = 30$ and the minimum distance between the actions $d_A(a,b) = 1$; so $C_r = 30$.

   Recall the definition of C_p which is similarly defined as:

   $$C_p := \max_{s,s',a,b} \frac{|P_t(s'|s,a) - P_t(s'|s,b)|}{d_A(a,b)} \tag{43}$$

   Since the probability transition matrices are deterministic we have that:

   $$|P_t(s'|s,a) - P_t(s'|s,b)| \leq 1 \tag{44}$$

   We also note that $d_A(a,b) = 1$; so we finally see that $C_p = 1$.

   We then have that $V_b := \frac{A}{1-\gamma}$, computation of V_b with $|A| = 20$ and $\gamma = 1/1000$ which gives us:

   $$V_b := 20.02 \tag{45}$$

   We have the action Lipschitz bound D_A to be given by:

   $$D_A := \max_{a \in A, b \in A} \left[d_A(a,b) \right] = 1 \tag{46}$$

   The computation for D_A works by taking $[0,1]$ and $[1,0]$ for $a, b \in A$.

   With the above knowledge, we can upper bound the loss in performance of the policy by C_{int}:

   $$C_{int} := \frac{C_r + \gamma V_b |S|^2 C_p}{(1-\gamma|S|)} \sum_{n=0}^{T-1} \frac{f_n(d_n, \pi^*)}{T} + \frac{\gamma |S| C_r D_A}{1-\gamma|S|} \tag{47}$$

   In this scenario, the interpretability constant is given by 834.4. The hyperparameter values are described more effectively in the next table:

Hyperparameter	Value
γ	0.001
C_r	30.0
C_p	1.0
$\|A\|$	20
D_A	1
V_b	20.02
$\|S\|$	500.0

Table 1: List of hyperparameters and their values for the Taxi environment.

   We calculate the optimal policies using the model-based value iteration method. We would then proceed to approximate the optimal policy using Bergeroff's approximation method.

2. We now perform Bergeroff's approximation procedure from (Bergerhoff, Weickert, and Dar 2019b) to compute the piecewise constant approximation for the final policy. The approximation is such that we need to compute $(x_i)_{i=0}^{N-1}, (u_i)_{i=0}^{N-1}$ which minimizes the following equation:

   $$\sum_{i=0}^{N-1} e_i := \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} (f(y) - u_i)^2 dy \tag{48}$$

   Assuming first order differentiability in the f map, we compute $(x_i)_{i=0}^{N-1} \forall i \in [N]$ values by solving:

   $$\int_{x_i}^{x_{i+1}} (f'(y))^{\frac{2}{3}} dy \approx \frac{1}{N} \int_a^b (f'(y))^{\frac{2}{3}} dy := m \tag{49}$$

   With the knowledge of the $(x_i)_{i=0}^{N-1}$, we compute the u_i values which occur as the means of f over the interval $[x_i, x_{i+1}], \forall i \in [N]$. In particular we get:

   $$u_i := \frac{1}{(x_{i+1} - x_i)} \int_{x_i}^{x_{i+1}} f(y) dy \tag{50}$$

   We cast 49 as the following algorithmic problem : Given a list $a[0, 1, ..., M-1]$ of M numbers which are greater than or equal to 0, can we find a partition $\sqcup_{i=0}^{N} [x_i, x_{i+1}]$ of $[0, 1, 2, ..., M-1]$ such that :

   $$\sum_{i=0}^{N-1} \left| \sum_{j=x_i}^{x_{i+1}} a[j] - \frac{1}{N} \sum_{i=0}^{M-1} a[j] \right| \tag{51}$$

   In our setting, we have $a[0, 1, ..., M-1] = (df)^{2/3}(y_i)$ for all $y_i \in Lin(a_i, b_i, \delta_i)$. We solve 51 by using the following greedy heuristic. At each timestep, it suffices to check which interval $a[0]$ belongs to. We evaluate the index i for which $\sum_{j=0}^{i} a[j]$ is greater than $\frac{1}{N} \sum_{j=0}^{M-1} a[j]$. We proceed to consider two cases for the problem - one where we consider indices $0 - i$, and the other where we consider indices $0 - i - 1$ as the first interval. The game would be to find i_2 such that $a[i+1] + a[i+2] + .. + a[i_2] > 1/N \sum_{j=0}^{M-1} a[j]$ (when we choose i in the first interval) and $a[i] + a[i+1] + .. + a[i_2] > 1/N \sum_{j=0}^{M-1} a[j]$ (when we leave out i in the first interval). In each of the recursive cases, we keep track of the error terms $\left| \sum_{j=x_i}^{x_{i+1}} a[j] - \frac{1}{N} \sum_{i=0}^{M-1} a[j] \right|$ and output the set of intervals that minimizes the total error $\sum_{i=0}^{N} \left| \sum_{j=x_i}^{x_{i+1}} a[j] - \frac{1}{N} \sum_{i=0}^{M-1} a[j] \right|$

   For the 1 dimensional case, more details on this approximation scheme can be found in the appendix.

We compute the function summarization cost when we vary the N and δ parameters. The table is given in Table 2. Since we average out the function approximation errors by dividing with the N factor, the function summarization costs seem to be independent of N. The cost also seems to lower with stepsize since the number of elements that are δ linearly spaced from a to b becomes lesser.

N	δ	function_summarization_cost
5	0.1	0.332954423
5	0.2	0.328848039
5	0.5	0.321677418
5	1	0.297453226
10	0.1	0.332954423
10	0.2	0.328848039
10	0.5	0.321677418
10	1	0.297453226
20	0.1	0.332954423
20	0.2	0.328848039
20	0.5	0.321677418
20	1	0.297453226
30	0.1	0.332954423
30	0.2	0.328848039
30	0.5	0.321677418
30	1	0.297453226
40	0.1	0.332954423
40	0.2	0.328848039
40	0.5	0.321677418
40	1	0.297453226

Table 2: Optimal policy summarization cost for varying N and δ for Taxi environment

3. We perform blocked value iteration for different values of N_{int} - the partition size, and note the average difference between the obtained policy value functions and the optimal value functions for the different timesteps. The error table is given in 3.

Method	N_{int}	**Error** $(\times 10^{10})$
Blocked Value Iteration	50	8.4953
	100	5.6694
	200	5.6634
	300	4.5377
	400	1.3257
	500	0.0000

Table 3: Errors in the cost of interpretability for BVI policies in the Taxi-v3 environment for $\delta_{int} = 0.2$.

Conclusion

In this paper, we came up with a notion of interpretability for a Markov Decision Process. We assumed a general definition of region-wise constant maps which could potentially give us flexibility as to how we choose the regions or do the partitioning of the space. We then use a blocked value iteration approach in the presence of a leader set to derive bounds on the cost of interpretability. We do this by performing value iteration on the leaders and under the assumption that every element of the subset follows the action proposed by the leaders.

This gives us region-wise constant maps where we obtain policies that take the same action on every element of the partition. Further, we show that we can bound the cost of interpretability as the region-wise constant approximation on the final policy. This leads us to design tree-based splits to perform the optimization which ensures we get a regionwise constant final policy.

We design experiments to compute the cost of interpretability for the Taxi reinforcement learning environments. Further we design tree-based splitting routines to check how the cost of interpretability scales with the dimension and length of the state space. The details and figures obtained from the following can be found in the sheet submitted as part of the supplementary section.

References

Bergerhoff, L.; Weickert, J.; and Dar, Y. 2019a. Algorithms for Piecewise Constant Signal Approximations. In *2019 27th European Signal Processing Conference (EUSIPCO)*, 1–5.

Bergerhoff, L.; Weickert, J.; and Dar, Y. 2019b. Algorithms for Piecewise Constant Signal Approximations. arXiv:1903.01320.

Dar, Y.; and Bruckstein, A. M. 2019. On high-resolution adaptive sampling of deterministic signals. *Journal of Mathematical Imaging and Vision*, 61(7): 944–966.

Givan, R.; Dean, T.; and Greig, M. 2003. Equivalence notions and model minimization in Markov decision processes. *Artificial Intelligence*, 147(1): 163–223. Planning with Uncertainty and Incomplete Information.

Grand-Clement, J.; Goh, Y. H.; Chan, C.; Goyal, V.; and Chuang, E. 2021. Interpretable machine learning for resource allocation with application to ventilator triage. *arXiv preprint arXiv:2110.10994*.

Hein, D.; Udluft, S.; and Runkler, T. A. 2018. Interpretable policies for reinforcement learning by genetic programming. *Engineering Applications of Artificial Intelligence*, 76: 158–169.

Petrik, M.; and Luss, R. 2016. Interpretable policies for dynamic product recommendations. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'16, 607–616. Arlington, Virginia, USA: AUAI Press. ISBN 9780996643115.

Steinberg, D. 2009. CART: Classification and Regression Trees.

Zhang, Y.; Laber, E. B.; Tsiatis, A.; and Davidian, M. 2016. Interpretable Dynamic Treatment Regimes. arXiv:1606.01472.

Zhu, R.; Zeng, D.; and Kosorok, M. R. 2015. Reinforcement Learning Trees. *Journal of the American Statistical Association*, 110(512): 1770–1784. PMID: 26903687.